Computational Problems Related to Open Quantum Systems

by

Chunhao Wang

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

Waterloo, Ontario, Canada, 2018

Examining Committee Membership

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

External Examiner:	Hoi Fung Chau Professor, Department of Physics, The University of Hong Kong
Supervisor(s):	Richard Cleve Professor, Cheriton School of Computer Science, University of Waterloo
Internal Member:	Debbie Leung Professor, Department of Combinatorics and Optimization, University of Waterloo
Internal Member:	John Watrous Professor, Cheriton School of Computer Science, University of Waterloo
Internal-External Member:	Ashwin Nayak Professor, Department of Combinatorics and Optimization, University of Waterloo

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

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

Abstract

A quantum system that has interaction with external resources, such as probability distribution, dissipation, and noise, is referred to as an open quantum system. Not only do open quantum systems play a vital role in the field of quantum physics, but they are also fundamental objects in quantum information and quantum computing. In this thesis, we focus on computational problems related to open quantum systems. In particular, we study efficient constructions of open quantum systems and their algorithmic applications.

A unitary 2-design is a quantum analogue of universal 2-hash functions. It is an example of open quantum systems in the sense that it is a probability distribution of unitaries. As unitary 2-designs inherit many properties of the Haar randomness on the unitary group, they have many applications in quantum information, such as benchmarking and decoupling. We study the structures of unitary 2-designs and present efficient methods for their constructions.

The continuous-time evolution of a closed quantum system can be described by the Schrödinger equation. A natural generalization of the Schrödinger equation to Markovian open quantum systems, in the sense of generating dynamical semigroups, is called the Lindblad equation. We show that it is impossible for a simple reductionist approach to simulate Lindblad evolution with gate complexity that has linear dependence in evolution time. Moreover, we use a novel variation of the "linear combination of unitaries" construction that pertains to quantum channels to achieve the desired linear dependence in evolution time and poly-logarithmic dependence in precision.

Open quantum systems can also be used as building blocks of quantum algorithms. We present a dissipative query model, which is based on the amplitude damping process. With this dissipative query model, we provide a quantum algorithm that performs a fixed-point quantum search while preserving the quadratic speedup against classical algorithms.

Acknowledgements

First of all, I would like to thank my supervisor Richard Cleve for his advice, guidance, and support during my Ph.D. program. From every single meeting and discussion with him, I gained knowledge and insight, which are influential in my research and future career.

I want to thank my committee members Hoi Fung Chau, Richard Cleve, Debbie Leung, Ashwin Nayak, and John Watrous for reading my thesis and giving me the valuable feedback. I am extremely grateful to Richard Cleve, Peter Høyer, Debbie Leung, Ashwin Nayak, and John Watrous for the discussion and guidance of the research presented in this thesis. I also would like to thank Li Liu and Leonard Wossnig for the productive collaborations and fruitful discussions. I am deeply indebted to the faculty members at IQC and CS who helped me, and to all my friends in Waterloo who enriched my life.

I would like to give special thanks to my wife Chang Liu for her endless support. She also contributed to this thesis with love and understanding.

Dedication

This thesis is dedicated to my grandparents and my wife.

Table of Contents

List of Figures

Chapter 1

Introduction

A quantum system that is isolated from the environment (i.e., some external system) is referred to as a *closed quantum system*. Examples of such quantum systems are Hamiltonian evolution and unitary operators, which are fundamental objects in quantum physics and quantum computing. *Open quantum systems* can be viewed as an extension of closed quantum systems by allowing interaction with external resources, such as randomness, dissipation, and noise. Examples of open quantum systems arise in probability distribution on unitary operators, Lindblad evolution, and quantum channels. Unitary 2-designs are probability distributions on unitary operators. They serve as a source of randomness in quantum information, and they have many applications related to estimating properties of other quantum channels. Lindblad evolution is a generalization of the Schrödinger evolution to Markovian open quantum systems in the sense of generating dynamical semigroups.

Closed quantum systems are the main objects of interest for solving computational problems: most quantum algorithms are built based on unitary operators. Open quantum systems are often related to communication (e.g., quantum channels), and hence a lot of research has been conducted on the information-theoretical problems related to open quantum systems, such as estimating the capacity, fidelity, and information complexity of quantum channels. In this thesis, we focus on the *computational problems* related to open quantum systems. In particular, we study efficient quantum algorithms for simulating Lindblad evolution and constructing open quantum systems such as unitary 2-designs. We also demonstrate that open quantum systems can be used to design quantum algorithms. In particular, we present a dissipative quantum search algorithm which achieves properties that were not achieved with pure unitary operators, e.g., the error-converging fixed-point property with the quadratic speedup.

1.1 Unitary 2-designs

In classical computer science, it has been observed that 2-universal hash functions [CW79] have the property that, with at most two queries, they cannot be distinguished from random functions[1]. Analogues of 2-universal hash functions in quantum computing have recently received increasing attention. One such analogue is called a unitary 2-design. A unitary 2-design is a probability distribution on some finite subset of the unitary group (the group of all unitaries of a certain dimension, where the formal definition is in Section 2.1). Sampling from this probability distribution simulates the procedure of sampling from the *Haar measure* (i.e., the unique measure that is invariant under right- and left-multiplications of group elements) of the unitary group in the following sense: if a distinguishing procedure can query a unitary at most twice, it cannot distinguish whether this unitary is sampled from a unitary 2-design or sampled from the Haar measure of the unitary group.

The Haar-randomness on the unitary group facilitates many analyses in quantum information (for example, in [Has09, HHWY08, HLSW04, HLW06, HW08]). In particular, the *full bilateral twirl* (applying a Haar-random unitary in parallel on a bipartite system, introduced in [BBP+96, BDSW96]) appears in various mathematical proofs in quantum information [HHWY08, SDTR13]; the *full channel twirl* (applying a Haar-random unitary U on the input state of some quantum channel and then applying $U^\dagger$ on the output state) has important applications such as estimating the average channel fidelity of quantum devices [DCEL09] and error estimating in quantum key distribution [Cha05]. Roughly speaking, a unitary 2-design has the property that, sampling from it implements the full bilateral twirl and the full channel twirl. Moreover, appropriate notions of approximate unitary 2-designs in the sense of approximating the full bilateral twirl and the full channel twirl have also been explored in [DCEL09, DLT02, HL09].

Implementing Haar-random unitaries requires very high computational complexity: it requires many bits to describe and a lot of randomness to sample due to the fact that a unitary acting on n qubits has $2^n \times 2^n$ entries. Since unitary 2-designs inherit many properties of the Haar randomness on the unitary group, it is desirable to implement unitary 2-designs with small quantum circuits and a small sampling cost. Unitary 2-designs are closely related to the Clifford group, which is the subgroup of the unitary group that permutes Paulis (Definition 3.6). The uniform distribution over the Clifford group is a unitary 2-design. The unitaries in the Clifford group can be implemented by a quantum circuit of $O(n^2/\log n)$ gates [AG04]. The sampling cost is $O(n^2)$ random bits of entropy. In this thesis, we show three constructions of exact unitary 2-designs of gate

[1] Here we assume that for any input, the output of a 2-universal hash function is uniformly distributed.

complexity (measured by 1- and 2-qubit gates) $\widetilde{O}(n)$ (where $\widetilde{O}(f(n))$ is a shorthand for $O(f(g)\log^k(f(g)))$ for some $k \geq 0$), that need $O(n)$ random bits of entropy for sampling. Note that our constructions require ancilla qubits.

Because the constructions presented in this thesis are for exact unitary 2-designs, they can be used for all notions and definitions of approximate unitary 2-designs. Therefore, the constructions achieve the best-known gate complexity, circuit depth, and sampling complexity simultaneously, for both exact and approximate unitary 2-designs.

Efficient constructions of unitary 2-designs have many computational applications in the context of full bilateral twirl and full channel twirl, such as data hiding [DLT02], designing codes for transmitting data through noisy quantum channels [HHWY08], decoupling of two systems (breaking coherence between two systems) [SDTR13], estimating the average channel fidelity of quantum devices [DCEL09] and error estimating in quantum key distribution [Cha05]. For example, as suggested in [HHWY08], there exists an encoding operation in any unitary 2-design that achieves the quantum channel capacity when concatenated with an appropriate inner code. As a result, the constructions presented in this thesis automatically imply the existence of such encoding circuits with gate complexity $O(n \log n \log\log n)$ and circuit depth $O(\log n)$.

In some applications such as decoupling, the approximation error ϵ of an approximate unitary 2-design (e.g., [DCEL09]) can be amplified by a factor that is exponential in n (for example., in [SDTR13, Theorem 1]). This issue can be overcome by using exact constructions of unitary 2-designs. Therefore, for these applications, the constructions presented in this thesis yield tighter bounds than approximate unitary 2-designs while maintaining the gate complexity $\widetilde{O}(n)$.

1.2 Lindblad evolution

The problem of simulating the evolution of closed quantum systems (captured by the Schrödinger equation) was proposed by Feynman [Fey82] in 1982 as a motivation for building quantum computers. Since then, several quantum algorithms have appeared for this problem. However, many quantum systems of interest are open quantum systems but are well-captured by the Lindblad master equation [GKS76, Lin76]. Examples exist in quantum physics [LCD+87, Wei12], quantum chemistry [MK08, Nit06], and quantum biology [DGV12, HP13, MRE+12]. Lindblad evolution also arises in quantum computing and quantum information in the context of entanglement preparation [KBD+08, KRS11, RRS16], thermal state preparation [KB16], quantum state engineering [VWC09], and studying the noise of quantum circuits [MPGC13].

More precisely, the Lindblad equation is the natural generalization of the Schrödinger equation to the dynamics of *Markovian open quantum systems* in the sense of generating dynamical semigroups, and it is defined as

$$\frac{d}{dt}\rho = -i[H,\rho] + \sum_{j=1}^{m}\left(L_j\rho L_j^\dagger - \frac{1}{2}L_j^\dagger L_j\rho - \frac{1}{2}\rho L_j L_j^\dagger\right), \tag{1.1}$$

where H is a Hamiltonian, $L_1,\ldots,L_m$ are linear operators, and $[H,\rho]$ denotes the operator $H\rho-\rho H$. We denote the superoperator defined as Eq. (1.1) by $\mathcal{L}$, and call it a *Lindbladian* (i.e., $\frac{d}{dt}\rho = \mathcal{L}[\rho]$). The evolution by $\mathcal{L}$ for time t corresponds to a quantum channel $e^{\mathcal{L}t}$.

For example, the *depolarizing process* can be described by the following Lindblad equation:

$$\frac{d}{dt}\rho = \sum_{j=1}^{3}\left(L_j\rho L_j^\dagger - \frac{1}{2}L_j^\dagger L_j\rho - \frac{1}{2}\rho L_j L_j^\dagger\right), \tag{1.2}$$

where $L_1 = X/\sqrt{3}, L_2 = Y/\sqrt{3}$, and $L_3 = Z/\sqrt{3}$ for Pauli operators X, Y, and Z. The *phase damping process* can be described by the following Lindblad equation:

$$\frac{d}{dt}\rho = \sum_{j=1}^{2}\left(L_j\rho L_j^\dagger - \frac{1}{2}L_j^\dagger L_j\rho - \frac{1}{2}\rho L_j L_j^\dagger\right), \tag{1.3}$$

where $L_1 = \left(\begin{smallmatrix}1&0\\0&0\end{smallmatrix}\right)$, and $L_2 = \left(\begin{smallmatrix}0&0\\0&1\end{smallmatrix}\right)$. The *amplitude damping process* can be described by the following Lindblad equation:

$$\frac{d}{dt}\rho = L\rho L^\dagger - \frac{1}{2}L^\dagger L\rho - \frac{1}{2}\rho L L^\dagger, \tag{1.4}$$

where $L = \left(\begin{smallmatrix}0&1\\0&0\end{smallmatrix}\right)$.

In this thesis, we focus on the methods for simulating Lindblad evolution. By *simulating the evolution*, we mean providing a quantum circuit that computes the quantum channel $e^{\mathcal{L}t}$. We are interested in the cost of the simulation, which is measured by the number of 1- and 2-qubit gates in this circuit.

Lindblad evolution can be intuitively thought of as Hamiltonian evolution in a larger system that includes an ancilla register, but the ancilla register is being continually reset to its initial state to preserve its Markov property (i.e., the system state at time $t+\delta$ is completely determined by the system state at time t for any $\delta > 0$). This suggests a

reductionist approach to simulate Lindblad evolution as Hamiltonian evolution in a larger space. However, in this thesis, we show that it is impossible to achieve the gate complexity with linear dependence in evolution time using this simple reductionist approach.

As an algorithmic contribution, we show a quantum algorithm for simulating the evolution of an n-qubit system for time t within precision ϵ. If the Lindbladian consists of $\mathrm{poly}(n)$ operators that can each be expressed as a linear combination of $\mathrm{poly}(n)$ tensor products of Pauli operators then the gate cost of this algorithm is $O(t\,\mathrm{polylog}(t/\epsilon)\mathrm{poly}(n))$. We also obtain similar bounds for the case where the Lindbladian consists of local operators, and where the Lindbladian consists of sparse operators. This algorithm is based on a novel *linear combination of unitaries (LCU) for channels* and *oblivious amplitude amplification for isometries*, where the standard linear combination of unitaries (LCU) and the oblivious amplitude amplification for unitaries have been used in Hamiltonian simulation algorithms (as briefly discussed in Section 5.2).

1.3 Dissipative quantum search

Open quantum systems can also be used as building blocks in quantum algorithms. We demonstrate that, based on the amplitude damping process and Grover's search algorithm [Gro96], a quantum search algorithm can achieve the fixed-point property without losing its quadratic speedup.

Grover's search algorithm and its generalizations are important tools in quantum computing. These algorithms provide a quadratic speedup against their classical counterparts for the search problem. As a consequence, classical algorithms for solving NP-complete problems (which are believed hard to solve unless NP=P) that are based on the obvious brute-force search gain quadratic speedup by using Grover's algorithm on a quantum computer. In a general setting of an unordered search problem, there is a search space of N items, and M of them are marked. A boolean function $f : \{0, \ldots, N-1\} \to \{0, 1\}$ determines whether an item $j \in \{0, \ldots, N-1\}$ is marked ($f(j) = 1$) or not ($f(j) = 0$). The goal is to find a marked item using as few queries to f as possible. Both randomized and deterministic classical algorithms require $\Omega(N/M)$ queries, but Grover's search algorithm uses $O(\sqrt{N/M})$ queries. Around the same time when Grover's search algorithm was proposed, Farhi and Gutmann [FG98] proposed a search algorithm in the form of Hamiltonian evolution, which also achieves quadratic speedup over classical algorithms. Often being viewed as the continuous-time version of Grover's search algorithm, Farhi and Gutmann's algorithm sheds light on quantum algorithms design and provides a different point of view for quantum search algorithms.

There exist other variations of Grover's search algorithm. In the context of boosting success probability of some random experiment, suppose the probability that a random variable takes a certain value is p. In the classical case, $\Omega(1/p)$ samples are required to achieve constant probability that the desired value is observed. However, in a more generalized Grover's search algorithm, which is often referred to as *amplitude amplification* [BHMT02], only $O(\sqrt{1/p})$ samples suffice.

Despite its ubiquitousness and versatility, a notorious problem of Grover's search algorithm – the "over-cooking" problem cannot be ignored. In the M-out-of-N search problem, Grover's search algorithm detects a marked item in $O(\sqrt{N/M})$ queries with high probability. However, when the number of marked items M is unknown, the number of iterations of Grover's search algorithm is unknown. If more iterations are applied (even a constant factor more than optimal), the overlap between the resulting state and the desired state could be (exponentially!) small, or even zero.

Previous work for solving this "over-cooking" problem is based on unitary operators. However, to the best of our knowledge, these methods do not possess the desirable error-converging property, i.e., the error (not necessarily monotonically) decreases with more iterations (where each iteration requires a constant number of queries), without losing the quadratic speedup. In this thesis, we propose a query model based on the amplitude damping process, namely, the *dissipative query*, and present a dissipative quantum search algorithm which has the fixed-point property while preserving the quadratic speedup.

1.4 Organization of this thesis

This thesis is organized as follows.

Chapter 2 : We provide the necessary background for this thesis. We first present the terminologies and notations for quantum information, and then show some properties of finite fields.

Chapter 3 : We study an example of open quantum systems – unitary 2-designs. The goal of this chapter is to develop efficient constructions of unitary 2-designs.

Chapter 4 : We study the notion of continuous-time evolution for open quantum systems. In particular, we show a derivation of the Lindblad equation and prove a lower bound for simulating Lindblad evolution as Hamiltonian evolution in a larger Hilbert space.

Chapter 5 : We present a quantum algorithm for simulating Lindblad evolution, achieving linear dependence in evolution time and poly-logarithmic dependence in precision.

Chapter 6 : We demonstrate that open quantum systems can be used in quantum algorithms. In particular, we show a fixed-point quantum search algorithm that preserves the quadratic speedup. This quantum algorithm is built upon a novel dissipative query model, which is based on the amplitude damping process.

Chapter 7 : We finish this thesis with concluding remarks.

Chapter 2

Notation and Preliminaries

2.1 Basics for quantum computing

We assume that readers have a basic knowledge of linear algebra, measure theory, and group theory. The purpose of this section is to provide the minimum prerequisites to follow this thesis. We present terminologies and notations that will be used through the following chapters. To get a more comprehensive background in quantum computing and quantum information, readers may refer to the book by Nielsen and Chuang [NC00] and the book by Watrous [Wat18].

Hilbert spaces and quantum states

In this thesis, when we refer to a *Hilbert space*, we mean a *finite-dimensional Hilbert space*. Therefore, it is convenient to use the notation of a *complex Euclidean space* of dimension N, $\mathbb{C}^N$, to denote a Hilbert space of dimension N. The *tensor product* for two Hilbert spaces $\mathbb{C}^{N_1}$ and $\mathbb{C}^{N_2}$, denoted by $\mathbb{C}^{N_1} \otimes \mathbb{C}^{N_2}$ is defined as the Hilbert space $\mathbb{C}^{N_1 N_2}$.

We use the bra-ket notation (a.k.a. the Dirac notation) $|u\rangle_N \in \mathbb{C}^N$ to represent a column vector and use $\langle u|_N$ to denote its *complex conjugate transpose*. The *computational basis* for $\mathbb{C}^N$, is the set $\{|0\rangle_N, \ldots, |N-1\rangle_N\}$, where for all $j \in \{0, \ldots, N-1\}$, $|j\rangle_N$ is a column vector where the entry corresponding to position j is 1, and all other entries are 0. For example, when $N = 2$, $|0\rangle_2 = \binom{1}{0}$, and $|1\rangle_2 = \binom{0}{1}$. Note the difference between notations $|u\rangle_N$ and $|j\rangle_N$: u is just a symbol and $|u\rangle_N$ is used for any column vector, while $j \in \{0, \ldots, N-1\}$ is a variable, and $|j\rangle_N$ is used for a computational basis vector. When the dimension is clear from the context, we omit the subscript from the bra-ket notation, and simply denote a column vector by $|u\rangle$.

