Control, Gates, and Error Suppression with Hamiltonians in Quantum Computation

by

Adam Darryl Bookatz

Submitted to the Department of Physics
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Physics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2016

Author .... Signature redacted ....
Department of Physics
May 12, 2016

Certified by.......... Signature redacted ....
Edward Farhi
Cecil and Ida Green Professor of Physics;
Director, Center for Theoretical Physics
Thesis Supervisor

Accepted by.............. Signature redacted ......
Nergis Mavalvala
Curtis and Kathleen Marble Professor of Astrophysics
Associate Department Head for Education, Physics

Control, Gates, and Error Suppression with Hamiltonians in Quantum Computation

by

Adam Darryl Bookatz

Submitted to the Department of Physics
on May 12, 2016, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Physics

Abstract

In this thesis we are primarily interested in studying how to suppress errors, perform simulation, and implement logic gates in quantum computation within the context of using Hamiltonian controls. We also study the complexity class QMA-complete.

We first investigate a method (introduced by Jordan, Farhi, and Shor) for suppressing environmentally induced errors in Hamiltonian-based quantum computation, involving encoding the system with a quantum error-detecting code and enforcing energy penalties against leaving the codespace. We prove that this method does work in principle: in the limit of infinitely large penalties, local errors are completely suppressed. We further derive bounds for the finite-penalty case and present numerical simulations suggesting that the method achieves even greater protection than these bounds indicate.

We next consider the task of Hamiltonian simulation, i.e. effectively changing a system Hamiltonian to some other desired Hamiltonian by applying external time-dependent controls. We propose protocols for this task that rely solely on realistic bounded-strength control Hamiltonians. For systems coupled to an uncontrollable environment, our approach may be used to perform simulation while simultaneously suppressing unwanted decoherence.

We also consider the scenario of removing unwanted couplings in many-body quantum systems obeying *local* system Hamiltonians and local environmental interactions. We present protocols for *efficiently* switching off the Hamiltonian of a system, i.e. simulating the zero Hamiltonian, using bounded-strength controls. To this end, we introduce the combinatorial concept of balanced-cycle orthogonal arrays, show how to construct them from classical error-correcting codes, and show how to use them to decouple n-qudit ℓ-local Hamiltonians using protocols of length at most $O(n^{\ell-1}\log n)$.

We then present a scheme for implementing high-fidelity quantum gates using a few interacting bosons obeying a Bose-Hubbard Hamiltonian on a line. We find high-fidelity logic operations for a gate set (including the CNOT gate) that is universal for quantum information processing.

Lastly, we discuss the quantum complexity class QMA-complete, surveying all known such problems, and we introduce the "quantum non-expander" problem, proving that it is QMA-complete. A quantum expander is a type of rapidly-mixing quantum channel; we show that estimating its mixing time is a co-QMA-complete problem.

Thesis Supervisor: Edward Farhi
Title: Cecil and Ida Green Professor of Physics; Director, Center for Theoretical Physics

Acknowledgements

First and foremost, I would like to thank my wife, Josepha, for her steadfast support and encouragement and for cheerfully putting up with having a graduate student for a husband.

I am especially grateful to my advisor, Eddie Farhi, for being a great thesis supervisor, giving me both direction and freedom, as well as exemplifying how to write academic papers with precision and clarity. I also especially thank Pawel Wocjan, for being like a second advisor to me, sharing much wisdom – in science, career, and life – and working on many projects with me. In addition, I greatly appreciate the help of all of my collaborators, Yoav Lahini, Martin Roetteler, Leo Zhou, Greg Steinbrecher, Stephen Jordan, Yi-Kai Liu, Lorenza Viola, and Dirk Englund. Thank you to my thesis committee members, Edward Farhi, Aram Harrow, and Boleslaw Wyslouch for carefully reading and checking my thesis, thereby comforting me that at least someone has read it.

I feel privileged to have been able to attend MIT, with its strong series of quantum information courses, and I thank my teachers, Professors Isaac Chuang, Scott Aaronson, and Peter Shor for their classes from which I benefited greatly. The help, advice, and discussions with MIT postdocs and students has also been invaluable, and I want to thank Shelby Kimmel, Cedric Lin, Han-Hsuan Lin, David Gosset, Lior Eldar, Kristen Temme, Iman Marvian, as well as professors Barton Zwiebach, Aram Harrow, Seth Lloyd, and Sam Gutmann. I thank the many people of the faculty and staff of the CTP, the Department of Physics, and MIT who have helped me and taught me, and I also thank the MIT Rowing Club for helping me ward off physical atrophy long enough to write this thesis.

Lastly, I thank my parents, Brian and Sandra, for their continuous support throughout my entire life, and my brothers, David and Gidon, for being great brothers and keeping me suitably distracted when appropriate; I am truly fortunate to have such a family.

תושלב״ע

Contents

Chapter 0

Thesis introduction

Quantum computing proposes the exciting possibility of exploiting the ‘weirdness’ of quantum mechanics to develop a new paradigm of computing. While quantum computers are widely believed to be more powerful than classical computers for certain important tasks (such as factoring and unstructured searches), the question of what computational power practical quantum computation yields remains open. Moreover, it appears that the technology of scalable quantum computers is still beyond the near-future, so studying how to overcome the obstacles that impede building even small-scale quantum computers is a crucial topic to the field. This thesis addresses, in part, both of these questions.

In classical computing, by which we mean regular, non-quantum computing, the basic unit of information is the bit – an object that can be in precisely one of two states, 0 or 1. If we have n bits, each of them is either 0 or 1, so a system of n bits can always be specified by n parameters – one simply records the value of each bit. Note that there are 2^n possible n-bit values. The quantum setting is much richer. And, to the best of our knowledge, it is also fundamentally how our world really operates. Quantum mechanics allows for two-state objects (e.g. an electron’s spin), so like a bit, such a system can be 0 or 1. But unlike a bit, the quantum state need not be only 0 or 1; it can be in a *superposition* of both 0 and 1 “at the same time”. Such a quantum bit is called a *qubit*. However, the fundamental difference between classical and quantum computing is seen not from a single qubit but from many qubits. For with n qubits, each of the 2^n possible n-bit values are now merely the basis states allowed in the superposition. Specifying a system of n qubits involves, in general, keeping track of all 2^n coefficients of each of those basis states, i.e. “how much” of the state is in each of the 2^n n-bit values. This is not to say that quantum computers can yield exponentially more information than classical computers. The information contained in these coefficients is not immediately accessible. To obtain information from the state, one must measure it, and when one measures a quantum state, the superposition collapses, giving just one of the 2^n classical bit strings. Indeed, someone with a background in stochastic (random) processes may not think this quantum picture, as described so far, sounds particularly powerful – one may think these coefficients represent mere probabilities, making a collection of n qubits no different from a probabilistic mixture of n bits. But unlike probabilities, these coefficients are not non-negative numbers; they can be negative (or even complex), so that when processing quantum information, components of the superposition may undergo complicated cancellations (interference) that cannot occur with random bits.

If this sounds complicated, it is because it is. We still have a long way to go in understanding how to harness the power of the quantum world for use in computation. Much progress has been made in the past three decades. Of particular interest are two algorithms – Grover's algorithm and Shor's algorithm – that have attracted a lot of attention to the field of quantum computing. Grover's algorithm allows a quantum computer to search for a marked item amongst N items, taking only time proportional to $\sqrt{N}$ to do so (as opposed to the classical scenario, where such a search could require looking at all N items and therefore takes time proportional to N). Shor's algorithm allows a quantum computer to factor integers ($15 = 5 \times 3$ being a very simple example). This problem is believed to be difficult for regular computers – one can factor $15 = 5 \times 3$ in their head, but even with a modern supercomputer, factoring a 400-digit number seems to be extremely difficult. In fact, our confidence in the difficulty of this problem has led to it being widely used in cryptography: currently, sending sensitive information (like your credit card) over the internet relies on techniques like RSA encryption to keep it secret, and the security of RSA relies solely on the difficulty of factoring large numbers. Shor's algorithm shows that a quantum computer could factor integers efficiently and break RSA encryption easily.

Although there is much interest in the field of quantum computing, the goal of actually building a large-scale quantum computer currently remains beyond reach. One of the greatest obstacles to accomplishing this goal is overcoming errors due to environmental disturbance. A quantum system with unwanted interactions from an uncontrollable environment will decohere, i.e. lose much of the extra quantum information held within the superposition states. Practically, attempting to isolate the system from the environment has limitations – after all, the device is meant to be a computer, necessitating control and measurement via external interaction. A major push in the field of quantum computing has therefore been towards suppressing and correcting errors arising during computation, and much progress has been made, at least theoretically. Indeed, in the most common model of quantum computing – the circuit model – it has been shown that, provided error rates are below some constant value, arbitrarily-accurate quantum computation can, in principle, be performed efficiently.

The circuit model describes quantum computation in terms of instantaneous operations (gates), similar to the notion of logical AND, OR, NOT, and NAND gates in classical circuits. It is not, however, the only model of quantum computation. It is also somewhat of an idealization, as it treats these gates as being implemented instantaneously without dealing with the underlying physical machinery performing the operations. In quantum physics, systems are generally described by *Hamiltonians* – operators defining the energy of the system and responsible, via the Schrödinger equation, for their evolution. There are a number of models of quantum computation described in terms of Hamiltonians, including the original vision of quantum computation by Richard Feynman, who is often credited with pioneering the field, as well as the model of *adiabatic quantum computation*, upon which the widely-publicized (but non-universal) D-Wave architecture is based. Such models, which we term *Hamiltonian-based quantum computation* in this thesis, involve specifying the Hamiltonian of a system, evolving the system in time under this Hamiltonian (with no intermediate measurements or instantaneous operations allowed), and performing a final measurement. We will primarily focus on such a setting in this thesis, where systems, and our control over them, are restricted to (non-instantaneous) Hamiltonian evolution. Within this context:

- we analyse an error suppression technique for Hamiltonian-based quantum computation, proving that it works in principle;
- we develop protocols to efficiently eliminate a system's unwanted internal and system-

environment couplings;

- we develop protocols for Hamiltonian simulation (i.e. having a system defined by one Hamiltonian behave as though it were defined by a different Hamiltonian), again suppressing unwanted environmental disturbance; and
- we develop realizable or near-future-realizable Hamiltonians for implementing useful quantum gates.

Another topic of interest when studying Hamiltonian models of quantum computing is what computational power they provide. Historically, there has been a strong connection between this question (at least for adiabatic quantum computing) and the study of the quantum complexity class called *QMA-complete*. In this thesis,

- we discuss what problems are known to be in this class and
- we prove that it contains a particular problem involving so-called *quantum expanders*, which are, among other things, related to the thermalization of quantum systems coupled to an environment.

In the next section, we elaborate on these points, outlining the content that comprises this thesis.

0.1 Outline

Each chapter in this thesis is self-contained, with its own bibliography and appendices (if applicable) at the end of the chapter. First, in **Chapter 1**, we provide much of the background required for the remainder of the thesis, presenting a broad overview of the topics needed to understand the later material. More technical background will be introduced in the individual chapters as needed, as will references for further study. We now outline the contents of these chapters (with references to the applicable background sections given in parentheses).

As noted, a challenging obstacle towards building quantum computers is protecting them from unwanted environmental disturbance. In the usual circuit model of quantum computation (Sec. 1.3), the theory of quantum error correction has been well-developed, suggesting that, in principle, quantum computation can be performed in a manner resistant to such environmental disturbance; however, the question of how well Hamiltonian-based quantum computation models (Sec. 1.2) can be protected from error remains open. One proposal (introduced by Jordan, Farhi, and Shor) for suppressing environmentally induced errors in Hamiltonian-based quantum computation is the use of quantum error-detecting codes (Sec. 1.5.2), together with energy penalties against leaving the codespace. In **Chapter 2** we prove that this method does work in principle: in the limit of infinitely large penalties, errors are completely suppressed. We also derive bounds for the finite-penalty case and perform numerical simulations that suggest that the energy penalty method achieves even greater protection than these bounds guarantee.

Related to the goal of the previous paragraph are Hamiltonian decoupling and Hamiltonian simulation, in the context of quantum control theory. In Hamiltonian decoupling, one seeks to switch off the Hamiltonian of a system, removing unwanted internal and system-environment couplings (the latter being responsible for decoherence errors). More generally, in Hamiltonian simulation one is interested in removing these unwanted couplings while simultaneously effectively changing the system Hamiltonian to simulate a different, desired Hamiltonian. These tasks would be useful, for example, for quantum memories and analogue quantum simulators (Sec. 1.2.4), respectively, while addressing the ever-important objective of suppressing environmental errors. In contrast to many previous protocols for these tasks that rely on using instantaneous unitary pulses (which, in terms of Hamiltonians, require controls of unbounded strength, and are therefore fundamentally unphysical), we are interested in the setting in which all controls can be described using bounded-strength Hamiltonians. In **Chapter 3** we develop a control protocol for Hamiltonian simulation that uses only bounded-strength controls by combining a Hamiltonian simulation scheme that uses controls of unbounded-strength, with a bounded-strength decoupling scheme called Eulerian decoupling.

Another issue for general-purpose decoupling and simulation protocols is that they are often inefficient, requiring very long control sequences. By making physically-reasonable assumptions about the quantum system of interest, however, we can hope to devise better protocols. In **Chapter 4** we develop *efficient* bounded-strength decoupling protocols for *local Hamiltonians*, i.e. for quantum systems in which each qubit interacts with only a few other qubits, as is typically the case in nature (Sec. 1.1.4). To do so, we introduce the combinatorial concept of balanced-cycle orthogonal arrays, demonstrate how to construct them from classical error-correcting codes (Sec. 1.5.1), and show how to exploit the locality of the system with them to perform decoupling more efficiently.

While the previous topics address Hamiltonian-based quantum computation and Hamiltonian simulation, the most common model of quantum computation is the circuit model (Sec. 1.3), in which computation is described by a sequence of unitary operations (gates). As noted, this model is somewhat of an idealization, ignoring the question of how these unitary operations are to be performed. Nonetheless, in a real physical system actualizing this computation, each step is generally performed by implementing some corresponding Hamiltonian. Designing a physically-realizable Hamiltonian to implement a desired unitary gate, while obeying the constraints of the physical platform available, is in general a non-trivial problem. Inspired by continuous-time quantum walks (Sec. 1.2.3), in **Chapter 5** we present a method for implementing high-fidelity quantum logic gates using interacting bosons on a one-dimensional lattice (Sec. 1.1.6). Specifically, constraining ourselves to experimentally feasible system parameters, we present high-fidelity logic operations for a gate set, including the CNOT gate (Sec. 1.3.2), that is universal for quantum information processing.

Aside from the important question of how to practically implement quantum computation, another important question is what computational power the quantum setting provides. This question is addressed by the field of quantum complexity theory (Sec. 1.4.2). In this thesis, we are primarily interested in the quantum complexity class known as QMA-complete, the quantum analogue of the classical class NP-complete. Informally, QMA-complete consists of the problems whose solutions are believed to be hard to find – but easy to verify – using a quantum computer. The development of the QMA-complete class and its most famous member, the LOCAL HAMILTONIAN problem (Sec. 1.4.3), has been highly connected to analysing the power of using local Hamiltonians for Hamiltonian-based computation methods, notably that of *adiabatic quantum computation* (Sec. 1.2.1). A survey of all known QMA-complete problems is the content of **Chapter 7**.

Adding to the list of known QMA-complete problems, in **Chapter 6** we classify the complexity of the QUANTUM NON-EXPANDER problem as being QMA-complete. Quantum expanders are the quantum analogues of expander graphs (which play a prominent role in computer science and discrete mathematics), and are related to the thermalization of open quantum systems. They are quantum operations (Sec. 1.1.5) that rapidly take quantum states towards the maximally mixed state (Sec. 1.1.5), i.e. they always add entropy to states that are far from totally random. We show that checking whether a given quantum operation is a poor quantum expander is a QMA-complete problem. Aside from its applications to physics, the result is interesting because QMA (unlike its classical counterpart) has relatively few known complete problems aside from LOCAL HAMILTONIAN problems.

Chapter 1

Background material

This chapter is primary tasked with reviewing the basics of quantum mechanics, classical and quantum computing, computational complexity theory, and error-correcting codes. Although we will briefly review some basics of quantum mechanics, this thesis assumes the reader is comfortable with linear algebra and the Dirac ket notation of quantum physics. We will also use, without much background explanation, some very basic terminology (and occasionally facts) about groups, graphs, finite fields, and representation theory, but thorough background in these topics is certainly not required. The majority of the information present in this chapter can be found, with much greater thoroughness, in the excellent textbook by Nielsen and Chuang [1]. The notes of Preskill [2] is also an excellent recommended resource for the interested reader.

1.1 Quantum mechanics

We start our background material by reviewing the postulates of quantum mechanics as they will pertain to quantum computing and the content later in this thesis.

1.1.1 Quantum states

To any (isolated) quantum system is associated a Hilbert space $\mathcal{H}$ (for our purposes, a vector space endowed with an inner product), known as its *state space*. The state of the quantum system is a normalized vector in this space. The simplest non-trivial system has the state space $\mathbb{C}^2$, describing a single *qubit*, whose states can be written as

$$|\psi\rangle = c_0|0\rangle + c_1|1\rangle = c_0 \begin{pmatrix} 1 \\ 0 \end{pmatrix} + c_1 \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

where $|0\rangle$ and $|1\rangle$ are the standard basis vectors, written in Dirac ket notation, and $c_0, c_1 \in \mathbb{C}$ are complex coefficients (also called amplitudes) satisfying $|c_0|^2 + |c_1|^2 = 1$. This latter condition stems from the normalization condition of $|\psi\rangle$, namely $\langle\psi|\psi\rangle = 1$. The overall phase of the state does not matter: $|\psi\rangle$ and $e^{i\alpha}|\psi\rangle$ represent the same state and are considered to be *equal up to phase*.

Note that a qubit is a 2-dimensional system, with two basis vectors ($|0\rangle$ and $|1\rangle$), and may naturally represent the state of a single 2-level particle, e.g. the spin of a single electron (with $|0\rangle$ representing *spin up* and $|1\rangle$ representing *spin down*). One can also speak more generally of d-dimensional systems, which are called *qudits*. In this case, the state space is $\mathbb{C}^d$. A

d-dimensional qudit may, e.g., represent an atomic system that has d levels of excitation available. Quantum systems can also be infinite-dimensional, such as those describing a particle travelling in continuous space; however, in quantum computing we are generally interested in finite-dimensional systems and in this thesis we will implicitly assume that all systems are finite-dimensional unless otherwise noted.

1.1.2 Measurement

Let $\{|\phi_1\rangle, |\phi_2\rangle, \ldots, |\phi_D\rangle\}$ be a set of orthonormal basis vectors of the state space. Given a state $|\psi\rangle$, we can *measure* $|\psi\rangle$ in the basis. The result will be one of these basis vectors with some probability. Specifically, we will obtain a result of $|\phi_i\rangle$ with probability

$$p_i = |\langle\phi_i|\psi\rangle|^2 .$$

Note that the state of the system changes by virtue of performing the measurement: it was originally in the state $|\psi\rangle$ but, as a result of the measurement, has changed to the basis vector $|\phi_i\rangle$ that we obtained from our measurement. Only if $|\psi\rangle$ was equal (up to phase) to one of these basis vectors, say $|\phi_j\rangle$, are we guaranteed that the state will not be modified, for in that case, the measurement will result in $|\phi_j\rangle = |\psi\rangle$ with probability $p_j = 1$. Much more general and powerful formulations of measurement exist in quantum mechanics, but they will not be needed in this thesis; we direct the interested reader to [1].

1.1.3 Hamiltonians and evolution

Having specified the state of a quantum system, we would like to understand how that system evolves in time, in order that we may determine the state of the system at any future time.

Hamiltonians

The time-evolution of a quantum system is governed by an linear operator called the Hamiltonian, H, of the system. The Hamiltonian may be time-dependent, in which case we often write $H(t)$, or time-independent, in which case we simply write H. The Hamiltonian is a Hermitian operator, meaning that $H^\dagger = H$, where $\dagger$ is used to denote the conjugate transpose. Since we are primarily interested in finite-dimensional systems, H can be thought of as a matrix such that if one takes its transpose and complex conjugate, one obtains H again.

In addition to governing time evolution, about which we shall elaborate shortly, Hamiltonians also govern the energetics of a system, as they are the operator corresponding to energy. The allowed energy levels $\{E_\alpha\}$ of the system are precisely the eigenvalues of H, with corresponding energy eigenstates $\{|E_\alpha\rangle\}$,

$$H|E_\alpha\rangle = E_\alpha|E_\alpha\rangle .$$

Observe that, at least in discrete systems, the spectrum of allowed energies is restricted, in contrast to the continuum allowed in classical mechanics.

In principle, we can measure the energy of a system, i.e. measure H. Indeed, we can in principle measure any Hermitian operator M. Suppose M has eigenvalues $\{m_\alpha\}$ and eigenvectors $\{|m_\alpha\rangle\}$, which because M is Hermitian, we can take to be an orthonormal basis for the system. One way to consider measuring M for a state $|\psi\rangle$ is to measure $|\psi\rangle$ in the $\{|m_\alpha\rangle\}$ basis to obtain some resulting M-eigenstate $|m_\alpha\rangle$ with probability $|\langle m_\alpha|\psi\rangle|^2$; the *measurement outcome* is then the corresponding eigenvalue m_α. The expectation value is therefore $\sum_\alpha m_\alpha |\langle m_\alpha|\psi\rangle|^2 = \langle\psi| M |\psi\rangle$. Thus, if a state of the system is $|\psi\rangle$, its average energy is

$$E_\psi = \sum_\alpha E_\alpha |\langle E_\alpha|\psi\rangle|^2 = \langle\psi| H |\psi\rangle \ .$$

In the case of qubits, four extremely important Hermitian linear operators (matrices) are the identity

$$\mathbb{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

and the three Pauli matrices,

$$X = \sigma_X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \sigma_Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \sigma_Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

In fact, these four matrices form a basis for the linear operators on $\mathbb{C}^2$. If A is a linear operator acting on qubits then it can be written as a linear combination of the Pauli matrices and the identity,

$$A = a_0 \, \mathbb{1} + a_x X + a_y Y + a_z Z$$

for some complex numbers a_0, a_x, a_y, a_z. We note that A is Hermitian if and only if these coefficients a_0, a_x, a_y, a_z are all real. We also note that A is traceless if and only if $a_0 = 0$. Without loss of generality, we can always take a Hamiltonian to be traceless by shifting the overall energy of the system by $-a_0$, and so we often will decompose a Hamiltonian on a qubit system as

$$H = a_x X + a_y Y + a_z Z = \vec{a} \cdot \vec{\sigma} \quad \text{with} \quad a_x, a_y, a_z \in \mathbb{R} \, ,$$

where $\vec{a} = (a_x, a_y, a_z)$ and $\vec{\sigma} = (X, Y, Z)$.

As a very simple example, one could imagine a simple 1-qubit system operating under the Hamiltonian $H = \omega X$, where ω is a constant with units of energy. The eigenstates of this H are proportional to $|0\rangle \pm |1\rangle$ with energy eigenvalues $\pm\omega$. In principle, any Hermitian operator is eligible to be a Hamiltonian, although the Hamiltonians that arise in nature and experiment are typically of a much more restricted form.

The Schrödinger equation and unitary evolution

Given a Hamiltonian $H(t)$ for a quantum system, the state of the system evolves according to the Schrödinger equation

$$i\hbar \frac{\mathrm{d}}{\mathrm{d}t} |\psi(t)\rangle = H(t) |\psi(t)\rangle$$

where $|\psi(t)\rangle$ is the state of the system at time t, $\hbar$ is the reduced Planck constant, and i is the imaginary unit, $i = \sqrt{-1}$. Note that throughout this thesis, we generally use a unit system in which $\hbar = 1$ and therefore ignore $\hbar$ entirely. The Schrödinger equation implies that if one knows the initial state of a system $|\psi(0)\rangle$ at time $t = 0$, and one knows the

Hamiltonian of the system, one can, in principle, calculate[1] the state $|\psi(t)\rangle$ of the system at any future time t. Indeed, we can write

$$|\psi(t)\rangle = U(t)|\psi(0)\rangle$$

where $U(t)$ is called the *time-evolution operator* (or sometimes the *propagator*). Mathematically, because $H(t)$ is Hermitian, $U(t)$ is a unitary linear operator, meaning that $U^\dagger(t) = U^{-1}(t)$. Explicitly relating a unitary evolution operator and a Hermitian Hamiltonian, however, can be difficult.

If $H(t) = H$ is time-independent, we can solve the Schrödinger equation by exponentiating, obtaining

$$U(t) = e^{-iHt}.$$

For example, for a single qubit, if $H = \omega\,\hat{n}\cdot\vec{\sigma}$ for some real ω, then $U(t) = e^{-i\omega\hat{n}\cdot\vec{\sigma}t} = \cos(\omega t) - i\sin(\omega t)\hat{n}\cdot\vec{\sigma}$, where $|\hat{n}|^2 = 1$. If $H(t)$ is time-dependent, but always commutes with itself at any time, i.e. $[H(t_1), H(t_2)] = 0$ for all t_1, t_2, we can then write

$$U(t) = e^{-i\int_0^t H(\tau)\mathrm{d}\tau}.$$

However, in the general time-dependent case, we cannot write a simple explicit expression for $U(t)$ in terms of $H(t)$. In this case we may use a Dyson series expansion and formally write

$$U(t) = \mathcal{T}\exp\left\{-i\int_0^t H(\tau)\mathrm{d}\tau\right\}$$

where $\mathcal{T}$ denotes the time-ordering operator,

$$\mathcal{T}\{A(t_1)B(t_2)\} = \begin{cases} A(t_1)B(t_2), & \text{if } t_1 > t_2, \\ B(t_2)A(t_1), & \text{if } t_1 < t_2. \end{cases}$$

This expression can also be considered as shorthand for

$$\begin{aligned} U(t) = \mathbb{1} + (-i)\int_0^t \mathrm{d}t' H(t') + (-i)^2\int_0^t \mathrm{d}t'\int_0^{t'} \mathrm{d}t'' H(t')H(t'') + \cdots \\ + (-i)^m\int_0^t \mathrm{d}t'\int_0^{t'} \mathrm{d}t''\cdots\int_0^{t^{(m-1)}} \mathrm{d}t^{(m)} H(t')H(t'')\cdots H(t^{(m)}) + \cdots. \end{aligned}$$

Evidently, it is not easy to calculate the evolution due to a time-dependent Hamiltonian in general, which makes analysing the consequences of modifying the Hamiltonian of a system (as will be done in Chapters 2, 3, and 4) quite challenging.

An alternative expansion for treating the general case is provided by the Magnus expansion. Let us say that we are interested in evaluating $U(T)$ at some fixed time $T \geqslant 0$. We can associate an effective time-independent Hamiltonian $\bar{H}$ to $U(T)$, so that $U(T)$, which is the description of evolving under the time-dependent $H(t)$ for time T, is mathematically equivalent to evolving under the time-independent $\bar{H}$ for the same length of time T. We

[1] This is, of course, assuming that no measurements were made – if the system is at any point measured, the Measurement axiom of Sec. 1.1.2 dictates that the system abruptly changes probabilistically as we saw above. We will evade the (perhaps philosophical) question of whether these two types of evolution can be reconciled in a single framework.

can therefore write

$$U(T) = \exp(-i\bar{H}T).$$

The Magnus expansion allows us to calculate $\bar{H}$ as

$$\bar{H} = \bar{H}^{(0)} + \bar{H}^{(1)} + \bar{H}^{(2)} + \cdots ,$$

where

$$\bar{H}^{(0)} = \frac{1}{T}\int_0^T H(\tau)d\tau ,$$

$$\bar{H}^{(1)} = \frac{-i}{2T}\int_{\tau_1=0}^{T}\int_{\tau_2=0}^{\tau_1} [H(\tau_1), H(\tau_2)]\, d\tau_2 d\tau_1 ,$$

with higher order terms involving more integrals and more complicated commutators. In this thesis we will only need to make use of the first order term, $\bar{H}^{(0)}$, which can be seen to be the average of $H(t)$ over time T. For more information, including implicit formulae for higher order terms, convergence theorems, and bounds on the accuracy of truncating the series, consult [3].

1.1.4 Composite systems

Tensor products and bases

So far, we have described quantum systems acting on some Hilbert space, without worrying about the structure of that Hilbert space. For a single qubit, qudit, or particle with only one degree of freedom, this is often sufficient. However, we will usually be interested in composite systems, composed of multiple particles or representing multiple qubits (or qudits). For this we need *tensor products spaces*, which we very briefly review.

Let V and W be two Hilbert spaces with orthonormal bases $\{|i\rangle\}$ and $\{|j\rangle\}$ respectively. Their tensor product, $V \otimes W$, is the space spanned by $\{|i\rangle \otimes |j\rangle\}$, i.e. consists of vectors that can be written in the form $\sum_{ij} c_{ij}|i\rangle \otimes |j\rangle$. Note that where no confusion will arise, we may omit the tensor product symbol for states, writing $|i\rangle|j\rangle$, or even $|ij\rangle$, to mean $|i\rangle \otimes |j\rangle$. If the state of a system can be written in the form $|\psi_V\rangle \otimes |\psi_W\rangle = \Big(\sum_i a_i|i\rangle\Big) \otimes \Big(\sum_j b_j|j\rangle\Big)$ then we say the state is *separable* or *unentangled*; otherwise, we say it is *entangled*.

Suppose that A and B are linear operators on V and W respectively. Then their tensor product, $A \otimes B$, acts as $(A \otimes B)(\sum_k c_k|v_k\rangle \otimes |w_k\rangle) = \sum_k c_k(A|v_k\rangle) \otimes (B|w_k\rangle)$, where $\{|v_k\rangle\}$ and $\{|w_k\rangle\}$ are vectors in V and W respectively, and $c_k \in \mathbb{C}$. Any linear operator on $V \otimes W$ can be written as a linear combination of tensor products, i.e. of the form $\sum_k c_k A_k \otimes B_k$, which acts as $\Big(\sum_k c_k A_k \otimes B_k\Big)|v\rangle \otimes |w\rangle = \sum_k \Big(c_k(A_k \otimes B_k)(|v\rangle \otimes |w\rangle)\Big)$.

It is convenient to represent states and operators on finite-dimensional tensor product spaces in a matrix form. Suppose $\{|i\rangle : i = 0, \ldots, D_V\}$ and $\{|j\rangle : j = 0, \ldots, D_W\}$ are orthonormal bases of V and W, with dimensions $|V| = D_V + 1$ and $|W| = D_W + 1$ respectively.[2] If $|v\rangle \in V$ and A acts on V, recall that we may write these in matrix form with

[2] Note that, for notational consistency, we start the enumeration at 0 as is typical in computer science, but for notational simplicity here, end it at D_V, so that $|V| = D_V + 1$.

respect to $\{|i\rangle : i = 0, \dots, D_V\}$ as

$$|v\rangle \equiv \begin{pmatrix} \langle 0|v\rangle \\ \langle 1|v\rangle \\ \langle 2|v\rangle \\ \vdots \\ \langle D_V|v\rangle \end{pmatrix}, \qquad A \equiv \begin{pmatrix} \langle 0|A|0\rangle & \cdots & \langle 0|A|D_V\rangle \\ \langle 1|A|0\rangle & \cdots & \langle 1|A|D_V\rangle \\ \langle 2|A|0\rangle & \cdots & \langle 2|A|D_V\rangle \\ \vdots & \vdots & \vdots \\ \langle D_V|A|0\rangle & \cdots & \langle D_V|A|D_V\rangle \end{pmatrix}.$$

Note that we have used the natural ordering

$$|0\rangle, |1\rangle, |2\rangle, \dots, |D_V\rangle$$

when ordering the rows and columns. Now, $\{|i\rangle \otimes |j\rangle : i = 0, \dots, D_V,\ j = 0, \dots, D_W\}$ is an orthonormal basis of $V \otimes W$. To use this in matrix form, we consider these basis vectors to be ordered primarily according to V and secondarily according to W; i.e. the order is

$$|00\rangle, |01\rangle, \dots, |0D_W\rangle, |10\rangle, |11\rangle, \dots, |1D_W\rangle, |20\rangle, \dots, |D_V D_W\rangle.$$

where we have suppressed the $\otimes$ notation. For example, if $V = W = \mathbb{C}^2$ each represented qubits, then we order their tensor product basis as $|00\rangle, |01\rangle, |10\rangle, |11\rangle$; this ordering is consistent with the order of integers represented in binary $(00, 01, 10,$ and $11)$.

Adopting this convention, the $(D_V+1)(D_W+1) \times (D_V+1)(D_W+1)$ matrix form of the tensor product $A \otimes B$ is given by the Kronecker product

$$A \otimes B \equiv \begin{pmatrix} a_{00}B & a_{01}B & \cdots & a_{0D_V}B \\ a_{10}B & a_{11}B & \cdots & a_{1D_V}B \\ \vdots & \vdots & \vdots & \vdots \\ a_{D_V0}B & a_{D_V1}B & \cdots & a_{D_VD_V}B \end{pmatrix}$$

where $a_{k\ell}$ is the (k, ℓ) component of A, i.e. $a_{k\ell} = \langle k|A|\ell\rangle$.

We can define larger multi-qudit spaces similarly; an n-qudit space is the Hilbert space given by $(\mathbb{C}^d)^{\otimes n} = \mathbb{C}^d \otimes \cdots \otimes \mathbb{C}^d$ with n copies of $\mathbb{C}^d$. Of particular importance is the case of n qubits $(d = 2)$. The standard basis, called the *computational basis*, is given by $\{|i\rangle : i \in \{0, 1\}^n\}$, where $\{0, 1\}^n$ denotes the set of all n-tuples of binary numbers. Again, when writing vectors and matrices, the order of these basis elements is in ascending numerical order. For example, for $n = 3$ the computational basis is, in order,

$$\{|000\rangle, |001\rangle, |010\rangle, |011\rangle, |100\rangle, |101\rangle, |110\rangle, |111\rangle\}.$$

Consider a 1-qubit system, $\mathbb{C}^2$. Recall that any linear operator A on this space can always be written as a linear combination of the Pauli matrices and the identity, $A = n_0 \mathbb{1} + n_x X + n_y Y + n_z Z$, for some complex numbers n_0, n_x, n_y, n_z. Moving to the n-qubit case, if A is a linear operator on n qubits, we may write A as a linear combination of the n-fold tensor products of Pauli matrices,

$$A = \sum_{\substack{\sigma_i \in \{\mathbb{1}, X, Y, Z\} \\ i=1,\dots,n}} c_{\sigma_1,\dots,\sigma_n}\, \sigma_1 \otimes \cdots \otimes \sigma_n,$$

where the sum is over all possible choices of $\sigma_i \in \{\mathbb{1}, X, Y, Z\}$ for each qubit $i = 1, \dots, n$

and where $c_{\sigma_1,\ldots,\sigma_n}$ are complex coefficients (or, if A is Hermitian, real coefficients). In general, this sum includes terms in which $\sigma_1 \otimes \cdots \otimes \sigma_n$ act non-trivially on each qubit, i.e. in which none of the σ_i are equal to $\mathbb{1}$. For example, one can imagine a Hamiltonian $H = X \otimes X \otimes \cdots \otimes X$. This Hamiltonian indicates an interaction involving all n qubits at the same time. Indeed, even for very small t, the evolution of the system $U = \mathbb{1} - iHt + O(t^2)$ will include the operation $X \otimes X \otimes \cdots \otimes X$, affecting all of the qubits together. In the next subsection, we shall deal with more restricted operators, in which the number of non-trivial σ_i in any given term $\sigma_1 \otimes \cdots \otimes \sigma_n$ is limited.

Locality

In principle, any Hermitian matrix can serve as the Hamiltonian of a system. In nature (and experiment), however, Hamiltonians generally have certain constraints. In particular, it is typically the case that the interactions between different particles involve only a few particles at a time. This property is called *locality*. We can further classify how local a Hamiltonian is: we say that a Hamiltonian H on n qubits is an *ℓ-local Hamiltonian* if it can be written as

$$H = \sum_k H_k$$

where each H_k acts non-trivially on at most ℓ of the n qubits. That is, H is ℓ-local if we can write it as the sum of tensor products

$$H = \sum_k H_k = \sum_k h_{k,1} \otimes \cdots \otimes h_{k,n}$$

where each $h_{k,i}$ is a Hermitian operator on qubit i alone and where, for each k, at most ℓ of these $h_{k,i}$ are not equal to the identity, i.e. for each k, $|\{h_{k,i} \neq \mathbb{1} : i = 1, \ldots, n\}| \leqslant \ell$.

Obviously, any Hamiltonian on n qubits is ℓ-local for $\ell = n$, but if it is also ℓ-local for some small ℓ, typically a small constant like 2 or 3, then it is considered to be a *local* Hamiltonian. In fact, in nature Hamiltonians are typically 2-local, meaning that only pairwise interactions are present. Due to the ubiquity of locality in nature, locality will be a reoccurring theme in this thesis. We will make locality assumptions in Chapters 2 and 4 to derive stronger results than might be true in the general case. Moreover, as we will discuss in Sec. 1.4.3, the concept of local Hamiltonians features prominently in the story of QMA-completeness, which is studied in Chapters 6 and 7.

Note that in this definition of locality, which is standard in the quantum computing field, there are no constraints on which particle (or qubit) interacts with which other particle – any particle can interact with any other particle, as long as each of those interactions involves only a few particles. Typically, however, there are also constraints as to which particles may interact with each other. A common situation is that of particles fixed on a lattice, where the particles can only interact substantially with their immediate neighbours. This additional condition is referred to as *geometric locality*, and should not be confused with the more general definition of locality used in this thesis and defined above.

Systems and Environments

Another situation in which treating a system as composite is important is when describing a system coupled to an environment (also called a *bath*). In this case, we can imagine the system and environment each described by some Hilbert space $\mathcal{H}_\mathcal{S}$ and $\mathcal{H}_\mathcal{B}$, respectively, so that the composite system-environment is the space $\mathcal{H} = \mathcal{H}_{\mathcal{S}} \otimes \mathcal{H}_{\mathcal{B}}$. The Hamiltonian of the system-environment can generically be written as

$$H = H_{\mathcal{S}} \otimes \mathbb{1}_{\mathcal{B}} + \mathbb{1}_{\mathcal{S}} \otimes H_{\mathcal{B}} + \sum_{\alpha} S_{\alpha} \otimes B_{\alpha}$$

where $H_{\mathcal{S}}$ and each S_{α} are Hermitian operators on the system, $H_{\mathcal{B}}$ and each B_{α} are Hermitian operators on the environment, and $\mathbb{1}_{\mathcal{S}}$ and $\mathbb{1}_{\mathcal{B}}$ are the identity operators on the system and environment respectively. In the absence of the $S_{\alpha} \otimes B_{\alpha}$ terms, the system and environment are not coupled, and evolve separately from each other. Indeed, if $H = H_{\mathcal{S}} \otimes \mathbb{1}_{\mathcal{B}} + \mathbb{1}_{\mathcal{S}} \otimes H_{\mathcal{B}}$ then, necessarily, the unitary evolution $U(t)$ of the system-environment can be decomposed as $U(t) = U_{\mathcal{S}}(t) \otimes U_{\mathcal{B}}(t)$ where $U_{\mathcal{S}}(t)$ and $U_{\mathcal{B}}(t)$ are unitary operators on the system and environment, respectively; consequently, any state $|\psi_{\mathcal{SB}}\rangle = |\psi_{\mathcal{S}}\rangle \otimes |\psi_{\mathcal{B}}\rangle$ evolves as $|\psi_{\mathcal{SB}}(t)\rangle = U_{\mathcal{S}}(t)|\psi_{\mathcal{S}}(0)\rangle \otimes U_{\mathcal{B}}(t)|\psi_{\mathcal{B}}(0)\rangle$, with the system unambiguously in the state $U_{\mathcal{S}}(t)|\psi_{\mathcal{S}}(0)\rangle$. However, in the presence of a coupling, i.e. where $\sum_{\alpha} S_{\alpha} \otimes B_{\alpha} \neq 0$, this is no longer true: the combined evolution cannot generally be written in this form, and the presence of the environment affects the evolution of the system, becoming entangled with it.

Typically, when using this formalism, the system is something that we can (at least partially) control and that we are interested in measuring, say, for the purpose of computation. The environment, on the other hand, is uncontrollable, and we are not interested in – or even capable of – measuring it. A leading challenge in developing scalable, useful quantum computers is overcoming environmental disturbance, and much of this thesis grapples with this challenge.

1.1.5 Mixed states

So far, we have discussed quantum systems that are in some definite state, say $|\psi\rangle$. We now briefly review the formalism applicable to statistical mixtures of quantum states, i.e. a formalism that describes the situation where we only have partial information about the state of the quantum system. This formalism makes use of what is called a *density matrix* or *density operator*.

Suppose we have a quantum system that is known to be in one of a number of states $\{|\psi_i\rangle\}$, each with probability p_i. Then the density operator for this ensemble is defined to be the Hermitian operator

$$\rho = \sum_{i} p_i \, |\psi_i\rangle\langle\psi_i| \ .$$

The states $\{|\psi_i\rangle\}$ need not be orthogonal and therefore the p_i are not necessarily eigenvalues. The eigenvalues of ρ are nonetheless non-negative because ρ is a *positive* operator: for any $|\phi\rangle$, we have $\langle\phi|\rho|\phi\rangle = \sum_i p_i |\langle\phi|\psi_i\rangle|^2 \geqslant 0$. Also note that because the system is definitely in one of the states $\{|\psi_i\rangle\}$, we have $\sum_i p_i = 1$, and therefore $\operatorname{Tr}\rho = 1$. Indeed, these last two features are the defining features of density matrices, and it can be shown that any operator ρ is a density matrix for some set of states $\{|\psi_i\rangle\}$ with some probabilities $\{p_i\}$ if and only if ρ is a positive operator and satisfies $\operatorname{Tr}\rho = 1$.

In the case where the quantum system is in a definite state, say $|\psi\rangle$, then $\rho = |\psi\rangle\langle\psi|$, with a probability of 1 associated with $|\psi\rangle$. In this case, we describe the state as *pure.* If the state is not pure, we call it *mixed.* If no information is known about the system at all, the density matrix will be proportional to the identity matrix $\mathbb{1}$, and is referred to as *maximally mixed.*

To consider how measurement and evolution work in the density matrix formalism, one need only realize that a density matrix is a convex combination of pure states. Evolution and measurement distributions on the density matrix simply correspond to evolution and measurement of each of these pure states, leaving the sum over the ensemble and the associated probabilities p_i unchanged. One thereby obtains the following.

Measurement: Let $\{|\phi_j\rangle\}$ be a set of orthonormal basis vectors of the state space. Given a density operator ρ, we can *measure* ρ in this basis, obtaining a result of $|\phi_k\rangle$ with probability

$$p_{\phi_k} = \langle\phi_k|\rho|\phi_k\rangle$$

at which point the density operator of the system becomes $|\phi_k\rangle\langle\phi_k|$.

Unitary Evolution: $\rho(t)$ evolves under the time-evolution operator $U(t)$ according to

$$\rho(t) = U(t)\,\rho(0)\,U(t)^\dagger\,.$$

Although the density matrix formalism described above is defined for an ensemble of quantum states with uncertainty as to which state the system is in, its primary use in this thesis will be for describing parts of composite systems. Suppose a joint quantum system AB is in the state $|\psi\rangle$. We would like to describe the system of just A alone. In the general case, there is no pure quantum state associated with A, since the state may be entangled with B as well. However, using the density operator formalism, we can ascribe a density matrix ρ to the state of A in such a way that all measurement outcomes for measurements performed on A alone are reproduced (in the sense of yielding the same measurement probability distribution and final state). To this end, we define the *reduced density operator* for A to be

$$\rho = \mathrm{Tr}_B\,|\psi\rangle\langle\psi|$$

where Tr_B is the *partial trace over* B, given by

$$\begin{aligned}\mathrm{Tr}_B\Big(\sum_{ijk\ell} c_{ijk\ell}\,|a_i\rangle\langle a_j|\otimes|b_k\rangle\langle b_\ell|\Big) &= \sum_{ijk\ell} c_{ijk\ell}\,|a_i\rangle\langle a_j|\,\mathrm{Tr}\Big(\,|b_k\rangle\langle b_\ell|\,\Big)\\ &= \sum_{ijk\ell} c_{ijk\ell}\,\langle b_\ell|b_k\rangle\,|a_i\rangle\langle a_j|\end{aligned}$$

for any vectors $\{|a_i\rangle\}$ and $\{|b_i\rangle\}$ on A and B respectively. This formalism is particularly useful for joint system-environment situations, where one is interested in the state of the system but not the environment.

Suppose the system and environment are initially unentangled, say with the system in the state $|\psi_{\mathcal{S}}\rangle$ and the environment in the state $|\psi_{\mathcal{B}}\rangle$, so that the joint system-environment state is $|\psi_{\mathcal{S}}\rangle\otimes|\psi_{\mathcal{B}}\rangle$. The density matrix of the joint system is therefore $\rho_{\mathcal{SB}} = |\psi_{\mathcal{S}}\rangle\langle\psi_{\mathcal{S}}|\otimes|\psi_{\mathcal{B}}\rangle\langle\psi_{\mathcal{B}}|$, and the partial trace over the environment gives $\mathrm{Tr}_{\mathcal{B}}(\rho_{\mathcal{SB}}) = |\psi_{\mathcal{S}}\rangle\langle\psi_{\mathcal{S}}|$, in agreement with our initial statement that the system is in the pure state $|\psi_{\mathcal{S}}\rangle$. More generally, we could imagine the system and environment are initially unentangled but are each described by some density matrix on their respective subsystems, so that $\rho_{\mathcal{SB}} = \rho_{\mathcal{S}}\otimes\rho_{\mathcal{B}}$. Again, $\mathrm{Tr}_{\mathcal{B}}(\rho_{\mathcal{SB}}) = \rho_{\mathcal{S}}$. More general still, the system-environment may not be unentangled at all, in which case we cannot write $\rho_{\mathcal{SB}}$ as the tensor product of two states. The density matrix $\mathrm{Tr}_{\mathcal{B}}(\rho_{\mathcal{SB}})$ on the system, however, will still describe the state of the system, in that any measurement performed on the system alone will give the same outcome distribution whether one uses $\rho_{\mathcal{SB}}$ or $\mathrm{Tr}_{\mathcal{B}}(\rho_{\mathcal{SB}})$; the reader is invited to consult [1] for an explanation of why this is the case.

Quantum operations

It is an axiom of quantum mechanics that evolution of a quantum system is governed by the Schrödinger equation, and therefore time-evolution is described by a unitary operator. However, this only applies to the evolution of an *isolated* quantum system. To describe how only a *part* of a quantum system evolves – say, a way to describe how the state of the system evolves, without tracking how its adjoined environment evolves – we use the formalism of so-called *quantum operations*.

An operator that maps linear operators (such as density matrices) to linear operators is called a *superoperator*. Suppose that the system-environment is initially in the state $\rho_{\mathcal{SB}}$ and evolves under the unitary U, taking it to the state $U\rho_{\mathcal{SB}}U^\dagger$. The system alone (after the environment is traced out) is therefore initially described by the density matrix $\rho_{\mathcal{S}} = \mathrm{Tr}_{\mathcal{B}}(\rho_{\mathcal{SB}})$, and afterwards is described by

$$\mathcal{E}(\rho_{\mathcal{S}}) = \mathrm{Tr}_{\mathcal{B}}(U\rho_{\mathcal{SB}}U^\dagger)\,,$$

where the superoperator $\mathcal{E}$ depends on the initial system-environment state $\rho_{\mathcal{SB}}$ and its unitary evolution. Thus, while the system-environment evolves under the unitary operator U, the evolution of the density matrix of the system alone is described by the superoperator $\mathcal{E}$ acting on just the system. In general, the action of $\mathcal{E}$ is more complicated than conjugation by a unitary matrix (as would have been the case for an isolated system); we now proceed to review the mathematical framework of such superoperators.

We define a superoperator $\mathcal{E}$ to be a *quantum operation* if its action on density matrices ρ can be written as

$$\mathcal{E}(\rho) = \sum_k E_k \rho E_k^\dagger$$

where the *Kraus operators* (also known as *operation elements*) $\{E_k\}$ satisfy the condition

$$\sum_k E_k^\dagger E_k = \mathbb{1}\ .$$

Quantum operations may also be called *quantum channels* and in Chapter 6 are referred to as being *admissible superoperators*. We note in passing that this definition is equivalent to saying that $\mathcal{E}$ is a completely-positive trace-preserving linear map (a CPTP map), meaning that $\mathrm{Tr}\,\mathcal{E}(\rho) = \mathrm{Tr}\,\rho$ for all ρ and $(\mathcal{E}\otimes\mathcal{I})(A)$ is positive for any positive operator A, where $\mathcal{I}$ is the identity superoperator ($\mathcal{I}(\rho) = \rho$). For more information and generalizations, including proofs that these mathematical formulations are equivalent to the idea of tracing out the environment of a unitarily evolved system-environment, the reader is invited to consult the chapter on quantum operations in [1].

1.1.6 Bosonic systems

Although quantum computing is often primarily interested in finite-dimensional systems, we briefly mention here the case of bosonic systems, as will be relevant in Chapter 5. The formalism of a simple isolated bosonic system is that of the quantum harmonic oscillator. An arbitrary number of bosons can be present in such a system, with corresponding basis states

$\{|n\rangle : n = 0, 1, 2, \ldots\}$ where n is the number of bosons present. Two important operators for this system are the annihilation (lowering) and creation (raising) operators, a and $a^\dagger$, which act as

$$a|n\rangle = \sqrt{n}|n-1\rangle$$
$$a^\dagger|n\rangle = \sqrt{n+1}|n+1\rangle$$

and are related by the bosonic commutation relation $[a, a^\dagger] = 1$. The states described above, representing the number of bosons present, are eigenstates of the number operator $N = a^\dagger a$. The Hamiltonian describing a quantum harmonic oscillator is (up to a constant) proportional to N, so that the number basis states are also the energy eigenstates.

It is important to note that in the simple quantum harmonic oscillator Hamiltonian, H is *linearly* proportional to N, so that each additional boson contributes the same amount of energy. In this sense, there are no interactions, since each boson only has the effect of adding a constant amount of energy to the system. The Hamiltonian could also contain terms that are non-linear in N, representing interactions between bosons. For example, if an additional term of $N(N-1)$ were present, which is non-zero if and only if at least two bosons are present, this would indicate a multiple-particle interaction term.

As a further generalization of the harmonic oscillator Hamiltonian, one can consider systems with multiple bosonic *modes*. For example, each mode might represent a lattice site, again with an arbitrary number of particles (bosons) allowed in each site. Each mode m will have its own creation, annihilation, and number operators, $a_m^\dagger, a_m$, and N_m, and may then contribute a term to the Hamiltonian, so that H has a term proportional to $\sum_m N_m$. Indeed, this sum represents the simple case where these modes are all independent and no interactions exist. However, the modes may not be independent.

Consider a system with two modes ($m = 1, 2$), starting in state $|1, 0\rangle$, meaning that mode 1 has one particle in it, while mode 2 has none. If we apply the operator $a_2^\dagger a_1$ to this state, the result will be the state $|0, 1\rangle$; the particle has "hopped" from mode 1 to mode 2. If such operators are present in the Hamiltonian, then the fact that the Hamiltonian governs time evolution implies that such operations act during evolution, allowing such hopping over time.

An important Hamiltonian with all of these features is the (generalized) Bose-Hubbard model,

$$H = \sum_m E_m N_m + \sum_{\langle l,m \rangle} J_{l,m} a_l^\dagger a_m + \frac{\Gamma}{2} \sum_m N_m(N_m - 1) \tag{1.1}$$

where E_m is the energy corresponding to mode m (if there were no interactions or hopping), $J_{l,m}$ governs the inter-mode hopping, and Γ is a non-linear interaction energy resulting from the presence of multiple particles in the same mode. This Hamiltonian, which is especially applicable to modelling cold atoms in optical lattices, will be useful in Chapter 5.

1.2 Hamiltonian-based quantum computing

As we noted before, a quantum system's energetics, interactions, and time evolution is governed by its Hamiltonian. In this sense, from a physicist's perspective, it is natural to develop quantum computing architectures defined by Hamiltonians. This differs, however, from the more common, computer-science-like perspective of theoretical quantum computation defined by the application of local unitary operations, called gates. The circuit model will be discussed later in Sec. 1.3. In the current section, we discuss some ideas behind the Hamiltonian-based approach, which is a main theme in this thesis.

There are a number of quantum computing models based explicitly on Hamiltonians. Many of these are provably equivalent in power to the circuit model of quantum computation. We will review several of these models, all of whose basic idea is as follows. We start with some initial quantum state $|\psi(0)\rangle$ that is easy to prepare. We then evolve under some relatively simple, possibly time-dependent Hamiltonian, $H(t)$, for some length of time T, to obtain a final state $|\psi(T)\rangle$. We then measure this final state, generally in an easy-to-measure basis such as the computational basis. We design our algorithm so that measuring $|\psi(T)\rangle$ should yield some information about the solution to the computational problem of interest, with the details of the problem being used in some form to dictate our choice of $|\psi(0)\rangle$, $H(t)$, and/or T. Although $H(t)$ may be time-dependent, it generally is simple to describe and implement and is of bounded strength (not requiring unbounded amounts of energy). Preferably, it should also be physically reasonable, satisfying, for example, locality constraints, although one may still be interested in theoretical models that employ less-than-physically-reasonable Hamiltonians too. Note that in this model as we have defined it, only a single measurement is performed, at the very end of the computation.

1.2.1 Adiabatic quantum computation

One of the most famous Hamiltonian-based quantum computing models is *adiabatic quantum computation*, introduced in [4]. Suppose we have a Hamiltonian H_f, whose ground state (the state with the lowest eigenvalue) is the solution to a computational problem of interest. For example, perhaps we seek to solve a classical optimization problem, finding the minimum value $z = z_{\text{min}}$ of some cost function $C(z)$, a real function of binary strings z. Then we can interpret $C(z)$ to be a Hamiltonian, $H_f = \sum_z C(z) |z\rangle\langle z|$, diagonal in the binary string $\{|z\rangle\}$ basis, with ground state $|z_{\text{min}}\rangle$. Preparing this ground state directly may be difficult, equivalent to solving the optimization problem, since we do not know ahead of time the value of z_{min}. However, in the adiabatic model, we start by preparing the ground state of a different Hamiltonian H_i, whose ground state is easy to prepare, and evolve the system according to a time-dependent Hamiltonian $H(t)$ that interpolates between the two. For example, we might use the Hamiltonian

$$H(t) = \Big(1 - \frac{t}{T}\Big) H_i + \frac{t}{T} H_f, \qquad 0 \leqslant t \leqslant T,$$

where T is the total evolution time; initially, $H(0) = H_i$, and finally, $H(T) = H_f$. According to the *adiabatic theorem* [5, 6] of quantum mechanics, provided that this evolution is done slowly enough (i.e. provided that T is large enough in the interpolation given above), if we start in the ground state of H_i then we will finish in a state very close to the ground state of H_f. For example, we may choose $H_i = -\sum_i X_i$ to be the sum of Pauli X operators on each qubit, whose ground state is the uniform superposition, proportional to

$$\sum_z |z\rangle = (|0\rangle + |1\rangle) \otimes \cdots \otimes (|0\rangle + |1\rangle),$$

and (relatively) easy to prepare for our initial state $|\psi(0)\rangle$. Under the adiabatic scenario of a slowly-varying Hamiltonian, since the final state $|\psi(T)\rangle$ is close to $|z_{\text{min}}\rangle$, a final measurement gives $z_{\min}$ with high probability. For an analysis on what constitutes as sufficiently slowly-varying, and the corresponding requirements on the eigenvalue spectrum of $H(t)$, the reader is invited to consult [4].

Adiabatic quantum computation is known to be universal, i.e., is equivalent in computational power to the circuit model [7], albeit not in the form of the optimization algorithm described above. Indeed, an algorithm for factoring has been developed and implemented using this model [8], reportedly [9] factoring $56153 = 233 \times 241$. A related computational method, *quantum annealing*, is essentially a form of the adiabatic quantum computing model in the presence of an environment, and is most famous for being the goal of the D-Wave processor [10].

1.2.2 Feynman's model

Another class of Hamiltonian-based quantum computation models are those based on Feynman's original vision of a quantum computing device [11] in 1985. One way of formulating this type of computer is as follows. Suppose we have some initial state $|\psi_0\rangle$, to which we wish to apply a sequence of unitary operations, $U = U_L U_{L-1} \cdots U_2 U_1$, obtaining a final state $|\psi_L\rangle = U|\psi_0\rangle$ that encodes the solution to our computation. Such a scenario is easy to understand in light of the circuit model of computation, which we address shortly in Sec. 1.3, with the unitaries U_i representing individual computational gates that we wish to apply. Suppose, however, that we wish to do this using Hamiltonian-based quantum computation. Perhaps, for example, we need to use a single time-independent Hamiltonian, ruling out the application of a series of unitary gates. A time-independent Hamiltonian giving rise to U via the Schrödinger equation may be horrendously complicated and non-local, even if each of the individual gates U_i is easy to implement with some local Hamiltonian.

The Feynman-type method of computation proceeds as follows. In addition to the computational space in which $|\psi_0\rangle$ lives, which we call the *work register*, we adjoin another space $\mathrm{span}\{|t\rangle : t = 0, \ldots, L\}$, called the *clock register*. We then construct the time-independent Hamiltonian

$$H = \sum_{t=1}^{L} U_t \otimes |t\rangle\langle t-1| + U_t^\dagger \otimes |t-1\rangle\langle t| \ .$$

Note that t here does not refer to time, but rather the clock-states; H is time-independent. This operator is Hermitian, and therefore in principle can serve as a Hamiltonian. Assuming the unitary gates U_i are local (operating on only a few qubits each), H can also be made local by a clever choice of the clock register $\{|t\rangle\}$. If one starts with the initial state $|\psi_0\rangle \otimes |0\rangle$ and evolves under H, the state will become a "history" state – a superposition over states $\Big(U_t U_{t-1} \cdots U_1 |\psi_0\rangle\Big) \otimes |t\rangle$. Measuring the clock register will yield a value of $t = L$ with a reasonably large probability (assuming L is not too large), and when this occurs, the work register will contain the final state $|\psi_L\rangle = U_L U_{L-1} \cdots U_2 U_1 |\psi_0\rangle$ as desired. Straightforward techniques exist to amplify the probability of obtaining this outcome to be very high.

Aside from being one of the first models of quantum computation, this model is useful in providing equivalencies between the circuit model and Hamiltonian-based models of quantum computation (such as the adiabatic model discussed above). It is also the father of similar models, such as the Margolus [12] and Lloyd-Terhal [13] models of quantum computation. Furthermore, it has played an important role in analysing the computational power of different Hamiltonians, as will be discussed in Sec. 1.4.3.

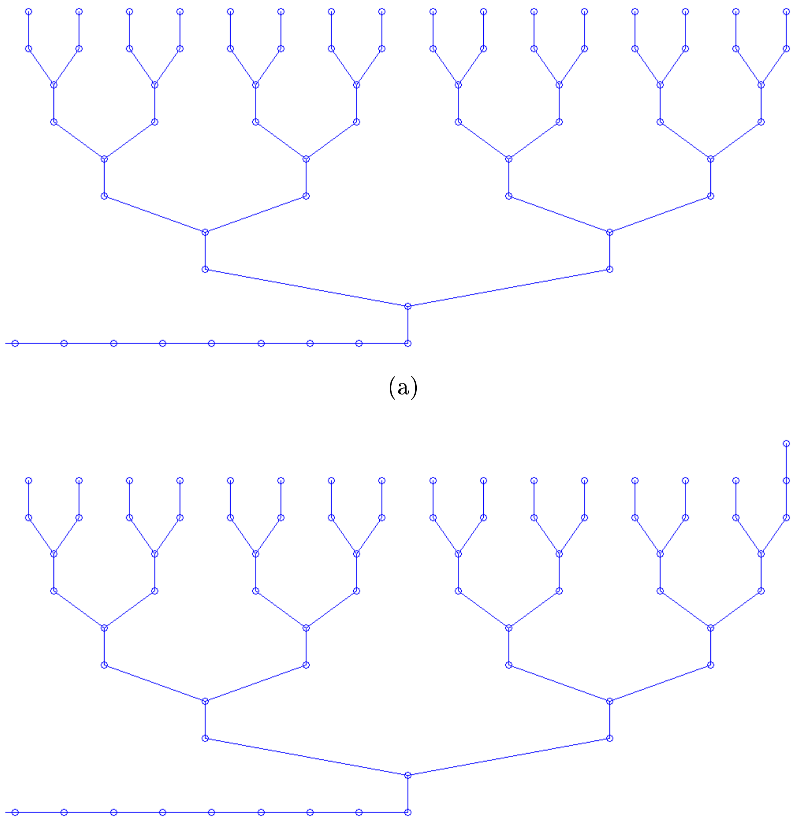
1.2.3 Continuous-time quantum walks

The continuous-time quantum walk model of computation was introduced in [14]. The idea is to perform a computation by analysing the dynamics of a quantum particle on a graph (i.e. a set of discrete points in space, connected to each other in some way). Suppose we have some weighted, undirected graph with vertices $\{i\}$. Associate with this a state space $\{|i\rangle\}$ corresponding to the vertices. For our purposes, we define a continuous-time quantum walk by interpreting the *adjacency matrix* of the graph as a time-independent Hamiltonian H, meaning that $\langle i|H|j\rangle$ is equal to the weight of the edge connecting vertices i and j (or equal to 0 if no edge connects them). The state of the system is some $|\psi\rangle$, which can be interpreted as a wavefunction on the discrete space of the vertices. We start in some initial state $|\psi(0)\rangle$ (e.g. a state localized to some particular vertex, or a state representing a travelling wave through some part of the graph), evolve under the Hamiltonian for some time t, and then measure whether the wavefunction is found in some subset S of the vertices, giving *yes* with probability $\sum_{i\in S} |\langle i|\psi(t)\rangle|^2$. In some sense, the dynamics of any time-independent Hamiltonian can be interpreted as a quantum walk, including the Feynman model discussed above. In the present discussion, however, we use quantum walks to denote a walk described explicitly on a graph, which is typically sparse or planar.

Continuous-time quantum walks are the quantum analogue of continuous-time random walks and can be used for computation. In this model, one encodes a problem in a graph, and therefore in a Hamiltonian, such that the above procedure constitutes an algorithm for the problem. To show one way in which this model can work, we briefly present (without proof) how a continuous-time quantum walk can be used to solve a Grover-type problem, formulated as computing the N-bit $\textsc{or}$ function (i.e. for any N-bit input x, $\textsc{or}(x) = 0$ for $x = 0$ and $\textsc{or}(x) = 1$ otherwise). A graph, i.e. Hamiltonian H_x, to solve this problem is shown in Fig. 1-1, for the problem of distinguishing whether $x = 0$ vs. x has precisely one non-zero bit. This graph involves a "runway" attached to a modified binary tree with an extra connection on the top row of the tree depending on the input x. One can show that if the initial state is a travelling wave on the runway (specifically, $\langle v|\psi(0)\rangle \propto e^{-iv\pi/2}$ for v on the runway) and evolved for time proportional to $\sqrt{N}$ under H_x, a measurement on the even nodes of the last third of the runway sites yields *no* with high probability if $x = 0$, and *yes* otherwise. In this way, starting with the appropriate initial state, evolving under the input-dependent Hamiltonian H_x, to which we assume we have access (even though we assume we do not necessarily know the value of x), and measuring the appropriate sites, one can calculate the value of $\textsc{or}(x)$. Many other continuous-time quantum walk algorithms exist, and the model has been proven to be universal for quantum computation [15] (and see also a generalization in [16]).

1.2.4 Analogue Hamiltonian simulation and algorithms

While the previous models are (at least theoretically) capable of general-purpose quantum computation, we note that it may also be useful to consider quantum computing devices with more modest goals. One of the main motivations for quantum computing is quantum simulation, where we would like to study the properties of some specific quantum system, say one modelled by a Hamiltonian $\tilde{H}$, without having direct access to such a system. To simulate a quantum system on a classical computer is typically difficult, due to the former's much larger state space – for example, an n-bit state has n parameters, but to describe a generic quantum system of n spin-$\frac{1}{2}$ particles requires 2^n parameters. While a universal



(a)

(b)

Figure 1-1: Graphs for the 16-bit OR problem for (a) an input of $x = 0000000000000000$ (with an empty top row in the graph) and (b) an input of $x = 0000000000000001$ (with a single node in the top row). The Hamiltonian H_x consists of two parts: a runway attached to a modified binary tree, and an input-specific part consisting of an extra vertex and edge corresponding to the location of the 1 (if any is present) in the input x. Straight edges have weight 1 and slanted edges have weight $\frac{1}{\sqrt[4]{2}}$. The tree to accomplish this task was derived using span programs [17].

quantum computer – say, one using the circuit model – can simulate quantum systems [18], we can also imagine an analogue, possibly-non-universal quantum device designed to simulate some class of quantum systems. That is, we can imagine a system obeying some Hamiltonian H but that is able to effectively approximate the evolution of a system obeying $\tilde{H}$. This idea will be made clearer in Chapter 3 where we describe one such method.

One obvious use of quantum simulation might be for quantum chemistry or condensed-matter physics, where one may want to determine properties of a quantum system, modelled by some Hamiltonian, that one has not yet manufactured. We may also be interested in Hamiltonians that, rather than modelling a physical system, implement an algorithm. As an example of an interesting Hamiltonian used for algorithmic purposes, we briefly discuss the "analogue" (continuous-time) version of Grover's algorithm presented in [19]. Suppose we are looking for some special n-bit string z_0 out of all possible n-bit strings, but have no information about z_0 itself other than some way of querying whether or not a given number z is the desired bit-string z_0. This is the problem addressed by Grover's algorithm [20], which presents a quantum circuit given access to a quantum oracle that may be thusly queried. In the Hamiltonian-based algorithm of [19] for this problem, we assume that we have access to (but no information about) the Hamiltonian $H_{z_0} = |z_0\rangle\langle z_0|$, as well as the problem-independent Hamiltonian $H_D = |s\rangle\langle s|$, where $|s\rangle = 2^{-n/2}\sum_z |z\rangle$ is the uniform superposition over all n-bit bit-strings. We evolve under the time-independent Hamiltonian

$$H = H_D + H_{z_0} = |s\rangle\langle s| + |z_0\rangle\langle z_0|$$

starting from the uniform superposition $|s\rangle$. Then by measuring the state of the system (in the $|z\rangle$ basis) at time $T = \pi 2^{n/2-1}$, the desired bit-string z_0 will be obtained.

1.3 Circuit model

An alternative approach for quantum computation, and the one discussed most often, is the *circuit model* of quantum computation. Rather than specifying the Hamiltonian describing a quantum system, the circuit model focuses on viewing the evolution of the system as a sequence of instantaneous unitary operations, called *gates*. Each quantum gate acts on a small number of qubits and serves as a building block for the overall quantum computation. They are (to some extent) analogous to the logic gates that appear in classical computing, such as NOT, AND, OR, and NAND. Before discussing the quantum case, let us first review the classical case.

1.3.1 Classical circuits

The classical circuit model is a model of classical computation, equivalent in power to other common models (such as "Turing machines"), and is in some respects an idealization of the circuitry in modern digital devices. The model involves two main parts. There are wires, each of which carries a bit of information (0 or 1), and logical operations, or gates, that are performed on these wires. Conventionally, the circuit is taken to be acyclic, i.e. none of the wires can form loops. Some gates act on a single wire, others act on multiple wires. The most important 1-bit (i.e. single-wire) gate is the NOT gate; important 2-bit gates include the AND, OR, XOR (i.e. "exclusive-OR"), and NAND (i.e. "NOT-AND") gates. Additionally, wires can copy information, in the FANOUT operation, where a wire splits into two, each carrying a copy of the original information. The action of these gates and operations can

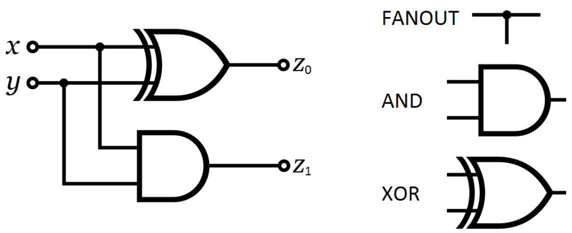


Figure 1-2: A circuit that takes in two 1-bit numbers, x and y, and computes $z = x + y$, with output z written in binary, $z = z_1 z_0$. The gate elements present in the circuit are indicated on the right.

be understood from the following truth tables; understand 0 to represent FALSE and 1 to represent TRUE.

x	y	AND(x, y)	OR(x, y)	XOR(x, y)	NAND(x, y)
0	0	0	0	0	1
0	1	0	1	1	1
1	0	0	1	1	1
1	1	1	1	0	0

x	NOT(x)
0	1
1	0

x	FANOUT(x)
0	0, 0
1	1, 1

A circuit, composed of wires and gates, can perform computational tasks: the front of the wires are initialized to some bit values (typically encoding the problem input), flow through the gates, and the information present at the ends of the wires is then read as output. As an extremely simple example, the *half-adder* circuit shown in Fig. 1-2 adds two 1-bit numbers together, giving a 2-bit output.

In classical circuits, one generally take for granted access to constant bits and to FANOUT. But we may ask what other gates are required to perform certain computations. It is well known that the set of AND, OR, and NOT is universal – any circuit, no matter how complicated, can be built up from these operations, with more complicated gates built out of just these three gates. This set is larger than is necessary – the set of AND and NOT are universal, as OR can be simulated with AND and NOT. In fact, for universality it suffices to have access to only the NAND gate.

1.3.2 Quantum circuits

The basic idea of classical circuits, namely starting with some input bits, applying gates to them in sequence, and measuring the final output, carries over to the quantum domain. Instead of wires carrying bits (0 or 1), they now carry qubits (in general, a superposition over bit-strings $|0 \cdots 0\rangle, \ldots, |1 \cdots 1\rangle$), and instead of applying logic gates to the bits, there are now quantum gates applied to the qubits. Measurement of each qubit can be performed at the end in the computational basis, $\{|0\rangle, |1\rangle\}$, although due to the probabilistic nature of quantum measurements, the outcome will in general be non-deterministic.

We have already discussed qubits and measurement above, so our discussion here will primarily focus on the nature of quantum gates. Observe that the equivalent of the classical

NOT gate is the Pauli X operation, since it flips $|0\rangle \leftrightarrow |1\rangle$. Many other classical logic gates, however, do not carry over directly into the quantum domain. In quantum mechanics, quantum states (including qubits), evolve in time under the action of a unitary operator. Quantum gates must therefore be unitary. A consequence of this is that quantum gates must be reversible: if a quantum system evolves in time according to some unitary U, then, by definition of unitarity, there exists an inverse operation $U^{-1} = U^\dagger$ that will undo this evolution. This excludes the possibility of the 2-bit-to-1-bit functions like AND, OR, XOR, and NAND discussed above, since given their (1-bit) output, there is no way to always reconstruct their (2-bit) input. Moreover, FANOUT is impossible due to the *no-cloning theorem*, which states that arbitrary quantum information cannot be copied perfectly. So a quantum gate with n input qubits must also have n output qubits, and the gates we saw in classical computing are not the gates that appear in quantum circuits.

Furthermore, while in classical computing only four 1-bit-to-1-bit functions are possible, namely the NO-OP ($x \to x$), NOT ($0 \leftrightarrow 1$), CONSTANT-1 ($x \to 1$) and CONSTANT-0 ($x \to 0$), there are many different 1-qubit-to-1-qubit gates, namely the whole set of single-qubit unitaries

$$\{e^{-i(\alpha \mathbb{1} + \theta \hat{n}\cdot\vec{\sigma})} : \alpha \in \mathbb{R}, \theta \in \mathbb{R}, \hat{n} \in \mathbb{R}^3, |\hat{n}|^2 = 1\}.$$

When considering multi-qubit operations, even more possible unitaries exist. Depending on physical capabilities, the vast majority of these might be very hard to produce directly, and the continuous nature of the parameter space can make such a general set of limited value for algorithm design. We therefore often limit our focus to much smaller gate sets that include only certain unitary matrices.

Some very important single-qubit unitary gates include the identity and the Pauli matrices,

$$\mathbb{1} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

Note that X, Y, and Z here are the same Pauli Hermitian matrices discussed earlier; they are very important matrices both when thinking about Hamiltonians as well as unitary matrices. In the former case, this is because they, along with the identity $\mathbb{1}$, form a basis for 2×2 Hermitian matrices. In the latter case, this is because X serves as the quantum bit-flip operator,

$$X(a|0\rangle + b|1\rangle) = a|1\rangle + b|0\rangle,$$

while Z serves as the phase-flip operator,

$$Z(a|0\rangle + b|1\rangle) = a|0\rangle - b|1\rangle,$$

and $Y = iXZ$ performs both operations together.

Some other important single-qubit gates are the Hadamard gate H, the phase gates R_θ, and the T gate,

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad R_\theta = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}, \quad T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}.$$

The Hadamard H (not to be confused with the Hamiltonian of a quantum system, which is also designated by H) is useful as an easy way to generate superpositions, since

$$H|0\rangle = \tfrac{1}{\sqrt{2}}(|0\rangle + |1\rangle),$$

while the phase gates are a generalization of the Z gate, but with phases other than -1 being applied to $|1\rangle$:

$$R_\theta(a|0\rangle + b|1\rangle) = a|0\rangle + e^{i\theta}b|1\rangle\,.$$

The T gate is simply an important special case, $T = R_{\pi/4}$.

A very important two-qubit gate is the controlled-NOT gate, or CNOT gate,

$$\textsc{cnot} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

which acts on the computational basis by flipping the second qubit if and only if the first qubit is a $|1\rangle$:

$$\begin{aligned} \textsc{cnot}\ |0\rangle \otimes |\psi\rangle &= |0\rangle \otimes |\psi\rangle \\ \textsc{cnot}\ |1\rangle \otimes |\psi\rangle &= |1\rangle \otimes (X\,|\psi\rangle). \end{aligned}$$

In this sense, the action of the CNOT is controlled by the first qubit, which dictates whether it performs a NOT (X) operation. As a quantum gate, its action on superposition states is defined by linearity. We can also consider generalizations of the CNOT gate that are controlled by more than one qubit, and we can consider operations beyond just X. A controlled-U gate on $m+1$ qubits that is controlled by m qubits will apply U to the $(m+1)^{\text{th}}$ qubit if all of the first m qubits are $|1\rangle$, but perform no action if any of the first m qubits are $|0\rangle$. The basic elements of a quantum circuit, including circuit element diagrams for some of the gates just described, is shown in Fig. 1-3.

As in the classical circuit case, where we discussed some logic gate sets that are universal for classical circuits, one may ask what quantum gate sets are universal for quantum computation. One universal set is CNOT together with all single-qubit unitary operations, $\{\textsc{cnot}\} \cup \{e^{-i(\alpha \mathbb{1} + \theta \hat{n}\cdot\vec{\sigma})}\}$: any unitary operator acting on n qubits can be decomposed exactly as the product of some sequence of CNOT and single-qubit gates applied to those qubits. One may be concerned by the fact that this gate set is infinite – we have included in it an uncountably infinite number of single-qubit gates; contrast this with the classical case, where a single NAND gate suffices. Fortunately, it can be shown that *approximately* universal gate sets exist, containing only a finite number of gates, with which any unitary operator can be *approximately* decomposed to arbitrary precision as a product of gates drawn from the set. An (approximate) universal gate set of particular importance is the set $\{\textsc{cnot}, H, T\}$, since it can be shown that any single-qubit unitary can be approximated arbitrarily accurately as some product involving only H and T operations.