For any two column vectors $|u\rangle_N, |v\rangle_N \in \mathbb{C}^N$, we use $|u\rangle\langle v|_N$ as a shorthand for $|u\rangle_N\langle v|_N$, which is an $N \times N$ matrix, and use $\langle u|v\rangle_N$ as a shorthand for $\langle u|_N|v\rangle_N$, which is a complex number. The subscript N in $|u\rangle\langle v|_N$ and $\langle u|v\rangle_N$ are usually omitted if the dimension is clear from the context.

The quantity $\langle u|v\rangle$ is called the *inner product* of two column vectors $|u\rangle, |v\rangle \in \mathbb{C}^N$. We can express $|u\rangle \in \mathbb{C}^N$ with respect to the computational basis as

$$|u\rangle = u_0|0\rangle + u_1|1\rangle + \cdots + u_{N-1}|N-1\rangle, \tag{2.1}$$

where $u_j = \langle j|u\rangle$ for all $j \in \{0, \ldots, N-1\}$. Let $|u\rangle \in \mathbb{C}^{N_1}$ and $|v\rangle \in \mathbb{C}^{N_2}$ be two column vectors represented as $|u\rangle = u_0|0\rangle + u_1|1\rangle + \cdots + u_{N_1-1}|N_1-1\rangle$ and $|v\rangle = u_0|0\rangle + v_1|1\rangle + \cdots + v_{N_2-1}|N_2-1\rangle$, respectively. The *tensor product* of column vectors $|u\rangle$ and $|v\rangle$, denoted by $|u\rangle \otimes |v\rangle$ (or simply $|u\rangle|v\rangle$), is defined as the vector $|w\rangle \in \mathbb{C}^{N_1N_2}$ that can be represented as

$$|w\rangle = \sum_{j,k=0}^{N_1-1,N_2-1} u_j v_k |j,k\rangle, \tag{2.2}$$

where $|j,k\rangle$ is defined as $|jN_2 + k\rangle$.

The *Euclidean norm* of $|u\rangle \in \mathbb{C}^N$ is defined as

$$\||u\rangle\| = \sqrt{\langle u|u\rangle} = \sqrt{\sum_{j=0}^{N-1} |u_j|^2}. \tag{2.3}$$

A column vector $|u\rangle$ is *normalized* if $\||u\rangle\| = 1$. An n-qubit *normalized quantum state* is usually represented as a normalized column vector $|u\rangle \in \mathbb{C}^{2^n}$. A 2^n*-dimensional quantum register* is a device the can store and process n-qubit quantum states (and therefore can be associated with a Hilbert space $\mathbb{C}^{2^n}$). The operators on a quantum register include initialization, as well as unitaries and measurements, which are defined in the subsequent content. In this thesis, all quantum states in the bra-ket notation are normalized unless otherwise specified.

A *measurement* on a quantum register that is in some state $|\psi\rangle \in \mathbb{C}^{2^n}$ with respect to a basis $\{|u_0\rangle, \ldots, |u_{2^n-1}\rangle\}$ is a non-reversible operation, after which, the probability of observing the outcome j is $|\langle\psi|u_j\rangle|^2$, and the state of this quantum register *collapses* to $|u_j\rangle$.

Linear operators

There are *linear operators* that map vectors in $\mathbb{C}^N$ to vectors in $\mathbb{C}^M$. We use $\mathrm{L}(\mathbb{C}^M, \mathbb{C}^N)$ to denote the set of all linear operator of the form:

$$A : \mathbb{C}^M \to \mathbb{C}^N. \tag{2.4}$$

If the input space and output space are the same, we use the notation $\mathrm{L}(\mathbb{C}^N)$ as a shorthand for $\mathrm{L}(\mathbb{C}^N, \mathbb{C}^N)$. With respect to the computational basis, we can associate each $A \in \mathrm{L}(\mathbb{C}^M, \mathbb{C}^N)$ with a matrix $M \in \mathbb{C}^{N \times M}$ as follows

$$(M)_{j,k} = \langle j|A|k\rangle, \tag{2.5}$$

for $j \in \{0, \ldots, N-1\}$ and $k \in \{0, \ldots, M-1\}$, where $(M)_{j,k}$ denotes the (j,k)-entry of M. Due to this association, in this thesis, we do not distinguish an operator from its matrix representation (with respect to the computational basis), and do not distinguish the Hilbert space $\mathrm{L}(\mathbb{C}^M, \mathbb{C}^N)$ from the set of matrices $\mathbb{C}^{N \times M}$ if it does not cause ambiguity. We use the bold symbol $\mathbf{0}_{\mathbb{C}^N}$ to represent the zero-operator, which is corresponding to the all-zero matrix (omitting the subscript if it is clear from the context).

The *identity operator* $I_{\mathbb{C}^N} \in \mathrm{L}(\mathbb{C}^N)$ is the operator satisfying

$$I_{\mathbb{C}^N}|u\rangle_N = |u\rangle_N, \tag{2.6}$$

for all $|u\rangle_N \in \mathbb{C}^N$. It can also be specified as the matrix:

$$(I_{\mathbb{C}^N})_{j,k} = \begin{cases} 1 & \text{if } j = k \\ 0 & \text{if } j \neq k. \end{cases} \tag{2.7}$$

In this thesis, we usually omit the subscript for dimension and simply write I when the dimension is clear from the context.

For every $A \in \mathrm{L}(\mathbb{C}^M, \mathbb{C}^N)$, we define three operators A^T, A^*, and $A^\dagger$ as follows.

1. The operator $A^T \in \mathrm{L}(\mathbb{C}^N, \mathbb{C}^M)$ is the operator obtained by transposing A (with respect to the computational basis):

   $$(A^T)_{j,k} = (A)_{k,j}, \tag{2.8}$$

   for all $j \in \{0, \ldots, M-1\}$ and $k \in \{0, \ldots, N-1\}$.

2. The operator $A^* \in \mathrm{L}(\mathbb{C}^M, \mathbb{C}^N)$ is the operator obtained by taking entry-wise complex conjugate of A (with respect to the computational basis):

   $$(A^*)_{j,k} = (A)^*_{j,k}, \tag{2.9}$$

   for all $j \in \{0, \ldots, N-1\}$ and $k \in \{0, \ldots, M-1\}$.

3. The operator $A^\dagger \in \mathrm{L}(\mathbb{C}^N, \mathbb{C}^M)$ is the operator obtained by performing both of the operation defined in items 1 and 2 on A:

$$A^\dagger = \left(A^T\right)^*. \tag{2.10}$$

For every $A \in \mathrm{L}(\mathbb{C}^N)$, the *trace function*, denoted by $\mathrm{Tr}(A)$, is the linear function defined as

$$\mathrm{Tr}(A) = \sum_{j=0}^{N-1} (A)_{j,j}. \tag{2.11}$$

For any pair of operators $A, B \in \mathrm{L}(\mathbb{C}^N)$, the *commutator* of A and B, denoted by $[A, B]$ is defined as

$$[A, B] = AB - BA. \tag{2.12}$$

Now, we define some classes of operators that will be used in this thesis:

1. *Hermitian operators.* An operator $A \in \mathrm{L}(\mathbb{C}^N)$ is *Hermitian* if $A = A^\dagger$.

2. *Positive semidefinite operators.* An operator $A \in \mathrm{L}(\mathbb{C}^N)$ is *positive semidefinite* if there exists some operator $B \in \mathrm{L}(\mathbb{C}^N)$ such that $A = B^\dagger B$. Every positive semidefinite operator is Hermitian.

3. *Density operators.* An operator $A \in \mathrm{L}(\mathbb{C}^N)$ is a *density operator* if A is positive semidefinite and $\mathrm{Tr}(A) = 1$. In this thesis, we usually use Greek letters such as ρ, σ to denote density operators. We use the notation

   $$\mathrm{D}(\mathbb{C}^N) = \{\rho : \rho \text{ is positive semidefinite and } \mathrm{Tr}(\rho) = 1\} \tag{2.13}$$

   to denote the set of density operators acting on $\mathbb{C}^N$.

4. *Isometries.* An operator $A \in \mathrm{L}(\mathbb{C}^N, \mathbb{C}^M)$ is an *isometry* if it preserves the Euclidean norm, i.e., $\|A|u\rangle\| = \||u\rangle\|$ for all $|u\rangle \in \mathbb{C}^N$.

5. *Unitaries.* An operator $A \in \mathrm{L}(\mathbb{C}^N)$ is a *unitary* if it is an isometry. Every unitary operator has the property that $UU^\dagger = U^\dagger U = I$. We use the notation

   $$\mathrm{U}(\mathbb{C}^N) = \left\{A \in \mathrm{L}(\mathbb{C}^N) : A^\dagger A = I\right\} \tag{2.14}$$

   to denote the set of unitaries acting on $\mathbb{C}^N$. This set forms a group. When $N = 2^n$ for some integer n, we call $\mathrm{U}(\mathbb{C}^N)$ the *unitary group* on n qubits.

Note that density operators are often used to represent *mixed quantum states*, i.e., $\rho \in \mathrm{D}(\mathbb{C}^{2^n})$ can be viewed as a probability mixture of quantum states in the following form

$$\rho = p_0|u_0\rangle\langle u_0| + \cdots + p_{2^n-1}|u_{2^n-1}\rangle\langle u_{2^n-1}|, \tag{2.15}$$

where $p_j \geq 0$ for all $j \in \{0, \ldots, 2^n - 1\}$, $\sum_{j=0}^{2^n-1} p_j = 1$, and $|u_0\rangle, \ldots, |u_{2^n-1}\rangle \in \mathbb{C}^{2^n}$ are n-qubit quantum states. When it is clear from the context, we also refer to a density operator ρ as a quantum state.

Now we define some fundamental unitaries acting on one qubit ($\mathbb{C}^2$), or two qubits ($\mathbb{C}^{2^2}$) that will be used extensively in this thesis.

1. The 1-qubit *Paulis* are $I_{\mathbb{C}^2}, X, Y, Z$, whose matrix representations are the following

$$I_{\mathbb{C}^2} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \text{ and } Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{2.16}$$

2. The *phase gate* acting on one qubit, denoted by S, is defined as

$$S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \tag{2.17}$$

3. The *Hadamard gate* acting on one qubit, denoted by H, is defined as

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{2.18}$$

4. The *controlled-not gate* acting on two qubits, denoted by CNOT, is defined as

$$\mathrm{CNOT}|b_1\rangle|b_2\rangle = |b_1\rangle|b_1 \oplus b_2\rangle, \tag{2.19}$$

   for all $b_1, b_2 \in \{0, 1\}$.

5. The *swap gate* acting on two qubits, denoted by $\mathrm{SWAP}_{\mathbb{C}^2 \otimes \mathbb{C}^2}$, is defined as

$$\mathrm{SWAP}_{\mathbb{C}^2 \otimes \mathbb{C}^2}|b_1\rangle|b_2\rangle = |b_2\rangle|b_1\rangle, \tag{2.20}$$

   for all $b_1, b_2 \in \{0, 1\}$.

The *tensor product* of operators $A \in \mathrm{L}(\mathbb{C}^{N_1}, \mathbb{C}^{M_1})$ and $B \in \mathrm{L}(\mathbb{C}^{N_2}, \mathbb{C}^{M_2})$, denoted by $A \otimes B$, is defined as the unique operator in $\mathrm{L}(\mathbb{C}^{N_1} \otimes \mathbb{C}^{N_2}, \mathbb{C}^{M_1} \otimes \mathbb{C}^{M_2})$ that satisfies

$$(A \otimes B)|u\rangle|v\rangle = A|u\rangle \otimes B|v\rangle, \tag{2.21}$$

for all $|u\rangle \in \mathbb{C}^{N_1}$ and $|v\rangle \in \mathbb{C}^{N_2}$. We use the notation $A^{\otimes n}$ as a shorthand for the n-fold tensor product

$$A \otimes \cdots \otimes A. \tag{2.22}$$

Two norms of operators will be used extensively in this thesis. We define them as follows:

1. The *spectral norm* of an operator $A \in \mathrm{L}(\mathbb{C}^N)$, denoted by $\|A\|$, is defined as

$$\|A\| = \max\left\{\||A|u\rangle\| : |u\rangle \in \mathbb{C}^N, \||u\rangle\| \leq 1\right\}, \tag{2.23}$$

   which is equal to the largest singular value of A.

2. The *trace norm* of an operator $A \in \mathrm{L}(\mathbb{C}^N)$, denoted by $\|A\|_1$, is defined as

$$\|A\|_1 = \mathrm{Tr}\left(\sqrt{A^\dagger A}\right), \tag{2.24}$$

   which is equal to the sum of singular values of A.

For two density operators ρ and σ. The value $\|\rho - \sigma\|_1$ is often referred to as the *trace distance* between ρ and σ.

Linear maps

Linear maps of the form

$$\mathcal{M} : \mathrm{L}(\mathbb{C}^N) \to \mathrm{L}(\mathbb{C}^M) \tag{2.25}$$

are often referred to as *superoperators*. The set of all such linear maps is denoted by $\mathrm{T}(\mathbb{C}^N, \mathbb{C}^M)$. When the input space and the output space are the same, we use the notation $\mathrm{T}(\mathbb{C}^N)$ as a shorthand for $\mathrm{T}(\mathbb{C}^N, \mathbb{C}^N)$. The *identity map* $\mathcal{I}_{\mathrm{L}(\mathbb{C}^N)} \in \mathrm{T}(\mathbb{C}^N)$ is the linear map defined as

$$\mathcal{I}_{\mathrm{L}(\mathbb{C}^N)}[A] = A \tag{2.26}$$

for all $A \in \mathrm{L}(\mathbb{C}^N)$. Note that we use square bracket to denote the actions of linear maps on operators.

For $\mathcal{M}_1 \in \mathrm{T}(\mathbb{C}^{N_1}, \mathbb{C}^{M_1}), \ldots, \mathcal{M}_k \in \mathrm{T}(\mathbb{C}^{N_k}, \mathbb{C}^{M_k})$, the *tensor product* of these linear maps

$$\mathcal{M}_1 \otimes \cdots \otimes \mathcal{M}_k \in \mathrm{T}(\mathbb{C}^{N_1} \otimes \cdots \otimes \mathbb{C}^{N_k}, \mathbb{C}^{M_1} \otimes \cdots \otimes \mathbb{C}^{M_k}) \tag{2.27}$$

is defined as the unique linear map satisfying

$$(\mathcal{M}_1 \otimes \cdots \otimes \mathcal{M}_k)[A_1 \otimes \cdots \otimes A_k] = \mathcal{M}_1[A_1] \otimes \cdots \otimes \mathcal{M}_k[A_k], \tag{2.28}$$

for all $A_1 \in \mathrm{L}(\mathbb{C}^{N_1}), \ldots, A_k \in \mathrm{L}(\mathbb{C}^{N_k})$.

In this thesis, the following classes of linear maps are considered:

1. *Hermitian-preserving maps.* A map $\mathcal{M} \in \mathrm{T}(\mathbb{C}^N, \mathbb{C}^M)$ is *Hermitian-preserving* if it holds that $\mathcal{M}[A]$ is Hermitian for every Hermitian operator $A \in \mathrm{L}(\mathbb{C}^N)$.

2. *Completely positive maps.* A map $\mathcal{M} \in \mathrm{T}(\mathbb{C}^N, \mathbb{C}^M)$ is *completely positive* if it holds that

$$\left(\mathcal{M} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^{M'})}\right)[A] \tag{2.29}$$

   is positive semidefinite for every positive semidefinite operator $A \in \mathrm{L}(\mathbb{C}^N \otimes \mathbb{C}^{M'}, \mathbb{C}^M \otimes \mathbb{C}^{M'})$, and for every Hilbert space $\mathbb{C}^{M'}$.

3. *Trace-preserving maps.* A map $\mathcal{M} \in \mathrm{T}(\mathbb{C}^N, \mathbb{C}^M)$ is *trace-preserving* if it holds that

$$\mathrm{Tr}(\mathcal{M}[A]) = \mathrm{Tr}(A) \tag{2.30}$$

   for all $A \in \mathrm{L}(\mathbb{C}^N)$.

The *partial trace* is a linear map defined as

$$(\mathrm{Tr} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^M)})[A \otimes B] = \mathrm{Tr}(A)B \tag{2.31}$$

for all operators $A \in \mathrm{L}(\mathbb{C}^N)$ and $B \in \mathrm{L}(\mathbb{C}^M)$. Here, we consider the trace function as a linear map: $\mathrm{Tr} \in \mathrm{T}(\mathbb{C}^N, \mathbb{C})$. Sometimes, it is fruitful to give Hilbert spaces names. If we use S to refer to the Hilbert $\mathbb{C}^N$ of the first system, and use E to refer to the Hilbert space $\mathbb{C}^M$ of the second system, the partial trace can be denoted by Tr_{S} as

$$\mathrm{Tr}_{\mathrm{S}} = \mathrm{Tr} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^M)}. \tag{2.32}$$

Similarly, the map $\mathrm{Tr_E}$ can be defined as

$$\mathrm{Tr_E} = \mathcal{I}_{\mathrm{L}(\mathbb{C}^N)} \otimes \mathrm{Tr}. \tag{2.33}$$

The following two norms of linear maps will be used in this thesis:

1. The *induced trace norm* of a map $\mathcal{M} \in \mathrm{T}(\mathbb{C}^N, \mathbb{C}^M)$ is defined as

$$\|\mathcal{M}\|_1 = \max\left\{\|\mathcal{M}[A]\|_1 : A \in \mathrm{L}(\mathbb{C}^N), \|A\|_1 \leq 1\right\}. \tag{2.34}$$

   This norm is also known as the $1 \to 1$ norm.

2. The *diamond norm* of a map $\mathcal{M} \in \mathrm{T}(\mathbb{C}^N, \mathbb{C}^M)$ is defined as

$$\|\mathcal{M}\|_\diamond = \left\|\mathcal{M} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^N)}\right\|_1. \tag{2.35}$$

   This norm is also known to as the *completely bounded trace norm*.

Next, we discuss different representations of linear maps. Readers may refer to [Wat18] for a thorough discussion of these representations such as their existence and their relationship.

1. *Kraus representation.* A map $\mathcal{M} \in \mathrm{T}(\mathbb{C}^N, \mathbb{C}^M)$ can be represented as

$$\mathcal{M}[A] = \sum_{j=0}^{k-1} E_j A F_j^\dagger, \tag{2.36}$$

   for all $A \in \mathrm{L}(\mathbb{C}^N)$, where the operators $E_j, F_j \in \mathrm{L}(\mathbb{C}^N, \mathbb{C}^M)$ for $j \in \{0, \ldots, k-1\}$ are called the *Kraus operators*.

2. *Stinespring representation.* A map $\mathcal{M} \in \mathrm{T}(\mathbb{C}^N, \mathbb{C}^M)$ can be represented as

$$\mathcal{M}[A] = \mathrm{Tr_B}[EAF^\dagger], \tag{2.37}$$

   for all $A \in \mathrm{L}(\mathbb{C}^N)$, and for any Hilbert space $\mathbb{C}^{M'}$ referred to as B, where the operators $E, F \in \mathrm{L}(\mathbb{C}^N, \mathbb{C}^M \otimes \mathbb{C}^{M'})$ are referred to as the *Stinespring dilation*.

3. The *Choi matrix.* A map $\mathcal{M} \in \mathrm{T}(\mathbb{C}^N, \mathbb{C}^M)$ can be associated with a unique matrix $J(\mathcal{M})$ as

$$J(\mathcal{M}) = \left(\mathcal{M} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^N)}\right)\left[\sum_{j,k=0}^{N-1} |j\rangle\langle k| \otimes |j\rangle\langle k|\right]. \tag{2.38}$$

   The matrix $J(\mathcal{M})$ is called the *Choi matrix* for $\mathcal{M}$.

A map $\mathcal{M} \in \mathrm{T}(\mathbb{C}^N, \mathbb{C}^M)$ is a *quantum channel* if $\mathcal{M}$ is completely positive and trace-preserving. The collection of all such quantum channels is denoted by $\mathrm{C}(\mathbb{C}^N, \mathbb{C}^M)$ (and we use $\mathrm{C}(\mathbb{C}^N)$ as a shorthand for $\mathrm{C}(\mathbb{C}^N, \mathbb{C}^N)$. A quantum channel $\mathcal{M} \in \mathrm{C}(\mathbb{C}^N, \mathbb{C}^M)$ has a Kraus representation of the form

$$\mathcal{M}[A] = \sum_{j=0}^{k-1} E_j A E_j^\dagger, \tag{2.39}$$

for all $A \in \mathrm{L}(\mathbb{C}^N)$, where the Kraus operators $E_0, \ldots, E_{k-1} \in \mathrm{L}(\mathbb{C}^N, \mathbb{C}^M)$ satisfy

$$\sum_{j=0}^{k-1} E_j^\dagger E_j = I. \tag{2.40}$$

A Stinespring representation of $\mathcal{M}$ is of the form

$$\mathcal{M}[A] = \mathrm{Tr}_{\mathrm{B}}[UAU^\dagger], \tag{2.41}$$

for all $A \in \mathrm{L}(\mathbb{C}^N)$, and for any Hilbert space $\mathbb{C}^{M'}$ referred to as B, where $U \in \mathrm{L}(\mathbb{C}^N, \mathbb{C}^M \otimes \mathbb{C}^{M'})$ is an isometry. Quantum channels can be used to model some quantum systems that has interaction with some external resources, such as probability distribution, dissipation, and noise.

2.2 Properties of finite fields

In mathematics, a *field* is informally defined as a set that is closed under addition, multiplication, subtraction, and division with certain axioms. Examples of fields are real numbers, rational number, and complex numbers. More formally, we give the following precise definition:

Definition 2.1. *A field is a set* F *together with two operations, addition (denoted by* $+$*) and multiplication (denoted by* $\cdot$*), such that the following properties are satisfied for all* $a, b, c \in \mathrm{F}$*:*

1. *Closure of* F *under addition and multiplication:* $a + b \in \mathrm{F}$ *and* $a \cdot b \in \mathrm{F}$*.*
2. *Associativity of addition and multiplication:* $a + (b + c) = (a + b) + c$ *and* $a \cdot (b \cdot c) = (a \cdot b) \cdot c$*.*

3. *Commutativity of addition and multiplication:* $a + b = b + a$ *and* $a \cdot b = b \cdot a$.

4. *Additive and multiplicative identity: there exists an element in* F*, denoted by 0, such that* $a + 0 = a$*; there exists an element in* F*, denoted by 1, such that* $a \cdot 1 = a$.

5. *Additive and multiplicative inverse: there exists an element in* F*, denoted by* $-a$*, such that* $a + (-a) = 0$*; there exists an element in* F*, denoted by* a^{-1}*, such that* $a \cdot a^{-1} = 1$.

6. *Distributivity of multiplication over addition:* $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$.

Note that in the above definition, subtraction and division are implicitly defined by additive and multiplicative inverse. We often use the simplified notation ab instead of $a \cdot b$ to denote the multiplication of two elements in a field.

A field with finitely many elements is called a *finite field* (often referred to as *Galois field*). More information about these fields can be found in [LN94]. Let $\mathrm{GF}(2^n)$ denote the finite field of size 2^n. The elements of this field form a vector space over $\mathrm{GF}(2)$ so the notion of a basis of of this field is well-defined: a set $\{\omega_1, \ldots, \omega_n\} \subseteq \mathrm{GF}(2^n)$ is a *basis* if the elements in this set are linearly independent and span the field. With a basis, it is possible to associate the elements of $\mathrm{GF}(2^n)$ with n-bit strings by taking coordinates with respect to this basis. For example, if $a = a_1\omega_1 + a_2\omega_2 + \cdots + a_n\omega_n \in \mathrm{GF}(2^n)$, where $a_1, \ldots, a_n \in \{0, 1\}$, we can associate a with the n-bit string $a_1 \cdots a_n$.

A *polynomial basis* of $\mathrm{GF}(2^n)$ is a basis that is of the form $\{1, \alpha, \alpha^2, \ldots, \alpha^{n-1}\}$ for some $\alpha \in \mathrm{GF}(2^n)$. The standard constructions of $\mathrm{GF}(2^n)$ in terms of irreducible polynomials result in a representation with respect to a polynomial basis. However, there exist bases that are not necessarily of this form. For instance, a *normal basis* of $\mathrm{GF}(2^n)$ has the form $\{\alpha^{2^0}, \alpha^{2^1}, \ldots, \alpha^{2^{n-1}}\}$ for some $\alpha \in \mathrm{GF}(2^n)$ which we call a *normal element*. We define the *field-trace* function $T : \mathrm{GF}(2^n) \to \mathrm{GF}(2)$ as

$$T(a) = a^{2^0} + a^{2^1} + \cdots + a^{2^{n-1}}. \tag{2.42}$$

The field-trace function is linear in the sense that $T(a + b) = T(a) + T(b)$, for all $a, b \in \mathrm{GF}(2^n)$. In terms of T, we define the *field-trace inner product* of $a, b \in \mathrm{GF}(2^n)$ as $T(ab)$. Now, we define a notion of the dual of a basis. For an arbitrary basis $\{\omega_1, \ldots, \omega_n\} \subseteq \mathrm{GF}(2^n)$, that we refer to as the *primal basis*, we can define its *dual basis* as the unique $\{\hat{\omega}_1, \ldots, \hat{\omega}_n\} \subseteq \mathrm{GF}(2^n)$ such that

$$T(\omega_j \hat{\omega}_k) = \begin{cases} 1 & \text{if } j = k \\ 0 & \text{if } j \neq k. \end{cases} \tag{2.43}$$

We use the following notation to distinguish the representations with respect to a primal basis and its dual basis. With respect to any primal basis $\{\omega_1, \ldots, \omega_n\}$ and its dual $\{\hat{\omega}_1, \ldots, \hat{\omega}_n\}$, for any $a \in \mathrm{GF}(2^n)$,

- $\lceil a \rceil \in \{0,1\}^n$ denotes the coordinates of a in the primal basis, i.e., $a = \lceil a \rceil_1 \omega_1 + \cdots + \lceil a \rceil_n \omega_n$, where $\lceil a \rceil_j = T(a\hat{\omega}_j)$ for all $j \in \{1, \ldots, n\}$.
- $\lfloor a \rfloor \in \{0,1\}^n$ denotes the coordinates of a in the dual basis, i.e., $a = \lfloor a \rfloor_1 \hat{\omega}_1 + \cdots + \lfloor a \rfloor_n \hat{\omega}_n$, where $\lfloor a \rfloor_j = T(a\omega_j)$ for all $j \in \{1, \ldots, n\}$.

When the meaning is clear from the context, it is convenient to write a in place of $\lceil a \rceil$. Also, it is convenient to identify n-bit binary strings with $\{0,1\}$-valued column vectors of length n. Thus, we can view $\lceil a \rceil$ and $\lfloor a \rfloor$ as binary column vectors of length n. In this thesis, binary matrices acting on these vectors (in mod 2 arithmetic) are written with square brackets.

The conversion from primal to dual basis coordinates corresponds to multiplication by the $n \times n$ binary matrix

$$W = \begin{bmatrix} T(\omega_1\omega_1) & \cdots & T(\omega_1\omega_n) \\ \vdots & \ddots & \vdots \\ T(\omega_n\omega_1) & \cdots & T(\omega_n\omega_n) \end{bmatrix}. \tag{2.44}$$

That is, $\lfloor a \rfloor = W\lceil a \rceil$ (with matrix-vector multiplication in mod 2 arithmetic). This can be verified by the action of W on each element of the primal basis. Let $a = \omega_j$. Then the vector $\lceil a \rceil$ is a vector with 1 in the j-th entry. The k-th entry of $W\lceil a \rceil$ is $T(\omega_k\omega_j)$, and hence we have $W\lceil a \rceil = \lfloor a \rfloor$. We also note that $T(ab)$ is the dot-product of the coordinates of a in the primal basis and the coordinates of b in the dual basis:

$$T(ab) = \lceil a \rceil \cdot \lfloor b \rfloor = \lceil a \rceil_1 \lfloor b \rfloor_1 + \cdots + \lceil a \rceil_n \lfloor b \rfloor_n \mod 2. \tag{2.45}$$

The dual of the dual basis is the primal basis. A basis is *self-dual* if $\omega_j = \hat{\omega}_j$ for all $j \in \{1, \ldots, n\}$.

With respect to any basis, multiplication by any particular $r \in \mathrm{GF}(2^n)$ is a linear operator in the following sense. There exists a binary $n \times n$ matrix M_r such that, for all $s \in \mathrm{GF}(2^n)$, $\lceil rs \rceil = M_r \lceil s \rceil$ (with matrix-vector multiplication in mod 2 arithmetic). Concretely, this matrix M_r is

$$M_r = \begin{bmatrix} T(r\hat{\omega}_1\omega_1) & \cdots & T(r\hat{\omega}_1\omega_n) \\ \vdots & \ddots & \vdots \\ T(r\hat{\omega}_n\omega_1) & \cdots & T(r\hat{\omega}_n\omega_n) \end{bmatrix}, \tag{2.46}$$

and its transpose M_r^T corresponds to multiplication by r in the dual basis (i.e., $\lfloor rs \rfloor = M_r^T \lfloor s \rfloor$). It should be noted that the algorithms for multiplication in GF(2^n) are basis dependent; the obvious cost of converting between any two bases is $O(n^2)$.

Chapter 3

An Example of Open Quantum Systems: Unitary 2-Designs

In this chapter, we study an example of open quantum systems in the sense of probability distribution of unitaries: unitary 2-designs. First, we give definitions of unitary 2-designs and show their connections. Then, we show efficient quantum algorithms for constructing unitary 2-designs. An overview of this subject is presented in Section 1.1. This chapter is based on [CLLW16].

3.1 Previous work and main results

Previous research on unitary 2-designs was considered for different applications. In the context of bilateral twirl (applying a Haar-random bilateral unitary on a bipartite system) [DLT02] and channel twirl (applying a Haar-random unitary U on the input state of some quantum channel and then applying $U^\dagger$ on the output state) [DCEL09], it has been shown that the uniform distribution over the Clifford group is an exact unitary 2-design. This implies a construction of unitary 2-designs with $O(n^2/\log n)$ 1- and 2-qubit gates from the Clifford group [AG04]. Moreover, the sampling cost of this construction is $O(n^2)$ bits of entropy.

Approximate unitary 2-designs were also studied in previous research. Let ϵ (or ϵ') be the distance of the resulting operation from the ideal one. Based on a certain process of random circuit generation (introduced in [EWS+03]), Harrow and Low [HL09] gave a

construction of ϵ-approximate unitary 2-designs (in the context of bilateral twirl) of gate complexity $O(n(n + \log 1/\epsilon))$.

Dankert *et al.* [DCEL09] gave another construction of ϵ'-approximate unitary 2-designs (in the context of channel twirl) of gate complexity $O(n \log 1/\epsilon')$; however, as pointed out in [HL09] and [BF15], this construction could potentially incur a blow-up by a factor that is exponential in n in the context of bilateral twirl, due to the notion of approximation. To the best of our knowledge, we need $\epsilon' \leq \epsilon/2^{2n}$. Thus, the gate complexity of this construction becomes $O(n(n + \log 1/\epsilon))$.

In the context of bilateral twirl, all of the above constructions for both exact and approximate unitary 2-designs incur circuits of size $\widetilde{\Omega}(n^2)$ and require $\Omega(n^2)$ random bits of entropy.

It is proven by Chau [Cha05] that a small subgroup of the Clifford group is sufficient for constructing an exact unitary 2-design with a sampling cost of approximately $5n$ random bits of entropy; however, other than the $O(n^2/\log n)$ bound that holds for constructing a unitary in the Clifford group, the gate complexity of this construction is unknown. The necessary and sufficient entropy for exact and approximate unitary 2-designs was studied by [GAE07] and [RS09]: approximately $4n$ random bits of entropy are necessary.

Brown and Fawzi [BF15] gave a method to generate random circuits of gate complexity $O(n \log^2 n)$ and circuit depth $O(\log^3 n)$. Although this construction does not imply a unitary 2-design, it can be used for decoupling and building quantum error correcting codes with small encoding circuits [BF13]. No ancilla qubits are needed in their approach; however, the circuit depth is higher, and a considerable amount of analysis is required to show that their construction achieves the tasks with the desired gate complexity and accuracy. It may require additional analysis to adapt their construction to other applications.

In this remainder of this chapter, we first give a new characterization (Definition 3.2) of unitary 2-designs in terms of 2-query indistinguishability that may be of independent interest. Then we present three constructions of *exact* unitary 2-designs on n qubits with the following gate complexity:

1. $O(n \log n \log\log n)$ 1- and 2-qubit gates (all Clifford gates) for infinitely many n, assuming the *extended Riemann Hypothesis* is true.

2. $O(n \log n \log\log n)$ 1- and 2-qubit gates (including non-Clifford gates) for all n, unconditionally.

3. $O(n \log^2 n \log\log n)$ 1- and 2-qubit gates (all Clifford gates) for all n, unconditionally.

By the fact that efficient multiplication/convolution algorithms can be performed with circuit depth $O(\log n)$ [Sch77], the circuit depth is $O(\log n)$ for the first two constructions and $O(\log^2 n)$ for the third construction. In Subsection 3.4.3, we show that to construct any exact or approximate unitary 2-design, a high probability set of unitaries have size $\Omega(n)$ and circuit depth $\Omega(\log n)$, which implies that the above constructions are nearly optimal.

For the three above constructions, we need to sample from a uniform distribution on a set of size $2^{5n} - 2^{3n}$, and hence they use $5n$ bits of randomness. All three constructions consist of Clifford unitaries (in the second construction, non-Clifford gates are used to compute Clifford unitaries efficiently). The circuits use $\widetilde{O}(n)$ ancilla qubits where the initial state of each ancilla qubit is $|0\rangle$ and it will be restored at the end of the computation. The cost of the classical process that outputs a description of the quantum circuit is polynomial in n, which is dominated by the complexity of computing square roots in the finite field of size 2^n.

3.2 Definitions of unitary 2-designs

In this section, we discuss several definitions of unitary 2-designs and the applications where the corresponding definition arises.

The definitions of unitary 2-designs are closely related to distributions over the unitary group $\mathrm{U}(\mathbb{C}^N)$, especially the Haar measure on $\mathrm{U}(\mathbb{C}^N)$, which is the unique measure on $\mathrm{U}(\mathbb{C}^N)$ that is invariant under left and right multiplication by any $U \in \mathrm{U}(\mathbb{C}^N)$. For any $U \in \mathrm{U}(\mathbb{C}^N)$, we use $\mu(U)$ to denote the Haar measure of U on $\mathrm{U}(\mathbb{C}^N)$.

Sampling from the Haar measure is a powerful tool in quantum information theory. For example, in applications such as estimating fidelity [DCEL09], and quantum data hiding [DLT02], a physical procedure that averages over such random choices of unitary operators is used. In the proof of quantum channel capacity [Dev05, HHWY08, Llo97, Sho], a randomized argument is used to evaluate the average performance over all possible unitary encodings.

However, a Haar-random unitary is prohibitively hard to implement, as an exponential number of random bits are required. There are other resources for implementing a Haar-random unitary, such as shared randomness, and communication, that are desired to be reduced. Therefore, we are interested in contexts in which such sampling from the Haar measure can be replaced by sampling from a probability distribution over a finite set of

unitaries $\{U_1, U_2, \ldots, U_k \in \mathrm{U}(\mathbb{C}^N)\}$ with corresponding probability p_j with $\sum_j p_j = 1$. We refer to this probability distribution as an *ensemble* and denote it by $\mathcal{E} = \{p_j, U_j\}_{j=1}^k$.

In the first context, we consider the expected value of polynomials of the entries of unitary matrices sampled from some ensemble. This gives rise to the original definition of unitary 2-designs in [DCEL09], and provides results in other contexts. We give the definition of an ensemble that is *degree-2 expectation preserving* as the following.

Definition 3.1. *We say that an ensemble $\mathcal{E}$ is* degree-2 expectation preserving *if, for every polynomial $\gamma(U)$ of degree at most 2 in the matrix elements of U and at most 2 in the matrix elements of $U^\dagger$, it holds that*

$$\sum_{j=1}^{k} p_j \gamma(U_j) = \int_{\mathrm{U}(\mathbb{C}^N)} d\mu(U) \gamma(U). \tag{3.1}$$

The second context is related to the quantum analogue of universal 2-hash functions [CW79]: we consider the task of distinguishing whether a random sample U is drawn from the Haar measure or from some ensemble $\mathcal{E}$. We allow an arbitrary distinguishing circuit that makes a total of at most two queries of U or $U^\dagger$. The most general circuit of this form is depicted in Figure 3.1.

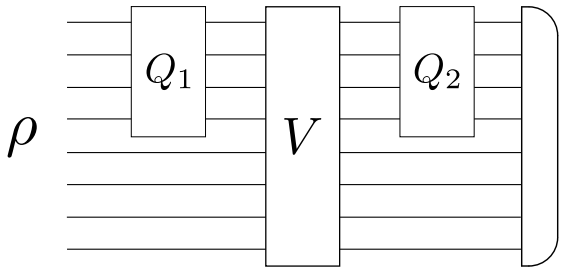

Figure 3.1: Illustration of a 2-*query distinguishing circuit*. The first query Q_1 can be U or $U^\dagger$, likewise for the second query Q_2. The initial state ρ is arbitrary, V is an arbitrary unitary, and the final measurement is also arbitrary and outputs one bit.