We conclude this section by noting that an advantage of the quantum circuit model is that it is relatively straightforward, in principle, to determine what the output of the circuit will be: one need merely perform matrix and vector multiplication of the gates and states. One generally assumes that gates are performed instantaneously, acting in sequence, so that the evolution of the quantum state can be mapped through the passage of a discretized time. One need not worry about the complicated piecewise-time-dependent

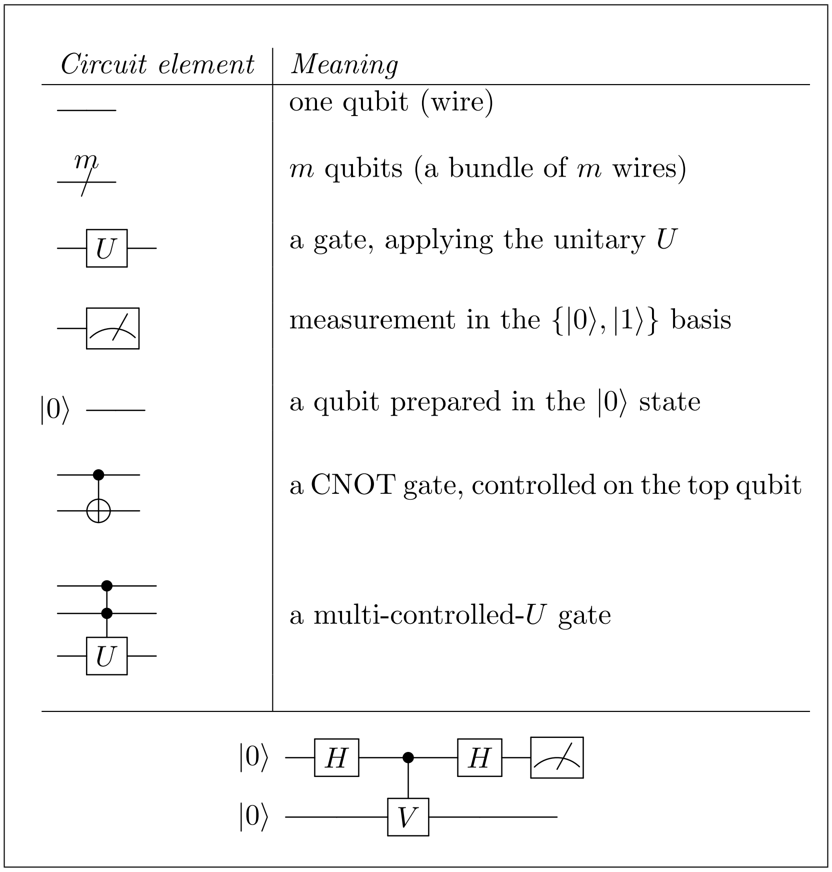

Circuit element
Meaning
one qubit (wire)
m
m qubits (a bundle of m wires)
U
a gate, applying the unitary U
measurement in the {|0⟩, |1⟩} basis
|0⟩
a qubit prepared in the |0⟩ state
a CNOT gate, controlled on the top qubit
U
a multi-controlled-U gate
|0⟩
H
H
|0⟩
V

Figure 1-3: (Top) Elements of a quantum circuit. (Bottom) A very simple quantum circuit, containing Hadamard gates and a controlled-V gate for some single-qubit unitary operator V. This particular circuit is of no particular interest to us here, but for the interested reader we note that it is the so-called Hadamard test, used for estimating matrix entries of the unitary V. The top qubit register, after measuring, will output a 0 with probability $\frac{1}{2}(1 + \mathrm{Re}\langle 0|V|0\rangle)$, and therefore repeated uses of the circuit will allow one to estimate the value of $\mathrm{Re}\langle 0|V|0\rangle$. The Hadamard test will be used in Chapter 6.

Hamiltonian that may be required to give rise to this evolution, as one need only worry about implementing each gate in sequence. Fundamentally it is important to realize, however, that any actual implementation using a quantum apparatus is governed by some Hamiltonian, so it is important to consider the task of engineering physically-realizable Hamiltonians that give rise to useful gates (as we do in Chapter 5), and natural to also focus on inventing – and exploring the viability of – Hamiltonian-based models of computation, like the ones mentioned in Sec. 1.2.

1.4 Complexity theory

Having discussed models of computation, a natural question to ask, for a given algorithm, is how long it takes to run. In the circuit setting, we may ask how many gates a particular algorithm requires, or how deep its circuit must be. To perform a given algorithm on n bits (or qubits), does one need a circuit that has 2^n gates, or does there exist a circuit that uses only n^2 gates? Conversely, we may ask what can be done assuming access to only some number of gates, some amount of time, or some other set of resources. These are the questions of computational complexity theory. In this section we provide a very brief background to the aspects of complexity theory that will appear in this thesis.

1.4.1 Brief mathematical background

Decision problems

A decision problem is a yes-no problem. For example, the question "given integers x, y, and z, is it true that $x + y = z$?" is a decision problem. A specific instance of the problem, say "is $2 + 3 = 78$?", will have an answer of either yes or no. We formalize this notion using bit-strings.

We define a *language* to be some subset $L \subset \{0,1\}^*$, where $\{0,1\}^*$ denotes the set of all possible bit-strings of any finite length. Let x be a bit-string, $x \in \{0,1\}^*$. We sometimes denote the length of x, i.e. the number of bits in x, by the notation $|x|$. The bit-string x represents a specific instance of a problem. Any language L represents a decision problem, the problem being "given a bit-string x, is x in L?"

For example, consider the very simple decision problem of determining, given an input x, whether x contains an equal number of 0s and 1s. Then L is the set of all (finite-length) strings that do have an equal number of 0s and 1s. The problem can be equivalently formulated as determining, given a specific bit-string x, whether it is in L or not. Here, x is a specific instance of the problem (e.g. 10011111), in that we ask of x the question posed by L (whether $x \in L$, or equivalently, whether x has an equal number of 0s and 1s).

Languages and their instances can be much more interesting than this example, however. One might, for example, consider the problem of 3-COLOURING which asks, given a graph, can the vertices be coloured using only 3 colours in such a way that no two vertices of the same colour are connected by an edge. Here the problem instance x represents a graph, but there are ways of formulating a description of a graph using a bit-string, so 3-COLOURING fits the definition of a decision problem given above.

On the one hand, decision problems are very limited, containing only functions that map bit-strings to a single bit (representing TRUE or FALSE). Nevertheless, it turns out that many, many useful computational problems can be phrased in terms of decision problems. Furthermore, many other problems involving more complicated functions or optimization, can often be converted into decision problems without significantly changing their overall computational difficulty. Therefore, because their true-false answers makes them simpler to analyse, they are a natural focal point for the field of computational complexity.

Asymptotics

We say that a function $f(n)$ is $O(g(n))$, or $f(n) = O(g(n))$, or $f(n) \leqslant O(g(n))$, where $g(n)$ is some function, if there exist constants N and c such that for all $n > N$, we have $f(n) \leqslant cg(n)$. In other words, $f(n)$ is $O(g(n))$ if $f(n)$ is eventually always less than some multiple of $g(n)$. Loosely speaking, $f(n)$ is not growing qualitatively faster than $g(n)$ is. We say that $f(n)$ is $\text{poly}(n)$ if there exists some integer m such that $f(n)$ is $O(n^m)$, i.e. if $f(n)$ is bounded by some polynomial in n, and then we may write $f(n) \leqslant \text{poly}(n)$ or even $f(n) = \text{poly}(n)$. For example, the function $n \log n$ is asymptotically bounded above by n^2, i.e. $n \log n \leqslant O(n^2)$, and therefore is also $\text{poly}(n)$. On the other hand, 2^n is *not* $\text{poly}(n)$: there is no polynomial that is eventually always larger than 2^n.

1.4.2 Complexity classes

P, NP, and QMA

While the field of computational complexity is very large, we draw the reader's attention to a few particularly important classes. The class P is the class of problems solvable in polynomial time. Loosely speaking, a decision problem language L is in P if there is a classical (deterministic) algorithm that solves L in polynomial time. Recall that by solving L, we mean that given any bit-string x, we wish to determine whether x is in L or not, i.e. have an algorithm that answers TRUE if $x \in L$ and FALSE if $x \notin L$. If there is a classical algorithm that, given any n-bit x, correctly determines whether $x \in L$, requiring a runtime of no more than $\text{poly}(n)$, then we say that L is in P.

More precisely, we say that L is in P ($L \in$P) if there exists a family of classical algorithms (e.g. in the form of classical circuits) $\{C_n\}$, with C_n taking n bits and outputting 1 bit, such that for any input x, $C_{|x|}$ outputs

$$C_{|x|}(x) = \begin{cases} 1, x \in L \\ 0, x \notin L \end{cases}$$

and takes time $\text{poly}(|x|)$ to do so. Further, this family of algorithms has to be simple to describe and generate; this can be made precise using the notion of Turing machines, but is beyond our scope here. Most languages are not in L: they simply cannot be solved using merely polynomial time. Those that are in P are generally considered easy to solve, i.e. they can be solved *efficiently*.

Some problems may be difficult to solve, but are easy to verify if an explanation of the answer is provided. A good analogy of this idea can be seen in the game of Sudoku, where it may be difficult and time-consuming to solve a given Sudoku puzzle, but if someone gives you the puzzle already filled in, it is very easy to validate that the solution is correct. The class of problems in which it is easy to verify the solution, given sufficient but concise evidence, is called NP. This evidence, called a *witness* or *proof*, must be a bit-string whose length is short. Loosely, a language L is in NP if there exist verifier algorithms $\{C\}$ such that

- if $x \in L$ then there exists a witness w of poly($|x|$) bits such that $C(x, w) = 1$,
- if $x \notin L$ then for all possible witnesses w of poly($|x|$) bits, $C(x, w) = 0$.

The actual acronym NP stands for "non-deterministic polynomial time", for reasons that we shall not explain here; it emphatically does not stand for "not polynomial time". A famous open question is whether every problem that is easy to check is also easy to solve, i.e. whether P=NP.

A quantum analogue of P is the class known as BQP (bounded-error quantum polynomial time), in which the algorithms now must be quantum circuits that run in polynomial time and are simple to describe. Because the output of quantum circuits are generally probabilistic, the requirement on their output is relaxed: the algorithm need only output the correct answer with high probability. Similarly, a quantum analogue of NP is QMA (Quantum Merlin-Arthur), in which the verifier algorithm is a quantum circuit and the witness is a quantum state. More precise definitions will be given in later chapters as needed.

In addition to the open question of whether P=NP, it is unknown precisely where the quantum analogues fit. Although it is known that P $\subseteq$ BQP $\subseteq$ QMA and P $\subseteq$ NP $\subseteq$ QMA, the following relationships are widely (but not universally) conjectured to be true:

$$\text{(conjectures)} \quad \text{P} \subsetneq \text{BQP} \subsetneq \text{QMA}, \quad \text{P} \subsetneq \text{NP} \subsetneq \text{QMA}, \quad \text{NP} \neq \text{BQP},$$

i.e. it is conjectured that all four classes are distinct. The first of these reflects the belief that quantum computers are more powerful than classical computers. The second conjectured inequality reflects the belief that the hardest QMA problems cannot be solved efficiently, even by quantum computers.

Hardness and completeness

In addition to classifying problems according to their complexity class, we would like to have a notion of what it means for one problem to be harder than another, or for one problem to be harder than all the problems in a particular class. Suppose that one can easily transform one problem, L_1, into another problem, L_2. The ability to solve L_2 then implies the ability to solve L_1. Such a transformation is called a *hardness reduction*, and the fact that it exists from L_1 to L_2 indicates that solving L_2 is at least as *hard* as solving L_1 – after all, it means that if I have the ability to solve L_2 then I can also solve L_1.

To be more precise, suppose that we would like to determine whether a given problem instance x is in the language L_1. Further suppose that we have a hardness reduction[3] from L_1 to L_2, meaning that we have a way of converting x into a (possibly) different problem $\bar{x}$ for the language L_2 such that

$$x \in L_1 \quad \text{if and only if} \quad \bar{x} \in L_2 \,.$$

If we have access to an algorithm for L_2, along with our algorithm for converting problem instances from L_1 to L_2, then we can also solve L_1: given x, convert it into $\bar{x}$, use the algorithm to determine whether $\bar{x} \in L_2$, and thereby determine whether $x \in L_1$. To reflect our demand that the transformation is easy to perform, we further demand that this reduction algorithm itself take only polynomial time (i.e. be a *polynomial-time reduction*).

[3] This type of reduction is called a *Karp reduction*. Other types of reductions exist.

Now suppose that a problem L is at least as hard as *every* problem L' in a complexity class A, in the sense that for every $L' \in A$ there is a reduction from L' to L. Then we say that L is A-hard. If L itself is also in A, we say that L is A-complete:

L is A-complete iff $L \in A$ and every $L' \in A$ can be reduced to L in polynomial time.

For example, the problem of 3-COLOURING is NP-complete, which means that it is the hardest problem in NP. There are many such problems, all equivalent in hardness to each other, that are the (or more precisely, a) hardest problem in NP.

1.4.3 Hamiltonian complexity and QMA-completeness

The primary interest of quantum complexity theory in this thesis will be in Chapters 6 and 7, where we will focus on QMA-completeness. It is therefore worthwhile to note an important example of a QMA-complete problem here, namely the LOCAL HAMILTONIAN problem, which is particularly important to Hamiltonian-based quantum computation. The k-LOCAL HAMILTONIAN problem is the problem of estimating the lowest eigenvalue of a k-local Hamiltonian, or slightly more precisely, given a k-local Hamiltonian (in the sense described in Sec. 1.1.4), determining whether its lowest eigenvalue is below a specified value or else is above another value. A rigorous definition will be given in Chapter 7. This problem is QMA-complete for arbitrary k-local Hamiltonians with constant $k \geqslant 2$ [21]. It is in QMA because if given the *ground state* of the local Hamiltonian, i.e. the state with the lowest eigenvalue, it is easy to estimate its eigenvalue (using a "phase estimation" algorithm) and therefore verify whether the ground-state energy is sufficiently low. To prove that the problem is QMA-hard, one has to show that any QMA problem can be converted into a LOCAL HAMILTONIAN problem. To do this, one can use the Feynman-type Hamiltonian-based quantum computation method of Sec. 1.2.2: by definition, any QMA problem has a verifier circuit that accepts, with high probability, valid (and only valid) witness states, so one can construct a Feynman-style Hamiltonian based on this verifier circuit that has a sufficiently small lowest eigenvalue if and only if its ground state is a history state corresponding to the verification of a valid witness. Moreover, one can ensure that this Hamiltonian is local. Consequently, estimating the ground-state energy of this local Hamiltonian is equivalent to determining whether a valid witness exists, and therefore to solving the original QMA problem from which the Hamiltonian was constructed.

One can further ask whether the problem of estimating the ground-state energy remains QMA-complete when the type of Hamiltonian is even further restricted. That is, in addition to demanding locality, one may also stipulate other physical requirements of the Hamiltonian – such as that it represents a translationally invariant system of particles on a line or that it obeys the Bose-Hubbard model of Eq. (1.1) – and ask whether estimating the ground-state energy is still QMA-complete. These are important questions in the field of *Hamiltonian complexity*, which studies how "powerful" different Hamiltonians are. A question of particular interest is how powerful an adiabatic quantum computer (as described in Sec. 1.2.1) can be if its Hamiltonian is restricted to some physically-realizable form. As we noted, the adiabatic quantum computation model is theoretically universal for quantum computation, but if the type of Hamiltonian used is restricted, is this still true? There is a strong similarity between the Feynman-like circuit-to-Hamiltonian methods used for proving that various Hamiltonians are universal for adiabatic quantum computation and for proving that the LOCAL HAMILTONIAN problem remains QMA-complete when restricted to various Hamiltonian forms, so the study of these topics is highly interlinked. In this sense, the field of QMA-completeness is closely tied to the study of quantum computation with Hamiltonians.

1.5 Error-correcting codes

So far, in our discussions of classical and quantum computation, we have mostly dealt with the ideal scenario in which no errors occur. For many purposes in classical computing, this idealization is justifiable – the error rate of RAM in household computers is sufficiently low that we don't generally worry about it; however, in other cases, such as storing information on CDs, the possibility of errors needs to be taken into account. In quantum computation, errors are a severe issue, and their suppression and correction is a major area of research. This issue is particularly acute because states in superposition are often extremely fragile, and if they *decohere* and consequently "act classically", the advantage of quantum computation over classical computation can be lost. One way to detect and correct errors is by first *encoding* information. We now review this method, first for classical information and then for quantum information.

1.5.1 Classical linear error-correcting codes

Suppose that we have a *noisy* classical system of bits, so that each bit has some probability that its value will randomly flip $0 \leftrightarrow 1$. To protect our information, we use the idea of redundancy in order to encode each bit of information as several bits. If any one of these bits gets corrupted, we can detect that an error has occurred and potentially fix it. For example, we can use a 3-bit repetition code, mapping a single *logical* bit of information to three *physical* bits as

$$0 \to 000 \quad \text{and} \quad 1 \to 111\,.$$

If a bit flip occurs to any one bit of these three bits, the original information can still be recovered, by mapping

$$\begin{aligned} 000, 001, 010, 100 &\to 0 \\ 111, 110, 101, 011 &\to 1\,. \end{aligned}$$

Of course, if two bits had flipped, the recovery process will err, corrupting the data. But provided that the probability of a bit flip is sufficiently low, the probability of a double-bit-flip is very small, and we have reduced the overall probability of error.

This 3-bit repetition code is very simple, encoding a single bit as three physical bits, but more sophisticated methods can be used for encoding larger numbers of bits in efficient ways. The key requirement is that when decoding corrupted encoded messages, one can determine the original message, distinguishing it from other potential messages. One type of classical code that is particularly important, especially in the context of this thesis, is that of *linear codes*. A linear code can be understood as a code that uses a linear transformation, i.e. a matrix, to encode messages. For example, for the 3-bit repetition code, we can see that the matrix

$$G = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

maps the vector 0 to $\left(\begin{smallmatrix}0\\0\\0\end{smallmatrix}\right)$ and 1 to $\left(\begin{smallmatrix}1\\1\\1\end{smallmatrix}\right)$, as described above.

More formally, a classical linear $[n, k]$ code maps k-bit messages into n-bit messages. The number n is called the *length* of the code, while k is called its *dimension.* The matrix G that is associated with the code is an $n \times k$ matrix over $\mathbb{Z}_2$ (i.e. of 0s and 1s) called the *generator matrix.* If $x \in \mathbb{Z}_2^k$ is a message then its encoded form is $Gx \in \mathbb{Z}_2^n$, where all operations are performed mod 2, and is called a *codeword.* The space of all possible codewords, $C = \{Gx : x \in \mathbb{Z}_2^k\}$, is called the *codespace*, and is a vector space of dimension k.

Any $n \times k$ matrix G defines a linear code, provided that the columns of G are linearly independent (otherwise the encodes messages would not be unique), but not all such codes are useful. For the code to be useful, the encoded messages must be sufficiently different from each other (as 000 and 111 are from each other) so that bit-flip errors do not corrupt the information. To consider the effectiveness of the code, we make the following definitions.

Define the *weight* of a vector $x = (x_1, \ldots, x_n)^T \in \mathbb{Z}_2^n$, to be the number of non-zero bits in x, i.e. $\mathrm{wt}(x) = |\{i \in \{1, \ldots, n\} : x_i \neq 0\}|$. The *distance between two vectors* $x, y \in \mathbb{Z}_2^n$ is the weight of their difference, $\mathrm{wt}(x - y)$, i.e. the number of coordinates in which x and y differ. The *distance of a linear code* is defined to be $d = \min\{\mathrm{wt}(c) : c \in C, c \neq o\}$, where o denotes the zero vector. That is, the distance of a linear code is the minimum distance of any non-zero vector from the zero vector. Equivalently (by linearity), this is the minimum distance between any two distinct codewords of the code. An $[n, k]$ code with distance d is also called an $[n, k, d]$ code, explicitly specifying its distance.

Suppose x is a codeword of an $[n, k, d]$ code and suppose that we flip $t > 0$ of its bits, giving us a new vector x'. Provided that $t \leqslant d - 1$, the resulting vector x' cannot be a codeword (otherwise this codeword would be distance $t < d$ from x, contradicting the definition of d). Thus, by simply checking whether the encoded message is a codeword or not, we can detect that an error involving fewer than d bit-flips has occurred. We can therefore say that this code is a $(d-1)$-bit error-*detecting* code.

Moreover, if $2t + 1 \leqslant d$, we can not only detect, but also correct the error, as we now explain. Since $t < d$ we can certainly tell that an error has occurred, as x' will not be a codeword. We can then ask what the original codeword that gave rise to x' could be. Suppose that we guess that y was the original codeword, where y is any codeword that could give rise to x' with at most t bit-flips. Then because x and x' differ in at most t coordinates, and because y and x' can differ in at most t coordinates, x and y can differ in at most $2t$ coordinates. That is, the distance between the codewords x and y is at most $2t < 2t + 1 \leqslant d$. Thus, by definition of d, we must have $y = x$, i.e. there is only one possible candidate codeword that could have given rise to x', and so we can (in principle) determine the original encoded message x. We can therefore say that this code is a $\left(\frac{d-1}{2}\right)$-bit error-*correcting* code. We see clearly that the distance of the code determines how useful the code is for error detection and correction.

The above discussion is for linear codes over $\mathbb{Z}_2$, i.e. involving bits (0 and 1). One can also speak more generally of linear codes over any finite field $\mathbb{F}$, but the ideas are similar. This more general formulation will be used (and precisely defined) in Chapter 4. Also note that we have only discussed how to efficiently *encode* data using linear codes. An equally important aspect of linear codes is being able to then efficiently *decode* the encoded messages. We will not make use of decoding in this thesis; interested readers may consult [1] or any number of resources on the topic of linear codes.

1.5.2 Quantum error-detecting/correcting codes

The theory of classical error-detecting codes does not directly apply to the quantum setting, largely because in the quantum case more can go wrong than simple bit-flips. The states $|000\rangle + |111\rangle$ and $|000\rangle - |111\rangle$ are distinct (and in fact orthogonal) states, even though they are not related by exchanging any number of 0s and 1s; the error that occurs in going from one of these states to the other is a *phase-flip*, flipping the relative phase in the superposition (in the computational basis). Indeed, a bit-flip on a single qubit corresponds to applying the Pauli operator X, since it interchanges the states $|0\rangle \leftrightarrow |1\rangle$, but we can also imagine an error caused by applying the Z operation,

$$Z(a|0\rangle + b|1\rangle) = a|0\rangle - b|1\rangle,$$

i.e. a phase-flip. We could further have an error caused by applying $Y = iXZ$, which is equivalent (up to phase) to applying a Z and X error sequentially. Most generally (but ignoring irrelevant global phase), we could write any single-qubit unitary error as $e^{i\epsilon\hat{n}\cdot\vec{\sigma}} = \cos(\epsilon)\,\mathbb{1} + i\sin(\epsilon)\hat{n}\cdot\vec{\sigma}$ for some ϵ and unit vector $\hat{n}$. At this point it may seem that quantum computation is like analogue classical computation rather than digital computation, suffering from continuous noise and unpredictability.

Fortunately, a rich theory of a quantum error-correcting codes has been developed. In this thesis, we will need very little of this theory, and therefore will content ourselves here to give an example of an error-correcting code and to address a few of the features that we will need in later chapters. We do note that it suffices to be able to detect (and correct) X and Z errors. The idea is as follows. Suppose we could detect any arbitrary single-qubit X and/or Z error. Then if a state $|\psi\rangle$ has an arbitrary error $e^{i\epsilon\hat{n}\cdot\vec{\sigma}}\,|\psi\rangle$ applied, the state becomes

$$\cos(\epsilon)\,|\psi\rangle + i\sin(\epsilon)n_x X\,|\psi\rangle + i\sin(\epsilon)n_z Z\,|\psi\rangle - \sin(\epsilon)n_y XZ\,|\psi\rangle\,.$$

This state can be interpreted as a superposition over four states, namely the states in which one of the following four errors is applied to $|\psi\rangle$: no error, an X error, a Z error, or both a Z and X error. If we can perform a measurement to detect X and/or Z errors, this superposition will collapse into the state corresponding to the error we determined, and therefore the state truly becomes $E\,|\psi\rangle$ with the error (or lack thereof) $E \in \{\mathbb{1}, X, Z, XZ\}$ that we ascertained from our measurement. In other words, the ability to detect arbitrary single-qubit errors is equivalent to the ability to detect single-qubit Pauli (X, Y, Z) errors.

Shor's 9-qubit code

Recall that in the classical case, an $[n, k]$ code encodes k-bit messages into n-bit messages. We use a similar notation in the quantum case, that an $[[n, k]]$ quantum code encodes k qubits as n qubits. Consider the following $[[9, 1]]$ code mapping 1 qubit into 9 qubits as

$$\begin{aligned}
|0\rangle \to |0_L\rangle &= \frac{1}{\sqrt{8}}(|000\rangle + |111\rangle)(|000\rangle + |111\rangle)(|000\rangle + |111\rangle) \\
|1\rangle \to |1_L\rangle &= \frac{1}{\sqrt{8}}(|000\rangle - |111\rangle)(|000\rangle - |111\rangle)(|000\rangle - |111\rangle)\,.
\end{aligned}$$

We say that a state in $\{a|0_L\rangle + b|1_L\rangle\}$ is a *logical qubit* or *logical state*, composed of 9 *physical qubits*: such as state is physically 9 qubits large, but we regard it as representing only a single qubit of ("logical") information. Observe that both $|0_L\rangle$ and $|1_L\rangle$ are $+1$ eigenstates of the operators

$$Z_1Z_2, \quad Z_4Z_5, \quad Z_7Z_8,$$

$$Z_2Z_3, \quad Z_5Z_6, \quad Z_8Z_9,$$

$$X_1X_2X_3X_4X_5X_6, \quad X_4X_5X_6X_7X_8X_9,$$

where X_i denotes the Pauli X operator on qubit i (and similarly for Z_i). Therefore, by measuring any encoded state $a|0_L\rangle + b|1_L\rangle$ using these operators, the state will not be modified, and a value of $+1$ will be obtained. However, if a bit-flip error X_i occurs on qubit i, the state will become a -1 eigenstate of at least one of the above operators, namely the ZZ operator(s) that involve qubit i. Similarly, if a phase-flip error Z_i occurs on qubit i, the state will become a -1 eigenstate of the $XXXXXX$ operator(s) that involve qubit i. Thus, by measuring the state using these operators, one can ascertain whether a bit-flip or a phase-flip has occurred (by the presence of the -1 measured value for the appropriate operator). This encoding is evidently a quantum error-detecting code.

Moreover, it is not hard to see that that this code is a quantum error-*correcting* code, capable of correcting arbitrary single-qubit errors. Suppose, for example, that measurement of these operators yielded $+1$, except for the Z_1Z_2 and Z_2Z_3 operators, which yielded -1. Suppose further that we knew that the error was a single-qubit bit-flip and/or phase-flip error on one of the qubits. Then the error must have been a bit-flip on qubit 2, because that is the only error that would be consistent with our measurements. Applying X_2 therefore restores the original state. Thus we have not only been able to detect that an error has occurred, but have obtained enough information to *correct* the error as well, without damaging the state.

Codespace projectors

An alternative way of viewing a quantum code is in terms of its codespace and its codespace projector. Consider a single-qubit code, mapping $|0\rangle$ and $|1\rangle$ to $|0_L\rangle$ and $|1_L\rangle$, respectively. The codespace, i.e. the vector space of all valid codewords formed by this mapping, is $C = \{a|0_L\rangle + b|1_L\rangle : |a|^2 + |b|^2 = 1\}$. Suppose that the code can *detect* Pauli errors, by which we mean that any Pauli error $\sigma \in \{X, Y, Z\}$ takes codewords that are in this codespace to states that are orthogonal to the codespace:

$$|\psi\rangle \in C \text{ implies that } \langle\phi|\big(\sigma\,|\psi\rangle\big) = 0 \text{ for all } |\phi\rangle \in C \text{ and } \sigma \in \{X, Y, Z\}\,.$$

Measuring whether a given state is in the codespace thus reveals whether a quantum error has occurred. Note that if no error occurred, the state remains unaffected by this measurement (since the measurement merely reveals whether the state is in the codespace). However, if a Pauli error has occurred, this will be detected. It is in this sense that the code is a quantum error-detecting code.

It will be useful to consider the codespace projection operator,

$$P = |0_L\rangle\langle 0_L| + |1_L\rangle\langle 1_L|\,,$$

of the quantum error-detecting code. For any valid codeword $|\psi\rangle = a|0_L\rangle + b|1_L\rangle$, we have $P\,|\psi\rangle = |\psi\rangle$, so $|\psi\rangle$ has P-eigenvalue 1; however, if a Pauli error σ occurred then we have

$$P\sigma\,|\psi\rangle = 0 \text{ for } \sigma \in \{X, Y, Z\}\,,$$

so $P\sigma\,|\psi\rangle$ has P-eigenvalue 0. Thus, measuring the value of the observable P, i.e. whether the state is in the codespace, reveals whether an error has occurred, with 0 indicating an error and 1 indicating no error. Since this is true for any $|\psi\rangle \in C$, we can equivalently write the condition of our error-detecting code as

$$P\sigma P = 0 \text{ for } \sigma \in \{X, Y, Z\}.$$

We note in passing that this condition is actually more stringent than is necessary – this condition is sufficient, not necessary – but we adopt this viewpoint for simplicity. We also note that there is a similar condition for quantum error-*correcting* codes, but we need not elaborate on this here since we do not make use of it in this thesis. For more details, consult [1].

In addition to the codespace projection operator, used for detecting whether a state is a codeword, we can also consider other useful operators that act on codewords. Consider a single qubit encoded using a quantum error-detecting code. Some very useful operations on a single qubit are the Pauli operators X and Z (and $Y = iXZ$) that map

$$\begin{aligned} X|0\rangle &= |1\rangle, \qquad & Z|0\rangle &= |0\rangle, \\ X|1\rangle &= |0\rangle, \qquad & Z|1\rangle &= -|1\rangle. \end{aligned}$$

We can also find operations on the codewords (i.e. on logical qubits) in the same way; such operators are called *logical operators*, as they act on the logical states in the same way as their counterparts do on unencoded states. The logical operators X_L and Z_L corresponding to X and Z satisfy

$$\begin{aligned} X_L|0_L\rangle &= |1_L\rangle, \qquad & Z_L|0_L\rangle &= |0_L\rangle, \\ X_L|1_L\rangle &= |0_L\rangle, \qquad & Z_L|1_L\rangle &= -|1_L\rangle. \end{aligned}$$

These operators are generally not unique but always exist; indeed, the following are immediate tautological examples:

$$X_L = |0_L\rangle\langle 1_L| + |1_L\rangle\langle 0_L|\,, \quad Z_L = |0_L\rangle\langle 0_L| - |1_L\rangle\langle 1_L|\,, \quad \text{and} \quad Y_L = iZ_LX_L.$$

In the circuit model, where useful operators are generally unitary, we would choose logical operators that are unitary (unlike the examples give above). Such logical operators, together with special methods of measurement and state preparation, are important ingredients in *fault-tolerant* circuit-model quantum computation, where it has been shown that, provided error rates are sufficiently low, arbitrary precision can efficiently be achieved in the quantum circuit-model. In Hamiltonian-based quantum computing, useful operators are generally Hermitian, so we would choose logical operators that are Hermitian (like the examples above, although these examples are not necessarily the best choice). Although no fault-tolerant theorems are known for Hamiltonian-based quantum computation, we will make use of Hermitian logical operators in the next chapter, where we will prove how quantum error-detecting codes can be used to suppress errors in Hamiltonian-based quantum computation.

Chapter bibliography

[1] M. A. Nielsen and I. L. Chuang. *Quantum computation and quantum information*. Cambridge University Press, Cambridge, UK [2000]

[2] J. Preskill. *Lecture notes for Physics 229: Quantum information and computation*. `http://www.theory.caltech.edu/people/preskill/ph229/`

[3] S. Blanes, F. Casas, J. A. Oteo, and J. Ros. *The Magnus expansion and some of its applications*. Physics Reports, 470(5–6):151 [2009]. `http://dx.doi.org/10.1016/j.physrep.2008.11.001`

[4] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser. *Quantum computation by adiabatic evolution*. arXiv:quant-ph/0001106 [2000]. `http://arxiv.org/abs/quant-ph/0001106`

[5] M. Born and V. Fock. *Beweis des Adiabatensatzes*. Z. Physik, 51(3-4):165 [1928]. `http://dx.doi.org/10.1007/BF01343193`

[6] A. Messiah. *Quantum mechanics*, volume 2. John Wiley & Sons, New York, NY [1966]

[7] D. Aharonov, W. van Dam, J. Kempe, Z. Landau, S. Lloyd, and O. Regev. *Adiabatic quantum computation is equivalent to standard quantum computation*. SIAM J. Comput., 37(1):166 [2007]. `http://dx.doi.org/10.1137/S0097539705447323`

[8] X. Peng, Z. Liao, N. Xu, G. Qin, X. Zhou, D. Suter, and J. Du. *Quantum adiabatic algorithm for factorization and its experimental implementation*. Phys. Rev. Lett., 101(22):220405 [2008]. `http://dx.doi.org/10.1103/PhysRevLett.101.220405`

[9] N. S. Dattani and N. Bryans. *Quantum factorization of 56153 with only 4 qubits*. arXiv:1411.6758 [2014]. `http://arxiv.org/abs/1411.6758`

[10] M. W. Johnson, M. H. S. Amin, S. Gildert, T. Lanting, F. Hamze, N. Dickson, R. Harris, A. J. Berkley, J. Johansson, P. Bunyk, E. M. Chapple, C. Enderud, J. P. Hilton, K. Karimi, E. Ladizinsky, N. Ladizinsky, T. Oh, I. Perminov, C. Rich, M. C. Thom, E. Tolkacheva, C. J. S. Truncik, S. Uchaikin, J. Wang, B. Wilson, and G. Rose. *Quantum annealing with manufactured spins*. Nature, 473(7346):194 [2011]. `http://dx.doi.org/10.1038/nature10012`

[11] R. P. Feynman. *Quantum mechanical computers*. Found Phys, 16(6):507 [1986]. `http://dx.doi.org/10.1007/BF01886518`

[12] N. Margolus. *Quantum qomputation*. Annals of the New York Academy of Sciences, 480(1):487 [1986]. `http://dx.doi.org/10.1111/j.1749-6632.1986.tb12451.x`

[13] S. Lloyd and B. M. Terhal. *Adiabatic and Hamiltonian computing on a 2D lattice with simple two-qubit interactions.* New J. Phys., 18(2):023042 [2016]. `http://dx.doi.org/10.1088/1367-2630/18/2/023042`

[14] E. Farhi and S. Gutmann. *Quantum computation and decision trees.* Phys. Rev. A, 58(2):915 [1998]. `http://dx.doi.org/10.1103/PhysRevA.58.915`

[15] A. M. Childs. *Universal computation by quantum walk.* Phys. Rev. Lett., 102(18):180501 [2009]. `http://dx.doi.org/10.1103/PhysRevLett.102.180501`

[16] A. M. Childs, D. Gosset, and Z. Webb. *Universal computation by multiparticle quantum walk.* Science, 339(6121):791 [2013]. `http://dx.doi.org/10.1126/science.1229957`

[17] B. Reichardt and R. Spalek. *Span-program-based quantum algorithm for evaluating formulas.* Theory of Computing, 8(1):291 [2012]. `http://dx.doi.org/10.4086/toc.2012.v008a013`

[18] S. Lloyd. *Universal quantum simulators.* Science, 273(5278):1073 [1996]. `http://dx.doi.org/10.1126/science.273.5278.1073`

[19] E. Farhi and S. Gutmann. *Analog analogue of a digital quantum computation.* Phys. Rev. A, 57(4):2403 [1998]. `http://dx.doi.org/10.1103/PhysRevA.57.2403`

[20] L. K. Grover. *Quantum mechanics helps in searching for a needle in a haystack.* Phys. Rev. Lett., 79(2):325 [1997]. `http://dx.doi.org/10.1103/PhysRevLett.79.325`

[21] J. Kempe, A. Kitaev, and O. Regev. *The complexity of the local Hamiltonian problem.* SIAM J. Comput., 35(5):1070 [2006]. `http://dx.doi.org/10.1137/S0097539704445226`

Chapter 2

Error suppression in Hamiltonian-based quantum computation using energy penalties

In this chapter, we consider the use of quantum error-detecting codes, together with energy penalties against leaving the codespace, as a method for suppressing environmentally induced errors in Hamiltonian-based quantum computation. This method was introduced in [1] in the context of quantum adiabatic computation, but we consider it more generally. Specifically, we consider a computational Hamiltonian, which has been encoded using the logical qubits of a single-qubit error-detecting code, coupled to an environment of qubits by interaction terms that act one-locally on the system. Additional energy penalty terms penalize states outside of the codespace. We prove that in the limit of infinitely large penalties, one-local errors are completely suppressed (Theorem 2.1), and we derive some bounds for the finite penalty case [Eqs. (2.30) and (2.32)]. In particular, we show that, under reasonable physical assumptions, to attain good protection it suffices for the energy penalty to grow polynomially in the size of the system. Our proof technique involves exact integration of the Schrödinger equation, making no use of master equations or their assumptions. We perform long-time numerical simulations on a small (one logical qubit) computational system coupled to an environment and the results suggest that the energy penalty method achieves even greater protection than our bounds indicate.

This chapter is adapted from [2], which was joint work with Edward Farhi and Leo Zhou.

2.1 Introduction

A major problem on the road to building scalable quantum computers is the difficult task of protecting the system from errors, such as those due to unwanted environmental interactions. In the usual circuit model of quantum computation, the theory of quantum error correction has been well-developed, culminating in the threshold theorem [3–7], which proves that, provided the error rate in a quantum computing system can be reduced to below a certain threshold, errors can be suppressed arbitrarily well using quantum error-correcting codes. The situation for the Hamiltonian model of quantum computation as used in, for example, adiabatic quantum computing, continuous-time quantum walks, and Hamiltonian simulation problems, is less understood and no fault-tolerant theorem is known. In this chapter, we take steps towards establishing such a theorem.

In the Hamiltonian model, the computational system is described by a Hamiltonian, which is a (possibly time-dependent) Hermitian operator, H_{comp}, that governs the time-evolution of the system according to

$$i\frac{\mathrm{d}}{\mathrm{d}t}|\phi(t)\rangle = H_{\mathrm{comp}}(t)|\phi(t)\rangle\,,$$

where $|\phi(t)\rangle$ is the state of the computational system at time t. In this model, the goal is to evolve some initial state $|\phi(0)\rangle$ to a final state $|\phi(T)\rangle$, the measurement of which reveals some information about the problem to be solved. Note that no instantaneous unitary gates are applied, nor are any intermediate measurements performed. To consider the effects of unwanted environmental interaction, one must consider the Hamiltonian $H_{\mathrm{comp}}+H_{\mathrm{environment}}+H_{\mathrm{interaction}}$ that governs the evolution of the entire system-environment supersystem. The goal of error suppression is to ensure that the state of the system at time T is approximately as though the evolution had been governed by just H_{comp} alone.

It is not clear how to adapt the successful error-correcting code techniques of the circuit model to the Hamiltonian model. In a conventional quantum error-correcting code [8], each qubit is encoded as a logical qubit, comprised of several physical qubits, so that the occurrence of any single-qubit error on any physical qubit can be detected. The use of such a code in the error-correcting circuit model essentially consists of four steps: the state is encoded, the state is allowed to evolve, a measurement is made to determine what error has occurred (if any), and gates are applied to correct that error. In our Hamiltonian model, we do not allow intermediate measurements or the application of instantaneous gates, and therefore rule out any active determination and correction of errors; thus, a different strategy is required.

The error suppression strategy used in this chapter is that of *energy penalties*, first suggested in [1], in which the system Hamiltonian is modified according to a quantum error-detecting code and a constant (time-independent) term is added to the Hamiltonian. This extra term, the energy penalty, penalizes states that have been corrupted by, say, single-qubit errors. It is believed that such a penalty will suppress the occurrence of environmentally induced errors, as it imposes an energy barrier that must be surmounted for an error to occur. In this work, we prove that, in principle, this energy penalty method does indeed work; we show that it successfully suppresses errors arbitrarily well when the penalty is arbitrarily large. (Throughout the chapter we concentrate on 1-local errors and use a 1-qubit error-detecting code. In the appendix, however, we show that this result can be generalized to k-local errors when using a k-qubit error-detecting code.) We also explore (in the 1-local error case) how well the penalty terms work when the penalty is not infinite but of a reasonable size. We then show the results of small-system numerical simulations that suggest that the achieved protection is even better than our bounds can predict.

We note that since we will not be performing active error correction, we do not need an error-*correcting* code, which gives information about which error occurred; rather, it suffices to use an error-*detecting* code, which only detects whether any error has occurred.

An error suppression technique using energy penalties has recently [9] been applied to the quantum annealing paradigm of computation. However, that method differs from the one discussed in this chapter, as its energy penalty does not suppress phase-flip errors, while the penalty used in this chapter suppresses arbitrary single-qubit errors. Other previously suggested Hamiltonian-model error-suppression methods include exploiting the Zeno effect [10, 11] and using dynamical decoupling [12–15]. Some of these techniques require intermediate measurements, which is outside of the Hamiltonian paradigm, or require the ability to add rapidly time-dependent control terms to the Hamiltonian. The energy penalty method used in this chapter remains in the Hamiltonian model paradigm and requires only the addition of a constant term to the encoded Hamiltonian. It would therefore be useful even when intermediate measurements and fast, active control are not available. A discussion of the similarity between the energy penalty, Zeno, and dynamical decoupling methods can be found in [16, 17].

2.2 Quantum error-detecting codes

We first review some basic facts about quantum error-detecting codes. Suppose that we have an $[[\ell, 1]]$ quantum error-detecting code, meaning that by encoding a single qubit as a logical qubit comprised of ℓ physical qubits, we can detect arbitrary 1-qubit errors. Throughout this chapter, we use this code to protect our system of n qubits, meaning that each qubit of the original H_{comp} is encoded to be ℓ qubits, so that the full encoded system consists of $n_s = \ell n$ qubits.

Specifically, for each qubit register i, the original computational basis states $|0\rangle_i$ and $|1\rangle_i$ are encoded as the ℓ-qubit logical states $|0_L\rangle_i$ and $|1_L\rangle_i$. The codespace of the ith logical qubit is then the span of the logical states, $\{a|0_L\rangle_i + b|1_L\rangle_i : |a|^2 + |b|^2 = 1\}$. Associated with this codespace is the projection operator

$$P_i = |0_L\rangle\langle 0_L|_i + |1_L\rangle\langle 1_L|_i \ ,$$

where P_i acts as the identity on all physical qubits other than those associated with the logical qubit i. Note that states in the codespace are invariant under P_i, whereas P_i kills states that are orthogonal to the codespace of the ith qubit.

Saying that the code can detect arbitrary 1-qubit errors is equivalent to saying that the code detects all single-qubit Pauli errors, i.e. an error caused by the application of a Pauli operator (X, Y, or Z) to any single physical qubit. Thus, for any single Pauli operator σ acting on one of the ℓ physical qubits comprising logical qubit i, we have

$$P_i \sigma P_i = 0 \,. \tag{2.1}$$

The full codespace for the entire logical space (over all n logical qubits) corresponds to the projector

$$P = P_1 P_2 \cdots P_n \,. \tag{2.2}$$

The quantum code also allows us to "encode" the Pauli operators X, Y, and Z as *logical operators* X_L, Y_L, and Z_L. Logical operators are Hermitian operators that have the same effect on the logical basis states as their corresponding Pauli operators have on the corresponding basis states. Furthermore, the logical operators associated with qubit i commute with the codespace projector P_i, i.e. $X_L P_i = P_i X_L$, and similarly for Y_L and Z_L.

As a concrete example, consider the 4-qubit code of Jordan-Farhi-Shor [1], in which

$$\begin{aligned}
|0_L\rangle &= \frac{1}{2}\Big(|0000\rangle + i|0011\rangle + i|1100\rangle + |1111\rangle\Big) \\
|1_L\rangle &= \frac{1}{2}\Big(- |1010\rangle + i|1001\rangle + i|0110\rangle - |0101\rangle\Big)
\end{aligned}$$

and

$$
\begin{aligned}
X_L &= \quad Y \otimes \mathbb{1} \otimes Y \otimes \mathbb{1} \\
Y_L &= -\mathbb{1} \otimes X \otimes X \otimes \mathbb{1} \\
Z_L &= \quad Z \otimes Z \otimes \mathbb{1} \otimes \mathbb{1} \ .
\end{aligned}
$$

Observe that the logical operators have the same effect on logical qubits as do the operators to which they correspond have on unencoded qubits; e.g. $X_L|0_L\rangle = |1_L\rangle$.

Using the logical operators, we can encode the Hamiltonian that acts on the system. Suppose that H_{comp} is some Hermitian operator on the original (n-qubit) system. Because the Pauli matrices (along with the identity) form a basis for all 2×2 matrices, we may generically write

$$
H_{\text{comp}}(t) = \sum_{\substack{\sigma_i \in \{\mathbb{1}, X, Y, Z\} \\ i=1,\ldots,n}} c_{\sigma_1,\ldots,\sigma_n}(t)\ \sigma_1 \otimes \cdots \otimes \sigma_n,
$$

where the sum is over all possible choices of $\sigma_i \in \{\mathbb{1}, X, Y, Z\}$ for each i. We may therefore encode the Hamiltonian by replacing X, Y, Z with X_L, Y_L, Z_L in the sum above, to obtain

$$
H^{\text{L}}_{\text{comp}}(t) = \sum_{\substack{\sigma_i \in \{\mathbb{1}, X_L, Y_L, Z_L\} \\ i=1,\ldots,n}} c_{\sigma_1,\ldots,\sigma_n}(t)\ \sigma_1 \otimes \cdots \otimes \sigma_n,
$$

which is a Hamiltonian on the n_s-qubit encoded space built entirely out of logical operators (and $\mathbb{1}$). Since each logical operator commutes with each P_i, $H^{\text{L}}_{\text{comp}}$ also commutes with each P_i and with P.

Observe that the logical operators in the Jordan-Farhi-Shor code are all 2-local. The encoding in this case thus doubles the locality of the original Hamiltonian, so that if the original Hamiltonian is 2-local, the encoded one is 4-local. As [1] points out, such an encoding is optimal (in terms of locality) if the code is to protect against arbitrary 1-qubit errors.

2.3 The Hamiltonian model and energy penalties

In this chapter we consider a system coupled to an environment. We do not attempt to modify the environment or the system-environment interaction. However, we assume that we can modify the Hamiltonian of the system, and do so in two ways. As just discussed, we encode the original computational Hamiltonian in a quantum code. Furthermore, we add extra terms (acting only on the system) that penalize system states that are outside of the codespace.

The combined system-environment Hamiltonian, H, after encoding and penalty modifications, consists of three parts, and can be written as

$$
H = H_0 + \lambda V + E_P \tilde{Q} \, .
$$

We discuss each of these parts in turn.

1. The first term is
$$
H_0 = H^{\text{L}}_{\text{comp}} \otimes \mathbb{1}_{\text{env}} + \mathbb{1}_{\text{sys}} \otimes H_{\text{env}} \, ,
$$
which governs the evolution in the absence of any system-environment interaction. Both $H^{\text{L}}_{\text{comp}}$ and H_{env} are in general time-dependent. Evolution under $H^{\text{L}}_{\text{comp}}$ alone is

equivalent to evolution under H_{comp} and represents the desired evolution we wish to protect.

Because the system Hamiltonian is encoded, the system consists of $n_s = \ell n$ qubits. The size of the environment will play no role in our discussion, except when we do simulations, and can be thought of as much larger than the system size.
Recall that $H^{\text{L}}_{\text{comp}}$ is built up from only logical operators and therefore commutes with each P_i. Since H_{env} (which acts only on the environment) trivially commutes with each P_i, we have

$$[H_0, P_i] = 0 \quad \text{for } i = 1, \ldots, n.$$

2. λV is the error Hamiltonian, reflecting the coupling of the system to the environment, with λ serving as a time-independent (presumably small) parameter indicating the strength of the interaction (with units of energy), and V a Hermitian operator acting on the full system-environment space. Our code is designed to protect against 1-qubit errors, so we assume a 1-qubit error model, i.e. that V acts 1-locally on the system. Thus, we can write V as a sum of terms

$$V = \sum_{s=1}^{n_s} \sum_{\mu=X,Y,Z} \sigma^s_\mu \otimes B^s_\mu \tag{2.3}$$

where σ^s_μ is the μ^{th} Pauli matrix acting on physical system qubit s and each B^s_μ is some operator acting on a set of environmental qubits. We also allow the possibility that $B^s_\mu = \mathbb{1}$, which could represent 1-local system control errors independent of the environment.

For convenience, we group terms in V according to the logical system qubit on which they act, so that

$$V = \sum_{i=1}^{n} V_i \tag{2.4}$$

where each V_i is an operator whose 1-local action on the system is only on the ℓ system qubits that comprise logical qubit i. Observe that V_i causes 1-local errors on the system, as per Eq. (2.3), and that we are using a code that can detect arbitrary 1-qubit errors, as per Eq. (2.1). Thus, we have that

$$P_i V_i P_i = 0 \quad \text{for } i = 1, \ldots, n$$

which is crucial to our later analysis.

3. $E_P \tilde{Q}$ is our time-independent energy penalty, which penalizes states outside of the codespace. Specifically, E_P is a real constant with units of energy and $\tilde{Q}$ is the sum of the projectors $Q_i = \mathbb{1} - P_i$, i.e.

$$\tilde{Q} = \sum_{i=1}^{n} Q_i = \sum_{i=1}^{n} (\mathbb{1} - P_i)\,, \tag{2.5}$$

so we have a separate energy penalty for each logical qubit. In this context, $\tilde{Q}$ is to be understood as $\tilde{Q} \otimes \mathbb{1}_{\text{env}}$, since only the system is encoded. Observe that a state $|\psi\rangle$ is in the codespace if and only if $\tilde{Q}\,|\psi\rangle = 0$, so $E_P \tilde{Q}$ applies an energy penalty of magnitude at least $|E_P|$ to states outside of (i.e. orthogonal to) the codespace.

We point out that $\tilde{Q}$ is the sum of codespace projectors, differing from the penalty used in [1] which is a sum of codespace stabilizer generators. Note that the locality of $\tilde{Q}$ is that of each P_i, which is at most ℓ (i.e. 4 in the case of the Jordan-Farhi-Shor code).

The key point in this model is that V acts precisely 1-locally on the system and we are using a quantum code that can detect 1-qubit system errors. This enables us to penalize the states that arise from the action of V, and therefore have hope of suppressing V's effect. We can similarly consider the case in which V acts k-locally on the system as long as the quantum code can detect k-qubit errors. However, we consider only the 1-local case throughout the main text of the chapter, leaving the more general case to the appendix.

2.4 Error suppression through energy penalties

2.4.1 The infinite E_P case

We first address the question of whether adding an energy penalty works even in principle; that is, we want to show that if E_P is arbitrarily large, errors are suppressed arbitrarily well. Let $U_0(t)$ and $U(t)$ be the evolution operators corresponding to the desired Hamiltonian, $H_0 = H^{\mathrm{L}}_{\mathrm{comp}} + H_{\mathrm{env}}$, and the actual Hamiltonian, $H = H_0 + \lambda V + E_P\tilde{Q}$, respectively. That is, $U_0(t)$ and $U(t)$ obey

$$\begin{aligned} i\frac{\mathrm{d}}{\mathrm{d}t}U_0(t) &= H_0(t)U_0(t), \qquad U_0(0) = \mathbb{1} \\ i\frac{\mathrm{d}}{\mathrm{d}t}U(t) &= H(t)U(t), \qquad\quad U(0) = \mathbb{1}\ . \end{aligned} \tag{2.6}$$

We wish to show that in the codespace, as $E_P \to \infty$, $U(t)$ acts as $U_0(t)$. Our approach will be to show that the error induced by V is modulated by a term oscillating with frequency E_P in such a way so that for large E_P such errors are suppressed.

Our first step is to view λV as a perturbation and to work in the interaction picture using

$$H_{0P}(t) = H_0(t) + E_P\tilde{Q}$$

as the reference Hamiltonian. This corresponds to the evolution operator $U_{0P}(t)$, which obeys

$$i\frac{\mathrm{d}}{\mathrm{d}t}U_{0P}(t) = H_{0P}(t)U_{0P}(t), \quad U_{0P}(0) = \mathbb{1}\ .$$

Because H_0 commutes with each P_i, and therefore with $\tilde{Q}$, we have that

$$U_{0P}(t) = U_0(t)U_P(t)\,, \tag{2.7}$$

where the evolution operator due to the error penalty alone is

$$U_P(t) = e^{-iE_P\tilde{Q}t}\,.$$

Now, the interaction picture evolution operator

$$U_I \equiv U_{0P}^{\dagger}U$$

obeys

$$i\frac{\mathrm{d}}{\mathrm{d}t}U_I = \lambda V_I U_I\,,$$

where

$$\lambda V_I(t) = \lambda U_{0P}^\dagger(t)V(t)U_{0P}(t)\,. \tag{2.8}$$

These are just the usual interaction picture equations with a reference Hamiltonian H_{0P} and a perturbation λV. Taking conjugates, we get

$$U_I^\dagger = U^\dagger U_{0P} = U^\dagger U_0 U_P \tag{2.9}$$

and

$$\frac{\mathrm{d}}{\mathrm{d}t}U_I^\dagger = i\lambda U_I^\dagger V_I\,, \tag{2.10}$$

which upon integration gives

$$U_I^\dagger(T) = \mathbb{1} + i\lambda\int_0^T U_I^\dagger V_I \mathrm{d}t\,. \tag{2.11}$$

Note that $\tilde{Q}P = 0$, so

$$U_P(t)P = e^{-iE_P\tilde{Q}t}P = P \tag{2.12}$$

and therefore

$$U_I^\dagger P = U^\dagger U_0 P\,.$$

Now, we multiply Eq. (2.11) on the right by P, and use this last relation, to get

$$U^\dagger(T)U_0(T)P = P + i\lambda\int_0^T U_I^\dagger V_I P\mathrm{d}t\,.$$

Multiplying this by $U(T)$ gives

$$U_0(T)P = U(T)P + i\lambda U(T)\int_0^T U_I^\dagger V_I P\mathrm{d}t\,, \tag{2.13}$$

which we can use to track the difference between the evolutions (in the codespace) with and without the coupling to the environment. Our goal is to show that as E_P goes to infinity, this difference goes to zero. To this end, let

$$F(t) = \int_0^t V_I(\tau)P\mathrm{d}\tau\,. \tag{2.14}$$

Using integration by parts, we see that

$$\begin{aligned}\int_0^T U_I^\dagger V_I P\mathrm{d}t = \int_0^T U_I^\dagger\frac{\mathrm{d}F}{\mathrm{d}t}\mathrm{d}t &= U_I^\dagger(T)F(T) - \int_0^T \frac{\mathrm{d}U_I^\dagger}{\mathrm{d}t}F\mathrm{d}t\\ &= U_I^\dagger(T)F(T) - i\lambda\int_0^T U_I^\dagger V_I F\mathrm{d}t\,,\end{aligned}$$

where Eq. (2.10) was used to obtain the final equality. Applying Eqs. (2.9) and (2.8) we can write this as

$$\int_0^T U_I^\dagger V_I P \mathrm{d}t = U^\dagger(T) U_{0P}(T) F(T) - i\lambda \int_0^T U^\dagger V U_{0P} F \mathrm{d}t$$

and using this in Eq. (2.13) we find that

$$\boxed{U(T)P = U_0(T)P - i\lambda \Big[U_{0P}(T) F(T) - i\lambda U(T) \int_0^T U^\dagger V U_{0P} F \mathrm{d}t \Big]\,,} \tag{2.15}$$

which is an exact expression and not just an expansion in λ.

We now focus on the operator $F(t)$ defined in Eq. (2.14), which using Eq. (2.8) for V_I and Eq. (2.7) for U_{0P} is

$$F(t) = \int_0^t U_P^\dagger U_0^\dagger V U_0 U_P P \mathrm{d}\tau\,.$$

P commutes with H_0, and therefore also with U_0. Because of this and Eq. (2.12) we have

$$F(t) = \int_0^t U_0^\dagger e^{iE_P \tilde{Q}\tau} V P U_0 \mathrm{d}\tau\,.$$

Consider

$$e^{iE_P \tilde{Q}\tau} V P = e^{iE_P \tilde{Q}\tau} (V_1 + \cdots + V_n) P \tag{2.16}$$

where we have written V as the sum over terms associated with each logical qubit, as in Eq. (2.4). From the definitions in Eqs. (2.2) and (2.5), the first term is

$$e^{iE_P \tilde{Q}\tau} V_1 P = e^{iE_P Q_1 \tau} e^{iE_P Q_2 \tau} \cdots e^{iE_P Q_n \tau} V_1 P_1 \cdots P_n\,.$$

But, $P_2 P_3 \cdots P_n$ commutes with V_1, and $P_i Q_i = 0$ for all i, so we get

$$e^{iE_P \tilde{Q}\tau} V_1 P = e^{iE_P Q_1 \tau} V_1 P_1 \cdots P_n\,. \tag{2.17}$$

Our code protects against single-qubit errors and we are assuming that the coupling to the environment involves only single-qubit terms, so again,

$$P_1 V_1 P_1 = 0$$

which implies that

$$V_1 P_1 = Q_1 V_1 P_1\,. \tag{2.18}$$

Because Q_1 is a projector, we have that

$$e^{iE_P Q_1 \tau} Q_1 = e^{iE_P \tau} Q_1\,. \tag{2.19}$$

The previous equations combine to give

$$e^{iE_P \tilde{Q}\tau} V_1 P = e^{iE_P \tau} V_1 P$$

and accordingly,

$$e^{iE_P \tilde{Q}\tau} V P = e^{iE_P \tau} V P\,. \tag{2.20}$$

Returning to $F(t)$, we thus have

$$\boxed{F(t) = \int_0^t e^{iE_P\tau} U_0^\dagger(\tau) V(\tau) U_0(\tau) P \mathrm{d}\tau\,.} \tag{2.21}$$

Observe that $U_0^\dagger(\tau)V(\tau)U_0(\tau)P$ is independent of E_P. Therefore, we see that when E_P is large, the integrand of F is a rapidly oscillating function of τ due to the $e^{iE_P\tau}$. We can apply the Riemann-Lebesgue lemma to conclude that F vanishes as E_P goes to infinity. To be explicit, we perform an integration by parts to see that

$$\begin{aligned} F(t) &= \int_0^t e^{iE_P\tau} U_0^\dagger V U_0 P \mathrm{d}\tau \\ &= \frac{1}{iE_P}\Big[e^{iE_Pt}U_0^\dagger(t)V(t)U_0(t) - V(0) - \int_0^t e^{iE_P\tau}\frac{\mathrm{d}}{\mathrm{d}\tau}(U_0^\dagger V U_0)\mathrm{d}\tau\Big]P\,. \end{aligned} \tag{2.22}$$

The first two terms in the brackets do not grow with E_P and the third is bounded by t times the maximum magnitude of $\frac{\mathrm{d}}{\mathrm{d}\tau}(U_0^\dagger V U_0)$ which is independent of E_P. So as E_P goes to infinity, $F(t)$ goes to zero. Since both terms in the brackets in Eq. (2.15) contain F and are otherwise bounded independent of E_P, we have our E_P goes to infinity result:

Theorem 2.1. Suppose that the Hamiltonian of a system coupled to an environment is

$$H = H^{\mathrm{L}}_{\mathrm{comp}} + H_{\mathrm{env}} + \lambda V + E_P\tilde{Q},$$

where V acts 1-locally on the system, $H^{\mathrm{L}}_{\mathrm{comp}}$ is encoded in a quantum code that can detect single-qubit errors, and $\tilde{Q}$ is the operator defined in Eq. (2.5). Then, in the limit of an infinitely large energy penalty (positive or negative), the actual evolution in the codespace is as if there were no errors due to V; i.e. for any time T,

$$\boxed{\lim_{E_P\to\pm\infty} U(T)P = U_0(T)P\,,}$$

where U and U_0 are the actual and error-free evolution operators defined in Eq. (2.6) and P is the codespace projection operator of Eq. (2.2).

This result applies to the evolution of both the system and the environment, and is therefore stronger than what we need, which is only that the system evolution be protected. We view our infinite E_P result as the starting point for large, but finite, E_P investigations.

Although throughout this chapter we have focused only on the simplest case, where V acts 1-locally on the system and a 1-qubit quantum error-detection code is used, this simplification is not necessary. The theorem still holds as long as the error-detecting code can detect the errors that V causes, i.e. as long as $PVP = 0$, and therefore includes cases where V acts k-locally as long as the code can detect k-local errors. We show a proof of this in the appendix. The remainder of the chapter addresses the case where V acts 1-locally but in which we use a finite, rather than infinite, penalty E_P.

2.4.2 The finite E_P case

Frequency Analysis

We have just seen that for infinitely large E_P, the evolution in the codespace in the presence of noise is the same as the desired noise-free evolution. We now want to know how large E_P must be to assure us that $F(t)$ is very small, so that the actual evolution in the codespace is close to the desired one. It is helpful to consider the "natural frequencies" present in the expression for $F(t)$, as given by Eq. (2.21), which we informally analyse now.

If $f(t)$ is a (suitably nice) complex function, and $\tilde{f}(\omega)$ is its Fourier transform, then

$$\int_0^t \mathrm{d}\tau e^{iE_P\tau} f(\tau) = \int_0^t \mathrm{d}\tau e^{iE_P\tau} \int_{-\infty}^{\infty} \mathrm{d}\omega e^{-i\omega\tau} \tilde{f}(\omega) = \int_{-\infty}^{\infty} \mathrm{d}\omega \frac{e^{i(E_P-\omega)t}-1}{i(E_P-\omega)} \tilde{f}(\omega)\,.$$

Suppose that there exists an ω_c such that $\tilde{f}(\omega)$ is non-negligible only for $|\omega| < \omega_c$. Then

$$\int_0^t \mathrm{d}\tau e^{iE_P\tau} f(\tau) \approx \int_{|\omega|\leqslant\omega_c} \mathrm{d}\omega \frac{e^{i(E_P-\omega)t}-1}{i(E_P-\omega)} \tilde{f}(\omega)\,.$$

Now, if E_P is much larger than ω_c, we can replace $1/(E_P-\omega)$ by $1/E_P$ in this integral, and may therefore conclude that the integral is small (shrinking as $1/E_P$).

The question is, therefore, what are the natural frequencies of $U_0^\dagger(\tau)V(\tau)U_0(\tau)$? If they are not too large, then $F(t)$ should be small for reasonably large values of E_P. Consider first the time-independent case, in which H_0 and V are time-independent, so $U_0(\tau) = e^{-iH_0\tau}$. Certainly U_0 will have extremely large frequencies, namely $e^{-iE\tau}$ where E are eigenenergies of H_0; since H_0 includes the environment, E can scale with the size of the environment and be extremely large. However, the frequencies of $U_0^\dagger V U_0$, are expected to be much smaller. Inserting two complete sets of H_0 energy eigenstates, $|E\rangle$, we see that in the time-independent case,

$$U_0^\dagger(\tau)VU_0(\tau) = \sum_{E,E'} e^{i(E-E')\tau}|E\rangle\langle E|V|E'\rangle\langle E'|\,,$$

indicating that the frequencies are the energy *differences* induced by V. If V acts locally, we expect it would be unable to change the energy of the system/environment by a large amount – for example, it is unlikely that flipping just two spins in a spin chain will change the energy of the entire chain by more than a small amount. Therefore, we expect that $\langle E|V|E'\rangle$ is very small when $|E-E'|$ is large. If we make E_P larger than the largest $|E-E'|$ corresponding to any non-negligible $\langle E|V|E'\rangle$, we can conclude that F is small. To be more precise would require a specific model for the system, environment, and interaction. Still, we can make some progress on bounding F, even in the general time-dependent case.

Bounding F

We now bound the norm of $F(t) = \int_0^t e^{iE_P\tau}U_0^\dagger(\tau)V(\tau)U_0(\tau)P\mathrm{d}\tau$. Since the norm of V is expected to grow linearly in the size of the system, and therefore in n, one would naively expect the same of F. However, the fact that each logical qubit is independently encoded allows us to do slightly better. Recall from Eq. (2.4) that we can write V as a sum of terms,

V_i, where each V_i acts only on the ith logical system qubit (as well as the environment). Let

$$F_i(t) = \int_0^t e^{iE_P\tau} U_0^\dagger(\tau) V_i(\tau) U_0(\tau) P \mathrm{d}\tau$$

so

$$F = \sum_{i=1}^n F_i \,.$$

We now show that

$$\|F\| \leqslant \sqrt{n} \max_i \|F_i\| \,. \tag{2.23}$$

Proof. Observe that

$$\begin{aligned} F^\dagger F &= \sum_{i,j=1}^n F_i^\dagger F_j \\ &= \sum_{i,j=1}^n \int_0^t \int_0^t \mathrm{d}\tau_1 \mathrm{d}\tau_2 e^{iE_P(\tau_2-\tau_1)} [P U_0^\dagger(\tau_1) V_i(\tau_1) U_0(\tau_1)][U_0^\dagger(\tau_2) V_j(\tau_2) U_0(\tau_2) P] \end{aligned}$$

and consider the terms with $i \neq j$. In the leftmost P there is a P_j (i.e. $P = PP_j$) and it commutes with $U_0^\dagger(\tau_1), V_i(\tau_1)$, $U_0(\tau_1)$, and $U_0^\dagger(\tau_2)$. But $P_j V_j P = 0$ so these terms are 0. Consequently, the sum is only over $i = j$, i.e.

$$F^\dagger F = \sum_{i=1}^n F_i^\dagger F_i \,,$$

and the claim follows since $\|F\|^2 = \max_{|\psi\rangle} \langle\psi| F^\dagger F |\psi\rangle$. □

We now consider how to bound F_i (for any logical qubit i). In deriving Eq. (2.22), we assumed that $\frac{\mathrm{d}}{\mathrm{d}\tau}(U_0^\dagger V U_0)$ is finite; we now explicitly bound this term. By Eq. (2.6), we have

$$\frac{\mathrm{d}}{\mathrm{d}\tau}(U_0^\dagger V_i U_0) = -iU_0^\dagger [V_i, H_0] U_0 + U_0^\dagger \frac{\mathrm{d}V_i}{\mathrm{d}\tau} U_0 \,.$$

Using this, Eq. (2.22) becomes

$$F_i(t) = \frac{1}{iE_P}\Big[e^{iE_Pt} U_0^\dagger(t) V_i(t) U_0(t) - V_i(0) + i\int_0^t e^{iE_P\tau} U_0^\dagger [V_i, H_0] U_0 \mathrm{d}\tau - \int_0^t e^{iE_P\tau} U_0^\dagger \frac{\mathrm{d}V_i}{\mathrm{d}\tau} U_0 \mathrm{d}\tau\Big] P$$

and taking the norm, using that $\|A + B\| \leqslant \|A\| + \|B\|$, $\|AB\| \leqslant \|A\|\|B\|$, $\|U_0\| = 1$, and $\|P\| = 1$, we obtain

$$\boxed{\Big\|F_i(t)\Big\| \leqslant \frac{1}{|E_P|}\left(\Big\|V_i(t)\Big\| + \Big\|V_i(0)\Big\| + \max_\tau \Big\|[V_i(\tau), H_0(\tau)]\Big\| t + \max_\tau \left\|\frac{\mathrm{d}V_i}{\mathrm{d}\tau}\right\| t\right) \,.} \tag{2.24}$$

The norm $\left\|\frac{\mathrm{d}V_i}{\mathrm{d}\tau}\right\|$ will be bounded for reasonable V. For example, if the system control operations do not greatly change the environment surrounding each qubit, one expects that each V_i will likely stay fairly constant. Accordingly, we will ignore this term and the time dependence of V_i, in which case

$$\left\|F_i(t)\right\| \leqslant \frac{1}{|E_P|}\left(2\left\|V_i\right\| + \max_{\tau}\left\|[V_i, H_0(\tau)]\right\| t\right) . \tag{2.25}$$

The commutator $[V_i, H_0] = [V_i, H_{\text{comp}}^{\text{L}} + H_{\text{env}}]$ involves the environment Hamiltonian, which may be extremely large; however, we now show that by making some reasonable physical assumptions, $\left\|[V_i, H_0]\right\|$ is independent of the size of the system and environment.

First, we assume that $H_{\text{comp}}^{\text{L}}, H_{\text{env}}$, and V_i are local operators. They can therefore each be written as a sum of terms, each term involving only a few qubits. Second, we make the assumption that each qubit (of the system and environment) appears in at most a few of these local terms of $H_{\text{comp}}^{\text{L}}, H_{\text{env}}$, and V_i. For example, if a Hamiltonian is 2-local and *geometrically* local, say on a cubic lattice, so that each qubit only interacts with its immediate neighbours, then this restricts the number of terms in which any qubit appears, say to six for the cubic lattice. In terms of operator norms, these assumptions translate as follows.

For V_i we have

$$V_i = \sum_{s=1}^{\ell} \sum_{\mu=X,Y,Z} \sigma_\mu^s \otimes B_\mu^s$$

where the sum over s is only over the ℓ system qubits that comprise the ith logical qubit. B_μ^s is an environmental operator that couples to σ_μ^s and only consists of a few local terms (because system qubit s only appears in a few local terms of V_i), each acting on only a few environmental qubits (by locality). Therefore,

$$\left\|B_\mu^s\right\| = \mathcal{O}(1)$$

independent of the system and environment sizes. (Recall that the coupling, λ, has units of energy, so the B_μ^s are dimensionless.) We thus have that

$$\|V_i\| = \ell\mathcal{O}(1) . \tag{2.26}$$

Now, $H_0 = H_{\text{comp}}^{\text{L}} + H_{\text{env}}$ and both terms contribute to the commutator $[V_i, H_0]$. Let h_{sys}^s be the sum of all terms in $H_{\text{comp}}^{\text{L}}$ involving system qubit s, where s is a part of logical qubit i. Since there are only a few such terms, each of which acts on only a few system qubits, we can assert that

$$\left\|h_{\text{sys}}^s\right\| = \mathcal{E}\mathcal{O}(1)$$

where $\mathcal{E}$ is an energy scale parameter whose size is on the order of the size of the individual terms in $H_{\text{comp}}^{\text{L}}$. Similarly, let h_{env}^s be the sum of all terms in H_{env} that contain the environmental qubits that appear in B_μ^s for $\mu = X, Y, Z$. Since B_μ^s involves only a few environment qubits, which each appear in H_{env} in only a few, local terms, we have that

$$\|h_{\text{env}}^s\| = \mathcal{E}\mathcal{O}(1) .$$

Then, since $\|A+B\| \leqslant \|A\|+\|B\|$, $\|[A,B]\| \leqslant 2\|A\|\|B\|$, and $\|A\otimes B\| = \|A\|\|B\|$,

$$\begin{aligned}\Big\|[V_i,H_0]\Big\| &\leqslant \sum_{s=1}^{\ell}\sum_{\mu=X,Y,Z}\Big\|[\sigma_\mu^s\otimes B_\mu^s, H_{\text{comp}}^{\text{L}}]\Big\| + \Big\|[\sigma_\mu^s\otimes B_\mu^s, H_{\text{env}}]\Big\| \\ &= \sum_{s=1}^{\ell}\sum_{\mu=X,Y,Z}\Big\|[\sigma_\mu^s\otimes B_\mu^s, h_{\text{sys}}^s]\Big\| + \Big\|[\sigma_\mu^s\otimes B_\mu^s, h_{\text{env}}^s]\Big\| \\ &\leqslant 2\sum_{s=1}^{\ell}\sum_{\mu=X,Y,Z}\|\sigma_\mu^s\|\|B_\mu^s\|\left(\|h_{\text{sys}}^s\|+\|h_{\text{env}}^s\|\right)\,.\end{aligned}$$

Thus,

$$\Big\|[V_i,H_0]\Big\| = \ell\mathcal{E}\mathcal{O}(1) \tag{2.27}$$

independent of n and the size of the environment.

Applying the bounds of Eqs. (2.26) and (2.27) to Eq. (2.25) gives

$$\|F_i(t)\| \leqslant \frac{1}{|E_P|}\big[\ell\mathcal{O}(1)+\ell\mathcal{E}t\mathcal{O}(1)\big]$$

and using this in Eq. (2.23), we obtain

$$\|F(t)\| \leqslant \frac{\sqrt{n}}{|E_P|}\ell\big[\mathcal{O}(1)+\mathcal{E}t\mathcal{O}(1)\big]\,. \tag{2.28}$$

The term that grows with t represents a very weak bound for large t. We see from Eq. (2.21) that $F(t)$ is an integral over $[0,t]$ of an oscillating integrand and such integrals typically do not grow with t. For example, bounding

$$\int_0^t \sin(\omega\tau)\mathrm{d}\tau \leqslant t\,,$$

while true, is not very helpful for large t. However, this is the best that we have been able to do for the general problem at hand. In Sec. 2.5 we will look at the full t dependence of small systems using numerical simulation.

Fidelity calculation

Suppose the system/environment is initially in the pure state $|\psi\rangle$, and it evolves under U for time T. We begin in the codespace of the system, so $P\,|\psi\rangle = |\psi\rangle$. The fidelity squared, $\mathcal{F}^2$, between the desired final state, $U_0\,|\psi\rangle$, and the actual final state, $U\,|\psi\rangle$, is given by

$$\mathcal{F}^2 = \Big|\langle\psi|U_0^\dagger U|\psi\rangle\Big|^2 = \Big|\langle\psi|PU_0^\dagger UP|\psi\rangle\Big|^2\,.$$

To evaluate this, we left-multiply Eq. (2.15) by $PU_0^\dagger$, and use Eq. (2.12) to give

$$PU_0^\dagger UP = P - i\lambda PF - \lambda^2 PU_0^\dagger U\int_0^T U^\dagger V U_{0P}F\mathrm{d}t\,.$$

Because P commutes with U_0 and $PVP = 0$, we see from Eq. (2.21) that $PF = 0$. Therefore,

$$PU_0^\dagger UP = P - \lambda^2 PU_0^\dagger U \int_0^T U^\dagger V U_{0P} F \mathrm{d}t\,,$$

so

$$\langle\psi|U_0^\dagger U|\psi\rangle = 1 - \lambda^2 \langle\psi|\, PU_0^\dagger U \int_0^T U^\dagger V U_{0P} F \mathrm{d}\tau \,|\psi\rangle\ . \tag{2.29}$$

Making the physical assumptions discussed above, we can immediately derive a bound on the fidelity. From Eq. (2.26), $\|V\| = n\ell\mathcal{O}(1)$, and using Eq. (2.28), along with the norm properties of $\|AB\| \leqslant \|A\|\|B\|$, $\|U_0\| = \|U\| = 1$, and $\|P\| = 1$, we obtain

$$\left|\lambda^2 \langle\psi|\, PU_0^\dagger U\int_0^T U^\dagger V U_{0P} F \mathrm{d}\tau \,|\psi\rangle\right| \ \leqslant\ \lambda^2 \max_{0\leqslant t\leqslant T} \|V\|\, \|F(t)\|\, T \ \leqslant\ \frac{\lambda^2 n^{3/2}\ell^2}{|E_P|}\big[T\mathcal{O}(1){+}\mathcal{E}T^2\mathcal{O}(1)\,\big]\,.$$

Therefore, by the reverse triangle inequality, the fidelity is bounded by

$$\mathcal{F} = \left|\langle\psi|U_0^\dagger U|\psi\rangle\right| = \left|1{-}\lambda^2 \langle\psi|\, PU_0^\dagger U\int_0^T U^\dagger V U_{0P} F \mathrm{d}\tau \,|\psi\rangle\right| \geqslant 1{-}\frac{\lambda^2 n^{3/2}\ell^2}{|E_P|}\big[T\mathcal{O}(1){+}\mathcal{E}T^2\mathcal{O}(1)\,\big],$$

so we are guaranteed good fidelity if we have

$$E_P \gtrsim \lambda^2 n^{3/2}\ell^2\big[T\mathcal{O}(1) + \mathcal{E}T^2\mathcal{O}(1)\,\big]\,, \tag{2.30}$$

where by $\mathcal{O}(1)$ we mean the constants from Eqs. (2.26) and (2.27). For any efficient algorithm, $T \leqslant \mathrm{poly}(n)$, so since λ, ℓ, and $\mathcal{E}$ are independent of n, it suffices for E_P to grow polynomially in the number of logical qubits n.