Let $\mathbb{C}^N$ and $\mathbb{C}^M$ be two Hilbert spaces. The *2-query distinguishing* circuit consists of a unitary $Q_1 \in \mathrm{U}(\mathbb{C}^N)$ which can be either U or $U^\dagger$, an arbitrary unitary $V \in \mathrm{U}(\mathbb{C}^N \otimes \mathbb{C}^M)$, and a unitary $Q_2 \in \mathrm{U}(\mathbb{C}^N)$ which can be either U or $U^\dagger$, followed by an arbitrary measurement. We use the measurement outcome to distinguish whether U is drawn from the Haar measure or not. For an arbitrary initial state ρ and for any U (either drawn from

the Haar measure or from an ensemble) the quantum state right before the measurement, denoted by σ_U is

$$\sigma_U = (Q_2 \otimes I_{\mathbb{C}^M})V(Q_1 \otimes I_{\mathbb{C}^M})\rho(Q_1^\dagger \otimes I_{\mathbb{C}^M})V^\dagger(Q_2^\dagger \otimes I_{\mathbb{C}^M}). \tag{3.2}$$

If U is drawn from an ensemble $\mathcal{E}$, the quantum state right before the measurement is $\sum_{j=1}^k p_j \sigma_U$; similarly, if U is drawn from the Haar measure, the quantum state right before the measurement is $\int_{\mathrm{U}(\mathbb{C}^N)} d\mu(U)\sigma_U$. The outcome of the measurement has the same distribution regardless of whether U is drawn from a Haar measure or from an ensemble, if and only if the above two quantum states are equal. In the following definition, we describe ensembles that cannot be distinguished from the Haar measure by any 2-query distinguishing circuit.

Definition 3.2. *We say that $\mathcal{E}$ is* 2-query indistinguishable*, if, for any 2-query distinguishing circuit and for any initial state ρ, it holds that*

$$\sum_{j=1}^k p_j \sigma_U = \int_{\mathrm{U}(\mathbb{C}^N)} d\mu(U)\sigma_U. \tag{3.3}$$

The third context is a special case of a 2-query distinguishing circuit, where U is queried twice in parallel, as illustrated in Figure 3.2. Let us consider bipartite operations where the same unitary drawn from some ensemble or the Haar measure is applied on two disjoint systems. These operations are often called bilateral twirls [BDSW96, DLT02]. The $\mathcal{E}$ *bilateral twirl*, denoted by $\mathcal{T}_\mathcal{E} \in \mathrm{T}(\mathbb{C}^N)$, is a linear map defined as

$$\mathcal{T}_\mathcal{E}[\rho] = \sum_{j=1}^k p_j (U_j \otimes U_j)\rho(U_j^\dagger \otimes U_j^\dagger). \tag{3.4}$$

Likewise, the *full bilateral twirl*, denoted by $\mathcal{T}_\mu \in \mathrm{T}(\mathbb{C}^N)$, is defined as the linear map

$$\mathcal{T}_\mu[\rho] = \int_{\mathrm{U}(\mathbb{C}^N)} d\mu(U)(U \otimes U)\rho(U^\dagger \otimes U^\dagger). \tag{3.5}$$

The full bilateral twirl is motivated operationally by applications such as error correcting [BDSW96] and data hiding [DLT02], and it is used in various mathematical proofs in quantum information [HHWY08, SDTR13]. In the following definition, we describe ensembles that derandomize the full bilateral twirl.

Definition 3.3. *We say that the ensemble $\mathcal{E}$* implements the full bilateral twirl *if $\mathcal{T}_\mathcal{E}[\rho] = \mathcal{T}_\mu[\rho]$ for all $\rho \in \mathrm{D}(\mathbb{C}^N)$.*

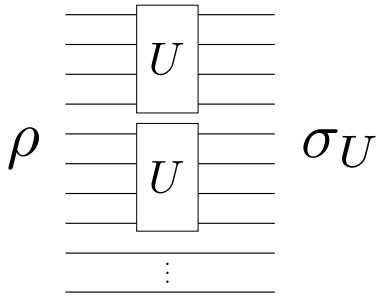

Figure 3.2: Illustration of the *bilateral twirl*: querying U twice in parallel. The initial sate ρ is arbitrary.

The fourth context arises in the task of averaging any quantum channel with a depolarizing channel. This averaging process has many important applications, such as *benchmarking* (for estimating average channel fidelity) of quantum devices [DCEL09] and error estimation (for detecting eavesdropping) in quantum key distribution [Cha05]. This process can be realized by a quantum channel which is often referred to as a channel twirl.

Let $\mathcal{M} \in \mathrm{C}(\mathbb{C}^N)$ be any quantum channel. An $\mathcal{E}$ *channel twirl* of $\mathcal{M}$, denoted by $\mathcal{M}_{\mathcal{E}}$, is defined as the quantum channel

$$\mathcal{M}_{\mathcal{E}}[\rho] = \sum_{j=i}^{k} p_j U_j^\dagger \mathcal{M}[U_j \rho U_j^\dagger] U_j. \tag{3.6}$$

Operationally, a random change of basis is applied to the system before applying the channel $\mathcal{M}$ and it is reverted afterward. Likewise, a *full channel twirl* of $\mathcal{M}$, denoted by $\mathcal{M}_\mu$, is defined as the quantum channel

$$\mathcal{M}_\mu[\rho] = \int_{\mathrm{U}(\mathbb{C}^N)} d\mu(U) U^\dagger \mathcal{M}[U \rho U^\dagger] U. \tag{3.7}$$

In the following definition, we describe ensembles that derandomize the full channel twirl.

Definition 3.4. *We say that* $\mathcal{E}$ implements the full channel twirl *if* $\mathcal{M}_{\mathcal{E}} = \mathcal{M}_\mu$ *for all* $\mathcal{M} \in \mathrm{C}(\mathbb{C}^N)$.

In the following lemma, we show that these four relationships between ensembles and the Haar measure are equivalent. Therefore, we can think of an ensemble satisfying one of the conditions in different ways.

Lemma 3.1. *Let $\mathcal{E}$ be any ensemble of unitaries in $\mathrm{U}(\mathbb{C}^N)$. The following statements are equivalent:*

1. *$\mathcal{E}$ is degree-2 expectation preserving.*
2. *$\mathcal{E}$ is 2-query indistinguishable.*
3. *$\mathcal{E}$ implements the full bilateral twirl.*
4. *$\mathcal{E}$ implements the full channel twirl.*

In addition, we have the following corollary of Lemma 3.1, which is not obvious from Definitions 3.3 and 3.4 alone.

Corollary 3.2. *For an ensemble $\mathcal{E} = \{p_j, U_j\}_{j=1}^k$, let $\mathcal{E}^\dagger := \{p_j, U_j^\dagger\}_{j=1}^k$. The following statements hold:*

1. *$\mathcal{E}$ implements the full bilateral twirl if and only if $\mathcal{E}^\dagger$ does.*
2. *$\mathcal{E}$ implements the full channel twirl if and only if $\mathcal{E}^\dagger$ does.*

Due to Lemma 3.1, we can just refer to an ensemble satisfying any one of the four conditions as a "unitary 2-design" when we do not need to specify the context.

Note that additional definitions have been discussed in literature [GAE07, RS09, HL09, Low09]. Some equivalence relations in Lemma 3.1 have been proven in previous work [DCEL09, HL09, Low09]. In particular, the relation between Statements 1, 3, and 4 with bounds on the approximations has been shown in [Low09]. In the following, we provide a complete (alternative) proof of Lemma 3.1 and Corollary 3.2.

Proof of Lemma 3.1. We show the following implications of the four statements, which are sufficient to prove the equivalence:

$$(1) \Rightarrow (2) \Rightarrow (3) \Rightarrow (1)$$
$$(2) \Rightarrow (4) \Rightarrow (3)$$

(1) $\Rightarrow$ (2): Consider any distinguishing circuit making up to two queries of U or $U^\dagger$. The output state σ_U is a product of matrices with at most two factors of U and two factors of $U^\dagger$. Thus, each entry of σ_U is a polynomial of degree at most 2 in the matrix elements of

U and at most 2 in the matrix elements of $U^\dagger$. By hypothesis, $\mathcal{E}$ is degree-2 expectation preserving. Thus, the following holds entry-wise:

$$\sum_{j=1}^{k} p_j \sigma_{U_j} = \int_{\mathrm{U}(\mathbb{C}^N)} d\mu(U) \sigma_U. \tag{3.8}$$

It follows that $\mathcal{E}$ is 2-query indistinguishable.

(2) $\Rightarrow$ (3): This implication follows from the fact that the bilateral twirl circuit is a special case of a 2-query distinguishing circuit.

(3) $\Rightarrow$ (1): Let $\{|0\rangle, \ldots, |N-1\rangle\}$ be the computational basis for $\mathbb{C}^N$. Suppose $\mathcal{E}$ implements the full bilateral twirl. Then, we have

$$\sum_{j=1}^{k} p_j U_j \otimes U_j \rho U_j^\dagger \otimes U_j^\dagger = \int_{\mathrm{U}(\mathbb{C}^N)} d\mu(U) U \otimes U \rho U^\dagger \otimes U^\dagger, \tag{3.9}$$

for all density matrices $\rho \in \mathrm{D}(\mathbb{C}^N)$. Since the density matrices span the Hilbert space of all possible square complex matrices of the same dimension, the above equation holds if we replace ρ by $|a_1\rangle\langle a_3| \otimes |a_2\rangle\langle a_4|$ for all $a_1, a_2, a_3, a_4 \in \{0, \ldots, N-1\}$. Furthermore, we left- and right-multiply the above by $\langle a_5| \otimes \langle a_6|$ and $|a_7\rangle \otimes |a_8\rangle$, respectively. This gives

$$\begin{aligned} &\sum_{j=1}^{k} p_j \langle a_5|U_j|a_1\rangle\langle a_6|U_j|a_2\rangle\langle a_3|U_j^\dagger|a_7\rangle\langle a_4|U_j^\dagger|a_8\rangle \\ &\qquad = \int_{\mathrm{U}(\mathbb{C}^N)} d\mu(U) \langle a_5|U|a_1\rangle\langle a_6|U|a_2\rangle\langle a_3|U^\dagger|a_7\rangle\langle a_4|U^\dagger|a_8\rangle. \end{aligned} \tag{3.10}$$

Repeating the above for all possible $a_1, \ldots, a_8$ and by linearity, Eq. (3.1) follows. Hence, $\mathcal{E}$ is degree-2 expectation preserving.

(2) $\Rightarrow$ (4): This implication follows from the fact that the channel twirl circuit is a special case of a 2-query distinguishing circuit.

(4) $\Rightarrow$ (3): Suppose for every quantum channel $\mathcal{M} \in \mathrm{C}(\mathbb{C}^N)$, $\mathcal{M}_\mathcal{E} = \mathcal{M}_\mu$. Then, they have the same Choi matrix, i.e., $J(\mathcal{M}_\mathcal{E}) = J(\mathcal{M}_\mu)$. Rephrasing this equality using Eqns. (3.6)

and (3.7), we have

$$\sum_{j=1}^{k} p_j (U_j^\dagger \otimes I) \left(\mathcal{M} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^N)}\right) \left[(U_j \otimes I) \left(\sum_{\ell,k=0}^{N-1} |\ell\rangle\langle k| \otimes |\ell\rangle\langle k| \right) (U_j^\dagger \otimes I) \right] (U_j \otimes I)$$
$$= \int_{\mathrm{U}(\mathbb{C}^N)} d\mu(U) (U^\dagger \otimes I) \left(\mathcal{M} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^N)}\right) \left[(U \otimes I) \left(\sum_{\ell,k=0}^{N-1} |\ell\rangle\langle k| \otimes |\ell\rangle\langle k| \right) (U^\dagger \otimes I) \right] (U \otimes I). \tag{3.11}$$

For each side of the above equation, we use three steps to turn the Choi matrix of the channel twirl into the bilateral twirl of an operator that is closely related to the Choi matrix of $\mathcal{M}$. First, for the LHS of Eq. (3.11), we apply the transpose trick

$$(U_j \otimes I) \left(\sum_{\ell,k=0}^{N-1} |\ell\rangle\langle k| \otimes |\ell\rangle\langle k| \right) (U_j^\dagger \otimes I) = (I \otimes U_j^T) \left(\sum_{\ell,k=0}^{N-1} |\ell\rangle\langle k| \otimes |\ell\rangle\langle k| \right) (I \otimes U_j^*). \tag{3.12}$$

Second, we commute the conjugation by $(I \otimes U_j^T)$ with $\mathcal{M} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^N)}$. We apply similar manipulations on the RHS of Eq (3.11). Then, Eq. (3.11) becomes

$$\sum_{j=1}^{k} p_j (U_j^\dagger \otimes U_j^T) \left(\mathcal{M} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^N)}\right) \left[\sum_{\ell,k=0}^{N-1} |\ell\rangle\langle k| \otimes |\ell\rangle\langle k| \right] (U_j \otimes U_j^*)$$
$$= \int_{\mathrm{U}(\mathbb{C}^N)} d\mu(U) (U^\dagger \otimes U^T) \left(\mathcal{M} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^N)}\right) \left[\sum_{\ell,k=0}^{N-1} |\ell\rangle\langle k| \otimes |\ell\rangle\langle k| \right] (U \otimes U^*). \tag{3.13}$$

Third, we apply to Eq. (3.13) the *partial transpose* of the second system: for any $A_1, A_2 \in \mathrm{L}(\mathbb{C}^N)$, this linear map takes $A_1 \otimes A_2$ to $A_1 \otimes A_2^T$. Let $\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N} \in \mathrm{L}(\mathbb{C}^N \otimes \mathbb{C}^N)$ be the swap operator, i.e., $\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N} = \sum_{k,\ell=0}^{N-1} |\ell\rangle\langle k| \otimes |k\rangle\langle \ell|$. Then, the partial transpose of

$$(I \otimes U_j^T) \left(\sum_{\ell,k=0}^{N-1} |\ell\rangle\langle k| \otimes |\ell\rangle\langle k| \right) (I \otimes U_j^*) = \sum_{\ell,k=0}^{N-1} |\ell\rangle\langle k| \otimes (U_j^T |\ell\rangle\langle k| U_j^*) \tag{3.14}$$

is equal to

$$\sum_{\ell,k=0}^{N-1} |\ell\rangle\langle k| \otimes (U_j^\dagger |k\rangle\langle \ell| U_j) = (I \otimes U_j^\dagger) \mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N} (I \otimes U_j). \tag{3.15}$$

Now, Eq. (3.13) becomes

$$\sum_{j=1}^{k} p_j (U_j^\dagger \otimes U_j^\dagger) \left(\mathcal{M} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^N)}\right) [\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}] (U_j \otimes U_j)$$
$$= \int_{\mathrm{U}(\mathbb{C}^N)} d\mu(U) (U^\dagger \otimes U^\dagger) \left(\mathcal{M} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^N)}\right) [\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}] (U \otimes U), \tag{3.16}$$

which is equivalent to

$$\mathcal{T}_{\mathcal{E}^\dagger} \left[\left(\mathcal{M} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^N)}\right) [\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}]\right] = \mathcal{T}_\mu \left[\left(\mathcal{M} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^N)}\right) [\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}]\right], \tag{3.17}$$

which follows from the fact that $d\mu(U) = d\mu(U^\dagger)$. In the above, the transpose trick, the commutation, and the partial transpose applying on Eq. (3.11) transform the equality of the Choi matrices of two channel twirls for $\mathcal{M}$ into the equality of two bilateral twirls of the matrix $\left(\mathcal{M} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^N)}\right) [\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}]$ in the form of Eq. (3.17).

Now, we apply Eq. (3.17) to a set of carefully chosen $\mathcal{M}$'s to show that $\mathcal{T}_{\mathcal{E}^\dagger}[A] = \mathcal{T}_\mu[A]$ for a basis $\{A\}$ for the input space $\mathrm{L}(\mathbb{C}^N \otimes \mathbb{C}^N)$, which will show that $\mathcal{E}^\dagger$ implements the full bilateral twirl. We consider $\mathcal{M}$'s with a specific form. Let $\mathcal{R} \in \mathrm{C}(\mathbb{C}^N)$ be the completely depolarizing channel acting on $\mathrm{L}(\mathbb{C}^N)$, i.e., $\mathcal{R}[\rho] = (\mathrm{Tr}(\rho)) I_{\mathbb{C}^N}/N$ for all $\rho \in \mathrm{L}(\mathbb{C}^N)$. Note that $J(\mathcal{R}) = (I_{\mathbb{C}^N} \otimes I_{\mathbb{C}^N})/N$. Consider any linear map $\widetilde{\mathcal{M}}$ that is trace-preserving and Hermitian-preserving. The latter property implies that $J(\widetilde{\mathcal{M}})$ is Hermitian. Then, for sufficiently small and positive λ, the Choi matrix of $\mathcal{M} = (1-\lambda)\mathcal{R} + \lambda\widetilde{\mathcal{M}}$ is positive semidefinite, because the Choi matrix of $\mathcal{R}$ is proportional to the identity. Therefore, such $\mathcal{M}$ is completely positive (as a linear map is completely positive if and only if its Choi matrix is positive semidefinite [Cho75, Leu03, Wat18]). Because $\mathcal{M}$ is completely positive and trace-preserving, it is a quantum channel. When we apply Eq. (3.17) to such $\mathcal{M}$'s, the $\mathcal{R}$ terms cancel out, because $\left(\mathcal{R} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^N)}\right) [\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}] = (I_{\mathbb{C}^N} \otimes I_{\mathbb{C}^N})/N$, which is invariant under either bilateral twirl. Therefore, Eq. (3.17) holds for all , trace-preserving and Hermitian-preserving linear maps $\widetilde{\mathcal{M}}$, which are easier to construct than quantum channels.

For a basis of the input space $\mathrm{L}(\mathbb{C}^N \otimes \mathbb{C}^N)$, we take $A = G_\ell \otimes G_j$ where $\{G_\ell\}_{\ell=0}^{N^2-1}$ is a basis for $\mathrm{L}(\mathbb{C}^N)$ with the following properties:

1. Each G_ℓ is Hermitian.
2. $G_0 = I_{\mathbb{C}^N}/\sqrt{N}$.

3. For all $\ell, j \in \{0, \ldots, N^2 - 1\}$, $\mathrm{Tr}(G_\ell G_j) = 1$ if $\ell = j$, and $\mathrm{Tr}(G_\ell G_j) = 0$ otherwise. In particular, G_ℓ is traceless for all $\ell > 0$.