We assume that λ, the system-environment coupling, can be engineered to be small compared to the magnitudes of the individual terms in H_0. Accordingly, let us consider $\langle\psi|U_0^\dagger U|\psi\rangle$ to order λ^2. Working to this order, we can set $U = U_{0P}$ (as would occur if λ were zero) on the right hand side of Eq. (2.29):

$$\langle\psi|U_0^\dagger U|\psi\rangle = 1 - \lambda^2 \langle\psi|\, PU_0^\dagger U_{0P} \int_0^T U_{0P}^\dagger V U_{0P} F \mathrm{d}t \,|\psi\rangle\ + \mathcal{O}\big(\lambda^3\big)\,.$$

Recall from Eqs. (2.7) and (2.12) that $U_{0P} = U_0 U_P$ and $PU_P = P$, so that $PU_0^\dagger U_{0P} = P$. Recalling the notation of Eq. (2.8) from the interaction picture, i.e. of $V_I \equiv U_{0P}^\dagger V U_{0P}$, and the definition of F in Eq. (2.14), we therefore have

$$\begin{aligned}\langle\psi|U_0^\dagger U|\psi\rangle &= 1 - \lambda^2 \langle\psi|\, P\int_0^T V_I(t)F(t)\mathrm{d}t \,|\psi\rangle\ + \mathcal{O}\big(\lambda^3\big)\\ &= 1 - \lambda^2 \langle\psi|\, P\int_0^T V_I(t)\int_0^t V_I(\tau)P\ \mathrm{d}\tau\mathrm{d}t \,|\psi\rangle\ + \mathcal{O}\big(\lambda^3\big)\,.\end{aligned}$$

With perfect error suppression, $\mathcal{F}^2 \to 1$, so $1 - \mathcal{F}^2$ is a measure of error suppression failure. We calculate

$$1 - \mathcal{F}^2 = 1 - \left|\langle\psi|U_0^\dagger U|\psi\rangle\right|^2 = \lambda^2 \langle\psi|\, P\int_0^T \mathrm{d}t\int_0^t \mathrm{d}\tau\ V_I(t)V_I(\tau)P\,|\psi\rangle + h.c. + \mathcal{O}\big(\lambda^3\big)$$

where *h.c.* denotes the Hermitian conjugate. But this conjugate involves

$$\left(\int_0^T \mathrm{d}t \int_0^t \mathrm{d}\tau\ V_I(t)V_I(\tau)\right)^\dagger = \int_0^T \mathrm{d}t \int_0^t \mathrm{d}\tau\ V_I^\dagger(\tau)V_I^\dagger(t) = \int_0^T \mathrm{d}\tau \int_0^\tau \mathrm{d}t\ V_I(t)V_I(\tau)$$

where in the last step we used the fact that V_I is Hermitian and relabelled $t \leftrightarrow \tau$, showing that this term is identical to the term of which it is the conjugate, except for the integration region. The original integrates over a region with $\tau < t$, while the conjugate integrates the same integrand over a region with $t < \tau$, so their sum integrates over all $0 \leqslant \tau, t \leqslant T$. Thus,

$$1 - \mathcal{F}^2 = \lambda^2 \langle\psi| P \int_0^T \mathrm{d}t V_I(t) \int_0^T \mathrm{d}\tau V_I(\tau) P |\psi\rangle + \mathcal{O}\big(\lambda^3\big)$$

i.e.

$$1 - \mathcal{F}^2 = \lambda^2 \langle\psi| F^\dagger F |\psi\rangle + \mathcal{O}\big(\lambda^3\big)$$

so

$$\boxed{1 - \mathcal{F}^2 \leqslant \lambda^2 \|F\|^2 + \mathcal{O}\big(\lambda^3\big)\ .} \tag{2.31}$$

We see that a small $\|F\|$ corresponds to good error suppression.

We can combine this expression with Eq. (2.28) to obtain, at time T,

$$1 - \mathcal{F}^2(T) \leqslant \frac{\lambda^2 n}{E_P^2}\ell^2 \big[\mathcal{O}(1) + \mathcal{O}(1)\,\mathcal{E}T\big]^2 + \mathcal{O}\big(\lambda^3\big)\ . \tag{2.32}$$

It is possible to write an expression for the λ^3 contribution. We find that the leading term in $1/E_P$ in the λ^3 contribution goes like $\lambda^3 T/E_P^2$. Again, we do not believe that this gives a useful bound for large T, but it may be useful in the small T regime.

2.5 Numerical simulation for one logical qubit

In this section, we discuss the results of a numerical simulation of 1 logical qubit, encoded as 4 physical qubits using the Jordan-Farhi-Shor [1] code, coupled to an 8-qubit environment according to

$$H = H^{\mathrm{L}}_{\mathrm{comp}} + H_{\mathrm{env}} + \lambda V + E_P \tilde{Q}\,.$$

Since we track the evolution over long times, we find it too computationally expensive to work with more than 12 qubits total; therefore, we analyse only one logical qubit coupled to a modest size environment.

We choose the environment and the couplings as follows. The environment qubits are arranged on a randomly chosen 3-regular graph and have 2-local interactions between nearest neighbours. Each physical system qubit couples to a single, unique, randomly-selected environment qubit. For simplicity, the environment and coupling Hamiltonians, H_{env} and V, are time-independent.

The environment Hamiltonian takes the form

$$H_{\mathrm{env}} = \sum_{a=1}^{8} \alpha_a (\hat{n}_a \cdot \vec{\sigma}^a) + \sum_{\langle b,c\rangle} \alpha_{bc} (\hat{m}_b \cdot \vec{\sigma}^b) \otimes (\hat{\ell}_c \cdot \vec{\sigma}^c)$$

where each $\hat{n}_a, \hat{m}_b$, and $\hat{\ell}_c$ is a randomly chosen unit vector, $\vec{\sigma}^a = (\sigma_X^a, \sigma_Y^a, \sigma_Z^a)$ are the Pauli operators acting on environment qubit a, each α_a and α_{bc} is a coefficient chosen at random in the range of $[0.9, 1.1]$, and $\sum_{\langle b,c\rangle}$ denotes a sum over neighbouring environment qubits on the 3-regular graph.

In this small simulation, with 1 logical qubit, the system size is 4. The system-environment coupling has the form of Eq. (2.3), with the environmental operators chosen to be simple single-qubit terms, and is given by

$$V = \sum_{s=1}^{4} \beta_s (\hat{n}_s \cdot \vec{\sigma}^s) + \sum_{s=1}^{4} \gamma_s (\hat{m}_s \cdot \vec{\sigma}^s) \otimes (\hat{\ell}_s \cdot \vec{\sigma}^s_{\text{env}})$$

where each $\hat{n}_s$, $\hat{m}_s$, and $\hat{\ell}_s$ is a randomly chosen unit vector, $\vec{\sigma}^s$ are the Pauli operators acting on system qubit s, and $\vec{\sigma}^s_{\text{env}}$ are the Pauli operators acting on the environment qubit that is coupled to system qubit s. Note that we have included single-qubit error terms, $\hat{n}_s \cdot \vec{\sigma}^s$, that are not coupled to any environment qubits but may arise from pure system errors. The coefficients β_s and γ_s are each chosen at random in the range of $[0.9, 1.1]$. By design, V acts 1-locally on the system.

The initial state is taken to be a pure product state of the system and environment,

$$|\psi\rangle = |\psi^s\rangle \otimes |\psi^e\rangle ,$$

where the initial environment state $|\psi^e\rangle$ is a random 8-qubit state. We will study different choices for the initial system state $|\psi^s\rangle$ and the computational Hamiltonian $H^{\text{L}}_{\text{comp}}$. In order to compare the actual and desired dynamics, we evolve with U and U_0 defined in Eq. (2.6) to obtain

$$\begin{aligned} |\phi(t)\rangle &= U(t)\,|\psi\rangle, \qquad t \in [0, T] \\ |\phi_0(t)\rangle &= U_0(t)\,|\psi\rangle, \qquad t \in [0, T] . \end{aligned}$$

Note that because the system and environment are not coupled by H_0, we can write

$$|\phi_0(t)\rangle = |\phi_0^s(t)\rangle \otimes |\phi_0^e(t)\rangle$$

so that the state of the system at time t is $|\phi_0^s(t)\rangle$, independent of the environment. In the coupled case, on the other hand, the state of the system at time $t > 0$ is described by a density matrix,

$$\rho(t) = \text{Tr}_{\text{env}}\, |\phi(t)\rangle\langle\phi(t)| ,$$

where the environment qubits have been traced out.

At any time t, we compare the actual versus coupling-free evolutions using the following measures:

- The squared fidelity of the total evolution,

  $$\mathcal{F}^2(t) = |\,\langle\phi_0(t)|\phi(t)\rangle\,|^2.$$

  As a result of our theorem, this measure goes to 1 as $E_P \to \pm\infty$. This fidelity also contains the fidelity of the environment's evolution, and accordingly is a stronger measure than what we need to track how well the computation is protected.

- The squared fidelity of the *system* evolution,
$$\mathcal{F}_s^2(t) = \langle \phi_0^s(t) | \rho(t) | \phi_0^s(t) \rangle.$$
This measure determines if the quantum computation in the presence of the coupling to the environment is following the desired evolution. The irrelevant bath degrees of freedom are traced out.

We first perform numerical simulations for the time-independent computational Hamiltonian $H_{\text{comp}}^{\text{L}} = X_L$. Figure 2-1 shows the results of a typical simulation with $\lambda = 0.1$ and for a variety of E_P values, for both fidelity measures defined above. The initial system state in this case is a random superposition of $|0_L\rangle$ and $|1_L\rangle$, which can be viewed as a random superposition of the codespace eigenstates of $H_{\text{comp}}^{\text{L}} = X_L$, i.e. of

$$|\pm_L\rangle = \frac{1}{\sqrt{2}}\Big(|0_L\rangle \pm |1_L\rangle\Big).$$

We make the following observations:

- In the absence of an energy penalty, i.e. when $E_P = 0$, the fidelities rapidly fall. We see that $\mathcal{F}_s^2$ falls to a value of about $1/16$, which is the expected fidelity between two random 4-qubit system states. In other words, the state of the system is outside the codespace and is uncorrelated with the state resulting from the desired evolution.

- For large E_P, near-perfect fidelity is maintained for a long time, both for the system ($\mathcal{F}_s$) and the system-environment ($\mathcal{F}$). However, the fidelity eventually falls, and does so fairly abruptly (on a log-scale). This kind of behaviour would not be seen in a low-order power series expansion in time and is certainly not seen in expressions like Eq. (2.24) that have a linear term in t. Note that the larger the value of E_P, the longer near-perfect fidelity is maintained.

- For sufficiently large E_P, the general behaviour is for the system fidelity $\mathcal{F}_s$ to approach an asymptotic value for large t, about which it has small fluctuations. We have data for times greater than what we plot here that supports this observation, but of course we cannot draw firm conclusions about what happens as $t \to \infty$. Still, we can say that the system fidelity stays fairly level away from zero at time scales much larger than any natural time scale involved in the simulation.

- The total fidelity, $\mathcal{F}$, always falls to very close to 0 for very large t, indicating that the environment state is not as well protected as the system state is. This is unsurprising, as there is no preferred codespace for the environment.

- In Fig. 2-2, we see qualitatively the same behaviour for the same randomly chosen H_{env}, V, and $|\psi\rangle$, but with $\lambda = 0.01$ (rather than $\lambda = 0.1$). Note that for each E_P, the smaller λ value allows for good protection for longer times than the larger λ value allows.

It is interesting to compare the bounds of Eq. (2.31) and Eq. (2.25) with our numerical observations. For the parameters used to generate Figs. 2-1 and 2-2, we have $\|V\| \approx 7$, $\|H_0\| \approx 12$, and $\big\|[V, H_0]\big\| \approx 17$ (significantly less than $\|V\| \cdot \|H_0\|$, in accordance with our previous discussion on locality). Equation (2.31) suggests that good fidelity squared, say of

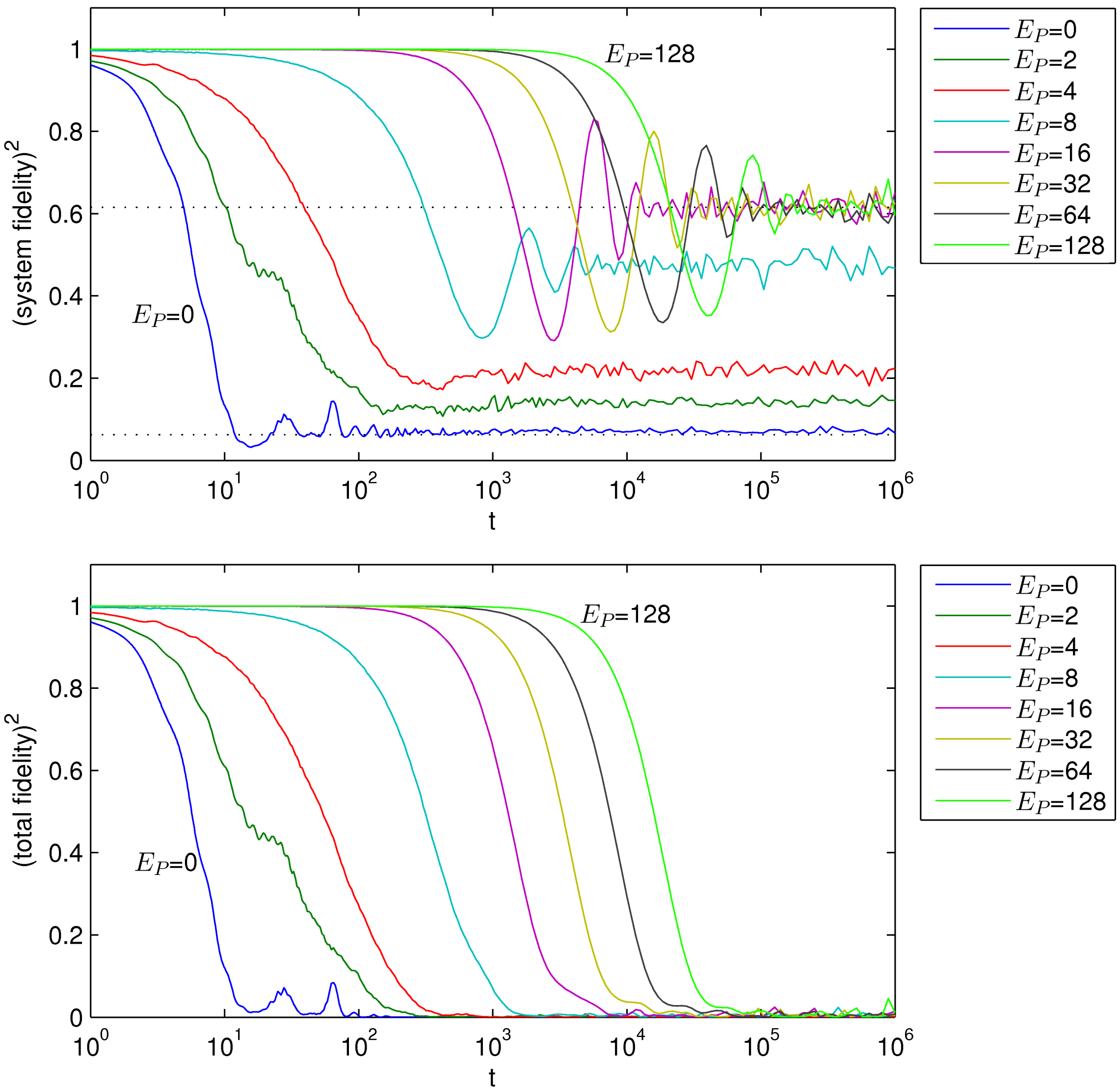


Figure 2-1: (Top) squared system fidelity, $\mathcal{F}_s^2$, and (bottom) squared total fidelity, $\mathcal{F}^2$, as functions of time t on a log-scale for $\lambda = 0.1$ and initial system state $|\psi^s\rangle = \alpha|+_L\rangle + \beta|-_L\rangle$ with a random choice of α and β obeying $|\alpha|^2 + |\beta|^2 = 1$. Results are shown for increasing energy penalty strengths, E_P. All data are for $H^{\mathrm{L}}_{\mathrm{comp}} = X_L$ on a 4-qubit system, and for a particular random instance of H_{env}, V, $|\psi^s\rangle$, and $|\psi^e\rangle$ with 8 environment qubits. The dashed line in the top panel is at $|\alpha|^4 + |\beta|^4$, which is 0.615 for this particular choice of $|\psi^s\rangle$; its significance will be explained later. The dashed line at $1/16$ is the expected long-time system fidelity in the absence of protection.

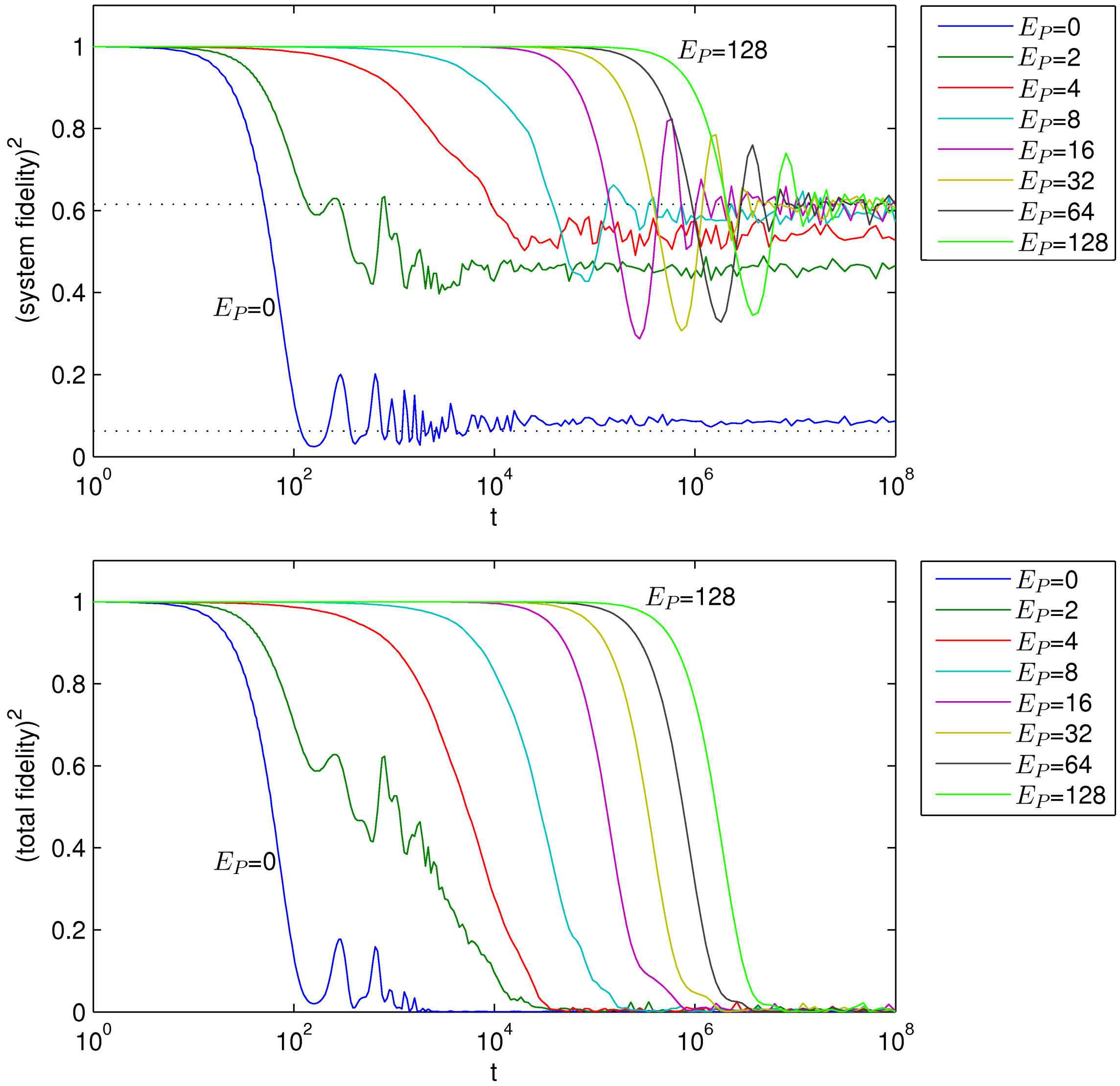


Figure 2-2: (Top) squared system fidelity, $\mathcal{F}_s^2$, and (bottom) squared total fidelity, $\mathcal{F}^2$, as functions of time t on a log-scale for $\lambda = 0.01$. All other values are identical to those of Fig. 2-1, but the time scale has been increased because there is better protection for the smaller value of λ.

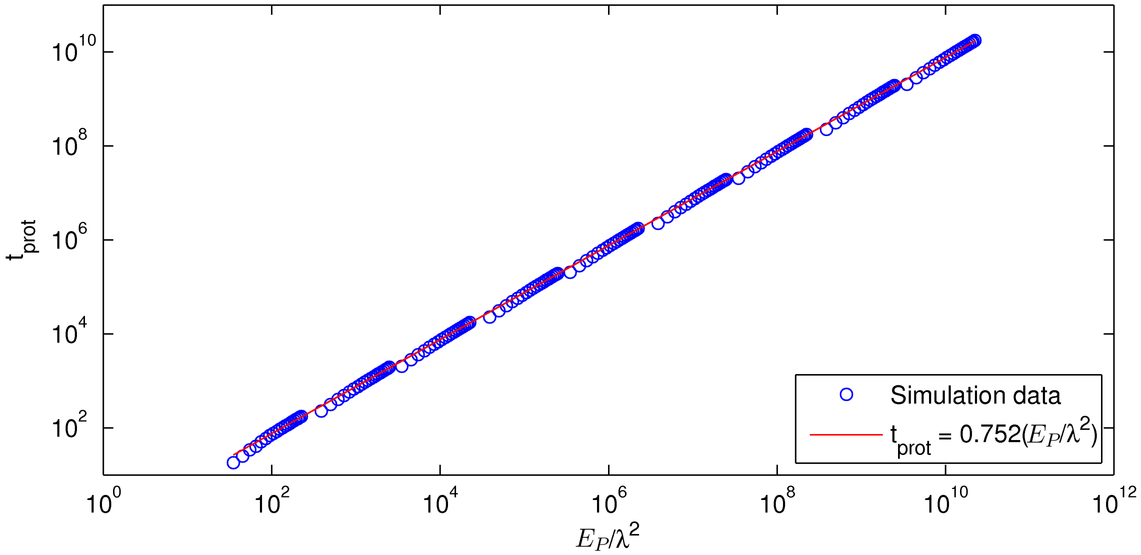


Figure 2-3: The protection time, t_{prot}, defined as the time at which the squared system fidelity $\mathcal{F}_s^2$ drops to 0.9, versus E_P/λ^2 for a range of E_P and λ values, specifically, $E_P \in \{35, 45, \ldots, 225\}$ and $\lambda \in \{10^{-4}, 3 \cdot 10^{-4}, 10^{-3}, 3 \cdot 10^{-3}, 10^{-2}, 3 \cdot 10^{-2}, 10^{-1}, 3 \cdot 10^{-1}, 1\}$ (each of the 9 "clusters" of data in the figure corresponding to a different λ value.) All Hamiltonian and initial state values (other than E_P and λ) are kept identical to those of Fig. 2-1. The line shows that a linear relationship between t_{prot} and E_P/λ^2 fits the data well.

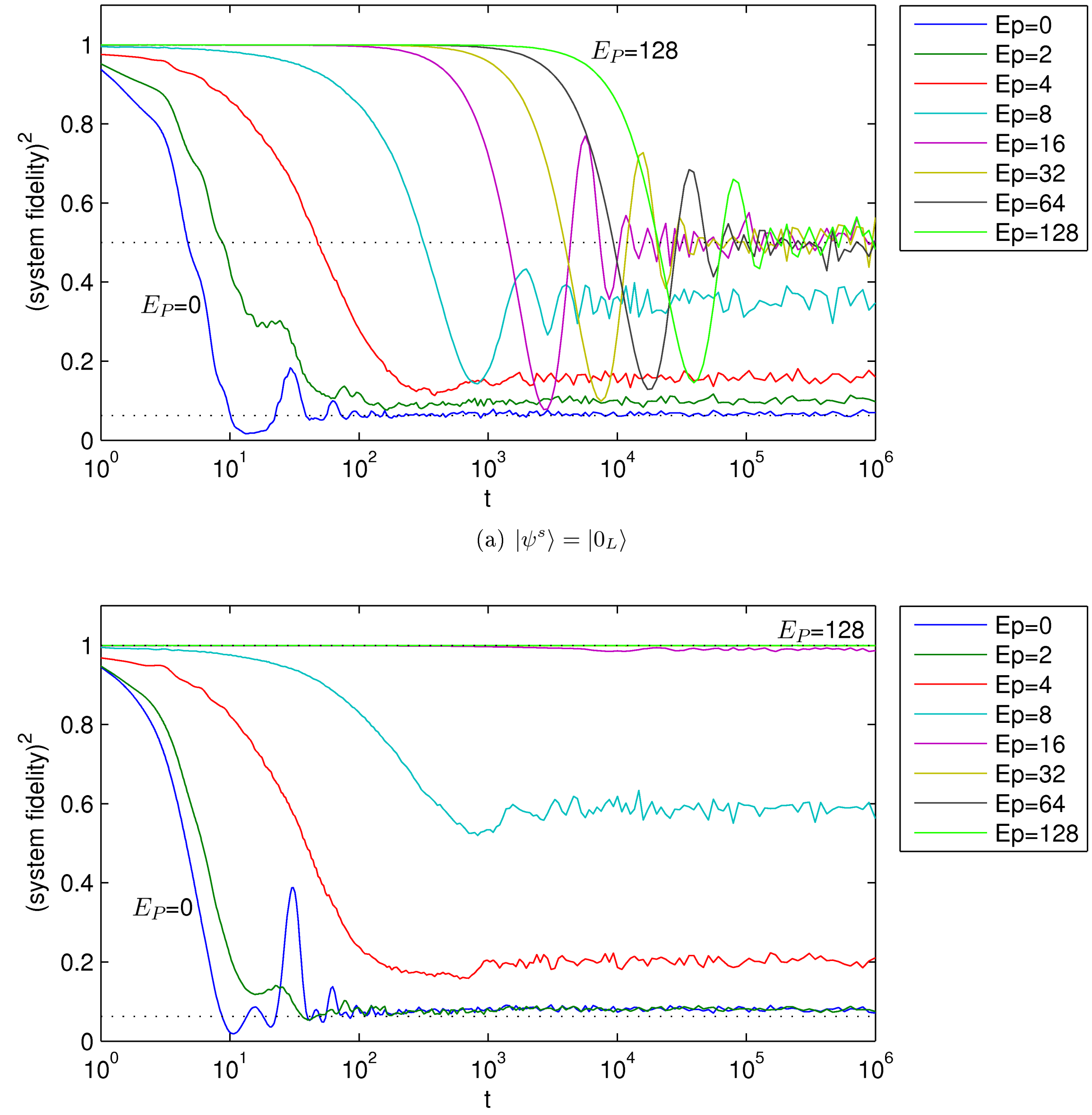
0.9, can be achieved if $\lambda^2\|F\|^2 \lesssim 0.1$, so for $\lambda = 0.1$ we expect that we need $\|F\| \lesssim 3$. The bound in Eq. (2.25) indicates that for $E_P = 32$, $\|F\| \lesssim 3$ for $T \lesssim 5$, so that these two bounds together suggest that good fidelity can be maintained for time $T \lesssim 5$ if $E_P = 32$. However, in Fig. 2-1 we see that, in this case, we can maintain good $\mathcal{F}^2$ up to $T = 1000$. Similarly, for $\lambda = 0.01$ we expect that we need $\|F\| \lesssim 30$ (from Eq. (2.31)), which for $E_P = 32$ can be guaranteed for $T \lesssim 60$ (by Eq. (2.25)); however, Fig. 2-2 indicates that in this case we can maintain good $\mathcal{F}^2$ up to $T = 100,000$. We thus see that Eq. (2.25) is not really useful for large T, as our numerical results show good fidelity for far longer than our bounds can guarantee.

To address the question of how long good fidelity can be maintained, we note that for successful quantum computation it suffices to have high system fidelity $\mathcal{F}_s$; high total fidelity $\mathcal{F}$ is not required. Accordingly, we define the *protection time*, t_{prot}, to be the time at which the squared system fidelity $\mathcal{F}_s^2$ first drops to 0.9. In Fig. 2-3 we plot t_{prot} for a variety of values of $E_P \in [35, 225]$ and $\lambda \in [10^{-4}, 1]$ (with all other Hamiltonian and initial state values held fixed). Observe that, to a very good approximation, the data fit the relation

$$t_{\text{prot}} \propto E_P/\lambda^2$$

for larger values of E_P. We will later show a simple model that is consistent with this behaviour.

We next address the question of what the system fidelity falls to at late times for large E_P. For the Hamiltonian $H^{\text{L}}_{\text{comp}} = X_L$, given $|\psi^s\rangle$ we can actually predict the long-term system fidelity. To help uncover this relationship, we plot in Fig. 2-4(a) the system fidelity



(a) $|\psi^s\rangle = |0_L\rangle$

(b) $|\psi^s\rangle = |+_L\rangle$

Figure 2-4: Squared system fidelity $\mathcal{F}_s^2$ as a function of time t on a log-scale for initial states (a) $|\psi^s\rangle = |0_L\rangle$ and (b) $|\psi^s\rangle = |+_L\rangle$, with $H^{\mathrm{L}}_{\mathrm{comp}} = X_L$ and $\lambda = 0.1$. All Hamiltonian and initial environment state values are identical to those of Fig. 2-1. The dashed lines at $1/2$ (in the top figure) and $1/16$ (in both figures) serve as guides for the eye. Note that in the bottom figure, for $E_P \geqslant 16$, $\mathcal{F}_s^2$ remains close to 1 for the duration of the simulation.

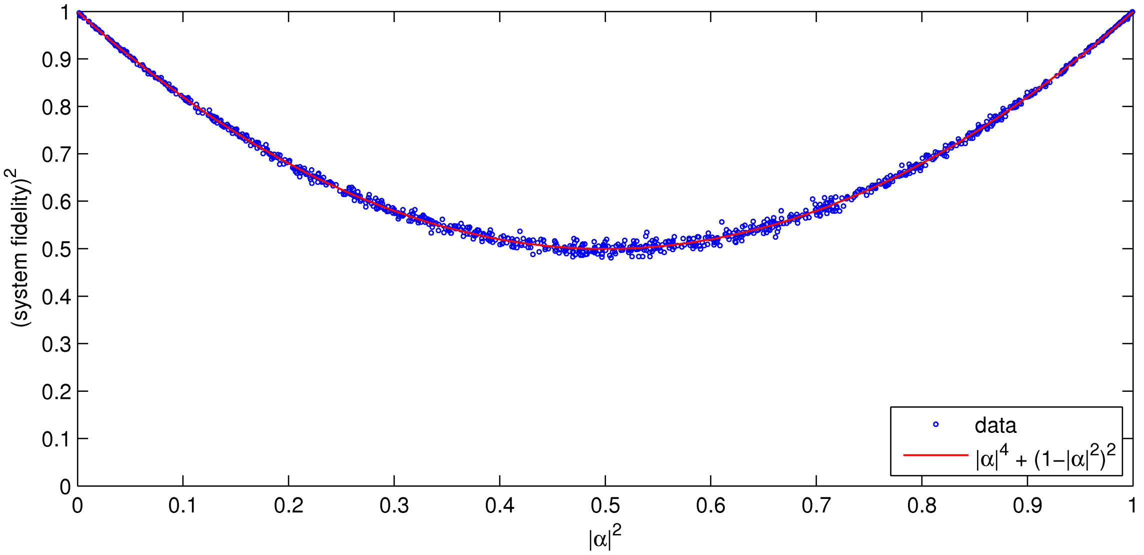


Figure 2-5: The long-term squared system fidelity, $\mathcal{F}_s^2$, as a function of $|\alpha|^2$, where the initial system state is $|\psi^s\rangle = \alpha|+_L\rangle + \beta|-_L\rangle$ and $|\alpha|^2 + |\beta|^2 = 1$. The curve $y = |\alpha|^4 + (1 - |\alpha|^2)^2$ is also shown, and the good fit is apparent. Each data point represents a random choice for α and β, as well as V, H_{env}, and the initial environment state $|\psi^e\rangle$. The computational Hamiltonian is $H^{\text{L}}_{\text{comp}} = X_L$ and $E_P = 128$. Each data point is the average $\mathcal{F}_s^2(T)$ over the times $T = \{1, 2, \ldots, 10\} \times 10^8$ to account for fluctuations in time of $\mathcal{F}_s$ about the long-term system fidelity.

as a function of time, with $|\psi^s\rangle$ taken to be $|0_L\rangle$. Note that the long-term system fidelity is very near $\frac{1}{2}$ for $E_P \geqslant 16$. In Fig. 2-4(b) we show the same thing but with $|\psi^s\rangle = |+_L\rangle$, an eigenstate of X_L, and see that the long-term system fidelity is very near 1 for $E_P \geqslant 16$. More generally, we observe that if we write

$$|\psi^s\rangle = \alpha|+_L\rangle + \beta|-_L\rangle\,, \tag{2.33}$$

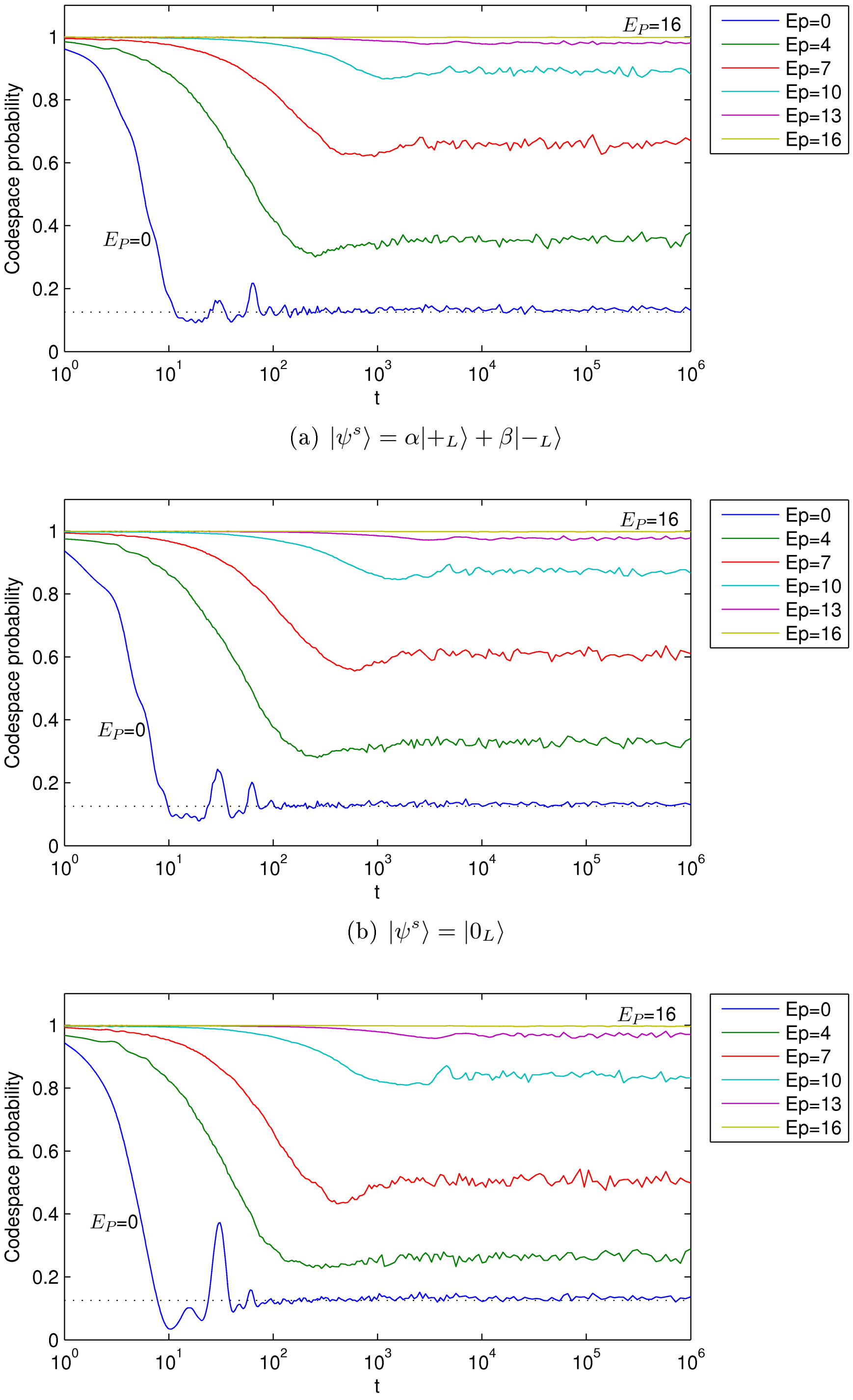
with $|\pm_L\rangle$ being the codespace eigenstates of X_L and $|\alpha|^2 + |\beta|^2 = 1$, then the long-term system fidelity is well approximated by $|\alpha|^4 + |\beta|^4$. In Fig. 2-5 we show the long-term system fidelity versus $|\alpha|^4 + (1 - |\alpha|^2)^2$ for a set of randomly chosen $|\psi^s\rangle$ and the good fit is apparent.

We show in Fig. 2-6, for the three choices of $|\psi^s\rangle$ displayed in Figs. 2-1, 2-4(a), and 2-4(b), the probability to remain in the codespace, $\langle\phi(t)|P|\phi(t)\rangle$. We see that it is close to 1 for all displayed times for $E_P \geqslant 16$, indicating that any loss of system fidelity is occurring because of errors inside the codespace. With $H^{\text{L}}_{\text{comp}} = X_L$, the desired evolution, starting with the state in Eq. (2.33), is

$$|\phi_0^s(t)\rangle = \alpha e^{-it}|+_L\rangle + \beta e^{it}|-_L\rangle$$

since the codespace eigenvalues of X_L are ±1. Imagine that the only effect of the coupling to the environment is to induce dephasing in the $H^{\text{L}}_{\text{comp}}$ energy eigenbasis. Then the density matrix of the system will approach

$$\rho(t) = |\alpha|^2 \big|+_L\big\rangle\big\langle+_L\big| + |\beta|^2 \big|-_L\big\rangle\big\langle-_L\big|$$



(a) $|\psi^s\rangle = \alpha|+_L\rangle + \beta|-_L\rangle$

(b) $|\psi^s\rangle = |0_L\rangle$

(c) $|\psi^s\rangle = |+_L\rangle$

Figure 2-6: The probability $\langle\phi(t)|P|\phi(t)\rangle$ of the system being found in the codespace for the three different initial states used in Figs. 2-1, 2-4(a), and 2-4(b). The dashed line at 1/8 represents the expected probability for a maximally mixed 4-qubit state to be found in the 1-qubit codespace. In all three cases, for $E_P \geqslant 16$ the codespace probability is very near 1 for all displayed times.

and the squared system fidelity, $\langle \phi_0^s(t)|\rho(t)|\phi_0^s(t)\rangle$, is $|\alpha|^4 + |\beta|^4$. That the data in Fig. 2-5 match this is good evidence that the effect of the coupling to the environment is to cause dephasing in the energy eigenbasis of $H^{\mathrm{L}}_{\mathrm{comp}}$.

In our simulation we see that, for sufficiently large energy penalties, the system remains in the codespace and decoheres inside the codespace via dephasing of the energy eigenstates. We now present a simple phenomenological model that allows us to estimate t_{prot}, the time at which the effects of decoherence become appreciable. The model has three states. The first two states are the codespace eigenstates, $|+\rangle$ and $|-\rangle$, of the logically-encoded two-level computational Hamiltonian with energies ω and $-\omega$. The third state is a penalty state, representing all the states orthogonal to the codespace, and accordingly has energy $E_P \gg \omega$. The third state is coupled to the first two as a result of interactions with the environment, so that the effective Hamiltonian is

$$H_{\mathrm{eff}} = \begin{bmatrix} \omega & 0 & \lambda_+ \\ 0 & -\omega & \lambda_- \\ \lambda_+ & \lambda_- & E_P \end{bmatrix} .$$

Here, λ_+ and λ_- are the effective couplings of the first two states to the penalty state, and we assume that they are small compared to ω. We imagine that λ_+ and λ_- are proportional to some constant λ that represents the overall scale of the effective couplings. Expanding to lowest order in λ_+, λ_-, and $1/E_P$, we find that

$$\left\| \langle -|e^{-iH_{\mathrm{eff}}t}|+\rangle \right\|^2 \lesssim \left(\frac{\lambda_+\lambda_-}{\omega E_P} \right)^2 ,$$

so in this model the transition probability between states $|+\rangle$ and $|-\rangle$ is negligible for all time.

Treating the coupling as a perturbation, the effect of the coupling of the system states to the penalty state is to shift their energies. The perturbed energies are calculated to be

$$E_\pm = \pm\omega - \frac{\lambda_\pm^2}{E_P}$$

to lowest order in λ_+, λ_-, and $1/E_P$. Thus in this little model, at time t, the interaction-induced phase difference between $|+\rangle$ and $|-\rangle$ is

$$\big(E_+ - E_- - 2\omega\big)t = -\frac{\lambda_+^2 - \lambda_-^2}{E_P}\, t$$

so that the characteristic dephasing time is proportional to E_P/λ^2 . Generalizing from the toy model to an encoded two-level logical system with a coupling to the environment of size λ and energy penalty term of size E_P, we guess that for large E_P and small λ,

$$t_{\mathrm{prot}} \propto \frac{E_P}{\lambda^2} ,$$

in agreement with the behaviour seen in Fig. 2-3.

Returning to the simulation results, we have seen that a sufficiently large energy penalty keeps the system in the codespace, even for large t. We also presented evidence that decoherence inside the codespace occurs via dephasing in the energy basis. In particular, with a time-independent $H^{\mathrm{L}}_{\mathrm{comp}} = X_L$, starting in an energy eigenstate, say $|+_L\rangle$, we find that for sufficiently large E_P the system remains approximately in that eigenstate for the duration of the simulation. In adiabatic quantum computation [18], the state of the system is initially the ground state of a time-dependent computational Hamiltonian and, provided that the computational Hamiltonian is changed slowly enough, the evolving state is expected to remain near the instantaneous ground state. One might therefore expect good fidelity in the adiabatic computation case as well.

We now show the results of simulations for the one-logical-qubit adiabatic computation

$$H^{\mathrm{L}}_{\mathrm{comp}}(t) = \left(1 - \frac{t}{T}\right) X_L + \frac{t}{T} Z_L \,,$$

where the initial system state $|\psi^s\rangle = \frac{1}{\sqrt{2}}(|0_L\rangle - |1_L\rangle)$ is the ground state of $H^{\mathrm{L}}_{\mathrm{comp}}(0)$. The results are shown in Fig. 2-7 for $T = 10,000$. Observe that for $E_P \geqslant 16$, the system fidelity remains very high for the duration of the computation.

We emphasize that our numerical results are for a small system (1 logical qubit made of 4 physical qubits) coupled to a small environment (of 8 qubits). We do not know if the observations we have made for one logical qubit will hold in more complicated systems with many logical qubits. In particular, we would like to know if with a large number of qubits, modest energy penalties can keep the system in the codespace and, if inside the codespace, whether the decoherence is limited to dephasing in the energy basis. If so, this would be of great help in protecting adiabatic quantum computation. Furthermore, we are concerned that in our simulations, the size of the environment may be too small, especially given the large values of E_P that we are exploring. It would be disappointing if our encouraging small-system simulation results are artifacts of having too small an environment or do not reflect what actually happens in large systems. Nevertheless, these numerical results, in conjunction with the proof that the energy penalty method works in the infinite E_P limit, suggest that the energy penalty method may be a useful approach towards the development of fault-tolerant Hamiltonian-based quantum computing.

2.6 Outlook

To use the energy penalty method in an actual device, some practical hurdles remain to be overcome. The logical operators used by the codes discussed in this chapter need to be at least 4-local (in order to detect arbitrary 1-local errors), whereas physically implementable Hamiltonians are generally constrained to be 2-local. The usual procedure to overcome such locality constraints is to use so-called perturbative gadgets (as introduced in [19]), which allow one to construct a 2-local Hamiltonian whose low-energy subspace approximates a given desired Hamiltonian. Such techniques can perhaps be used here, too, to achieve error suppression using only 2-local operations and energy penalties. Another technique that might work for certain situations is to use codes that do not correct arbitrary errors but have a smaller locality, similar to what was done in Ref. [9] in the context of quantum annealing. This would be useful in situations in which it is known that only certain types of errors are problematic. In addition to allowing for only 2-local Hamiltonians, such codes may admit fewer physical qubits per logical qubit (i.e. a smaller value of ℓ), reducing the total error on the system and enabling numerical simulations for a larger number of logical qubits than we have been able to do here.

Another potential hurdle is the scaling of the energy penalty. To maintain a given desired fidelity, it is possible that the magnitude of the required energy penalty E_P depends on the

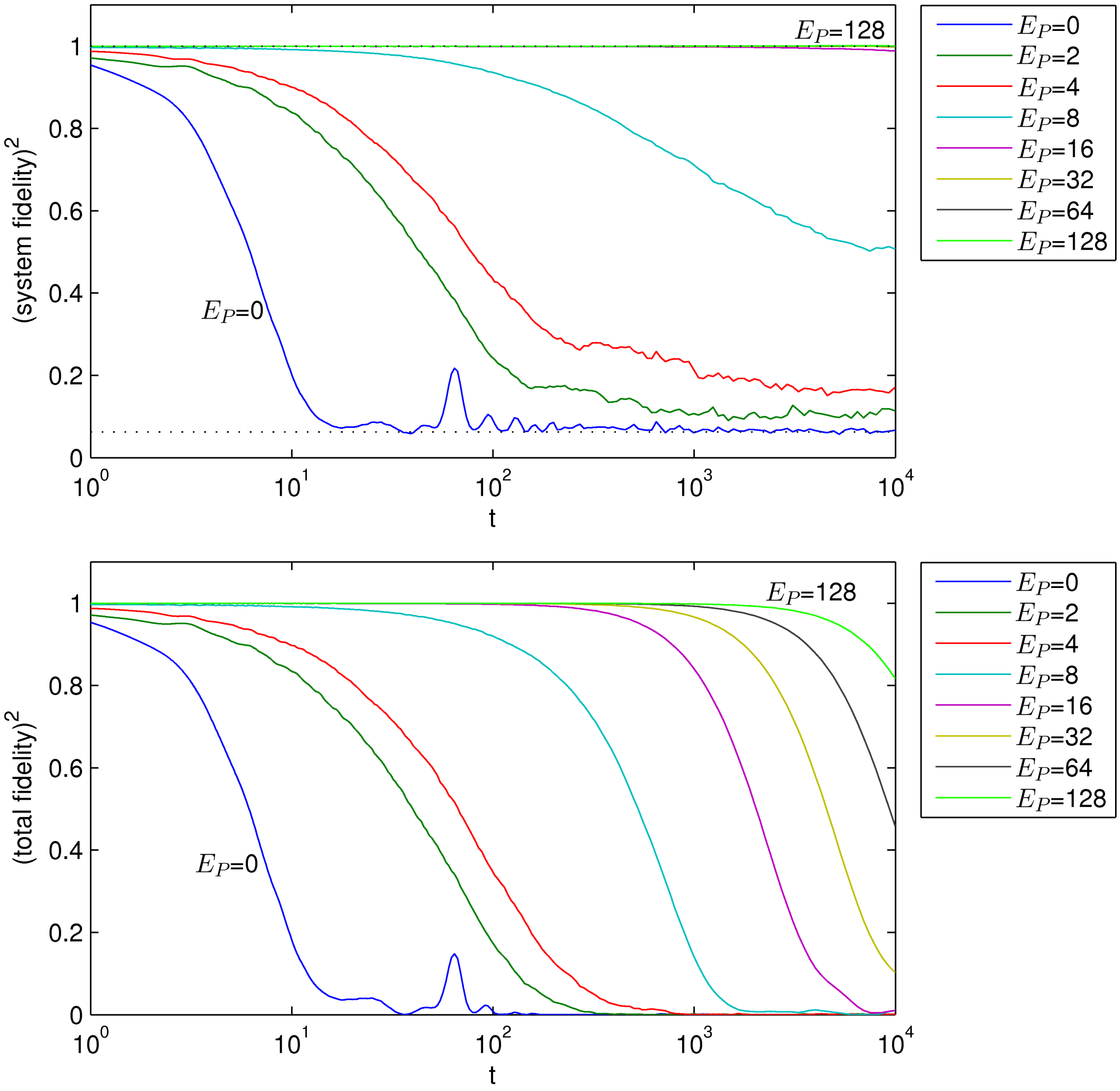


Figure 2-7: For the adiabatic computation, $H^{\mathrm{L}}_{\mathrm{comp}}(t) = (1 - \frac{t}{T})X_L + \frac{t}{T}Z_L$ with $T = 10000$, the (top) squared system fidelity, $\mathcal{F}_s^2$, and (bottom) squared total fidelity, $\mathcal{F}^2$, as functions of time t for $\lambda = 0.1$. All data are for a particular random instance of H_{env}, V, and $|\psi^e\rangle$ with 8 environment qubits, with the system initially in the ground state of $H^{\mathrm{L}}_{\mathrm{comp}}(0)$. Note that for $E_P \geqslant 16$, we have nearly perfect system fidelity throughout the evolution.

size of the system. Fortunately, Eq. (2.30) shows that under reasonable physical assumptions, such a scaling is at most polynomial in the number, n, of logical qubits. We hope that this inequality can be tightened further. For scalable implementation of Hamiltonian-based quantum computing with error suppression, it is likely that changing the energy penalty (even polynomially) to accommodate increases in the logical system size may be difficult to do. For a practical fault-tolerant theorem, it would be desirable for the error-suppression to be ultimately achieved through the addition of extra qubits, as in the circuit-model case, rather than requiring hardware modifications (such as increasing the magnitude of energy penalty terms). Recently, it has been shown [20] how 2-local perturbative gadgets can be used to achieve effectively large energy penalties using much weaker energy interactions, at the expensive of having additional qubits. It would be of great interest to see if such a technique could be applied here to develop a threshold theorem for scalable Hamiltonian-based quantum computing. Unfortunately, the technique in [20] requires a large overhead in the number of interaction terms per qubit, which is likely physically unrealistic and is in opposition to the physical assumptions we made in deriving our fidelity bounds. Nonetheless, it may be a fruitful avenue for future research.

2.7 Conclusion

In this chapter, we considered the energy penalty method of error suppression, i.e., the method of achieving error suppression by encoding a Hamiltonian using a quantum error-detecting code and adding a constant term that penalizes states outside of the codespace. We proved that this method does indeed work in principle. Specifically, we showed that, in the limit of an infinitely large energy penalty, the actual evolution of the system is precisely the evolution in the absence of unwanted control errors and environmental interactions, provided that the code can detect these errors. Moreover, we have provided some bounds governing the finite energy penalty scenario, allowing one to bound the energy penalty required to attain the desired evolution with good fidelity. We believe that these bounds can be improved, as supported by our numerical evidence for a single logical qubit, and leave their tightening as an interesting open problem. We hope that progress in this area will eventually lead to a practical fault-tolerant paradigm for Hamiltonian-based quantum computation.

2.8 Afterword

The content of this chapter has generated interest, and after its publication in [2], several papers advancing the topic have been published. In [21], Marvian and Lidar use a non-perturbative approach that achieves similar bounds as our perturbative derivation of Eq. (2.32). Of particular importance, using similar assumptions to the ones made in our derivation of Eq. (2.28), namely locality and “geometric locality” of the environment and its interactions, Marvian [22] showed that there exists a unitary describing the unwanted evolution within the codespace alone that, if it could be rectified, would allow for protection time scaling exponentially in E_P. This helps explain the observation we found numerically that for sufficiently large E_P, decoherence occurs only within the codespace. His paper also shows precisely how the degree of locality affects this protection.

Chapter appendices

2.A Beyond 1-local errors

In the main text of this chapter we focused on the simplest case, where V acts 1-locally on the system and the quantum error-detecting code can detect 1-qubit errors. In this appendix we show that this simplification is not necessary. As long as the error-detecting code can detect the errors that V causes, our infinite energy penalty theorem still holds. This includes, for example, the case where V acts k-locally and the code can detect k-local errors. Specifically, the only requirement on V is that

$$PVP = 0\,. \tag{2.34}$$

We now present a proof of this general case.

Define R_r (for $r = 0, \dots, n$) to be

$$R_r = \sum \left\{ A_1 \otimes \cdots \otimes A_n : A_i \in \{P_i, Q_i\} \text{ such that } |\{i : A_i = Q_i\}| = r \right\},$$

where as before, P_i is the codespace projector for the ith logical qubit and $Q_i = \mathbb{1} - P_i$. In other words, R_r is the sum of all terms, each of which is a tensor product of a total of n P_i's and Q_i's, one for each logical qubit, such that precisely r of these projectors are Q_i's. For example, $R_0 = P$, $R_n = Q_1 Q_2 \cdots Q_n$, and

$$R_1 = Q_1 P_2 \cdots P_n \quad + \quad \cdots \quad + \quad P_1 \cdots P_{n-1} Q_n\,.$$

Observe that the R_r are in fact a complete set of orthogonal projectors:

$$\begin{aligned} R_r^2 &= R_r \qquad \text{for all } r \\ R_r R_{r'} &= 0 \qquad \text{for } r \neq r' \\ \sum_{r=0}^{n} R_r &= \mathbb{1}\,, \end{aligned}$$

where the last equality can be obtained by expanding out $\mathbb{1} = \prod_i (P_i + Q_i)$.

Now, recall that $e^{iE_P \tau Q_i} P_i = P_i$ and that $e^{iE_P \tau Q_i} Q_i = e^{iE_P \tau} Q_i$. Therefore, using the definition of $\tilde{Q}$ in Eq. (2.5), we see that for any r,

$$\begin{aligned} U_P^\dagger(\tau) R_r &= e^{iE_P\tau\tilde{Q}} R_r \\ &= \prod_{i=1}^{n} e^{iE_P\tau Q_i} R_r \\ &= e^{irE_P\tau} R_r \end{aligned}$$

because each term in R_r consists of precisely r Q_i's. Applying $U_P^\dagger$ to $\mathbb{1} = \sum_{r=0}^{n} R_r$ therefore lets us write

$$U_P^\dagger(\tau) = \sum_{r=0}^{n} e^{irE_P\tau} R_r$$

so that applying $U_P^\dagger$ to VP gives

$$U_P^\dagger VP = \sum_{r=0}^{n} e^{irE_P\tau} R_r VP\,.$$

We now apply our key requirement of Eq. (2.34) to see that the $r = 0$ term is $R_0VP = PVP = 0$. Thus, we have

$$U_P^\dagger VP = \sum_{r=1}^{n} e^{irE_P\tau} R_r VP\,,$$

instead of the 1-local version in Eq. (2.20), and our formula for F from Eq. (2.21) therefore generalizes to

$$F(t) = \sum_{r=1}^{n} \int_0^t e^{irE_P\tau} R_r U_0^\dagger(\tau) V(\tau) U_0(\tau) P \mathrm{d}\tau\,.$$

Note that every term in $F(t)$ has a phase of $e^{irE_P\tau}$ for some $r \geqslant 1$. Applying the Riemann-Lebesgue lemma, we again conclude that in the infinite E_P limit, $F(t) \to 0$ and our theorem follows. This form of F may be useful in deriving finite energy penalty bounds in the case where we have a code that can protect against more than 1-local errors.

Chapter bibliography

[1] S. P. Jordan, E. Farhi, and P. W. Shor. *Error-correcting codes for adiabatic quantum computation.* Phys. Rev. A, 74(5):052322 [2006]. http://dx.doi.org/10.1103/PhysRevA.74.052322

[2] A. D. Bookatz, E. Farhi, and L. Zhou. *Error suppression in Hamiltonian-based quantum computation using energy penalties.* Phys. Rev. A, 92(2):022317 [2015]. http://dx.doi.org/10.1103/PhysRevA.92.022317

[3] D. Aharonov and M. Ben-Or. *Fault-tolerant quantum computation with constant error.* In *Proceedings of the Twenty-ninth Annual ACM Symposium on Theory of Computing*, STOC '97, pp. 176–188. ACM, New York, NY, USA [1997]. http://dx.doi.org/10.1145/258533.258579

[4] A. Y. Kitaev. *Quantum error correction with imperfect gates.* In O. Hirota, A. S. Holevo, and C. M. Caves (eds.), *Quantum Communication, Computing, and Measurement*, pp. 181–188. Springer US, Boston, MA, USA [1997]. http://dx.doi.org/10.1007/978-1-4615-5923-8_19

[5] A. Y. Kitaev. *Quantum computations: algorithms and error correction.* Russian Mathematical Surveys, 52(6):1191 [1997]. http://dx.doi.org/10.1070/RM1997v052n06ABEH002155

[6] D. Gottesman. *Stabilizer codes and quantum error correction.* Ph.d. thesis, Caltech, Pasadena, CA, USA [1997]. http://arxiv.org/abs/quant-ph/9705052

[7] J. Preskill. *Reliable quantum computers.* Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, 454(1969):385 [1998]. http://dx.doi.org/10.1098/rspa.1998.0167

[8] M. A. Nielsen and I. L. Chuang. *Quantum computation and quantum information.* Cambridge University Press, Cambridge, UK [2000]

[9] K. L. Pudenz, T. Albash, and D. A. Lidar. *Error-corrected quantum annealing with hundreds of qubits.* Nat Commun, 5:3243 [2014]. http://dx.doi.org/10.1038/ncomms4243

[10] B. Misra and E. C. G. Sudarshan. *The Zeno's paradox in quantum theory.* Journal of Mathematical Physics, 18(4):756 [1977]. http://dx.doi.org/10.1063/1.523304

[11] G. A. Paz-Silva, A. T. Rezakhani, J. M. Dominy, and D. A. Lidar. *Zeno effect for quantum computation and control.* Phys. Rev. Lett., 108(8):080501 [2012]. http://dx.doi.org/10.1103/PhysRevLett.108.080501

[12] L. Viola, E. Knill, and S. Lloyd. *Dynamical decoupling of open quantum systems.* Phys. Rev. Lett., 82(12):2417 [1999]. `http://dx.doi.org/10.1103/PhysRevLett.82.2417`

[13] D. A. Lidar. *Towards fault tolerant adiabatic quantum computation.* Phys. Rev. Lett., 100(16):160506 [2008]. `http://dx.doi.org/10.1103/PhysRevLett.100.160506`

[14] L. Viola and E. Knill. *Robust dynamical decoupling of quantum systems with bounded controls.* Phys. Rev. Lett., 90(3):037901 [2003]. `http://dx.doi.org/10.1103/PhysRevLett.90.037901`

[15] A. D. Bookatz, P. Wocjan, and L. Viola. *Hamiltonian quantum simulation with bounded-strength controls.* New J. Phys., 16(4):045021 [2014]. `http://dx.doi.org/10.1088/1367-2630/16/4/045021`. See Chapter 3 of this thesis.

[16] K. C. Young, M. Sarovar, and R. Blume-Kohout. *Error suppression and error correction in adiabatic quantum computation: techniques and challenges.* Phys. Rev. X, 3(4):041013 [2013]. `http://dx.doi.org/10.1103/PhysRevX.3.041013`

[17] P. Facchi, D. A. Lidar, and S. Pascazio. *Unification of dynamical decoupling and the quantum Zeno effect.* Phys. Rev. A, 69(3):032314 [2004]. `http://dx.doi.org/10.1103/PhysRevA.69.032314`

[18] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser. *Quantum computation by adiabatic evolution.* arXiv:quant-ph/0001106 [2000]. `http://arxiv.org/abs/quant-ph/0001106`

[19] J. Kempe, A. Kitaev, and O. Regev. *The complexity of the local Hamiltonian problem.* SIAM J. Comput., 35(5):1070 [2006]. `http://dx.doi.org/10.1137/S0097539704445226`

[20] Y. Cao and D. Nagaj. *Perturbative gadgets without strong interactions.* Quantum Info. Comput., 15(13-14):1197 [2015]. `http://arxiv.org/abs/1408.5881`

[21] I. Marvian and D. A. Lidar. *Quantum speed limits for leakage and decoherence.* Phys. Rev. Lett., 115(21):210402 [2015]. `http://dx.doi.org/10.1103/PhysRevLett.115.210402`

[22] I. Marvian. *Exponential suppression of decoherence and relaxation of quantum systems using energy penalty.* arXiv:1602.03251 [2016]. `http://arxiv.org/abs/1602.03251`

Chapter 3

Hamiltonian quantum simulation with bounded-strength controls

Hamiltonian simulation is the task of effectively changing a fixed system Hamiltonian H to a desired target Hamiltonian $\tilde{H}$ by applying some external time-dependent control Hamiltonian $H_c(t)$. In many situations, the available controls are limited. In particular, the control may be restricted to only part of the system, being unable to modify the environment. Moreover, the available control Hamiltonians may be only one-local, whereas the system Hamiltonian and the desired Hamiltonian may contain pairwise-interactions. Consequently, this simulation goal may need to be accomplished only approximately and stroboscopically (i.e., only at certain times), and the original system Hamiltonian H may be an essential ingredient for success (the controls alone being insufficient). Past schemes for simulating general Hamiltonians have relied on the ability to effect sequences of instantaneous, arbitrarily-strong control Hamiltonians (bang-bang control); however, in many realistic settings, such control is not available.

In this chapter, we propose dynamical control protocols for Hamiltonian simulation in many-body quantum systems that avoid instantaneous control operations and rely solely on realistic *bounded-strength control Hamiltonians*. Each simulation protocol consists of periodic repetitions of a basic control block, constructed as a suitable modification of an "Eulerian decoupling cycle," that would otherwise implement a trivial (zero) target Hamiltonian. For an open quantum system coupled to an uncontrollable environment, our approach may be employed to engineer an effective evolution that simulates a target Hamiltonian on the system, while suppressing unwanted decoherence to leading order.

We also present illustrative applications to both closed- and open-system simulation settings, with emphasis on the *simulation of non-local (two-body) Hamiltonians using only local (one-body) controls*. In particular, we provide simulation schemes applicable to Heisenberg-coupled spin chains exposed to general linear decoherence, and show how to simulate Kitaev's honeycomb lattice Hamiltonian starting from Ising-coupled qubits, as potentially relevant to the dynamical generation of a topologically protected quantum memory. Additional implications for quantum information processing are discussed.

To be more precise, a version of our result can be phrased as follows. Consider the situation in which one has a system obeying a time-independent traceless Hamiltonian H, but that one desires to simulate a system evolving under a time-independent traceless Hamiltonian $\tilde{H}$ for a time $\tilde{T}$. Suppose that one has access to bounded-strength control Hamiltonians such that they can implement, over a time Δ, unitaries $\{U_\gamma\}$ corresponding to the generators

$\Gamma = \{\gamma\}$ of a group $\mathcal{G}$. Suppose further that the group $\mathcal{G} = \{g\}$ was chosen (as is always, in principle, possible) to enable $\tilde{H}$ to be written in terms of H and the unitaries U_g (that correspond to the group elements $g \in \mathcal{G}$) in the form

$$\tilde{H} = \sum_{g\in\mathcal{G}} w_g U_g^\dagger H U_g$$

for some non-negative weights w_g. Then one can simulate a system obeying $\tilde{H}$ for time $\tilde{T}$ using a protocol that takes time $T_c = N\Delta + W\tilde{T}$, where $N = |\Gamma||\mathcal{G}|$ and $W = \sum_{g\in\mathcal{G}} w_g$. Specifically, the unitary evolution $U(t_M)$ of the system at times $t_M = MT_c$, for $M = 1, 2, \ldots$, obeys $U(t_M) = \tilde{U}(\tilde{t}_M) + O[(t_M\|H\|)^3]$ where $\tilde{U}(\tilde{t}_M)$ is the desired evolution corresponding to $\tilde{H}$ at the stroboscopic simulation times $\tilde{t}_M = M\tilde{T}$. This protocol is guaranteed to work if the representation mapping $\mathcal{G}$ to $\{U_g\}$ is irreducible, but we discuss – and provide explicit examples of – situations using reducible representations (which can be more efficient) and using more restricted controls. We also discuss the scenario of performing Hamiltonian simulation while simultaneously decoupling from an uncontrollable environment.

This chapter is adapted from [1], which was joint work with Lorenza Viola and Pawel Wocjan.

3.1 Introduction

The ability to accurately engineer the Hamiltonian of complex quantum systems is both a fundamental control task and a prerequisite for quantum simulation [2–6], as originally envisioned by Feynman. The basic idea underlying Hamiltonian simulation is to use an available quantum system, together with available (classical or quantum) control resources, to emulate the dynamical evolution that would have occurred under a different, desired Hamiltonian not directly accessible to implementation. In fact, this idea may be more generally applied to emulate a desired non-unitary (dissipative) evolution, see e.g. [7] for a recent survey. From a control-theory standpoint, the simplest setting is provided by *open-loop Hamiltonian engineering in the time domain* [8, 9], whereby coherent control over the system of interest is achieved solely based on suitably-designed time-dependent modulation (most commonly, sequences of control pulses), without access to ancillary quantum resources and/or measurement and feedback. While open-loop Hamiltonian engineering techniques have their origin and a long tradition in nuclear magnetic resonance (NMR) [10, 11], the underlying physical principles of "coherent averaging" have recently found widespread use in the context of quantum information processing (QIP), leading in particular to dynamical symmetrization and dynamical decoupling (DD) schemes for control and decoherence suppression in open quantum systems [12–17].

As applications for both universally programmable ("digital") and purpose-built ("analogue") quantum simulators continue to emerge across physics and chemistry [4–6, 18–20], and implementations become closer to experimental reality [21–23], it is imperative to expand the repertoire of available quantum-simulation procedures, and scrutinize the validity of the underlying control assumptions. While existing approaches differ considerably in their details and an extended comparison is not our scope here, we are specifically interested in advancing open-loop (analogue) Hamiltonian simulation schemes which, as mentioned, employ purely unitary control resources. With a few exceptions (notably, the use of so-called "perturbation theory gadgets" [24]), such schemes have relied thus far on the ability to implement sequences of effectively *instantaneous*, "bang-bang" (BB) control pulses [25–33].

Although this is a convenient and often reasonable first approximation, instantaneous pulses necessarily involve unbounded control amplitude and/or power, something which is out of reach for many control devices of interest and is fundamentally unphysical. In the context of DD, a general approach for achieving (to at least leading order) the same dynamical symmetrization as in the BB limit was proposed in [34], based on the idea of continuously applying bounded-strength control Hamiltonians according to an Eulerian cycle, performing so-called *Eulerian dynamical decoupling* (EDD). From a Hamiltonian engineering perspective, EDD protocols translate directly into bounded-strength simulation schemes for *specific* effective Hamiltonians – most commonly, the trivial (zero) Hamiltonian in the case of "non-selective averaging" for quantum memory (or "time-suspension" in NMR terminology). More recently, EDD has also served as the starting point for constructing bounded-strength *gate simulation* schemes in the presence of decoherence, i.e. so-called *dynamically corrected gates* (DCGs) for universal quantum computation [35–38].

In this work, we show that the approach of Eulerian control can be further systematically exploited to construct *bounded-strength Hamiltonian simulation schemes* for a broad class of target evolutions on both closed and open (finite-dimensional) quantum systems. In addition to being device-independent, our approach requires rather limited control resources – basically, the implementer need only apply *local* (single-qubit) Hamiltonians with bounded control strength to suitable subsets of target qubits. As such, these *Eulerian simulation* protocols may substantially expand the control toolbox for programming complex Hamiltonians into a broad variety of existing or near-term quantum simulators subject to realistic control assumptions.

The content is organized as follows. We begin in Sec. 3.2 by introducing the appropriate control-theoretic framework and by reviewing the basic principles of open-loop simulation via average Hamiltonian theory, along with its application to Hamiltonian simulation in the BB setting. Section 3.3 is devoted to constructing and analysing simulation schemes that employ bounded-strength controls: while Sec. 3.3.1 reviews the required background on EDD, Sec. 3.3.2 introduces our new Eulerian simulation protocols for a general closed quantum system, and Sec. 3.3.3 provides an explicit application to a simple two-qubit example. In Sec. 3.3.4 we address the important problem of Hamiltonian simulation in the presence of slowly-correlated (non-Markovian) decoherence, by identifying conditions under which a desired Hamiltonian may be enacted on the target system while simultaneously decoupling the latter from its environment, and by contrasting Eulerian simulation protocols with DCGs. Section 3.4 presents a number of illustrative applications of Eulerian simulation in interacting multi-qubit networks. In particular, we provide explicit protocols to simulate a large family of two-body Hamiltonians in Heisenberg-coupled spin systems that are additionally exposed to arbitrary single-qubit depolarization or dephasing. We also present a protocol to achieve Kitaev's honeycomb lattice Hamiltonian starting from Ising-coupled qubits. We conclude in Sec. 3.5.

3.2 Principles of Hamiltonian simulation

3.2.1 Control-theoretic framework

We consider a quantum system with associated Hilbert space $\mathcal{H}$, whose evolution is described by a time-independent Hamiltonian H. As mentioned, *Hamiltonian simulation* is the task of making this system evolve under some other time-independent target Hamiltonian, say, $\tilde{H}$. Without loss of generality, both the input and the target Hamiltonians may be taken to be traceless.

Two related scenarios are worth distinguishing for QIP purposes, depending on how the entire quantum system is related to the quantum system of interest, $\mathcal{S}$ (also referred to as the "target" henceforth):

• **Closed-system simulation**, in which case the entire system coincides with the quantum system of interest, $\mathcal{H} = \mathcal{H}_\mathcal{S}$, which undergoes purely *unitary* (coherent) dynamics;

• **Open-system simulation**, in which case the system is a bipartite system on $\mathcal{H} = \mathcal{H}_\mathcal{S} \otimes \mathcal{H}_\mathcal{B}$, where $\mathcal{B}$ represents an uncontrollable environment (also referred to as a "bath" henceforth), and the reduced dynamics of the target system $\mathcal{S}$ is *non-unitary*.

In both cases, we shall assume the target system $\mathcal{S}$ to be a network of interacting qudits, hence $\mathcal{H}_\mathcal{S} \simeq (\mathbb{C}^d)^{\otimes n}$, for finite d and n. In the general open-system scenario, the joint Hamiltonian on $\mathcal{H}$ may be expressed in the form

$$H = H_\mathcal{S} \otimes \mathbb{1}_\mathcal{B} + \mathbb{1}_\mathcal{S} \otimes H_\mathcal{B} + \sum_\alpha S_\alpha \otimes B_\alpha, \tag{3.1}$$

where the operators $H_\mathcal{S}$ and S_α act on $\mathcal{H}_\mathcal{S}$, $H_\mathcal{B}$ and B_α act on $\mathcal{H}_\mathcal{B}$, and all the bath operators are assumed to be norm-bounded, but otherwise unspecified (potentially unknown). The closed-system setting is recovered from Eq. (3.1) in the limit $S_\alpha = 0$. Likewise, we may express the target Hamiltonian $\tilde{H}$ in a similar form, with two simulation tasks being of special relevance: $\tilde{S}_\alpha = 0$, in which case the objective is to realize a desired system Hamiltonian $\tilde{H}_\mathcal{S}$ while decoupling $\mathcal{S}$ from its bath $\mathcal{B}$, thereby suppressing unwanted decoherence [14]; or, more generally, $H_\mathcal{S} \mapsto \tilde{H}_\mathcal{S}$ *and* $S_\alpha \mapsto \tilde{S}_\alpha$, where the simulated, dynamically symmetrized error generators $\tilde{S}_\alpha$ may for instance allow for decoherence-free subspaces or subsystems to exist [16, 39].

To accomplish the goal of Hamiltonian simulation, we modify the free dynamics of the system by an open-loop (i.e. feedback-free) controller acting on the target system according to

$$H \mapsto H + H_c(t). \tag{3.2}$$

The control Hamiltonian $H_c(t)$ is (in general) piecewise time-dependent,

$$H_c(t) = \begin{cases} h_1(t), & t \in [0, t_1] \\ h_2(t - t_1), & t \in [t_1, t_2] \\ h_3(t - t_2), & t \in [t_2, t_3] \\ \qquad \vdots & \end{cases},$$

successively applying Hamiltonians

$$h_u(t) = f_u(t) x_u \tag{3.3}$$

where the Hermitian operators $\{x_u\}$ and the real functions $\{f_u(t)\}$ represent the available control Hamiltonians and the corresponding, generally time-dependent, control inputs respectively.

Let t and $\tilde{t}$ denote the actual and the simulated time, respectively; that is, we wish to simulate the evolution under $\tilde{H}$ for time $\tilde{t}$ by actually evolving under H for time t. We allow for $t \neq \tilde{t}$ in order to account for time-overhead in the simulation – for instance, an overall scale factor $t = s\tilde{t}$, with $s > 0$, in the simplest case [28, 29]. If the operator $\tilde{H} - H$ is contained in the admissible control set $\{x_u\}$, the corresponding simulation problem is trivial and the desired time-evolution

$$\tilde{U}(\tilde{t}) = e^{-i\tilde{H}\tilde{t}}$$

can be exactly simulated continuously in time, with no overhead. However, this level of control need not be available in settings of interest. For example, in open quantum systems the control actions are necessarily restricted to the target system $\mathcal{S}$ alone, i.e. $H_c(t) = H_c(t) \otimes \mathbb{1}_{\mathcal{B}}$ in Eq. (3.2).

In line with the general idea of "analogue" quantum simulation [4,5], we shall assume in what follows a *restricted* set of control Hamiltonians (in a sense to be made more precise later) and focus on the task of *approximately* simulating the desired time evolution $\tilde{U}(\tilde{t})$ at a single *final time* $\tilde{t} = \tilde{T}_f$, or more generally, *stroboscopically* at multiple times, that is, at instants $\tilde{t} = \tilde{t}_M$, where

$$\tilde{t}_M = M\tilde{T}, \qquad M \in \mathbb{N},$$

and $\tilde{T}$ is a fixed minimum time interval. Choosing $\tilde{T}$ sufficiently small allows, in principle, any desired accuracy in the approximation to be met, with the limit $\tilde{T} \to 0$ formally recovering the continuous limit.

Let $U(t)$ denote the unitary time-evolution propagator associated with the total Hamiltonian $H + H_c(t)$ in Eq. (3.2),

$$U(t) = \mathcal{T} \exp\left\{-i\int_0^t [H + H_c(\tau)]\, d\tau\right\}, \tag{3.4}$$

where we have set $\hbar = 1$ and $\mathcal{T}$ indicates time-ordering, as usual. For a given pair $(H, \tilde{H})$, we shall provide sufficient conditions for $\tilde{H}$ to be "reachable" from H. Moreover, when this is satisfied, we show how to devise a *cyclic* control procedure $H_c(t)$ with some period T_c such that the resulting time-evolution at times $t = t_M$, where

$$t_M = MT_c\,, \qquad M \in \mathbb{N},$$

approximately yields the desired time-evolution at times $\tilde{t}_M$, that is

$$U(t_M) \approx \tilde{U}(\tilde{t}_M), \qquad M \in \mathbb{N}. \tag{3.5}$$

Note that for non-integer M, $U(MT_c) \not\approx \tilde{U}(M\tilde{T})$ in general – approximate equality is only guaranteed stroboscopically. If, for a given set of control resources and a *fixed* input Hamiltonian H, *arbitrary* target Hamiltonians $\tilde{H}$ are reachable, then the simulation scheme is referred to as *universal*. In this case, complete controllability must be ensured by the tunable Hamiltonians x_u in conjunction with the always-on "drift" $H_{\mathcal{S}}$ [9]. In contrast, we shall be especially interested in situations where control over $\mathcal{S}$ is more limited.

Hamiltonian simulation protocols are most conveniently constructed and analysed by moving to the "toggling" frame (interaction picture) that is rotating with the control propagator associated with $H_c(t)$,

$$U_c(t) = \mathcal{T} \exp\left\{-i\int_0^t H_c(\tau)\, d\tau\right\}.$$

The evolution in this frame is generated by the time-dependent control-modulated Hamiltonian

$$H'(t) = U_c^\dagger(t)\; H\; U_c(t) \tag{3.6}$$

with corresponding time-evolution operator

$$U'(t) = \mathcal{T} \exp \left\{ -i \int_0^t H'(\tau)\, d\tau \right\} .$$

The overall evolution, when written back in the Schrödinger picture, is given by

$$U(t) = U_c(t) U'(t) .$$

We design the periodic control Hamiltonian $H_c(t)$ such that $U_c(T_c) = \mathbb{1}$; consequently, $U_c(t)$ is also periodic, and in particular, $U_c(t_M) = \mathbb{1}$. Stroboscopically, the time-evolution in the Schrödinger picture and the rotating frame are therefore the same:

$$U(t_M) = U'(t_M).$$

Moreover, the periodicity of $U_c(t)$ implies that $H'(t)$ will also be periodic with period T_c, so that the stroboscopic dynamics is given by $U'(t_M) = U'(T_c)^M$; thus,

$$U(t_M) = U'(T_c)^M. \tag{3.7}$$

Average Hamiltonian theory [11,40] may be invoked to associate an effective *time-independent* Hamiltonian $\bar{H}$ to $U'(T_c)$ so that

$$U'(T_c) = \exp(-i\bar{H}T_c)$$

or, using (3.7) to relate this to $U(t_M)$,

$$U(t_M) = \exp(-i\bar{H}t_M) , \tag{3.8}$$

where $\bar{H}$ is determined by the Magnus expansion [41],

$$\bar{H} = \bar{H}^{(0)} + \bar{H}^{(1)} + \bar{H}^{(2)} + \cdots .$$

Explicitly, the leading-order term, determining evolution over a cycle up to the first order in time, is given by

$$\bar{H}^{(0)} = \frac{1}{T_c} \int_0^{T_c} H'(\tau) d\tau = \frac{1}{T_c} \int_0^{T_c} U_c^\dagger(\tau) H U_c(\tau)\, d\tau , \tag{3.9}$$

with (absolute) convergence being ensured as long as $t\|H\| < \pi$ [42]. Subject to the convergence condition, higher-order corrections for evolution over time t can also be upper-bounded by (see Lemma 4 in [43])

$$\Big\| \sum_{\ell=\kappa}^{\infty} t\bar{H}^{(\ell)} \Big\| \leqslant c_\kappa [\, (t\|H\|)^{\kappa+1}\,], \quad c_\kappa = O(1). \tag{3.10}$$

Ideally, one would like to achieve $\bar{H}T_c = \tilde{H}\tilde{T}$, so that equality would hold in Eq. (3.5). In what follows, we shall primarily focus on achieving *first-order simulation* instead, by engineering the control propagator $U_c(t)$ in such a way that $\bar{H}^{(0)}T_c = \tilde{H}\tilde{T}$, so that to first order

$$U(t_M) = e^{-i\bar{H}MT_c} \approx e^{-i\bar{H}^{(0)}MT_c} = e^{-i\tilde{H}M\tilde{T}} = \tilde{U}(\tilde{t}_M), \tag{3.11}$$

or more precisely, using Eq. (3.10) with $\kappa = 1$,

$$U(t_M) = e^{-i\bar{H}^{(0)}MT_c} + O[(MT_c\|H\|)^2] = \tilde{U}(\tilde{t}_M) + O[(MT_c\|H\|)^2].$$

Thus if one desires to evolve under $\tilde{H}$ for a time $\tilde{t}_M$ then they can, to first order, instead evolve under $H + H_c(t)$ for a time t_M.

In general, the accuracy of the approximation $\bar{H} \approx \bar{H}^{(0)}$ improves as the "fast control limit", $T_c \to 0$, is approached. Physically, this corresponds to requiring that the shortest control time scale (e.g., pulse separation) involved in the control sequence be sufficiently small relative to the shortest correlation time of the dynamics induced by H [40,44]. While the problem of constructing arbitrary high-order Hamiltonian simulation schemes is of separate interest, *second-order simulation* can be readily achieved, in principle, by ensuring that $U_c(t)$ is time-symmetric, namely, $U_c(t) = U_c(T_c - t)$ for $t \in [0, T_c]$. Since all odd-order Magnus corrections vanish in this case [40], it follows (by again using Eq. (3.10), with $\kappa = 2$), that $\bar{H}T_c = \tilde{H}\tilde{T} + O[(\|H\|T_c)^3]$, as desired.

3.2.2 Hamiltonian simulation with bang-bang controls

If instantaneous control pulses are realizable, BB Hamiltonian simulation provides the simplest control setting for achieving the intended objective in Eq. (3.5). Two main assumptions are involved: (i) First, we must be able to express the target Hamiltonian $\tilde{H}$ as

$$\tilde{H} = \sum_{j=1}^{N} w_j U_j^\dagger H U_j \,, \tag{3.12}$$

where $\{U_j\}$ are unitary operators on $\mathcal{S}$ and the weights $\{w_j\}$ are non-negative real numbers (not all zero); (ii) Second, we assume the available control resources in Eq. (3.3) include a discrete set of instantaneous pulses on $\mathcal{S}$, allowing the control propagator $U_c(t)$ to be a piecewise-constant function over $[0, T_c]$, so that the time-average in Eq. (3.9) can be expressed as a convex, positive-weighted sum (as will be shown below). Equation (3.12) may be interpreted as a definition for the target Hamiltonian $\tilde{H}$ to be considered reachable from H given BB unitary control on $\mathcal{S}$ alone. Such reachable Hamiltonians must then be at least as "disordered" as the input one in the sense of majorization [17,28,29].

Equation (3.12) leads naturally to the following BB simulation scheme. Given simulation weights $\{w_j\}$, define the following simulation intervals

$$\tau_j = w_j \tilde{T}, \tag{3.13}$$

and timing pattern

$$t_0 = 0\,, \quad t_j = \sum_{k=1}^{j} \tau_k. \tag{3.14}$$

We construct the piecewise-constant control propagator (for the basic simulation block to be repeated) as

$$U_c(t_{j-1} + \theta) = U_j\,, \quad \theta \in [0, \tau_j], \qquad j = 1, \ldots, N\,. \tag{3.15}$$

The length of the control cycle is evidently

$$T_c \equiv t_N = \sum_{j=1}^{N} w_j \tilde{T} = W\tilde{T} \tag{3.16}$$

where

$$W = \sum_{j=1}^{N} w_j > 0\,.$$

By using Eq. (3.9), it is immediate to verify that

$$\bar{H}^{(0)} = \frac{1}{T_c}\int_{t=0}^{T_c} U_c(t)^\dagger H U_c(t) dt = \frac{1}{T_c}\sum_{j=1}^{N} \tau_j U_j^\dagger H U_j = \frac{\tilde{T}}{T_c}\tilde{H}\,,$$

implementing the desired evolution, Eq. (3.11), with time overhead $s = W$, provided that the convergence conditions for first-order simulation under H are obeyed. Since in practice, even in the absence of any control errors, technological limitations always constrain the cycle duration to a *finite* minimum value $T_c > 0$ [44], such convergence conditions upper-bound the maximum simulated time $\tilde{t}_M$ up to which evolution under $\tilde{H}$ may be reliably simulated using the physical Hamiltonian H.

In analogy with BB DD schemes, realizing the prescription of Eq. (3.15) requires one to discontinuously change the control propagator $U_c(t)$ from U_j to $U_{j+1} = (U_{j+1}U_j^\dagger)U_j$, via an instantaneous BB pulse $U_{j+1}U_j^\dagger$ at the jth endpoint t_j. As a result, despite its conceptual simplicity, BB simulation is unrealistic whenever large control amplitudes are not an option, and so the evolution induced by H *during* the application of a control pulse must be considered from the outset. This demands redesigning the basic control block in such a way that the actions of H and $H_c(t)$ are simultaneously taken into account.

3.3 Hamiltonian simulation with bounded controls

3.3.1 Eulerian simulation of the trivial Hamiltonian

The key to overcome the disadvantages of BB Hamiltonian simulation is to ensure that the control propagator $U_c(t)$ varies continuously in time during each control cycle. We achieve this goal by invoking *Eulerian control design* [34]. We begin by revisiting how, for the special case of a target identity evolution (that is, $\tilde{H} = 0$, corresponding to a NO-OP gate, $\tilde{U}(\tilde{t}_M) = \mathbb{1}$), EDD can be naturally interpreted as a bounded-strength simulation scheme. In the next section, we can then take up the task of non-trivial target Hamiltonians.

In the Eulerian approach, the available control resources include a specially-chosen discrete set of unitary operations on $\mathcal{S}$, say, $\{U_\gamma\}$, $\gamma = 1, \ldots, L$, which are realized over a finite time interval Δ through application of some bounded-strength control Hamiltonians $\{h_\gamma(t)\}$, $\gamma = 1, \ldots, L$, with $|h_\gamma(t)| \leqslant h_{\max} < \infty$. That is,

$$U_\gamma = u_\gamma(\Delta)\,, \quad u_\gamma(\delta) = \mathcal{T}\exp\Big\{ -i\int_0^\delta h_\gamma(\tau)d\tau\Big\}\,. \tag{3.17}$$

Note that for any given $\{U_\gamma\}$, the choice of the control Hamiltonians $h_\gamma(t)$ is not unique, which allows for implementation flexibility.

The unitaries $\{U_\gamma\}$ are chosen to be the image of a generating set of a finite group under a faithful, unitary, projective[1] representation ρ [34]. To elaborate, let $\mathcal{G} = \{g\}$ be a finite group of order $|\mathcal{G}|$ with a generating set $\Gamma = \{\gamma\}$ of order $|\Gamma| = L$, so that each element in $\mathcal{G}$ may be written as an ordered product of elements in Γ. The representation ρ is a map from $\mathcal{G}$ into the set of unitaries on $\mathcal{H}_\mathcal{S}$,

$$\rho : g \mapsto \rho(g) = U_g \,, \tag{3.18}$$

with image $G = \{U_g\}$. The unitaries U_γ in Eq. (3.17) are $\rho(\gamma)$ for the generators, $\gamma \in \Gamma$. To help specify the order of application of these unitaries in the protocol, we use a directed graph as follows.

The *Cayley graph* $C(\mathcal{G}, \Gamma)$ of $\mathcal{G}$ relative to Γ is a directed graph whose vertices are labelled by the group elements of $\mathcal{G}$ and whose edges are "coloured" (labelled) by the generators. More precisely, there is an arrow from vertex g to another vertex g' by a directed edge labelled by generator γ if and only if $g' = \gamma g$. The number of edges in $C(\mathcal{G}, \Gamma)$ is thus equal to $N = |\Gamma||\mathcal{G}|$.

An *Eulerian cycle* is a traversal of a directed graph that uses each edge exactly once and starts and ends on the same vertex. Because a Cayley graph is regular, it always has an Eulerian cycle whose length is necessarily N [45, 46]. Without loss of generality, we assume that this cycle starts (and ends) at the identity element of $\mathcal{G}$. The Eulerian cycle can therefore be described by the ordered list of the edges (generators) it traverses, which we denote by

$$\mathcal{C} = (\gamma_1, \dots, \gamma_N).$$

Note that the ordered list of vertices (group elements) visited is then $(g_0, \dots, g_N)$ where $g_j = \gamma_j g_{j-1}$ for $j = 1, \dots, N$ and $g_N = g_0$ is the identity element.

Once a control Hamiltonian $h_\gamma(\tau)$ for implementing each generator γ as in Eq. (3.17) is chosen, an EDD protocol (which is for simulating $\tilde{H} = 0$) is constructed by assigning a cycle time as $T_c = N\Delta$ and by applying the control Hamiltonians $h_\gamma(t)$ sequentially in time, following the order determined by the Eulerian cycle $\mathcal{C}$. That is,

$$\begin{aligned} &U_c(0) = \mathbb{1}, \\ &U_c\big(j\Delta + \delta\big) = u_{\gamma_{j+1}}(\delta) U_c\left(j\Delta\right) \quad \text{for j} = 0, \dots, \text{N} - 1, \quad \delta \in [0, \Delta] \,, \end{aligned} \tag{3.19}$$

so that Eq. (3.17) implies

$$U_c(j\Delta) = U_{\gamma_j} U_c\left((j-1)\Delta\right) \,, \quad j = 1, \dots, N \,, \tag{3.20}$$

where U_{γ_j} is the image of the generator labelling the jth edge in $\mathcal{C}$. As established in [34], and as will be made clear in the next section, the lowest-order average Hamiltonian associated to the above EDD cycle has the form

$$\bar{H}^{(0)} = \Pi_\mathcal{G}[F_\Gamma(H)],$$

[1] A projective representation need only be a homomorphism up to phase, i.e., it obeys $U_{gg'} \propto U_g U_{g'}$ for $g, g' \in \mathcal{G}$, with proportionality rather than equality.

where for any operator A acting on $\mathcal{H}_{\mathcal{S}}$, the map

$$\Pi_{\mathcal{G}}(A) = \frac{1}{|\mathcal{G}|}\sum_{g\in\mathcal{G}} U_g^\dagger A U_g \tag{3.21}$$

projects onto the centralizer of $\mathcal{G}$ (i.e., $\Pi_{\mathcal{G}}(A)$ commutes with all $U_g \in G$), and

$$F_\Gamma(A) = \frac{1}{|\Gamma|}\sum_{\gamma\in\Gamma}\frac{1}{\Delta}\int_0^\Delta u_\gamma(\tau)^\dagger A u_\gamma(\tau) d\tau \tag{3.22}$$

implements an average of A over both the control interval and the group generators. Accordingly, bounded-strength simulation of $\tilde{H} = 0$ is achieved provided that the following DD condition is obeyed:

$$\Pi_{\mathcal{G}}\big[F_\Gamma(H)\big] = 0\,. \tag{3.23}$$

Note that, because H is traceless, F_Γ is trace-preserving, and $\Pi_{\mathcal{G}}(A)$ commutes with all $U_g \in G$ for any A, Eq. (3.23) is automatically ensured if the group representation acts irreducibly on $\mathcal{H}_{\mathcal{S}}$, since then by Schur's lemma, $\Pi_{\mathcal{G}}\big[F_\Gamma(H)\big] \propto \mathrm{Tr}(F_\Gamma(H)) = \mathrm{Tr}(H) = 0$. In general, however, we do not assume that ρ is irreducible.

3.3.2 Eulerian simulation protocols beyond NO-OP

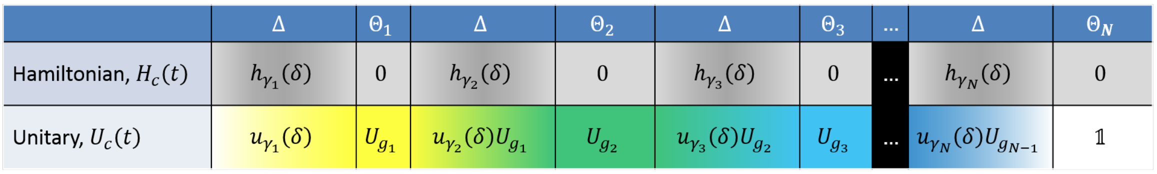
We now show how the Eulerian cycle method can be extended to bounded-strength simulation of a non-trivial class of target Hamiltonians (i.e. for $\tilde{H} \neq 0$). We assume that $\tilde{H}$ is *reachable* from H in the sense that it may be expressed as a convex mixture using the group representatives U_g:

$$\tilde{H} = \sum_{g\in\mathcal{G}} w_g U_g^\dagger H U_g\,, \qquad w_g \geqslant 0, \quad W = \sum_{g\in\mathcal{G}} w_g > 0\,; \tag{3.24}$$

this is similar to the condition in Eq. (3.12) for BB simulation, but with unitaries delineated by a representation ρ in Eq. (3.18). Similar to the EDD case of Sec. 3.3.1, we construct the desired protocol starting from an Eulerian cycle $\mathcal{C} = (\gamma_1, \ldots, \gamma_N)$ on $C(\mathcal{G}, \Gamma)$, but now incorporating the weights that appear in Eq. (3.24).

The idea behind Eulerian simulation is to append, to each of the N control slots that define an EDD ($\tilde{H} = 0$) scheme, a variable-length free-evolution (or "coasting") period implementing a zero-Hamiltonian control, in such a way that the net simulated Hamiltonian is modified from $\tilde{H} = 0$ to $\tilde{H} \neq 0$ as given in Eq. (3.24). A pictorial representation of the basic control block is given in Fig. 3-1. Each generator U_γ is implemented as in Eq. (3.17), with Δ the time duration required to implement each γ, i.e. to smoothly change (or "ramp-up") the control propagator $U_c(t)$ from any U_g to $U_{\gamma g} \cong U_\gamma U_g$ along the cycle, similar to Eq. (3.19) of EDD. We emphasize that all such "ramping-up" control intervals have the same length Δ. By contrast, each "coasting" interval is designed to keep the control propagator constant at U_g for a duration determined by the corresponding weight w_g, similar to Eq. (3.15) of BB simulation. Since the control is switched off during coasting, continuity of the control Hamiltonian $H_c(t)$ may be ensured, if desired, by additionally requiring that

$$h_\gamma(0) = h_\gamma(\Delta) = 0\,, \quad \gamma = 1, \ldots, L. \tag{3.25}$$



	Δ	Θ_1	Δ	Θ_2	Δ	Θ_3	...	Δ	Θ_N
Hamiltonian, $H_c(t)$	$h_{\gamma_1}(\delta)$	0	$h_{\gamma_2}(\delta)$	0	$h_{\gamma_3}(\delta)$	0	...	$h_{\gamma_N}(\delta)$	0
Unitary, $U_c(t)$	$u_{\gamma_1}(\delta)$	U_{g_1}	$u_{\gamma_2}(\delta)U_{g_1}$	U_{g_2}	$u_{\gamma_3}(\delta)U_{g_2}$	U_{g_3}	...	$u_{\gamma_N}(\delta)U_{g_{N-1}}$	$\mathbb{1}$

Figure 3-1: Schematics of an Eulerian simulation protocol. The basic control block consists of N time intervals, each involving a "ramping-up" subinterval of fixed duration Δ, during which $H_c(t) \neq 0$, followed by a "coasting" (free evolution) period of variable duration Θ_k, as defined in Eq. (3.27), during which no control is applied. During the jth ramping-up subinterval we apply h_{γ_j}, i.e., the control Hamiltonian that realizes the generator γ_j, smoothly changing the control propagator from $U_{g_{j-1}}$ to U_{g_j}. In this way, the control protocol corresponding to Eqs. (3.29)-(3.30) is implemented. By construction, a standard EDD cycle with $\tilde{H} = 0$ (as in Sec. 3.3.1) is recovered by letting $\Theta_k \to 0$ for all k, while in the limit $\Delta \to 0$ standard BB simulation of non-zero $\tilde{H}$ (as in Sec. 3.2.2) is implemented.

An Eulerian simulation protocol may be specified as follows. As before, let the jth time interval be denoted as $[t_{j-1}, t_j]$, $j = 1, \ldots, N$, with $t_0 = 0$. Part of this interval will be the Δ-long ramping-up stage, the other part will be the coasting stage whose length we discuss now. For $g \in \mathcal{G}$, let

$$\tau_g = w_g \tilde{T}\,, \tag{3.26}$$

similar to Eq. (3.13) of the BB case. Following the order of the Eulerian cycle $\mathcal{C} = (\gamma_1, \ldots, \gamma_N)$, we assign the duration of the jth coasting period as

$$\Theta_j = \frac{\tau_{g_j}}{|\Gamma|}\,, \tag{3.27}$$

where g_j is the jth vertex visited in the cycle, satisfying $g_j = \gamma_j g_{j-1}$. This results in the timing pattern

$$t_j = \sum_{k=1}^{j} (\Delta + \Theta_k) = j\Delta + \frac{1}{|\Gamma|} \sum_{k=1}^{j} \tau_{g_k}\,, \tag{3.28}$$

and, using Eqs. (3.26) and (3.24) and that each $g \in \mathcal{G}$ appears precisely $|\Gamma|$ times in the cycle,

$$T_c \equiv t_N = N\Delta + W\tilde{T}\,,$$

which can be compared to the BB timing pattern in Eqs. (3.14) and (3.16).

Over the interval from t_{j-1} to $t_j = t_{j-1} + \Delta + \Theta_j$, the Eulerian simulation control Hamiltonian is

$$\begin{aligned} H_c(t_{j-1} + \delta) &= h_{\gamma_j}(\delta) \qquad && \text{for } \delta \in [0, \Delta] \\ H_c(t_{j-1} + \Delta + \theta) &= 0 && \text{for } \theta \in [0, \Theta_j]. \end{aligned}$$

The resulting control propagator during the ramping-up and coasting subintervals is therefore

$$U_c\big(t_{j-1} + \delta\big) = u_{\gamma_j}(\delta) U_{g_{j-1}} \qquad \text{for } \delta \in [0, \Delta]\,, \tag{3.29}$$

$$U_c\big(t_{j-1} + \Delta + \theta\big) = U_{g_j} \qquad \text{for } \theta \in [0, \Theta_\mathrm{j}]\,. \tag{3.30}$$

As Eq. (3.28) for the cycle times makes clear, the resulting protocol may be equivalently interpreted in two ways: starting from an EDD cycle, corresponding to $T_c = N\Delta$ and $\tilde{H} = 0$, we introduce the coasting periods to allow for non-trivial simulated dynamics to emerge; or, starting from a BB simulation scheme for $\tilde{H}$, having cycle time $T_c = W\tilde{T}$, we introduce the ramping-up periods to allow for control Hamiltonians to be smoothly switched over time Δ. Either way, bounded-strength protocols imply a time-overhead $N\Delta$ relative to the BB case, recovering the BB limit as $\Delta \to 0$ as expected.

The resulting first-order Hamiltonian $\bar{H}^{(0)}$ under Eulerian simulation is derived by evaluating the time-average in Eq. (3.9) with the control propagator given by Eqs. (3.29)-(3.30). We obtain

$$\begin{aligned}
\bar{H}^{(0)} &= \frac{1}{T_c}\int_{t=0}^{T_c} U_c(t)^\dagger H U_c(t) dt \\
&= \frac{1}{T_c}\sum_{j=1}^{N}\Big[\int_{\delta=0}^{\Delta} U_c(t_{j-1}+\delta)^\dagger H U_c(t_{j-1}+\delta) d\delta \qquad (3.31)\\
&\qquad + \int_{\theta=0}^{\Theta_j} U_c(t_{j-1}+\Delta+\theta)^\dagger H U_c(t_{j-1}+\Delta+\theta) d\theta\Big] \\
&= \frac{1}{T_c}\sum_{j=1}^{N}\Big[\int_{\delta=0}^{\Delta} U_{g_{j-1}}^\dagger u_{\gamma_j}(\delta)^\dagger H u_{\gamma_j}(\delta) U_{g_{j-1}} d\delta + \int_{\theta=0}^{\Theta_j} U_{g_j}^\dagger H U_{g_j} d\theta\Big] \qquad (3.32)\\
&= \frac{1}{T_c}\sum_{g\in\mathcal{G}}\Big[U_g^\dagger\Big(\sum_{\gamma\in\Gamma}\int_{\delta=0}^{\Delta} u_\gamma(\delta)^\dagger H u_\gamma(\delta) d\delta\Big) U_g + |\Gamma|\frac{\tau_g}{|\Gamma|} U_g^\dagger H U_g\Big], \qquad (3.33)\\
&= \frac{|\mathcal{G}||\Gamma|\Delta}{T_c}\Pi_{\mathcal{G}}\big(F_\Gamma(H)\big) + \frac{\tilde{T}}{T_c}\sum_{g\in\mathcal{G}} w_g U_g^\dagger H U_g\,, \qquad (3.34)
\end{aligned}$$

where Π_G and F_Γ are defined in Eqs. (3.21) and (3.22). In this derivation, Eqs. (3.31) and (3.32) follow directly from the piecewise definition of $U_c(t)$. The next equality, Eq. (3.33), follows from two basic properties of Eulerian cycles: firstly, in traversing the Cayley graph, each group element g is left exactly once by a γ-labelled edge for each generator $\gamma \in \Gamma$; secondly, and consequently, the list $\{g_1, g_2, \ldots, g_N\}$ of the vertices that are being visited contains each element $g \in \mathcal{G}$ precisely $|\Gamma|$ times. Thus, provided that the DD condition of Eq. (3.23), namely $\Pi_{\mathcal{G}}\big(F_\Gamma(H)\big) = 0$, is obeyed, and recalling Eq. (3.24), we finally obtain

$$\bar{H}^{(0)} = \frac{N\Delta}{T_c}\Pi_{\mathcal{G}}\big(F_\Gamma(H)\big) + \frac{\tilde{T}}{T_c}\sum_{g\in\mathcal{G}} w_g U_g^\dagger H U_g = \frac{\tilde{T}}{T_c}\tilde{H}\,, \qquad (3.35)$$

which, as long as convergence holds, indeed achieves the intended first-order simulation goal, Eq. (3.11).

As noted earlier, we can improve the the simulation accuracy by symmetrizing $U_c(t)$ in time. In analogy to symmetrized EDD protocols [12], this can be easily accomplished by running the protocol and then suitably running it again in reverse. Specifically, let the duration of the coasting interval be changed as $\Theta_j \to \Theta_j/2$. Run the protocol as described above until time $t = N\Delta + \frac{1}{2}W\tilde{T}$, the $\frac{1}{2}$ resulting from the modified value of Θ_j. Then, from time $t = N\Delta + \frac{1}{2}W\tilde{T}$ until time $t = T_c = 2N\Delta + W\tilde{T}$, modify Eqs. (3.29)-(3.30) to be

$$\begin{aligned} U_c\big[T_c - (t_{j-1} + \Delta) + \delta\big] &= u_{\gamma_j}(\Delta - \delta)U_{g_{j-1}} \qquad \text{for } \delta \in [0, \Delta]\,, \\ U_c\big[T_c - (t_{j-1} + \Delta + \Theta_j) + \theta\big] &= U_{g_j} \qquad \text{for } \theta \in [0, \Theta_\mathrm{j}]\,, \end{aligned}$$

for $j = N, \ldots, 1$. Provided that one is able to implement $u_{\gamma_j}(\Delta - \delta)$, we again obtain

$$\bar{H}^{(0)} = 2\frac{N\Delta}{T_c}\Pi_{\mathcal{G}}\big[F_\Gamma(H)\big] + \frac{\tilde{T}}{T_c}\sum_{g\in\mathcal{G}} w_g U_g^\dagger H U_g\,,$$

while satisfying $U_c(t) = U_c(T_c - t)$ for $t \in [0, T_c]$, and hence ensuring that $\bar{H}^{(1)} = 0$.

3.3.3 Simple two-qubit example

By way of concrete illustration, it is useful to consider an explicit first-order Eulerian simulation scheme in the simplest instance of $n = 2$ qubits. Specifically, assume that the physical Hamiltonian is an isotropic Heisenberg coupling of the form

$$H = H_{\text{iso}} \equiv J(X \otimes X + Y \otimes Y + Z \otimes Z) = J(X_1X_2 + Y_1Y_2 + Z_1Z_2),$$

where J has units of energy and the third equality defines an equivalent compact notation. We are interested in a class of target XYZ Hamiltonians of the form

$$\tilde{H} = H_{\text{XYZ}} \equiv J_xX_1X_2 + J_yY_1Y_2 + J_zZ_1Z_2, \quad J_x, J_y, J_z \in \mathbb{R}. \tag{3.36}$$

For instance, $J_x = J_y$ but $J_z = 0$ corresponds to an isotropic XX model, whereas if $J_x = J_y \neq J_z \neq 0$, an XXZ interaction is obtained, the special value $J_z = -2J_x$ corresponding to the important case of a dipolar Hamiltonian. Our construction of a simulation protocol starts from observing that Hamiltonians in Eq. (3.36) are reachable from H, in the sense of Eq. (3.24), even limited to just *single-qubit control.*

Specifically, let $\mathcal{G} = \mathbb{Z}_2 \times \mathbb{Z}_2 = \mathbb{Z}_2^2$, and let the representation ρ map $(n, m) \in \mathcal{G}$ to $X^nZ^m \otimes \mathbb{1}$. That is, $\mathcal{G}$ is mapped (up to phase) to the set of unitaries

$$\{U_g\} = G_1 \equiv \{\mathbb{1}\otimes\mathbb{1}, X\otimes\mathbb{1}, Y\otimes\mathbb{1}, Z\otimes\mathbb{1}\} = \{\mathbb{1}, X_1, Y_1, Z_1\}. \tag{3.37}$$

Choosing the generators of $\mathcal{G}$ to be $\gamma_{x,1} = (1, 0) \mapsto X_1$ and $\gamma_{z,1} = (0, 1) \mapsto Z_1$, we assume that we have access to the control Hamiltonians

$$h_x(t) = f_x(t)X_1 \quad \text{and} \quad h_z(t) = f_z(t)Z_1\,,$$

where the control inputs $f_x(t)$ and $f_z(t)$ satisfy $f_u(0) = 0 = f_u(\Delta)$ and $\int_0^\Delta f_u(\tau)d\tau = \pi/2$, for $u = x, z$. Recalling Eq. (3.17), this yields the control propagators

$$\begin{aligned} u_x(\delta) &= \cos\left[\int_0^\delta f_x(\tau)d\tau\right]\mathbb{1} - i\,\sin\left[\int_0^\delta f_x(\tau)d\tau\right]X_1\,, \\ u_z(\delta) &= \cos\left[\int_0^\delta f_z(\tau)d\tau\right]\mathbb{1} - i\,\sin\left[\int_0^\delta f_z(\tau)d\tau\right]Z_1, \end{aligned}$$

with $u_x(\Delta) = X_1$ and $u_z(\Delta) = Z_1$ (up to phase), as desired.

Note that for any single-qubit Hamiltonians A and B, averaging over the unitary group in Eq. (3.37) results in the projection superoperator

$$\Pi_{\mathcal{G}}(A \otimes B) = \frac{1}{4} \sum_{U \in \{\mathbb{1}, X, Y, Z\}} U^\dagger A U \otimes B = \frac{1}{2} \mathrm{Tr}(A)\, \mathbb{1} \otimes B. \tag{3.38}$$

In general, the map F_Γ is trace-preserving and, in this case, it acts non-trivially only on the first qubit. Thus, F_Γ is trace-preserving on the first qubit. Since each term in H is traceless in the first qubit, the decoupling condition $\Pi_{\mathcal{G}}[F_\Gamma(H)] = 0$ follows directly from Eq. (3.38), even though the relevant representation ρ is, manifestly, reducible.

Having satisfied our main requirements, one can derive from Eq. (3.24) that the reachability of XYZ Hamiltonians in Eq. (3.36) is equivalent to the existence of a solution to the set of conditions

$$\begin{aligned} J(w_{\mathbb{1}} + w_{X_1} - w_{Y_1} - w_{Z_1}) &= J_x\,, \\ J(w_{\mathbb{1}} - w_{X_1} + w_{Y_1} - w_{Z_1}) &= J_y\,, \\ J(w_{\mathbb{1}} - w_{X_1} - w_{Y_1} + w_{Z_1}) &= J_z\,, \end{aligned} \tag{3.39}$$

for non-negative weights w_g. While infinitely many choices exist in general, one likely wishes for a solution that minimizes the total weight $W = \sum_g w_g$, keeping the simulation time overhead to a minimum. For instance, it is easy to verify that a dipolar Hamiltonian of the form

$$\tilde{H} = H_{\mathrm{dip}} \equiv -J\,(X_1X_2 + Y_1Y_2 - 2Z_1Z_2) \tag{3.40}$$

may be simulated with minimal time overhead by choosing weights

$$w_{\mathbb{1}} = \frac{1}{2}, \quad w_{X_1} = 0 = w_{Y_1}, \quad w_{Z_1} = \frac{3}{2}, \tag{3.41}$$

i.e.

$$\tilde{H} = \tfrac{1}{2}\, \mathbb{1}\, H\, \mathbb{1} + \tfrac{3}{2} Z_1 H Z_1.$$

The Cayley graph associated with the resulting Eulerian simulation protocol is depicted in Fig. 3-2, with the explicit timing structure of the control block as in Fig. 3-1 and $N = 2 \times 4 = 8$ control segments per block. It is worth observing that although the weights w_{X_1} and w_{Y_1} are zero in this particular case, *all* group members of $\mathcal{G}$ are nonetheless required, and the unitaries X_1 and Y_1 still appear in the simulation scheme (during the ramping-up subintervals, as evident from Eq. (3.29)). This is crucial to guarantee that the unwanted F_Γ term is projected out.

3.3.4 Eulerian simulation while decoupling from an environment

The ability to implement a desired Hamiltonian on the target system $\mathcal{S}$, while switching off (at least to leading order) the coupling to an uncontrollable environment $\mathcal{B}$, is highly relevant to realistic QIP applications. That is, with reference to Eq. (3.1), the objective is now to *simultaneously* achieve $\tilde{H}_{\mathcal{S}} = H_{\mathrm{target}}$ and $\tilde{S}_\alpha = 0$, using a unitary control operation $U_c(t)$ acting on $\mathcal{S}$ alone. Because the first-order Magnus term $\bar{H}^{(0)}$ is additive [recall Eq. (3.9)], it suffices to treat each summand of H individually, allowing us to write the average

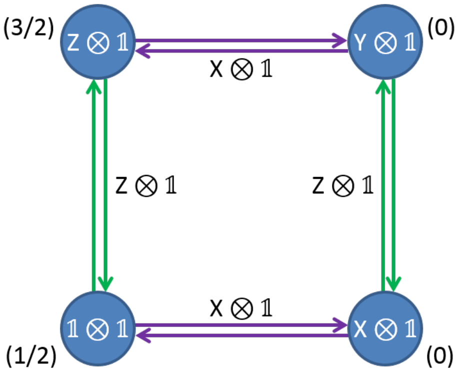


Figure 3-2: Cayley graph for the Eulerian simulation of the dipolar Hamiltonian in Heisenberg-coupled qubits. Vertices are labelled by group elements; edges are labelled by group generators. Numbers in parentheses next to vertices indicate the weights w_g of the corresponding group elements g in Eq. (3.41), which is proportional to the time $\tau_g = w_g\tilde{T}$ spent at vertex g during the coasting subinterval; see also Fig. 3-1.

Hamiltonian $\bar{H}^{(0)}$ in the form

$$\bar{H}^{(0)} = \bar{H}_{\mathcal{S}} \otimes \mathbb{1}_{\mathcal{B}} + \sum_{\alpha} \bar{S}_\alpha \otimes B_\alpha + \mathbb{1}_{\mathcal{S}} \otimes H_{\mathcal{B}}\,,$$

where for a generic operator A on $\mathcal{H}_{\mathcal{S}}$ we have denoted

$$\bar{A} = \frac{1}{T_c}\int_0^{T_c} U_c^\dagger(\tau) A U_c(\tau)\, d\tau\,.$$

We can then apply the analysis of Sec. 3.3.2 to the internal system Hamiltonian ($\bar{H}_{\mathcal{S}}$) and each error generator ($\bar{S}_\alpha$) separately to obtain, in both cases, a simulated operator of the form given in Eq. (3.34):

$$\bar{A} = \frac{N\Delta}{T_c}\Pi_{\mathcal{G}}\big[F_\Gamma(A)\big] + \frac{\tilde{T}}{T_c}\sum_{g\in\mathcal{G}} w_g U_g^\dagger A U_g\,.$$

Since the task is to decouple $\mathcal{S}$ from $\mathcal{B}$ while maintaining the non-trivial evolution due to $\tilde{H}_{\mathcal{S}} = H_{\text{target}}$, the reachability condition of Eq. (3.24) must now ensure that

$$\tilde{H}_{\mathcal{S}} = \sum_{g\in\mathcal{G}} w_g U_g^\dagger H_{\mathcal{S}} U_g\,, \tag{3.42}$$

$$0 = \sum_{g\in\mathcal{G}} w_g U_g^\dagger S_\alpha U_g\,, \quad \forall\alpha\,. \tag{3.43}$$

Similarly, it is necessary to extend the DD assumption of Eq. (3.23) to become

$$\Pi_{\mathcal{G}}\big[F_\Gamma(H_{\mathcal{S}})\big] = 0\,, \tag{3.44}$$

$$\Pi_{\mathcal{G}}\big[F_\Gamma(S_\alpha)\big] = 0\,, \quad \forall\alpha\,, \tag{3.45}$$

so that, similar to Eq. (3.35), $\bar{A} = (\tilde{T}/T_c)\tilde{A}$ holds for each of the summands in H. Altogether, we recover

$$\bar{H}^{(0)} = \frac{\tilde{T}}{T_c}\tilde{H}_{\mathcal{S}} \otimes \mathbb{1}_{\mathcal{B}} + \mathbb{1}_{\mathcal{S}} \otimes H_{\mathcal{B}},$$

simulating $\tilde{H}_{\mathcal{S}}$ while decoupling from the environment.

It is interesting in this context to highlight some similarities and differences to dynamically corrected gates (DCGs) [35], which also use Eulerian control as their starting point and are specifically designed to achieve a desired unitary evolution on the target system while simultaneously removing decoherence to leading order [35, 36, 38] or, in principle, arbitrarily high order [37]. By construction, the open-system simulation procedure just described *does* provide a first-order DCG implementation for the target gate $Q = \exp(-i\tilde{H}_{\mathcal{S}}\tilde{T})$; in particular, the requirement that Eqs. (3.42)-(3.43) be obeyed together (for the *same* weights w_g) is effectively equivalent to evading the "no-go theorem" for black-box DCG constructions established in [36], with the coasting intervals and the resulting "augmented" Cayley graph playing a role similar in spirit to a (first-order) "balance-pair" implementation. Despite these formal similarities, the key difference between the two approaches is that DCGs focus directly on synthesizing a particular unitary gate, as opposed to implementing some specified Hamiltonian. As one consequence, the goal of DCGs is generally to perform a desired unitary *final-time propagator* independently of the intervening dynamics, whereas in Hamiltonian simulation, one may be interested in the evolution at intermediate times as well (even if only strobscopically, at times $M\tilde{T}$ for $M \in \mathbb{N}$). As also discussed in [28, 29], gate synthesis is a weaker simulation notion in general, since inequivalent control protocols may lead to the same end-time propagator. Furthermore, while the internal system Hamiltonian, $H_{\mathcal{S}}$, is a crucial input in a Hamiltonian simulation problem, it is effectively treated as an unwanted error contribution in analytical DCG constructions, in which case complete controllability over the target system must be supplied by the controls alone. Although in more general (optimal-control inspired) DCG constructions [38], limited external control is assumed and $H_{\mathcal{S}}$ may become essential for universality to be maintained, emphasis remains, as noted above, on end-time synthesis of a target unitary propagator. Finally, a main intended application of DCGs is realizing low-error *single- and two-qubit gates* for use within fault-tolerant quantum computing architectures, as opposed to robust Hamiltonian engineering for many-body quantum simulators which is our focus here.

3.3.5 Eulerian simulation protocol requirements

Before presenting explicit applications, we summarize and critically assess the various requirements that should be obeyed for Eulerian simulation to achieve the intended control objective of Eq. (3.5) in a closed- or open-system setting:

1. *Time independence.* Both the internal Hamiltonian H and the target Hamiltonian $\tilde{H}$ are taken to be time-independent (and, without loss of generality, traceless).

2. *Reachability.* The target Hamiltonian $\tilde{H}$ must be reachable from H, i.e., there must be a control group $\mathcal{G}$, with a faithful, unitary projective representation mapping $g \mapsto \rho(g) = U_g$, such that Eq. (3.24) holds. For dynamically-corrected Eulerian simulation in the presence of an environment, this requires, as noted, that for the *same* weights $\{w_g\}$, the desired system Hamiltonian $\tilde{H}_{\mathcal{S}}$ is reachable from $H_{\mathcal{S}}$ while the trivial (zero) Hamiltonian is reachable from each error generator S_α separately, such that both Eqs. (3.42)-(3.43) hold together.

3. *Bounded control.* For each generator γ of the chosen control group $\mathcal{G}$, we need access to *bounded* control Hamiltonians $h_\gamma(t)$, such that application of $h_\gamma(t)$ over a time interval of duration Δ realizes the group representative $U_\gamma = \rho(\gamma) = u_\gamma(\Delta)$, additionally subject (if desired) to the continuity condition of Eq. (3.25).

4. *Decoupling conditions.* Suitable $\Pi_{\mathcal{G}}\big[F_\Gamma(H)\big] = 0$ DD conditions, Eq. (3.23) in a closed system or Eqs. (3.44)-(3.45) in the open-system error-corrected case, must be fulfilled, in order that undesired contributions to the simulated Hamiltonians be averaged out by symmetry to leading order.

5. *Time-efficiency.* If the choice of $\mathcal{G}$ is not unique for given $(H, \tilde{H})$, the smallest group should be chosen in order to keep the number of intervals per cycle, $N = |\mathcal{G}||\Gamma|$, to a minimum. In particular, *efficient* Hamiltonian simulation requires that $|\mathcal{G}|$ (hence also $|\Gamma|$) scales (at most) *polynomially* with the number of subsystems n. If, for fixed $\mathcal{G}$, the choice of the simulation weights $\{w_g\}$ is not unique, then the combination with the smallest total weight W should likely be chosen in order to minimize the time overhead.

The key simplification that the time-independence Assumption (1) introduces into the problem is that the periodicity of the control action is directly transferred to the toggling-frame Hamiltonian $H'(t)$ of Eq. (3.6), allowing one to simply focus on single-cycle evolution. Although this assumption is strictly not fundamental, general time-dependent Hamiltonians may need to be dealt with on a case-by-case basis (see also [47–49]). A situation of special practical relevance arises in this context for open systems exposed to *classical noise*, in which case $\mathcal{H}_\mathcal{B} \simeq \mathbb{C}$ and the system-bath interaction in Eq. (3.1) is effectively replaced by a classical, time-dependent stochastic field. Similar to DD and DCG schemes, Eulerian simulation protocols remain applicable as long as the noise process is stationary and is exhibiting correlations over sufficiently long time scales [12, 50].

The reachability Assumption (2) is a prerequisite for Eulerian Hamiltonian simulation schemes: to simulate a target Hamiltonian $\tilde{H}$ using this method, one must find a control group that appropriately relates $\tilde{H}$ to H. Although BB Hamiltonian simulation need not be group-based, most BB schemes follow this design principle too. Note that in the case of open systems, Assumption (2) also places demands on the relationship between S_α and $H_\mathcal{S}$; in particular, note that for non-zero $\tilde{H}_\mathcal{S}$, Eqs. (3.42)-(3.43) cannot hold simultaneously if $H_\mathcal{S}$ is a linear combination of S_α.

Assumption (3), restricting the admissible control resources to *physical* Hamiltonians with bounded amplitude (thus finite control durations, as opposed to instantaneous implementation of arbitrary group unitaries as in the BB case) is a basic assumption of the Eulerian control approach. As remarked, our premise is that the available Hamiltonian control is *limited*, restricted to only the target system (if the latter is coupled to an environment), and typically *non-universal* on $\mathcal{H}_\mathcal{S}$; in particular, we cannot directly express $\tilde{H} = H + H_c$ and apply $H_c = \tilde{H} - H$, or else the problem would be trivial. In addition to error-corrected Hamiltonian simulation in open quantum systems, scenarios of great practical interest may arise when the control Hamiltonians are subject to more restrictive *locality constraints* than the system and target Hamiltonians are (e.g., two-body simulation with only one-local controls, as in our examples in Secs. 3.3.3 and 3.4).

The required decoupling conditions in Assumption (4) are automatically obeyed if the representation ρ acts irreducibly on $\mathcal{H}_\mathcal{S}$. This follows from Schur's lemma, together with the fact that the map F_Γ defined in Eq. (3.22) is trace-preserving, and both $H_\mathcal{S}$ and S_α can be taken to be traceless. While convenient, however, irreducibility is not a requirement, as already demonstrated by the two-qubit example of Sec. 3.3.3. When the representation ρ is reducible, care must be taken in order to ensure that Assumption (4) is still obeyed. It should be stressed that this condition is *independent* of the target Hamiltonian $\tilde{H}$. Therefore, if the choice $(\mathcal{G}, \rho)$ works for one Eulerian simulation scheme (whether ρ is irreducible or not), then it can be used for Eulerian simulation with any target $\tilde{H}$ that belongs to the reachable set from H, i.e. that can satisfy Eq. (3.24).

We close this discussion by noting that it is always possible to find, for any finite-dimensional target system $\mathcal{S}$, a control group $\mathcal{G}$ for which both Assumptions (2) and (4) are satisfied, by resorting to the concept of a *transformer* [30]. A transformer is a pair $(\mathcal{G}, \rho)$, where $\mathcal{G}$ is a finite group and ρ is a faithful, unitary, projective representation from $\mathcal{G}$ into the set of unitaries on $\mathcal{H}_{\mathcal{S}}$, such that, for *any* traceless Hermitian operators A and B on $\mathcal{H}_{\mathcal{S}}$ with $A \neq 0$, there exist non-negative weights $\{w_g\}$ such that one may express

$$B = \sum_{g \in \mathcal{G}} w_g U_g^\dagger A U_g \,, \quad w_g \geqslant 0.$$

We illustrate this general idea in the simplest case of a single qubit, $\mathcal{H} = \mathcal{H}_{\mathcal{S}} = \mathbb{C}^2$. Let X, Y, Z denote the Pauli matrices and R the unitary matrix

$$R = \frac{i-1}{2} \begin{pmatrix} i & i \\ -1 & 1 \end{pmatrix}, \tag{3.46}$$

which corresponds to a rotation by an angle $4\pi/3$ about an axis $\hat{n} = (1,1,1)/\sqrt{3}$. Direct calculation shows that $R^3 = \mathbb{1}$ and that conjugation by R cyclically shifts the Pauli-matrices, i.e.,

$$R^\dagger X R = Y,$$
$$R^\dagger Y R = Z,$$
$$R^\dagger Z R = X.$$

Consider now the group $\mathcal{G}$ given by the presentation

$$\mathcal{G} = \langle x, y, z, r \mid x^2 = y^2 = z^2 = r^3 = 1, xz = y, r^{-1}xr = y, r^{-1}yr = z, r^{-1}zr = x \rangle.$$

Using the defining relations of this group, its elements can always be written as $x^a z^b r^c$, where $a, b \in \{0,1\}$ and $c \in \{0,1,2\}$. The assignment for ρ given by $x \mapsto X, y \mapsto Y, z \mapsto Z, r \mapsto R$ yields a faithful, unitary, irreducible projective representation. It is shown in [30] that the pair $(\mathcal{G}, \rho)$ defines a transformer, i.e., any 2×2 traceless matrix B may be reached from any fixed 2×2 traceless, nonzero matrix A, for suitable non-negative weights w_g. In the case of a transformer, Assumption (2) is satisfied by definition, and Assumption (4) is satisfied because the representation ρ will automatically be irreducible[2].

Since general transformer groups tend to be large, purely transformer-based simulation schemes are inefficient. In practice, given the system Hamiltonian $H_{\mathcal{S}}$, the challenge is to

[2] If ρ were reducible then there would exist a non-trivial invariant subspace $\mathcal{H}_{inv} \subset \mathcal{H}_{\mathcal{S}}$ such that $\rho(G)\mathcal{H}_{inv} \subseteq \mathcal{H}_{inv}$. Consequently, any Hamiltonian of the form $H = \sum_{ij} a_{ij} |v_i\rangle\langle v_j|$, where $\{|v_i\rangle\}$ is an orthonormal basis for $\mathcal{H}_{inv}$, can only be transformed to other Hamiltonians of this same form, and therefore (G, ρ) could not have been a transformer.

find a group $\mathcal{G}$ that grants a reasonably efficient scheme while satisfying Assumptions (2) and (4), subject to the ability to implement the required control operations. As we shall see in Sec. 3.4.3, transformer-inspired ideas may still prove useful in devising simulation schemes in the presence of additional symmetry conditions.

3.4 Illustrative applications

In this section, we analyse different paradigmatic Hamiltonian simulation tasks motivated by QIP applications. While a number of other interesting examples and generalizations may be envisioned (as also further discussed in the Conclusions), our goal here is to give a concrete sense of the usefulness and versatility of our Eulerian simulation approach in physically realistic control settings. In particular, we focus on achieving (first-order) *non-local Hamiltonian simulation using only bounded-strength local (single-qubit) control*, in both closed and open multi-qubit systems.

3.4.1 Eulerian simulation in closed Heisenberg-coupled qubit networks

We begin by noting that the analysis and simulation protocols described for two Heisenberg-coupled qubits in Sec. 3.3.3 may be easily generalized to a chain of n qubits (or spins) subject to nearest-neighbour (NN) homogeneous Heisenberg couplings, i.e., described by a physical Hamiltonian of the form

$$H = H_{\text{iso}}^{(\text{NN})} \equiv \sum_{i=1}^{n-1} J\,(X_i X_{i+1} + Y_i Y_{i+1} + Z_i Z_{i+1}) = \sum_{i=1}^{n-1} J\,\vec{\sigma}_i \cdot \vec{\sigma}_{i+1}, \tag{3.47}$$

where for later reference we have introduced the standard compact notation $\vec{\sigma}_i = (X_i, Y_i, Z_i)$ and we assume for concreteness that n is even. Similarly, we would like to simulate a nearest-neighbour XYZ Hamiltonian,

$$\tilde{H} = H_{\text{XYZ}}^{(\text{NN})} \equiv \sum_{i=1}^{n-1} (J_x X_i X_{i+1} + J_y Y_i Y_{i+1} + J_z Z_i Z_{i+1}) \,. \tag{3.48}$$

In this case, we need only change the unitary representation ρ on $\mathbb{Z}_2 \times \mathbb{Z}_2$ to be defined by the two generators

$$\begin{aligned}
\gamma_{x,\text{odd}} = (1,0) &\mapsto \quad X \otimes \mathbb{1} \otimes X \otimes \mathbb{1} \otimes \cdots \otimes X \otimes \mathbb{1} &&= X_1 X_3 \cdots X_{n-1} \\
\gamma_{z,\text{odd}} = (0,1) &\mapsto \quad Z \otimes \mathbb{1} \otimes Z \otimes \mathbb{1} \otimes \cdots \otimes Z \otimes \mathbb{1} &&= Z_1 Z_3 \cdots Z_{n-1},
\end{aligned}$$

resulting in the set of unitaries

$$\{U_g\} = G_{\text{odd}} \equiv \{\mathbb{1},\ X_1 X_3 \cdots X_{n-1},\ Y_1 Y_3 \cdots Y_{n-1},\ Z_1 Z_3 \cdots Z_{n-1}\}. \tag{3.49}$$

Physically, the required generators $\gamma_{x,\text{odd}}$ and $\gamma_{z,\text{odd}}$ correspond to control Hamiltonians that are still just sums of 1-local terms and that act non-trivially on odd qubits only:

$$h_x(t) = f_x(t)(X_1 + X_3 + \cdots + X_{n-1}), \quad h_z(t) = f_z(t)(Z_1 + Z_3 + \cdots + Z_{n-1}).$$

We expect that the design of Eulerian simulation schemes for more general scenarios where both the input and the target $(H, \tilde{H})$ are *arbitrary* two-body Hamiltonians (including, for instance, long-range couplings) will greatly benefit from the existence of combinatorial approaches for constructing efficient DD groups [48, 51]. A more in-depth analysis of this topic is, however, beyond our current purpose.

3.4.2 Error-corrected Eulerian simulation in open Heisenberg-coupled qubit networks

Imagine now that the Heisenberg-coupled system $\mathcal{S}$ considered in the previous section, i.e. Eq. (3.47), is coupled to an environment $\mathcal{B}$, and that the task is to achieve the desired XYZ Hamiltonian simulation for Eq. (3.48) while also removing *arbitrary linear decoherence* to leading order. The total input Hamiltonian has the form

$$H = H^{(\mathrm{NN})}_{\mathrm{iso}} \otimes \mathbb{1}_{\mathcal{B}} + \mathbb{1}_{\mathcal{S}} \otimes H_{\mathcal{B}} + \sum_{i=1}^{n} \vec{\sigma}_i \otimes \vec{B}_i, \quad \vec{B}_i = (B_{x,i}, B_{y,i}, B_{z,i}), \tag{3.50}$$

where $H_{\mathcal{B}}$ and $B_{u,i}$, for each i and $u = x, y, z$, are operators acting on $\mathcal{H}_{\mathcal{B}}$, whose norm is sufficiently small to ensure convergence of the relevant Magnus series, similar to first-order DCG constructions [35, 36]. The target Hamiltonian then reads

$$\tilde{H} = H^{(\mathrm{NN})}_{\mathrm{XYZ}} \otimes \mathbb{1}_{\mathcal{B}} + \mathbb{1}_{\mathcal{S}} \otimes H_{\mathcal{B}},$$

in terms of suitable coupling-strength parameters J_u as in Eq. (3.36). As before, we start by analysing the case of $n = 2$ qubits in full detail.

We must create a dynamically corrected simulation scheme that satisfies Eqs. (3.42) and (3.43). Our strategy to synthesize this scheme involves two stages: (i) We will first decouple $\mathcal{S}$ from $\mathcal{B}$, while leaving the system Hamiltonian $H_{\mathcal{S}} = H_{\mathrm{iso}}$ unaffected; (ii) We will then apply the closed-system protocol of Sec. 3.3.3 to convert H_{iso} into the target system Hamiltonian $\tilde{H}_{\mathcal{S}} = H_{\mathrm{XYZ}}$. Once a suitable group and weights are identified in this way, both stages are carried out simultaneously in application.

A suitable DD group able to suppress general linear decoherence is provided by $\mathcal{G}_{\mathrm{DD}} = \mathbb{Z}_2 \times \mathbb{Z}_2$, under the n-fold tensor product representation yielding

$$\{U_h\} = G_{\mathrm{GL}} \equiv \{\mathbb{1}, X^{(\mathrm{all})}, Y^{(\mathrm{all})}, Z^{(\mathrm{all})}\} \equiv \{\mathbb{1}, X_1X_2, Y_1Y_2, Z_1Z_2\}, \tag{3.51}$$

generated, for instance, by $\gamma_{x,\mathrm{all}} = (1,0) \mapsto X^{(\mathrm{all})} = X_1X_2$ and $\gamma_{z,\mathrm{all}} = (0,1) \mapsto Z^{(\mathrm{all})} = Z_1Z_2$, which can be implemented using 1-local Hamiltonians. Define the superoperator Φ_{DD} to operate on generic operators A on $\mathcal{H} = \mathcal{H}_{\mathcal{S}} \otimes \mathcal{H}_{\mathcal{B}}$ as

$$\Phi_{\mathrm{DD}}(A) = \frac{1}{4}\Big(A + X^{(\mathrm{all})} A X^{(\mathrm{all})} + Y^{(\mathrm{all})} A Y^{(\mathrm{all})} + Z^{(\mathrm{all})} A Z^{(\mathrm{all})}\Big)$$

corresponding to weights $\{w_h\}$ given by $w_{\mathbb{1}} = w_{X_1X_2} = w_{Y_1Y_2} = w_{Z_1Z_2} = 1/4$. Observe that $\Phi_{\mathrm{DD}}(\vec{\sigma}_i \otimes \vec{B}_i) = 0$, whereas $\Phi_{\mathrm{DD}}(H_{\mathrm{iso}} \otimes \mathbb{1}) = H_{\mathrm{iso}} \otimes \mathbb{1}$ because

$$[H_{\mathrm{iso}}, U_h] = 0, \quad \forall U_h \in G_{\mathrm{GL}}. \tag{3.52}$$

Thus, using these weights allows us to accomplish step (i), decoupling $\mathcal{S}$ from $\mathcal{B}$ while leaving $H_{\mathcal{S}}$ unaffected; moreover, it does so with the order of G_{GL} being minimal, $|G_{\mathrm{GL}}| = 4$ independent of n.

In step (ii), to convert H_{iso} to H_{XYZ}, we still rely on the group $\mathbb{Z}_2 \times \mathbb{Z}_2$, but now under a different representation. As in Sec. 3.3.3, we choose the representation yielding the set $G_1 = \{\mathbb{1}, X_1, Y_1, Z_1\}$ of Eq. (3.37), with the same single-qubit generators $\gamma_{x,1} = (1,0) \mapsto X_1$, $\gamma_{z,1} = (0,1) \mapsto Z_1$, and the corresponding weights $\{w_{g_1}\}$ determined by the solution of Eqs. (3.39). With these weights, define the superoperator Φ_1 to act as

$$\Phi_1(A) = w_{\mathbb{1}} A + w_{X_1} X_1 A X_1 + w_{Y_1} Y_1 A Y_1 + w_{Z_1} Z_1 A Z_1.$$

Then the combined action of the two superoperators Φ_{DD} and Φ_1 yields

$$\Phi_1[\Phi_{\text{DD}}(A)] = \sum_{U_{g_1} \in G_1} \sum_{U_h \in G_{\text{GL}}} w_{g_1} w_h U_{g_1}^\dagger U_h^\dagger A U_h U_{g_1} = \sum_{g \in \mathcal{G}} w_g U_g^\dagger A U_g, \tag{3.53}$$

where $\mathcal{G} = [\mathbb{Z}_2 \times \mathbb{Z}_2] \times [\mathbb{Z}_2 \times \mathbb{Z}_2] \simeq \mathbb{Z}_2^4$, with unitary representation elements corresponding to the *full* Pauli group on two qubits,

$$\{U_g\} = \{U_h U_{g_1} : U_h \in G_{\text{GL}}, U_{g_1} \in G_1\} \cong \{\mathbb{1}, X, Y, Z\}^{\otimes 2}.$$

With these corresponding weights $\{w_g\}$, we will satisfy both Eqs. (3.42) and (3.43) as required.

For the protocol to work, we additionally must show that (3.44) and (3.45) are satisfied. The above representation from $\mathcal{G}$ to $\{U_g\}$ is irreducible, with $\Pi_{\mathcal{G}}$ manifestly implementing the complete depolarizing channel on two qubits,

$$\Pi_{\mathcal{G}}(A) = \frac{1}{16} \sum_{g \in \mathcal{G}} U_g^\dagger A U_g = \frac{\text{Tr}(A)}{4} \mathbb{1}, \quad \forall A.$$

Since all of the system terms in H are traceless and F_Γ is trace-preserving on the system, we therefore see that the DD conditions of (3.44) and (3.45) are indeed satisfied. We thus conclude that the Eulerian simulation scheme accomplishes its goal. Since $|\mathcal{G}| = 16$ and $|\Gamma| = 4$, each simulation cycle will involve $N = 64$ time segments, with the number of non-zero weights (hence W and the simulation time-overhead) being determined by the details of the error model and/or the target Hamiltonian.

A practically important case, where simpler simulation schemes are possible, occurs if qubits couple to their environment along a fixed axis, effectively corresponding to a *purely dephasing* interaction – say, for concreteness, that $B_{x,i} \neq 0$ but $B_{y,i} = B_{z,i} = 0$ for $i = 1, 2$ in Eq. (3.50). A smaller DD group suffices in this case [36], namely $\mathcal{G}_{\text{DD}} = \mathbb{Z}_2$, represented again in terms of collective qubit rotations,

$$\{U_h\} = G_{\text{Deph}} \equiv \{\mathbb{1}, Z^{(\text{all})}\} = \{\mathbb{1}, Z_1 Z_2\}, \tag{3.54}$$

and generated by the single element $\gamma_{z,\text{all}}$, rather than requiring use of G_{GL} in Eq. (3.51). The commutation relationship in Eq. (3.52) is maintained, still allowing our two-step procedure to be followed. In this case, the combined group for simulation is $\mathcal{G} = \mathbb{Z}_2 \times [\mathbb{Z}_2 \times \mathbb{Z}_2] \simeq \mathbb{Z}_2^3$, with $|\mathcal{G}| = 8$, $|\Gamma| = 3$, *reducibly* represented on the two-qubit space as

$$\{U_g\} = \{U_h U_{g_1} : U_h \in G_{\text{Deph}}, U_{g_1} \in G_1\} \cong \{\mathbb{1},\, X_1,\, Y_1,\, Z_1,\, Z_1 Z_2,\, Y_1 Z_2,\, X_1 Z_2,\, Z_2\}. \tag{3.55}$$

Suppose, for instance, that the task is to simulate the dipolar Hamiltonian $\tilde{H}_{\mathcal{S}} = H_{\text{dip}}$ of Eq. (3.40). By following the above general procedure, with DD weights $\{w_h\}$ for G_{Deph}

alone given by

$$w_{\mathbb{1}} = \tfrac{1}{2},\ w_{Z_1Z_2} = \tfrac{1}{2},$$

and the G_1 weights $\{w_{g_1}\}$ given by Eq. (3.41) as

$$w_{\mathbb{1}} = \tfrac{1}{2},\ w_{Z_1} = \tfrac{3}{2},\ w_{X_1} = 0,\ w_{Y_1} = 0,$$

it is easy to see that Eq. (3.53) yields simulation weights $\{w_g = w_h w_{g_1}\}$ of

$$w_{\mathbb{1}} = \tfrac{1}{4},\ w_{Z_1} = \tfrac{3}{4},\ w_{Z_2} = \tfrac{3}{4},\ w_{Z_1Z_2} = \tfrac{1}{4},$$

with the remaining four weights equal to 0. While this implies that the simulation can now be achieved with only $N = 8 \times 3 = 24$ segments per cycle and minimum weight $W = 2$, care is needed in ensuring that the DD conditions in Eqs. (3.44)-(3.45) for $H_{\mathcal{S}} = H_{\text{iso}}$ and $S_\alpha = X_1, X_2$, are *still* obeyed despite the representation being reducible. This may be checked by inspection. In particular, suppose that in Eq. (3.17) we use control Hamiltonians $h_\gamma(t)$ proportional to X_1, Z_1, and $Z_1 + Z_2$ to produce the generators $\gamma_{x,1} \mapsto X_1$, $\gamma_{z,1} \mapsto Z_1$, and $\gamma_{z,\text{all}} \mapsto Z_1Z_2$, respectively. The fact that $\Pi_{\mathcal{G}}\big[F_\Gamma(H_{\text{iso}})\big] = 0$ and $\Pi_{\mathcal{G}}\big[F_\Gamma(X_1)\big] = 0$ follow from a trace argument like in Sec. 3.3.3. Each h_{γ_j} is 1-local (or a sum thereof), so each u_{γ_j} is a tensor product and conjugating by it is trace-preserving on the first qubit; consequently, because X_1 is traceless on the first qubit, all terms in $F_\Gamma(X_1)$ in Eq. (3.22) are traceless on the first qubit and are therefore eliminated by averaging over the G_1 unitaries in Eq. (3.37). Similarly for H_{iso}, which is also traceless on the first qubit. On the other hand, X_2 is not traceless on the first qubit, but $\Pi_{\mathcal{G}}\big[F_\Gamma(X_2)\big] = 0$ follows by analysing, for each of the h_{γ_j}, the structure of each $u^\dagger_{\gamma_j}(t)X_2u_{\gamma_j}(t)$ arising in $F_\Gamma(X_2)$: all such terms are linear combinations of X_2 and Y_2 and are all therefore eliminated by averaging over the G_{Deph} unitaries in Eq. (3.54). Thus, Eulerian Hamiltonian simulation in the presence of single-axis errors can be efficiently achieved.

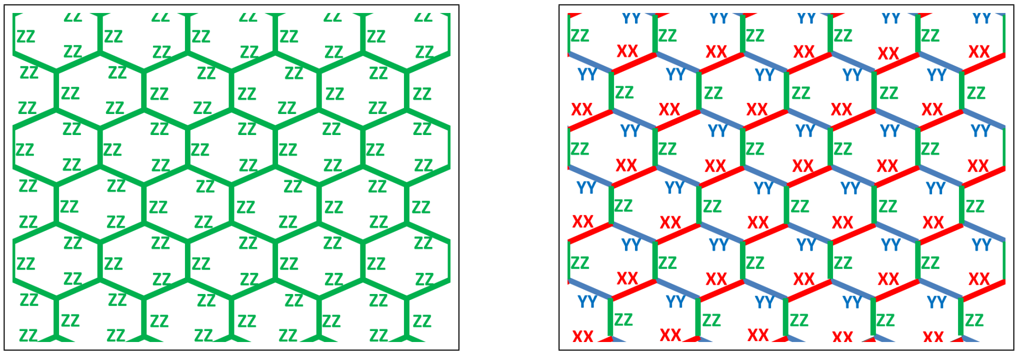
Again, the schemes we have presented for $n = 2$ can be generalized to a chain consisting of n spins that interact according to a nearest-neighbour Heisenberg interaction and are each linearly coupled to the environment, according to Eq. (3.50). As in Sec. 3.4.1, we can use the unitaries of G_{odd} in Eq. (3.49) for converting $H^{(\text{NN})}_{\text{iso}}$ into $H^{(\text{NN})}_{\text{XYZ}}$, and combine them with the DD unitaries G_{GL} in Eq. (3.51) for decoupling. This yields the representation from the group $\mathcal{G} \simeq \mathbb{Z}_2^4$ to the set

$$\{U_g\} = G_{\text{GL}} \times G_{\text{odd}},$$

corresponding to generators $\gamma_{x,\text{all}}, \gamma_{z,\text{all}}, \gamma_{x,\text{odd}}, \gamma_{z,\text{odd}}$, all of which can be implemented using only 1-local Hamiltonians. As before, each simulation cycle will consist, in the general case of arbitrary linear decoherence, of $N = 16 \times 4 = 64$ time segments. Despite the reducibility of the above representation (with the full Pauli group on n qubits consisting of 4^n elements), the DD conditions given by Eqs. (3.44)-(3.45) remain valid for reasons similar to those outlined for $n = 2$ under pure dephasing.

3.4.3 Eulerian simulation of Kitaev's honeycomb lattice Hamiltonian

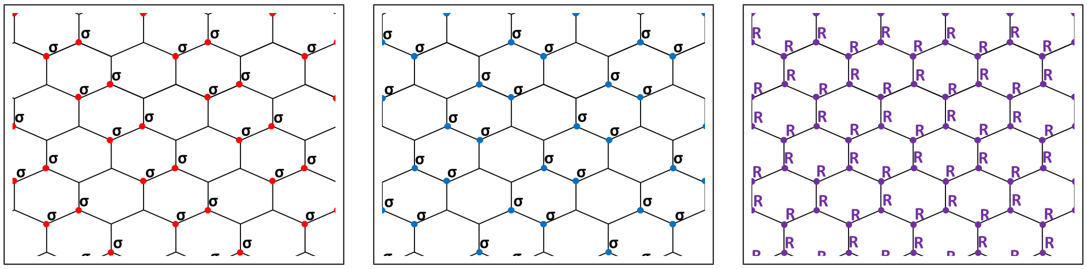
We return to Eulerian simulation in closed quantum systems, but tackle a more complicated Hamiltonian of paradigmatic relevance to topological quantum memories, namely, Kitaev's honeycomb lattice model [52]. Suppose that our system consists of a network of qubits arranged on a honeycomb lattice and interacting via near-neighbour Ising couplings. This



(a) H, the honeycomb lattice with ZZ couplings

(b) $\tilde{H}$, Kitaev's honeycomb lattice with XX, YY, and ZZ couplings

Figure 3-3: Input and target Hamiltonians on a 2D honeycomb lattice, where qubits are placed at each vertex. (a) The system Hamiltonian H describes a system where all adjacent vertices have ZZ Ising couplings. (b) The target Hamiltonian $\tilde{H}$ realizes Kitaev's honeycomb lattice model, with XX, YY, and ZZ couplings depending on the type of the edge.



(a) ρ_σ

(b) τ_σ

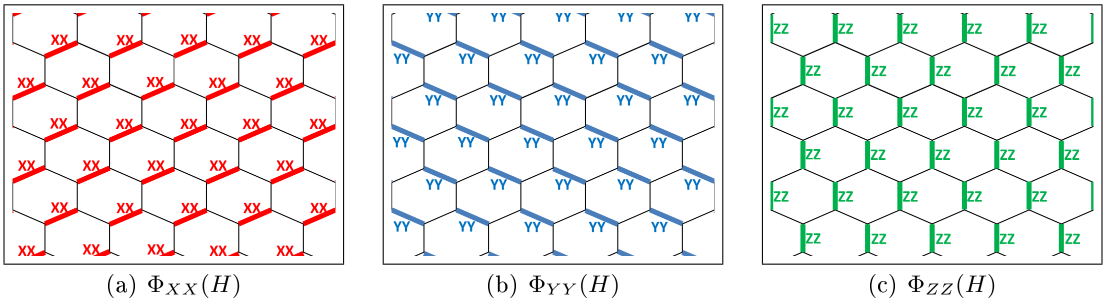
(c) R_{global}

Figure 3-4: (a) The unitary ρ_σ, with σ on the vertices of every second forward-slash and $\mathbb{1}$ on all other vertices, where σ is a fixed X, Y, or Z operator. When $\sigma = X$ this is the generator ρ_X. (b) The unitary τ_σ, with σ on the vertices of every second back-slash, where σ is a fixed X, Y, or Z operator. When $\sigma = X$ this is the generator τ_X. (c) The generator R_{global}, with R at every vertex.

Hamiltonian H is graphically displayed in Fig. 3-3(a), where vertices represent qubits and edges represent two-qubit couplings of the form $Z_k Z_\ell$, with vertices k and ℓ being adjacent in the graph and Z_k indicating, as before, the Pauli Z operator acting non-trivially only on qubit k. The target Hamiltonian $\tilde{H}$ is shown in Fig. 3-3(b), where some of the edges are now of the form $X_k X_\ell$ and $Y_k Y_\ell$. In accordance with the figure, we shall henceforth call the XX-edges *forward-slashes*, the YY-edges *back-slashes*, and the ZZ-edges *verticals*.

The basic idea to accomplish this simulation is to exploit the matrix R, given in Eq. (3.46), in conjunction with the symmetry of our problem: since all Hamiltonian terms are *precisely* two-local and of the homogeneous form $\sigma \otimes \sigma$, it will be possible to avoid using the full machinery of a transformer.

Consider the group $\mathcal{G}$ generated by the three unitaries, ρ_X, τ_X, and R_{global}, where ρ_X, shown in Fig. 3-4(a) with $\sigma = X$, has X's on every second forward-slash, τ_X, shown in Fig. 3-4(b) with $\sigma = X$, has X's on every second back-slash, and R_{global}, shown in Fig. 3-4(c), has



(a) $\Phi_{XX}(H)$ (b) $\Phi_{YY}(H)$ (c) $\Phi_{ZZ}(H)$

Figure 3-5: The actions of the superoperators (a) $\Phi_{XX}(H) = \frac{1}{2}R_{\text{global}}^{\dagger} H R_{\text{global}} + \frac{1}{2}(\rho_X R_{\text{global}})^{\dagger} H(\rho_X R_{\text{global}})$, leaving XX terms at forward-slashes only; (b) $\Phi_{YY}(H) = \frac{1}{2}R_{\text{global}}^{2\,\dagger} H R_{\text{global}}^2 + \frac{1}{2}(\tau_X R_{\text{global}}^2)^{\dagger} H(\tau_X R_{\text{global}}^2)$, leaving YY terms at back-slashes only; and (c) $\Phi_{ZZ}(H) = \frac{1}{2}\,\mathbb{1}^{\dagger}\,H\,\mathbb{1} + \frac{1}{2}(\rho_X\tau_X)^{\dagger} H(\rho_X\tau_X)$, leaving ZZ terms at verticals only.

R applied to every vertex. These unitaries can be generated by one-local Hamiltonians that we assume that we can implement. By repeatedly conjugating ρ_X and τ_X with R_{global}, we immediately see that we can also perform ρ_σ and τ_σ, shown in Figs. 3-4(a) and 3-4(b), for any $\sigma = X, Y, Z$. Note that up to phase, all such ρ and τ commute. Because conjugation by R maps Pauli matrices to Pauli matrices, for any Pauli σ we have $R\sigma = (R\sigma R^{-1})R = \sigma' R$, where σ' is another Pauli matrix. Thus, up to phase, we can write any element of $\mathcal{G}$ in the canonical form

$$U_g = \rho\tau R_{\text{global}}^a, \tag{3.56}$$

where $\rho \in \{\mathbb{1}, \rho_X, \rho_Y, \rho_Z\}$, $\tau \in \{\mathbb{1}, \tau_X, \tau_Y, \tau_Z\}$, $a \in \{0, 1, 2\}$, and R_{global}^a only appears on the right.

To construct an Eulerian simulation protocol we must ensure two things: we must be able to choose weights w_g so that $\tilde{H}$ is reachable from H, i.e., obeys Eq. (3.24), and we must ensure that the DD condition of Eq. (3.23) is also fulfilled. To accomplish the former, we start with the fact that

$$\frac{1}{2}\,\mathbb{1}(Z\otimes Z)\,\mathbb{1} + \frac{1}{2}(X\otimes\sigma)(Z\otimes Z)(X\otimes\sigma) = \begin{cases} Z\otimes Z & \text{if} \quad \sigma = \mathrm{X} \\ 0 & \text{if} \quad \sigma = \mathbb{1} \end{cases}.$$

Observe that in ρ_X, all forward-slash edges connect vertices that are acted upon by either $\mathbb{1}\otimes\mathbb{1}$ or $X\otimes X$, while all other edges connect vertices that are acted upon by $X\otimes\mathbb{1}$. Consequently, $\frac{1}{2}\,\mathbb{1}^{\dagger}\,H\,\mathbb{1} + \frac{1}{2}\rho_X^{\dagger} H \rho_X$ removes all Hamiltonian terms except for those along the forward-slashes; upon conjugating by R_{global}, we may then convert these surviving ZZ terms to XX terms, as desired. To summarize,

$$\Phi_{XX}(H) = \frac{1}{2}R_{\text{global}}^{\dagger} H R_{\text{global}} + \frac{1}{2}(\rho_X R_{\text{global}})^{\dagger} H(\rho_X R_{\text{global}})$$

gives the Hamiltonian shown in Fig. 3-5(a). Similarly, the effect of $\frac{1}{2}\,\mathbb{1}^{\dagger}\,H\,\mathbb{1} + \frac{1}{2}\tau_X^{\dagger} H\tau_X$ is to leave precisely the back-slash edges, which can be converted from ZZ to YY by conjugation by R_{global}^2. Thus,

$$\Phi_{YY}(H) = \frac{1}{2}R_{\text{global}}^{2\,\dagger} H R_{\text{global}}^2 + \frac{1}{2}(\tau_X R_{\text{global}}^2)^{\dagger} H(\tau_X R_{\text{global}}^2)$$

gives the Hamiltonian shown in Fig. 3-5(b). Lastly, it is not hard to see that the product $\rho_X\tau_X$ has X's on every second row of verticals; accordingly,

$$\Phi_{ZZ}(H)=\frac{1}{2}\mathbb{1}^\dagger\, H\,\mathbb{1}+\frac{1}{2}(\rho_X\tau_X)^\dagger H(\rho_X\tau_X)$$

isolates precisely the verticals, giving the Hamiltonian shown in Fig. 3-5(c). In this case, no R-conjugation is necessary since we wish to maintain ZZ edges along the verticals. Putting all these steps together, we conclude that

$$\begin{aligned}\tilde{H} &= \frac{1}{2}R^\dagger_{\text{global}}HR_{\text{global}}+\frac{1}{2}(\rho_X R_{\text{global}})^\dagger H(\rho_X R_{\text{global}})+\frac{1}{2}R^{2\,\dagger}_{\text{global}}HR^2_{\text{global}}\\ &+ \frac{1}{2}(\tau_X R^2_{\text{global}})^\dagger H(\tau_X R^2_{\text{global}})+\frac{1}{2}\,\mathbb{1}^\dagger\, H\,\mathbb{1}+\frac{1}{2}(\rho_X\tau_X)^\dagger H(\rho_X\tau_X),\end{aligned}$$

thus providing the desired weights for the Eulerian protocol. Since there are $|\Gamma|=3$ generators and, from Eq. (3.56), $|\mathcal{G}|=4\times4\times3=48$ group elements, each control block consists of $N=|\mathcal{G}||\Gamma|=144$ time intervals.

Lastly, we must verify that $\Pi_{\mathcal{G}}\big[F_\Gamma(H)\big]=0$ of Eq. (3.23) holds. Note that $F_\Gamma(H)$ acts via conjugating each vertex by unitaries (since the generating pulses are one-local), and since such an operation is trace-preserving at each vertex, this necessarily takes the precisely two-local terms in H to precisely two-local terms in $F_\Gamma(H)$. Since no one-local terms can arise, all terms are of the form $\sigma_u^{(k)}\otimes\sigma_v^{(\ell)}$, where k and ℓ are adjacent vertices, $\sigma_u,\sigma_v\in\{X,Y,Z\}$, and $\sigma_u^{(k)}$ denotes σ_u acting on vertex k. Thus, we may write

$$F_\Gamma(H)=\sum_{k,\ell\text{ adjacent}}\sum_{u,v}a^{(k,\ell)}_{u,v}\sigma_u^{(k)}\otimes\sigma_v^{(\ell)}$$

for some coefficients $a^{(k,\ell)}_{u,v}$. Due to the canonical form of our group elements, Eq. (3.56), the action of $\Pi_{\mathcal{G}}$ reads

$$\Pi_{\mathcal{G}}[F_\Gamma(H)]=\frac{1}{|\mathcal{G}|}\sum_{a=0}^{2}\sum_\tau\sum_\rho R^{a\dagger}\tau\rho\,F_\Gamma(H)\,\rho\tau R^a,$$

where we sum over $\rho=\mathbb{1},\rho_X,\rho_Y,\rho_Z$ and $\tau=\mathbb{1},\tau_X,\tau_Y,\tau_Z$. Just as we saw above that the map $\frac{1}{2}\,\mathbb{1}\,H\,\mathbb{1}+\frac{1}{2}\rho_X H\rho_X$ removes all non-forward-slash ZZ terms, the map $\sum_\rho \rho F_\Gamma(H)\rho$ depolarizes precisely one vertex of each pair of non-forward-slash vertices, and therefore suppresses all non-forward-slash terms. With only forward-slash terms remaining,

$$\sum_\tau \tau[\sum_\rho \rho F_\Gamma(H)\rho]\tau=0,$$

since the τ-sum removes all non-back-slash terms. Thus, we conclude that $\Pi_{\mathcal{G}}\big[F_\Gamma(H)\big]=0$, as desired.

3.5 Conclusion and outlook

We have shown that the Eulerian cycle technique, successfully employed in both dynamical decoupling schemes and dynamically corrected gates, can be extended to also enable

Hamiltonian quantum simulation with realistic *bounded-strength* controls. For given internal dynamics and control resources, we have characterized the family of reachable target Hamiltonians and provided constructive open-loop control protocols for stroboscopically implementing a desired evolution in the family with accuracy (at least) up to the second order in the sense of average Hamiltonian theory. We have additionally shown how Hamiltonian simulation may be accomplished in an open quantum system while *simultaneously* suppressing unwanted decoherence, provided that appropriate time-scale requirements and decoupling conditions are fulfilled, paving the way to dynamically corrected quantum simulation. The usefulness and flexibility of our Eulerian simulation techniques have been explicitly illustrated through several QIP-motivated examples involving both unitary and open-system dynamics on interacting qubit networks. In all cases, access to purely *local* (single-qubit) control Hamiltonians is assumed, subject to the finite-amplitude constraint and the ability to collectively apply such Hamiltonians to selected subsets of target qubits, for instance, qubits belonging to regular lattice patterns in one- or two-dimensional arrays. It is worth stressing that this level of control is in principle available in a variety of platforms for quantum simulation, with such "spatially periodic" control operations often being amenable to simple implementation, e.g. in optical lattices via globally applied pulses [4, 5, 7].