4. The swap operator has a simple representation in this basis:

$$\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N} = \sum_{\ell=0}^{N^2-1} G_\ell \otimes G_\ell. \tag{3.18}$$

Such a basis exists for all N. When $N = 2^n$, G_ℓ can be taken to be proportional to the Pauli matrices. For general N, we use the *generalized Gell-Mann matrices* to construct the basis as follows. Let $G_0 = I_{\mathbb{C}^N}/\sqrt{N}$. For $\ell = 1, \ldots, N-1$, let $G_\ell = D_\ell/\sqrt{\ell(\ell+1)}$ where D_ℓ is a diagonal matrix with $(D_\ell)_{0,0} = \cdots = (D_\ell)_{\ell-1,\ell-1} = 1$, $(D_\ell)_{\ell,\ell} = -\ell$, and $(D_\ell)_{j,j} = 0$ for all $\ell + 1 \leq j \leq N$. For $0 \leq j_1 < j_2 \leq N-1$, let $X_{j_1,j_2} = (|j_1\rangle\langle j_2| + |j_2\rangle\langle j_1|)/\sqrt{2}$, and $Y_{j_1,j_2} = i(-|j_1\rangle\langle j_2| + |j_2\rangle\langle j_1|)/\sqrt{2}$. Let $\{G_N, \ldots, G_{N^2-1}\} = \{X_{j_1,j_2}, Y_{j_1,j_2}\}_{0 \leq j_1 < j_2 \leq N-1}$ with any ordering. Then $\{G_\ell\}_{\ell=0}^{N^2-1}$ span $\mathrm{L}(\mathbb{C}^N)$, and each G_ℓ satisfy condition (1) and (3). Finally, the expression for the swap operator $\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}$ can be verified by checking that each of the N^4 matrix entries on the RHS has the value given by the LHS. The verification involves routine arithmetic: each off-diagonal element involves only two terms, and the diagonal elements can be expressed as simple telescopic sums.

Now, we verify that $\mathcal{T}_{\mathcal{E}^\dagger}[G_\ell \otimes G_j] = \mathcal{T}_\mu[G_\ell \otimes G_j]$ for all $0 \leq \ell, j \leq N^2 - 1$ by considering four cases. First, the equality is immediate for $\ell = j = 0$. Second, for each $0 < j \leq N^2 - 1$, consider $\widetilde{\mathcal{M}}_{0,j}$ defined by $\widetilde{\mathcal{M}}_{0,j}[G_0] = G_0 + G_j$, and $\widetilde{\mathcal{M}}_{0,j}[G_\ell] = \mathbf{0}$ for all $\ell \neq 0$. We have that $\widetilde{\mathcal{M}}_{0,j}$ is trace-preserving since each G_ℓ is traceless for $\ell > 0$. Furthermore, $(\widetilde{\mathcal{M}}_{0,j} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^N)})[\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}] = (G_0 + G_j) \otimes G_0$ and partial transposing the second system gives $J(\widetilde{\mathcal{M}}_{0,j})$, which implies that $\widetilde{\mathcal{M}}_{0,j}$ is Hermitian-preserving. Therefore, we apply Eq. (3.17) to $\widetilde{\mathcal{M}}_{0,j}$ and conclude that $\mathcal{T}_{\mathcal{E}^\dagger}[G_j \otimes G_0] = \mathcal{T}_\mu[G_j \otimes G_0]$. Third, because of the symmetry of the bilateral twirl, we have that $\mathcal{T}_{\mathcal{E}^\dagger}[G_0 \otimes G_j] = \mathcal{T}_\mu[G_0 \otimes G_j]$. Fourth, let $0 < j \leq \ell \leq N^2 - 1$ and consider $\widetilde{\mathcal{M}}_{j,\ell}$ such that $\widetilde{\mathcal{M}}_{j,\ell}[G_0] = G_0$, $\widetilde{\mathcal{M}}_{j,\ell}[G_j] = G_\ell$, and $\widetilde{\mathcal{M}}_{j,\ell}[G_{j'}] = \mathbf{0}$ for all $j' \neq 0$ and $j' \neq j$. With arguments similar to the second case, $\mathcal{T}_{\mathcal{E}^\dagger}[G_\ell \otimes G_j] = \mathcal{T}_\mu[G_\ell \otimes G_j]$.

Finally, because of the implications (1) $\Rightarrow$ (2) $\Rightarrow$ (3) $\Rightarrow$ (1), we have established the equivalence between (2) and (3). From Definition 3.2, $\mathcal{E}$ is 2-query indistinguishable if and only if $\mathcal{E}^\dagger$ is. Thus, $\mathcal{E}$ implements the full bilateral twirl if and only if $\mathcal{E}^\dagger$ does. Now, the relation $\mathcal{T}_{\mathcal{E}^\dagger}[G_\ell \otimes G_j] = \mathcal{T}_\mu[G_\ell \otimes G_j]$ implies that $\mathcal{T}_{\mathcal{E}}[G_\ell \otimes G_j] = \mathcal{T}_\mu[G_\ell \otimes G_j]$, which establishes the implication (4) $\Rightarrow$ (3).

□

Proof of Corollary 3.2. Statement 1 has been implicitly shown at the end of the proof of Lemma 3.1. From Definition 3.2, $\mathcal{E}$ is 2-query indistinguishable if and only if $\mathcal{E}^\dagger$ is. Therefore, by the equivalence between statements 2 and 3 of Lemma 3.1, $\mathcal{E}$ implements the full bilateral twirl if and only if $\mathcal{E}^\dagger$ does.

Statement 2 follows from the equivalence between statements 3 and 4 of Lemma 3.1, and statement 1 of this corollary. □

Alternatively, the implication (4) ⇒ (3) in Lemma 3.1 can be proven in a simpler way for the special case when $\mathcal{E} = \{p_j, U_j\}$ is an ensemble with Clifford unitaries and $N = 2^n$. We first show that if $\mathcal{E}$ implements the full channel twirl, it is Pauli mixing, and by Lemma 3.3, it follows that $\mathcal{E}$ implements the full bilateral twirl.

Consider an ensemble $\mathcal{E} = \{p_j, U_j\}$ with Clifford unitaries U_j such that $\mathcal{M}_\mathcal{E} = \mathcal{M}_\mu$ for all quantum channels $\mathcal{M} \in \mathrm{C}(\mathbb{C}^N)$. Define $\mathcal{M}[\rho] = P\rho P^\dagger$ for an arbitrary Pauli $P \in \mathrm{Q}(\mathbb{C}^{2^n})$ with $P \neq I$ and $P = P^\dagger$, where $\mathrm{Q}(\mathbb{C}^{2^n})$ is the quotient group of the Pauli group (see Definition 3.5). On the one hand, we have

$$\mathcal{M}_\mathcal{E}[\rho] = \sum_{j=1}^{k} p_j (U_j^\dagger P U_j)\rho(U_j^\dagger P U_j)^\dagger. \tag{3.19}$$

On the other hand,

$$\mathcal{M}_\mu[\rho] = (1-\lambda)\rho + \frac{\lambda}{2^{2n}-1} \sum_{Q \in \mathrm{Q}(\mathbb{C}^{2^n}) \setminus \{I\}} Q\rho Q^\dagger \tag{3.20}$$

for some $0 \leq \lambda \leq 1$. Note that for each j, U_j is a Clifford unitary, so $U_j^\dagger P U_j$ is a Pauli. Thus, we have two Kraus representations for the same channel twirl, both with Kraus operators in $\mathrm{Q}(\mathbb{C}^{2^n})$, which form a basis for $\mathrm{L}(\mathbb{C}^N)$. Now we consider the degrees of freedom over these Kraus operators. By [NC00, Theorem 8.2], the j-th term of the RHS of Eq. (3.19) can only contribute to Q in the RHS of Eq. (3.20) if and only if $U_j^\dagger P U_j$ is equivalent to Q in $\mathrm{Q}(\mathbb{C}^{2^n})$. Finally, each $Q \neq I$ appears with equal weight in $\mathcal{M}_\mu[\rho]$. Thus, the distribution $\{p_j, U_j^\dagger P U_j\}$ is uniform over $\mathrm{Q}(\mathbb{C}^{2^n}) \setminus \{I\}$, which completes the proof.

3.3 Pauli mixing and unitary 2-designs

3.3.1 Pauli mixing implies a unitary 2-design

In this subsection, we describe a sufficient condition for an ensemble $\mathcal{E}$ to be a unitary 2-design, which leads to efficient implementations of unitary 2-designs.

Recall the 2×2 Pauli matrices X, Y, and Z defined in Section 2.1. For any $a \in \{0,1\}^n$, define the Pauli operators X^a and Z^a acting on n qubits as $X^a = X^{a_1} \otimes \cdots \otimes X^{a_n}$ and $Z^a = Z^{a_1} \otimes \cdots \otimes Z^{a_n}$. We define some notions related to the Pauli group in the following definition.

Definition 3.5. *The Pauli group, denoted by* $\mathrm{P}(\mathbb{C}^{2^n})$*, is the set of all operators of the form* $i^k X^a Z^b$*, where* $k \in \{0,1,2,3\}$ *and* $a, b \in \{0,1\}^n$*. Let* $\mathrm{Q}(\mathbb{C}^{2^n}) = \mathrm{P}(\mathbb{C}^{2^n})/\{\pm 1, \pm i\}$ *denote the quotient group obtained by disregarding global phases in* $\mathrm{P}(\mathbb{C}^{2^n})$ *in the sense that each element of* $\mathrm{Q}(\mathbb{C}^{2^n})$ *can be represented as* $P_{a,b} = X^a Z^b$*. We call* $P_{0,0} = I_{\mathbb{C}^{2^n}}$ *the trivial Pauli.*

In the following definition, we describe a subgroup of the unitary group, called the Clifford group.

Definition 3.6. *The Clifford group, denoted by* $\mathrm{CL}(\mathbb{C}^{2^n})$*, is the set of all unitary operators that permutes the elements of* $\mathrm{P}(\mathbb{C}^{2^n})$ *(and thus* $\mathrm{Q}(\mathbb{C}^{2^n})$*) under conjugation.*

Recall the definitions of the 2×2 Hadamard matrix H, the phase gate S, and the controlled-not gate CNOT in Section 2.1. It is proven in [Got99] that these three operators form a generating set of the Clifford group $\mathrm{CL}(\mathbb{C}^{2^n})$. The operation of conjugating the elements in $\mathrm{P}(\mathbb{C}^{2^n})$ by some $U \in \mathrm{CL}(\mathbb{C}^{2^n})$ yields a permutation on $\mathrm{P}(\mathbb{C}^{2^n})$, and hence gives a permutation π_U on $\mathrm{Q}(\mathbb{C}^{2^n})$. In the following definition, we characterize the ensembles of unitaries in the Clifford group that uniformly mix Paulis.

Definition 3.7. *Let* $\mathcal{E} = \{p_j, U_j\}_{j=1}^k$ *be an ensemble of unitaries* $U_1, U_2, \ldots, U_k \in \mathrm{CL}(\mathbb{C}^{2^n})$*. We say that* $\mathcal{E}$ *is* Pauli mixing*, if for all* $P \in \mathrm{Q}(\mathbb{C}^{2^n})$ *such that* $P \neq I$*, the distribution* $\{p_j, \pi_{U_j}(P)\}$ *is uniform over* $\mathrm{Q}(\mathbb{C}^{2^n}) \setminus \{I\}$*.*

For any ensemble $\mathcal{E} = \{p_j, U_j\}_{j=1}^k$, define a new ensemble $\mathcal{E}_{\mathrm{Q}}$ as

$$\mathcal{E}_{\mathrm{Q}} = \{2^{-2n} p_j, U_j R_\ell\}_{j=1,\ell=0}^{k,2^{2n}-1}, \tag{3.21}$$

where R_ℓ ranges over all elements in $\mathrm{Q}(\mathbb{C}^{2^n})$. Intuitively, $\mathcal{E}_{\mathrm{Q}}$ is the ensemble where each element in $\mathcal{E}$ is proceeded by a random Pauli drawn from $\mathrm{Q}(\mathbb{C}^{2^n})$. The ensemble $\mathcal{E}$ of Clifford unitaries being Pauli mixing is a sufficient condition for the ensemble $\mathcal{E}_{\mathrm{Q}}$ to be a unitary 2-design, as concluded by the following lemma.

Lemma 3.3. *Let* $\mathcal{E}$ *be an ensemble of Clifford unitaries and* $\mathcal{E}_{\mathrm{Q}}$ *be defined as above. If* $\mathcal{E}$ *is Pauli mixing, then* $\mathcal{E}_{\mathrm{Q}}$ *implements the full bilateral twirl.*

Lemma 3.3 was first shown by DiVincenzo *et al.* [DLT02]. A short proof based on representation theory is shown by Gross *et al.* [GAE07]. In the following, we provide an elementary proof which uses some ideas from [DLT02] but has fewer assumptions. This new proof does not rely on evaluating the full bilateral twirl (using representation theory or the double commutant theorem).

Proof of Lemma 3.3. Let $N = 2^n$. The goal is to show that $\mathcal{T}_{\mathcal{E}_{\mathrm{Q}}}[\rho] = \mathcal{T}_{\mu}[\rho]$ for all $\rho \in \mathrm{D}(\mathbb{C}^N \otimes \mathbb{C}^N)$. Note that $\mathcal{T}_{\mathcal{E}_{\mathrm{Q}}}, \mathcal{T}_{\mu} \in \mathrm{T}(\mathbb{C}^N \otimes \mathbb{C}^N)$ are linear maps. It suffices to show that $\mathcal{T}_{\mathcal{E}_{\mathrm{Q}}}$ and $\mathcal{T}_{\mu}$ are identical on a basis for $\mathrm{L}(\mathbb{C}^N \otimes \mathbb{C}^N)$. We consider a basis that contains the identity matrix $I_{\mathbb{C}^N \otimes \mathbb{C}^N}$ and the swap operator $\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}$, and we complete this basis with matrices that are trace-orthonormal to $I_{\mathbb{C}^N \otimes \mathbb{C}^N}$ and $\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}$, i.e., matrices M such that $\mathrm{Tr}(I_{\mathbb{C}^N \otimes \mathbb{C}^N} M) = \mathrm{Tr}(\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N} M) = 0$. It suffices to prove the following claims:

1. $\mathcal{T}_{\mathcal{E}_{\mathrm{Q}}}[I_{\mathbb{C}^N \otimes \mathbb{C}^N}] = \mathcal{T}_{\mu}[I_{\mathbb{C}^N \otimes \mathbb{C}^N}] = I_{\mathbb{C}^N \otimes \mathbb{C}^N}$.
2. $\mathcal{T}_{\mathcal{E}_{\mathrm{Q}}}[\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}] = \mathcal{T}_{\mu}[\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}] = \mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}$.
3. If $\mathrm{Tr}(I_{\mathbb{C}^N \otimes \mathbb{C}^N} M) = \mathrm{Tr}(\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N} M) = 0$, then $\mathcal{T}_{\mathcal{E}_{\mathrm{Q}}}[M] = \mathcal{T}_{\mu}[M] = \mathbf{0}$ for all M in this basis.

By Eqns. (3.4) and (3.5), as well as the definition of $\mathcal{E}_{\mathrm{Q}}$, we have

$$\mathcal{T}_{\mathcal{E}_{\mathrm{Q}}}[\rho] = \sum_{j=1,\ell=0}^{k,N^2-1} p_j N^{-2} (U_j R_\ell \otimes U_j R_\ell) \rho (R_\ell^\dagger U_j^\dagger \otimes R_\ell^\dagger U_j^\dagger), \text{ and} \tag{3.22}$$

$$\mathcal{T}_{\mu}[\rho] = \int_{\mathrm{U}(\mathbb{C}^N)} d\mu(U) U \otimes U \rho U^\dagger \otimes U^\dagger. \tag{3.23}$$

The first claim follows trivially. Since $\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}(A \otimes B)\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N} = B \otimes A$ for all $A, B \in \mathrm{L}(\mathbb{C}^N)$, we have that $\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}(A \otimes B) = (B \otimes A)\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N}$, and the second claim follows.

Consider the third claim. We fist show that $\mathcal{T}_{\mathcal{E}_{\mathrm{Q}}}[M] = \mathbf{0}$ for all M satisfying the trace-orthonormal condition: $\mathrm{Tr}(I_{\mathbb{C}^N \otimes \mathbb{C}^N} M) = \mathrm{Tr}(\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N} M) = 0$. Observe that the swap operator acting $\mathrm{L}(\mathbb{C}^2 \otimes \mathbb{C}^2)$ can be written as $\mathrm{SWAP}_{\mathbb{C}^2 \otimes \mathbb{C}^2} = \frac{1}{2}(I \otimes I + X \otimes X + Y \otimes Y + Z \otimes Z)$, and thus

$$\mathrm{SWAP}_{\mathbb{C}^N \otimes \mathbb{C}^N} = \frac{1}{N} \sum_{R_\ell \in \mathrm{Q}(\mathbb{C}^N)} R_\ell \otimes R_\ell. \tag{3.24}$$

As $\mathrm{Q}(\mathbb{C}^N)$ is a basis for $\mathrm{L}(\mathbb{C}^N)$, we write M as

$$M = \sum_{a,b=0}^{N^2-1} \alpha_{a,b} R_a \otimes R_b \tag{3.25}$$

where $\alpha_{a,b} \in \mathbb{C}$ for $a, b \in \{0, \ldots, N^2-1\}$. Without loss of generality, we take $R_0 = I_{\mathbb{C}^N} \in \mathrm{Q}(\mathbb{C}^{2^n})$. Then, the two conditions on M can be rephrased as $\alpha_{0,0} = 0$ and $\sum_{a=0}^{N^2-1} \alpha_{a,a} = 0$. By linearity, we focus on analyzing $\mathcal{T}_{\mathcal{E}_\mathrm{Q}}[R_a \otimes R_b]$ for any $(a,b) \neq (0,0)$. We have

$$\mathcal{T}_{\mathcal{E}_\mathrm{Q}}[R_a \otimes R_b] = \sum_{j=1}^{k} p_j (U_j \otimes U_j) \left(\sum_{\ell=0}^{N^2-1} N^{-2} (R_\ell \otimes R_\ell)(R_a \otimes R_b)(R_\ell^\dagger \otimes R_\ell^\dagger) \right) (U_j^\dagger \otimes U_j^\dagger). \tag{3.26}$$

When $a \neq b$, there exists an index $c \in \{0, \ldots, N^2-1\}$ such that R_c commutes with R_a (i.e., $R_c R_a R_c^\dagger = R_a$) and anticommutes with R_b (i.e., $R_c R_b R_c^\dagger = -R_b$). So, we have

$$\begin{aligned}
&2 \sum_{\ell=0}^{N^2-1} (R_\ell \otimes R_\ell)(R_a \otimes R_b)(R_\ell^\dagger \otimes R_\ell^\dagger) \\
&\quad = \sum_{\ell=0}^{N^2-1} (R_\ell \otimes R_\ell)(R_a \otimes R_b)(R_\ell^\dagger \otimes R_\ell^\dagger) + \sum_{\ell=0}^{N^2-1} (R_\ell R_c \otimes R_\ell R_c)(R_a \otimes R_b)(R_c^\dagger R_\ell^\dagger \otimes R_c^\dagger R_\ell^\dagger) && (3.27) \\
&\quad = \mathbf{0}, && (3.28)
\end{aligned}$$

and it follows that $\mathcal{T}_{\mathcal{E}_\mathrm{Q}}[R_a \otimes R_b] = \mathbf{0}$. When $a = b$, we have that

$$\sum_{\ell=0}^{N^2-1} N^{-2} (R_\ell \otimes R_\ell)(R_a \otimes R_a)(R_\ell^\dagger \otimes R_\ell^\dagger) = R_a \otimes R_a, \tag{3.29}$$

as for all $\ell \in \{0, \ldots, N^2-1\}$, R_a either commutes or anticommutes with R_ℓ. Substituting the above into Eq. (3.26), we have

$$\begin{aligned}
\mathcal{T}_{\mathcal{E}_\mathrm{Q}}[R_a \otimes R_a] &= \sum_{j=1}^{k} p_j (U_j \otimes U_j)(R_a \otimes R_a)(U_j^\dagger \otimes U_j^\dagger) && (3.30) \\
&= \frac{1}{N^2-1} \sum_{R_\ell \in \mathrm{Q}(\mathbb{C}^{2^n}) \setminus \{I\}} R_\ell \otimes R_\ell, && (3.31)
\end{aligned}$$

where the second equality follows from the fact the $\mathcal{E}$ is Pauli mixing. Note that the above equation yields a matrix that is independent of a. Therefore, we have

$$\mathcal{T}_{\mathcal{E}_{\mathrm{Q}}}[M] = \sum_{a,b=0}^{N^2-1} \alpha_{a,b} \mathcal{T}_{\mathcal{E}_{\mathrm{Q}}}[R_a \otimes R_b] \tag{3.32}$$

$$= \sum_{a=1}^{N^2-1} \alpha_{a,a} \mathcal{T}_{\mathcal{E}_{\mathrm{Q}}}[R_a \otimes R_a] \tag{3.33}$$

$$= \left(\sum_{a=1}^{N^2-1} \alpha_{a,a} \right) \mathcal{T}_{\mathcal{E}_{\mathrm{Q}}}[R_a \otimes R_a] \tag{3.34}$$

$$= \mathbf{0}, \tag{3.35}$$

where the third equality follows from the fact that $\mathcal{T}_{\mathcal{E}_{\mathrm{Q}}}[R_a \otimes R_a]$ is a matrix independent of a, as shown in Eq. (3.31).