While it is our hope that our results may be of immediate relevance to ongoing efforts for developing and programming quantum simulators in the laboratory, several possible generalizations and further research questions are worth mentioning. As an additional simulation problem dual to the one we analysed for Heisenberg-coupled spin chains, exploring schemes where a target Heisenberg Hamiltonian is generated out of only Ising couplings would be of interest, given the experimental availability of the latter in existing large-scale trapped-ion simulators [22]. Likewise, an interesting issue is to explore the extent to which the proposed Eulerian approach may find application in simulation schemes for more exotic Hamiltonians involving higher-order interactions, notably, as arising in the Kitaev toric code [18] and lattice gauge theories [53].

From an implementation perspective, our present results call for further, dedicated analysis of the impact of *control errors*, which are inevitably present in experiments and effectively limit the maximum time over which the target dynamics may be reliably simulated. Since Eulerian control design is inherently robust (to leading order) against *systematic* Hamiltonian errors along a full cycle [34, 36], a similar degree of robustness may be expected for the "ramping-up" portion of a simulation block. While we also expect that the requisite timing precision in both "coasting" and "ramping-up" periods may be similar to the one demanded by dynamical decoupling protocols [54], a detailed analysis is needed to establish quantitative error bounds and avenues for enhancing fault-tolerance in a given physical architecture. Partly related to that, an ambitious goal is to determine whether Hamiltonian simulation schemes able to *guarantee* a minimum fidelity over arbitrarily long simulation times may be devised, in the spirit of [54] for the particular case of the zero Hamiltonian.

Building on existing results for dynamical decoupling schemes [49], the use and possible advantages of *randomized* simulation schemes in terms of robustness and/or efficiency may be yet another avenue of investigation, especially in connection with large control groups. Finally, it could be useful to explore whether bounded-strength simulation, as proposed here, may be made compatible with *open-loop filtering* techniques for modulating coupling strengths, such as recently proposed in [55], as well as in [56] in conjunction with non-unitary open-loop control via field gradients.

Chapter bibliography

[1] A. D. Bookatz, P. Wocjan, and L. Viola. *Hamiltonian quantum simulation with bounded-strength controls.* New J. Phys., 16(4):045021 [2014]. http://dx.doi.org/10.1088/1367-2630/16/4/045021

[2] R. P. Feynman. *Simulating physics with computers.* Int J Theor Phys, 21(6-7):467 [1982]. http://dx.doi.org/10.1007/BF02650179

[3] S. Lloyd. *Universal quantum simulators.* Science, 273(5278):1073 [1996]. http://dx.doi.org/10.1126/science.273.5278.1073

[4] I. Buluta and F. Nori. *Quantum simulators.* Science, 326(5949):108 [2009]. http://dx.doi.org/10.1126/science.1177838

[5] M. Georgescu, I. S. Ashhab, and F. Nori. *Quantum simulation.* Rev. Mod. Phys., 86(1):153 [2014]. http://dx.doi.org/10.1103/RevModPhys.86.153

[6] B. C. Sanders. *Efficient algorithms for universal quantum simulation.* In G. W. Dueck and D. M. Miller (eds.), *Reversible Computation*, number 7948 in Lecture Notes in Computer Science, pp. 1–10. Springer, Berlin, Germany [2013]. http://dx.doi.org/10.1007/978-3-642-38986-3_1

[7] M. Müller, S. Diehl, G. Pupillo, and P. Zoller. *Engineered open systems and quantum simulations with atoms and ions.* In E. A. Paul Berman and C. Lin (eds.), *Advances In Atomic, Molecular, and Optical Physics*, *Advances in Atomic, Molecular, and Optical Physics*, volume 61, pp. 1–80. Academic Press [2012]. http://dx.doi.org/10.1016/B978-0-12-396482-3.00001-6

[8] S. G. Schirmer. *Hamiltonian engineering for quantum systems.* In F. Allgüwer, P. Fleming, P. Kokotovic, A. B. Kurzhanski, H. Kwakernaak, A. Rantzer, J. N. Tsitsiklis, F. Bullo, and K. Fujimoto (eds.), *Lagrangian and Hamiltonian Methods for Nonlinear Control 2006*, number 366 in Lecture Notes in Control and Information Sciences, pp. 293–304. Springer, Berlin, Germany [2007]. http://dx.doi.org/10.1007/978-3-540-73890-9_23

[9] D. D'Alessandro. *Introduction to quantum control and dynamics.* Chapman & Hall/CRC Applied Mathematics & Nonlinear Science. Chapman and Hall/CRC [2007]. http://dx.doi.org/10.1201/9781584888833

[10] U. Haeberlen and J. S. Waugh. *Coherent averaging effects in magnetic resonance.* Phys. Rev., 175(2):453 [1968]. http://dx.doi.org/10.1103/PhysRev.175.453

[11] R. R. Ernst, G. Bodenhausen, and A. Wokaun. *Principles of nuclear magnetic resonance in one and two dimensions.* Clarendon Press, Oxford, UK [1987]

[12] D. A. Lidar, T. A. Brun, and T. Brun (eds.). *Quantum error correction.* Cambridge University Press, Cambridge, UK [2013]. `http://dx.doi.org/10.1017/CBO9781139034807`

[13] L. Viola and S. Lloyd. *Dynamical suppression of decoherence in two-state quantum systems.* Phys. Rev. A, 58(4):2733 [1998]. `http://dx.doi.org/10.1103/PhysRevA.58.2733`

[14] L. Viola, E. Knill, and S. Lloyd. *Dynamical decoupling of open quantum systems.* Phys. Rev. Lett., 82(12):2417 [1999]. `http://dx.doi.org/10.1103/PhysRevLett.82.2417`

[15] L. Viola, S. Lloyd, and E. Knill. *Universal control of decoupled quantum systems.* Phys. Rev. Lett., 83(23):4888 [1999]. `http://dx.doi.org/10.1103/PhysRevLett.83.4888`

[16] P. Zanardi. *Symmetrizing evolutions.* Phys. Lett. A, 258(2–3):77 [1999]. `http://dx.doi.org/10.1016/S0375-9601(99)00365-5`

[17] L. Viola. *Quantum control via encoded dynamical decoupling.* Phys. Rev. A, 66(1):012307 [2002]. `http://dx.doi.org/10.1103/PhysRevA.66.012307`

[18] C. M. Herdman, K. C. Young, V. W. Scarola, M. Sarovar, and K. B. Whaley. *Stroboscopic generation of topological protection.* Phys. Rev. Lett., 104(23):230501 [2010]. `http://dx.doi.org/10.1103/PhysRevLett.104.230501`

[19] H. Weimer, M. Müller, I. Lesanovsky, P. Zoller, and H. P. Büchler. *A Rydberg quantum simulator.* Nat Phys, 6(5):382 [2010]. `http://dx.doi.org/10.1038/nphys1614`

[20] S. Mostame, P. Rebentrost, A. Eisfeld, A. J. Kerman, D. I. Tsomokos, and A. Aspuru-Guzik. *Quantum simulator of an open quantum system using superconducting qubits: exciton transport in photosynthetic complexes.* New J. Phys., 14(10):105013 [2012]. `http://dx.doi.org/10.1088/1367-2630/14/10/105013`

[21] B. P. Lanyon, C. Hempel, D. Nigg, M. Müller, R. Gerritsma, F. Zähringer, P. Schindler, J. T. Barreiro, M. Rambach, G. Kirchmair, M. Hennrich, P. Zoller, R. Blatt, and C. F. Roos. *Universal digital quantum simulation with trapped ions.* Science, 334(6052):57 [2011]. `http://dx.doi.org/10.1126/science.1208001`

[22] J. W. Britton, B. C. Sawyer, A. C. Keith, C.-C. J. Wang, J. K. Freericks, H. Uys, M. J. Biercuk, and J. J. Bollinger. *Engineered two-dimensional Ising interactions in a trapped-ion quantum simulator with hundreds of spins.* Nature, 484(7395):489 [2012]. `http://dx.doi.org/10.1038/nature10981`

[23] P. Richerme, Z.-X. Gong, A. Lee, C. Senko, J. Smith, M. Foss-Feig, S. Michalakis, A. V. Gorshkov, and C. Monroe. *Non-local propagation of correlations in quantum systems with long-range interactions.* Nature, 511(7508):198 [2014]. `http://dx.doi.org/10.1038/nature13450`

[24] S. Bravyi, D. P. DiVincenzo, D. Loss, and B. M. Terhal. *Quantum simulation of many-body Hamiltonians using perturbation theory with bounded-strength interactions.*

Phys. Rev. Lett., 101(7):070503 [2008]. http://dx.doi.org/10.1103/PhysRevLett.101.070503

[25] P. Wocjan, D. Janzing, and T. Beth. *Simulating arbitrary pair-interactions by a given Hamiltonian: graph-theoretical bounds on the time-complexity.* Quantum Info. Comput., 2(2):117 [2002]. http://arxiv.org/abs/quant-ph/0106077

[26] J. L. Dodd, M. A. Nielsen, M. J. Bremner, and R. T. Thew. *Universal quantum computation and simulation using any entangling Hamiltonian and local unitaries.* Phys. Rev. A, 65(4):040301 [2002]. http://dx.doi.org/10.1103/PhysRevA.65.040301

[27] P. Wocjan, M. Rötteler, D. Janzing, and T. Beth. *Simulating Hamiltonians in quantum networks: Efficient schemes and complexity bounds.* Phys. Rev. A, 65(4):042309 [2002]. http://dx.doi.org/10.1103/PhysRevA.65.042309

[28] C. H. Bennett, J. I. Cirac, M. S. Leifer, D. W. Leung, N. Linden, S. Popescu, and G. Vidal. *Optimal simulation of two-qubit Hamiltonians using general local operations.* Phys. Rev. A, 66(1):012305 [2002]. http://dx.doi.org/10.1103/PhysRevA.66.012305

[29] L. Masanes, G. Vidal, and J. I. Latorre. *Time-optimal Hamiltonian simulation and gate synthesis using homogeneous local unitaries.* Quantum Info. Comput., 2(4):285 [2002]. http://arxiv.org/abs/quant-ph/0202042

[30] P. Wocjan, M. Rötteler, D. Janzing, and T. Beth. *Universal simulation of Hamiltonians using a finite set of control operations.* Quantum Info. Comput., 2(2):133 [2002]. http://arxiv.org/abs/quant-ph/0109063

[31] Y. C. Liu, Z. F. Xu, G. R. Jin, and L. You. *Spin squeezing: transforming one-axis twisting into two-axis twisting.* Phys. Rev. Lett., 107(1):013601 [2011]. http://dx.doi.org/10.1103/PhysRevLett.107.013601

[32] T. Tanamoto, V. M. Stojanović, C. Bruder, and D. Becker. *Strategy for implementing stabilizer-based codes on solid-state qubits.* Phys. Rev. A, 87(5):052305 [2013]. http://dx.doi.org/10.1103/PhysRevA.87.052305

[33] D. Becker, T. Tanamoto, A. Hutter, F. L. Pedrocchi, and D. Loss. *Dynamic generation of topologically protected self-correcting quantum memory.* Phys. Rev. A, 87(4):042340 [2013]. http://dx.doi.org/10.1103/PhysRevA.87.042340

[34] L. Viola and E. Knill. *Robust dynamical decoupling of quantum systems with bounded controls.* Phys. Rev. Lett., 90(3):037901 [2003]. http://dx.doi.org/10.1103/PhysRevLett.90.037901

[35] K. Khodjasteh and L. Viola. *Dynamically error-corrected gates for universal quantum computation.* Phys. Rev. Lett., 102(8):080501 [2009]. http://dx.doi.org/10.1103/PhysRevLett.102.080501

[36] K. Khodjasteh and L. Viola. *Dynamical quantum error correction of unitary operations with bounded controls.* Phys. Rev. A, 80(3):032314 [2009]. http://dx.doi.org/10.1103/PhysRevA.80.032314

[37] K. Khodjasteh, D. A. Lidar, and L. Viola. *Arbitrarily accurate dynamical control in open quantum systems.* Phys. Rev. Lett., 104(9):090501 [2010]. `http://dx.doi.org/10.1103/PhysRevLett.104.090501`

[38] K. Khodjasteh, H. Bluhm, and L. Viola. *Automated synthesis of dynamically corrected quantum gates.* Phys. Rev. A, 86(4):042329 [2012]. `http://dx.doi.org/10.1103/PhysRevA.86.042329`

[39] L. Viola, E. Knill, and S. Lloyd. *Dynamical generation of noiseless quantum subsystems.* Phys. Rev. Lett., 85(16):3520 [2000]. `http://dx.doi.org/10.1103/PhysRevLett.85.3520`

[40] U. Haeberlen. *High resolution NMR in solids: selective averaging*, *Adv. Magn. Res.*, volume 1. Academic Press [1976]. `http://dx.doi.org/10.1016/B978-0-12-025561-0.50001-0`

[41] W. Magnus. *On the exponential solution of differential equations for a linear operator.* Comm. Pure Appl. Math., 7(4):649 [1954]. `http://dx.doi.org/10.1002/cpa.3160070404`

[42] S. Blanes, F. Casas, J. A. Oteo, and J. Ros. *The Magnus expansion and some of its applications.* Phys. Rep., 470(5–6):151 [2009]. `http://dx.doi.org/10.1016/j.physrep.2008.11.001`

[43] K. Khodjasteh and D. A. Lidar. *Rigorous bounds on the performance of a hybrid dynamical-decoupling quantum-computing scheme.* Phys. Rev. A, 78(1):012355 [2008]. `http://dx.doi.org/10.1103/PhysRevA.78.012355`

[44] K. Khodjasteh, T. Erdélyi, and L. Viola. *Limits on preserving quantum coherence using multipulse control.* Phys. Rev. A, 83(2):020305 [2011]. `http://dx.doi.org/10.1103/PhysRevA.83.020305`

[45] B. Bollobás. *Modern graph theory*, *Graduate Texts in Mathematics*, volume 184. Springer, New York, NY, USA [1998]. `http://dx.doi.org/10.1007/978-1-4612-0619-4`

[46] C. Godsil and G. Royle. *Algebraic graph theory*, *Graduate Texts in Mathematics*, volume 207. Springer, New York, NY, USA [2001]. `http://dx.doi.org/10.1007/978-1-4613-0163-9`

[47] L. Viola and E. Knill. *Random decoupling schemes for quantum dynamical control and error suppression.* Phys. Rev. Lett., 94(6):060502 [2005]. `http://dx.doi.org/10.1103/PhysRevLett.94.060502`

[48] L. F. Santos and L. Viola. *Enhanced convergence and robust performance of randomized dynamical decoupling.* Phys. Rev. Lett., 97(15):150501 [2006]. `http://dx.doi.org/10.1103/PhysRevLett.97.150501`

[49] L. F. Santos and L. Viola. *Advantages of randomization in coherent quantum dynamical control.* New Journal of Physics, 10(8):083009 [2008]. `http://dx.doi.org/10.1088/1367-2630/10/8/083009`

[50] L. Cywiński, R. M. Lutchyn, C. P. Nave, and S. Das Sarma. *How to enhance dephasing time in superconducting qubits.* Phys. Rev. B, 77(17):174509 [2008]. http://dx.doi.org/10.1103/PhysRevB.77.174509

[51] M. Stollsteimer and G. Mahler. *Suppression of arbitrary internal coupling in a quantum register.* Phys. Rev. A, 64(5):052301 [2001]. http://dx.doi.org/10.1103/PhysRevA.64.052301

[52] A. Kitaev. *Anyons in an exactly solved model and beyond.* Annals of Physics, 321(1):2 [2006]. http://dx.doi.org/10.1016/j.aop.2005.10.005

[53] K. Stannigel, P. Hauke, D. Marcos, M. Hafezi, S. Diehl, M. Dalmonte, and P. Zoller. *Constrained dynamics via the Zeno effect in quantum simulation: Implementing non-Abelian lattice gauge theories with cold atoms.* Phys. Rev. Lett., 112(12):120406 [2014]. http://dx.doi.org/10.1103/PhysRevLett.112.120406

[54] K. Khodjasteh, J. Sastrawan, D. Hayes, T. J. Green, M. J. Biercuk, and L. Viola. *Designing a practical high-fidelity long-time quantum memory.* Nat. Commun., 4:2045 [2013]. http://dx.doi.org/10.1038/ncomms3045

[55] D. Hayes, S. T. Flammia, and M. J. Biercuk. *Programmable quantum simulation by dynamic Hamiltonian engineering.* New J. Phys., 16(8):083027 [2014]. http://dx.doi.org/10.1088/1367-2630/16/8/083027

[56] A. Ajoy and P. Cappellaro. *Quantum simulation via filtered Hamiltonian engineering: Application to perfect quantum transport in spin networks.* Phys. Rev. Lett., 110(22):220503 [2013]. http://dx.doi.org/10.1103/PhysRevLett.110.220503

Chapter 4

Improved bounded-strength decoupling schemes for local Hamiltonians

In this chapter, we address the task of switching off the Hamiltonian of a system by removing all internal and system-environment couplings. We propose dynamical decoupling schemes, that use only bounded-strength controls, for quantum many-body systems with local system Hamiltonians and local environmental couplings. To do so, we introduce the combinatorial concept of balanced-cycle orthogonal arrays (BOAs) and show how to construct them from classical error-correcting codes.

Like the Hamiltonian simulation protocols of Chapter 3, the *BOA decoupling schemes* derived here use an "Eulerian decoupling" technique to dictate a sequence of N bounded-strength control operations to be applied. However, the goal of decoupling differs from the content of Chapter 3 in that we are presently interested in simulating specifically the zero Hamiltonian, rather than general Hamiltonians. The decoupling schemes can be used for the purpose of quantum memories and may be useful as a primitive for more complex schemes, such as Hamiltonian simulation (by adapting the techniques of Chapter 3). Although more restricted in scope, the advantage of the BOA approach is that it exploits the presumed locality of the unwanted couplings in order to accomplish decoupling very efficiently, leading to much shorter control sequences than would otherwise be required. For use as a quantum memory, this enables greater availability of the preserved quantum information (since, as in Chapter 3, the decoupling method works only stroboscopically, simulating the absence of evolution only at certain times); for use as a primitive in more complex protocols, a shorter sequence translates to a less complicated building block that is faster and easier to implement with less accumulation of error.

For the case of n qubits and a 2-local Hamiltonian, the length, N, of the BOA decoupling schemes scale as $O(n \log n)$, improving over the previously best-known bounded-strength decoupling schemes that scaled quadratically with n. More generally, using BOAs constructed from families of BCH codes, we show that bounded-strength decoupling for any ℓ-local Hamiltonian, where $\ell \geqslant 2$, can be achieved using decoupling schemes of length at most $O(n^{\ell-1} \log n)$.

This chapter is adapted from [1], which was joint work with Martin Roetteler and Pawel Wocjan.

4.1 Introduction

Consider a quantum system of n interacting d-dimensional qudits with a time-independent (possibly unknown) Hamiltonian H acting on a Hilbert space $\mathcal{H} \cong (\mathbb{C}^d)^{\otimes n}$. We make the assumption that the system is ℓ-local, i.e. that H can be written as the sum of operators, each of which acts only on ℓ of the n qudits. In nature it is usually the case that ℓ is small even when n is large. Without loss of generality, we also take H to be traceless, and for technical reasons, we assume that d is a prime power (which includes the important case of qubits, i.e. $d = 2$).

We consider the task of *decoupling*, i.e. effectively switching off the Hamiltonian H (including removing any couplings to the environment) so that the system effectively evolves under the zero Hamiltonian. Such a task is important, for example, in the context of quantum memory, where one desires to preserve the state of a quantum system.

To achieve this task, we assume that the natural dynamics of the system can be modified by adjoining an open-loop (non-feedback) controller according to

$$H \mapsto H + H_c(t)\,.$$

In practice, physical limitations restrict the types of control Hamiltonians available for use. We consider the realistic setting in which $H_c(t)$ is only 1-local, i.e. due to our limited control of the system, H_c is the sum of operators that each act on only one qudit. We further impose the constraint that our control Hamiltonian $H_c(t)$ is limited to be bounded-strength, i.e. a sufficiently smooth bounded function. This is in contrast to the setting of bang-bang control in which $H_c(t)$ can be a discontinuous function that takes values of arbitrarily large norm. Our assumptions that the system Hamiltonian is an ℓ-local Hamiltonian acting on a system of n interacting qudits and that the control Hamiltonian is a 1-local bounded-strength Hamiltonian reflect the typical composite nature of quantum systems and their coupling locality as well as the limitations in implementing external controls.

Viola and Knill proposed a general method for bounded-strength decoupling; see [2] and [3, Chapter 4]. Their method, often referred to as *Eulerian decoupling*, relies on Eulerian cycles in Cayley graphs of a control group – a certain finite group of control unitaries that can be implemented by switching on control Hamiltonians, from a finite set of available control operations, for a fixed time. The Eulerian cycle dictates which control Hamiltonians are applied in the different time-slots of the decoupling protocol.

The Eulerian method, as introduced in [2], does not make it possible to directly leverage the fact that the system Hamiltonian is ℓ-local in order to obtain more efficient decoupling schemes. However, in the setting of bang-bang control there do exist efficient decoupling schemes that are specifically designed for composite quantum systems with ℓ-local system Hamiltonians; see [4–6] and [3, Chapter 15]. In these schemes, the specification of which bang-bang control unitaries are to be applied is chosen according to the entries of so-called *orthogonal arrays of strength ℓ*. They are matrices with the property that any submatrix formed by an arbitrary collection of ℓ rows satisfies a certain balancedness condition.

The work [7] presented a particular construction of decoupling schemes merging the approaches of Eulerian (bounded-strength) decoupling together with orthogonal array (bang-bang) decoupling. This construction yields schemes that require only bounded-strength controls and exploit the composite structure of the quantum system (namely, the locality of the system Hamiltonian) to achieve decoupling with fewer control operations. To do so, these schemes introduce the concept of so-called *Eulerian orthogonal arrays*.

The purpose of the present chapter is to further improve upon the method of [7] to obtain bounded-strength decoupling schemes of even greater efficiency. To this end, we first generalize the Eulerian method due to [2] by showing that it is also possible to achieve decoupling with the help of so-called *balanced cycles*, which encompass Eulerian cycles as a special case. We then show that bounded-strength decoupling of composite quantum systems with local Hamiltonians can be accomplished based on the new concept of *balanced-cycle orthogonal arrays*.

Note that all the schemes discussed above can also be applied to the situation of a general open quantum system with joint Hamiltonian H acting on a quantum system that is coupled to an uncontrollable environment. Such a Hamiltonian has the form

$$H = H_{\mathcal{S}} \otimes \mathbb{1}_{\mathcal{B}} + \mathbb{1}_{\mathcal{S}} \otimes H_{\mathcal{B}} + \sum_{\alpha} S_\alpha \otimes B_\alpha,$$

where the operators $H_{\mathcal{S}}$ and S_α act on the system and where the operators $H_{\mathcal{B}}$ and B_α act on the environment. We assume that the system Hamiltonian $H_{\mathcal{S}}$ and the operators S_α are all ℓ-local. The decoupling goal in this case is to effectively switch off the system Hamiltonian $H_{\mathcal{S}}$ and remove all couplings to the environment. If, using controls that act only on the system, one can effectively switch off all generic system Hamiltonians, then such an operation will switch off $H_{\mathcal{S}}$ and each S_α, thereby accomplishing decoupling.[1] For notational simplicity, the remainder of the chapter will therefore ignore the environment and treat only the case of effectively switching off an arbitrary ℓ-local operator H.

4.2 Description of the control-theoretic model

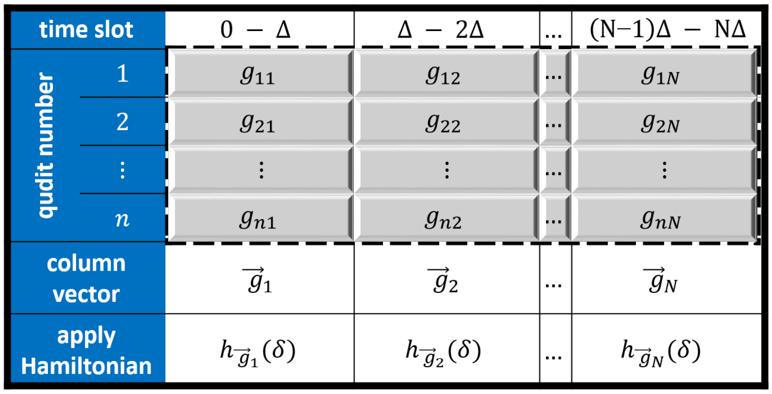
Consider the group $(\mathbb{F}_q, +)$, the additive group of the finite field of order $q = d^2$, where d (the dimension of the qudits) is some prime power. For the remainder of this chapter, let $\rho : \mathbb{F}_q \to \mathcal{U}(d)$ be a faithful, irreducible, unitary, projective[2] representation that maps the elements of $\mathbb{F}_q$ to $d \times d$ unitary matrices, say $\rho : g \mapsto U_g$. That q cannot be smaller than d^2 for such a representation will be justified later in Remark 4.2; that $q = d^2$ suffices is justified by the explicit example shown below.

We assume that for every $g \in \mathbb{F}_q$ we can implement U_g on any qudit of our system in the following sense: for every g, we can physically implement, over time $\delta \in [0, \Delta]$, a bounded-strength single-qudit Hamiltonian $h_g(\delta)$, corresponding to a single-qudit unitary evolution operator $u_g(\delta)$, such that $U_g = u_g(\Delta)$ where Δ is some fixed length of time. We assume that we can do this on any qudit and, moreover, that we can do so for each of the n qudits in parallel. Note that this assumption obeys the practical control limitations discussed earlier.

Of particular interest, in the case of qubits ($d = 2$, $q = 4$) we can consider the representation $\rho : \mathbb{F}_4 \to \mathcal{U}(2)$ that maps the four elements of $\mathbb{F}_4$ to the four 2×2 Pauli matrices $\{\mathbb{1}, X, Y, Z\}$. Thus, it is assumed that we can physically implement any Pauli operator on any qubit. Rather than assuming that $q = 4$, this chapter will treat q more generally; however the reader is invited to think of the special case of qubits if desired. For non-qubits, with $q > 4$, we can generalize this example as follows. For a prime p, define $\tilde{X} = \sum_{j=0}^{p-1} |j + 1 \bmod p\rangle\langle j|$ and $\tilde{Z} = \sum_{j=0}^{p-1} \omega^j |j\rangle\langle j|$, where ω is a p^{th} root of unity. For prime $d = p$, the map $(a, b) \mapsto \tilde{X}^a \tilde{Z}^b$ defines a faithful, irreducible, unitary, projec-

[1] The remaining Hamiltonian term of $\mathbb{1}_{\mathcal{S}} \otimes H_{\mathcal{B}}$ is inconsequential, as it does not affect the system at all.

[2] Projective representations need only be homomorphisms up to phase, i.e. obey $U_{g+h} \propto U_g U_h$ with proportionality rather than equality.



time slot		$0 - \Delta$	$\Delta - 2\Delta$	...	$(N-1)\Delta - N\Delta$
qudit number	1	g_{11}	g_{12}	...	g_{1N}
	2	g_{21}	g_{22}	...	g_{2N}
	$\vdots$	$\vdots$	$\vdots$	...	$\vdots$
	n	g_{n1}	g_{n2}	...	g_{nN}
column vector		$\vec{g}_1$	$\vec{g}_2$	...	$\vec{g}_N$
apply Hamiltonian		$h_{\vec{g}_1}(\delta)$	$h_{\vec{g}_2}(\delta)$	...	$h_{\vec{g}_N}(\delta)$

Figure 4-1: An $n \times N$ array, with each entry $g_{ij} \in \mathbb{F}_q$, shown within the dashed lines. Rows correspond to qudit numbers, columns to time slots (each of width Δ). This array encapsulates the control sequence, with $H_c(t) = h_{\vec{g}_j}(\delta)$ over $\delta \in [0, \Delta)$ during the interval $t \in \big[(j-1)\Delta, j\Delta\big)$.

tive representation from $\mathbb{Z}_d \times \mathbb{Z}_d$ to $\mathcal{U}(d)$. For a prime power $d = p^e$ (for some e), map $((a_1, b_1), \ldots, (a_e, b_e)) \mapsto \tilde{X}^{a_1}\tilde{Z}^{b_1} \otimes \cdots \otimes \tilde{X}^{a_e}\tilde{Z}^{b_e}$.

A decoupling protocol is defined by specifying a sequence of control Hamiltonians (equivalently, control unitaries) to be applied. As shown in Fig. 4-1, we construct an $n \times N$ array with entries from $\mathbb{F}_q$, which we regard as a sequence of N columns from $\mathbb{F}_q^n$. The jth column $\vec{g}_j = (g_{1j}, \ldots, g_{nj})^T$ corresponds to the jth time interval $\big[(j-1)\Delta, j\Delta\big]$ of our protocol, during which we apply the control Hamiltonian

$$h_{\vec{g}_j}(\delta) = h_{g_{1j}}(\delta) \otimes \mathbb{1} \otimes \cdots \otimes \mathbb{1} \quad + \cdots + \quad \mathbb{1} \otimes \cdots \otimes \mathbb{1} \otimes h_{g_{nj}}(\delta)$$

that gives rise to evolution $u_{\vec{g}_j}(\delta) = u_{g_{1j}}(\delta) \otimes \cdots \otimes u_{g_{nj}}(\delta)$ over $\delta \in [0, \Delta]$. In other words, for each $\delta \in [0, \Delta]$ and $j = 1, \ldots, N$, $H_c(t) = h_{\vec{g}_j}(\delta)$ where $t = (j-1)\Delta + \delta$. The total time required to apply the entire sequence, i.e. the control cycle length, is therefore $T_c = N\Delta$, at which point the control sequence can be repeated. Observe that for any $t = (j-1)\Delta + \delta$, the unitary evolution $U_c(t)$ corresponding to the control Hamiltonian consequently satisfies $U_c(t) = u_{\vec{g}_j}(\delta) U_c\big((j-1)\Delta\big)$.

According to average Hamiltonian theory [8–10], the resulting system evolution under $H + H_c(t)$ can be effectively approximated by

$$U(t) \approx e^{-i\bar{H}^{(0)}t}$$

at times t that are integer multiples of T_c, i.e. $t = mT_c$ for any $m \in \mathbb{N}$, where

$$\bar{H}^{(0)} = \frac{1}{T_c}\int_{t=0}^{T_c} U_c(t)^\dagger H U_c(t) dt$$

is time-independent and where $U_c(t)$ is the time evolution due to $H_c(t)$ alone. The goal of decoupling, therefore, is to choose $U_c(t)$ such that $\bar{H}^{(0)} = 0$ for any H. It is in this sense that we effectively switch off the Hamiltonian H. We refer the reader to Sec. 3.2.1 of Chapter 3 for a detailed description of the above control-theoretic model and the resulting effective time-evolution. We note, in particular, that although the approximation above is to leading order (in the Magnus expansion of $U_c(t)^\dagger H U_c(t)$), the second-order term may be eliminated by designing the control Hamiltonian to satisfy $U_c(t) = U_c(T_c - t)$ [10].

The efficiency of the protocol developed in this chapter is obtained by exploiting the composite structure of the Hamiltonian, namely the fact that H was assumed to be a *local* Hamiltonian. By definition, an ℓ-local Hamiltonian H on n qudits can be written as $H = \sum_k H_k$, where each H_k acts non-trivially on at most ℓ of the n qudits. In particular, the $\ell = 2$ case corresponds to Hamiltonians with only pairwise interactions. Our goal is to create a protocol that decouples each H_k simultaneously, and therefore decouples H. To see that this would work, observe that for any protocol $U_c(t)$,

$$\bar{H}^{(0)} = \frac{1}{T_c}\int U_c(t)^\dagger H U_c(t)dt = \sum_k \frac{1}{T_c}\int U_c(t)^\dagger H_k U_c(t)dt = \sum_k \bar{H}_k^{(0)}.$$

4.3 Balanced cycles

The success of the decoupling protocol introduced in this chapter will rely on some basic group theory, which we introduce now. Let $\mathcal{G}$ be an Abelian group with a generating set $\mathscr{S} \subset \mathcal{G}$, i.e. any element of $\mathcal{G}$ can be written as a sum of elements from $\mathscr{S}$.

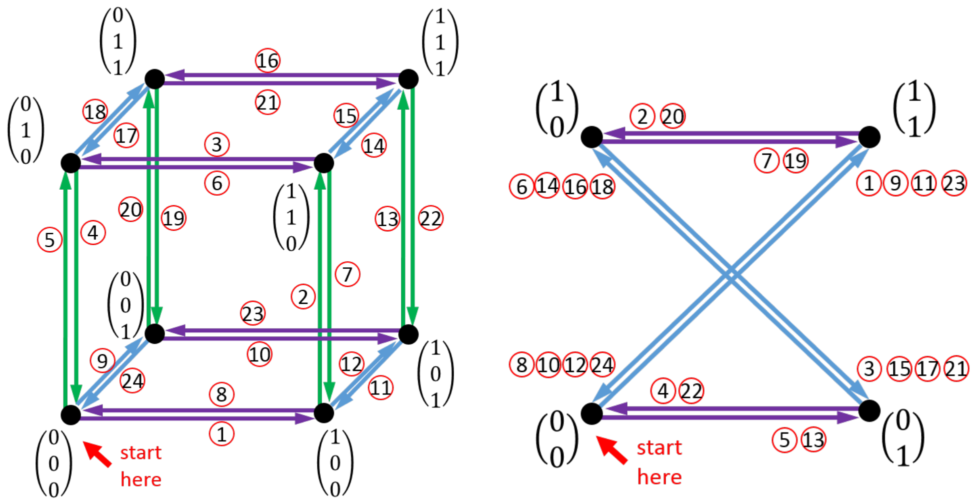
Definition 4.1 (Cayley graph). The *Cayley graph*, $\Gamma(\mathcal{G}, \mathscr{S})$, of $\mathcal{G}$ with respect to $\mathscr{S}$ is a directed graph whose vertices are labelled by the group elements and whose edges are labelled by the generators. More precisely, there is a directed edge labelled $\mathpzc{s}$ from vertex $\mathpzc{g} \in \mathcal{G}$ to vertex $\mathpzc{h} \in \mathcal{G}$ iff $\mathpzc{h} = \mathpzc{s} + \mathpzc{g}$ for the generator $\mathpzc{s} \in \mathscr{S}$.

Definition 4.2 (Cycle). A *cycle*, $\mathscr{C}$, on $\Gamma(\mathcal{G}, \mathscr{S})$ is a traversal on Γ that starts and ends on the same vertex. We describe the cycle by the ordered list $\mathscr{C}_\mathcal{G} = \left(\mathpzc{g}_0, \ldots, \mathpzc{g}_{N-1}\right)$ of elements from $\mathcal{G}$, indicating the order in which the elements are visited, with the understanding that the cycle visits $\mathpzc{g}_N = \mathpzc{g}_0$ immediately after visiting $\mathpzc{g}_{N-1}$. All the cycles in this chapter visit every vertex at least once, so we assume without loss of generality that the first vertex is the identity element, $\mathpzc{e}$, of $\mathcal{G}$. With this assumption we may equivalently represent the cycle $\mathscr{C}_\mathcal{G}$ by specifying the edges traversed, i.e. $\mathscr{C}_\mathscr{S} = (\mathpzc{s}_1, \ldots, \mathpzc{s}_N)$, where $\mathpzc{g}_j = \mathpzc{s}_j + \mathpzc{g}_{j-1}$ for $j = 1, \ldots, N$; observe that we differentiate between these representations by the subscript on $\mathscr{C}$, but they both refer to the same cycle.

Note that a cycle may visit vertices more than once and may traverse edges multiple times. We will be interested not only in the vertices, but also the specific labels leaving each vertex; we denote by $\mathpzc{g}\bullet\!\xrightarrow{\mathpzc{s}}$ the $\mathpzc{s}$-labelled edge leaving vertex $\mathpzc{g}$.

Definition 4.3 (Balanced cycle). We say that $\mathscr{C}$ is a *balanced cycle* if $\forall \mathpzc{s} \in \mathscr{S}, \exists \mu_\mathpzc{s} > 0$ such that $\forall \mathpzc{g} \in \mathcal{G}$, $\mathpzc{g}\bullet\!\xrightarrow{\mathpzc{s}}$ occurs exactly $\mu_\mathpzc{s}$ times; in other words, the cycle is balanced if it is balanced with respect to each label $\mathpzc{s} \in \mathscr{S}$ in the sense that it leaves each $\mathpzc{g}$ via label $\mathpzc{s}$ an equal number of times (independent of $\mathpzc{g}$). Consequently, each $\mathpzc{g}$ will appear in $\mathscr{C}$ precisely $\lambda = \sum_{\mathpzc{s}\in\mathscr{S}} \mu_\mathpzc{s}$ times, independent of $\mathpzc{g}$. Because a Cayley graph is a regular directed graph, it always has a balanced cycle whose length is then necessarily $N = \lambda|\mathcal{G}|$.

An important special case of a balanced cycle is an *Eulerian cycle* on $\Gamma(\mathcal{G}, \mathscr{S})$, for which $\mu_\mathpzc{s} = 1$ for every $\mathpzc{s} \in \mathscr{S}$. Examples of an Eulerian cycle and a non-Eulerian balanced cycle are


$\begin{pmatrix}0\\1\\1\end{pmatrix}$
$\begin{pmatrix}1\\1\\1\end{pmatrix}$
$\begin{pmatrix}0\\1\\0\end{pmatrix}$
$\begin{pmatrix}1\\1\\0\end{pmatrix}$
$\begin{pmatrix}0\\0\\1\end{pmatrix}$
$\begin{pmatrix}1\\0\\1\end{pmatrix}$
$\begin{pmatrix}0\\0\\0\end{pmatrix}$
$\begin{pmatrix}1\\0\\0\end{pmatrix}$
16
21
18
17
15
14
3
6
20
19
5
4
13
22
7
2
23
10
9
24
12
11
8
1
start here
$\begin{pmatrix}1\\0\end{pmatrix}$
$\begin{pmatrix}1\\1\end{pmatrix}$
2 20
7 19
6 14 16 18
1 9 11 23
8 10 12 24
3 15 17 21
4 22
5 13
$\begin{pmatrix}0\\0\end{pmatrix}$
$\begin{pmatrix}0\\1\end{pmatrix}$
start here

(a) Eulerian cycle

(b) Balanced cycle

Figure 4-2: (a) An Eulerian cycle on the Cayley graph $\Gamma\left(\mathbb{Z}_2^3, \left\{ \left(\begin{smallmatrix}1\\0\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}0\\1\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}0\\0\\1\end{smallmatrix}\right) \right\} \right)$, i.e. a balanced cycle in which each edge label leaves each vertex precisely once. Vertices correspond to the eight elements of $\mathbb{Z}_2^3$. Edge labels correspond to the three generators, namely $\left(\begin{smallmatrix}1\\0\\0\end{smallmatrix}\right)$ (purple), $\left(\begin{smallmatrix}0\\1\\0\end{smallmatrix}\right)$ (green), and $\left(\begin{smallmatrix}0\\0\\1\end{smallmatrix}\right)$ (blue). The cycle starts at $\left(\begin{smallmatrix}0\\0\\0\end{smallmatrix}\right)$ and follows the path indicated (in ascending numerical order) by the circled integers (red). (b) A balanced cycle on the Cayley graph $\Gamma\left(\mathbb{Z}_2^2, \left\{ \left(\begin{smallmatrix}0\\1\end{smallmatrix}\right), \left(\begin{smallmatrix}1\\1\end{smallmatrix}\right) \right\} \right)$. Vertices correspond to the four elements of $\mathbb{Z}_2^2$. Edge labels correspond to the two generators, namely $\left(\begin{smallmatrix}0\\1\end{smallmatrix}\right)$ (purple) and $\left(\begin{smallmatrix}1\\1\end{smallmatrix}\right)$ (blue). The cycle starts at $\left(\begin{smallmatrix}0\\0\end{smallmatrix}\right)$ and follows the path indicated (in ascending numerical order) by the circled integers (red). Observe that the cycle is indeed balanced: for each of the two edge labels, the edges leave each vertex the same number of times, irrespective of vertex. Specifically, the $\left(\begin{smallmatrix}0\\1\end{smallmatrix}\right)$ label leaves each vertex precisely $\mu_{\left(\begin{smallmatrix}0\\1\end{smallmatrix}\right)} = 2$ times, while the $\left(\begin{smallmatrix}1\\1\end{smallmatrix}\right)$ label leaves each vertex precisely $\mu_{\left(\begin{smallmatrix}1\\1\end{smallmatrix}\right)} = 4$ times.

shown in Fig. 4-2(a) and Fig. 4-2(b) respectively. In [2], Eulerian cycles were used to define decoupling protocols that avoided the discontinuous nature of bang-bang decoupling. More generally, one can define decoupling protocols based on balanced cycles (of which Eulerian decoupling is a special case), to which we soon turn our attention. Note, however, that this balanced-cycle decoupling protocol will not be the goal of this chapter. Indeed, such a protocol will not exploit the composite structure of the Hamiltonian. Later we will utilize the balanced-cycle decoupling on ℓ-qudit subsystems of a larger n qudit space to develop more efficient protocols; in the current section, however, we may regard ℓ as the size of the entire system.

In exploiting the ℓ-local nature of H, we will find that we are primarily interested in the group

$$\mathcal{G} = \mathbb{F}_q^\ell = \{(a_1, \ldots, a_\ell)^T : a_i \in \mathbb{F}_q\}$$

with some generating set $\mathscr{S}$ and the representation

$$\rho^{\otimes \ell} : \mathcal{G} \to \mathcal{U}(d^{\ell})$$

defined from our representation $\rho : \mathbb{F}_q \to \mathcal{U}(d)$. Specifically, if $\mathscr{g} = (a_1, \dots, a_\ell)^T \in \mathcal{G}$ and $\rho(a_i) = U_{a_i}$ then $\rho^{\otimes \ell}(\mathscr{g}) = U_{\mathscr{g}} = U_{a_1} \otimes \cdots \otimes U_{a_\ell}$. By our assumptions above, we can physically implement $U_{\mathscr{g}}$ by applying the control unitary $u_{\mathscr{g}}(\delta)$ (equivalently, the control Hamiltonian $h_{\mathscr{g}}(\delta)$) for time Δ. For example, in the case of qubits ($q = 4$), the group $\mathbb{F}_4$, whose elements we denote[3] as $\{0, 1, \alpha, \alpha+1\}$, is generated by the set $S_4 = \{1, \alpha\}$. We choose $\mathscr{S} = \{1^1, \alpha^1, \dots 1^\ell, \alpha^\ell\}$, which is a generating set of 2ℓ elements for the group $\mathcal{G} = \mathbb{F}_4^\ell$, where x^i here denotes the column $(0, \dots, 0, x, 0, \dots, 0)^T$ with $x \in \mathbb{F}_q$ in the ith position. In this case we assume $\rho(a_i) = U_{a_i}$ is a Pauli matrix, so $\rho^{\otimes \ell}(\mathscr{g})$ is a tensor product of Pauli matrices.

The purpose for the group theory used in this chapter resides in the following observation [3, Chapter 4]. We define the operator $\Pi_{\mathcal{G}}$ to act on matrices A as

$$\Pi_{\mathcal{G}}(A) = \frac{1}{|\mathcal{G}|} \sum_{\mathscr{g} \in \mathcal{G}} U_{\mathscr{g}}^\dagger A U_{\mathscr{g}}. \tag{4.1}$$

Note that for every matrix A, $\Pi_{\mathcal{G}}(A)$ commutes with all $U_{\mathscr{g}}$ ($\mathscr{g} \in \mathcal{G}$). Thus, by Schur's lemma, since ρ is irreducible[4], we have $\Pi_{\mathcal{G}}(A) = \frac{\mathrm{Tr}(A)}{D} \mathbb{1}$ (where D is the dimension of the Hilbert space). In particular then, if $\mathrm{Tr}(A) = 0$ then $\Pi_{\mathcal{G}}(A) = 0$.

Protocol 4.1 (Bounded-strength balanced-cycle decoupling). Let $\mathscr{L}$ be a balanced cycle on $\Gamma(\mathcal{G}, \mathscr{S})$ of length $N = |\mathcal{G}| \sum_{\mathscr{s}} \mu_{\mathscr{s}} = \lambda |\mathcal{G}|$, with group element representation $\mathscr{L}_{\mathcal{G}} = \left(\mathscr{g}_0, \dots, \mathscr{g}_{N-1}\right)$ and generator representation $\mathscr{L}_{\mathscr{S}} = (\mathscr{s}_1, \dots, \mathscr{s}_N)$. For $j = 1, \dots, N$, set $U_c(0) = U_{\mathscr{e}} = \mathbb{1}$ and

$$U_c\Big((j-1)\Delta + \delta\Big) = u_{\mathscr{s}_j}(\delta)\ U_c\Big((j-1)\Delta\Big), \qquad \delta \in [0, \Delta].$$

Note that because[5] $U_{\mathscr{s}_j} U_{\mathscr{g}_{j-1}} = U_{\mathscr{s}_j + \mathscr{g}_{j-1}} = U_{\mathscr{g}_j}$, this implies $U_c(j\Delta) = U_{\mathscr{g}_j}$ (for $j = 0, \dots, N$), i.e.

$$U_c\Big((j-1)\Delta + \delta\Big) = u_{\mathscr{s}_j}(\delta)\ U_{\mathscr{g}_{j-1}}, \qquad \delta \in [0, \Delta]. \tag{4.2}$$

The control cycle length is $T_c = N\Delta = |\mathcal{G}| \lambda \Delta$.

Theorem 4.1. The above balanced-cycle protocol performs bounded-strength decoupling.

Proof. Since $\mathscr{L}$ is a balanced cycle, $\mathscr{g} \bullet\!\xrightarrow{\mathscr{s}}$ occurs exactly $\mu_{\mathscr{s}}$ times for every $\mathscr{g}, \mathscr{s}$ pair. Thus $u_{\mathscr{s}}(\delta) U_{\mathscr{g}}$ appears exactly $\mu_{\mathscr{s}}$ times in the protocol for each $\mathscr{s}, \mathscr{g}$, and so we have, for any

[3] Here the reader may prefer to equivalently think of the Abelian group as $\{1, z, x, y = xz = zx\}$ with generating set $S_4 = \{z, x\}$. Then we can use $\mathscr{S} = \{z^{(1)}, x^{(1)}, \dots z^{(\ell)}, x^{(\ell)}\}$ and $\rho(x) = X$, $\rho(y) = Y$, and $\rho(z) = Z$. Be aware, however, that the group operation used throughout the chapter is denoted by + rather than by multiplication, since it is inherited from the finite field.

[4] Schur's lemma guarantees this directly when $\ell = 1$. But then it also applies for $\ell = 2$ since then for any matrix $A = \sum_i B_i \otimes C_i$, we have $\Pi_{\mathcal{G}}(A) = \frac{1}{|\mathcal{G}|} \sum_i \sum_{a_1, a_2 \in \mathbb{F}_q} U_{a_1}^\dagger B_i U_{a_1} \otimes U_{a_2}^\dagger C_i U_{a_2} \propto \sum_i \mathrm{Tr} B_i \mathrm{Tr} C_i = \mathrm{Tr} \sum_i B_i \otimes C_i = \mathrm{Tr} A$, and similarly for larger ℓ.

[5] up to phase, since ρ is a projective representation; since we will only ever conjugate by U_c, the overall phase is irrelevant and we shall simply ignore it.

traceless $d^\ell \times d^\ell$ Hamiltonian H,

$$\begin{aligned}\bar{H}^{(0)} &= \frac{1}{T_c}\int_{t=0}^{T_c} U_c(t)^\dagger H U_c(t)dt \\ &= \frac{1}{T_c}\sum_{\mathscr{g}} U_{\mathscr{g}}^\dagger \left[\sum_{\mathscr{s}} \mu_{\mathscr{s}} \int_{\delta=0}^{\Delta} u_{\mathscr{s}}(\delta)^\dagger H u_{\mathscr{s}}(\delta)d\delta\right] U_{\mathscr{g}} \\ &= \Pi_{\mathcal{G}}\Big(F_{\mathscr{S}}(H)\Big)\end{aligned}$$

where $\Pi_{\mathcal{G}}$ is defined in Eq. (4.1) and $F_{\mathscr{S}}$ is defined by

$$F_{\mathscr{S}}(H) = \sum_{\mathscr{s}} \frac{\mu_{\mathscr{s}}}{\lambda\Delta}\int_{\delta=0}^{\Delta} u_{\mathscr{s}}(\delta)^\dagger H u_{\mathscr{s}}(\delta)d\delta\,. \tag{4.3}$$

Recall that $\Pi_{\mathcal{G}}$ kills traceless matrices. Assuming that H is traceless, and observing that $F_{\mathscr{S}}$ is trace-preserving, we have that $\Pi_{\mathcal{G}}\Big(F_{\mathscr{S}}(H)\Big) = 0$. We conclude that $\bar{H}^{(0)} = 0$, i.e. the protocol succeeds at decoupling. □

Remark 4.1. For simplicity, we have assumed that ρ is irreducible. Then this protocol works for any traceless time-independent H, even if H is unknown. It is possible to define protocols in which ρ is not irreducible, in which case $\Pi_{\mathcal{G}}$ need not kill all traceless matrices. However, in such a case, one must take special care to ensure that $\Pi_{\mathcal{G}}$ still kills $F_{\mathscr{S}}(H)$ for the Hamiltonians of interest. See Chapter 3 for examples in a similar context, as well as Example 4.3 later in this chapter.

Remark 4.2. Although Protocol 4.1 performs bounded-strength decoupling, it would generally not be an efficient protocol were it applied to the entire system (i.e. if ℓ were the number of qudits of the entire system). Assuming that ρ is irreducible, the representation $\rho^{\otimes\ell} : \mathcal{G} \to \mathcal{U}(d^\ell)$ necessitates that $|\mathcal{G}|$, and therefore T_c, are exponential in ℓ. Indeed, suppose we have a representation from $\mathcal{G}$ to $\mathcal{U}(D)$ such that for any $D \times D$ matrix A, $\Pi_{\mathcal{G}}(A) = \frac{\mathrm{Tr}(A)}{D}\mathbb{1}_D$ as we used in Theorem 4.1. Consider sending the bipartite entangled state $|\psi\rangle = \frac{1}{\sqrt{D}}\sum_{j=1}^{D} |j\rangle \otimes |j\rangle$, or more precisely, $\Psi = |\psi\rangle\langle\psi|$, through the channel $\mathcal{I} \otimes \Pi_{\mathcal{G}}$ (where $\mathcal{I}$ is the identity channel on a D-dimensional space) obtaining

$$\sum_{\mathscr{g}\in\mathcal{G}} \frac{1}{|\mathcal{G}|}(\mathbb{1}_D \otimes U_{\mathscr{g}}^\dagger)\Psi(\mathbb{1}_D \otimes U_{\mathscr{g}}) = (\mathcal{I}\otimes\Pi_{\mathcal{G}})(\Psi) = \frac{1}{D^2}\mathbb{1}_D \otimes \mathbb{1}_D = \frac{1}{D^2}\mathbb{1}_{D^2}\ .$$

The matrix rank of the right-hand side is D^2. Using the fact that $\mathrm{rank}(A+B) \leqslant \mathrm{rank}(A) + \mathrm{rank}(B)$ and that for each $\mathscr{g}$, $\mathrm{rank}(\frac{1}{|\mathcal{G}|}(\mathbb{1}\otimes U_{\mathscr{g}}^\dagger)\Psi(\mathbb{1}\otimes U_{\mathscr{g}})) = \mathrm{rank}(\Psi) = 1$, the rank of the left-hand side is at most $|\mathcal{G}|$; thus, $|\mathcal{G}| \geqslant D^2$. Therefore, for the representation $\rho^{\otimes\ell} : \mathcal{G} \to \mathcal{U}(d^\ell)$ to succeed in the proof of Theorem 4.1, we require that $|\mathcal{G}| \geqslant d^{2\ell}$, which is exponential in ℓ. Incidentally, by considering the case of $\ell = 1$, we have justified why we could not have chosen q less than d^2 in our irreducible representation $\rho : \mathbb{F}_q \to \mathcal{U}(d)$.

Observe that the key to this protocol working is the fact that each $u_{\mathscr{s}}(\delta)U_{\mathscr{g}}$ shows up an equal number of times, independent of $\mathscr{g}$, i.e. $\forall\mathscr{s} \in \mathscr{S}\ \exists\mu_s > 0$ such that $\forall\mathscr{g} \in \mathcal{G}$, $\mathscr{g}\overset{\mathscr{s}}{\bullet\!\!\rightarrow}$ occurs $\mu_{\mathscr{s}}$ times (independent of $\mathscr{g}$). In an Eulerian cycle, $\mu_{\mathscr{s}} = 1$ for every $\mathscr{s}$, which is certainly sufficient. All else being equal, given the choice between Eulerian and other balanced cycles, we would choose Eulerian cycles as they will minimize N and therefore T_c. However, we will see that when considering the composite properties of a system (specifically that interactions are local), we will be able to exploit the notion of balanced cycles to come up with a much more efficient protocol.

4.4 Balanced-cycle Orthogonal Arrays

In Sec. 4.2 and Fig. 4-1, we indicated how we view our decoupling scheme as an array. For the protocol to be efficient, we shall ensure that this array corresponds to what we call a *balanced-cycle orthogonal array* (BOA). A BOA is a special type of orthogonal array (OA), which we first define. We refer the reader to [11] for a thorough introduction to OAs, particularly their relationship to linear codes (of which we shall later make use).

For notational consistency, we point out that throughout the remainder of this chapter we adopt the notation that $\mathcal{G}$ and $\mathscr{S}$ refer specifically to the group $\mathbb{F}_q^\ell$ and a generating set for $\mathbb{F}_q^\ell$, respectively. Elements of $\mathcal{G}$ will be denoted using script $\mathscr{g}$, elements of $\mathscr{S}$ will be denoted using script $\mathscr{s}$, and cycles on $\mathcal{G}$ will be denoted $\mathscr{L}$. When other groups (such as $\mathbb{F}_q$ or $\mathbb{F}_q^n$) are being considered, other notation (such as g, m, S, s and $\mathcal{L}$) will be used instead.

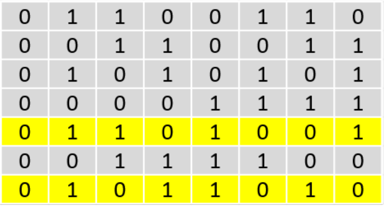
Definition 4.4 (Orthogonal array). An $OA_\lambda(N, n, q, \ell)$ *orthogonal array* on the alphabet $\mathbb{F}_q$ is an $n \times N$ array where each of the N columns is a vector from $\mathbb{F}_q^n$ such that every $\ell \times N$ subarray (obtained by only considering a selection of just ℓ of the n rows) contains each possible ℓ-tuple of elements of $\mathbb{F}_q$ (i.e. contains each $c \in \mathbb{F}_q^\ell$) precisely λ times as a column. The number ℓ is called the *strength* of the OA.

Remark 4.3. To relate these numbers to those appearing elsewhere in this chapter,

- N will correspond to number of steps in the decoupling protocol (i.e. the length of our balanced cycle),
- n will correspond to the number of d-dimensional qudits describing the system,
- $q = d^2$ (e.g. for qubits, $d = 2$ and $q = 4$),
- ℓ is the locality of the Hamiltonian (e.g. for pairwise interactions, $\ell = 2$), and
- $\lambda = N/q^\ell$ will be the same λ as in our discussion of balanced cycles, $\lambda = \sum_{\mathscr{s}} \mu_{\mathscr{s}}$.

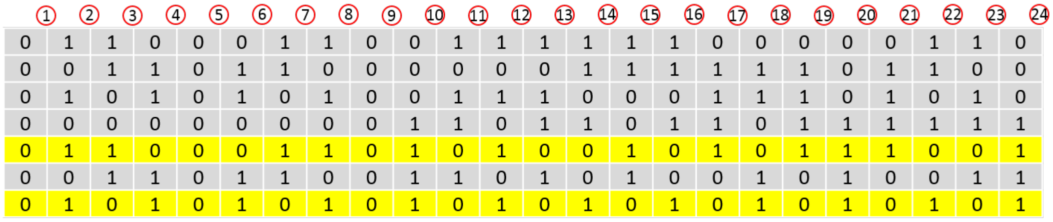
Remark 4.4. Note that the order of the columns in the OA is irrelevant to whether the array is an OA. Moreover, if $A = [\vec{a}_i]$ is an $OA_\lambda(N, n, q, \ell)$ with columns $\vec{a}_1, \ldots, \vec{a}_N$ then the matrix A', whose columns consist of precisely r copies of each $\vec{a}_i$ (in any order), is an $OA_{r\lambda}(rN, n, q, \ell)$. Note, however, that while the order of the columns does not affect the OA property of the array, when defining balanced-cycle orthogonal arrays (which we do next), we will be highly concerned with the order of the columns in the array.

An example of an $OA_2(8, 7, 2, 2)$ is shown in Fig. 4-3. Orthogonal arrays have been used to construct bang-bang decoupling schemes (see [4–6] and [3, Chapter 15]). In order to construct a bounded-strength scheme, we introduce the notion of a balanced-cycle orthogonal array, defined as follows.



0	1	1	0	0	1	1	0
0	0	1	1	0	0	1	1
0	1	0	1	0	1	0	1
0	0	0	0	1	1	1	1
0	1	1	0	1	0	0	1
0	0	1	1	1	1	0	0
0	1	0	1	1	0	1	0

Figure 4-3: Example of an $OA_2(8, 7, 2, 2)$, i.e. an $OA_\lambda(N, n, q, \ell)$ with $N = 8$ columns and $n = 7$ rows on the finite field $\mathbb{F}_q = \mathbb{Z}_2$ of order $q = 2$. Any subarray defined by any $\ell = 2$ rows contains each 2-tuple precisely $\lambda = 2$ times. For example, rows 5 and 7 (highlighted) form a 2×8 subarray in which $\binom{0}{0}$, $\binom{0}{1}$, $\binom{1}{0}$, and $\binom{1}{1}$ each occur precisely twice. Note that typically in this chapter, $q = d^2$ (for example, for qubits $q = 4$), but for simplicity, the example in this figure uses $q = 2$.



①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩	⑪	⑫	⑬	⑭	⑮	⑯	⑰	⑱	⑲	⑳	㉑	㉒	㉓	㉔
0	1	1	0	0	0	1	1	0	0	1	1	1	1	1	1	0	0	0	0	0	1	1	0
0	0	1	1	0	1	1	0	0	0	0	0	0	1	1	1	1	1	1	0	1	1	0	0
0	1	0	1	0	1	0	1	0	0	1	1	1	0	0	0	1	1	1	0	1	0	1	0
0	0	0	0	0	0	0	0	0	1	1	0	1	1	0	1	1	0	1	1	1	1	1	1
0	1	1	0	0	0	1	1	0	1	0	1	0	0	1	0	1	0	1	1	1	0	0	1
0	0	1	1	0	1	1	0	0	1	1	0	1	0	1	0	0	1	0	1	0	0	1	1
0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Figure 4-4: Example of a $BOA(24, 7, 2, 2)$, i.e. a $BOA(N, n, q, \ell)$ with $N = 24$ columns and $n = 7$ rows on the finite field $\mathbb{F}_q = \mathbb{Z}_2$ of order $q = 2$. Any subarray defined by any $\ell = 2$ rows defines a balanced cycle on the Cayley graph $\Gamma(\mathcal{G}, \mathscr{S})$ of $\mathcal{G} = \mathbb{F}_q^\ell = \mathbb{Z}_2^2$ with respect to some generating set $\mathscr{S}$ (which may depend on the subarray). For example, rows 5 and 7 (highlighted) form a 2×24 subarray that defines the balanced cycle shown in Fig. 4-2(b). The circled integers (red) correspond to the steps taken by the balanced cycle as shown in that figure. Note that typically in this chapter, $q = d^2$ (for example, for qubits $q = 4$), but for simplicity, the BOA example shown here uses $q = 2$. The method by which this BOA was constructed is detailed in Example 4.1 of Sec. 4.8.

Definition 4.5 (Balanced-cycle orthogonal array). A $BOA(N, n, q, \ell)$ *balanced-cycle orthogonal array* on the alphabet $\mathbb{F}_q$ is an $n \times N$ array, A, where each of the N columns is a vector from $\mathbb{F}_q^n$ such that every $\ell \times N$ subarray (obtained by only considering a selection of just ℓ of the n rows) defines a balanced cycle on the Cayley graph of $\mathcal{G} = \mathbb{F}_q^\ell$ with respect to some generating set for $\mathcal{G}$ (which may depend on the subarray). Specifically, if the entries of A are denoted a_{ij} (with $1 \leqslant i \leqslant n$ and $0 \leqslant j \leqslant N - 1$), then for every choice of ℓ distinct integers $i_1, \ldots, i_\ell \in \{1, \ldots, n\}$, there is a generating set $\mathscr{S}$ for $\mathcal{G}$ (which may, in general, depend on $i_1, \ldots, i_\ell$) such that if $\mathfrak{g}_j = (a_{i_1 j}, \ldots, a_{i_\ell j})^T$ denotes the jth column of A restricted to rows $i_1, \ldots, i_\ell$, then $\mathscr{L}_\mathcal{G} = \left(\mathfrak{g}_0, \ldots, \mathfrak{g}_{N-1}\right)$ defines a balanced cycle on $\Gamma(\mathcal{G}, \mathscr{S})$.

An example of a BOA is shown in Fig. 4-4. We defer the proof that BOAs exist to Sec. 4.5. The remainder of the current section defines a decoupling protocol based on BOAs and proves that it works to decouple ℓ-local Hamiltonians in n qudit systems ($\ell \leqslant n$). Working with ℓ qudits (rather than n qudits), along with the promise that H is ℓ-local, will enable us to give an efficient protocol.

Protocol 4.2 (Efficient, bounded-strength balanced-cycle decoupling based on BOAs). Let $A = [\vec{a}_j]_{j=0,\dots,N-1}$ be a $BOA(N, n, q, \ell)$ whose columns are denoted by the vectors $\vec{a}_j = (a_{1j}, \dots, a_{nj})^T$, where $a_{ij} \in \mathbb{F}_q$ is the (i, j) entry of A. For $j = 1, \dots, N$, let $\vec{b}_j = \vec{a}_j - \vec{a}_{j-1}$ be the transitions between the columns, treating $\vec{a}_N = \vec{a}_0 = 0$.

For $j = 1, \dots, N$, set $U_c(0) = \mathbb{1}$ and

$$U_c\Big((j-1)\Delta + \delta\Big) = u_{\vec{b}_j}(\delta) U_c\Big((j-1)\Delta\Big), \qquad \delta \in [0, \Delta];$$

note that this implies that $U_c(j\Delta) = U_{\vec{a}_j}$ (for $j = 0, \dots, N$). The control cycle length is thus $T_c = N\Delta$.

Theorem 4.2. The above protocol performs bounded-strength decoupling.

Proof. H is an ℓ-local Hamiltonian, $H = \sum_k H_k$ with each H_k acting non-trivially on at most ℓ qudits. Consider a term H_k, which acts non-trivially only on qudits denoted $i_1, \dots, i_\ell$ and write $H_k = h_k \otimes \mathbb{1}_{n-\ell}$, where h_k is understood to be a $d^\ell \times d^\ell$ matrix acting only on these ℓ qudits and $\mathbb{1}_{n-\ell}$ is the identity matrix on the other $n - \ell$ qudits. By definition of a BOA, the $\ell \times N$ subarray of A restricted to rows $i_1, \dots, i_\ell$ defines a balanced cycle $\mathscr{L}$ on $\Gamma(\mathcal{G}, \mathscr{S})$ where $\mathscr{S}$ is some generating set of $\mathcal{G} = \mathbb{F}_q^\ell$. The idea of the proof is to observe that the protocol involving the columns $\vec{a}_j$ for decoupling H_k is equivalent to a protocol involving the subarray's columns for decoupling h_k; since the subarray defines a balanced cycle, we can then invoke Protocol 4.1 to successfully decouple h_k and therefore H_k.

Let $\mathscr{g}_j = (a_{i_1 j}, \dots, a_{i_\ell j})^T$ denote the jth column of A restricted to rows $i_1, \dots, i_\ell$ and let $\mathscr{s}_j = \mathscr{g}_j - \mathscr{g}_{j-1} = (b_{i_1 j}, \dots, b_{i_\ell j})^T$, where b_{ij} is the ith entry of $\vec{b}_j$. Then the cycle $\mathscr{L}$ is represented as $\mathscr{L}_{\mathcal{G}} = \Big(\mathscr{g}_0, \dots, \mathscr{g}_{N-1}\Big)$ and $\mathscr{L}_{\mathscr{S}} = (\mathscr{s}_1, \dots, \mathscr{s}_N)$.

As in the proof of Theorem 4.1, we are interested in $U_c(t)^\dagger H_k U_c(t)$. The control unitary at time $t = (j-1)\Delta + \delta$ is

$$\begin{aligned} U_c\Big((j-1)\Delta + \delta\Big) &= u_{\vec{b}_j}(\delta)\, U_c\Big((j-1)\Delta\Big) \\ &= u_{\vec{b}_j}(\delta)\, U_{\vec{a}_{j-1}} \\ &= \big(u_{b_{1j}}(\delta) \otimes \cdots \otimes u_{b_{nj}}(\delta)\big) \Big(U_{a_{1(j-1)}} \otimes \cdots \otimes U_{a_{n(j-1)}}\Big) . \end{aligned}$$

Thus, when conjugating $H_k = h_k \otimes \mathbb{1}_{n-\ell}$ by $U_c\big((j-1)\Delta + \delta\big)$, all of the unitaries not acting on the ℓ-qudit subspace of h_k will commute through H_k and cancel, leaving only those corresponding to the ℓ-qudit subspace, i.e. those corresponding to the labels $\mathscr{s}_j$ and $\mathscr{g}_j$. Explicitly,

$$\begin{aligned} & U_c\Big((j-1)\Delta + \delta\Big)^\dagger\; H_k\; U_c\Big((j-1)\Delta + \delta\Big) \\ &= \Bigg[\Big(U^\dagger_{a_{i_1(j-1)}} \otimes \cdots \otimes U^\dagger_{a_{i_\ell(j-1)}}\Big) \Big(u_{b_{i_1 j}}(\delta)^\dagger \otimes \cdots \otimes u_{b_{i_\ell j}}(\delta)^\dagger\Big)\; h_k \\ &\qquad \Big(u_{b_{i_1 j}}(\delta) \otimes \cdots \otimes u_{b_{i_\ell j}}(\delta)\Big) \Big(U_{a_{i_1(j-1)}} \otimes \cdots \otimes U_{a_{i_\ell(j-1)}}\Big) \Bigg] \otimes \mathbb{1}_{n-\ell} \\ &= U^\dagger_{\mathscr{g}_{j-1}}\, u_{\mathscr{s}_j}(\delta)^\dagger\; h_k\; u_{\mathscr{s}_j}(\delta) U_{\mathscr{g}_{j-1}} \otimes \mathbb{1}_{n-\ell} . \end{aligned}$$

Thus, the protocol of applying U_c to H_k is effectively the same as applying a protocol $u_{\mathscr{s}_j}(\delta)U_{\mathscr{g}_{j-1}}$ to h_k, following the balanced cycle $\mathscr{L}$. Since this is precisely the scheme defined in Protocol 4.1 applied to h_k (see Eq. (4.2)), we conclude from Theorem 4.1 that it decouples h_k. Consequently, $\bar{H}_k^{(0)} = \bar{h}_k^{(0)} \otimes \mathbb{1}_{n-\ell} = 0$. This occurs for every term H_k in $H = \sum_k H_k$, whence H itself is decoupled: $\bar{H}^{(0)} = \sum_k \bar{H}_k^{(0)} = 0$.

□

Remark 4.5. Once we have a BOA scheme that can decouple a system of n qudits, the same scheme can be used (with the same BOA and therefore same length N) for a system of $n' < n$ qudits. This can be accomplished by simply ignoring $n - n'$ of the qudits, i.e. by having U_c act as $\mathbb{1}$ on these $n - n'$ extra qudits (rather than as dictated by the original protocol). The proof of Theorem 4.2 remains unaffected because H_k acts trivially on these extra qudits, i.e. they are not acted upon by h_k.

Theorem 4.2 showed that decoupling protocols based on BOAs work, with control cycle length proportional to the BOA parameter N. We next show that BOAs can indeed be constructed and, moreover, that the construction gives rise to an *efficient* decoupling protocol, in the sense that N does not increase exponentially with n.

4.5 Construction of balanced-cycle orthogonal arrays

The existence of balanced-cycle orthogonal arrays follows naturally from constructions of orthogonal arrays generated using classical linear codes, which we shall define shortly. We first give a brief outline of our BOA construction. This construction is via the generator matrix G of a linear code, which is a linear mapping from $\mathbb{F}_q^k$ to $\mathbb{F}_q^n$ for some $k \leqslant n$. If we enumerate all elements of $\mathbb{F}_q^k$ in an arbitrary order and consider their image under G, this will form an OA of strength ℓ (for an appropriately chosen k). To obtain a BOA, we do this enumeration according to the prescription of an Eulerian cycle on $\mathbb{F}_q^k$. In doing so, we can guarantee that we always obtain a balanced cycle when we consider any submatrix of ℓ rows, ultimately ensuring that any ℓ-local Hamiltonian term on those corresponding qudits will be decoupled. We now prove this, starting with a definition of a classical linear code.

Definition 4.6 (Classical linear code). A *classical linear* $[n, k]_q$ *code*, C, is a k-dimensional subspace of the vector space $\mathbb{F}_q^n$. For any vector $x = (x_1, \dots, x_n)^T \in \mathbb{F}_q^n$, define $\mathrm{wt}(x) = |\{i \in \{1, \dots, n\} : x_i \neq 0\}|$. The *distance* of a linear code C is defined to be $\min\{\mathrm{wt}(c) : c \in C, c \neq o\}$, where o denotes the zero vector. An $[n, k]_q$ linear code can be described by a *generator matrix* G of size $n \times k$ with entries from $\mathbb{F}_q$. G maps the vectors $m \in \mathbb{F}_q^k$ onto the elements (*codewords*) of C so that $C = G[\mathbb{F}_q^k] = \{Gm \in \mathbb{F}_q^n : m \in \mathbb{F}_q^k\}$.

The *dual code* $C^\perp$ of C is defined by $C^\perp = \{y \in \mathbb{F}_q^n : x \cdot y = 0 \ \forall x \in C\}$ with the dot product $x \cdot y = \sum_{i=1}^n x_i y_i$. The dual code is also a classical linear code, namely an $[n, n-k]_q$ code with some distance $\delta^\perp$ that we will refer to as the *dual distance of* C. Orthogonal arrays can be constructed from linear codes, as the following theorem [11, Theorem 4.6] establishes.

Theorem 4.3 (OAs from linear codes). Let C be a linear $[n, k]_q$ code with dual distance $\delta^\perp$. The $n \times q^k$ matrix $[Gm]_{m \in \mathbb{F}_q^k}$, whose columns are the q^k vectors $Gm \in \mathbb{F}_q^n$ ($\forall m \in \mathbb{F}_q^k$), is an $OA(q^k, n, q, \ell)$ with strength $\ell = \delta^\perp - 1$.

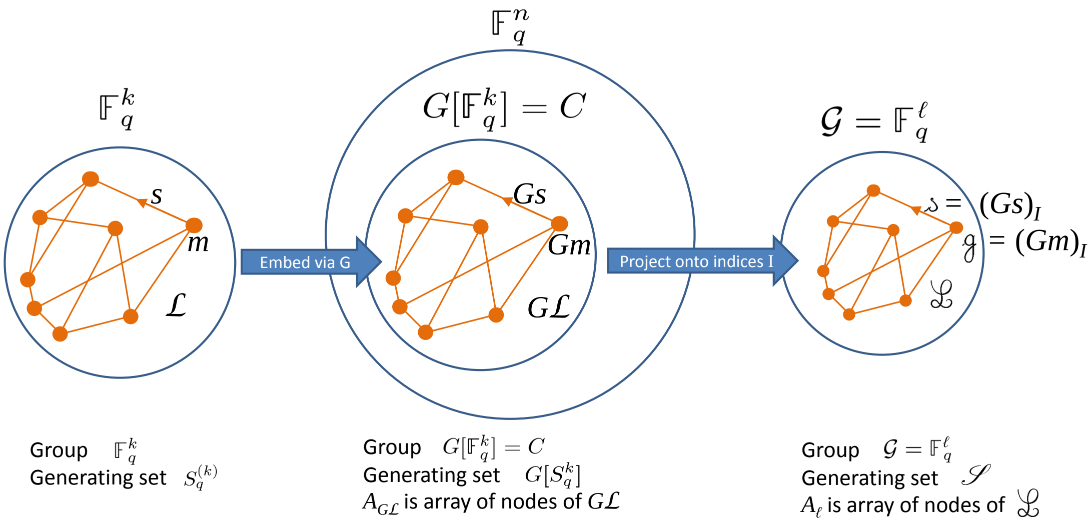

$\mathbb{F}_q^k$
s
m
$\mathcal{L}$
Embed via G
$\mathbb{F}_q^n$
$G[\mathbb{F}_q^k] = C$
Gs
Gm
$G\mathcal{L}$
Project onto indices I
$\mathcal{G} = \mathbb{F}_q^\ell$
$\mathscr{s} = (Gs)_I$
$\mathscr{g} = (Gm)_I$
$\mathscr{L}$
Group $\mathbb{F}_q^k$
Generating set $S_q^{(k)}$
Group $G[\mathbb{F}_q^k] = C$
Generating set $G[S_q^k]$
$A_{G\mathcal{L}}$ is array of nodes of $G\mathcal{L}$
Group $\mathcal{G} = \mathbb{F}_q^\ell$
Generating set $\mathscr{S}$
A_ℓ is array of nodes of $\mathscr{L}$

Figure 4-5: Names and relationships between various groups and cycles in the BOA construction. The graph is a schematic of a Cayley graph. As explained in the text, the BOA, $A_{G\mathcal{L}}$, is the array form of $G\mathcal{L}$, which is the result of mapping an Eulerian cycle $\mathcal{L}$ under the linear code generating matrix G. As a BOA, $A_{G\mathcal{L}}$ has the property that if one considers a subarray of ℓ rows, the result describes a balanced cycle. Specifically, let $I \subset \{1, 2, \ldots, n\}$ be a subset of ℓ indices. $(Gm)_I$ denotes the ℓ-tuple of elements of Gm (itself an n-tuple) corresponding to the indices I. The cycle $\mathscr{L}$, composed of nodes $(Gm)_I$ (in the same order in which $\mathcal{L}$ was composed of m), is shown to be a balanced cycle.

Let C be an $[n, k]_q$ with dual distance $\delta^\perp = \ell + 1$ and generating matrix G. Let $\mathcal{L}$ be an Eulerian cycle on the Cayley graph $\Gamma(\mathbb{F}_q^k, S_q^{(k)})$, where $S_q^{(k)}$ is a generating set for $\mathbb{F}_q^k$; thus, we can write $\mathcal{L}_{\mathbb{F}_q^k} = (m_0, \ldots, m_{N-1})$ with transitions $\mathcal{L}_{S_q^{(k)}} = (s_1, \ldots, s_N)$ and $N = q^k\,|S_q^{(k)}|$. Because d is a prime power, say $d = p^e$ for some prime p, the minimal generating set is of size $|S_q^{(k)}| = 2ke$. We are interested in the image of the cycle in the codespace; thus, consider the Eulerian cycle, denoted $G\mathcal{L}$, on $\Gamma(G[\mathbb{F}_q^k], G[S_q^{(k)}])$, where $G[\mathbb{F}_q^k] = C \subset \mathbb{F}_q^n$ is the image of $\mathbb{F}_q^k$ under G, i.e. is the codespace. In other words, $G\mathcal{L}_{\mathbb{F}_q^k} = (Gm_0, \ldots, Gm_{N-1})$ and $G\mathcal{L}_{S_q^{(k)}} = (Gs_1, \ldots, Gs_N)$.

To avoid possible confusion, we emphasize here that although we will use $G\mathcal{L}$ to construct a BOA, neither $\mathcal{L}$ nor $G\mathcal{L}$ will serve as the balanced cycle to which Theorem 4.1 applies (which is why we have used the notation $\mathcal{L}$ rather than $\mathscr{L}$). Rather, for our efficient decoupling scheme, we construct an array $A_{G\mathcal{L}}$, dictated by $G\mathcal{L}$, and prove that the result is a BOA by showing that if we consider any subarray of ℓ rows, it gives rise to some balanced cycle $\mathscr{L}$ on $\mathcal{G} = \mathbb{F}_q^\ell$. The notation and relationships of the various groups and cycles used in this chapter is sketched in Fig. 4-5.

We turn $G\mathcal{L}$ into an array $A_{G\mathcal{L}}$ in the obvious way as follows. Each element Gm_j of $G\mathcal{L}_{\mathbb{F}_q^k}$ is a column vector in $\mathbb{F}_q^n$. Therefore we may associate to $G\mathcal{L}$ the $n \times N$ matrix $A_{G\mathcal{L}} = [Gm]_{m \in \mathcal{L}_{\mathbb{F}_q^k}}$ with elements $a_{ij} = (Gm_j)_i$, so that the jth column of $A_{G\mathcal{L}}$ is the vector Gm_j, and the columns are arranged in the order of the Eulerian cycle $G\mathcal{L}$. Note that since we assumed that Eulerian cycles always start with the (additive) identity element, i.e. the zero vector $o \in \mathbb{F}_q^k$, and since G maps the zero vector to the zero vector ($Go = o \in \mathbb{F}_q^n$), the first column of $A_{G\mathcal{L}}$ is the zero vector of $\mathbb{F}_q^n$.

Lemma 4.4. *$A_{G\mathcal{L}}$ is an $OA_{N/q^\ell}(N, n, q, \ell)$ with $N = q^k\,|S_q^{(k)}|$.*

Proof. By Theorem 4.3, an array whose q^k columns are the vectors of the codespace is an OA. The columns of $A_{G\mathcal{L}}$ are precisely $|S_q^{(k)}|$ copies of each vector in the codespace, and therefore (using Remark 4.4), $A_{G\mathcal{L}}$ is an OA. □

Let $s \in S_q^{(k)}$. $Gs \in \mathbb{F}_q^n$, so $Gs = \Big((Gs)_1, \ldots, (Gs)_n\Big)^T$, where we use the notation $(Gs)_i \in \mathbb{F}_q$ to denote the ith component of the column vector Gs. Fix ℓ distinct numbers $i_1, \ldots, i_\ell \in \{1, \ldots, n\}$ and write $I = \{i_1, \ldots, i_\ell\}$. Let $(Gs)_I$ denote the ℓ-tuple $\Big((Gs)_{i_1}, \ldots, (Gs)_{i_\ell}\Big)^T$. Let $\mathscr{S} = \Big\{(Gs)_I : s \in S_q^{(k)}\Big\}$.

Lemma 4.5. *$\mathscr{S}$ is a generating set for $\mathcal{G} = \mathbb{F}_q^\ell$.*

Proof. Let $\mathpzc{g} \in \mathcal{G}$. By definition, since $A_{G\mathcal{L}} = [Gm]_{m \in \mathcal{L}_{\mathbb{F}_q^k}}$ is an OA of strength ℓ, the $\ell \times N$ subarray obtained by only considering rows $i_1, \ldots, i_\ell$ contains each possible ℓ-tuple of elements of $\mathbb{F}_q$, and therefore contains $\mathpzc{g}$. Thus, $\exists\ Gm$ such that $(Gm)_I = \mathpzc{g}$. Since $S_q^{(k)}$ is a generating set for $\mathbb{F}_q^k$, $\exists\ u_1, \ldots, u_r \in S_q^{(k)}$ such that $m = u_1 + \cdots + u_r$, and therefore $Gm = Gu_1 + \cdots + Gu_r$. But then $(Gu_j)_I \in \mathscr{S}$ for every $j = 1, \ldots, r$ and $\mathpzc{g} = (Gm)_I = (Gu_1)_I + \cdots + (Gu_r)_I$, whence $\mathscr{S}$ generates $\mathcal{G}$. □

Recall $A_{G\mathcal{L}} = [Gm]_{m \in \mathcal{L}_{\mathbb{F}_q^k}}$ and consider the $\ell \times N$ submatrix $A_\ell = [\mathpzc{g}_j]$ of $A_{G\mathcal{L}}$, whose jth column is $\mathpzc{g}_j = (Gm_j)_I \in \mathcal{G}$. Define the ordered list $\mathpzc{L}_{\mathcal{G}} = \Big(\mathpzc{g}_0, \ldots, \mathpzc{g}_{N-1}\Big)$. Although $\mathpzc{L}$ depends on I, we suppress mention of this for notational simplicity.

Lemma 4.6. *$\mathpzc{L}$ is a balanced cycle on $\Gamma(\mathcal{G}, \mathscr{S})$.*

Proof. Each $\mathpzc{g} \in \mathcal{G}$ is present in $\mathpzc{L}_{\mathcal{G}}$ an equal number of times because $A_{G\mathcal{L}}$ is an OA of strength ℓ. The transitions in this cycle are $\mathpzc{s}_j = \mathpzc{g}_j - \mathpzc{g}_{j-1} = (Gm_j)_I - (Gm_{j-1})_I = (Gs_j)_I \in \mathscr{S}$, so the transition representation $\mathpzc{L}_{\mathscr{S}} = (\mathpzc{s}_1, \ldots, \mathpzc{s}_N)$ consists of generators from $\mathscr{S}$; $\mathpzc{L}$ is therefore a cycle on the Cayley graph $\Gamma(\mathcal{G}, \mathscr{S})$. Moreover, because $\mathcal{L}$ is an Eulerian cycle and $A_{G\mathcal{L}}$ is an OA, $\mathpzc{L}$ is a balanced cycle (although not an Eulerian cycle): informally, each $Gm{\bullet}\!\xrightarrow{Gs}$ occurs in $G\mathcal{L}$ an equal (non-zero) number of times (namely once, independent of Gm) for each Gs, so each $\mathpzc{g}{\bullet}\!\xrightarrow{\mathpzc{s}}$ occurs in $\mathpzc{L}$ an equal (non-zero) number of times (independent of $\mathpzc{g} = (Gm)_I$, since $A_{G\mathcal{L}}$ is an OA) for each $\mathpzc{s} = (Gs)_I$.

Explicitly, consider any $\mathpzc{g} \in \mathcal{G}$ and $\mathpzc{s} \in \mathscr{S}$. Let $M_{\mathpzc{g}} = \{m \in \mathbb{F}_q^k : (Gm)_I = \mathpzc{g}\}$. $\mathcal{L}_{\mathbb{F}_q^k}$ is an Eulerian cycle so each element in $\mathbb{F}_q^k$ shows up precisely $|S_q^{(k)}|$ times in $\mathcal{L}_{\mathbb{F}_q^k}$. In particular, therefore, each $m \in M_{\mathpzc{g}}$ appears precisely $|S_q^{(k)}|$ times in $\mathcal{L}_{\mathbb{F}_q^k}$, and consequently, $\mathpzc{g}$ shows up in $\mathpzc{L}$ precisely $|M_{\mathpzc{g}}||S_q^{(k)}|$ times. But $A_{G\mathcal{L}}$ is an OA of strength ℓ, so $|M_{\mathpzc{g}}||S_q^{(k)}|$ must then be independent of $\mathpzc{g}$, and therefore $|M_{\mathpzc{g}}|$ is also independent of $\mathpzc{g}$. Since $\mathpzc{s} \in \mathscr{S}$, let $S_{\mathpzc{s}} = \{s \in S_q^{(k)} : (Gs)_I = \mathpzc{s}\}$. This set is non-empty by definition of $\mathscr{S}$. In general, $|S_{\mathpzc{s}}|$

may depend on $\mathscr{s}$. Now, $\forall m \in M_{\mathscr{g}}$ and $\forall s \in S_{\mathscr{s}}$, the Eulerian property of $\mathcal{L}$ guarantees that $Gm\bullet\xrightarrow{Gs}$ occurs precisely once in $G\mathcal{L}$. Therefore, $\mathscr{g}\bullet\xrightarrow{\mathscr{s}}$ occurs in $\mathscr{L}$ precisely $|S_{\mathscr{s}}||M_{\mathscr{g}}| \geqslant 1$ times, which is independent of $\mathscr{g}$. Thus $\mathscr{L}$ is a balanced cycle. □

Together, the above lemmas prove the existence of BOAs and how to construct them from classical linear codes.

Theorem 4.7. Let C, $\mathcal{L}$, and $A_{G\mathcal{L}}$ be as above, i.e. C is an $[n,k]_q$ code with dual distance $\delta^{\perp} = \ell + 1$ and generating matrix G, $\mathcal{L}$ is an Eulerian cycle on the Cayley graph $\Gamma(\mathbb{F}_q^k, S_q^{(k)})$, written $\mathcal{L}_{\mathbb{F}_q^k} = (m_0, \ldots, m_{N-1})$ and $\mathcal{L}_{S_q^{(k)}} = (s_1, \ldots, s_N)$, and $A_{G\mathcal{L}} = [Gm]_{m \in \mathcal{L}_{\mathbb{F}_q^k}}$ is an OA whose columns are the vectors Gm_j. Then $A_{G\mathcal{L}}$ is a $BOA(N, n, q, \ell)$ with $N = q^k|S_q^{(k)}|$.

Proof. For every choice of ℓ distinct integers $I = \{i_1, \ldots, i_\ell\} \subset \{1, \ldots, n\}$, the set $\mathscr{S} = \left\{(Gs)_I : s \in S_q^{(k)}\right\}$ is a generating set for $\mathcal{G}$ (by Lemma 4.5) such that if $\mathscr{g}_j$ denotes the jth column of $A_{G\mathcal{L}}$ restricted to rows $i_1, \ldots, i_\ell$, then (by Lemma 4.6) $\mathscr{L}_{\mathcal{G}} = \left(\mathscr{g}_0, \ldots, \mathscr{g}_{N-1}\right)$ defines a balanced cycle on $\Gamma(\mathcal{G}, \mathscr{S})$. □

For n interacting qudits of dimension $d = p^e$ (for some prime p and positive integer e) that obey an ℓ-local Hamiltonian, this construction therefore allows

$$N = q^k|S_q^{(k)}| = q^k 2ke \tag{4.4}$$

where k is the dimension of the code used and $q = d^2$. Observe that the BOA decoupling protocol (Protocol 4.2) for this BOA construction has a control cycle length of

$$T_c = N\Delta = d^{2k} 2ke\Delta$$

where Δ is some fixed length of time. For example, in the qubit ($d = 2$) case discussed above, $|S_4^{(k)}| = 2k$, whence $T_c = N\Delta$ with $N = (2k)4^k$. To maximize efficiency for a given n and ℓ, one should select a code that minimizes k (equivalently, select a dual code that maximizes $k^{\perp} = n - k$).

There exist many good families of classical linear codes. For instance, for 2-local interactions, we can (as was done in [12] for OAs) rely on $[n,k]_q$ Hamming codes with dual distance 3 such that $k = \log_q\big((q-1)n + 1\big)$; our scheme then has N scaling like $n\log(n)$. This protocol is therefore much more efficient than a naive protocol of applying balanced-cycle decoupling (including Eulerian decoupling) without exploiting the ℓ-local structure of the Hamiltonian, which would have a control cycle length that scales exponentially with n. It is also more efficient than the method of [7], which required $N = d^{4k}$, i.e. whose scaling for this case ($\ell = 2$, using Hamming codes) is quadratic in n. Next, we address codes for BOA construction with values of ℓ greater than 2.

4.6 BOA decoupling schemes from BCH codes

In this section we show how to construct schemes that achieve decoupling for ℓ-local Hamiltonians on $\mathcal{H} \cong (\mathbb{C}^d)^{\otimes n}$ for arbitrary ℓ, n, and prime power d. Besides the machinery of balanced-cycle orthogonal arrays (BOAs) that was introduced in the previous sections, our construction relies on BCH codes as a particular vehicle to construct good BOAs. The choice of BCH codes results from the fact that they are among the best known codes for the particular situation where the distance is a fixed, small number and the goal is to maximize the overall code dimension. Using the dual of a BCH code when constructing the corresponding orthogonal arrays, we obtain schemes with a designed OA strength (i.e. locality ℓ) while having a small N in the corresponding decoupling protocol. We begin by briefly recalling some basics about BCH codes; for more details on finite fields and BCH codes see, for example, the textbooks [13–15].

Definition 4.7 (BCH code). Let α be a primitive n-th root of unity in the finite field $\mathbb{F}_{q^m}$, where q is a prime power, $n \geqslant 2$, and $m \geqslant 1$. A *BCH code* over $\mathbb{F}_q$ of length n and designed distance D, where $2 \leqslant D \leqslant n$, is a cyclic polynomial code defined by the zeros

$$\alpha^b, \alpha^{b+1}, \ldots, \alpha^{b+D-2},$$

where $b \geqslant 1$ is a positive integer.

The generator polynomial $g(x)$ of the cyclic code introduced in Definition 4.7 is given by $g(x) = \mathrm{lcm}(M_b(x), M_{b+1}(x), \ldots, M_{b+D-2}(x))$, where $M_i(x)$ denotes the minimal polynomial of α^i over $\mathbb{F}_q$. Note that even though the zeros of the code lie in an extension field $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$, the BCH code itself is a cyclic code over the ground field $\mathbb{F}_q$. Furthermore, it is known that a BCH code defined this way has a distance δ that is at least D, which is why D is sometimes called the "designed distance." Note that the actual distance δ of the code might exceed D. The possible lengths of BCH codes are quite restricted, as any admissible length n must be a divisor of the order of the multiplicative group of $\mathbb{F}_{q^m}$, i.e. must be a divisor of $q^m - 1$. In the following we restrict ourselves to the case where $n = q^m - 1$, which is called the case of *primitive BCH codes.* Furthermore we only consider the case where $b = 1$, which is called the case of *narrow-sense BCH codes.* We denote these codes by $\mathrm{BCH}(\mathbb{F}_{q^m}/\mathbb{F}_q, D)$, and we note that they always exist.

For any linear error-correcting code $C = [n, k, \delta]_q$ of length n, dimension k, and distance δ, an *extension* $C' = [n+1, k, \delta' \geqslant \delta]_q$ can be defined by adding another coordinate and an overall parity check. At the level of parity check matrices, this corresponds to appending the parity check matrix M of C with an all-zeros column $\mathbf{0}$ and an all-ones row $\mathbf{1}^T$ so that C' has the new parity check matrix $\begin{bmatrix} \mathbf{1}^T & 1 \\ M & \mathbf{0} \end{bmatrix}$. For binary codes, the distance of the extension is easy to characterize: if $\delta \equiv 0 \mod 2$ then $\delta' = \delta$ and if $\delta \equiv 1 \mod 2$ then $\delta' = \delta + 1$. In general over larger alphabets, however, it is possible that the distance increases even when δ is even. When applying an extension to the BCH codes introduced above, we use the notation $\mathrm{BCH}^{\mathrm{ext}}(\mathbb{F}_{q^m}/\mathbb{F}_q, D)$. We make use of the following theorem about such codes.

Theorem 4.8. Let $\mathbb{F}_q$ be a finite field and let $\mathrm{BCH}^{\mathrm{ext}}(\mathbb{F}_{q^m}/\mathbb{F}_q, D) = [n, k, \delta]_q$ be the extension of the primitive narrow-sense BCH code with designed distance D constructed in Definition 4.7, so $n = q^m$ and $\delta \geqslant D$. Assume that $D \leqslant q^{\lceil m/2 \rceil} + 2$. Then the dimension k of the code satisfies $k \geqslant n - m\lceil \frac{q-1}{q}(D-2) \rceil - 1 \geqslant n - m(D-2) - 1$.

See [16] for a proof of Theorem 4.8 that leverages the fact that the extended primitive narrow-sense BCH codes are subfield subcodes of the Reed-Solomon codes. See also [15, Problem 8.12] and [17]. By combining Theorem 4.8 with the construction of Theorem 4.7 we now obtain the following result regarding bounded-strength decoupling for ℓ-local Hamiltonians.

Theorem 4.9. For any $\ell \geqslant 2$, $n \geqslant (\ell-1)^2$, and $q = d^2$ with $d \geqslant 2$ a prime power, there exists a $BOA(N, n, q, \ell)$ whose length N scales as $N = O(n^{\ell-1}\log n)$. That is, there exists a bounded-strength BOA decoupling scheme to switch off ℓ-local Hamiltonians on n interacting d-dimensional qudits that uses $N = O(n^{\ell-1}\log n)$ time slices.

Proof. First, note that if n is not of the special form $n = q^m$ where $m \geqslant 1$, then we can always embed the n qudits into a larger system of q^m qudits with $m = \lceil \log_q(n) \rceil$, construct a scheme for the larger system, and ignore the additional qudits (as per Remark 4.5). This increases n by a factor of at most q and therefore doesn't affect the statement of the theorem, i.e. we can without loss of generality assume that $n = q^m$ where $m \geqslant 1$.

Now, we consider the code C that is the *dual* of a $k^\perp$-dimensional $\mathrm{BCH}^{\mathrm{ext}}(\mathbb{F}_{q^m}/\mathbb{F}_q, D)$ code with designed distance $D = \ell+1$. Thus C has length n, dual distance $\delta^\perp \geqslant D = \ell+1$, and, according to Theorem 4.8, dimension $k = n - k^\perp \leqslant m(D-2)+1 = m(\ell-1)+1$. By Theorem 4.3, this means that we can construct an $n \times N_{OA}$ orthogonal array of strength $\delta^\perp - 1 \geqslant \ell$ from this code, where $N_{OA} = q^k \leqslant q^{m(\ell-1)+1} = qn^{\ell-1}$. According to Theorem 4.7, the corresponding BOA has an overhead that scales at most logarithmically in n since from Eq. (4.4) we obtain the following bound on the length of the bounded-strength decoupling scheme corresponding to the BOA: $N_{BOA} = q^k|S_q^{(k)}| = q^k 2ke \leqslant [qn^{\ell-1}][2(m(\ell-1)+1)e] = 2qen^{\ell-1}[(\ell-1)\log_q n + 1] = O(n^{\ell-1}\log n)$. This establishes the claimed bound. □

In physical systems, the locality ℓ is generally a small fixed number, so the requirement of $n \geqslant (\ell-1)^2$ is inconsequential asymptotically, while for small n, one can (by Remark 4.5) always artificially increase n to satisfy it. Our main focus in Theorem 4.9 is on the asymptotic cost for fixed locality ℓ as the number n of qudits grows. It should be noted that, depending on the particular choice of q, ℓ, and n, further improvements over the bound in Theorem 4.9 are possible; see e.g., [17, 18]. This in turn leads to further improvements in the length of the decoupling schemes constructed via Theorem 4.7. For instance, for 2-local qubit Hamiltonians we saw at the end of Sec. 4.5 that Hamming codes can be used to construct BOA decoupling schemes of length $N = 2[3n+1]\log_4[3n+1]$, giving a slight improvement over schemes constructed from primitive BCH codes which lead to a scaling of $N \leqslant 8n[\log_4(n)+1]$.

4.7 Tables of best known BOA schemes for small systems

In the following, we present a summary of the best known BOA schemes for qubit ($d=2$, $q=4$) and qutrit ($d=3$, $q=9$) systems for a variety of small localities ℓ and system sizes n. All schemes are obtained by our main construction in Theorem 4.7, where the underlying classical linear codes are either taken from the literature or from the Magma [19] database of best known linear codes which can be accessed using the Magma command `BestDimensionLinearCode(<field>, <length>, <distance>)`.

Recall from Remark 4.5 that if we have a BOA decoupling scheme for n qudits, it can also be used for smaller systems of $n' < n$ qudits. Therefore, the best known BOA for n qudits is also the best known BOA for all $n' < n$ qudits unless a better BOA scheme for n' is known. Table 4.1 summarizes the best known schemes for systems of n qubits ($d=2$), for small values of n, that can be obtained from good linear codes. Similarly, Table 4.2 summarizes the best known schemes for systems of n qutrits ($d=3$), for small values of n.

$\ell \setminus N$	64	384	2 048	10 240	49 152	229 376	1 048 576
2	2–5^{a}	6–21^{a}	22–85^{a}	86–341^{a}	342–1 365^{a}	1 366–5 461^{a}	5 462–21 845^{a}
3	-	3–6^{b}	7–17^{c}	18–41^{c}	42–126^{c}	127–288^{c}	289–756^{c}
4	-	-	4–5	6–11^{d}	12–21^{e}	22–43	44–85
5	-	-	-	5–6	7–12^{f}	13–20	21–27
6	-	-	-	-	6–7	8–9	10–17
7	-	-	-	-	-	7–8	9–10
8	-	-	-	-	-	-	8–9

Table 4.1: Table of the best known balanced-cycle orthogonal arrays (BOAs) for qubit ($d = 2$) systems, indicating the number of qubits that can be decoupled by a BOA scheme for the given locality and length. Shown are the locality ℓ of the underlying Hamiltonian from 2 up to 8 and length $N = 4^k 2k$ of the BOA cycles from 64 up to 1 048 576, corresponding to the values $k = 2, \ldots, 8$ in Eq. (4.4) with $q = 4$ and $e = 1$. Each entry in the table denotes the range of the number n of qubits that can be achieved by a BOA scheme of the corresponding locality and length. For instance, the entry 7–17 at location $(3, 2\,048)$ indicates that in order to decouple a 3-local Hamiltonian on a system with n qubits, where $n \in \{7, \ldots, 17\}$, the best known BOA schemes have 2 048 time steps. If the number of qubits is one higher, e. g., $n = 18$, then the currently best known BOA scheme would require 10 240 time steps. Superscripts indicate if the dual codes $[n, k^{\perp}, \delta^{\perp}]_4$ underlying the BOAs were obtained by a particular construction: a) all codes for $\ell = 2$ were obtained from the Hamming code family $[n, n-k, 3]_4$ with $k = \log_4(3n + 1)$; b) the code $[6, 3, 4]_4$ is the Hexacode [13]; c) the codes with parameters $[17, 13, 4]_4$, $[41, 36, 4]_4$, $[126, 120, 4]_4$, $[288, 281, 4]_4$, and $[756, 748, 4]_4$ are based on caps in finite projective spaces which are sets of points of which no three are collinear, see [18]; d) the code $[11, 6, 5]_4$ is a quadratic residue code, see [11, 5.13] and [13]; e) the code $[21, 15, 5]_4$ is the Kschischang-Pasupathy code, see [20]; and f) the code $[12, 6, 6]_4$ is a quadratic residue code, see [11, 5.13] and [13]. All other codes in the table are based on the database of best known linear codes that is available in Magma [19].

$\ell \setminus N$	324	4 374	52 488	590 490	6 377 292	66 961 566
2	2–10^a	11–91^a	92–820^a	821–7 381^a	7 382–66 430^a	66 431–597 871^a
3	-	3–10^b	11–82^b	83–212^b	213–840^b	841–6 723^b
4	-	-	4–10	11–20	21–72	73–96
5	-	-	-	5–10	11–16	17–73
6	-	-	-	-	6–10	11–17
7	-	-	-	-	-	7–10

Table 4.2: Table of the best known balanced-cycle orthogonal arrays (BOAs) for qutrit ($d = 3$) systems, indicating the number of qutrits that can be decoupled by a BOA scheme for the given locality and length. Shown are the locality ℓ of the underlying Hamiltonian from 2 up to 7 and length $N = 9^k 2k$ of the BOA cycles from 324 up to 66 961 566, corresponding to the values $k = 2, \ldots, 7$ in Eq. (4.4) with $q = 9$ and $e = 1$. Each entry in the table denotes the range of the number n of qutrits that can be achieved by a BOA scheme of the corresponding locality and length. Superscripts indicate if the dual codes $[n, k^\perp, \delta^\perp]_9$ underlying the BOAs were obtained by a particular construction: a) all codes for $\ell = 2$ were obtained from the Hamming code family $[n, n-k, 3]_9$ with $k = \log_9(8n+1)$; and b) the codes with parameters $[10, 7, 4]_9$, $[82, 78, 4]_9$, $[212, 207, 4]_9$, $[840, 834, 4]_9$, and $[6\,723, 6\,716, 4]_9$ are based on caps in finite projective spaces, see [18]. All other codes in the table are based on the database of best known linear codes that is available in Magma [19].

4.8 Examples

Example 4.1 (2-local decoupling of a diagonal Hamiltonian). We first consider a simple case of decoupling a 2-local Hamiltonian on 7 qubits, where we assume (to simplify the example) that the Hamiltonian is diagonal, i.e. consists only of Pauli Z operators. In this case, it turns out that we need not use an irreducible representation and can consequently use $q = d = 2$ (instead of $q = d^2 = 4$); to avoid clutter, we defer the proof that this works to Example 4.3 where we will consider a similar situation. Because $q = 2$, we use the group $\mathbb{F}_2 = \mathbb{Z}_2 = \{0, 1\}$ and choose the representation

$$\rho : \mathbb{Z}_2 \to \{\mathbb{1}, X\}, \qquad \text{with} \qquad \rho(0) = \mathbb{1}, \quad \rho(1) = X$$

and corresponding control unitaries

$$u_0(\delta) = \mathbb{1}, \quad u_1(\delta) = e^{-iX\delta}, \quad \text{over time } \delta \in [0, \tfrac{\pi}{2}]. \tag{4.5}$$

Note that by evolving over time $\Delta = \frac{\pi}{2}$, we can therefore implement (up to phase) $u_0(\Delta) = \mathbb{1} = \rho(0)$ and $u_1(\Delta) = X = \rho(1)$. We assume that we can perform these control unitaries on any qubit.

With our constraints of 7 qubits ($n = 7$) with 2-local interactions ($\ell = 2$) and our ability to use $q = 2$, we seek an $[n, k]_q = [7, k]_2$ code with dual distance $\delta^\perp = 3 = \ell + 1$ for some (hopefully small) dimension k. We find that there is a $[7, 3]_2$ code with this desired dual distance, given by the generator matrix

$$G = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}.$$

Observe that this code has dimension $k = 3$, which will dictate the efficiency of the protocol.

Although irrelevant for our concerns here, one may observe that, as guaranteed by Theorem 4.3, an array built from the codewords of this code is an orthogonal array; indeed, the OA shown in Fig. 4-3 was constructed from this code. We, on the other hand, wish to create a BOA from this code. As per Theorem 4.7, we start with the (additive) group $\mathbb{F}_q^k = \mathbb{Z}_2^3$ and choose the generating set $S_q^{(k)} = S_2^{(3)} = \left\{ \left(\begin{smallmatrix}1\\0\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}0\\1\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}0\\0\\1\end{smallmatrix}\right) \right\}$. We set $\mathcal{L}$ to be the Eulerian cycle on the Cayley graph $\Gamma\Big(\mathbb{Z}_2^3, S_2^{(3)}\Big)$ shown in Fig. 4-2(a), namely

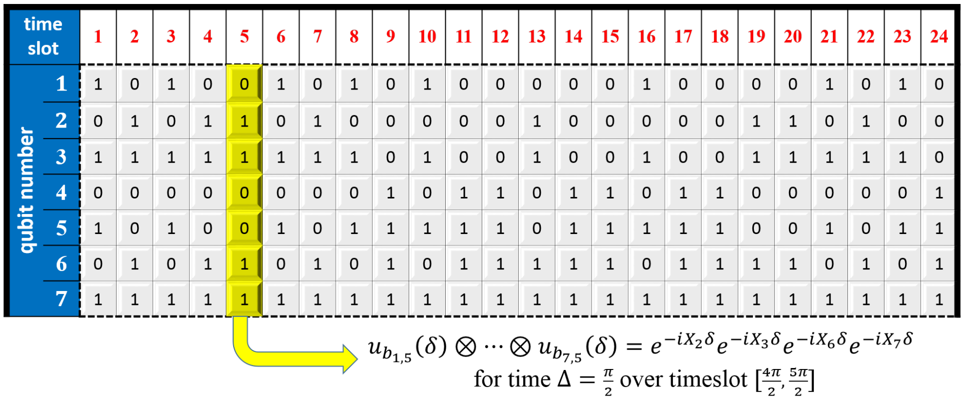
$$\begin{aligned} \mathcal{L}_{\mathbb{F}_2^3} = \Bigg(& \left(\begin{smallmatrix}0\\0\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}1\\0\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}1\\1\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}0\\1\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}0\\0\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}0\\1\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}1\\1\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}1\\0\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}0\\0\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}0\\0\\1\end{smallmatrix}\right), \left(\begin{smallmatrix}1\\0\\1\end{smallmatrix}\right), \\ & \left(\begin{smallmatrix}1\\0\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}1\\0\\1\end{smallmatrix}\right), \left(\begin{smallmatrix}1\\1\\1\end{smallmatrix}\right), \left(\begin{smallmatrix}1\\1\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}1\\1\\1\end{smallmatrix}\right), \left(\begin{smallmatrix}0\\1\\1\end{smallmatrix}\right), \left(\begin{smallmatrix}0\\1\\0\end{smallmatrix}\right), \left(\begin{smallmatrix}0\\1\\1\end{smallmatrix}\right), \left(\begin{smallmatrix}0\\0\\1\end{smallmatrix}\right), \left(\begin{smallmatrix}0\\1\\1\end{smallmatrix}\right), \left(\begin{smallmatrix}1\\1\\1\end{smallmatrix}\right), \left(\begin{smallmatrix}1\\0\\1\end{smallmatrix}\right), \left(\begin{smallmatrix}0\\0\\1\end{smallmatrix}\right) \Bigg). \end{aligned}$$

We map this cycle under the action of the generator matrix G to obtain the array $A_{G\mathcal{L}} = [Gm]_{m \in \mathcal{L}_{\mathbb{F}_2^3}}$, which is precisely the array that was shown in Fig. 4-4. According to Theorem 4.7, this is a $BOA(N, n, q, \ell) = BOA(24, 7, 2, 2)$ with $N = q^k |S_2^{(3)}| = 2^3 \cdot 3 = 24$. By definition, this means that every $\ell \times N$ subarray (obtained by only considering a selection of just ℓ of the n rows) defines a balanced cycle on the Cayley graph $\Gamma(\mathcal{G}, \mathscr{S})$ of $\mathcal{G} = \mathbb{F}_q^\ell = \mathbb{Z}_2^2$ with respect to some generating set $\mathscr{S}$ (which may depend on the subarray). For example, look at rows 5 and 7, highlighted in Fig. 4-4. This defines the balanced cycle on $\Gamma\Big(\mathbb{Z}_2^2, \left\{ \left(\begin{smallmatrix}0\\1\end{smallmatrix}\right), \left(\begin{smallmatrix}1\\1\end{smallmatrix}\right) \right\} \Big)$ that was shown in Fig. 4-2(b). The generating set and balanced cycle depend on the choice of rows, but by virtue of being a BOA, some balanced cycle will be obtained for any choice of 2 rows.

According to Theorem 4.2, the protocol of Protocol 4.2 defined by this BOA performs bounded-strength decoupling on our 7-qubit 2-local system. To construct this protocol, we consider the transitions between the columns of the BOA, thereby defining the schedule shown in Fig. 4-6. The control unitaries to be applied are defined by these transitions and our choice of Eq. (4.5), which was chosen to be consistent with our representation ρ. Specifically, according to Protocol 4.2, the control cycle evolution is $U_c\Big((j-1)\Delta + \delta\Big) = u_{\vec{b}_j}(\delta) U_c\Big((j-1)\Delta\Big)$, $\delta \in [0, \Delta]$, where $\vec{b}_j$ is the jth column in Fig. 4-6. For example, in time slot 5 the transition column is $\vec{b}_5 = (0, 1, 1, 0, 0, 1, 1)^T$, which corresponds to the unitary

$$u_{\vec{b}_5}(\delta) = e^{-iX_2\delta} e^{-iX_3\delta} e^{-iX_6\delta} e^{-iX_7\delta}$$

where X_i denotes the Pauli X operator on the ith qubit; in other words, we should apply



	time slot	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
qubit number	1	1	0	1	0	0	1	0	1	0	1	0	0	0	0	0	1	0	0	0	0	1	0	1	0
	2	0	1	0	1	1	0	1	0	0	0	0	0	1	0	0	0	0	0	1	1	0	1	0	0
	3	1	1	1	1	1	1	1	1	0	1	0	0	1	0	0	1	0	0	1	1	1	1	1	0
	4	0	0	0	0	0	0	0	0	1	0	1	1	0	1	1	0	1	1	0	0	0	0	0	1
	5	1	0	1	0	0	1	0	1	1	1	1	1	0	1	1	1	1	1	0	0	1	0	1	1
	6	0	1	0	1	1	0	1	0	1	0	1	1	1	1	1	0	1	1	1	1	0	1	0	1
	7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

$u_{b_{1,5}}(\delta) \otimes \cdots \otimes u_{b_{7,5}}(\delta) = e^{-iX_2\delta} e^{-iX_3\delta} e^{-iX_6\delta} e^{-iX_7\delta}$

for time $\Delta = \frac{\pi}{2}$ over timeslot $[\frac{4\pi}{2}, \frac{5\pi}{2}]$

Figure 4-6: A 7×24 array defining the decoupling protocol in Example 4.1 in the format of Fig. 4-1. Rows correspond to qubit numbers, columns correspond to time slots (each of width $\Delta = \frac{\pi}{2}$), and entries correspond to unitary operators on qubits according to Eq. (4.5). As per Protocol 4.2, the control cycle evolution is $U_c\Big((j-1)\Delta + \delta\Big) = u_{\vec{b}_j}(\delta) U_c\Big((j-1)\Delta\Big)$, $\delta \in [0, \Delta]$, where $\vec{b}_j$ is the jth column. For example, because $\vec{b}_5 = (0,1,1,0,0,1,1)^T$, we have $u_{\vec{b}_5}(\delta) = u_0 \otimes u_1 \otimes u_1 \otimes u_0 \otimes u_0 \otimes u_1 \otimes u_1\,(\delta) = e^{-iX_2\delta} e^{-iX_3\delta} e^{-iX_6\delta} e^{-iX_7\delta}$. This array was formed from the transitions between the columns of the BOA in Fig. 4-4. If $\vec{a}_j$ denotes the jth column in the Fig. 4-4 BOA (with columns labelled $j = 0, \ldots, 23$), then $\vec{b}_j = \vec{a}_j - \vec{a}_{j-1}$ is the jth column in the present array (with columns labelled $j = 1, \ldots, 24$, and treating $\vec{a}_{24} = \vec{a}_0 = 0$).

the control Hamilonian $H_c(t) = X_2 + X_3 + X_6 + X_7$ during this time slot. This protocol will decouple any 2-local 7-qubit diagonal Hamiltonian.

We emphasize that in this simple diagonal-Hamiltonian example, we were able to use $q = d = 2$ (for reasons that will be addressed in Example 4.3). If the Hamiltonian were not known to be diagonal, this would not in general have been possible, and we would have needed to instead use a $[7, k]_q$ code for $q = 4$.

Example 4.2 (2-local decoupling using a Hamming code). Consider an arbitrary 2-local Hamiltonian H on a system of 5 qubits. Then H can be decoupled by applying a BOA derived from the code dual to a $[5,3,3]_4$ Hamming code, namely the $[5,2]_4$ code over $\mathbb{F}_4$ with the generator matrix

$$G = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & \alpha^2 \\ \alpha^2 & \alpha^2 \\ \alpha^2 & 1 \end{bmatrix},$$

where α is a primitive element of order 3 of $\mathbb{F}_4$. Note that we arrange the codewords as *column* vectors, consistent with the notation used throughout this chapter and some – but not all – of the literature. Since here $k = 2$, $d = 2$, and $e = 1$, the corresponding BOA has a total number of time steps given by $N = d^{2k} 2ke = 64$. When arranged into the columns of a 5×64 matrix, each of the 64 control Hamiltonians that are applied in this scheme corresponds to one of the 16 codewords of the $[5,2]_4$ code.

Example 4.3 (5-local decoupling of a diagonal Hamiltonian using a BCH code). Recall from Remark 4.1 that if one is interested in decoupling a Hamiltonian of a particular form, it may not be necessary for ρ to be irreducible, and in such a case it may be possible to choose a code over a field $\mathbb{F}_q$ for which q is less than d^2. Consider a diagonal (i.e., Z-only) 5-local Hamiltonian H on a system of 16 qubits. Then H can be decoupled by applying a BOA derived from the dual code of a $\mathrm{BCH}^{\mathrm{ext}}(\mathbb{F}_2^4/\mathbb{F}_2, 6) = [16, 7, 6]_2$, i.e. from a code over $\mathbb{F}_2 = \mathbb{Z}_2$ with parameters $[16, 9]_2$ and generator matrix

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}.$$

Due to the special structure of the Hamiltonian, we are able to choose $q = d = 2$ (rather than $q = d^2 = 4$) in this case. Since $k = 9$, $d = 2$, $e = 1$, and the Hamiltonian is Z only, the corresponding BOA has a total number of time steps given by $N = d^k ke = 4\,608$. When arranged into the columns of an $16 \times 4\,608$ matrix, each of the $4\,608$ control Hamiltonians that are applied in this scheme corresponds to one of the 512 codewords of the $[16, 9]_2$ code.

To construct our protocol from this code we first choose a generating set $S_2^{(9)}$ for $\mathbb{F}_2^9$, such as the $k = 9$ standard basis vectors $\{(1, 0, 0, \ldots)^T, (0, 1, 0, 0, \ldots)^T, \ldots\}$. We then find an Eulerian cycle $\mathcal{L}$ on the Cayley graph $\Gamma(\mathbb{F}_2^9, S_2^{(9)})$ and map it to an Eulerian cycle $G\mathcal{L}$ on the Cayley graph $\Gamma(G[\mathbb{F}_2^9], G[S_2^{(9)}])$ using the generator matrix above. Our choice of $S_2^{(9)}$ as being the standard basis vectors would dictate that the transition labels $\vec{b} = Gs$, for $s \in S_2^{(9)}$, are simply the columns of G. Our BOA consists of the $2^9 = 512$ 16-bit codewords, each appearing exactly 9 times according to the order specified by $G\mathcal{L}$. To use the BOA as a decoupling scheme, we may choose

$$\rho : \{0, 1\} \to \{\mathbb{1}, X\}, \quad \text{with} \quad \rho(0) = \mathbb{1}, \quad \rho(1) = X$$

and choose the corresponding single-qubit control unitaries to be

$$u_0(\delta) = \mathbb{1}, \quad u_1(\delta) = e^{-iX\delta}, \quad \text{over time } \delta \in [0, \tfrac{\pi}{2}].$$

Observe that (ignoring global phase) $u_0(\frac{\pi}{2}) = \mathbb{1} = \rho(0)$ and $u_1(\frac{\pi}{2}) = X = \rho(1)$. The multi-qubit control unitaries are defined by $u_{\vec{b}} = u_{b_1} \otimes \cdots \otimes u_{b_{16}}$ (for $\vec{b} \in \{0, 1\}^{16}$). For example, if $s = (1, 0, 0, \ldots, 0)^T$, then $\vec{b} = Gs$ is the first column of G and

$$u_{\vec{b}} = e^{-iX_1\delta} e^{-iX_{10}\delta} e^{-iX_{13}\delta} e^{-iX_{14}\delta} e^{-iX_{15}\delta} e^{-iX_{16}\delta}$$

acting non-trivially on qubits 1, 10, 13, 14, 15, and 16. The control scheme in Protocol 4.2 is thus specified.

We now prove that this example works, even though ρ is reducible (i.e. even though we are choosing $q = 2$ rather than $q = 4$). As per the argument in the proof of Theorem 4.2, we need only focus on a single 5-local term of H (so assume without loss of generality that H consists of only one such term), we can ignore all but the 5 qubits on which it acts non-trivially, and we need only speak of the 5-qubit unitaries $u_{\mathscr{s}}(\delta)$ that act on those qubits and correspond to $\mathscr{s} \in \mathscr{S}$ (where $\mathscr{S}$ is the generator set for $\mathbb{F}_2^5$ derived from the BOA for those 5 qubits). According to the proof of Theorem 4.1, our scheme works if and only if $\Pi_{\mathcal{G}}\Big(F_{\mathscr{S}}(H)\Big) = 0$. Here, however, ρ is not irreducible, so $\Pi_{\mathcal{G}}$ will not kill all traceless operators; indeed, X-only operators commute with each $U_{\mathscr{g}}$ and are therefore unmodified by $\Pi_{\mathcal{G}}$.

To show that $\Pi_{\mathcal{G}}\Big(F_{\mathscr{S}}(H)\Big) = 0$ nevertheless holds, observe from Eq. (4.3) that each term in $F_{\mathscr{S}}(H)$ is of the form $u_{\mathscr{s}}^{\dagger} H u_{\mathscr{s}}$. Now, H is diagonal, i.e. a tensor product of only $\mathbb{1}$ and Z, and $u_{\mathscr{s}}(\delta)$ is a tensor product of only $\mathbb{1}$ and $e^{-iX\delta}$. Therefore, because $e^{iX\delta} Z e^{-iX\delta} = \cos(2\delta)Z + \sin(2\delta)Y$, we see that $u_{\mathscr{s}}^{\dagger} H u_{\mathscr{s}}$ can be expanded as a sum of tensor products of $\mathbb{1}$, Z and Y. Moreover, because H is traceless and conjugation by a unitary is trace-preserving, this sum cannot contain a term proportional to the identity, $\mathbb{1}^{\otimes 5}$. Thus, each of these terms consists of at least one operator (which for notational purposes we take to be on the first qubit) that is a Z or a Y, i.e. each can be written in the form $\sigma \otimes A$, where $\sigma \in \{Y, Z\}$ and A is some 4-fold tensor product of operators from $\{\mathbb{1}, Y, Z\}$. Our protocol is defined by a BOA of strength 5, so any subset of 5 rows of the BOA consists of all 2^5 5-tuples in $\mathbb{F}_2^5$ repeated an equal number of times. Thus the sum in $\Pi_{\mathcal{G}}$ involves conjugating by each $U_{\mathscr{g}}$ where $U_{\mathscr{g}}$ ranges over all 2^5 possible tensor products that can be formed on 5 qubits using $\mathbb{1}$ and X. Focusing on the first qubit, we can equivalently say that the $U_{\mathscr{g}}$ range over all possible $\mathbb{1} \otimes B$ and $X \otimes B$, where B ranges over $\{B_2 \otimes B_3 \otimes B_4 \otimes B_5 : B_i \in \{\mathbb{1}, X\}\}$. Conjugating $\sigma \otimes A$ by $\mathbb{1} \otimes B$ yields either $\sigma \otimes A$ or $-\sigma \otimes A$, whereas conjugating instead by $X \otimes B$ yields the same result but with the opposite sign (since $\sigma \in \{Y, Z\}$). In other words, $(\mathbb{1} \otimes B)(\sigma \otimes A)(\mathbb{1} \otimes B) + (X \otimes B)(\sigma \otimes A)(X \otimes B) = 0$. Thus, the sum in $\Pi_{\mathcal{G}}$ cancels in pairs, i.e. $\Pi_{\mathcal{G}}\Big(F_{\mathscr{S}}(H)\Big)$ is indeed 0.

4.9 Conclusion

We have shown how to use bounded-strength controls to decouple n interacting qudits of dimension $d = p^e$ (for some prime p and positive integer e) that obey an ℓ-local Hamiltonian. The system may be either closed or open (i.e. coupled to an environment), as long as both the system Hamiltonian and the environmental couplings are ℓ-local on the system. The decoupling scheme is described using a balanced-cycle orthogonal array, which we introduced and showed how to construct from classical linear codes. To determine the best possible scheme based on our method, we have to find the best linear error-correcting code $C^{\perp} = [n, k^{\perp}]_q$ of length n and distance at least $\ell + 1$. By *the best*, we mean $k^{\perp}$ should be maximized for the given system size (n) and locality (ℓ). The construction in the present chapter yields a decoupling scheme that uses $N = d^{2k} 2ke$ time slices (of fixed length) where $k = n - k^{\perp}$.

Finding the best code is a key problem in the theory of error-correcting codes; extensive code tables have been compiled for small distances. For the important case of qubits with 2-local interactions, for example, one can use Hamming codes over $\mathbb{F}_4$ such that $k = \log_4(3n + 1)$, whence N scales like $n \log n$. For higher degrees of locality, we can use families of BCH codes to construct the decoupling schemes. The designed distance of these codes is chosen based on the locality ℓ of the Hamiltonian, leading to a scaling of N as $n^{\ell-1} \log n$. An open question is whether the schemes so derived are optimal in the asymptotic sense, i.e. whether, for fixed ℓ and qudit dimension d, a better scaling with n is possible. We note that it is known [21] that when using bang-bang pulses, time at least $\Omega(n)$ is necessary to decouple general 2-body Hamiltonians, whereas our bounded-strength scheme takes time $O(n \log n)$ using Hamming codes for such Hamiltonians. Another interesting open question is to develop a theory for systems with mixed qudit dimensions. All schemes derived here are decoupling schemes up to first order, and while it is easy to extend this to second order using symmetry, it would be interesting to find schemes that also achieve decoupling to higher orders. Finally, we mention as an avenue for future research the application of the derived bounded-strength decoupling schemes for the purpose of Hamiltonian simulation.

Chapter bibliography

[1] A. D. Bookatz, M. Roetteler, and P. Wocjan. *Improved bounded-strength decoupling schemes for local Hamiltonians.* IEEE Transactions on Information Theory, 62(5):2881 [2016]. http://dx.doi.org/10.1109/TIT.2016.2535183. © 2016 IEEE; reprinted with permission.

[2] L. Viola and E. Knill. *Robust dynamical decoupling of quantum systems with bounded controls.* Phys. Rev. Lett., 90(3):037901 [2003]. http://dx.doi.org/10.1103/PhysRevLett.90.037901

[3] D. A. Lidar and T. A. Brun (eds.). *Quantum error correction.* Cambridge University Press, Cambridge, UK [2013]

[4] M. Stollsteimer and G. Mahler. *Suppression of arbitrary internal coupling in a quantum register.* Phys. Rev. A, 64(5):052301 [2001]. http://dx.doi.org/10.1103/PhysRevA.64.052301

[5] P. Wocjan, M. Rötteler, D. Janzing, and T. Beth. *Simulating Hamiltonians in quantum networks: Efficient schemes and complexity bounds.* Phys. Rev. A, 65(4):042309 [2002]. http://dx.doi.org/10.1103/PhysRevA.65.042309

[6] M. Rotteler and P. Wocjan. *Equivalence of decoupling schemes and orthogonal arrays.* IEEE Transactions on Information Theory, 52(9):4171 [2006]. http://dx.doi.org/10.1109/TIT.2006.880059

[7] P. Wocjan. *Efficient decoupling schemes with bounded controls based on Eulerian orthogonal arrays.* Phys. Rev. A, 73(6):062317 [2006]. http://dx.doi.org/10.1103/PhysRevA.73.062317

[8] R. R. Ernst, G. Bodenhausen, and A. Wokaun. *Principles of nuclear magnetic resonance in one and two dimensions.* Clarendon Press, Oxford, UK [1987]

[9] J. S. Waugh, L. M. Huber, and U. Haeberlen. *Approach to high-resolution NMR in solids.* Phys. Rev. Lett., 20(5):180 [1968]. http://dx.doi.org/10.1103/PhysRevLett.20.180

[10] U. Haeberlen. *High resolution NMR in solids: selective averaging*, *Adv. Magn. Res.*, volume 1. Academic Press [1976]. http://dx.doi.org/10.1016/B978-0-12-025561-0.50001-0

[11] A. S. Hedayat, N. J. A. Sloane, and J. Stufken. *Orthogonal arrays: theory and applications.* Springer Series in Statistics. Springer, New York, NY, USA [1999]. http://dx.doi.org/10.1007/978-1-4612-1478-6

[12] M. Rötteler. *Dynamical decoupling schemes derived from Hamilton cycles.* Journal of Mathematical Physics, 49(4):042106 [2008]. `http://dx.doi.org/10.1063/1.2904471`

[13] F. J. MacWilliams and N. J. A. Sloane. *The theory of error-correcting codes*, volume 16. North Holland Publishing Co., Amsterdam, Netherlands [1977]

[14] S. Lin and D. J. Costello. *Error control coding.* Prentice-Hall, Upper Saddle River, NJ, USA, 2 edition [2004]

[15] R. Roth. *Introduction to coding theory.* Cambridge University Press, New York, NY, USA [2006]

[16] M. Sudan. *Lecture notes for Algorithmic introduction to coding theory (MIT 6.897).* `http://people.csail.mit.edu/madhu/FT01/`

[17] S. Yekhanin and I. Dumer. *Long nonbinary codes exceeding the Gilbert-Varshamov bound for any fixed distance.* IEEE Transactions on Information Theory, 50(10):2357 [2004]. `http://dx.doi.org/10.1109/TIT.2004.834744`

[18] Y. Edel and J. Bierbrauer. *Large caps in small spaces.* Designs, Codes and Cryptography, 23(2):197 [2001]. `http://dx.doi.org/10.1023/A:1011216716700`

[19] W. Bosma, J. Cannon, and C. Playoust. *The Magma algebra system I: the user language.* Journal of Symbolic Computation, 24(3–4):235 [1997]. `http://dx.doi.org/10.1006/jsco.1996.0125`

[20] F. R. Kschischang and S. Pasupathy. *Some ternary and quaternary codes and associated sphere packings.* IEEE Transactions on Information Theory, 38(2):227 [1992]. `http://dx.doi.org/10.1109/18.119683`

[21] D. Janzing, P. Wocjan, and T. Beth. *Complexity of decoupling and time reversal for n spins with pair interactions: Arrow of time in quantum control.* Phys. Rev. A, 66(4):042311 [2002]. `http://dx.doi.org/10.1103/PhysRevA.66.042311`

Chapter 5

Quantum logic with interacting bosons in 1D

In this chapter, we present a scheme for implementing high-fidelity quantum logic gates using a few interacting bosons tunnelling on a one-dimensional lattice. The gate operation is carried out by a single compact lattice described by a one-dimensional Bose-Hubbard model with only nearest-neighbour hopping and on-site interactions. We find high-fidelity deterministic logic operations for a gate set (including the CNOT gate) that is universal for quantum information processing. We discuss the applicability of this scheme in light of recent developments in controlling and monitoring cold-atoms in optical lattices.

This chapter is adapted from [1], which was joint work with Yoav Lahini, Gregory R. Steinbrecher, and Dirk Englund.

5.1 Introduction

The tunnelling or “hopping” of quantum particles on lattice potentials is a text-book example of quantum dynamics. While a fundamental concept in the description of the solid state, it was usually not directly relevant to experiments, as those usually could not resolve the dynamics of individual quantum particles in a sample. In recent years, new ways of monitoring the evolution of quantum particles in lattice potentials have emerged in optics [2–8], as well as in the field of ultra-cold atoms [9–14]. The degree of experimental control is truly remarkable: it is possible to prepare an initial state with single-site and single-particle resolution, to create a wide range of one- or two-dimensional lattice potentials, to determine the interaction between the particles, and to directly monitor in real space the evolving many-body distribution.

The ability to control and monitor quantum particles with such precision offers an interesting route to the implementation of quantum information processing and quantum computing schemes. Universal quantum computation has been theoretically shown possible using the quantum walk [15,16] of interacting particles on certain non-trivial two-dimensional lattices [17,18] and on one-dimensional lattices with a large number of degrees of freedom at each lattice site [19–21]. However, an implementation using quantum particles hopping on a simple one-dimensional lattice, without any additional degrees of freedom, has not been reported. Such a geometry would greatly simplify the experimental implementation, bringing it into the realm of recently reported experimental techniques [11, 13]. Furthermore, one-dimensional implementations offer other important practical advantages, for example

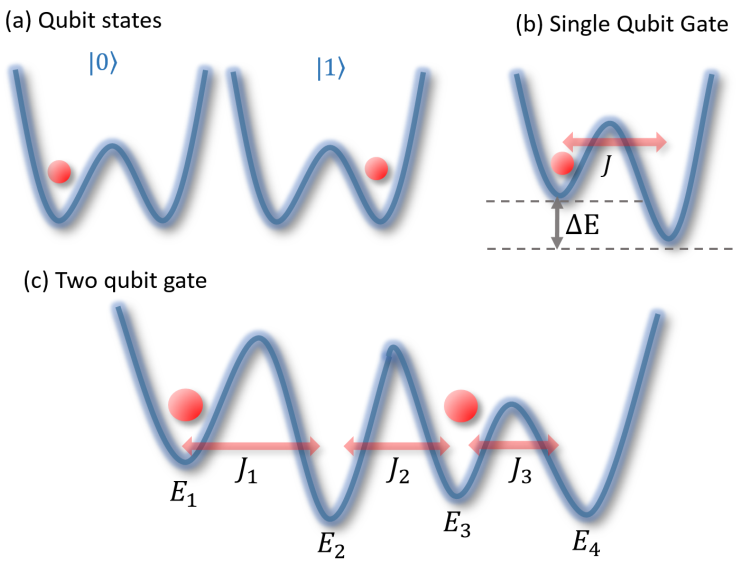


Figure 5-1: Illustration of one-dimensional quantum gates. (a) Qubits in the state $|0\rangle$ and $|1\rangle$ with dual-rail encoding. (b) Implementation of a single-qubit gate. (c) Schematic of a two-qubit system on a lattice.

freeing the second spatial dimension for important tasks such as error correction or connecting remote qubits. Other important tasks such as process tomography could still be performed in 1D (see Appendix 5.B).

In this work, we show how it is possible to use multi-particle hopping in a simple geometry — a one-dimensional lattice with only nearest-neighbour hopping and on-site interactions — as a compact platform for implementing quantum logic. We demonstrate our approach by detailing a set of lattice potentials that yield, with high fidelity, a universal set of quantum gates with only two sites per qubit. Moreover, the required lattice potential for each gate is time-invariant, simplifying the experimental implementation and possibly reducing the total operation time. Thus, high-fidelity gates can be constructed with lower probabilities of qubit-loss errors (i.e. lost particles during the computation) that necessitate computationally costly error-correction procedures. While our analysis is general and should be applicable to different quantum lattice systems, here we focus on interacting ultra-cold bosonic atoms trapped in an optical lattice, which allows us to include experimentally relevant bounds in the analysis. See [1] for a discussion of its application to nonlinear quantum photonic systems.

The dynamics of bosonic particles on a lattice is described by the time-independent many-body Bose-Hubbard Hamiltonian

$$H = \sum_m E_m N_m + \sum_{\langle l,m \rangle} J_{l,m} a_l^\dagger a_m + \frac{\Gamma}{2} \sum_m N_m (N_m - 1) \ , \tag{5.1}$$

where E_m is the on-site energy of site m, $a_m^\dagger \backslash a_m$ is the creation\annihilation operator for a boson in site m, $N_m = a_m^\dagger a_m$ is the number operator, $J_{l,m} \leqslant 0$ is the tunnelling rate between nearest neighbours, and Γ is the on-site interaction energy that arises when two or more bosons occupy the same site. The unitary transformation describing the evolution of multiple quantum particles propagating on the lattice is given by e^{-iHt}, where t is the propagation time. The quantum logic gates discussed here will be implemented by evolving under this Hamiltonian (with a suitable choice of parameters) for some predefined time t_{final} which we take to be $t_{\text{final}} = 1$.

5.2 Defining qubits on a lattice

The basic element of interest for quantum gates of the type discussed here is the quantum bit, or *qubit*. The quantum particles, however, evolve on a lattice according to the Hamiltonian described in Eq. (5.1). To define our qubits on the lattice, we use a spatial encoding where a qubit is physically implemented by a single boson in a pair of neighbouring potential wells (see Fig. 5-1), with the states $|0\rangle$ and $|1\rangle$ of the qubit defined by the particle being in the left or right well (i.e. dual-rail encoding). A single quantum particle can occupy the two sites in a superposition, encoding a qubit without the need for additional degrees of freedom. In this way, a system of n qubits can be realized in one dimension with n bosons and $2n$ lattice sites, with one boson in the first two sites (representing the first qubit), one boson in the next two sites (representing the second qubit), and so forth. Note that in this geometry, many physically permitted states (e.g. those with more than one particle on the same site) are not members of the logic space (i.e., the multi-qubit tensor-product space). Nevertheless, we show that it is possible to engineer the lattice parameters such that, at time $t = t_{\text{final}} = 1$, the transformation $U = e^{-iH}$ maps logic states only to other logic states with high fidelity, even though states outside this subbasis are allowed at intermediate times.

5.3 Implementing quantum gates

Having defined our qubits, we turn to the task of designing a universal set of quantum gates, i.e. finding lattice parameters that yield desired unitary transformations on the logical space. Designing and building quantum logic gates remains one of the most difficult aspects of quantum computing, and our case is no exception. From the physical description of a given device — in our case, the lattice parameters — it is straightforward to write down the many-particle Hamiltonian and to calculate the unitary evolution operator $U = e^{-iH}$ that fully describes the operation of the device. The inverse problem, however, is hard: given a desired unitary U, it is difficult to find a corresponding Hamiltonian that meets the physical and geometrical constraints of the device, e.g. the one-dimensionality of the lattice. Furthermore, if the logical quantum states are only a subset of the full Hilbert space, then the quantum gate operation is only a sub-matrix of the overall evolution operator U. In this case, U is not even uniquely defined by the desired gate operation. As described below, we tackle these difficulties with a computational approach that finds appropriate lattice parameters to approximate a given ideal gate operation with high fidelity.

There are many options for the selection of a universal set of gates. A useful choice is the gate set of the controlled-NOT (CNOT) operation, along with either all single-qubit rotations or the Hadamard and phase-shift single-qubit gates [22]. We first discuss the single-qubit gates, which are straightforward to calculate. We then elaborate on the construction of the CNOT gate, for which we take a computational approach.

5.3.1 Single-qubit gates

We present the exact construction for a set of single-qubit gates that, together with the CNOT gate, make a universal quantum gate set. Since these are one-qubit operations, they are implemented using one particle in two lattice sites. As such, the interaction (Γ) is irrelevant and the matrix of lattice parameters

$$G = \begin{pmatrix} E_1 & J_{12} \\ J_{12} & E_2 \end{pmatrix}$$

(with $J_{12} \leqslant 0$) can be directly interpreted as the Hamiltonian governing the single particle. The unitary gate, obtained by evolving with G for a time $t_{\text{final}} = 1$, is then $U = e^{-iG}$.

One simple universal quantum gate set includes the Hadamard gate and the phase-shift gate (in addition to the CNOT gate). The phase-shift gate is the simplest to implement. It is composed of two decoupled lattice sites in which the on-site energy between the sites is detuned. Specifically, to implement the single-qubit phase-shift operator $R_\theta = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}$, one may apply the single-qubit Hamiltonian

$$G_{R_\theta} = \begin{pmatrix} 0 & 0 \\ 0 & -\theta \end{pmatrix} . \tag{5.2}$$

Next in complexity is the single-qubit Hadamard gate $H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$. The Hamiltonian and propagation time that generates the Hadamard transformation can be solved analytically, and is given by

$$G_H = \frac{\pi}{2\sqrt{2}} \begin{pmatrix} \sqrt{2}-1 & -1 \\ -1 & \sqrt{2}+1 \end{pmatrix} . \tag{5.3}$$

Note that a simple tunnelling between two identical wells for half the tunnelling time (analogous to the beamsplitter operation in linear optics) will not reproduce the Hadamard gate in this case. Unlike the bulk quantum optics situation, in which the optical beamsplitter can be asymmetric in phase and thus can reproduce the Hadamard gate exactly, under our Hamiltonian dynamics the splitting is symmetric in phase and therefore modified tunnelling rates and additional diagonal terms are required to correct the output phases. We note this holds true also for integrated quantum photonics gates [23].

Alternatively, one can use the universal gate set of CNOT together with all single-qubit unitaries. Any single-qubit unitary, U, can be implemented by first decomposing it in the form [22]

$$U = e^{i\alpha} R_z(\beta) R_x(\gamma) R_z(\delta) = e^{i\alpha} R_z(\beta) H R_z(\gamma) H R_z(\delta) .$$

The z-rotation $R_z(\theta) = \begin{pmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{pmatrix}$ can be implemented with the Hamiltonian

$$G_{R_z(\theta)} = \frac{\theta}{2} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} ,$$

or, equivalently up to phase, using G_{R_θ} above. The x-rotation $R_x(\theta) = \exp \begin{pmatrix} 0 & -i\frac{\theta}{2} \\ -i\frac{\theta}{2} & 0 \end{pmatrix}$

can be implemented either with the Hamiltonian

$$G_{R_x(\theta)} = \frac{4\pi - \theta}{2} \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix} \tag{5.4}$$

or by conjugating $R_z(\theta)$ by the Hadamard operation H described earlier. The phase $e^{i\alpha}$ can be implemented with $G_\alpha = -\alpha \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ if desired.

5.3.2 CNOT gate

To design the CNOT gate, i.e. the two-qubit gate

$$\text{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix},$$

with the dual-rail encoding, we consider a lattice with four sites and two bosons. This problem then is defined by eight lattice parameters in Eq. (5.1): four on-site potential terms (E_m), three tunnelling terms ($J_{l,m}$), and the interaction parameter (Γ). The complete two-body Hamiltonian H is described by a 10×10 matrix (the size of the Hilbert space for two identical bosons in four modes). To perform the logical gate operation, the system is evolved according to $U = e^{-iH}$. The CNOT gate operation is then given by a 4×4 sub-matrix of U over the logical basis states $|1010\rangle$, $|1001\rangle$, $|0110\rangle$, $|0101\rangle$ (presented here in the occupation number basis); the other six basis states, while physically allowed, are not members of the logical basis.

As explained above, finding the physical lattice parameters from the desired gate is a non-trivial inverse problem. Using nonlinear optimization techniques [24–27] (as we will detail in Sec. 5.5), we optimized the eight parameters of the system to maximize the fidelity of the gate when acting on the logical input states under the constraint that the parameters represent a physical one-dimensional lattice, i.e. that the on-site parameters are real, that the tunnelling parameters are real and non-positive and connect only nearest-neighbouring sites, and that the values of the on-site, tunnelling, and interaction terms are within experimentally relevant bounds. Specifically, we demanded that $-J_{max} \leqslant J_{l,m} \leqslant 0$, $-J_{max} \leqslant E_m \leqslant J_{max}$, and $\Gamma \leqslant \Gamma_{max}$, where J_{max} and Γ_{max} are the largest allowed tunnelling rate and interaction level in the optimization protocol. In our optimization we set $J_{max} = 4\pi$, limiting the maximum number of tunnelling events (or Rabi-oscillations) to 4. In practice, this experimental bound is dictated by the loss and decoherence rate of the system, determining the maximal relevant propagation time. We also set $\Gamma_{max} = 10 J_{max}$.

An example of a resulting lattice that yields the two-qubit CNOT gate is given (to two decimal places) by

$$G_{\text{CNOT}} = \pi \begin{pmatrix} 0.40 & 0 & 0 & 0 \\ 0 & 1.82 & -1.03 & 0 \\ 0 & -1.03 & -0.37 & -3.80 \\ 0 & 0 & -3.80 & -0.66 \end{pmatrix} \tag{5.5}$$

with interaction strength $\Gamma = 21.68\pi$. Here, the diagonal and off-diagonal entries of G_{CNOT} represent the parameters E_m and $J_{l,m}$, respectively, of the Hamiltonian H. Eq. (5.5) repre-

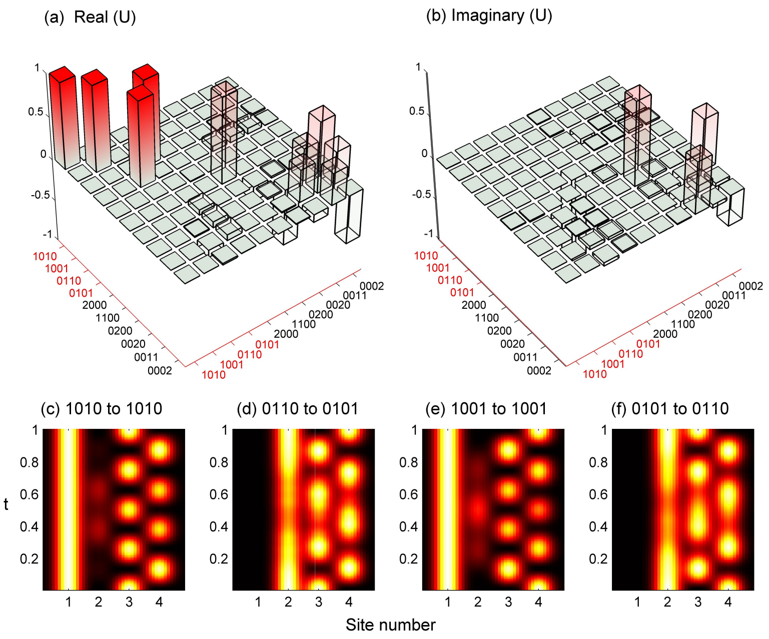


Figure 5-2: An implementation of the controlled-NOT (CNOT) gate according to the recipe in Eq. (5.5). (a) The real part and (b) the imaginary part of the two-particle unitary transform, U. The CNOT gate operation corresponds to the sub-matrix of the logic states, shown in solid-colour bars and marked with red axis labels. Plots (c)-(f) show the position (in terms of the lattice sites, 1-4) of the particle density as a function of time, t, revealing the operation principle of the gate on each logical state ($|00\rangle$, $|10\rangle$, $|01\rangle$, and $|11\rangle$, respectively). One observes that the target qubit (in sites 3 & 4) performs Rabi-oscillations that are perturbed by the state of the control qubit (in sites 1 & 2) — the target qubit performs one fewer Rabi-flip if the control qubit is in the $|1\rangle$ state.

sents a recipe for a four-site lattice that yields a CNOT gate with fidelity[1] of 99.6%, whose operation is summarized in Fig. 5-2.

If the bounds on the parameters are relaxed, the fidelity approaches even closer to unity. Fig. 5-3 summarizes the optimization results. Fig. 5-3(a) shows the convergence of independent runs with random starting points to the same final result. Fig. 5-3(b) presents the expected gate fidelity vs. the maximally allowed values of the interaction Γ_{max}. For a fixed maximal tunnelling of $J_{max} = 4\pi$, the fidelity achieves a value close to 0.95 at $\Gamma_{max}/J_{max} = 0.5$ and then slowly approaches unity as this value is further increased. In a system with a given Γ_{max}, it is still possible to improve the fidelity further by allowing more tunnelling events to take place, i.e. increasing J_{max}; see Fig. 5-3(c).

We have performed an analysis of the effect of imperfections and noise, and found that these are expected to have a negligible effect on the fidelity of the CNOT gate. These results are presented in Appendix 5.A. We therefore expect the main reduction in fidelity to occur between the application of sequential gates, when the parameters of the system are modified. Encouragingly, high-fidelity switching between potentials were reported recently [14].

[1] The fidelity used throughout this chapter is defined in Eq. (5.7) of Sec. 5.5.

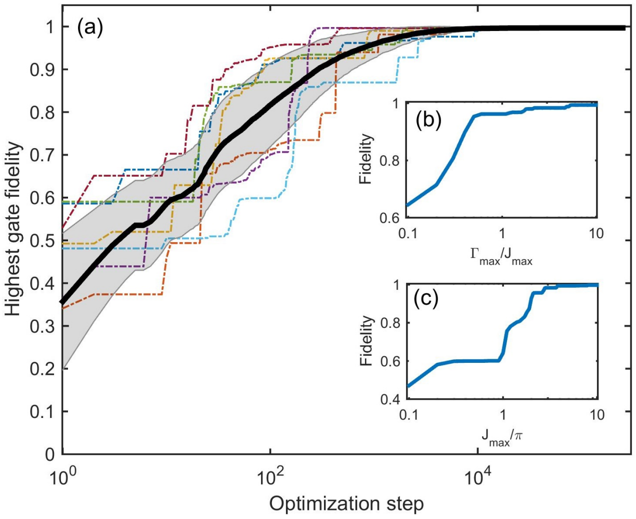


Figure 5-3: Optimization of the CNOT gate fidelity. (a) Convergence of different optimization runs to the optimal gate fidelity. The solid black line and the shaded area represents the average and the standard deviation values over 512 runs. Seven example runs are shown in the background (dotted lines). (b) Gate fidelity versus the maximum allowed interaction level Γ_{max}, at a constant $J_{max} = 4\pi$. (c) Gate fidelity for different maximal tunnelling rates J_{max} at a constant maximal interaction level of $\Gamma_{max} = 20\pi$.

5.4 Compiling a three-qubit primitive

Implementing a quantum algorithm using the scheme presented in this chapter will involve several lattice configurations operating in sequence, as gates are sequentially applied in the algorithm. In principle, because the gate set presented in this work is universal, any multi-qubit operation can be broken down into a sequence of single- and two-qubit gates, and thus implemented using the gates already presented. However, compiling common multi-step operations into a single primitive based on a single, time-independent Hamiltonian could reduce the possibility of errors arising from dynamic changes to the lattice. As an example, we constructed a 3-qubit gate, shown in Fig. 5-4. This gate is useful, for instance, in the 2-bit Deutsch-Jozsa algorithm [28], performing the oracle for the function $f(x,y) = x \oplus y$. (All other oracles for the 2-bit Deutsch-Jozsa algorithm are either a simple variation of this oracle or require only single-qubit gates plus at most one CNOT gate.) Our computational approach allowed us to find a set of lattice parameters that realizes the complete three-qubit operation in a single gate. Fig. 5-4 presents an implementation of this three-qubit operation, at a fidelity of 99.8%, using a single, one-dimensional six-site lattice:

$$G = \pi \begin{pmatrix} 5.98 & 0 & 0 & 0 & 0 & 0 \\ 0 & 7.13 & -1.21 & 0 & 0 & 0 \\ 0 & -1.21 & 0.14 & -12.04 & 0 & 0 \\ 0 & 0 & -12.04 & 0.18 & -1.37 & 0 \\ 0 & 0 & 0 & -1.37 & 11.69 & 0 \\ 0 & 0 & 0 & 0 & 0 & -8.03 \end{pmatrix} \tag{5.6}$$

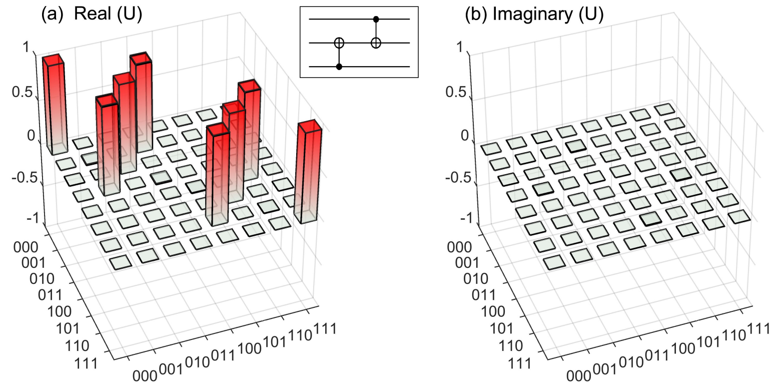


Figure 5-4: A 3-qubit operation of 2 CNOT gates (inset), compiled into a single gate U. (a) The real and (b) imaginary parts of our implementation of U. Only the logical basis states are shown.

with interaction strength $\Gamma = 108.24\pi$. In this case too, the fidelity could be improved by allowing larger tunnelling rates.

5.5 Computational methods

In this section, we detail the numerical methods used to find the gates presented in Secs. 5.3.2 and 5.4. We used a free-software implementation of a variety of numerical optimization algorithms [24]. This enabled us to, with a single specification of cost function and constraints, compare the success and computational cost of a number of different optimization approaches. We found that a randomly-seeded global optimization algorithm [25, 26] combined with a gradient-free local algorithm [27] gave the best performance, both in terms of number of iterations and computational run-time.

Careful selection of the cost function was crucial to the success of this work, and interacted with the choice of the aforementioned algorithms, particularly the local optimizer. Throughout this chapter, we define the fidelity of the gate in terms of the Hilbert-Schmidt inner-product between the target unitary gate operation U_0 and the unitary operation U generated by the Hamiltonian at a given step of the optimization (restricted to the logical subspace). Specifically, the fidelity is defined to be

$$F(U_0, U) = |\langle U_0, U\rangle_{\mathrm{C}}| \tag{5.7}$$

with

$$\langle U_0, U\rangle_{\mathrm{C}} = \frac{\mathrm{Tr}(U_0^\dagger U)}{D},$$

where D is the dimension of the logical space (4 for two-qubit gates). This fidelity can be interpreted as a lower-bound average fidelity of the gate.

To be specific about the iterative numerical process, at each step we generated U from a vector corresponding to lattice parameters and calculated $F(U_0, U)$. Numerically, we found that minimizing the function $1 - F^2$, rather than $1 - F$, gave superior performance. In the case of the algorithm given in Ref. [27], the reason for this is clear: the algorithm assumes a quadratic cost function. However, we found that even with algorithms designed for linear cost functions (e.g. [29]), convergence was much slower than for the quadratic cost function.

Finally, in order to ensure that U has the same global phase as U_0 (this is for aesthetic purposes, as $F(U_0, U)$ is invariant under multiplication by a global phase), we placed a cost on the phase of the matrix element $u_{1,1}$. We found this to be most efficiently implemented by adding the term $\sin(\arg(u_{1,1}))^2$ to the cost function. This function is quadratic when perturbed about zero, is non-negative, and is symmetric about $k\pi$ for all $k \in \mathbb{Z}$, making it an ideal candidate function. We verified that the introduction of this additional cost both yielded a U with appropriate phase (see Fig. 5-2) and did not result in a decreased fidelity compared to optimization without this constraint.

5.6 Conclusions

We have shown how quantum logic gates can be realized with high fidelity using interacting quantum particles hopping on a one-dimensional lattice. In particular, we gave a design for a high-fidelity CNOT gate along with exact descriptions of single-qubit rotations, a computationally complete set. Additionally, we demonstrated the compilation of a higher-order gate operation into a single operation. Our approach carries several important advantages over previous schemes. First, due to the dual-rail encoding we employ, the states of the system can be prepared and measured by simply placing and detecting single quantum particles at certain positions, both of which are straightforward in present experimental systems using cold atoms [11]. Second, each quantum operation is carried out by a single, one-dimensional, time-invariant lattice potential. Third, the devices we propose are compact lattices of size $2n$ (where n is the number of qubits) that can be realized on a line of potential wells with only nearest-neighbour hopping, in agreement with experimental capabilities. While focusing on ultra-cold atoms for concreteness and to include experimental constraints, our scheme is general and should be adaptable to other lattice quantum systems such as certain nonlinear quantum-optical systems. In a broader context, the results reported here suggest that the computational complexity of even simple 1D lattices is high, meaning it should be possible to experimentally implement many-body processes that cannot be efficiently simulated on classical computers. Although quantum computing in the gate model is the most obvious application, this computational power need not be so limited and invites the construction of new, different computational schemes that are more natural for the continuous nature of the dynamics on a lattice.

Chapter appendices

5.A Noise analysis

We have performed an analysis to test the robustness of our scheme to two general types of perturbations. In the first, the perturbations are static (i.e. time-independent): the parameters of the system are slightly randomized but are fixed in time. This simulates the finite accuracy in experimentally implementing the different parameters of the system (on-site potentials, tunnelling rates, etc.). The second type of perturbation is dynamic (i.e. time-dependent), simulating noise – for example due to fluctuation in the power of the laser that is generating the optical potential. We then compare them to typical experimental values.

For our analysis, we performed Monte-Carlo simulations of perturbations to the lattice parameters E_m, $J_{l,m}$, and Γ for the CNOT gate presented in Sec. 5.3.2. The perturbations were drawn from zero-mean Normal distributions with varying standard deviations. The standard deviation for each lattice parameter (defining the level of the perturbation) was chosen as a fraction of the allowed experimental range for that parameter. Equivalently, the RMS value of the noise measured in dBFS (decibels full-scale) was constant for all lattice parameters at a given level of noise.

For the static perturbations, simulations were performed by calculating the fidelity of the unitary transformation resulting from the disturbed lattice parameters, over 100,000 realizations. For the dynamic perturbations, simulations were performed by breaking down the propagation into 1,000 steps, each with an independently perturbed unitary. The lattice parameters for each of the 1,000 unitaries were drawn independently and in the same fashion as for the static unitaries. Here, for each time-dependent level of disorder we simulated 1,000 perturbations (again, with 1,000 time-steps per simulation). This is not a comprehensive analysis of time-dependent noise, but we believe it gives the correct order of magnitude for these effects.

In Fig. 5-5 we plot the average relative change in fidelity $\Delta F = \frac{F_{\text{Ideal}} - F_{\text{Noise}}}{F_{\text{Ideal}}}$ vs. the noise level in dBFS for both static and dynamic disorder. For noise levels at and below -20 dBFS, these follow simple power-law relationships for which we have included fits. Note that while ΔF has a square-law relationship to static disorder, it has a linear relationship to dynamic disorder.

The data points in the figure below correspond to the means of the Monte-Carlo samples while the error bars correspond to one standard deviation. On this log-log scale and at low noise, the magnitudes of these standard deviations seem to be independent of noise level; for static disorder, the standard deviations are all approximately 4 dB while for dynamic disorder they are approximately 5 dB.