Now, we show that $\mathcal{T}_{\mu}[M] = \mathbf{0}$. Note that, for all $V \in \mathrm{U}(\mathbb{C}^N)$, and for all $M \in \mathrm{L}(\mathbb{C}^N \otimes \mathbb{C}^N)$, it holds that $\mathcal{T}_{\mu}[M] = \mathcal{T}_{\mu}[(V \otimes V) M (V^\dagger \otimes V^\dagger)]$. Apply this equality to each unitary in $\mathcal{E}_{\mathrm{Q}}$ and by linearity, we have that $\mathcal{T}_{\mu}[M] = \mathcal{T}_{\mu}[\mathcal{T}_{\mathcal{E}_{\mathrm{Q}}}[M]] = \mathcal{T}_{\mu}[\mathbf{0}] = \mathbf{0}$, which completes the proof. □

3.3.2 Pauli mixing and $\mathrm{SL}_2(\mathrm{GF}(2^n))$

To facilitate the analysis of the Clifford group and its action on the elements $X^a Z^b = (X^{a_1} \otimes \cdots \otimes X^{a_n})(Z^{b_1} \otimes \cdots \otimes Z^{b_n})$ of the Pauli group, we associate a and b with the elements of the finite field $\mathrm{GF}(2^n)$. In addition, we need to represent a and b in different bases, namely, a primal basis and its dual basis. The preliminaries of finite fields and the bases are provided in Section 2.2.

In light of Lemma 3.3, to construct a unitary 2-design, it suffices to construct an ensemble of Clifford unitaries that is Pauli mixing. Recall that with some primal basis, $\lceil a \rceil$ denotes the representation of a, and $\lfloor b \rfloor$ denotes the representation of b with respect to the corresponding dual basis. With respect to a (primal) basis, we associate each pair $a, b \in \mathrm{GF}(2^n)$ with the Pauli group element $X^{\lceil a \rceil} Z^{\lfloor b \rfloor} = (X^{\lceil a \rceil_1} \otimes \cdots \otimes X^{\lceil a \rceil_n})(Z^{\lfloor b \rfloor_1} \otimes \cdots \otimes Z^{\lfloor b \rfloor_n})$. As proven by Chau [Cha05], there is a subgroup of the Clifford group of size $2^{O(n)}$ such that sampling uniformly over this subgroup performs Pauli mixing. Now we give an overview of this approach. This subgroup is isomorphic to the special linear group of 2×2 matrices

over $\mathrm{GF}(2^n)$:

$$\mathrm{SL}_2(\mathrm{GF}(2^n)) = \left\{ \begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix} : \alpha, \beta, \gamma, \delta \in \mathrm{GF}(2^n) \text{ such that } \alpha\delta - \beta\gamma = 1 \right\}. \tag{3.36}$$

Note that $\mathrm{SL}_2(\mathrm{GF}(2^n))$ has $2^{3n} - 2^n$ elements, and it consists of actions on the Pauli group by conjugation by certain Clifford unitaries. More precisely, we have the following definition.

Definition 3.8. *With respect to a primal basis of* $\mathrm{GF}(2^n)$*, we say that a Clifford unitary* $U \in \mathrm{CL}(\mathbb{C}^{2^n})$ induces $M \in \mathrm{SL}_2(\mathrm{GF}(2^n))$ *if, for all* $a, b \in \mathrm{GF}(2^n)$ *and*

$$\begin{pmatrix} a' \\ b' \end{pmatrix} = M \begin{pmatrix} a \\ b \end{pmatrix}, \tag{3.37}$$

$$U X^{\lceil a \rceil} Z^{\lfloor b \rfloor} U^\dagger \equiv X^{\lceil a' \rceil} Z^{\lfloor b' \rfloor}, \tag{3.38}$$

where $\equiv$ *means equal up to a global phase in* $\{\pm 1, \pm i\}$ *that is a function of* M*,* a*, and* b*.*

The intuition of the above definition is the following. Let us suppose $M = \left(\begin{smallmatrix} \alpha & \beta \\ \gamma & \delta \end{smallmatrix}\right) \in \mathrm{SL}_2(\mathrm{GF}(2^n))$. Then, for all $a, b \in \mathrm{GF}(2^n)$, conjugating $X^{\lceil a \rceil} Z^{\lfloor b \rfloor}$ by the Clifford unitary U results in $X^{\lceil \alpha a + \beta b \rceil} Z^{\lfloor \gamma a + \delta b \rfloor}$ up to a phase.

Through this chapter, we write matrices in $\mathrm{SL}_2(\mathrm{GF}(2^n))$ and vector of length 2 over $\mathrm{GF}(2^n)$ using parenthesis (see above) to distinguish the binary matrices and vectors described in Section 2.2 which use square brackets.

Note that in [Cha05], the mapping specified in Eq. (3.38) is expressed using different notation for Paulis: $X_a|c\rangle = |a + c\rangle$ and $Z_b|c\rangle = (-1)^{T(bc)}|c\rangle$. It is easy to express these in terms of our notation as $X_a = X^{\lceil a \rceil}$ and $Z_b = Z^{\lfloor b \rfloor}$, because $T(bc) = \lfloor b \rfloor \cdot \lceil c \rceil$. The occurrence of the dual basis in Eq. (3.38) allows us to associate a unitary that induces M: without the representation in the dual basis, for general $M \in \mathrm{SL}_2(\mathrm{GF}(2^n))$ there *does not exist* a unitary U that induces M in the sense that $U X^{\lceil a \rceil} Z^{\lceil b \rceil} U^\dagger \equiv X^{\lceil a' \rceil} Z^{\lceil b' \rceil}$. To associate such a unitary, the following lemma holds in terms of definition 3.8.

Lemma 3.4 ([Cha05])**.** *With respect to any primal basis for* $\mathrm{GF}(2^n)$*, there exists an* n*-qubit Clifford unitary* $U \in \mathrm{CL}(\mathbb{C}^{2^n})$ *that induces* M*, for all* $M \in \mathrm{SL}_2(\mathrm{GF}(2^n))$*.*

Consider $M \in \mathrm{SL}_2(\mathrm{GF}(2^n))$. Let U_M denote a unitary that induces M with respect to the primal basis. Similarly, let $\widehat{U}_M$ denote a unitary that induces M with respect to the dual basis. This U_M is unique up to multiplication by a Pauli, which is proven in [Got97, Got98]. We also prove this uniqueness in the following lemma.

Lemma 3.5. *Suppose unitaries U and V have the property that they induce the same permutation on the Pauli group in the sense that, for all $a, b \in \{0,1\}^n$,*

$$UX^aZ^bU^\dagger \equiv VX^aZ^bV^\dagger, \tag{3.39}$$

where $\equiv$ means equal up to a global phase that can be a function of a and b. Then $V = UX^cZ^d$ for some $c, d \in \{0,1\}^n$ (up to some global phase). (Here a and b are binary strings, instead of elements of $\mathrm{GF}(2^n)$, so we do not require the notations $\lceil a \rceil$ and $\lfloor b \rfloor$.)

Proof. First note that Eq. (3.39) is equivalent to

$$X^aZ^b(U^\dagger V)(X^aZ^b)^\dagger = \lambda_{a,b}U^\dagger V \tag{3.40}$$

for all $a, b \in \{0,1\}^n$ where $\lambda_{a,b}$ is some global phase in Eq. (3.39). The unitary $U^\dagger V$ can be expressed as

$$U^\dagger V = \sum_{c,d\in\{0,1\}^n} \alpha_{c,d}X^cZ^d. \tag{3.41}$$

The *symplectic inner product* of (a, b) and (c, d) is defined as $(a, b) \cdot (c, d) = (\oplus_{k=0}^{n-1} a_k d_k) \oplus (\oplus_{k=0}^{n-1} b_k c_k)$. It is easy to verify that X^aZ^b and X^cZ^d either commute (when $(a, b)\cdot(c, d) = 0$) or anticommute (when $(a, b) \cdot (c, d) = 1$). Then, by Eqns. (3.40) and (3.41), we have

$$\sum_{c,d\in\{0,1\}^n} (-1)^{(a,b)\cdot(c,d)}\alpha_{c,d}X^cZ^d = \sum_{c,d\in\{0,1\}^n} \lambda_{a,b}\alpha_{c,d}X^cZ^d. \tag{3.42}$$

Since the Paulis X^cZ^d for $c, d \in \{0,1\}^n$ are linearly independent, the coefficients must match.

Now, we show that at most one $\alpha_{c,d}$ can be nonzero. Suppose there exist distinct (c_1, d_1) and (c_2, d_2) such that $\alpha_{c_1,d_1} \neq 0 \neq \alpha_{c_2,d_2}$. Then, there exists (a, b) such that $(a, b) \cdot (c_1, d_1) \neq (a, b) \cdot (c_2, d_2)$. From Eq. (3.42), it follows that

$$(-1)^{(a,b)\cdot(c_1,d_1)} = \lambda_{a,b} = (-1)^{(a,b)\cdot(c_2,d_2)}, \tag{3.43}$$

which is a contradiction. Therefore, there is a unique $\alpha_{c,d}$ that can be nonzero. By Eq. (3.41), we conclude that

$$V = \alpha_{c,d}UX^cZ^d. \tag{3.44}$$

□

In [Cha05], a possible choice of U_M for any $M \in \mathrm{SL}_2(\mathrm{GF}(2^n))$ is exhibited; however, other than the fact that such U_M is in the Clifford group, it is unclear how to implement such U_M as a small quantum circuit, so its gate complexity is $O(n^2/\log n)$ by [AG04]. In the remainder of this chapter, we aim for an alternative proof of Lemma 3.4 for certain bases of $\mathrm{GF}(2^n)$, as well as a modified version of this lemma, which enables us to obtain constructions of unitary 2-designs with gate complexity $\widetilde{O}(n)$. The connection between Lemma 3.4 and unitary 2-designs is based on the fact that the uniform ensemble over $\{U_M : M \in \mathrm{SL}_2(\mathrm{GF}(2^n))\}$ is Pauli mixing, which is shown in the following lemma.

Lemma 3.6 ([Cha05])**.** *Let $M \in \mathrm{SL}_2(\mathrm{GF}(2^n))$ be chosen uniformly at random. Then, for any $\binom{a}{b} \in (\mathrm{GF}(2^n) \times \mathrm{GF}(2^n)) \setminus \binom{0}{0}$,*

$$\binom{c}{d} = M \binom{a}{b} \tag{3.45}$$

is a uniform distribution over $(\mathrm{GF}(2^n) \times \mathrm{GF}(2^n)) \setminus \binom{0}{0}$.

Proof. We first show that $\mathrm{SL}_2(\mathrm{GF}(2^n))$ acts transitively on $(\mathrm{GF}(2^n) \times \mathrm{GF}(2^n)) \setminus \binom{0}{0}$. Let $\binom{c}{d} \in (\mathrm{GF}(2^n) \times \mathrm{GF}(2^n)) \setminus \binom{0}{0}$. If $c \neq 0$, we have that $\left(\begin{smallmatrix} c & 0 \\ d & c^{-1} \end{smallmatrix}\right)\binom{1}{0} = \binom{c}{d}$. If $c = 0$, then $d \neq 0$, and we have that $\left(\begin{smallmatrix} 0 & d^{-1} \\ d & 0 \end{smallmatrix}\right)\binom{1}{0} = \binom{c}{d}$. Thus, we can map any $\binom{c_1}{d_1} \in (\mathrm{GF}(2^n) \times \mathrm{GF}(2^n)) \setminus \binom{0}{0}$ to $\binom{1}{0}$ and then to any $\binom{c_2}{d_2} \in (\mathrm{GF}(2^n) \times \mathrm{GF}(2^n)) \setminus \binom{0}{0}$ using elements of $\mathrm{SL}_2(\mathrm{GF}(2^n))$.

Now, we prove the uniform distribution. Suppose there are distinct $\binom{c_1}{d_1}$ and $\binom{c_2}{d_2}$ such that $\Pr_M\{M\binom{a}{b} = \binom{c_1}{d_1}\} > \Pr_M\{M\binom{a}{b} = \binom{c_2}{d_2}\}$. Since there exists an $M' \in \mathrm{SL}_2(\mathrm{GF}(2^n))$ such that $M'\binom{c_1}{d_1} = \binom{c_2}{d_2}$, it follows that $\Pr_M\{M'M\binom{a}{b} = \binom{c_2}{d_2}\} \geq \Pr_M\{M\binom{a}{b} = \binom{c_1}{d_1}\}$. Note that the distribution over M is the same as the distribution over $M'M$, so the last equality implies that $\Pr_M\{M\binom{a}{b} = \binom{c_1}{d_1}\} \leq \Pr_M\{M\binom{a}{b} = \binom{c_2}{d_2}\}$, which is a contradiction. □

The goal is to implement U_M for $M \in \mathrm{SL}_2(\mathrm{GF}(2^n))$ with quantum circuit of $\widetilde{O}(n)$ gates. We use two approaches to address the complication caused by interplaying between the primal basis and the dual basis. The approach in Subsection 3.4.1 is based on a self-dual basis for $\mathrm{GF}(2^n)$ and the structure of $\mathrm{SL}_2(\mathrm{GF}(2^n))$. The approaches in Subsection 3.4.2 are based on a polynomial basis and its dual for $\mathrm{GF}(2^n)$ and the structure of two subgroups of $\mathrm{SL}_2(\mathrm{GF}(2^n))$: the *lower-triangular subgroup* and the *upper-triangular subgroup*, which are

defined respectively as

$$\Delta_2(\mathrm{GF}(2^n)) = \left\{ \begin{pmatrix} \alpha & 0 \\ \beta & \alpha^{-1} \end{pmatrix} : \alpha, \beta \in \mathrm{GF}(2^n) \text{ and } \alpha \neq 0 \right\}, \tag{3.46}$$

$$\nabla_2(\mathrm{GF}(2^n)) = \left\{ \begin{pmatrix} \alpha & \beta \\ 0 & \alpha^{-1} \end{pmatrix} : \alpha, \beta \in \mathrm{GF}(2^n) \text{ and } \alpha \neq 0 \right\}. \tag{3.47}$$

3.3.3 A framework for implementing elements of $\mathrm{SL}_2(\mathrm{GF}(2^n))$

In this subsection, we first show a generating set of $\mathrm{SL}_2(\mathrm{GF}(2^n))$ such that all elements of $\mathrm{SL}_2(\mathrm{GF}(2^n))$ can be written as a product of a small constant number of matrices in this generating set. We also give a more restrictive generating set for $\Delta_2(\mathrm{GF}(2^n))$ and $\nabla_2(\mathrm{GF}(2^n))$. Then we present Clifford unitaries that induce these generating matrices. In Section 3.4, we show how to implement these Clifford unitaries with $\widetilde{O}(n)$ gates, which implements elements of $\mathrm{SL}_2(\mathrm{GF}(2^n))$. To begin with, we have the following lemma.

Lemma 3.7. *Every $M \in \mathrm{SL}_2(\mathrm{GF}(2^n))$ can be expressed as a product of a constant number of the following elements of $\mathrm{SL}_2(\mathrm{GF}(2^n))$:*

$$\begin{pmatrix} r & 0 \\ 0 & r^{-1} \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \tag{3.48}$$

for some $r \in \mathrm{GF}(2^n)$ that is nonzero.

Proof. For any $M = \left(\begin{smallmatrix} \alpha & \gamma \\ \beta & \delta \end{smallmatrix}\right) \in \mathrm{SL}_2(\mathrm{GF}(2^n))$, we can decompose it as follows:

$$\begin{pmatrix} \alpha & \gamma \\ \beta & \delta \end{pmatrix} = \begin{cases} \begin{pmatrix} 1 & 0 \\ \beta/\alpha & 1 \end{pmatrix} \begin{pmatrix} 1 & \alpha\gamma \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \alpha & 0 \\ 0 & \alpha^{-1} \end{pmatrix} & \text{if } \alpha \neq 0 \\ \begin{pmatrix} 1 & 0 \\ \delta/\gamma & 1 \end{pmatrix} \begin{pmatrix} \gamma & 0 \\ 0 & \gamma^{-1} \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} & \text{if } \alpha = 0. \end{cases} \tag{3.49}$$

Furthermore, for any nonzero $s \in \mathrm{GF}(2^n)$, there exists $t = s^{2^{n-1}} \in \mathrm{GF}(2^n)$ such that $t^2 = s$. This yields the following further decompositions:

$$\begin{pmatrix} 1 & s \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ s & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \text{ and} \tag{3.50}$$

$$\begin{pmatrix} 1 & 0 \\ s & 1 \end{pmatrix} = \begin{pmatrix} t^{-1} & 0 \\ 0 & t \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} t & 0 \\ 0 & t^{-1} \end{pmatrix}. \tag{3.51}$$

This completes the proof. □

We also specialize the above lemma to the lower-triangular and upper-triangular subgroups of $\mathrm{SL}_2(\mathrm{GF}(2^n))$ as the following lemma.