We note that for the cold atoms experiments, both dynamic and static perturbations are on the order of 10^{-3} of the expected value, i.e -30 dBFS [13, 30] . According to our analysis

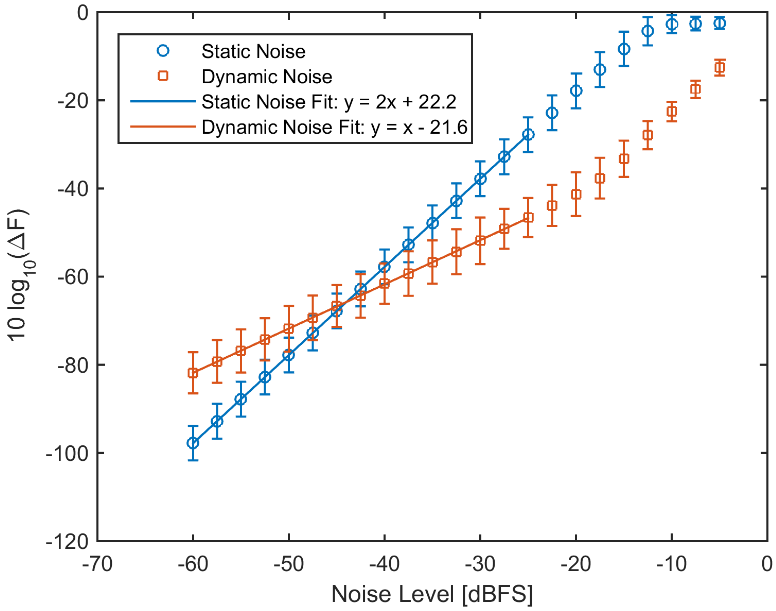


Figure 5-5: Results of the perturbation analysis for the CNOT gate in Sec. 5.3.2, showing the relative change in fidelity ΔF due to time-independent (blue circles) and time-dependent (red squares) perturbations. The error bars represent the standard deviation of the results across realizations. The plot also includes fits to a linear (for dynamic perturbations) and quadratic (for static perturbations) dependencies (solid lines).

this would correspond to a negligible cost in fidelity of about 10^{-4}.

5.B Quantum process tomography for CNOT

In this appendix, we outline a very basic process tomography method for the CNOT gate described in this chapter. The idea of quantum process tomography [31] is to completely determine the mathematical structure of a given operation (in our case, the CNOT on two qubits), by applying the operation to a variety of different input states and measuring each of the resulting states in a variety of bases.

Tomography procedures allow much freedom in choosing input states and measurement bases. We want to make such choices wisely to facilitate performing the procedure on our particular experimental apparatus. Two main points guide these choices. One is that the only two-qubit gate be the purported CNOT gate itself – all other gates required for the tomography should be relatively simple so that we adequately assess the CNOT as the major source of errors. The second is that the other gates be easy for the Bose-Hubbard model to implement, preferably with as few gates as possible (since limitations in coherency times may limit the number of gates reasonably performed using current technology). Here we suggest input states and measurement bases to accomplish these goals, requiring only two single-qubit Bose-Hubbard Hamiltonians (one prior to the CNOT, and one following it) per (computational) measurement. Specifically, each measurement in the tomography involves applying an operation of the form $(V_3 \otimes V_4)$ CNOT $(V_1 \otimes V_2)$ to a computational basis state, followed by a measurement in the computational basis, where each V_i can be performed with a single one-qubit Bose-Hubbard Hamiltonian; we therefore need only keep the state coherent for three consecutive operations.

In the following, we follow the quantum process tomography procedure outlined in Ref. [22], which requires measuring Pauli observables. To measure a Pauli matrix $\sigma \in \{\mathbb{1}, X, Y, Z\}$, one must perform a basis-transformation gate to the state so as to measure in the eigenbasis of σ. For $\sigma = \mathbb{1}$ and Z, this is trivial, as the computational basis suffices; however, for $\sigma = X$ and Y, it is recommended to measure in the following eigenbases, which can be performed by first applying the following basis-transformation matrices,

$$X : \left\{ \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} \right\}, \quad U_X = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{5.8}$$

and

$$Y : \left\{ \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ i \end{pmatrix}, \frac{1}{\sqrt{2}} \begin{pmatrix} i \\ 1 \end{pmatrix} \right\}, \quad U_Y = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -i \\ -i & 1 \end{pmatrix}. \tag{5.9}$$

Observe that U_X is the Hadamard matrix, the Hamiltonian for which was given in Eq. (5.3), and that $U_Y = R_x(\pi/2)$, which is generated by the Hamiltonian in Eq. (5.4) with $\theta = \pi/2$. Note that for Y we chose a non-standard orthonormal eigenbasis so that the basis-transformation can be accomplished using a single Bose-Hubbard gate.

The input states required to perform the quantum process tomography procedure of Ref. [22] are straightforward to produce. For example, one may use tensor products of the following single-qubit input states: the computational basis states $|0\rangle$ and $|1\rangle$, the +1 X-eigenvector $|+\rangle = U_X|0\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$, and the +1 Y-eigenvector $|-\rangle = U_Y^\dagger|0\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ i \end{pmatrix}$, where U_X and U_Y are given in Eqs. (5.8) and (5.9). Note that $U_Y^\dagger = R_x(7\pi/2)$ can be generated by the Hamiltonian in Eq. (5.4) with $\theta = 7\pi/2$.

Chapter bibliography

[1] Y. Lahini, G. R. Steinbrecher, A. D. Bookatz, and D. Englund. *Quantum logic with interacting bosons in 1D.* arXiv:1501.04349 [2015]. http://arxiv.org/abs/1501.04349

[2] H. B. Perets, Y. Lahini, F. Pozzi, M. Sorel, R. Morandotti, and Y. Silberberg. *Realization of quantum walks with negligible decoherence in waveguide lattices.* Phys. Rev. Lett., 100(17):170506 [2008]. http://dx.doi.org/10.1103/PhysRevLett.100.170506

[3] Y. Bromberg, Y. Lahini, R. Morandotti, and Y. Silberberg. *Quantum and classical correlations in waveguide lattices.* Phys. Rev. Lett., 102(25):253904 [2009]. http://dx.doi.org/10.1103/PhysRevLett.102.253904

[4] A. Peruzzo, M. Lobino, J. C. F. Matthews, N. Matsuda, A. Politi, K. Poulios, X.-Q. Zhou, Y. Lahini, N. Ismail, K. Wörhoff, Y. Bromberg, Y. Silberberg, M. G. Thompson, and J. L. OBrien. *Quantum walks of correlated photons.* Science, 329(5998):1500 [2010]. http://dx.doi.org/10.1126/science.1193515

[5] M. A. Broome, A. Fedrizzi, B. P. Lanyon, I. Kassal, A. Aspuru-Guzik, and A. G. White. *Discrete single-photon quantum walks with tunable decoherence.* Phys. Rev. Lett., 104(15):153602 [2010]. http://dx.doi.org/10.1103/PhysRevLett.104.153602

[6] A. Schreiber, K. N. Cassemiro, V. Potoček, A. Gábris, P. J. Mosley, E. Andersson, I. Jex, and C. Silberhorn. *Photons walking the line: A quantum walk with adjustable coin operations.* Phys. Rev. Lett., 104(5):050502 [2010]. http://dx.doi.org/10.1103/PhysRevLett.104.050502

[7] A. Regensburger, C. Bersch, B. Hinrichs, G. Onishchukov, A. Schreiber, C. Silberhorn, and U. Peschel. *Photon propagation in a discrete fiber network: An interplay of coherence and losses.* Phys. Rev. Lett., 107(23):233902 [2011]. http://dx.doi.org/10.1103/PhysRevLett.107.233902

[8] P. P. Rohde, A. Schreiber, M. Štefaňák, I. Jex, and C. Silberhorn. *Multi-walker discrete time quantum walks on arbitrary graphs, their properties and their photonic implementation.* New J. Phys., 13(1):013001 [2011]. http://dx.doi.org/10.1088/1367-2630/13/1/013001

[9] C. Weitenberg, M. Endres, J. F. Sherson, M. Cheneau, P. Schauß, T. Fukuhara, I. Bloch, and S. Kuhr. *Single-spin addressing in an atomic Mott insulator.* Nature, 471(7338):319 [2011]. http://dx.doi.org/10.1038/nature09827

[10] T. Fukuhara, P. Schauß, M. Endres, S. Hild, M. Cheneau, I. Bloch, and C. Gross. *Microscopic observation of magnon bound states and their dynamics.* Nature, 502(7469):76 [2013]. http://dx.doi.org/10.1038/nature12541

[11] P. M. Preiss, R. Ma, M. E. Tai, A. Lukin, M. Rispoli, P. Zupancic, Y. Lahini, R. Islam, and M. Greiner. *Strongly correlated quantum walks in optical lattices.* Science, 347(6227):1229 [2015]. http://dx.doi.org/10.1126/science.1260364

[12] Y. Lahini, M. Verbin, S. D. Huber, Y. Bromberg, R. Pugatch, and Y. Silberberg. *Quantum walk of two interacting bosons.* Phys. Rev. A, 86(1):011603 [2012]. http://dx.doi.org/10.1103/PhysRevA.86.011603

[13] A. M. Kaufman, B. J. Lester, C. M. Reynolds, M. L. Wall, M. Foss-Feig, K. R. A. Hazzard, A. M. Rey, and C. A. Regal. *Two-particle quantum interference in tunnel-coupled optical tweezers.* Science, 345(6194):306 [2014]. http://dx.doi.org/10.1126/science.1250057

[14] R. Islam, R. Ma, P. M. Preiss, M. E. Tai, A. Lukin, M. Rispoli, and M. Greiner. *Measuring entanglement entropy through the interference of quantum many-body twins.* arXiv:1509.01160 [2015]. http://arxiv.org/abs/1509.01160

[15] Y. Aharonov, L. Davidovich, and N. Zagury. *Quantum random walks.* Phys. Rev. A, 48(2):1687 [1993]. http://dx.doi.org/10.1103/PhysRevA.48.1687

[16] E. Farhi and S. Gutmann. *Quantum computation and decision trees.* Phys. Rev. A, 58(2):915 [1998]. http://dx.doi.org/10.1103/PhysRevA.58.915

[17] A. M. Childs, D. Gosset, and Z. Webb. *Universal computation by multiparticle quantum walk.* Science, 339(6121):791 [2013]. http://dx.doi.org/10.1126/science.1229957

[18] M. S. Underwood and D. L. Feder. *Bose-Hubbard model for universal quantum-walk-based computation.* Phys. Rev. A, 85(5):052314 [2012]. http://dx.doi.org/10.1103/PhysRevA.85.052314

[19] D. Aharonov, D. Gottesman, S. Irani, and J. Kempe. *The power of quantum systems on a line.* Commun. Math. Phys., 287(1):41 [2009]. http://dx.doi.org/10.1007/s00220-008-0710-3

[20] B. A. Chase and A. J. Landahl. *Universal quantum walks and adiabatic algorithms by 1D Hamiltonians.* arXiv:0802.1207 [2008]. http://arxiv.org/abs/0802.1207

[21] D. Nagaj and P. Wocjan. *Hamiltonian quantum cellular automata in one dimension.* Phys. Rev. A, 78(3):032311 [2008]. http://dx.doi.org/10.1103/PhysRevA.78.032311

[22] M. A. Nielsen and I. L. Chuang. *Quantum computation and quantum information.* Cambridge University Press, Cambridge, UK, 1 edition [2000]

[23] A. Politi, M. J. Cryan, J. G. Rarity, S. Yu, and J. L. O'Brien. *Silica-on-silicon waveguide quantum circuits.* Science, 320(5876):646 [2008]. http://dx.doi.org/10.1126/science.1155441

[24] S. G. Johnson. *The NLopt nonlinear-optimization package.* http://ab-initio.mit.edu/nlopt

[25] A. H. G. R. Kan and G. T. Timmer. *Stochastic global optimization methods part I: Clustering methods.* Mathematical Programming, 39(1):27 [1987]. http://dx.doi.org/10.1007/BF02592070

[26] A. H. G. R. Kan and G. T. Timmer. *Stochastic global optimization methods part II: Multi level methods*. Mathematical Programming, 39(1):57 [1987]. http://dx.doi.org/10.1007/BF02592071

[27] M. J. D. Powell. *The BOBYQA algorithm for bound constrained optimization without derivatives*. Technical report NA2009/06, Department of Applied Mathematics and Theoretical Physics, Cambridge, UK [2009]. http://www.damtp.cam.ac.uk/user/na/NA_papers/NA2009_06.pdf

[28] D. Deutsch and R. Jozsa. *Rapid solution of problems by quantum computation*. Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, 439(1907):553 [1992]. http://dx.doi.org/10.1098/rspa.1992.0167

[29] M. J. D. Powell. *Direct search algorithms for optimization calculations*. Acta Numerica, 7:287 [1998]. http://dx.doi.org/10.1017/S0962492900002841

[30] P. P. J. Zupancic. *Dynamic holography and beamshaping using digital micromirror devices*. Master's thesis, Harvard University, Cambridge, MA, USA [2013]

[31] I. L. Chuang and M. A. Nielsen. *Prescription for experimental determination of the dynamics of a quantum black box*. Journal of Modern Optics, 44(11-12):2455 [1997]. http://dx.doi.org/10.1080/09500349708231894

Chapter 6

Testing quantum expanders is co-QMA-complete

A quantum expander is a unital quantum channel that is rapidly mixing, has only a few Kraus operators, and can be implemented efficiently on a quantum computer. In this chapter, we consider the problem of estimating the mixing time (i.e., the spectral gap) of a quantum expander. We show that this problem is co-QMA-complete. This has applications to testing randomized constructions of quantum expanders and to studying the thermalization of open quantum systems.

This chapter is adapted from [1], which was joint work with Stephen P. Jordan, Yi-Kai Liu, and Pawel Wocjan.

6.1 Introduction

A quantum expander is a unital quantum channel that is rapidly mixing. This means that, with repeated applications of the channel, every quantum state is rapidly contracted to the maximally mixed state, which is the channel's unique fixed point. In addition, a quantum expander has only a small number of Kraus operators, each of which is described by an efficient quantum circuit. Quantum expanders are quantum analogues of expander graphs, which play a prominent role in computer science and discrete mathematics [2]. The idea of quantum expanders was introduced in [3, 4]; since then, several explicit constructions of quantum expanders have been discovered, and quantum expanders have found various applications in quantum information theory, such as constructing quantum states with unusual entanglement properties, and simulating thermalization in quantum systems [5–10].

Here we study the problem of estimating the mixing rate of a quantum expander. Given a quantum channel Φ of the above form (a small number of Kraus operators, specified by quantum circuits), this problem is to estimate the spectral gap of Φ. Such a problem may arise in connection with randomized constructions of quantum expanders [10], where with high probability one obtains a good expander, but it is not obvious how to test that a particular instance of the construction is in fact good. In addition, this problem can be viewed as a special case of a more general question: given an open quantum system, determine whether it thermalizes, and on what time scale. (The behaviour of a quantum expander is roughly equivalent to that of a quantum system with a particular weak coupling to a bath of harmonic oscillators, as we shall discuss.)

Formally, we define the QUANTUM NON-EXPANDER problem (which is the complement of the above problem), and we give evidence that this problem is computationally intractable; we prove that it is QMA-complete. Here QMA (Quantum Merlin-Arthur) is a complexity class that is a quantum analogue of NP (Nondeterministic Polynomial Time) [11–13]. Proving that a problem is QMA-complete implies that it is equivalent (up to polynomial-time reductions) to all other QMA-complete problems, a survey of which can be found in Chapter 7 or [14]. In particular, this implies that the problem cannot be solved in polynomial time (unless QMA = BQP). Furthermore, this implies that our original problem, the QUANTUM EXPANDER problem, is co-QMA-complete and therefore cannot be in QMA (unless QMA = co-QMA). In other words, when a channel Φ is *not* a quantum expander, there is an efficiently-verifiable quantum proof of that fact, but when Φ *is* a quantum expander, there is no way of giving an efficiently-verifiable quantum proof (assuming QMA $\neq$ co-QMA).

6.2 Preliminaries

6.2.1 The QUANTUM NON-EXPANDER problem

We use the definition of explicit quantum expanders due to Ben-Aroya, Schwartz, and Ta-Shma [5]. For an N-dimensional Hilbert space $\mathcal{H}$, let $L(\mathcal{H})$ denote the space of linear operators from $\mathcal{H}$ to itself. A superoperator $\Phi : L(\mathcal{H}) \to L(\mathcal{H})$ is *admissible* if it is a completely positive and trace-preserving map. An admissible superoperator is *unital* if $\Phi(\tilde{I}) = \tilde{I}$, where $\tilde{I} = \frac{1}{N}\mathbb{1}$ is the maximally mixed state on $\mathcal{H}$ (where $\mathbb{1}$ is the identity operator on $\mathcal{H}$). A unital superoperator is *D-regular* if it can be written as

$$\Phi = \frac{1}{D}\sum_{d=1}^{D}\Phi_d,$$

with each Φ_d (for $d = 1, \ldots, D$) acting on arbitrary $A \in L(\mathcal{H})$ as

$$\Phi_d(A) = U_d A U_d^\dagger$$

where the U_d are unitary transformations on $\mathcal{H}$. The unitaries U_d are called the *operation elements*[1] (or *Kraus operators*) of Φ, and D is called the *degree* of Φ. A D-regular superoperator is *explicit* if each of its operation elements U_d can be implemented by a quantum circuit of size polylog(N), where N is the dimension of $\mathcal{H}$.

Definition 6.1 (Quantum expander). A D-regular superoperator $\Phi : L(\mathcal{H}) \to L(\mathcal{H})$ is a *κ-contractive expander* if, for all $A \in L(\mathcal{H})$ that are orthogonal to $\tilde{I}$ with respect to the Hilbert-Schmidt inner product (i.e. $\mathrm{Tr}\left[A\tilde{I}\right] = 0$, so A is traceless), it holds that

$$\|\Phi(A)\|_F \leq \kappa \|A\|_F. \tag{6.1}$$

Here the *Frobenius norm* is given by $\|A\|_F = \sqrt{\sum_{i,j}|a_{ij}|^2}$, where a_{ij} are the entries of the matrix A. The quantity $1 - \kappa$, for the smallest such κ for Φ, is called the *spectral gap* of Φ.

Remark 6.1. The motivation for this definition can easily be seen from the following argument. A good quantum expander Φ should rapidly send any density matrix ρ to the

[1] Strictly speaking, the operation elements are $\frac{1}{\sqrt{D}}U_d$, but we will nonetheless refer to U_d as being an operation element.

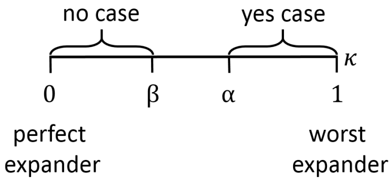


Figure 6-1: The QUANTUM NON-EXPANDER is to determine where, on the shown number line, is the value of the contractivity κ of the given D-regular superoperator Φ. The YES case corresponds to a bad expander (relative to the input parameters α and β).

maximally mixed state $\tilde{I}$. Because $\text{Tr}[\rho] = 1 = \text{Tr}\left[\tilde{I}\right]$ we can always write $\rho = \tilde{I} + A$ where $\text{Tr}[A] = 0$. The requirement of Eq. (6.1) therefore formalizes the idea of Φ bringing ρ towards $\tilde{I}$ by rapidly killing off the A term. In this context Eq. (6.1) is equivalent to demanding that $\left\|\Phi(\rho) - \tilde{I}\right\|_{\text{F}} \leqslant \kappa \left\|\rho - \tilde{I}\right\|_{\text{F}}$, which clearly encapsulates the idea of Φ rapidly sending density matrices towards the maximally mixed state. Note that in this argument $A = \rho - \tilde{I}$ is Hermitian; however, it can be shown that if Eq. (6.1) applies for traceless Hermitian matrices, it also applies for traceless matrices in general, thus justifying Definition 6.1.

We consider the problem of estimating the mixing time of a quantum expander. Formally, we study the following decision problem.

Definition 6.2 (QUANTUM NON-EXPANDER problem). Fix some encoding such that each string $x \in \{0,1\}^*$ specifies an explicit D-regular superoperator $\Phi : (\mathbb{C}^2)^{\otimes m} \to (\mathbb{C}^2)^{\otimes m}$, with operation elements $U_1, \ldots, U_D$, and two parameters $\alpha > \beta$.

We will consider instances which satisfy the following promises[2]: m and D are upper-bounded by (fixed) polynomials in $|x|$; the parameters α and β are polynomially separated, i.e., they satisfy $\alpha - \beta \geq \frac{1}{q(|x|)}$ for some (fixed) polynomial q; and the operation elements $U_1, \ldots, U_D$ are given as quantum circuits of size at most $r(|x|)$ for some (fixed) polynomial r.

The QUANTUM NON-EXPANDER problem is the task of deciding which of the following is correct, given the promise that exactly one of them is correct:

- Φ is not an α-contractive expander (YES case), or
- Φ is a β-contractive expander (NO case).

Thus the problem is to determine whether a given Φ is a bad quantum expander (has only low contractivity), as sketched in Fig. 6-1.

6.2.2 Thermalization of open quantum systems

To motivate the QUANTUM NON-EXPANDER problem, we now describe a connection between that problem and the study of thermalization in open quantum systems. We show an example of a quantum system, coupled to a bath, where the system thermalizes and the relaxation time is determined by the spectral gap of a certain quantum expander.

[2] Here $|x|$ denotes the length of the string x.

Let the system consist of m qubits, and fix some unitary transformations U_d (for $d = 1, \dots, D$) which act on $(\mathbb{C}^2)^{\otimes m}$. Let the bath consist of a large number of harmonic oscillators, with annihilation operators b_{dk} (for $d = 1, \dots, D$ and $k \in \Omega$, where Ω is some large set). Let the total Hamiltonian be

$$H = H_S + \varepsilon H_I + H_B,$$

where the system Hamiltonian is $H_S = 0$, the bath Hamiltonian is

$$H_B = \sum_d \sum_k \omega_k b_{dk}^\dagger b_{dk},$$

and the system-bath coupling, of strength ϵ, is described by the interaction Hamiltonian

$$H_I = \sum_d (U_d \otimes f_d) + (U_d^\dagger \otimes f_d^\dagger),$$

where the operators f_d are defined by $f_d = \frac{1}{\sqrt{|\Omega|}} \sum_k b_{dk}$.

In the weak-coupling limit ($\varepsilon \to 0$), the time evolution of the system is described by a master equation [15]. Suppose the bath is in a thermal state, $\rho_B = (1/Z_B) \exp(-H_B/T)$. Then the master equation takes the form

$$\frac{d}{dt}\rho_S(t) = R_0 \sum_d \Big(U_d \rho_S(t) U_d^\dagger - \rho_S(t) \Big) + R_1 \sum_d \Big(U_d^\dagger \rho_S(t) U_d - \rho_S(t) \Big), \tag{6.2}$$

where $\rho_S(t)$ is the state of the system at time t, and R_0 and R_1 are positive real numbers. Equation (6.2) has two special features: there is no contribution from a "Lamb shift" Hamiltonian, and the dissipator is in diagonal form with Lindblad operators that are unitary. (See [1] for the derivation of this equation.)

Now define the quantum channel

$$\Phi(\rho) = \frac{R_0}{(R_0 + R_1)D} \sum_d U_d \rho U_d^\dagger + \frac{R_1}{(R_0 + R_1)D} \sum_d U_d^\dagger \rho U_d.$$

This channel Φ is a (non-uniform) mixture of unitary operations. In the special case where the set of unitaries $\{U_d : d = 1, \dots, D\}$ is closed with respect to the adjoint operation (i.e., where for every $d \in \{1, \dots, D\}$, there exists some $e \in \{1, \dots, D\}$ such that $U_d = U_e^\dagger$), the channel Φ can be written as

$$\Phi(\rho) = \frac{1}{D} \sum_d U_d^\dagger \rho U_d,$$

hence Φ is a D-regular superoperator. The master equation can now be rewritten in terms of Φ:

$$\frac{d}{dt}\rho_S(t) = (R_0 + R_1)D \left(\Phi - \mathcal{I}\right)(\rho_S(t)),$$

where $\mathcal{I}$ denotes the identity channel. We can solve for $\rho_S(t)$, obtaining

$$\rho_S(t) = \exp\Big(t\,(R_0 + R_1)D \left(\Phi - \mathcal{I}\right) \Big)(\rho_S(0)).$$

This system converges to the maximally mixed state as $t \to \infty$, and the rate of convergence depends on the spectral gap of Φ. More precisely, write $\rho_S(t) = \tilde{I} + A(t)$ where $A(t)$ is traceless. Then it can be verified that

$$\|A(t)\|_F \leq \exp\big(-t\,(R_0 + R_1)D(1-\kappa)\big)\|A(0)\|_F\,,$$

where $1-\kappa$ is the spectral gap of Φ. The smaller the value of κ, i.e. the larger the spectral gap, the faster the state $\rho_S(t)$ will converge to the maximally mixed state $\tilde{I}$.

6.2.3 Quantum Merlin-Arthur

We will show that the QUANTUM NON-EXPANDER problem is QMA-complete, that is, it is contained in QMA and every problem in QMA can be reduced to it in polynomial time.

The complexity class QMA consists of decision problems such that YES instances have concise quantum proofs. The name QMA stands for Quantum Merlin-Arthur, which is motivated by the following protocol. Given a problem instance x (i.e. a string of $|x|$ bits), and a language $L \in$ QMA, a computationally unbounded but untrustworthy prover, Merlin, submits a quantum state of poly($|x|$) qubits as a purported proof that $x \in L$. A verifier, Arthur, who can perform polynomial size quantum computations, then processes this proof and either accepts or rejects it. If $x \in L$ then there exists some polynomial size quantum state causing Arthur to accept with high probability (i.e. Merlin can successfully convince Arthur), but if $x \notin L$ then Arthur will reject all states with high probability (i.e. Merlin cannot cheat). QMA is a quantum analogue of MA, which is the probabilistic analogue of NP. A formal definition of QMA is as follows.

Definition 6.3 (QMA(a,b)). A language L is in QMA(a,b) if, for each $x \in \{0,1\}^*$, one can efficiently generate a quantum circuit V with the following properties:

- V acts on the Hilbert space $\mathcal{W} \otimes \mathcal{A}$ where

$$\mathcal{W} = (\mathbb{C}^2)^{\otimes n_w}, \quad \mathcal{A} = (\mathbb{C}^2)^{\otimes n_a},$$

  and the functions $n_w, n_a : \mathbb{N} \to \mathbb{N}$ grow at most polynomially in $|x|$;

- V consists of $s(|x|)$ elementary gates where the function $s : \mathbb{N} \to \mathbb{N}$ grows at most polynomially in $|x|$;

- if $x \in L$ (YES case) then there exists a witness state $|\psi\rangle \in \mathcal{W}$ such that

$$\|PV|\psi\rangle|\mathbf{0}\rangle\|^2 \geq a, \tag{6.3}$$

  where P and $|\mathbf{0}\rangle$ are defined below;

- if $x \notin L$ (NO case) then for all states $|\psi\rangle \in \mathcal{W}$ we have that

$$\|PV|\psi\rangle|\mathbf{0}\rangle\|^2 \leq b. \tag{6.4}$$

Here $\mathcal{W}$ and $\mathcal{A}$ are the witness and ancilla registers, respectively, and $P = |1\rangle\langle 1| \otimes \mathbb{1}$ projects onto the subspace of the first qubit of $\mathcal{W} \otimes \mathcal{A}$ being in the state $|1\rangle$, i.e. passing the verification procedure. The state $|\mathbf{0}\rangle = |00\ldots 0\rangle$ is the all-zeros state on $\mathcal{A}$.

Observe that $V, \mathcal{W}, \mathcal{A}, n_a, n_w$ and P depend on x; however, to avoid unnecessarily complicated notation, we do not indicate this explicitly.

Remark 6.2. It is conventional to define QMA $=$ QMA$(\frac{2}{3}, \frac{1}{3})$. However, the complexity class QMA(a, b) is highly insensitive to the particular values of a and b. In fact, even if a and b are functions of the problem size n, it remains true that QMA$(a(n), b(n))$ $=$ QMA provided $a(n) - b(n) \geqslant \frac{1}{p(n)}$ for some polynomial p. It is always possible to achieve that $a = 1 - \varepsilon$ and $b = \varepsilon$ by increasing the size of the circuit by a factor polylog$(1/\varepsilon)$ and increasing n_a by polylog$(1/\varepsilon)$ qubits, with no change in n_w [16, 17].

6.3 Quantum non-expander is in QMA

We now show that the problem defined in Definition 6.2 is in QMA. We first consider the YES case. In this case, Merlin has to convince Arthur that there exists a traceless $2^m \times 2^m$ matrix A such that

$$\|\Phi(A)\|_F > \alpha \|A\|_F. \tag{6.5}$$

Since Φ acts linearly, we may assume without loss of generality that $\|A\|_F = 1$. Merlin cannot directly send the matrix A because it is an exponentially large matrix. Instead, he can send the quantum certificate

$$|\psi_A\rangle = \sum_{i,j=1}^{N} a_{ij} |i\rangle \otimes |j\rangle$$

encoding the matrix A, where $N = 2^m$ is the dimension of the Hilbert space on which $A = [a_{ij}]$ acts. We show that $|\psi_A\rangle$ can serve as a witness, making it possible to convince Arthur that the inequality in Eq. (6.5) holds.

Arthur's verification protocol makes use of the following facts:

$$\|A\|_F^2 = \langle \psi_A | \psi_A \rangle,$$

$$\mathrm{Tr}[A] = \sqrt{N} \langle \varphi | \psi_A \rangle,$$

where $|\varphi\rangle = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} |i\rangle \otimes |i\rangle$, and

$$\|\Phi(A)\|_F^2 = \langle \psi_A | W^\dagger W | \psi_A \rangle,$$

where

$$W = \frac{1}{D} \sum_{d=1}^{D} U_d \otimes \overline{U}_d$$

and $\overline{U}_d$ denotes the complex conjugate of U_d.

First, to check whether $\mathrm{Tr}[A] = 0$, Arthur verifies that $|\psi_A\rangle$ is orthogonal to $|\varphi\rangle$. Second, to estimate the contractive factor, Arthur estimates the expectation value $\langle \psi_A | W^\dagger W | \psi_A \rangle$ of $W^\dagger W$. For $d, e = 1, \ldots, D$, define the unitaries

$$V_{d,e} = (U_d^\dagger \otimes U_d^T)(U_e \otimes \overline{U}_e).$$

Note that $V_{d,e} = V_{e,d}^\dagger$ and $V_{d,d} = \mathbb{1}$. The expectation value can be expressed as

$$\langle \psi_A | W^\dagger W | \psi_A \rangle = \frac{1}{D^2} \sum_{d,e} \langle \psi_A | V_{d,e} | \psi_A \rangle = \frac{1}{D} + \frac{2}{D^2} \sum_{d<e} \mathrm{Re} \langle \psi_A | V_{d,e} | \psi_A \rangle.$$

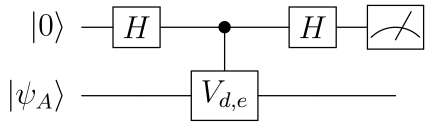


Figure 6-2: Hadamard test for $V_{d,e}$

Arthur can estimate the values $\mathrm{Re}\langle\psi_A|V_{d,e}|\psi_A\rangle$ using the Hadamard test [shown in Fig. (6-2)] since it will output 0 with probability $\Pr(0) = \frac{1}{2}(1 + \mathrm{Re}\langle\psi_A|V_{d,e}|\psi_A\rangle)$. From this Arthur can calculate $\langle\psi_A|W^\dagger W|\psi_A\rangle = \|\Phi(A)\|_F^2$ and ensure that it exceeds α^2.

Now consider the NO case. In this case, Arthur's first measurement above, if it passes, projects the state $|\psi_A\rangle$ onto the subspace orthogonal to $|\varphi\rangle$; and by definition, all states $|\psi_A\rangle$ in that subspace must satisfy

$$\langle\psi_A|W^\dagger W|\psi_A\rangle = \|\Phi(A)\|_F^2 \leqslant \beta^2.$$

This shows that Merlin cannot cheat, that is, make Arthur believe that there exists a quantum state with contractivity greater or equal to α, provided that Arthur estimates the expected value sufficiently well and with sufficiently high probability of confidence.

As in the original definition of QMA in [12], we may assume that Arthur has multiple copies of the quantum certificate $|\psi\rangle$ so that we can estimate the expected value sufficiently well. Using the powerful technique of in-place amplification [16], we can transform a quantum circuit requiring $|\psi\rangle^{\otimes k}$ into one that requires only a single copy of $|\psi\rangle$.

6.4 Some technical tools

6.4.1 The Frobenius norm

In the proof that QUANTUM NON-EXPANDER is QMA-hard, we will frequently make use of the Frobenius norm; we therefore present some useful facts about this norm here. If B is a matrix with entries b_{ij}, then the Frobenius norm is defined as

$$\|B\|_{\mathrm{F}} = \sqrt{\mathrm{Tr}[B^\dagger B]} = \sqrt{\sum_{ij} |b_{ij}|^2}. \tag{6.6}$$

We have the following identities: $\|A \otimes B\|_{\mathrm{F}} = \|A\|_{\mathrm{F}}\,\|B\|_{\mathrm{F}}$, $\mathrm{Tr}[A \otimes B] = \mathrm{Tr}[A]\,\mathrm{Tr}[B]$, and of course $\mathrm{Tr}[A + B] = \mathrm{Tr}[A] + \mathrm{Tr}[B]$. If $|\psi\rangle$ and $|\phi\rangle$ are pure states then

$$\Big\|\,|\psi\rangle\langle\phi|\,\Big\|_{\mathrm{F}} = \sqrt{\langle\psi|\psi\rangle\,\langle\phi|\phi\rangle} = \Big\||\psi\rangle\Big\|\Big\||\phi\rangle\Big\|. \tag{6.7}$$

Note that $\big\|\,|0\rangle\langle 0|\,\big\|_{\mathrm{F}} = \big\|\,|1\rangle\langle 1|\,\big\|_{\mathrm{F}} = 1$.

In this chapter we denote the Pauli matrices on one qubit by σ_i, with $\sigma_0 = \mathbb{1}$, $\sigma_1 = \sigma_x$, $\sigma_2 = \sigma_y$, and $\sigma_3 = \sigma_z$. Consider any traceless matrix A that acts on some space $\mathbb{C}^d \otimes \mathbb{C}^2$, where we will refer to the second subspace (i.e. single-qubit subspace) as the *indicator qubit register*. Because the Pauli matrices σ_i form a basis for the matrices acting on the indicator qubit register, we can decompose A as $\sum_{i=0}^{3} A_i \otimes \sigma_i$, where A_i are matrices on the multi-qubit

$\mathbb{C}^d$ subspace (the combined witness and ancilla registers that we will see later). Because σ_i are traceless for $i = 1, 2, 3$, the traceless condition on A therefore becomes $\mathrm{Tr}[A_0] = 0$. Moreover, because the Pauli matrices are orthogonal with respect to the trace inner product and all satisfy $\|\sigma_i\|_{\mathrm{F}}^2 = 2$, we have $\|\sum_i A_i \otimes \sigma_i\|_{\mathrm{F}}^2 = \sum_i \|A_i \otimes \sigma_i\|_{\mathrm{F}}^2 = 2\sum_i \|A_i\|_{\mathrm{F}}^2$, giving the inequality

$$\left\| \sum_{i=0}^{3} A_i \otimes \sigma_i \right\|_{\mathrm{F}} \geqslant \sqrt{2}\, \|A_0\|_{\mathrm{F}} \,. \tag{6.8}$$

A quantum operation G is called a *pinching operator* if $G(B) = \sum_P PBP$ where P are non-overlapping projectors with $\sum_P P = \mathbb{1}$. Pinching operators are trace preserving,

$$\mathrm{Tr}\left[\sum_P PBP\right] = \mathrm{Tr}[B] \,, \tag{6.9}$$

and moreover, (by the "pinching inequality") cannot increase Frobenius norm:

$$\left\| \sum_P PBP \right\|_{\mathrm{F}} \leqslant \|B\|_{\mathrm{F}} \,. \tag{6.10}$$

It should be noted that a quantum expander $\mathcal{E}$ is also norm-non-increasing,

$$\|\mathcal{E}(B)\|_{\mathrm{F}} \leqslant \|B\|_{\mathrm{F}} \,, \tag{6.11}$$

and similarly for any projector P,

$$\|PBP\|_{\mathrm{F}} \leqslant \|B\|_{\mathrm{F}} \,. \tag{6.12}$$

6.4.2 Controlled expanders

The remainder of our chapter will make repeated use of controlled expanders, which we introduce here. If U is a unitary gate, we use the notation ΛU to indicate a controlled-U operation.

Definition 6.4 (Controlled expander). Let $\mathcal{F}$ be a quantum expander with operation elements $\{U_i : i = 1 \ldots D\}$ so that $\mathcal{F}(B) = \frac{1}{D}\sum_{i=1}^{D} U_i B U_i^\dagger$. The controlled expander $\Lambda\mathcal{F}$ is defined to be the D-regular superoperator whose operation elements are the controlled unitaries $\{\Lambda U_i : i = 1 \ldots D\}$.

More explicitly, consider two registers, a control register and a target register, and suppose that an expander $\mathcal{F}$ acts on the target register as $\mathcal{F}(B) = \frac{1}{D}\sum_{i=1}^{D} U_i B U_i^\dagger$. Decompose the control register into two orthogonal subspaces, and let Q and P be projectors onto these two subspaces (so $Q + P = \mathbb{1}$ and $PQ = QP = 0$). Suppose that the controlled operations ΛU_i are to be applied when the control register is in the subspace corresponding to P; thus $\Lambda U_i = P \otimes U_i + Q \otimes \mathbb{1}$. Consider a matrix $A \otimes B$, where A and B act on the control and target registers, respectively. Then the controlled expander $\Lambda\mathcal{F}$, with operation elements

ΛU_i, acts on $A \otimes B$ as

$$\begin{aligned}
\Lambda\mathcal{F}(A \otimes B) &= \frac{1}{D}\sum_{i=1}^{D}\Big[(\Lambda U_i)(A \otimes B)(\Lambda U_i^\dagger)\Big] \\
&= \frac{1}{D}\sum_{i=1}^{D}\Big[(P \otimes U_i + Q \otimes \mathbb{1})(A \otimes B)(P \otimes U_i^\dagger + Q \otimes \mathbb{1})\Big] \\
&= \frac{1}{D}\sum_{i=1}^{D}\Big[PAP \otimes U_iBU_i^\dagger + PAQ \otimes U_iB + QAP \otimes BU_i^\dagger + QAQ \otimes B\Big] \\
&= PAP \otimes \frac{1}{D}\sum_i(U_iBU_i^\dagger) + PAQ \otimes \left(\frac{1}{D}\sum_i U_i\right)B \qquad (6.13) \\
&\qquad + QAP \otimes B\left(\frac{1}{D}\sum_i U_i^\dagger\right) + QAQ \otimes B.
\end{aligned}$$

Note that if we impose on $\mathcal{F}$ the requirement that

$$\sum_i U_i = 0 \qquad (6.14)$$

then we obtain

$$\Lambda\mathcal{F}(A \otimes B) = PAP \otimes \mathcal{F}(B) + QAQ \otimes B \qquad (6.15)$$

which is how we would naturally desire a controlled expander to act. Unfortunately, unlike Eq. (6.15), Eq. (6.13) has additional cross-terms whose elimination would greatly simplify our future analysis.

We will, however, freely assume that Eq. (6.14) is satisfied, justified by the following observation. If necessary, we may always increase the set of operation elements of $\mathcal{F}$ from $\{U_i : i = 1 \dots D\}$ to $\{U_i : i = 1 \dots D\} \cup \{-U_i : i = 1 \dots D\}$. Such a change has no effect on the original expander $\mathcal{F}$; the expander $\mathcal{F}(B) = \frac{1}{D}\sum(U_i\ B\ U_i^\dagger)$ is invariant under $U_i \leftrightarrow -U_i$, even though the controlled expander $\Lambda\mathcal{F}(B) = \frac{1}{D}\sum(\Lambda U_i\ B\ \Lambda U_i^\dagger)$ is not necessarily invariant under $U_i \leftrightarrow -U_i$. Thus, with only a factor of two overhead in the number of unitaries, we may satisfy the condition of Eq. (6.14), thereby eliminating the undesired cross-terms; as such, Eq. (6.15) may effectively be taken as the definition of a controlled expander.

A concrete example of a controlled expander – and one of particular importance in this chapter – is the *controlled complete depolarizer*. Throughout this chapter we use $\mathcal{D}$ to denote the complete depolarizing channel on a single qubit, which is normally defined to apply a unitary from $\{\mathbb{1}, X, Y, Z\}$ with uniform probability 1/4. To ensure that Eq. (6.14) is satisfied, we therefore define the effect of $\mathcal{D}$ on a matrix σ to be

$$\mathcal{D}(\sigma) = \frac{1}{8}\sum_W W\sigma W = \mathbb{1}\,\frac{\mathrm{Tr}[\sigma]}{2}$$

where the sum is over $W \in \{\mathbb{1}, X, Y, Z, -\mathbb{1}, -X, -Y, -Z\}$. Consequently, the controlled complete depolarizer, $\Lambda\mathcal{D}$, with a single-qubit target and (possibly multi-qubit) control projectors P (indicating apply $\mathcal{D}$) and Q (indicating do nothing), is the 8-regular superoperator

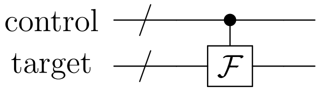


Figure 6-3: A controlled expander, $\Lambda\mathcal{F}$

with operation elements

$$\{\Lambda(\mathbb{1}), \Lambda(X), \Lambda(Y), \Lambda(Z), \Lambda(-\mathbb{1}), \Lambda(-X), \Lambda(-Y), \Lambda(-Z)\}$$

having the effect

$$\Lambda\mathcal{D}(A \otimes \sigma) = PAP \otimes \mathbb{1}\,\frac{\mathrm{Tr}[\sigma]}{2} + QAQ \otimes \sigma. \tag{6.16}$$

Although controlled expanders are not actually quantum gates, we will nevertheless include them in circuit diagrams. If $\Lambda\mathcal{F}(B) = \frac{1}{D}\sum_i(\Lambda U_i\; B\; \Lambda U_i^\dagger)$, then the circuit in Fig. 6-3 is to be interpreted as applying an element selected uniformly at random from the set $\{\Lambda U_i\}$ (or equivalently, as applying to the target register a unitary selected uniformly at random from the set $\{U_i\}$, but only if the control register is in the appropriate state). As a final remark, note that although a controlled expander is a unital map, it is not itself a good expander (firstly, because depending on the control qubit, the operator might not do anything at all, and secondly because even when the operator does act non-trivially, it only expands on the subspace of the target, not the entire space). For example, note that $|0\rangle\langle 0| \otimes |0\rangle\langle 0|$ is not contracted at all by the controlled complete depolarizer $\Lambda\mathcal{D}$, thus indicating that $\Lambda\mathcal{D}$ is not a good expander.

6.5 Quantum non-expander is QMA-hard

6.5.1 Outline of the proof

Let L be any language in QMA$(\frac{2}{3}, \frac{1}{3})$. We show that the quantum non-expander problem is QMA-hard by reducing L to a quantum non-expander problem. Specifically, let x be an $|x|$-bit problem instance whose inclusion in L, or lack-thereof, we wish to determine. Because $L \in$ QMA we have access to a verifier circuit satisfying Eqs. (6.3) and (6.4) acting on a witness space of $n_w = \mathrm{poly}(|x|)$ qubits and some ancilla space. For reasons that will become apparent later, we now use QMA amplification to give that $L \in$ QMA(a, b) for polynomially separated a and b where

$$a > 0.99 \quad \text{and} \quad b < (0.1)2^{-(n_w+1)}. \tag{6.17}$$

Note from Remark 6.2 that this can be done without increasing the size of the witness space of the verifier. Let the resulting QMA(a, b) verifier circuit be called V, which acts on the same witness space of $n_w = \mathrm{poly}(|x|)$ qubits and some ancilla space of $n_a = \mathrm{poly}(|x|)$ ancilla qubits. Merlin can provide Arthur a valid (with high probability) witness if and only if $x \in L$.

Let $\mathcal{E}$ be an explicit $\kappa_{\mathcal{E}}$-contracting expander of degree $D_{\mathcal{E}}$ acting on $n_w + n_a$ qubits, where

$$\kappa_{\mathcal{E}} \leqslant 0.1$$

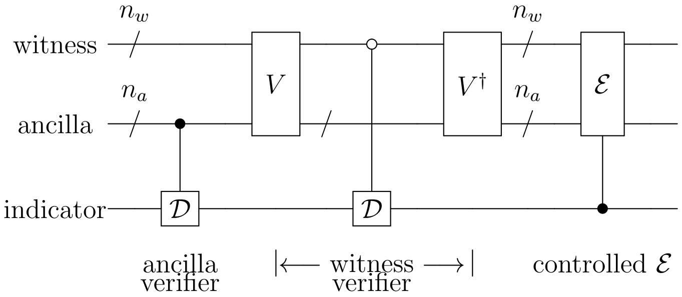


Figure 6-4: The map Φ constructed from the verifier circuit V, the complete depolarizer $\mathcal{D}$, and the $\kappa_{\mathcal{E}}$-contractive expander $\mathcal{E}$. The first controlled depolarizer is applied only if the ancillae are not all zero, and the second one only if the top output is zero. The controlled $\mathcal{E}$-expander is applied only if the bottom qubit is one. Note that this figure is not a true circuit because $\mathcal{D}$ and $\mathcal{E}$ are quantum expanders, not unitary gates.

and $D_{\mathcal{E}}$ is constant (independent of $|x|$). Such expanders are known to exist, as we outline in Appendix 6.A using Ref [18]. Using V and $\mathcal{E}$, we create a quantum expander Φ that is bad if $x \in L$ but good if $x \notin L$; indeed, we will present polynomially-separated (in fact, constant) α and β such that Φ is a β-contracting expander if $x \notin L$ but is not an α-contracting expander if $x \in L$. The ability to solve the QUANTUM NON-EXPANDER problem thereby allows us to solve L. The circuit for Φ is shown in Fig. 6-4, which we now describe in detail.

The map Φ acts on three registers, which, from top to bottom, are the witness register (of n_w qubits), the ancilla register (of n_a qubits), and an additional single-qubit register we call the *indicator qubit* register. The circuit is realized by composing the following three maps:

1. the ancilla verifier,
2. the witness verifier,
3. the controlled-$\mathcal{E}$.

The basic idea is that if $x \in L$ then Merlin can provide a valid witness and properly initialized ancillae that will pass the verifiers and not be mixed by the final controlled expander (indicating that our quantum expander is bad); conversely, if $x \notin L$ then no matter what witness and ancilla qubits Merlin provides, the indicator qubit will be depolarized and consequently his state will be well-mixed by the final controlled expander (indicating our expander to be good).

We now provide a detailed description of the three different maps and their purposes.

1. The ancilla verifier is the first operation in Fig. 6-4. It is the controlled expander $\Lambda_{\text{anc}}\mathcal{D}$, which applies the complete depolarizer $\mathcal{D}$ to the indicator qubit register only if any of the ancilla bits are 1 (i.e. if they are not all 0). More technically, it is

$$\Lambda_{\text{anc}}\mathcal{D}(B) = \frac{1}{8}\sum_{W} \Lambda_{\text{anc}}W \; B \; \Lambda_{\text{anc}}W^\dagger$$

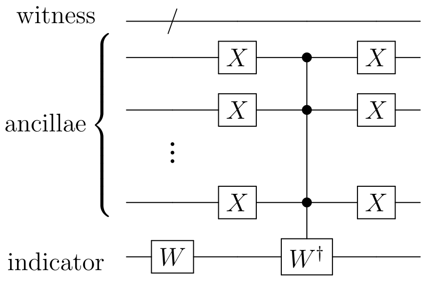


Figure 6-5: The controlled expander verifying the ancillae. The unitary W is selected from $\{\mathbb{1}, X, Y, Z, -\mathbb{1}, -X, -Y, -Z\}$ uniformly at random.

(with $W \in \{\mathbb{1}, X, Y, Z, -\mathbb{1}, -X, -Y, -Z\}$), where $\Lambda_{\text{anc}}W$ is the gate shown in Fig. 6-5. Note that $\Lambda_{\text{anc}}W$ requires a controlled-$W^\dagger$ gate controlled by n_a qubits, which can be implemented with ${n_a}^2$ gates using no extra work qubits [19]. (It is important that the implementation not require work qubits because we demand that there are no internal ancillae; our expander must be an expander on the entire space, not just a subspace.) Intuitively, if the ancilla qubits are not initialized to be all 0's, the verifier will depolarize the indicator qubit, whence the term *ancilla verifier.*

2. The witness verifier consists of the next three operations in Fig. 6-4. First, V operates on the witness and ancilla registers, with its output on the top qubit (with $|1\rangle$ signifying that the witness is valid, $|0\rangle$ signifying that it is invalid); the lower multi-qubit register on $n_w + n_a - 1$ qubits contains the rest of V's output (required by reversibility). A controlled-depolarizer then acts on the indicator qubit, conditioned upon the top qubit being $|0\rangle$ (i.e. failing the witness verification). The effects of V are then uncomputed with $V^\dagger$. At this point, intuitively, the indicator qubit has been depolarized if and only if the input failed either the ancilla verifier or the witness verifier (or both).

3. Finally, the last gate, which is the controlled expander $\Lambda_{\text{ind}}\mathcal{E}$, acts, conditioned on whether the indicator qubit is $|1\rangle$. Intuitively, if the input was $|\psi\rangle \otimes |\mathbf{0}\rangle \otimes |0\rangle$, with the indicator qubit initialized to $|0\rangle$, with the ancilla qubits initialized to $|\mathbf{0}\rangle = |00\dots0\rangle$, and with $|\psi\rangle$ a valid witness (for $x \in L$), then the indicator qubit will remain $|0\rangle$ and nothing will happen; if, on the other hand, the witness/ancillae failed any of the verifiers, thus depolarizing the indicator qubit to be $\frac{1}{2}\mathbb{1} = \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1|$, then $\mathcal{E}$ will act on the top registers, resulting in a highly mixed output (across all three registers).

Note that because $\mathcal{E}$ is an explicit $D_{\mathcal{E}}$-regular expander (where $D_{\mathcal{E}}$ is a constant), Φ, being the composition of two explicit 8-regular superoperators and $\Lambda\mathcal{E}$, is manifestly explicit and $64D_{\mathcal{E}}$-regular (i.e. of constant degree). We now proceed to show that Φ is indeed a good expander if $x \notin L$ (the NO case) but not if $x \in L$ (the YES case).

6.5.2 Analysis of NO case

First, consider the case in which $x \notin L$. We wish to show that Φ is a good expander, and therefore by Eq. (6.1), that it sufficiently decreases the Frobenius norm of any input traceless matrix. As discussed earlier, we may therefore take the input matrix to be $\sum_{i=0}^{3} A_i \otimes \sigma_i$ for some matrices A_i with $\mathrm{Tr}[A_0] = 0$, where σ_i are the Pauli matrices on the indicator qubit register.

Both the witness and ancilla verifiers are controlled depolarizers, and we can analyse each of them in the same way using projection operators that act on some subspace of the system; specifically, we will use $Q = \sum_{\phi \text{ passes}} |\phi\rangle\langle\phi|$ that projects onto the states that pass the verifier and $P = \sum_{\phi \text{ fails}} |\phi\rangle\langle\phi|$ that projects onto the states that fail it. For the ancilla verifier, these are $Q_a = |00\ldots0\rangle\langle00\ldots0|_{\text{anc}}$ (more properly written as $Q_a = \mathbb{1}_{\text{wit}} \otimes |00\ldots0\rangle\langle00\ldots0|_{\text{anc}} \otimes \mathbb{1}_{\text{ind}}$) and $P_a = \mathbb{1} - Q_a = \sum_{z\neq 00\ldots0} |z\rangle\langle z|_{\text{anc}}$. For the witness verifier, $Q_w = V^\dagger |1\rangle\langle1|_{\text{top}} V$ and $P_w = V^\dagger |0\rangle\langle0|_{\text{top}} V$ (so that $P_w + Q_w = \mathbb{1}$). Here the subscript *top* is used to indicate the top qubit register output from V.

Applying Eq. (6.16) and linearity, the effect of a verifier unit F on the input matrix $\sum_{i=0}^{3} A_i \otimes \sigma_i$ is therefore

$$\begin{aligned} F\left(\sum_{i=0}^{3} A_i \otimes \sigma_i\right) &= \sum_{i=0}^{3}\left[PA_iP \otimes \mathbb{1}\,\frac{\mathrm{Tr}[\sigma_i]}{2} + QA_iQ \otimes \sigma_i\right] \\ &= PA_0P \otimes \mathbb{1} + \sum_{i=0}^{3} QA_iQ \otimes \sigma_i. \end{aligned}$$

By linearity, it is easy to see that the effect of two such verifier units – the ancilla verifier with projectors $\{P_a, Q_a\}$ and witness verifier with projectors $\{P_w, Q_w\}$ – is

$$\begin{aligned} & F_w \circ F_a \left(\sum_{i=0}^{3} A_i \otimes \sigma_i\right) \\ = \; & F_w\left(P_aA_0P_a \otimes \mathbb{1}\right) + F_w\left(\sum_{i=0}^{3} Q_aA_iQ_a \otimes \sigma_i\right) \\ = \; & \left(P_wP_aA_0P_aP_w + Q_wP_aA_0P_aQ_w + P_wQ_aA_0Q_aP_w\right) \otimes \mathbb{1} + \sum_{i=0}^{3} Q_wQ_aA_iQ_aQ_w \otimes \sigma_i \\ = \; & \sum_{R} RA_0R^\dagger \otimes \mathbb{1} + \sum_{i=1}^{3} Q_{wa}A_iQ_{wa}^\dagger \otimes \sigma_i, \end{aligned}$$

where the first sum is over $R \in \{P_wP_a,\, P_wQ_a,\, Q_wP_a,\, Q_wQ_a\}$ and where Q_{wa} is the single product

$$Q_{wa} = Q_wQ_a$$

so $Q_{wa}^\dagger = Q_aQ_w$. Notice that the $i = 0$ term (involving $\sigma_0 = \mathbb{1}$) in the second sum has been transferred to the first sum, thereby allowing the first sum to include all possible projection combinations.

We can rewrite this as

$$F_w \circ F_a \left(\sum_{i=0}^{3} A_i \otimes \sigma_i \right) = C(A_0) \otimes \mathbb{1} + \sum_{i=1}^{3} Q_{wa} A_i Q_{wa}^\dagger \otimes \sigma_i \tag{6.18}$$

where

$$C(A_0) = \sum_R R A_0 R^\dagger = \sum_{R_w = P_w, Q_w} R_w \left(\sum_{R_a = P_a, Q_a} R_a A_0 R_a \right) R_w = (G_w \circ G_a)(A_0)$$

is the composition of the pinching operators $G_j(B) = P_j B P_j + Q_j B Q_j$ applied to A_0. Since C is the composition of pinching operators, Eqs. (6.9) and (6.10), along with Eq. (6.8), tell us that

$$\mathrm{Tr}[C(A_0)] = \mathrm{Tr}[A_0] = 0 \tag{6.19}$$

and

$$\|C(A_0)\|_{\mathrm{F}} \leqslant \|A_0\|_{\mathrm{F}} \leqslant \frac{1}{\sqrt{2}} \left\| \sum_i A_i \otimes \sigma_i \right\|_{\mathrm{F}} . \tag{6.20}$$

We are now ready to apply the final controlled expander, which by Eq. (6.15), with $P = |1\rangle\langle 1|$ and $Q = |0\rangle\langle 0|$, has the effect

$$\Lambda\mathcal{E}\,(B \otimes b) = \mathcal{E}(B) \otimes |1\rangle\langle 1|\, b\, |1\rangle\langle 1| + B \otimes |0\rangle\langle 0|\, b\, |0\rangle\langle 0| .$$

Applying this to the matrix Eq. (6.18), we conclude that the effect of the map in Fig. 6-4 on the initial traceless matrix $\sum_{i=0}^{3} A_i \otimes \sigma_i$ is

$$\begin{aligned} \Phi \left(\sum_{i=0}^{3} A_i \otimes \sigma_i \right) &= C(A_0) \otimes |0\rangle\langle 0| \;+\; \mathcal{E}\left(C(A_0)\right) \otimes |1\rangle\langle 1| \\ &\quad + Q_{wa} A_3 Q_{wa}^\dagger \otimes |0\rangle\langle 0| \;-\; \mathcal{E}(Q_{wa} A_3 Q_{wa}^\dagger) \otimes |1\rangle\langle 1| . \end{aligned} \tag{6.21}$$

To show that Φ is a good quantum expander, we must show that it sufficiently decreases the Frobenius norm of its traceless input. Since $\mathcal{E}$ is a $\kappa_{\mathcal{E}}$-contractive expander and $C(A_0)$ is traceless [see Eq. (6.19)] we are guaranteed that

$$\|\mathcal{E}\left(C(A_0)\right)\|_{\mathrm{F}} \leqslant \kappa_{\mathcal{E}} \|C(A_0)\|_{\mathrm{F}} . \tag{6.22}$$

Applying the triangle inequality to Eq. (6.21), and using Eqs. (6.22), (6.11), and (6.20), we therefore have

$$\begin{aligned} \left\| \Phi \left(\sum_{i=0}^{3} A_i \otimes \sigma_i \right) \right\|_{\mathrm{F}} &\leqslant \|C(A_0)\|_{\mathrm{F}} + \|\mathcal{E}\left(C(A_0)\right)\|_{\mathrm{F}} + \left\| Q_{wa} A_3 Q_{wa}^\dagger \right\|_{\mathrm{F}} + \left\| \mathcal{E}(Q_{wa} A_3 Q_{wa}^\dagger) \right\|_{\mathrm{F}} \\ &\leqslant (1 + \kappa_{\mathcal{E}}) \|C(A_0)\|_{\mathrm{F}} + 2 \left\| Q_{wa} A_3 Q_{wa}^\dagger \right\|_{\mathrm{F}} \\ &\leqslant \frac{1 + \kappa_{\mathcal{E}}}{\sqrt{2}} \left\| \sum_{i=0}^{3} A_i \otimes \sigma_i \right\|_{\mathrm{F}} + 2 \left\| Q_{wa} A_3 Q_{wa}^\dagger \right\|_{\mathrm{F}} . \end{aligned} \tag{6.23}$$

Note that we cannot make a claim similar to Eq. (6.22) for $\mathcal{E}(Q_{wa} A_3 Q_{wa}^\dagger)$ because $Q_{wa} A_3 Q_{wa}^\dagger$

need not be traceless.

In QMA$(1,0)$ we are guaranteed that provided the ancillae are initialized to be all 0's, no witness can pass the verifier (for a NO instance). Mathematically, this guarantee is equivalent to saying that $Q_{wa} = Q_w Q_a \equiv 0$. Consequently, the $Q_{wa} A_3 Q_{wa}^\dagger$ vanishes and we are done. In QMA(a,b), however, we must upper bound $\left\| Q_{wa} A_3 Q_{wa}^\dagger \right\|_{\mathrm{F}}$, which we now proceed to do.

Because $x \notin L \in$ QMA(a,b), we are assured that for any purported witness $|\psi\rangle$,

$$\| Q_w |\psi\rangle |\mathbf{0}\rangle \| \leqslant \sqrt{b}, \tag{6.24}$$

since Q_w projects onto the states that pass the verifier. Because Q_a projects onto the $|\mathbf{0}\rangle\langle\mathbf{0}|$ ancilla subspace, we may write

$$Q_a A_3 Q_a = \sum_{\psi_1, \psi_2} c(\psi_1, \psi_2)\, |\psi_1\rangle\langle\psi_2| \otimes |\mathbf{0}\rangle\langle\mathbf{0}|$$

where $\{|\psi_i\rangle\}$ is any orthonormal basis of the witness subspace. Note that because the witness register consists of n_w qubits, $c(\psi_1, \psi_2)$ can be regarded as a matrix with dimension $N = 2^{n_w} \times 2^{n_w}$. Thus using the triangle inequality and Eqs. (6.7) and (6.24),

$$\begin{aligned}
\left\| Q_{wa} A_3 Q_{wa}^\dagger \right\|_{\mathrm{F}} &= \left\| \sum_{\psi_1,\psi_2} c(\psi_1, \psi_2) Q_w |\psi_1\rangle|\mathbf{0}\rangle\langle\psi_2|\langle\mathbf{0}| Q_w \right\|_{\mathrm{F}} \\
&\leqslant \sum_{\psi_1,\psi_2} |c(\psi_1, \psi_2)| \left\| Q_w |\psi_1\rangle|\mathbf{0}\rangle\langle\psi_2|\langle\mathbf{0}| Q_w \right\|_{\mathrm{F}} \\
&= \sum_{\psi_1,\psi_2} |c(\psi_1, \psi_2)| \left\| Q_w |\psi_1\rangle|\mathbf{0}\rangle \right\|_{\mathrm{F}} \left\| Q_w |\psi_2\rangle|\mathbf{0}\rangle \right\|_{\mathrm{F}} \\
&\leqslant \sum_{\psi_1,\psi_2} |c(\psi_1, \psi_2)|\, b.
\end{aligned}$$

The matrix c has $(2^{n_w})^2$ elements, so its 1-norm and 2-norm are related by

$$\sum_{\psi_1,\psi_2} |c(\psi_1, \psi_2)| \leqslant 2^{n_w} \sqrt{\sum_{\psi_1,\psi_2} |c(\psi_1, \psi_2)|^2} = 2^{n_w} \left\| Q_a A_3 Q_a^\dagger \right\|_{\mathrm{F}}.$$

But by Eq. (6.12), and by the derivation of Eq. (6.8) applied to A_3, we have $\left\| Q_a A_3 Q_a^\dagger \right\|_{\mathrm{F}} \leqslant \|A_3\|_{\mathrm{F}} \leqslant \frac{1}{\sqrt{2}} \left\| \sum_{i=0}^3 A_i \otimes \sigma_i \right\|_{\mathrm{F}}$; thus we conclude,

$$\left\| Q_{wa} A_3 Q_{wa}^\dagger \right\|_{\mathrm{F}} \leqslant \frac{2^{n_w}}{\sqrt{2}} \left\| \sum_{i=0}^3 A_i \otimes \sigma_i \right\|_{\mathrm{F}} b. \tag{6.25}$$

Although 2^{n_w} is exponential in n_w, recall that b in Eq. (6.17) was chosen so that $2^{n_w+1} b \leqslant 0.1$. We conclude from Eqs. (6.23) and (6.25) that Φ is a β-contractive expander,

$$\left\| \Phi\left(\sum_{i=0}^3 A_i \otimes \sigma_i \right) \right\|_{\mathrm{F}} \leqslant \beta \left\| \sum_{i=0}^3 A_i \otimes \sigma_i \right\|_{\mathrm{F}}, \tag{6.26}$$

with

$$\beta = \frac{1+\kappa_{\mathcal{E}}+2^{n_w+1}b}{\sqrt{2}} < 0.85. \tag{6.27}$$

6.5.3 Analysis of YES case

Now consider the case in which $x \in L$. Since $L \in \mathrm{QMA}(a,b)$, there exists a valid witness $|\psi\rangle$ such that

$$\|Q_w|\psi\rangle|\mathbf{0}\rangle\|^2 \geqslant a. \tag{6.28}$$

From this witness we construct the density matrix $\Psi = |\psi\rangle\langle\psi| \otimes |\mathbf{0}\rangle\langle\mathbf{0}| \otimes |0\rangle\langle 0|$. Because Ψ passes the ancilla verifier unchanged and the witness verifier with very little change, Ψ is almost a fixed point of our expander Φ (and indeed, for QMA(1, 0) it is a fixed point); intuitively, therefore, Φ is a poor expander. The matrix $\tilde{I} = \frac{1}{2^{n_w+n_a+1}}\mathbb{1}$ is certainly a fixed point (for any unital map); therefore the traceless matrix

$$A = \Psi - \tilde{I} = |\psi\rangle\langle\psi| \otimes |\mathbf{0}\rangle\langle\mathbf{0}| \otimes |0\rangle\langle 0| - \frac{1}{2^{n_w+n_a+1}}\mathbb{1}$$

is also expected to change very little under Φ. By showing this to be the case, we will show that Φ is not an α-contractive expander for an α that is polynomially separated from the β found in the NO case.

Using an analysis similar to the previous case, it is easy to see that the effect of our circuit on Ψ is

$$\begin{aligned}
\Psi \quad & = \quad |\psi\rangle\langle\psi| \otimes |\mathbf{0}\rangle\langle\mathbf{0}| \otimes |0\rangle\langle 0| \\
& \xrightarrow{\text{Ancilla verifier}} \quad |\psi\rangle\langle\psi| \otimes |\mathbf{0}\rangle\langle\mathbf{0}| \otimes |0\rangle\langle 0| \\
& \xrightarrow{\text{Witness verifier}} \quad P_w\big(\,|\psi\rangle\langle\psi| \otimes |\mathbf{0}\rangle\langle\mathbf{0}|\,\big)P_w \otimes \frac{\mathbb{1}}{2} + Q_w\big(\,|\psi\rangle\langle\psi| \otimes |\mathbf{0}\rangle\langle\mathbf{0}|\,\big)Q_w \otimes |0\rangle\langle 0| \\
& \xrightarrow{\text{Controlled-}\mathcal{E}} \quad \frac{1}{2}\mathcal{E}\Big[P_w\big(\,|\psi\rangle\langle\psi| \otimes |\mathbf{0}\rangle\langle\mathbf{0}|\,\big)P_w\Big] \otimes |1\rangle\langle 1| \\
& \qquad\quad + \frac{1}{2}P_w\big(\,|\psi\rangle\langle\psi| \otimes |\mathbf{0}\rangle\langle\mathbf{0}|\,\big)P_w \otimes |0\rangle\langle 0| \\
& \qquad\quad + Q_w\big(\,|\psi\rangle\langle\psi| \otimes |\mathbf{0}\rangle\langle\mathbf{0}|\,\big)Q_w \otimes |0\rangle\langle 0|\,.
\end{aligned}$$

Note that the three final terms are mutually orthogonal because $|0\rangle\langle 0|1\rangle\langle 1| = 0$ and $P_wQ_w = 0$. Consequently, we have

$$\begin{aligned}
\|\Phi(\Psi)\|_{\mathrm{F}}^2 \quad & = \quad \frac{1}{4}\Big\|\mathcal{E}\Big[P_w\big(\,|\psi\rangle\langle\psi| \otimes |\mathbf{0}\rangle\langle\mathbf{0}|\,\big)P_w\Big]\Big\|_{\mathrm{F}}^2 \\
& \qquad + \frac{1}{4}\big\|P_w\big(\,|\psi\rangle\langle\psi| \otimes |\mathbf{0}\rangle\langle\mathbf{0}|\,\big)P_w\big\|_{\mathrm{F}}^2 \\
& \qquad + \big\|Q_w\big(\,|\psi\rangle\langle\psi| \otimes |\mathbf{0}\rangle\langle\mathbf{0}|\,\big)Q_w\big\|_{\mathrm{F}}^2 \\
& \geqslant \quad \big\|Q_w\big(\,|\psi\rangle\langle\psi| \otimes |\mathbf{0}\rangle\langle\mathbf{0}|\,\big)Q_w\big\|_{\mathrm{F}}^2 \\
& = \quad \|Q_w|\psi\rangle|\mathbf{0}\rangle\|^4 \\
& \geqslant \quad a^2
\end{aligned} \tag{6.29}$$

where we have used Eq. (6.7) and Eq. (6.28).

Now, because Ψ is a pure state density matrix, we use Eq. (6.6) to see that

$$\begin{aligned} \|A\|_{\mathrm{F}}^2 = \left\|\Psi - \tilde{I}\right\|_{\mathrm{F}}^2 &= \mathrm{Tr}\big[\Psi^2\big] + \mathrm{Tr}\Big[\tilde{I}^2\Big] - 2\mathrm{Tr}\Big[\Psi\tilde{I}\Big] \\ &= \mathrm{Tr}[\Psi] + \frac{\mathrm{Tr}[\mathbb{1}]}{\left(2^{n_w+n_a+1}\right)^2} - 2\frac{\mathrm{Tr}[\Psi]}{2^{n_w+n_a+1}} \\ &= 1 - \frac{1}{2^{n_w+n_a+1}}\,. \end{aligned} \tag{6.30}$$

Thus, using that Φ is linear and trace-preserving, that $\Phi\big(\tilde{I}\big) = \tilde{I}$, and Eqs. (6.29) and (6.30), we have

$$\begin{aligned} \|\Phi(A)\|_{\mathrm{F}}^2 &= \left\|\Phi(\Psi) - \Phi\big(\tilde{I}\big)\right\|_{\mathrm{F}}^2 \\ &= \mathrm{Tr}\Big[\Phi(\Psi)^\dagger\Phi(\Psi)\Big] + \mathrm{Tr}\Big[\tilde{I}^2\Big] - \mathrm{Tr}\Big[\Phi(\Psi)\tilde{I}\Big] - \mathrm{Tr}\Big[\Phi(\Psi)^\dagger\tilde{I}\Big] \\ &= \|\Phi(\Psi)\|_{\mathrm{F}}^2 + \mathrm{Tr}\Big[\tilde{I}^2\Big] - 2\mathrm{Tr}\Big[\Psi\tilde{I}\Big] \\ &\geqslant a^2 - \frac{1}{2^{n_w+n_a+1}} \\ &= \|A\|_{\mathrm{F}}^2 - (1-a^2) \\ &> \left[1 - \frac{8}{5}(1-a^2)\right]\|A\|_{\mathrm{F}}^2 \end{aligned}$$

where in the last inequality we have used from Eq. (6.30) that for $n_w \geqslant 1$ we have $\frac{5}{8} < \frac{3}{4} \leqslant \|A\|_{\mathrm{F}}^2 \leqslant 1$. Thus we conclude that Φ is not an α-contractive expander,

$$\|\Phi(A)\|_{\mathrm{F}} > \alpha\,\|A\|_{\mathrm{F}}\,, \tag{6.31}$$

with

$$\alpha = \sqrt{1 - \frac{8}{5}(1-a^2)} > 0.98, \tag{6.32}$$

the latter inequality coming from Eq. (6.17). Note that α and β are constants, and therefore certainly polynomially separated.

6.6 Conclusion

We have presented a new computational problem, QUANTUM NON-EXPANDER, and proved that it is QMA-complete. Consequently, the problem's complement, namely the QUANTUM EXPANDER problem for estimating how good a quantum expander is, is co-QMA-complete. This gives some insight into the computational complexity of estimating mixing rates of quantum channels and open quantum systems.

In contrast to the plethora of natural NP-complete problems, very few problems have been shown to be QMA-complete. We hope that it may be possible to find new QMA-complete problems using reductions from the QUANTUM NON-EXPANDER problem.

Chapter appendices

6.A Controlled expanders

In this appendix, we outline how we obtain the requisite controlled expander, $\Lambda\mathcal{E}$, needed for section 6.5. We use the results of Ben-Aroya, Schwartz, and Ta-Shma [18], whose Theorem 4.3 and 4.6 give the following result.

Theorem 6.1. There exists an integer D_0 such that for every $D > D_0$ and for every integer $t > 0$, there exists a explicit λ_t-contractive expander of degree D^2 on a space of dimension D^{8t}, where $\lambda_t \leqslant \lambda + c\lambda^2$ with c a constant and $\lambda = \frac{4\sqrt{D-1}}{D}$.

We will additionally use the following result, which follows directly from the definition of a quantum expander. Here we use the notation that $\mathcal{F}^r$ denotes the r-fold composition of $\mathcal{F}$.

Proposition 6.2. If $\mathcal{F}$ is a λ-contractive expander of degree D on a space of size N, then for any positive integer r, $\mathcal{F}^r$ is a λ^r-contractive expander of degree D^r on a space of size N.

In Sec. 6.5 we require a $\kappa_{\mathcal{E}}$-contractive expander $\mathcal{E}$ with $\kappa_{\mathcal{E}} \leqslant 0.1$ on a space of size $N = 2^{n_w+n_a}$. Note that N is actually allowed to exceed $2^{n_w+n_a}$ since we can always have extra input ancillae (artificially increasing n_a) that do nothing but are acted upon by the final controlled expander $\Lambda\mathcal{E}$. Fix D to be any power of 2 larger than D_0 and 15. Then $\lambda = \frac{4\sqrt{D-1}}{D} < 1$ is fixed. Let r be such that $(\lambda + c\lambda^2)^r \leqslant 0.1$. Let $t = \left\lceil \frac{n_w+n_a}{8\log_2 D} \right\rceil = \frac{n_w+n_a+n_{\text{extra}}}{8\log_2 D}$ for some $n_{\text{extra}} < 8\log_2 D$. Using the above theorem we are guaranteed the existence of a λ_t^r-contractive expander of degree D^{2r} on a space of size $D^{8t} = 2^{n_w+n_a+n_{\text{extra}}}$, where D and r are constants and $\lambda_t^r \leqslant 0.1$.

Chapter bibliography

[1] A. D. Bookatz, S. P. Jordan, Y.-K. Liu, and P. Wocjan. *Quantum nonexpander problem is quantum-Merlin-Arthur-complete.* Phys. Rev. A, 87(4):042317 [2013]. http://dx.doi.org/10.1103/PhysRevA.87.042317

[2] S. Hoory, N. Linial, and A. Wigderson. *Expander graphs and their applications.* Bull. Amer. Math. Soc., 43(4):439 [2006]. http://dx.doi.org/10.1090/S0273-0979-06-01126-8

[3] M. B. Hastings. *Entropy and entanglement in quantum ground states.* Phys. Rev. B, 76(3):035114 [2007]. http://dx.doi.org/10.1103/PhysRevB.76.035114

[4] A. Ben-Aroya and A. Ta-Shma. *Quantum expanders and the quantum entropy difference problem.* arXiv:quant-ph/0702129 [2007]. http://arxiv.org/abs/quant-ph/0702129

[5] A. Ben-Aroya, O. Schwartz, and A. Ta-Shma. *An explicit construction of quantum expanders.* arXiv:0709.0911 [2007]. http://arxiv.org/abs/0709.0911

[6] M. B. Hastings. *Random unitaries give quantum expanders.* Phys. Rev. A, 76(3):032315 [2007]. http://dx.doi.org/10.1103/PhysRevA.76.032315

[7] D. Gross and J. Eisert. *Quantum Margulis expanders.* Quantum Info. Comput., 8(8):722 [2008]. http://arxiv.org/abs/0710.0651

[8] A. W. Harrow. *Quantum expanders from any classical Cayley graph expander.* Quantum Info. Comput., 8(8):715 [2008]. http://arxiv.org/abs/0709.1142

[9] M. B. Hastings and A. W. Harrow. *Classical and quantum tensor product expanders.* Quantum Info. Comput., 9(3):336 [2009]. http://arxiv.org/abs/0804.0011

[10] F. G. S. L. Brandao, A. W. Harrow, and M. Horodecki. *Local random quantum circuits are approximate polynomial-designs.* arXiv:1208.0692 [2012]. http://arxiv.org/abs/1208.0692

[11] E. Knill. *Quantum randomness and nondeterminism.* arXiv:quant-ph/9610012 [1996]. http://arxiv.org/abs/quant-ph/9610012

[12] A. Y. Kitaev, A. H. Shen, and M. N. Vyalyi. *Classical and quantum computation.* American Mathematical Society, Boston, MA, USA [2002]

[13] J. Watrous. *Succinct quantum proofs for properties of finite groups.* In *41st Annual Symposium on Foundations of Computer Science, 2000. Proceedings*, pp. 537–546 [2000]. http://dx.doi.org/10.1109/SFCS.2000.892141

[14] A. D. Bookatz. *QMA-complete problems.* Quantum Info. Comput., 14(5&6):361 [2014]. http://arxiv.org/abs/1212.6312. See Chapter 7 of this thesis.

[15] H.-P. Breuer and F. Petruccione. *The theory of open quantum systems.* Oxford University Press, Oxford, UK [2007]. http://www.oxfordscholarship.com/view/10.1093/acprof:oso/9780199213900.001.0001/acprof-9780199213900

[16] C. Marriott and J. Watrous. *Quantum Arthur–Merlin games.* comput. complex., 14(2):122 [2005]. http://dx.doi.org/10.1007/s00037-005-0194-x

[17] D. Nagaj, P. Wocjan, and Y. Zhang. *Fast amplification of QMA.* Quantum Info. Comput., 9(11):1053 [2009]. http://arxiv.org/abs/0904.1549

[18] A. Ben-Aroya, O. Schwartz, and A. Ta-Shma. *Quantum expanders: motivation and constructions.* In *23rd Annual IEEE Conference on Computational Complexity, 2008. CCC '08*, pp. 292–303 [2008]. http://dx.doi.org/10.1109/CCC.2008.23

[19] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter. *Elementary gates for quantum computation.* Phys. Rev. A, 52(5):3457 [1995]. http://dx.doi.org/10.1103/PhysRevA.52.3457

Chapter 7

QMA-complete problems

In this chapter we give an overview of the quantum computational complexity class QMA and a description of known QMA-complete problems. Such problems are believed to be difficult to solve, even with a quantum computer, but have the property that if a purported solution to the problem is given, a quantum computer would easily be able to verify whether it is correct. An attempt has been made to make this chapter as self-contained as possible so that it can be accessible to computer scientists, physicists, mathematicians, and quantum chemists. Problems of interest to all of these professions can be found here.