Lemma 3.8. *Every $M \in \Delta_2(\mathrm{GF}(2^n))$ can be expressed as a product of a constant number of the following elements of $\Delta_2(\mathrm{GF}(2^n))$:*

$$\begin{pmatrix} r & 0 \\ 0 & r^{-1} \end{pmatrix} \text{ and } \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \tag{3.52}$$

and every $M \in \nabla_2(\mathrm{GF}(2^n))$ can be expressed as a product of a constant number of the following elements of $\nabla_2(\mathrm{GF}(2^n))$:

$$\begin{pmatrix} r & 0 \\ 0 & r^{-1} \end{pmatrix} \text{ and } \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \tag{3.53}$$

for some $r \in \mathrm{GF}(2^n)$ that is nonzero.

Because of Lemma 3.7, for every $M \in \mathrm{SL}_2(\mathrm{GF}(2^n))$, we can find a unitary that induces M if we find a unitary that induces each of $\left(\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right)$, $\left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$, and $\left(\begin{smallmatrix} r & 0 \\ 0 & r^{-1} \end{smallmatrix}\right)$ for any nonzero $r \in \mathrm{GF}(2^n)$. Similar statements hold for the lower-triangular subgroup and upper-triangular subgroup of $\mathrm{SL}_2(\mathrm{GF}(2^n))$ with respect to their generating sets shown in Lemma 3.8.

Let us first consider the element of $\mathrm{SL}_2(\mathrm{GF}(2^n))$ of the form $\left(\begin{smallmatrix} r & 0 \\ 0 & r^{-1} \end{smallmatrix}\right)$ for any nonzero $r \in \mathrm{GF}(2^n)$. Such elements can be induces by a Clifford unitary Π_r defined as $\Pi_r|\lceil c \rceil\rangle = |\lceil rc \rceil\rangle$, and Π_r is referred to as the *multiply-by-r* (in the primal basis) operator. In the remainder of this chapter, we use $|c\rangle$ to denote $|\lceil c \rceil\rangle$ to improve readability. For example, in this notation we have $\Pi_r|c\rangle = |rc\rangle$. To verify that Π_r induces $\left(\begin{smallmatrix} r & 0 \\ 0 & r^{-1} \end{smallmatrix}\right)$, observe that for all $c \in \mathrm{GF}(2^n)$,

$$\begin{aligned} \Pi_r X^{\lceil a \rceil} \Pi_r^\dagger |c\rangle &= \Pi_r X^{\lceil a \rceil} |r^{-1}c\rangle && (3.54)\\ &= \Pi_r |r^{-1}c + a\rangle && (3.55)\\ &= |c + ra\rangle && (3.56)\\ &= X^{\lceil ra \rceil} |c\rangle. && (3.57) \end{aligned}$$

Furthermore, we have

$$\begin{aligned} \Pi_r Z^{\lfloor b \rfloor} \Pi_r^\dagger |c\rangle &= \Pi_r Z^{\lfloor b \rfloor} |r^{-1}c\rangle && (3.58)\\ &= \Pi_r (-1)^{\lfloor b \rfloor \cdot \lceil r^{-1}c \rceil} |r^{-1}c\rangle && (3.59)\\ &= (-1)^{T(br^{-1}c)} |c\rangle && (3.60)\\ &= (-1)^{\lfloor br^{-1} \rfloor \lceil c \rceil} |c\rangle && (3.61)\\ &= Z^{\lfloor br^{-1} \rfloor} |c\rangle, && (3.62) \end{aligned}$$

where the third and the fourth equalities follow from Eq. (2.45). Therefore, for all $a, b \in \mathrm{GF}(2^n)$, it holds that $\Pi_r X^{\lceil a \rceil} Z^{\lfloor b \rfloor} \Pi_r^\dagger = X^{\lceil ra \rceil} Z^{\lfloor r^{-1} b \rfloor}$, and hence we can conclude that Π_r induces $\left(\begin{smallmatrix} r & 0 \\ 0 & r^{-1} \end{smallmatrix}\right)$.

Recall that we use parentheses to represent elements of $\mathrm{SL}_2(\mathrm{GF}(2^n))$ and use square brackets to represent their corresponding binary vectors in a primal-dual basis. For any $\left(\begin{smallmatrix} a \\ b \end{smallmatrix}\right) \in \mathrm{GF}_2(2^n) \times \mathrm{GF}(2^n)$, we represent it in a primal-dual basis as $\left[\begin{smallmatrix} \lceil a \rceil \\ \lfloor b \rfloor \end{smallmatrix}\right] \in \{0,1\}^{2n}$, where $\lceil a \rceil, \lfloor b \rfloor \in \{0,1\}^n$. With this notation, the effect of conjugating a Pauli $X^{\lceil a \rceil} Z^{\lfloor b \rfloor}$ by Π_r can be summarized by the following mapping:

$$X^{\lceil a \rceil} Z^{\lfloor b \rfloor} \mapsto X^{M_r \lceil a \rceil} Z^{M_{r^{-1}}^T \lfloor b \rfloor}, \tag{3.63}$$

where M_r is the linear operator corresponding to multiplication by r in the primal basis, as defined in Eq. (2.46), and $M_{r^{-1}}^T$, the transpose of $M_{r^{-1}}$ is the linear operator corresponding to multiplication by r^{-1} in the dual basis.

Now we consider the element $\left(\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right) \in \mathrm{SL}_2(\mathrm{GF}(2^n))$ and find a Clifford unitary that induces it. This Clifford unitary should perform the following mapping:

$$X^{\lceil a \rceil} Z^{\lfloor b \rfloor} \mapsto X^{\lceil a \rceil} Z^{\lfloor a \rfloor + \lfloor b \rfloor} = X^{\lceil a \rceil} Z^{W \lceil a \rceil + \lfloor b \rfloor}, \tag{3.64}$$

where W is the linear operator corresponding to the primal-to-dual basis conversion, as defined in Eq. (2.44). In the following, we show that a unitary Γ_W that induces such mapping is defined as

$$\Gamma_W |c\rangle = i^{\sum_{j=1}^n \sum_{k=1}^n (W)_{j,k} c_j c_k} |c\rangle = i^{\sum_{j=1}^n (W)_{j,j} c_j} (-1)^{\sum_{1 \le j < k \le n} (W)_{j,k} c_j c_k} |c\rangle, \tag{3.65}$$

where the second equality follows from the fact that W is symmetric (see Eq. (2.44)). By Eq. (3.65), one can verify that Γ_W is in the Clifford group, as it can be constructed by the following composition of S gates and controlled-Z gates: an S gate acting on each qubit j whenever $(W)_{j,j} = 1$, and a controlled-Z gate acting on qubits j and k for each $j < k$ whenever $(W)_{j,k} = 1$ (all these gates commute). This simple construction of Γ_W consists of $O(n^2)$ gates. In Subsections 3.4.1 and 3.4.2, we show circuits implementing Γ_W with $\widetilde{O}(n)$ gates.

To verify that Γ_W induces the mapping specified in Eq. (3.64), we first observe that

$$\begin{aligned} X^{\lceil a \rceil} Z^{\lfloor b \rfloor} \Gamma_W^\dagger |c\rangle &= X^{\lceil a \rceil} Z^{\lfloor b \rfloor} (-i)^{\sum_{j=1}^n (W)_{j,j} c_j} (-1)^{\sum_{1 \le j < k \le n} (W)_{j,k} c_j c_k} |c\rangle & (3.66) \\ &= (-i)^{\sum_{j=1}^n (W)_{j,j} c_j} (-1)^{\sum_{1 \le j < k \le n} (W)_{j,k} c_j c_k} X^{\lceil a \rceil} (-1)^{\lfloor b \rfloor \cdot \lceil c \rceil} |c\rangle & (3.67) \\ &= (-i)^{\sum_{j=1}^n (W)_{j,j} c_j} (-1)^{\sum_{1 \le j < k \le n} (W)_{j,k} c_j c_k + \lfloor b \rfloor \cdot \lceil c \rceil} |a + c\rangle. & (3.68) \end{aligned}$$

Furthermore, we have

$$\Gamma_W|a+c\rangle = i^{\sum_{j=1}^n (W)_{j,j}(a+c)_j}(-1)^{\sum_{1\le j<k\le n}(W)_{j,k}(a+c)_j(a+c)_k}|a+c\rangle \tag{3.69}$$
$$= i^{\sum_{j=1}^n (W)_{j,j}(a_j+c_j-2a_jc_j)}(-1)^{\sum_{1\le j<k\le n}(W)_{j,k}(a_j+c_j)(a_k+c_k)}|a+c\rangle \tag{3.70}$$
$$= i^{\sum_{j=1}^n (W)_{j,j}(a_j+c_j)}(-1)^{\sum_{j=1}^n (W)_{j,j}a_jc_j}(-1)^{\sum_{1\le j<k\le n}(W)_{j,k}(a_ja_k+a_jc_k+c_ja_k+c_jc_k)}|a+c\rangle, \tag{3.71}$$

where the second equality follows from the fact that, for any $a, c \in \{0,1\}$, $a \oplus c = a+c-2ac$. Combining Eqns. (3.68) and (3.71), we have

$$\Gamma_W X^{\lceil a\rceil} Z^{\lfloor b\rfloor}\Gamma_W^\dagger|c\rangle = i^{\sum_{j=1}^n\sum_{k=1}^n a_ja_k}(-1)^{\sum_{j=1}^n (W)_{j,j}a_jc_j+\sum_{1\le j<k\le n}(W)_{j,k}(a_jc_k+a_kc_j)+\lfloor b\rfloor\cdot\lceil c\rceil}|a+c\rangle \tag{3.72}$$
$$= i^{\sum_{j=1}^n\sum_{k=1}^n (W)_{j,k}a_ja_k}(-1)^{\sum_{k=1}^n\sum_{j=1}^n (W)_{k,j}a_jc_k+\lfloor b\rfloor\cdot\lceil c\rceil}|a+c\rangle \tag{3.73}$$
$$= i^{\sum_{j=1}^n\sum_{k=1}^n (W)_{j,k}a_ja_k}(-1)^{W\lceil a\rceil\cdot\lceil c\rceil+\lfloor b\rfloor\cdot\lceil c\rceil}|a+c\rangle \tag{3.74}$$
$$= i^{\sum_{j=1}^n\sum_{k=1}^n (W)_{j,k}a_ja_k}X^{\lceil a\rceil}(-1)^{W\lceil a\rceil\cdot\lceil c\rceil+\lfloor b\rfloor\cdot\lceil c\rceil}|c\rangle \tag{3.75}$$
$$= i^{\sum_{j=1}^n\sum_{k=1}^n (W)_{j,k}a_ja_k}X^{\lceil a\rceil}(-1)^{\lfloor a\rfloor\cdot\lceil c\rceil+\lfloor b\rfloor\cdot\lceil c\rceil}|c\rangle \tag{3.76}$$
$$= i^{\sum_{j=1}^n\sum_{k=1}^n (W)_{j,k}a_ja_k}X^{\lceil a\rceil}Z^{\lfloor a\rfloor+\lfloor b\rfloor}|c\rangle. \tag{3.77}$$

This implies that Γ_W implies the mapping specified in Eq. (3.64) (up to some global phase).

We also need to find the Clifford unitary that induces the element $\left(\begin{smallmatrix}0&1\\1&0\end{smallmatrix}\right) \in \mathrm{SL}_2(\mathrm{GF}(2^n))$. This is addressed in Subsections 3.4.1 and 3.4.2 in different ways, which yield different constructions for unitary 2-designs.

3.4 Efficient constructions of unitary 2-designs

Now, the construction of a unitary 2-design reduces to implementing unitaries that induce the generators of $\mathrm{SL}_2(\mathrm{GF}(2^n))$. The goal of this section is to find $\widetilde{O}(n)$-sized circuits to implement these unitaries.

3.4.1 Near-linear implementation based on self-dual basis for $\mathrm{GF}(2^n)$

The first approach is to represent the elements in $\mathrm{GF}(2^n)$ with respect to a self-dual basis. The advantage of using a self-dual basis is that, the change of basis operation defined in

Eq. (2.44) is trivial. In this subsection, we omit the $\lceil\ \rceil$ and $\lfloor\ \rfloor$ notations because the dual of a self-dual basis is itself. For all n-bit binary strings $a, b \in \{0,1\}^n$, it holds that $S^{\otimes n} X^a Z^b (S^\dagger)^{\otimes n} = i^{a_1+\cdots+a_n \bmod 4} X^a Z^{a+b}$ and $H^{\otimes n} X^a Z^b H^{\otimes n} = (-1)^{a\cdot b} X^b Z^a$. Therefore, $S^{\otimes n}$ induces $\left(\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right)$ and $H^{\otimes n}$ induces $\left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$.

It remains to implement the unitary Π_r (multiply-by-r operator) that induces $\left(\begin{smallmatrix} r & 0 \\ 0 & r^{-1} \end{smallmatrix}\right)$. Fast multiplication methods with respect to a polynomial basis are known; however, no polynomial basis of $\mathrm{GF}(2^n)$ is also self-dual if $n \geq 2$ [Haz96]. Here, we use special self-dual bases that can be efficiently converted to and from polynomial bases. These special self-dual bases are constructed with *Gauss periods* and exist for *admissible* n's (see Definition 3.9 below). According to [vzGP01], there are *infinitely many* admissible n's under the extended Riemann Hypothesis. Our implementation in this subsection is restricted to the values of n defined as follows.

Definition 3.9. *A natural number n is called* admissible *if the following two conditions hold:*

1. *$2n+1$ is prime*
2. *$\gcd(e, n) = 1$, where e is the index of the subgroup generated by 2 in $\mathbb{Z}^*_{2n+1}$,*

where $\mathbb{Z}^*_{2n+1}$ denotes the multiplicative group of $\mathbb{Z}_{2n+1}$. Since $\mathbb{Z}^*_{2n+1}$ has $2n$ elements when $2n+1$ is prime, $e = \frac{2n}{|\langle 2 \rangle|}$.

In the remainder of this subsection, we first describe the procedure of finding a self-dual basis using Gauss periods and explain the efficient conversion between the two representations with respect to a self-dual basis and a polynomial basis. Then we describe the implementation of Π_r.

Since, for an admissible n, $2n+1$ is prime, it holds that $2^{2n} \equiv 1 \bmod 2n+1$ by Fermat's Little Theorem. So $2n+1$ divides $2^{2n}-1$, which implies that there is a primitive $(2n+1)$-th root of unity $\beta \in \mathrm{GF}(2^{2n})$. One way to obtain such β is the following. Let ξ be a generator of the multiplicative group of $\mathrm{GF}(2^{2n})$. Because $\xi^{2^{2n}-1} = 1$, we can take $\beta = \xi^{(2^{2n}-1)/(2n+1)}$. Consider the set

$$\mathrm{S} = \{\beta + \beta^{-1}, \beta^2 + \beta^{-2}, \ldots, \beta^n + \beta^{-n}\}. \tag{3.78}$$

We first show that S is a self-dual normal basis of $\mathrm{GF}(2^n)$ over $\mathrm{GF}(2)$ (as defined in Section 2.2). Then we show how to efficiently convert between S and a polynomial basis.

To show that S is a self-dual normal basis, we first argue that for an admissible n, 2 and -1 generate $\mathbb{Z}^*_{2n+1}$ (i.e., $\langle 2, -1 \rangle = \mathbb{Z}^*_{2n+1}$). A proof is given in [GvzGPS00], and it

can be rephrased as follows. Let γ generate the cyclic group $\mathbb{Z}^*_{2n+1}$. If e is the index of $\langle 2\rangle$ in $\mathbb{Z}^*_{2n+1}$, then $2 = \gamma^e$. Furthermore, $\gamma^n = -1$. Since $\gcd(e, n) = 1$, there are integers k_1, k_2 such that $1 = ek_1 + nk_2$ and therefore, $\gamma \in \langle 2, -1\rangle$, so $\mathbb{Z}^*_{2n+1} = \langle 2, -1\rangle$. This further implies that

$$\{2^0, -2^0, 2^1, -2^1, \ldots, 2^{n-1}, -2^{n-1}\} \equiv \{1, -1, 2, -2, \ldots, n, -n\} \mod 2n + 1. \tag{3.79}$$

Note that the RHS of the above equation is equivalent to the whole group of $\mathbb{Z}^*_{2n+1}$. Then, we can reorder the elements of S as

$$\{\beta^{2^0} + \beta^{-2^0}, \beta^{2^1} + \beta^{-2^1}, \ldots, \beta^{2^{n-1}} + \beta^{-2^{n-1}}\}. \tag{3.80}$$

The set in Eq. (3.80), as a subset of $\mathrm{GF}(2^n)$, is equal to $\{\alpha^{2^0}, \alpha^{2^1}, \cdots, \alpha^{2^{n-1}}\}$ where $\alpha = \beta + \beta^{-1}$ is called a *Gauss period* of type $(n, 2)$ over $\mathrm{GF}(2)$. It is easy to see that $\beta + \beta^{-1} \in \mathrm{GF}(2^n)$, for one can verify that $(\beta + \beta^{-1})^{2^n} = \beta + \beta^{-1}$. To argue that the set $\{\alpha^{2^0}, \alpha^{2^1}, \cdots, \alpha^{2^{n-1}}\}$ is linearly independent, suppose there exists $a_0, \ldots, a_{n-1} \in \mathrm{GF}(2)$ that are not all zero such that

$$\begin{aligned} 0 &= a_0\alpha^{2^0} + \cdots + a_{n-1}\alpha^{2^{n-1}} && (3.81)\\ &= a_0(\beta + \beta^{-1})^{2^0} + \cdots + a_{n-1}(\beta + \beta^{-1})^{2^{n-1}} && (3.82)\\ &= a_0(\beta^{2^0} + \beta^{-2^0}) + \cdots + a_{n-1}(\beta^{2^{n-1}} + \beta^{-2^{n-1}}). && (3.83) \end{aligned}$$

Because of Eqns. (3.79) and (3.83), there exist $b_1, \ldots, b_{2n} \in \mathrm{GF}(2)$ that are not all zero such that

$$0 = b_1\beta + \cdots + b_{2n}\beta^{2n} = \beta(b_1\beta^0 + \cdots + b_{2n}\beta^{2n-1}) \mod 2n + 1. \tag{3.84}$$

The above equation holds for all $(2n + 1)$-th root of unity of $\mathrm{GF}(2^{2n})$, and β can be substituted by β^r for $r \in \{1, \ldots, 2n\}$. It follows that the polynomial $f(x) = u_1x^0 + \cdots + u_{2n}x^{2n-1}$ has $2n$ roots, which is a contradiction with that fact that $f(x)$ has degree at most $2n - 1$. As a result, the set $\{\alpha^{2^0}, \alpha^{2^1}, \ldots, \alpha^{2^{n-1}}\}$ (as well as S) forms a normal basis and α is a normal element in $\mathrm{GF}(2^n)$. Then, by [GvzGPS00, Corollary 3.5], it holds that any normal basis of Gauss period of type $(n, 2)$ over $\mathrm{GF}(2)$ is self-dual when $n > 2$. Therefore, S is a self-dual normal basis as claimed.