This chapter is adapted from [5], which was published in 2014 and covers known QMA-complete problems up until August 2013. Several interesting QMA-completeness results have been made since that time. Although we do not categorize and discuss these recent results along with the older results, we include references to them at the end, in Sec. 7.5.

7.1 Introduction

7.1.1 Background

The class QMA (Quantum Merlin-Arthur) is the natural extension of the classical class NP (Nondeterministic Polynomial time) to the quantum computing setting. NP is an extremely important class in classical complexity theory, containing (by definition) those problems that have a short proof (or witness) that can be efficiently checked to verify that a valid solution to the problem exists. The class NP is of great importance because many interesting and important problems have this property – they may be difficult to solve, but given a solution, it is easy to verify that the solution is correct.

The probabilistic extension of NP is the class MA, standing for "Merlin-Arthur". Unlike in NP where witnesses must be verifiable with certainty, in MA valid witnesses need only be accepted with probability greater than 2/3 (and invalid witnesses rejected with probability greater than 2/3). MA can be thought of as the class of languages L for which a computationally-unbounded but untrustworthy prover, Merlin, can convince (with high probability) a verifier, Arthur (who is limited to polynomial-time computation), that a particular instance x is in L. Furthermore, when the instance x is not in L, the probability of Merlin successfully cheating must be low.

Because quantum computers are probabilistic by nature (as the outcome of a quantum measurement can generally be predicted only probabilistically), the *natural* quantum analogue of NP is actually the quantum analogue of MA, whence the quantum class QMA –

Quantum Merlin-Arthur[1]. QMA, then, is the class of languages for which small *quantum* witness states exist that enable one to prove, with high probability, whether a given string belongs to the language by whether the witness state causes an efficient *quantum verifier circuit* to output 1. It was first studied by Kitaev [34] and Knill [35]. A more precise definition will be given later.

The history of QMA takes its lead from the history of NP. In complexity theory, one of the most important results about NP was the first proof that it contains complete problems. A problem is NP-hard if, given the ability to solve it, one can also efficiently (that is, with only polynomial overhead) solve *any* other NP problem; in other words, a problem is NP-hard if any NP problem can be *reduced* to it. If, in addition to being NP-hard, a problem is itself in NP, it is called an NP-complete problem, and can be considered among the hardest of all the problems in NP. Two simple examples of NP-complete problems are BOOLEAN SATISFIABILITY (SAT) and CIRCUIT SATISFIABILITY (CSAT). The problem CSAT is to determine, given a Boolean circuit, whether there exists an input that will be evaluated by the circuit as true. The problem SAT is to determine, given a set of clauses containing Boolean variables, whether there is an assignment of those variables that will satisfy all of the clauses. If the clauses are restricted to containing at most k literals each, the problem is called k-SAT. One may also consider the problem MAX-SAT, which is to determine, given a set of clauses (of Boolean variables) and an integer m, whether at least m clauses can be simultaneously satisfied.

The fact that a complete problem exists for NP is actually trivial, as the problem of deciding whether there exists a (small) input that will be accepted (in polynomial time) by a given Turing machine is trivially NP-complete; rather, the importance of NP-completeness is due to the existence of interesting NP-complete problems. The famous Cook-Levin theorem, which pioneered the study of NP-completeness, states that SAT is NP-complete. A common way of proving this theorem is to first show that the above trivial NP-complete problem can be reduced to CSAT, and to then show that CSAT can be reduced to SAT.

The quantum analogue of CSAT is the QUANTUM CIRCUIT SATISFIABILITY problem (QCSAT), which is trivially QMA-complete (since QMA is defined in terms of quantum circuits). But QMA was found to have other, natural complete problems too. The most important of these, the k-LOCAL HAMILTONIAN problem, was defined by Kitaev [34], inspired by Feynman's ideas on Hamiltonian quantum computing [17]. This problem is a quantum analogue of MAX-SAT, in which Boolean variables are replaced by qubits and clauses are replaced by local Hamiltonians (which may be viewed as local constraints on the qubits); it is defined formally below in H-1. Just as the Cook-Levin theorem opened the study of NP-completeness by showing that SAT is NP-complete, so too the study of QMA-completeness began by showing that 5-LOCAL HAMILTONIAN is QMA-complete.

However, unlike NP, for which thousands of complete problems are known, there are currently relatively few known QMA-complete problems. In this chapter we will survey many, if not all, of them. This chapter divides the QMA-complete problems into three main groups and one subgroup:

- Quantum circuit/channel property verification (V)
- Hamiltonian ground state estimation (H)
  - Quantum satisfiability (S)

[1] Initially QMA was referred to as BQNP [34]; the name QMA was coined in [50].

- Density matrix consistency (C)

The letters in parentheses are used as labels to identify the group.

7.1.2 Formal definition of QMA

We now give a formal definition of QMA.

Definition 7.1. QMA is the set of all languages $L \subset \{0,1\}^*$ for which there exists a (uniform family of) quantum polynomial-time verifier circuit V such that for every $x \in \{0,1\}^*$ of length $n = |x|$,

if $x \in L$ then there exists a poly(n)-qubit witness state $|\psi_x\rangle$ such that $V\big(x, |\psi_x\rangle\big)$ accepts with probability $\geqslant 2/3$

if $x \notin L$ then for every purported poly(n)-qubit witness state $|\psi\rangle$, $V\big(x, |\psi\rangle\,\big)$ accepts with probability $\leqslant 1/3$.

Although the definition above used the numbers $2/3$ and $1/3$ (as is standard), we can generally define the class QMA(b,a): Given functions $a, b : \mathbb{N} \to (0,1)$ with $b(n) - a(n) \geqslant 1/\mathrm{poly}(n)$, a language is in QMA$(b,a)$ if $2/3$ and $1/3$ in the definition above are replaced by b and a, respectively. It is important to note that doing this does not change the class: QMA$(2/3, 1/3)$ = QMA$(1-\epsilon, \epsilon)$ provided that $\epsilon \geqslant 2^{-\mathrm{poly}(n)}$. Moreover, in going from QMA$(2/3,1/3)$ to QMA$(1-\epsilon,\epsilon)$, the amplification procedure can be carried out in such a way that the same witness is used, i.e. Merlin need only ever send a single copy of the witness state. [39]

When $b = 1$, i.e. when the witness must be verifiable with no error, the class is called QMA$_1$; thus QMA$_1$ = QMA$(1, 1/3)$ = QMA$(1, \epsilon)$. For the classical complexity class MA it is known that MA = MA$_1$, and also for the class QCMA, which is QMA restricted to classical witnesses, it has been shown [29] that QCMA=QCMA$_1$, but it is still an open question whether QMA=QMA$_1$. Several QMA$_1$-complete problems are presented in this chapter.

Furthermore, it should be noted that QMA actually consists of promise problems, meaning that when considering whether Merlin can truthfully convince Arthur or trick Arthur, we restrict our consideration to a subset of possible instances – we may assume that we are promised that our instance of consideration falls in this subset. In physical problems, this restriction could correspond to a limitation in the measurement precision available to us. With the above remarks, we can write the definition of QMA in a style matching that of the problem definitions provided later in this chapter.

Definition 7.2 (QMA). A promise problem $L = L_{\text{yes}} \cup L_{\text{no}} \subset \{0,1\}^*$ is in QMA if there exists a quantum polynomial-time verifier circuit V such that for every $x \in \{0,1\}^*$ of length $n = |x|$,

(yes case) if $x \in L_{\text{yes}}$ then $\exists$poly(n)-qubit state $|\psi_x\rangle$ such that $\Pr\Big[V\big(x, |\psi_x\rangle\big) \text{ accepts}\Big] \geqslant b$

(no case) if $x \in L_{\text{no}}$ then $\forall$poly(n)-qubit states $|\psi\rangle$, $\Pr\Big[V\big(x, |\psi\rangle\,\big) \text{ accepts }\Big] \leqslant a$

promised that one of these is the case (i.e. either x is in L_{yes} or L_{no}),
where $b - a \geqslant 1/\mathrm{poly}(n)$ and $0 < \epsilon < a < b < 1 - \epsilon$, with $\epsilon \geqslant 2^{-\mathrm{poly}(n)}$. If, instead, the above is true with $b = 1$, then L is in the class QMA$_1$.

Except for a glossary at the end, which provides the definitions of several basic reoccurring mathematical terms that appear in this work, the remainder of this chapter is devoted to listing known QMA-complete problems, along with their description and sometimes a brief discussion. When a problem is given matrices, vectors, or constants as inputs, it is assumed that they are given to precision of poly(n) bits. When a problem is given a unitary or quantum circuit, U_x, it is assumed that the problem is actually given a classical description x of the corresponding quantum circuit, which consists of poly($|x|$) elementary gates. Likewise, quantum channels are specified by efficient classical descriptions. We denote the identity matrix and identity channel by $\mathbb{1}$ and $\mathcal{I}$, respectively. For the advanced reader we note that, to date, most of the hardness reductions used in QMA-completeness proofs have been "Karp reductions"; several proofs, however, have relied on "Turing reductions", and when this is the case, we have endeavoured to note this accordingly.

This chapter has attempted to be as self-contained as possible, but for a more complete description and motivation of a problem, the reader is invited to consult the references provided. An attempt has been made to include as many currently known QMA-complete and QMA$_1$-complete problems as possible, but it is, of course, unlikely that this goal has been accomplished in full. The reader is invited to share other proven QMA-complete problems with the author for their inclusion in future versions of this work. Note that this chapter has restricted itself to QMA-complete and QMA$_1$-complete problems; it does not include other QMA-inspired classes, such as QMA(2) (when there are multiple unentangled Merlins) or QCMA (when the witness is classical).

7.2 Quantum circuit/channel property verification

V-1 QUANTUM CIRCUIT-SAT (QCSAT)

Problem: Given a quantum circuit V on n witness qubits and $m = \text{poly}(n)$ ancilla qubits,
determine whether:

(yes case) $\exists$ n-qubit state $|\psi\rangle$ such that $V(|\psi\rangle|0\ldots0\rangle)$ accepts with probability $\geqslant b$, i.e. outputs a state with $|1\rangle$ on the first qubit with amplitude-squared $\geqslant b$

(no case) $\forall$ n-qubit state $|\psi\rangle$, $V(|\psi\rangle|0\ldots0\rangle)$ accepts with probability $\leqslant a$,

promised one of these to be the case,
where $b - a \geqslant 1/\text{poly}(n)$ and $|0\ldots0\rangle$ is the all-zero m-qubit ancilla state.

This problem is QMA-complete immediately from the definition of QMA.

$\boxed{\text{V-2}}$ NON-IDENTITY CHECK

Problem: Given a unitary U implemented by a quantum circuit on n qubits, determine whether U is *not* close to a trivial unitary (the identity times a phase), i.e., determine whether:

(yes case) $\forall \phi \in [0, 2\pi), \left\|U - e^{i\phi}\mathbb{1}\right\| \geqslant b$

(no case) $\exists \phi \in [0, 2\pi)$ such that $\left\|U - e^{i\phi}\mathbb{1}\right\| \leqslant a$,

promised one of these to be the case,
where $b - a \geqslant 1/\text{poly}(n)$.

Theorem: QMA-complete [proven by Janzing, Wocjan, and Beth [26]]
Theorem: QMA-complete even for small-depth quantum circuits [proven by Ji and Wu [27]]
Hardness reduction from: QCSAT (V-1)

$\boxed{\text{V-3}}$ NON-EQUIVALENCE CHECK

This problem, a generalization of NON-IDENTITY CHECK (V-2), is to determine whether two quantum circuits (do not) define approximately the same unitary (up to phase) on some chosen invariant subspace. The subspace could, of course, be chosen to be the entire space, but in many cases one is interested in restricting their attention to a proper subspace, e.g. one defined by a quantum error-correcting code.

Problem: Given two unitaries, U_1 and U_2, implemented by a quantum circuit on n qubits, let $\mathcal{V}$ be a common invariant subspace of $(\mathbb{C}^2)^{\otimes n}$ specified by a quantum circuit V (that ascertains with certainty whether a given input is in $\mathcal{V}$ or not). The problem is to determine, given U_1, U_2, and V, whether the restrictions of U_1 and U_2 to $\mathcal{V}$ are not approximately equivalent, i.e.,
determine whether:

(yes case) $\exists\ |\psi\rangle \in \mathcal{V}$ such that $\forall \phi \in [0, 2\pi), \left\|(U_1 U_2^\dagger - e^{i\phi}\mathbb{1})\,|\psi\rangle\right\| \geqslant b$

(no case) $\exists\ \phi \in [0, 2\pi)$ such that $\forall |\psi\rangle \in \mathcal{V}, \left\|(U_1 U_2^\dagger - e^{i\phi}\mathbb{1})\,|\psi\rangle\right\| \leqslant a$,

promised one of these to be the case,
where $b - a \geqslant 1/\text{poly}(n)$.

Theorem: QMA-complete [proven by Janzing, Wocjan, and Beth [26]]
Hardness reduction from: NON-IDENTITY CHECK (V-2)

$\boxed{\text{V-4}}$ MIXED-STATE NON-IDENTITY CHECK

In this problem, either the given circuit acts like some unitary U that is far from the identity, or else it acts like the identity. This is very similar to NON-IDENTITY CHECK (V-2), but allows mixed-state circuit inputs. The diamond norm used here is defined in the glossary (Appendix 7.A).

Problem: Given a quantum circuit C on n-qubit density matrices, determine whether:

(yes case) $\|C - \mathcal{I}\|_\diamond \geqslant 2 - \epsilon$ and there is an efficiently implementable unitary U and state $|\psi\rangle$ such that $\|C(|\psi\rangle\langle\psi|) - U\,|\psi\rangle\langle\psi|\,U^\dagger\|_{\mathrm{tr}} \leqslant \epsilon$ and $\|U\,|\psi\rangle\langle\psi|\,U^\dagger - |\psi\rangle\langle\psi|\,\|_{\mathrm{tr}} \geqslant 2 - \epsilon$

(no case) $\|C - \mathcal{I}\|_\diamond \leqslant \epsilon$,

promised one of these to be the case, where $1 > \epsilon \geqslant 2^{-\mathrm{poly}(n)}$.

Theorem: QMA-complete [proven by Rosgen [46]]
Hardness reduction from: Quantum circuit testing (see Appendix 7.B) (X-1)

V-5 NON-ISOMETRY TESTING

Preliminary information:
This problem tests to see if a quantum channel is not almost a linear isometry (given a mixed-state quantum circuit description of the channel).

Definition 7.3 (isometry). A *linear isometry* is a linear map $U : \mathcal{H}_1 \to \mathcal{H}_2$ that preserves inner products, i.e. $U^\dagger U = \mathbb{1}_{\mathcal{H}_1}$.

Note that this is more general than a unitary operator, as $\mathcal{H}_1$ and $\mathcal{H}_2$ may have different sizes and U need not be surjective. Practically speaking, isometries are the operations involving unitaries that have access to fixed ancillae (say, ancillae starting in the $|0\rangle$ state). This problem asks how far from an isometry the input is, so it requires a notion of approximate isometries. A characterizing property of isometries is that they map pure states to pure states, even in the presence of a reference system; therefore, the notion of an approximate isometry is defined in terms of how mixed the output of a channel is in the presence of a reference system.

Definition 7.4 (ϵ-isometry). A quantum channel Φ that is a linear transformation from $\mathcal{H}_1$ to $\mathcal{H}_2$ is an *ϵ-isometry* if $\forall|\psi\rangle \in \mathcal{H}_1 \otimes \mathcal{H}_1$, we have $\|(\Phi \otimes \mathcal{I}_{\mathcal{H}_1})(|\psi\rangle\langle\psi|)\| \geqslant 1 - \epsilon$. i.e. it maps pure states (in a combined input and reference system) to almost-pure states. The norm appearing in this definition is the operator norm[2].

Problem: Given a quantum channel Φ that takes density matrices of $\mathcal{H}_1$ to density matrices of $\mathcal{H}_2$,
determine whether:

(yes case) Φ is not an ϵ-isometry, i.e. $\exists|\psi\rangle$ such that $\|(\Phi \otimes \mathcal{I}_{\mathcal{H}_1})(|\psi\rangle\langle\psi|)\| \leqslant \epsilon$

(no case) Φ is an ϵ-isometry, i.e. $\forall|\psi\rangle$, $\|(\Phi \otimes \mathcal{I}_{\mathcal{H}_1})(|\psi\rangle\langle\psi|)\| \geqslant 1 - \epsilon$,

promised one of these to be the case.

Theorem: QMA-complete when $0 < \epsilon < 1/19$ [proven by Rosgen [45, 46]]

Hardness reduction from: QCSAT (V-1)

[2] definition provided in the glossary (Appendix 7.A)

V-6 DETECTING INSECURE QUANTUM ENCRYPTION

In this problem, we wish to determine whether the given purported encryption channel E is insecure on a large subspace (for any key), or is close to being perfectly secure. The diamond norm used here is defined in the glossary (Appendix 7.A).

Preliminary information:
A private channel is a quantum channel with a classical key such that the input state cannot be determined from the output state without the key. Formally, it is defined as follows.

Definition 7.5 (ϵ-private channel). Suppose E is a channel taking as input an integer $k \in \{1, \ldots, K\}$ and a quantum state in space $\mathcal{H}_1$, and producing a quantum output in space $\mathcal{H}_2$, with dim $\mathcal{H}_1 \leqslant$ dim $\mathcal{H}_2$. Let E_k be the quantum channel where the integer input is fixed as k. Let Ω be the completely depolarizing channel that maps all density matrices to the maximally mixed state. E is an *ϵ-private channel* if $\|\frac{1}{K}\sum_k E_k - \Omega\|_\diamondsuit \leqslant \epsilon$ (so if the key k is not known, the output of E gives almost no information about the input) and there is a polysize decryption channel D such that $\forall k, \|D_k \circ E_k - \mathcal{I}\|_\diamondsuit \leqslant \epsilon$ (i.e. if k is known, the output can be reversed to obtain the input).

Problem: Let $\delta \in (0, 1]$. Given a circuit for E, which upon input k implements channel E_k acting from space $\mathcal{H}_1$ to $\mathcal{H}_2$ (with $\dim \mathcal{H}_1 \leqslant \dim \mathcal{H}_2$), determine whether:

(yes case) $\exists$ subspace S, with $\dim S \geqslant (\dim \mathcal{H}_1)^{1-\delta}$, such that for any k and any reference space $\mathcal{R}$, if ρ is a density matrix on $S \otimes \mathcal{R}$ then $\|(E_k \otimes \mathcal{I}_R)(\rho) - \rho\|_{\text{tr}} \leqslant \epsilon$

(no case) E is an ϵ-private channel,

promised one of these to be the case,
where $1 > \epsilon \geqslant 2^{-\text{poly}}$.

Theorem: QMA-complete for $0 < \epsilon < 1/8$ [proven by Rosgen [46]]
Hardness reduction from: Quantum circuit testing (see Appendix 7.B) (X-1)

Notes: In this problem, channels are given as mixed-state circuits.

V-7 QUANTUM CLIQUE

This is the quantum analogue of the NP-complete problem LARGEST INDEPENDENT SET on a graph G, which asks for the size of the largest set of vertices in which no two vertices are adjacent. According to the analogy, the graph G becomes a channel, and two inputs are 'adjacent' if they can be confused after passing through the channel, i.e. if there is an output state that could have come from either of the two input states. In this quantum QMA-complete problem, the channel is a quantum entanglement-breaking channel Φ and the problem is to find the size of the largest set of input states that cannot be confused after passing through the channel, that is, to determine if there are k inputs $\rho_1, \ldots, \rho_k$ such that $\Phi(\rho_1), \ldots, \Phi(\rho_k)$ are (almost) orthogonal under the trace inner product. Regarding the name, note that the NP-complete problems

LARGEST INDEPENDENT SET and LARGEST CLIQUE (which asks for the largest set of vertices, all of which are adjacent) are essentially the same: a set of vertices is an independent set on a graph G if and only if it is a clique on the complement of G.

Preliminary information:
Let S be the SWAP gate, so $S|\psi\rangle \otimes |\phi\rangle = |\phi\rangle \otimes |\psi\rangle$. Note that $\mathrm{Tr}(\sigma_1\sigma_2) = \mathrm{Tr}(S\,\sigma_1 \otimes \sigma_2)$ for all density matrices σ_1 and σ_2, so the right hand side can be used to evaluate the trace inner product (and therefore determine orthogonality) of σ_1 with σ_2. For any density matrix ρ on k registers, let ρ^i denote the result of tracing out all but the ith register of ρ. Similarly, define $\rho^{i,j} = \mathrm{Tr}_{\{1,\ldots k\}\setminus\{i,j\}}(\rho)$.

Definition 7.6 (entanglement-breaking channel; q-c channel). A quantum channel Φ is *entanglement-breaking* if there are POVM (Hermitian, positive-semidefinite operators that sum to the identity) $\{M_i\}$ and states σ_i such that $\Phi(\chi) = \sum_i \mathrm{Tr}(M_i\chi)\sigma_i$. In this case it is a fact that $\Phi^{\otimes 2}(\rho^{1,2})$ is always a separable state. If the σ_i in the above definition can be chosen to be $\sigma_i = |i\rangle\langle i|$, where $|i\rangle$ are orthogonal states, then Φ is called a *quantum-classical channel* (q-c channel).

Problem: Given an integer k and a quantum entanglement-breaking channel Φ acting on n-qubit states,
determine whether:

(yes case) $\exists\ \rho_1 \otimes \cdots \otimes \rho_k$ such that $\sum_{i,j} \mathrm{Tr}(S\Phi(\rho_i) \otimes \Phi(\rho_j)) \leqslant a$

(no case) $\forall$ k-register state ρ, $\sum_{i,j} \mathrm{Tr}(S\Phi^{\otimes 2}(\rho^{i,j})) \geqslant b$,

promised one of these to be the case,
where b and a are inverse-polynomially separated.

There are two theorems associated with this problem.

(a) Theorem: QMA-complete [proven by Beigi and Shor [3]]

(b) Theorem: QMA_1-complete when $a = 0$ and Φ is further restricted to q-c channels [proven by Beigi and Shor [3]]

Hardness reduction from: k-LOCAL HAMILTONIAN (H-1)
Classical analogue: LARGEST INDEPENDENT SET is NP-complete.

V-8 QUANTUM NON-EXPANDER

A quantum expander is a superoperator that rapidly takes density matrices towards the maximally mixed state. The QUANTUM NON-EXPANDER problem is to check whether a given superoperator is *not* a good quantum expander. This problem uses the Frobenius norm[3].

Preliminary information:
A density matrix can always be written as $\rho = I + A$, where I is the maximally mixed state and A is traceless. A quantum expander is linear (and unital), so $\Phi(\rho) = I + \Phi(A)$,

[3] definition provided in the glossary (Appendix 7.A)

which differs from I by $\Phi(A)$. Thus a good quantum expander rapidly kills traceless matrices. We have the following formal definition.

Definition 7.7 (quantum expander). Let Φ be a superoperator acting on n-qubit density matrices and obeying $\Phi(\rho) = \frac{1}{D}\sum_d U_d \rho U_d^\dagger$ where $\{U_d : d = 1, \dots D\}$ is a collection of $D = \mathrm{poly}(n)$ efficiently-implementable unitary operators. Φ is a *κ-contractive quantum expander* if $\forall\ 2^n \times 2^n$ traceless matrix A, $\|\Phi(A)\|_F \leqslant \kappa \|A\|_F$.

Problem: Given a superoperator Φ that can be written in the form appearing in the above definition,
determine whether:

(yes case) Φ is not a b-contractive quantum expander

(no case) Φ is an a-contractive quantum expander,

promised one of these to be the case,
where $b - a \geqslant 1/\mathrm{poly}(n)$.

Theorem: QMA-complete [proven by Bookatz, Jordan, Liu, and Wocjan [6]]
Hardness reduction from: QCSAT (V-1)

7.3 Hamiltonian ground-state energy estimation

H-1 ***k*-LOCAL HAMILTONIAN**

This is the problem of estimating the ground-state energy[4] of a Hamiltonian in which all interactions are k-local, that is, they only ever involve at most k particles at a time. Formally, H is a k-local Hamiltonian if $H = \sum_i H_i$ where each H_i is a Hermitian operator acting (non-trivially) on at most k qubits. In addition to restricting the locality of a Hamiltonian in terms of the number of qubits on which it acts, one can also consider geometric restrictions on the Hamiltonian. Indeed, one can imagine a 2-local Hamiltonian in which interactions can only occur between neighbouring sites, e.g. $H = \sum_{i=1}^{n-1} H_{i,i+1}$ where each $H_{i,i+1}$ acts non-trivially only on particles i and $i+1$ arranged on a line. The results of these considerations will also be mentioned below. Note that all these problems use the operator norm[4].

Problem: Given a k-local Hamiltonian on n qubits, $H = \sum_{i=1}^{r} H_i$, where $r = \mathrm{poly}(n)$ and each H_i acts non-trivially on at most k qubits and has bounded operator norm $\|H_i\| \leqslant \mathrm{poly}(n)$,
determine whether:

(yes case) H has an eigenvalue less than a

(no case) all of the eigenvalues of H are larger than b,

promised one of these to be the case,
where $b - a \geqslant 1/\mathrm{poly}(n)$.

Theorem: QMA-complete for $k \geqslant 2$ [proven by Kempe, Kitaev, and Regev [33]]
Hardness reduction from: QCSAT (V-1)

[4] see the glossary (Appendix 7.A) for a very brief definition of these terms

Additionally, it has been proved that it is:

(a) Theorem: QMA-complete when $k = O(\log n)$ (still provided $k \geqslant 2$) [proven by Kitaev [34]]

(b) Theorem: QMA-complete even when $k = 3$ with constant norms, i.e. $\|H_i\| = O(1)$ [proven by Nagaj [42]]

(c) Theorem: QMA-complete even when 2-local on a line of 8-dimensional qudits[5], i.e. when the qudits are arranged on a line and only nearest-neighbour interactions are present [proven by Hallgren, Nagaj, and Narayanaswami[6] [24]]

(d) Theorem: QMA-complete even when 2-local on a 2-D lattice [proven by Oliveira and Terhal [43]]

(e) Theorem: QMA-complete even for interacting bosons under two-body interactions [proven by Wei, Mosca, and Nayak [51]]

(f) Theorem: QMA-complete even for interacting fermions under two-body interactions [proven by Whitfield, Love, and Aspuru-Guzik [52]]

(g) Theorem: QMA-complete even when restricted to real 2-local Hamiltonians [proven by Biamonte and Love [4]]

(h) Theorem: QMA-complete even for stochastic[5] Hamiltonians (i.e. symmetric Markov matrices) when $k \geqslant 3$ [proven by Jordan, Gosset, and Love [28]]

Notes: For $k = 1$, the 1-LOCAL HAMILTONIAN is in P [33].

Many other simple modifications of k-LOCAL HAMILTONIAN are also QMA-complete. For example[7], QMA-completeness is not changed when restricting to dense k-local Hamiltonians, i.e. for a negative-semidefinite Hamiltonian when the ground energy is (in absolute value) $\Omega(n^k)$.

Classical analogue: MAX-k-SAT is NP-complete for $k \geqslant 2$.
This problem may easily be rephrased in terms of satisfying constraints imposed by the H_i terms. The “yes” case corresponds to the existence of a state that violates, in expectation value, only fewer than a weighted-constraints; the “no” case, to all states violating, in expectation value, at least b weighted-constraints. This problem can therefore be viewed as estimating the largest number of simultaneously satisfiable constraints, whence the analogy to MAX-SAT.

[5] definition provided in the glossary (Appendix 7.A)

[6] improving the work by Aharonov, Gottesman, Irani, and Kempe [1] who showed this for 12-dimensional qudits

[7] This result is from [18], who actually defined their problem for finding the highest energy of a positive-semidefinite Hamiltonian. Their interest lay in finding approximation algorithms for this problem.

$\boxed{\text{H-2}}$ EXCITED k-LOCAL HAMILTONIAN

We have seen that estimating the ground-state energy of a Hamiltonian is QMA-complete. The current problem shows that estimating the low-lying excited energies of a Hamiltonian is QMA-complete; specifically, estimating the c^{th} energy eigenvalue of a k-local Hamiltonian is QMA-complete when $c = O(1)$.

> Problem: Given a k-local Hamiltonian H on n qubits,
> determine whether:
>
> (yes case) the c^{th} eigenvalue of H is $\leqslant a$
>
> (no case) the c^{th} eigenvalue of H is $\geqslant b$,
>
> promised one of these to be the case,
> where $b - a \geqslant 1/\text{poly}(n)$.

Theorem: QMA-complete for $c = O(1)$ and $k \geqslant 3$ [proven by Jordan, Gosset, and Love [28]]
Hardness reduction from: 2-LOCAL HAMILTONIAN (H-1)

$\boxed{\text{H-3}}$ HIGHEST ENERGY OF A k-LOCAL STOQUASTIC HAMILTONIAN

Problem H-1h states that finding the lowest eigenvalue of a stochastic[8] Hamiltonian is QMA-complete. Since if H is a stochastic Hamiltonian then $-H$ is stoquastic[8], we also have QMA-completeness for the problem of estimating the largest energy of a stoquastic Hamiltonian.

> Problem: Given a k-local stoquastic Hamiltonian H on n qubits,
> determine whether:
>
> (yes case) H has an eigenvalue greater than b
>
> (no case) all of the eigenvalues of H are less than a,
>
> promised one of these to be the case,
> where $b - a \geqslant 1/\text{poly}(n)$.

Theorem: QMA-complete for $k \geqslant 3$ [proven by Jordan, Gosset, and Love [28]]
Hardness reduction from: k-LOCAL STOCHASTIC HAMILTONIAN (H-1h) which itself is from (H-5a)

Notes: k-LOCAL STOQUASTIC HAMILTONIAN, i.e. finding the lowest energy rather than the highest energy, is in AM. [8]

$\boxed{\text{H-4}}$ SEPARABLE k-LOCAL HAMILTONIAN

This problem is the k-LOCAL HAMILTONIAN problem with the extra restriction that the quantum state of interest be a separable state, i.e. the question is whether there is a *separable* state with energy less than a (or greater than b). Separable, here, is with respect to a given partition of the space into two sets, between which the state must not be entangled.

[8] definition provided in the glossary (Appendix 7.A)

Problem: Given the same input as described in the k-LOCAL HAMILTONIAN problem, as well as a partition of the qubits into disjoint sets $\mathcal{A}$ and $\mathcal{B}$,
determine whether:

(yes case) $\exists\,|\psi\rangle = |\psi\rangle_A \otimes |\psi\rangle_B$, with $|\psi\rangle_A \in \mathcal{A}$ and $|\psi\rangle_B \in \mathcal{B}$, such that $\langle\psi|H|\psi\rangle \leqslant a$

(no case) $\forall\,|\psi\rangle = |\psi\rangle_A \otimes |\psi\rangle_B$, with $|\psi\rangle_A \in \mathcal{A}$ and $|\psi\rangle_B \in \mathcal{B}$, $\langle\psi|H|\psi\rangle \geqslant b$,

promised one of these to be the case,
where $b - a \geqslant 1/\mathrm{poly}(n)$.

Theorem: QMA-complete [proven by Chailloux and Sattath [12]]
Hardness reduction from: k-LOCAL HAMILTONIAN (H-1)

Notes: Interestingly, although the QMA-hardness proof follows immediately from k-LOCAL HAMILTONIAN, the "in QMA" proof is non-trivial and relies on the LOCAL CONSISTENCY PROBLEM (C-1).

H-5 PHYSICALLY RELEVANT HAMILTONIANS

2-LOCAL HAMILTONIAN is also QMA-complete when the Hamiltonian is restricted to various physically-relevant forms. These Hamiltonians may serve as good models for phenomena found in nature, or may be relatively easy to physically implement.

We will not explain all of the relevant physics and quantum chemistry here. However, we use the following notations:

The Pauli matrices X, Y, and Z are denoted as

$$\sigma^x = X, \qquad \sigma^y = Y, \qquad \sigma^z = Z, \qquad \boldsymbol{\sigma} = (\sigma^x, \sigma^y, \sigma^z).$$

When particles are on a lattice, $\langle i, j\rangle$ denotes nearest neighbours on the lattice. An electron on a lattice is located at some lattice site i and can be either spin-up ($\uparrow$) or spin-down ($\downarrow$). The operators $a_{i,s}^\dagger$ and $a_{i,s}$ are the fermionic raising and lowering operators, respectively; they create and annihilate an electron of spin $s \in \{\uparrow, \downarrow\}$ at site i, respectively. The operator corresponding to the number of electrons of spin s at site i is $n_{i,s} = a_{i,s}^\dagger a_{i,s}$.

Note that proving the QMA-completeness of physical Hamiltonians is related to the goal of implementing adiabatic quantum computation: techniques used to prove that a Hamiltonian is QMA-complete are often also used to prove that it is universal for adiabatic quantum computation.

(a) The 2-local Hamiltonian

$$H_{ZZXX} = \sum_i h_i \sigma_i^z + \sum_i d_i \sigma_i^x + \sum_{i,j} J_{ij} \sigma_i^z \sigma_j^z + \sum_{i,j} K_{ij} \sigma_i^x \sigma_j^x$$

where coefficients d_i, h_i, K_{ij}, J_{ij} are real numbers.

This Hamiltonian represents a 2-local Ising model with 1-local transverse field and a tunable 2-local transverse $\sigma^x\sigma^x$ coupling. The $\sigma^x\sigma^x$ is realizable, e.g., using capacitive coupling of flux qubits or with polar molecules [4].

Theorem: QMA-complete [proven by Biamonte and Love [4]]
Hardness reduction from: 2-LOCAL REAL HAMILTONIAN (H-1g)
Classical analogue: When when $K_{ij} = d_i = 0$ we obtain the famous Ising (spin glass) model with a magnetic field, which is NP-complete on a planar graph [2].

(b) The 2-local Hamiltonian

$$H_{ZX} = \sum_i h_i \sigma_i^z + \sum_i d_i \sigma_i^x + \sum_{i<j} J_{ij} \sigma_i^z \sigma_j^x + \sum_{i<j} K_{ij} \sigma_i^x \sigma_j^z$$

where coefficients d_i, h_i, K_{ij}, J_{ij} are real numbers. The $\sigma^x \sigma^z$ is realizable using flux qubits [4].
Theorem: QMA-complete [proven by Biamonte and Love [4]]
Hardness reduction from: 2-LOCAL REAL HAMILTONIAN (H-1g)

(c) The 2D Heisenberg Hamiltonian with local magnetic fields
The 2D Heisenberg Hamiltonian is a model for spins on a 2-dimensional lattice in a magnetic system, and is often used to study phase transitions. It takes the form

$$H_{\text{Heis}} = J \sum_{\langle i,j \rangle} \boldsymbol{\sigma_i} \cdot \boldsymbol{\sigma_j} - \sum_i \boldsymbol{\sigma_i} \cdot \boldsymbol{B}_i .$$

Here, sums over i range over all sites i in the lattice, and $\langle i, j \rangle$ range over nearest-neighbouring sites. The local magnetic field at site i is denoted by $\boldsymbol{B_i}$, and the coupling-constant J is a real constant. Hamiltonians restricted to this form are QMA-complete both for $J > 0$ and for $J < 0$.
Theorem: QMA-complete [proven by Schuch and Verstraete [48]]
Hardness reduction from: 2-local 2D-lattice Hamiltonian (H-1d)

(d) The 2D Hubbard Hamiltonian with local magnetic fields
The 2D Hubbard model describes a system of fermions on a 2-dimensional lattice and is therefore used to model electrons in solid-state systems. It takes the form

$$H_{\text{Hubb}} = -t \sum_{\langle i,j \rangle, s} a_{i,s}^\dagger a_{j,s} + U \sum_i n_{i,\uparrow} n_{i,\downarrow} - \sum_i \boldsymbol{\bar{\sigma}_i} \cdot \boldsymbol{B}_i \ .$$

Here, sums over i range over all sites i in the lattice, $\langle i, j \rangle$ range over nearest-neighbouring sites, and s ranges over spins $\{\uparrow, \downarrow\}$. In this model, $\boldsymbol{\bar{\sigma}_i}$ is the Pauli matrices vector converted into orbital pair operators: $\boldsymbol{\bar{\sigma}_i} = \{\bar{\sigma}_i^x, \bar{\sigma}_i^y, \bar{\sigma}_i^z\}$ with $\bar{\sigma}_i^\alpha = \sum_{s,s'} \sigma_{ss'}^\alpha a_{i,s}^\dagger a_{i,s'}$ where $\sigma_{ss'}^\alpha$ denotes the (s, s') element of Pauli matrix σ^α. The local magnetic field at site i is denoted by $\boldsymbol{B_i}$, and U and t are positive numbers representing the electron-electron Coulomb repulsion and electron tunnelling rate, respectively.
Theorem: QMA-complete [proven by Schuch and Verstraete [48]]
Hardness reduction from: Heisenberg Hamiltonian (H-5c)

H-6 TRANSLATIONALLY INVARIANT k-LOCAL HAMILTONIAN

There has been some interest in studying the k-LOCAL HAMILTONIAN (H-1) problem with the added restriction that the Hamiltonian be *translationally invariant*, i.e. that the Hamiltonian be identical at each particle (qudit[9]) in the system. Such systems are generally in a one-dimensional geometry with periodic boundary conditions. Some problems additionally employ geometric locality (which we refer to here as being on a line), such as constraining interactions to be between nearest-neighbouring particles, or between nearby (but not necessarily nearest-neighbouring) particles; some problems do not, however, have such geometric locality constraints. Current results are listed here. These results are all built on Ref. [1]. Finally, note that there may be complications in discussing QMA-completeness, since if a Hamiltonian is local and translationally invariant, the only input that scales is the number of qudits, n; it may need to be assumed that n is given to the problem in unary to avoid these complications, but we will not discuss this here.

The k-LOCAL HAMILTONIAN problem is:

(a) Theorem: QMA-complete even with a translationally invariant 3-local Hamiltonian with 22-state qudits, but where the interactions are not necessarily geometrically local. [proven by Vollbrecht and Cirac [49]]

(b) Theorem: QMA-complete even for translationally invariant 2-local Hamiltonians on poly(n)-state qudits [proven by Kay [30]]

(c) Theorem: QMA-complete even for translationally invariant $O(\log n)$-local Hamiltonians on 7-state qudits, where the interactions are geometrically local (albeit not restricted to nearest-neighbours) [proven by Kay [30]]

(d) Theorem: QMA-complete even for 2-local Hamiltonians on a line of 49-state qudits where all strictly-2-local Hamiltonian terms are translationally invariant, although the 1-local terms can still be position-dependent [proven by Kay [31]]

Notes: Although not discussed here, similar results exist for rotationally invariant Hamiltonians [32].
There exist translationally invariant 2-local Hamiltonian problems on constant-dimensional qudits, where the interactions are only between nearest-neighbours (and in which the only input is the size of the system, provided in *binary*) that are QMA_{EXP}-complete, where QMA_{EXP} is the quantum analogue of the classical complexity class NEXP; see [20].

H-7 UNIVERSAL FUNCTIONAL OF DFT

Preliminary information:
In quantum chemistry, *density functional theory* (DFT) is a method for approximating the ground-state energy of an electron system (see [48] and the references therein). The

[9] definition provided in the glossary (Appendix 7.A)

Hamiltonian for a system of N electrons is $H = T^e + V^{ee} + V^e$ where the kinetic energy, electron-electron Coulomb potential, and local potential are given respectively by

$$
\begin{aligned}
T^e &= -\frac{1}{2}\sum_{i=1}^{N} \nabla_i^2 \\
V^{ee} &= \sum_{1 \leqslant i < j \leqslant N} \frac{\gamma}{|\boldsymbol{r}_i - \boldsymbol{r}_j|} \\
V^e &= \sum_{i=1}^{N} V(\boldsymbol{x}_i)
\end{aligned}
$$

where $\gamma > 0$, $\boldsymbol{r}_i$ is the position of the ith electron, $\boldsymbol{x}_i = (\boldsymbol{r}_i, s_i)$ is the position $(\boldsymbol{r}_i)$ of the ith electron together with its spin (s_i), and ∇^2 is the Laplacian operator.

The ground-state energy of a system of N electrons can be found by minimizing the energy over all N-electron densities $\rho^{(N)}(\boldsymbol{x})$, but it can also be given by minimizing over all single-electron probability distributions $n(\boldsymbol{x})$ as

$$E_0 = \min_n \Big(\mathrm{Tr}[V^e n(\boldsymbol{x})] + F[n(\boldsymbol{x})]\Big)$$

where the *universal functional of DFT* is

$$F[n(\boldsymbol{x})] = \min_{\rho^{(N)} \to n} \mathrm{Tr}\Big[(T^e + V^{ee})\rho^{(N)}(\boldsymbol{x})\Big] .$$

In the universal functional, the minimization is over all N-electron densities $\rho^{(N)}(\boldsymbol{x})$ that give rise to the reduced-density $n(\boldsymbol{x})$; therefore $F[n]$ gives the lowest energy of $T^e + V^{ee}$ consistent with n. The difficult part of DFT is approximating $F[n(\boldsymbol{x})]$, which is independent of the external potential V^e and is therefore universal for all systems.

Problem: Given an integer N, representing the number of electrons, and a one-electron probability density $n(\boldsymbol{x})$,
determine whether:

(yes case) $F[n(\boldsymbol{x})] \leqslant a$

(no case) $F[n(\boldsymbol{x})] \geqslant b$,

promised one of these to be the case,
where $b - a \geqslant 1/\mathrm{poly}(N)$ and the strength of the Hamiltonian is bounded by $\mathrm{poly}(N)$.

Theorem: QMA-complete [proven by Schuch and Verstraete [48, 52]]
Hardness reduction from: Hubbard model (H-5d) [Turing reduction]

7.3.1 Quantum satisfiability

The quantum satisfiability problem, QUANTUM k-SAT, is really just the k-LOCAL HAMILTONIAN problem restricted to projection operators. Nonetheless, it is included here as a subsection of its own due to its high level of interest and study. Note that occasionally people speak of the problem MAX-QUANTUM-k-SAT; this is just another name for the k-LOCAL HAMILTONIAN problem (H-1), and is therefore QMA-complete for $k \geqslant 2$. The problem

QUANTUM k-SAT, however, is different.

S-1 QUANTUM k-SAT

QUANTUM k-SAT is the quantum analogue of the classical problem k-SAT. It is actually simply the k-LOCAL HAMILTONIAN problem restricted to the case of k-local projector Hamiltonians[10]. In classical k-SAT, the objective is to determine whether there exists a bit-string (so each character in the string can be either 0 or 1) that satisfies (all of) a set of Boolean clauses, each of which only involves at most k bits of the string. In the quantum analogue, rather than Boolean clauses we have projection operators. A QUANTUM k-SAT instance has a solution if there is a quantum state that passes (i.e., is a 0-eigenvalue of) each projection operator.

We provide two equivalent definitions of this problem here. The first emphasizes QUANTUM k-SAT as a special case of k-LOCAL HAMILTONIAN, and the second emphasizes the similarity to classical k-SAT.

Problem: Given k-local projection operators $\{\Pi_1, \dots, \Pi_m\}$ on n qubits, where $m = \text{poly}(n)$, and letting $H = \sum_{i=1}^{m} \Pi_i$,
determine whether:

(yes case) H has an eigenvalue of precisely 0

(no case) all of the eigenvalues of H are larger than b,

promised one of these to be the case,
where $b \geqslant 1/\text{poly}(n)$.

Equivalently, we can define the problem as follows.

Problem: Given polynomially many k-local projection operators $\{\Pi_i\}$,
determine whether:

(yes case) $\exists\ |\psi\rangle$ such that $\Pi_i|\psi\rangle = 0\ \forall i$

(no case) $\forall\ |\psi\rangle, \sum_i \langle\psi|\Pi_i|\psi\rangle \geqslant \epsilon$ (i.e. the expected number of "clause violations" is $\geqslant \epsilon$),

promised one of these to be the case,
where $\epsilon \geqslant 1/\text{poly}(n)$.

Theorem: QMA$_1$-complete for $k \geqslant 3$ [proven by Gosset and Nagaj[11] [19]]
Hardness reduction from: QCSAT (V-1)

Notes: QUANTUM k-SAT is in P for $k = 2$. [7]

QUANTUM k-SAT is still QMA$_1$-complete if instead of demanding that Π_i be projectors, we demand they be positive-semidefinite operators with zero ground-state energies and constant norms [42].

Classical analogue: k-SAT is NP-complete for $k \geqslant 3$.

[10] definition provided in the glossary (Appendix 7.A)

[11] improving the results of Bravyi [7], which showed this for $k \geqslant 4$

S-2 QUANTUM $(d_1, d_2, \ldots, d_k)$-SAT

Quantum $(d_1, d_2, \ldots, d_k)$-sat is a quantum k-sat problem but with qudits rather than qubits. Specifically, in a quantum $(d_1, d_2, \ldots, d_k)$-sat instance, the system consists of n qudits (of various dimension), and each projection operator acts non-trivially on at most k of these n qudits, of the types specified, namely one d_1-dimensional qudit, one d_2-dimensional qudit,. . . , and one d_k-dimensional qudit. Bear in mind that, e.g., if $d_1 \geqslant d_2$ then a d_2-dimensional qudit is itself considered a type of d_1-dimensional qudit, so a projection operator that acts only on two d_2-dimensional qudits is also allowed. For example, an instance of quantum (5,3)-sat involves a system of n qudits (3-dimensional qudits called qutrits and 5-dimensional qudits called cinquits) such that each projection operator acts (non-trivially) on a single qudit, on one cinquit and one qutrit, or on two qutrits (but not on two cinquits). For purposes of notation, we assume that $d_1 \geqslant d_2 \geqslant \ldots \geqslant d_k$.

For $k \geqslant 3$, this class is trivial[12]: since quantum 3-sat is QMA$_1$-complete for qubits, it is certainly QMA$_1$-complete for qudits. The cases of $k = 2$ is not fully understood; however, the following results are known.

(a) quantum (5,3)-sat, i.e. with a cinquit and a qutrit
Theorem: QMA$_1$-complete for $k = 2$ with $d_1 \geqslant 5$, $d_2 \geqslant 3$ [proven by Eldar and Regev [16]]

(b) quantum (11,11)-sat on a (one-dimensional) line
Theorem: QMA$_1$-complete [proven by Nagaj [41]]

Notes: Quantum (2,2)-sat, i.e. quantum 2-sat, is in P.
Quantum (d_1, d_2)-sat when $d_1 < 5$ or $d_2 = 2$ are open questions. They are known to be NP-hard (except for $d_1 = d_2 = 2$ which is in P). [16]

Classical analogue: even though classical 2-sat is in P, classical (3,2)-sat, where one of the binary variables is replaced by a ternary variable, is NP-complete [16].

S-3 STOCHASTIC k-SAT

This problem is like quantum k-sat, except that instead of projection operators it uses stochastic, Hermitian, positive-semidefinite operators (see glossary, Appendix 7.A, for definitions).

Problem: Given polynomially many k-local stochastic, Hermitian, positive-semidefinite operators $\{H_1, \ldots, H_m\}$ on n-qubits with norms bounded by poly(n), determine whether:

(yes case) the lowest eigenvalue of $H = \sum_i H_i$ is 0

(no case) all eigenvalues of H are $\geqslant b$,

promised one of these to be the case,
where $b \geqslant 1/\text{poly}(n)$.

[12] Earlier work by Nagaj and Mozes [42] that quantum (3,2,2)-sat is QMA$_1$-complete is now subsumed by the result that quantum 3-sat is QMA$_1$-complete.

Theorem: QMA$_1$-complete for $k = 6$ [proven by Jordan, Gosset, and Love [28]]
Hardness reduction from: QUANTUM 4-SAT (S-1)

Notes: STOQUASTIC QUANTUM k-SAT, where the word 'stochastic' is replaced by 'stoquastic' above, is in MA and is MA-complete for $k \geqslant 6$. [9] Note that STOCHASTIC k-SAT makes no mention of projection operators, and therefore is not really a quantum k-SAT problem. In STOQUASTIC QUANTUM k-SAT, its MA-complete cousin, however, the operators can be converted to equivalent operators that are projectors, whence the relation to QUANTUM k-SAT. No connection to projectors is known in the stochastic case.

7.4 Density matrix consistency

C-1 k-**LOCAL DENSITY MATRIX CONSISTENCY**

Given a set of density matrices on subsystems of a constant number of qubits, this problem is to determine whether there is a global density matrix on the entire space that is consistent with the subsystem density matrices.

Problem: Consider a system of n qubits. Given $m = \text{poly}(n)$ k-local density matrices $\rho_1, \ldots, \rho_m$, so that each ρ_i acts only on a subset $C_i \subseteq \{1, \ldots n\}$ of qubits with $|C_i| \leqslant k$,
determine whether:

(yes case) $\exists$ n-qubit density matrix σ such that $\forall i$, $\|\rho_i - \tilde{\sigma}_i\|_{\text{tr}} = 0$ where $\tilde{\sigma}_i = \text{Tr}_{\{1,\ldots,n\}\smallsetminus C_i}(\sigma)$

(no case) $\forall$ n-qubit density matrix σ, $\exists i$ such that $\|\rho_i - \tilde{\sigma}_i\|_{\text{tr}} \geqslant b$ where $\tilde{\sigma}_i = \text{Tr}_{\{1,\ldots,n\}\smallsetminus C_i}(\sigma)$,

promised one of these to be the case,
where $b \geqslant 1/\text{poly}(n)$.

Theorem: QMA-complete even for $k = 2$ [proven by Liu [37]]
Hardness reduction from: k-LOCAL HAMILTONIAN (H-1) [Turing reduction]

Classical analogue: CONSISTENCY OF MARGINAL DISTRIBUTIONS is NP-hard.

C-2 N-**REPRESENTABILITY**

This is the same problem as 2-LOCAL DENSITY MATRIX CONSISTENCY (C-1), but specialized to fermions (particles whose quantum state must be antisymmetric under interchange of particles).

Problem: Given a system of N fermions and d possible modes, with $N \leqslant d \leqslant \text{poly}(N)$, and a $\frac{d(d-1)}{2} \times \frac{d(d-1)}{2}$ 2-fermion density matrix ρ,
determine whether:

(yes case) $\exists \binom{d}{N} \times \binom{d}{N}$ N-fermion density matrix σ such that $\text{Tr}_{\{3,\dots,N\}}(\sigma) = \rho$

(no case) $\forall$ N-fermion density matrices σ, $\|\rho - \text{Tr}_{\{3,\dots,N\}}(\sigma)\|_{\text{tr}} \geqslant b$,

promised one of these to be the case,
where $b \geqslant 1/\text{poly}(N)$.

Theorem: QMA-complete [proven by Liu, Christandl, and Verstraete [38]]
Hardness reduction from: 2-LOCAL HAMILTONIAN (H-1) [Turing reduction]

C-3 **BOSONIC N-REPRESENTABILITY**
This is the same problem as 2-LOCAL DENSITY MATRIX CONSISTENCY (C-1), but specialized to bosons (particles whose quantum state must be symmetric under interchange of particles).

Problem: Given a system of N bosons and d possible modes, with $d \geqslant cN$ (for some constant $c > 0$), and a $\frac{d(d+1)}{2} \times \frac{d(d+1)}{2}$ 2-boson density matrix ρ,
determine whether:

(yes case) $\exists \binom{N+d-1}{N} \times \binom{N+d-1}{N}$ N-boson density matrix σ such that $\text{Tr}_{\{3,\dots,N\}}(\sigma) = \rho$

(no case) $\forall$ N-boson density matrices σ, $\|\rho - \text{Tr}_{\{3,\dots,N\}}(\sigma)\|_{\text{tr}} \geqslant b$,

promised one of these to be the case,
where $b \geqslant 1/\text{poly}(N)$.

Theorem: QMA-complete [proven by Wei, Mosca, and Nayak [51]]
Hardness reduction from: 2-local Hamiltonian (H-1) [Turing reduction]

7.5 Afterword

Since the publication of the original survey [5], several QMA-complete problems and results have been discovered. We list these results here, albeit not in the same level as detail as the problems above.

Of particular interest, [15] have characterized the complexity of the k-LOCAL HAMILTONIAN problem with restricted Hamiltonian forms, including a determination of which are QMA-complete. This supplants much of H-5. Their result includes, for example, proofs of QMA-completeness for XYZ, XXZ, XY, and general Heisenberg Hamiltonian models (all without requiring the inclusion of 1-local terms). The work of [44] furthers some of these results, proving QMA-completeness for some antiferromagnetic versions (in particular, the antiferromagnetic Heisenberg and antiferromagnetic XY models) and various geometries (square and triangle lattices). Also of importance, it has been proven [13] that the Bose-Hubbard model is QMA-complete, even on simple graphs [14]. These two papers also prove the QMA-completeness of the special case of α-FRUSTRATION-FREE BOSE-HUBBARD HAMILTONIANS (even on simple graphs) [14], the MINIMUM GRAPH EIGENVALUE problem (i.e. the task of computing the smallest eigenvalue of sparse, symmetric 0-1 matrices) [13], and XY Hamiltonians (with different restrictions from that of [44] in terms of geometry, uniformity, and disallowing 1-local magnetic fields) [14]. Similar to H-5d, another proof of QMA-completeness for a two-dimensional interacting fermion model is provided in [10].

A number of QMA-complete problems have emerged related to the recent interest in proving or disproving a quantum PCP theorem. The work [11] claims the QMA-completeness of the 2-LOCAL HAMILTONIAN problem with $O(1)$ terms and an $O(1)$ promise gap, while [25] proves QMA-completeness for 2D local Hamiltonians with the restriction that the ground state satisfies an entanglement entropy area law, along with extensions to 3D cubic lattice Heisenberg and Hubbard Hamiltonians (again, satisfying an area law). In [21], the problems SET LOCAL HAMILTONIAN and CLASSICAL AND RESTRICTED-ENTANGLEMENT SWAPPING-PROVERS are defined and shown to be QMA-complete.

Several new QMA-complete problems that are not part of the k-LOCAL HAMILTONIAN family have also been found, including estimating the maximum acceptance probability when measuring stabilizer states [40], BASIS STATE CHECK ON SUBSET STATES [22], the one-way LOCC version of SEPARABLE ISOMETRY OUTPUT [23], and a variant of the CLOSE IMAGE TO TOTALLY MIXED problem [36]. While the above progress was all for QMA-completeness, the paper [47] considers a form of QUANTUM k-SAT that restricts the number of terms in which each qubit appears and proves that it is QMA$_1$-complete.

Chapter appendices

7.A Glossary

The definitions given here are not necessarily the most general or precise possible, but they suffice for the needs of this chapter.

$i^{\mathbf{th}}$ energy (level) of a Hamiltonian H – the i^{th} smallest eigenvalue of H.

ground-state energy of a Hamiltonian H – the smallest eigenvalue of H.

Hamiltonian – the generator of time-evolution in a quantum system. Its eigenvalues correspond to the allowable energies of the system. It also dictates what interactions are present in a system. As a matrix, it is Hermitian.

Hermitian matrix – a square matrix H that is equal to its own conjugate-transpose, i.e. $H^\dagger = H$.

norms of matrices – Several different matrix norms appear in this chapter. Given a matrix A with elements a_{ij}, the

operator norm of A is $\|A\| = \max\{\|A|\psi\rangle\|_2 : \||\psi\rangle\|_2 = 1\}$. For a square matrix, it is also known as the spectral norm; it is the largest singular value of A, and if A is normal, then it is the largest absolute value of the eigenvalues of A.

Frobenius norm of A is $\|A\|_F = \sqrt{\mathrm{Tr}[A^\dagger A]} = \sqrt{\sum_{i,j} |a_{ij}|^2}$.

trace norm of A is $\|A\|_{\mathrm{tr}} = \mathrm{Tr}\left[\sqrt{A^\dagger A}\right]$, which when A is normal is the sum of the absolute value of its eigenvalues. It is often written $\|A\|_{\mathrm{tr}} = \mathrm{Tr}|A|$ where $|A|$ denotes $\sqrt{A^\dagger A}$.

norms of quantum superoperators – Occasionally norms of superoperators are required in this chapter.

diamond norm of a superoperator Φ that acts on density matrices that act on a Hilbert space $\mathcal{H}$ is $\|\Phi\|_\diamond = \sup_X \|(\Phi \otimes \mathcal{I})(A)\|_{\mathrm{tr}} / \|A\|_{\mathrm{tr}}$ where the supremum is taken over all linear operators $A : \mathcal{H} \otimes \mathcal{H} \to \mathcal{H} \otimes \mathcal{H}$.

positive-semidefinite matrix – a Hermitian matrix whose eigenvalues are all non-negative.

k-local projector on n qubits – a Hermitian matrix of the form $\Pi = \mathbb{1}^{\otimes(n-k)} \otimes \sum_i |\psi\rangle\langle\psi|_i$ where the $|\psi\rangle_i$ are orthonormal k-qubit states. It satisfies $\Pi^2 = \Pi$.

stochastic matrix – a square matrix of non-negative real numbers such that each row sums to 1. If additionally each column sums to 1, it is called a doubly stochastic matrix.

stoquastic Hamiltonian – a Hamiltonian in which the off-diagonal matrix elements are non-positive real numbers in the standard basis.

qudit – generalization of a qubit: for some d, a d-state quantum-mechanical system, or mathematically, a unit-normalized vector in $\mathbb{C}^d$ (but where global phase is irrelevant). When $d = 2$ it is called a qubit, when $d = 3$ it is called a qutrit. When $d = 5$ it may be called a cinquit [16], but to avoid headaches, I advise against trying to name the $d = 4$ version.

7.B QMA-hard theorems

This appendix contains a theorem that allows one to prove that several quantum circuit verification problems are QMA-hard. Note that it does not prove QMA-completeness, only QMA-hardness, so it is relegated to the appendix.

X-1 QUANTUM CIRCUIT TESTING

This problem involves testing the behaviour of a quantum circuit. Given an input circuit C, one wishes to determine whether it acts like (on a large input space) a circuit from a uniform circuit family $\mathscr{C}_0$, or acts like (for all inputs) a circuit from uniform circuit family $\mathscr{C}_1$, promised that the two families are significantly different.

Problem: Let $\delta \in (0,1]$ and let $\mathscr{C}_0$ and $\mathscr{C}_1$ be two uniform families of quantum circuits. Given an input quantum circuit C acting on an n-qubit input space $\mathcal{H}$, let $C_0 \in \mathscr{C}_0$ and $C_1 \in \mathscr{C}_1$ act on the same input space $\mathcal{H}$. The problem is,
determine whether:

(yes case) $\exists$ subspace S, with $\dim S \geqslant (\dim \mathcal{H})^{1-\delta}$, such that for any reference space $\mathcal{R}$, if ρ is a density matrix on $S \otimes \mathcal{R}$ then $\|(C \otimes \mathcal{I}_{\mathcal{R}})(\rho) - (C_0 \otimes \mathcal{I}_{\mathcal{R}})(\rho)\|_{\mathrm{tr}} \leqslant \epsilon$

(no case) for any reference space $\mathcal{R}$, if ρ is a density matrix on the full space $\mathcal{H} \otimes \mathcal{R}$ then $\|(C \otimes \mathcal{I}_{\mathcal{R}})(\rho) - (C_1 \otimes \mathcal{I}_{\mathcal{R}})(\rho)\|_{\mathrm{tr}} \leqslant \epsilon$,

promised one of these to be the case,
where $1 > \epsilon \geqslant 2^{-\mathrm{poly}(n)}$, and provided that $\nexists$ subspace S such that $\|(C_0 \otimes \mathcal{I}_{\mathcal{R}})(\sigma) - (C_1 \otimes \mathcal{I}_{\mathcal{R}})(\sigma)\|_{\mathrm{tr}} \leqslant 2\epsilon$ for all density matrices σ on $S^{\perp} \otimes \mathcal{R}$.

Theorem: QMA-hard for constant δ [proven by Rosgen [46]]
Hardness reduction from: QCSAT (V-1)

Notes: leads to: MIXED-STATE NON-IDENTITY CHECK (V-4), NON-ISOMETRY TESTING (V-5), DETECTING INSECURE QUANTUM ENCRYPTION(V-6)

7.C Diagram of QMA-complete problems

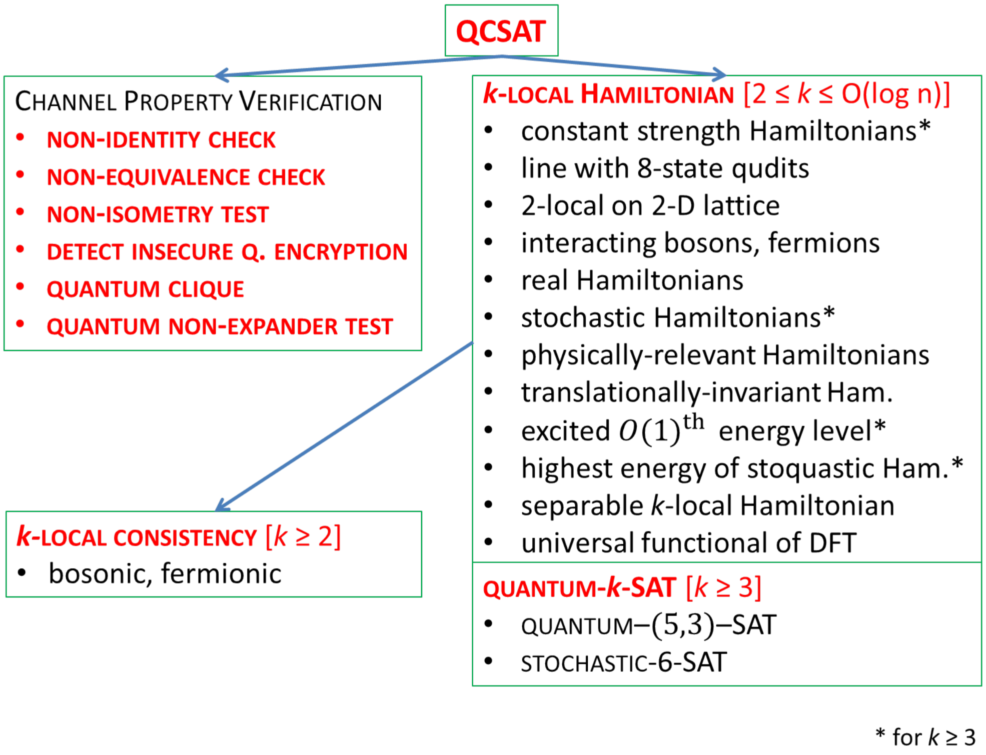

QCSAT
CHANNEL PROPERTY VERIFICATION
NON-IDENTITY CHECK
NON-EQUIVALENCE CHECK
NON-ISOMETRY TEST
DETECT INSECURE Q. ENCRYPTION
QUANTUM CLIQUE
QUANTUM NON-EXPANDER TEST
k-LOCAL HAMILTONIAN [$2 \leq k \leq O(\log n)$]
constant strength Hamiltonians*
line with 8-state qudits
2-local on 2-D lattice
interacting bosons, fermions
real Hamiltonians
stochastic Hamiltonians*
physically-relevant Hamiltonians
translationally-invariant Ham.
excited $O(1)^{\text{th}}$ energy level*
highest energy of stoquastic Ham.*
separable *k*-local Hamiltonian
universal functional of DFT
k-LOCAL CONSISTENCY [$k \geq 2$]
bosonic, fermionic
QUANTUM-*k*-SAT [$k \geq 3$]
QUANTUM–(5,3)–SAT
STOCHASTIC-6-SAT
* for $k \geq 3$

Figure 7-1: Schematic showing the QMA-complete problems listed in this chapter (up to August 2013), according to their categories. Lines show hardness reductions.

Chapter bibliography

For convenience, this survey chapter's bibliography is sorted alphabetically rather than by order of citation.

[1] D. Aharonov, D. Gottesman, S. Irani, and J. Kempe. *The power of quantum systems on a line.* Commun. Math. Phys., 287(1):41 [2009]. http://dx.doi.org/10.1007/s00220-008-0710-3

[2] F. Barahona. *On the computational complexity of Ising spin glass models.* J. Phys. A: Math. Gen., 15(10):3241 [1982]. http://dx.doi.org/10.1088/0305-4470/15/10/028

[3] S. Beigi and P. W. Shor. *On the complexity of computing Zero-error and Holevo capacity of quantum channels.* arXiv:0709.2090 [2007]. http://arxiv.org/abs/0709.2090

[4] J. D. Biamonte and P. J. Love. *Realizable Hamiltonians for universal adiabatic quantum computers.* Phys. Rev. A, 78(1):012352 [2008]. http://dx.doi.org/10.1103/PhysRevA.78.012352

[5] A. D. Bookatz. *QMA-complete problems.* Quantum Info. Comp., 14(5-6):0361 [2014]. http://arxiv.org/abs/1212.6312

[6] A. D. Bookatz, S. P. Jordan, Y.-K. Liu, and P. Wocjan. *Quantum nonexpander problem is quantum-Merlin-Arthur-complete.* Phys. Rev. A, 87(4):042317 [2013]. http://dx.doi.org/10.1103/PhysRevA.87.042317. See Chapter 6 of this thesis.

[7] S. Bravyi. *Efficient algorithm for a quantum analogue of 2-SAT.* In K. Mahdavi, D. Koslover, and L. Brown (eds.), *Cross Disciplinary Advances in Quantum Computing, Contemporary Mathematics*, volume 536, pp. 33–48. American Mathematical Society, Providence, RI, USA [2011]. http://arxiv.org/abs/quant-ph/0602108

[8] S. Bravyi, D. P. DiVincenzo, R. I. Oliveira, and B. M. Terhal. *The complexity of stoquastic local Hamiltonian problems.* Quantum Info. Comp., 8(5):0361 [2008]. http://arxiv.org/abs/quant-ph/0606140

[9] S. Bravyi and B. Terhal. *Complexity of stoquastic frustration-free Hamiltonians.* SIAM J. Comput., 39(4):1462 [2009]. http://dx.doi.org/10.1137/08072689X

[10] N. P. Breuckmann and B. M. Terhal. *Space-time circuit-to-Hamiltonian construction and its applications.* J. Phys. A: Math. Theor., 47(19):195304 [2014]. http://dx.doi.org/10.1088/1751-8113/47/19/195304

[11] Y. Cao and D. Nagaj. *Perturbative gadgets without strong interactions.* Quantum Info. Comput., 15(13-14):1197 [2015]. http://arxiv.org/abs/1408.5881

[12] A. Chailloux and O. Sattath. *The complexity of the separable Hamiltonian problem.* In *2012 IEEE 27th Annual Conference on Computational Complexity (CCC)*, pp. 32–41 [2012]. http://dx.doi.org/10.1109/CCC.2012.42

[13] A. M. Childs, D. Gosset, and Z. Webb. *The Bose-Hubbard model is QMA-complete.* In J. Esparza, P. Fraigniaud, T. Husfeldt, and E. Koutsoupias (eds.), *Automata, Languages, and Programming*, number 8572 in Lecture Notes in Computer Science, pp. 308–319. Springer, Berlin, Germany [2014]. http://dx.doi.org/10.1007/978-3-662-43948-7_26

[14] A. M. Childs, D. Gosset, and Z. Webb. *Complexity of the XY antiferromagnet at fixed magnetization.* Quantum Info. Comput., 16(1-2):1 [2016]. http://arxiv.org/abs/1503.07083

[15] T. Cubitt and A. Montanaro. *Complexity classification of local Hamiltonian problems.* arXiv:1311.3161 [2013]. http://arxiv.org/abs/1311.3161

[16] L. Eldar and O. Regev. *Quantum SAT for a qutrit-cinquit pair is QMA1-complete.* In L. Aceto, I. Damgård, L. A. Goldberg, M. M. Halldórsson, A. Ingólfsdóttir, and I. Walukiewicz (eds.), *Automata, Languages and Programming*, number 5125 in Lecture Notes in Computer Science, pp. 881–892. Springer, Berlin, Germany [2008]. http://dx.doi.org/10.1007/978-3-540-70575-8_72

[17] R. P. Feynman. *Quantum mechanical computers.* Found Phys, 16(6):507 [1986]. http://dx.doi.org/10.1007/BF01886518

[18] S. Gharibian and J. Kempe. *Approximation algorithms for QMA-complete problems.* SIAM J. Comput., 41(4):1028 [2012]. http://dx.doi.org/10.1137/110842272

[19] D. Gosset and D. Nagaj. *Quantum 3-SAT is QMA1-complete.* In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 756–765 [2013]. http://dx.doi.org/10.1109/FOCS.2013.86

[20] D. Gottesman and S. Irani. *The quantum and classical complexity of translationally invariant tiling and Hamiltonian problems.* Theory of Computing, 9(1):31 [2013]. http://dx.doi.org/10.4086/toc.2013.v009a002

[21] A. B. Grilo, I. Kerenidis, and A. Pereszlényi. *Pointer quantum PCPs and multi-prover games.* arXiv:1603.00903 [2016]. http://arxiv.org/abs/1603.00903

[22] A. B. Grilo, I. Kerenidis, and J. Sikora. *QMA with subset state witnesses.* In G. F. Italiano, G. Pighizzini, and D. T. Sannella (eds.), *Mathematical Foundations of Computer Science 2015*, number 9235 in Lecture Notes in Computer Science, pp. 163–174. Springer, Berlin, Germany [2015]. http://dx.doi.org/10.1007/978-3-662-48054-0_14

[23] G. Gutoski, P. Hayden, K. Milner, and M. M. Wilde. *Quantum interactive proofs and the complexity of separability testing.* Theory of Computing, 11(1):59 [2015]. http://dx.doi.org/10.4086/toc.2015.v011a003

[24] S. Hallgren, D. Nagaj, and S. Narayanaswami. *The local Hamiltonian problem on a line with eight states is QMA-complete.* Quantum Info. Comput., 13(9-10):721 [2013]. http://arxiv.org/abs/1312.1469

[25] Y. Huang. *2D local Hamiltonian with area laws is QMA-complete.* arXiv:1411.6614 [2014]. `http://arxiv.org/abs/1411.6614`

[26] D. Janzing, P. Wocjan, and T. Beth. *"Non-identity-check" is QMA-complete.* Int. J. Quantum Inform., 03(03):463 [2005]. `http://dx.doi.org/10.1142/S0219749905001067`

[27] Z. Ji and X. Wu. *Non-identity check remains QMA-complete for short circuits.* arXiv:0906.5416 [2009]. `http://arxiv.org/abs/0906.5416`

[28] S. P. Jordan, D. Gosset, and P. J. Love. *Quantum-Merlin-Arthur–complete problems for stoquastic Hamiltonians and Markov matrices.* Phys. Rev. A, 81(3):032331 [2010]. `http://dx.doi.org/10.1103/PhysRevA.81.032331`

[29] S. P. Jordan, H. Kobayashi, D. Nagaj, and H. Nishimura. *Achieving perfect completeness in classical-witness quantum Merlin-Arthur proof systems.* Quantum Info. Comput., 12(5-6):461 [2012]. `http://arxiv.org/abs/1111.5306`

[30] A. Kay. *Quantum-Merlin-Arthur-complete translationally invariant Hamiltonian problem and the complexity of finding ground-state energies in physical systems.* Phys. Rev. A, 76(3):030307 [2007]. `http://dx.doi.org/10.1103/PhysRevA.76.030307`

[31] A. Kay. *Computational power of symmetric Hamiltonians.* Phys. Rev. A, 78(1):012346 [2008]. `http://dx.doi.org/10.1103/PhysRevA.78.012346`

[32] A. Kay. *Role of rotational invariance in the properties of Hamiltonians.* Phys. Rev. A, 80(4):040301 [2009]. `http://dx.doi.org/10.1103/PhysRevA.80.040301`

[33] J. Kempe, A. Kitaev, and O. Regev. *The complexity of the local Hamiltonian problem.* SIAM J. Comput., 35(5):1070 [2006]. `http://dx.doi.org/10.1137/S0097539704445226`

[34] A. Y. Kitaev, A. H. Shen, and M. N. Vyalyi. *Classical and quantum computation.* American Mathematical Society, Boston, MA, USA [2002]

[35] E. Knill. *Quantum randomness and nondeterminism.* arXiv:quant-ph/9610012 [1996]. `http://arxiv.org/abs/quant-ph/9610012`

[36] H. Kobayashi, F. Le Gall, and H. Nishimura. *Generalized quantum Arthur-Merlin games.* In *Proceedings of the 30th Conference on Computational Complexity*, CCC '15, pp. 488–511. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Germany [2015]. `http://dx.doi.org/10.4230/LIPIcs.CCC.2015.488`

[37] Y.-K. Liu. *Consistency of local density matrices is QMA-complete.* In J. Díaz, K. Jansen, J. D. P. Rolim, and U. Zwick (eds.), *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, number 4110 in Lecture Notes in Computer Science, pp. 438–449. Springer, Berlin, Germany [2006]. `http://dx.doi.org/10.1007/11830924_40`

[38] Y.-K. Liu, M. Christandl, and F. Verstraete. *Quantum computational complexity of the N-representability problem: QMA-complete.* Phys. Rev. Lett., 98(11):110503 [2007]. `http://dx.doi.org/10.1103/PhysRevLett.98.110503`

[39] C. Marriott and J. Watrous. *Quantum Arthur–Merlin games.* comput. complex., 14(2):122 [2005]. http://dx.doi.org/10.1007/s00037-005-0194-x

[40] T. Morimae. *Quantum Arthur-Merlin with single-qubit measurements.* arXiv:1602.08656 [2016]. http://arxiv.org/abs/1602.08656

[41] D. Nagaj. *Local Hamiltonians in quantum computation.* Ph.d. thesis, Massachusetts Institute of Technology, Cambridge, MA, USA [2008]. http://dx.doi.org/1721.1/45162

[42] D. Nagaj and S. Mozes. *New construction for a QMA complete three-local Hamiltonian.* Journal of Mathematical Physics, 48(7):072104 [2007]. http://dx.doi.org/10.1063/1.2748377

[43] R. Oliveira and B. M. Terhal. *The complexity of quantum spin systems on a two-dimensional square lattice.* Quantum Info. Comput., 8(10):900 [2008]. http://arxiv.org/abs/quant-ph/0504050

[44] S. Piddock and A. Montanaro. *The complexity of antiferromagnetic interactions and 2D lattices.* arXiv:1506.04014 [2015]. http://arxiv.org/abs/1506.04014

[45] B. Rosgen. *Testing non-isometry is QMA-complete.* In W. van Dam, V. M. Kendon, and S. Severini (eds.), *Theory of Quantum Computation, Communication, and Cryptography*, number 6519 in Lecture Notes in Computer Science, pp. 63–76. Springer, Berlin, Germany [2010]. http://dx.doi.org/10.1007/978-3-642-18073-6_6

[46] B. Rosgen. *Testing quantum circuits and detecting insecure encryption.* In K. Iwama, Y. Kawano, and M. Murao (eds.), *Theory of Quantum Computation, Communication, and Cryptography*, number 7582 in Lecture Notes in Computer Science, pp. 74–86. Springer, Berlin, Germany [2012]. http://dx.doi.org/10.1007/978-3-642-35656-8_6

[47] O. Sattath. *An almost sudden jump in quantum complexity.* Quantum Info. Comput., 15(11-12):1048 [2015]. http://arxiv.org/abs/1310.5372

[48] N. Schuch and F. Verstraete. *Computational complexity of interacting electrons and fundamental limitations of density functional theory.* Nat Phys, 5(10):732 [2009]. http://dx.doi.org/10.1038/nphys1370

[49] K. G. H. Vollbrecht and J. I. Cirac. *Quantum simulators, continuous-time automata, and translationally invariant systems.* Phys. Rev. Lett., 100(1):010501 [2008]. http://dx.doi.org/10.1103/PhysRevLett.100.010501

[50] J. Watrous. *Succinct quantum proofs for properties of finite groups.* In *41st Annual Symposium on Foundations of Computer Science, 2000. Proceedings*, pp. 537–546 [2000]. http://dx.doi.org/10.1109/SFCS.2000.892141

[51] T.-C. Wei, M. Mosca, and A. Nayak. *Interacting Boson problems can be QMA hard.* Phys. Rev. Lett., 104(4):040501 [2010]. http://dx.doi.org/10.1103/PhysRevLett.104.040501

[52] J. D. Whitfield, P. J. Love, and A. Aspuru-Guzik. *Computational complexity in electronic structure*. Phys. Chem. Chem. Phys., 15(2):397 [2012]. `http://dx.doi.org/10.1039/C2CP42695A`