Next, we show how to efficiently convert between S and a polynomial basis. We define a map from $\mathrm{GF}(2^n)$ to $\{0, 1\}^{n+1}$ as follows. If $a \in \mathrm{GF}(2^n)$, then it is mapped to the vector $a' = (0, a_1, \ldots, a_n)$, where $a = a_1(\beta + \beta^{-1}) + \cdots + a_n(\beta^n + \beta^{-n})$. In other words, a' is the coordinate of a with respect to the spanning set $\{1, \beta + \beta^{-1}, \beta^2 + \beta^{-2}, \ldots, \beta^n + \beta^{-n}\}$.

Including the element 1 makes this spanning set not a basis, but significantly simplifies the conversion between the following two spanning sets:

$$\mathrm{S}' = \{1, \beta + \beta^{-1}, \beta^2 + \beta^{-2}, \ldots, \beta^n + \beta^{-n}\}, \tag{3.85}$$

$$\mathrm{M} = \{1, \beta + \beta^{-1}, (\beta + \beta^{-1})^2, \ldots, (\beta + \beta^{-1})^n\}. \tag{3.86}$$

Note that the set M arises from adding 1 to a polynomial basis. We call S′ a *self-dual spanning set* and M a *polynomial spanning set*. The fact that M is not a basis does not affect how we represent a field element as a polynomial based on M, i.e., $a = \sum_{j=0}^{n} a_j(\beta + \beta^{-1})^j$, and fast multiplication of two polynomials of this form still works.

For $j \in \{1, \ldots, n\}$, let $s_j = \beta^j + \beta^{-j}$, $t_j = (\beta + \beta^{-1})^j$, and let s'_j and t'_j be the $(n+1)$-bit string output by the map defined earlier. In the following, we describe the linear transformation L_{n+1} that maps s'_j to t'_j for all j (by right multiplication). The transformation L_{n+1} is not unique. A simple choice for L_{n+1} is based on the binomial expansion $(\beta + \beta^{-1})^j = \sum_{\ell=0}^{j} \binom{j}{\ell} \beta^{j-2\ell}$. More precisely, for general k, we can choose L_k as

$$(L_k)_{\ell,j} = \begin{cases} 0 & \text{if } \ell > j \text{ or } j - \ell \text{ is odd,} \\ \binom{j}{(j-\ell)/2} \bmod 2 & \text{otherwise,} \end{cases} \tag{3.87}$$

where $0 \leq \ell, j < k$. The operation L_k is upper-triangular with 1's on the diagonal, which implies $\det(L_k) = 1$, so L_k is invertible.

Now, we find a unitary Λ_k that induces L_k. (More precisely, Λ_k induces the matrix with identical diagonal blocks that is the $(k-1) \times (k-1)$ submatrix of L_k with the first row and column omitted.) The unitary Λ_n also induces a conversion from S′ to M. In [vzGSS07], the following theorem is proven.

Theorem 3.9 ([vzGSS07])**.** *Right multiplying L_{n+1} (L_{n+1}^{-1} respectively) by the vector representation (a') of an element $a \in \mathrm{GF}(2^n)$ described above can be done using $O(n \log n)$ operations (additions and multiplications) in* GF(2).

From this theorem, an efficient (classical) circuit that induces L_{n+1} can be built with $O(n \log n)$ CNOT gates. The intuition is that L_{n+1} can be decomposed as a product of $O(\log n)$ matrices, each with $O(n)$ 1's, as shown in [vzGSS07]. Since the linear transformation can be done with GF(2) additions, this circuit contains only CNOT gates. A circuit for L_{n+1}^{-1} can be obtained by running the circuit for L_{n+1} backward.

In the following, we present a different circuit that induces L_{n+1} – a recursive construction that also requires $O(n \log n)$ CNOT gates, which yields a different proof of Theorem 3.9. First consider L_k as defined in Eq. (3.87) where k is a power of 2. Taking $k = 8$

as an example,

$$L_8 = \left[\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right]. \tag{3.88}$$

We use two properties of L_k when k is a power of 2 (see L_8 above for an illustration):

1. Each L_k consists of three nonzero blocks: two identical diagonal blocks which is $L_{k/2}$ and a block above the diagonal which we call $L^{\Gamma}_{k/2}$ (which is almost like $L_{k/2}$ turned upside down).

2. The first row of $L^{\Gamma}_{k/2}$ contains only zeros. The $(\ell{+}2)$-th row of $L^{\Gamma}_{k/2}$ is the $\left(\frac{k}{2}-\ell\right)$-th row of $L_{k/2}$ (where $0 \le \ell \le k/2-2$).

An illustration of the two properties is shown in Figure 3.3. Take the Pascal's triangle (mod 2) with k rows, and rotate the entries 90 degrees counter-clockwise. This gives the (ℓ, j) entries of L_k when $\ell \ge j$ and $\ell - j$ is even. The above two properties for L_k primarily come from the fact that Pascal's triangle (mod 2) with k rows consists of 4 triangles of $k/2$ rows, the middle one only has zero entries, and the other three are identical copies of Pascal's triangle (mod 2) with $k/2$ rows. Also, the triangle is always left-right symmetric. Proofs of these are readily obtained from Lucas' Theorem [1] [Fin47] (a more accessible proof can be found online at [Rid]).

A left-multiplication of a vector by L_k yields

$$\left[\begin{array}{c|c} L_{k/2} & L^{\Gamma}_{k/2} \\ \hline 0 & L_{k/2} \end{array}\right] \left[\begin{array}{c} v_1 \\ v_2 \end{array}\right] = \left[\begin{array}{c} L_{k/2}\, v_1 + L^{\Gamma}_{k/2}\, v_2 \\ L_{k/2}\, v_2 \end{array}\right]. \tag{3.89}$$

Due to the relation between $L^{\Gamma}_{k/2}$ and $L_{k/2}$, the above map can be induced by the unitary Λ_k implemented by the circuit in Figure 3.4. Using standard recursion analysis, the circuit contains $O(k \log k)$ CNOT gates.

[1] Consider the base-p representation of integers m and n, where $m \ge n \ge 0$, and p is prime: $m = m_0 + m_1 p + \cdots + m_k p^k, n = n_0 + n_1 p + \cdots + n_k p^k$. Then, $\binom{m}{n} \equiv \binom{m_0}{n_0}\binom{m_1}{n_1}\cdots\binom{m_k}{n_k} \bmod p$

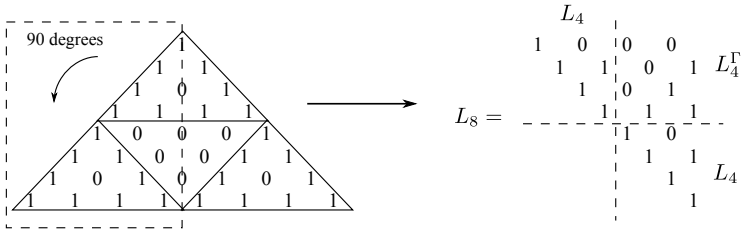

Figure 3.3: An illustration of the Pascal's triangle structure of the matrix L_8 . Taking the left half of an 8-level Pascal's triangle and rotating counter-clockwise by 90 degrees, we obtain L_8. Note that the block L_4^Γ is the horizontal reflection of the lower-diagonal block L_4 with a downward shift, as described by property 2 of L_k.

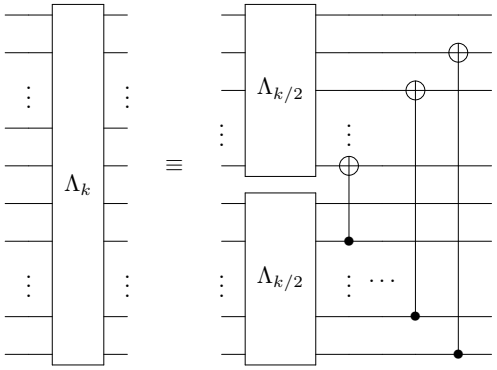

Figure 3.4: An example of representation conversion circuit which demonstrates the recursive structure.

For general values of k, we apply the above construction to obtain $L_{2^{\lceil \log_2 k \rceil}}$. We restrict the circuit for $L_{2^{\lceil \log_2 k \rceil}}$ to a sub-circuit with the first k registers and the CNOT gates between them to obtain a circuit for L_k^Γ that still has size $O(k \log k)$.

The circuit for Λ_k^{-1} that converts a vector from the self-dual representation to the polynomial representation can be obtained by running the circuit for Λ_k backward. The first qubit which corresponds to the additional "1" in S' is always $|0\rangle$ and it remains intact during the computation, and therefore can be safely removed in the circuit. It is kept in the analysis for conceptual simplicity.

Finally, we are ready to give the recipe for the fast multiplication of two elements $a, r \in \mathrm{GF}(2^n)$ represented with respect to S':

1. Insert a zero at the beginning of the vector representations of a and r to get the vectors a' and r' with respect to the spanning set S$'$.

2. Convert a' and r' to new representations $\tilde{a}$ and $\tilde{r}$ with respect to the polynomial spanning set M, using the circuit for Λ_{n+1}^{-1}.

3. Multiply $\tilde{a}$ by $\tilde{r}$ using Schönhage's multiplication algorithm [Sch77] (denoted by $\tilde{\Pi}_r$ in Figure 3.5). The result is a vector with respect to the polynomial spanning set $\{1, \beta+\beta^{-1}, (\beta+\beta^{-1})^2, \ldots, (\beta+\beta^{-1})^{2n}\}$.

4. Apply the unitary Λ_{2n+1} to the vector above so it is represented in the spanning set $\{1, \beta+\beta^{-1}, \beta^2+\beta^{-2}, \ldots, \beta^{2n}+\beta^{-2n}\}$. Then, discard the first element which is always 0. The result is the vector representation with respect to the spanning set $\{\beta+\beta^{-1}, \beta^2+\beta^{-2}, \ldots, \beta^{2n}+\beta^{-2n}\}$. Since β is the $(2n+1)$-th root of unity in GF(2^{2n}) (i.e., $\beta^{2n+1}=1$), we have $\beta+\beta^{-1} = \beta^{2n}+\beta^{-2n}$, $\beta^2+\beta^{-2} = \beta^{2n-1}+\beta^{-2n+1}, \ldots$. Therefore with n additional GF(2) CNOT gates, the resulting vector can be reduced to the one with respect to the permuted self-dual normal basis S.

In step 3, Schönhage's multiplication algorithm [Sch77] uses a radix-3 FFT algorithm to do fast convolution. Readers may refer to [vzGG13] for another description of Schönhage's algorithm. This multiplication algorithm requires $O(n \log n \log\log n)$ operations (additions and multiplications). Additions can be implemented with CNOT gates. Multiplications involved in this radix-3 FFT are the ones between an element of the polynomial ring GF(2)$[x]/\langle x^{2m}+x^m+1\rangle$ (for certain m) and x (which is a $3m$-th root of unity in GF(2)$[x]/\langle x^{2m}+x^m+1\rangle$). The result of this kind of multiplications is a shift of coefficients and it can be implemented by SWAP gates. Therefore, the whole multiplication method can be implemented with $O(n \log n \log\log n)$ CNOT gates. As an example, Figure 3.5 shows the implementation of Π_r in GF(2^5).

We show that the radix-3 FFT algorithm has logarithmic depth: if the current step of this algorithm is working on a polynomial of degree k, in the next recursion step, it will work in parallel on three polynomials of degree $\lceil k/3 \rceil$. The total number of steps (i.e., the depth of the circuit) is, therefore, $O(\log n)$ for a polynomial of degree n. To multiply two polynomials of degree at most n, each recursion step essentially consists of three components: computing the radix-3 FFT, recursively doing $\lceil \sqrt{n} \rceil$ multiplications of polynomials of degree at most $\lceil \sqrt{n} \rceil$ (in parallel), and computing the inverse radix-3 FFT. Using a similar analysis, the depth of the polynomial multiplication circuit is $O(\log(n) + \log(n^{1/2}) + \log(n^{1/4}) + \cdots + 1) = O(\log n)$. The logarithmic depth of the basis conversion circuit can be shown by its recursive structure (e.g., Figure 3.4). Therefore, the depth of the circuit for Π_r is $O(\log n)$.

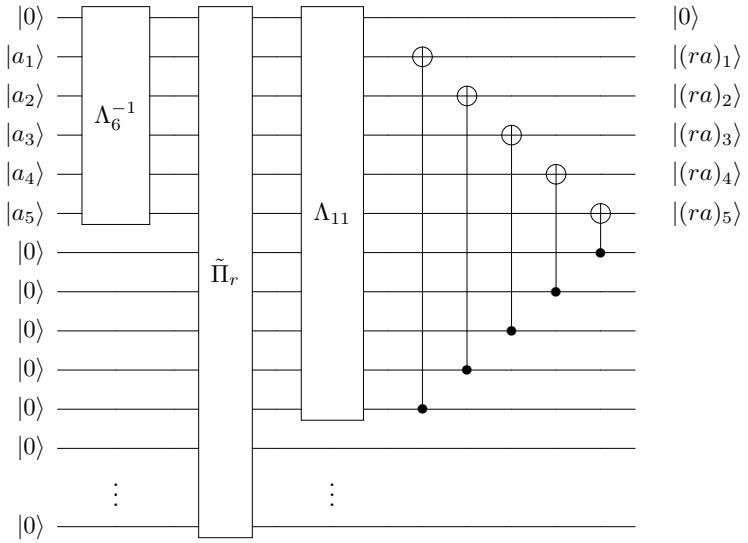

Figure 3.5: The implementation of Π_r for multiplication of a by r where $a, r \in \mathrm{GF}(2^5)$. $\tilde{\Pi}_r$ is an implementation of Schönhage's multiplication algorithm. The input and output bits are with respect to a self-dual basis.

The ancilla qubits can be reset to $|0\rangle$ using standard techniques in reversible computing. The result is a circuit for Π_r for any nonzero $r \in \mathrm{GF}(2^n)$ with $O(n \log n \log\log n)$ CNOT gates.

3.4.2 Near-linear implementations based on polynomial basis for $\mathrm{GF}(2^n)$

In this subsection, we present two alternative constructions for unitary 2-designs based on polynomial bases for $\mathrm{GF}(2^n)$. The benefit of using polynomial bases is that the unitary that induces $\left(\begin{smallmatrix} r & 0 \\ 0 & r^{-1} \end{smallmatrix}\right)$ for $r \neq 0$ (which is a generator of $\mathrm{SL}_2(\mathrm{GF}(2^n))$) is straightforward to implement with $O(n \log n \log\log n)$ Clifford gates and circuit depth $O(\log n)$, as described in Subsection 3.4.1. For the generator $\left(\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right)$, we provide two different $\widetilde{O}(n)$ circuit implementations in this subsection. However, an efficient implementation of the generator $\left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$ with respect to a polynomial basis is not known. Instead of implementing every element of $\mathrm{SL}_2(\mathrm{GF}(2^n))$, we implement its lower-triangular subgroup $\Delta_2(\mathrm{GF}(2^n))$ based on the primal basis, and its upper-triangular subgroup $\nabla(\mathrm{GF}(2^n))$ based on the dual basis. At the end of this subsection, we show how to combine the implementations of the two

subgroups to achieve Pauli mixing, which results in an exact unitary 2-design with the desired complexity. Note that Lemma 3.8 gives the necessary elements (generating set) we need to implement for both the lower- and the upper-triangular subgroups.

In the following, we give two different implementations of the unitary that induces $\left(\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right)$. Together with the implementation of the unitary Π_r that induces $\left(\begin{smallmatrix} r & 0 \\ 0 & r^{-1} \end{smallmatrix}\right)$, we can implement the lower-triangular subgroup $\Delta_2(\mathrm{GF}(2^n))$ because of Lemma 3.8. The upper-triangular subgroup $\nabla_2(\mathrm{GF}(2^n))$ can be implemented with respect to the dual basis by conjugating the unitaries for $\Delta_2(\mathrm{GF}(2^n))$ by $H^{\otimes n}$.

Implementation with $O(n \log n \log\log n)$ non-Clifford gates

Now, we show how to implement the generator $\left(\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right)$ using $O(n \log n \log\log n)$ gates with depth $O(\log n)$. This construction contains non-Clifford gates, but they compose to a Clifford unitary.

We need to implement the unitary Γ_W, which is defined in Eq. (3.65), and W is the primal-to-dual basis conversion matrix defined in Eq. (2.44). Since the primal basis is a polynomial basis, W has the property that if $j + k = j' + k'$ then $(W)_{j,k} = (W)_{j',k'}$ for all j, k, j', k'. Such a matrix is referred to as a *Hankel matrix*. Recall that Γ_W is defined as

$$\Gamma_W |c\rangle = i^{\sum_{j=1}^{n} \sum_{k=1}^{n} (W)_{j,k} c_j c_k} |c\rangle. \tag{3.90}$$

To implement Γ_W, it suffices to compute the exponent of i using mod 4 arithmetic, and the exponent can be written as

$$\begin{bmatrix} c_1 & \cdots & c_n \end{bmatrix} W \begin{bmatrix} c_1 \\ \vdots \\ c_n \end{bmatrix}. \tag{3.91}$$

The computation of Eq. (3.91) is related to the problem of computing convolutions. The *convolution* of two d-dimensional vectors u and v is defined as the $(2d-1)$-dimensional vector w satisfying

$$w_0 + w_1 T + w_2^2 T^2 + \cdots + w_{2d-2} T^{2d-2} \tag{3.92}$$

$$= \left(u_0 + u_1 T + u_2^2 T^2 + \cdots + u_{d-1} T^{d-1}\right) \left(v_0 + v_1 T + v_2^2 T^2 + \cdots + v_{d-1} T^{d-1}\right), \tag{3.93}$$

for polynomials over T. The product of a Hankel matrix with a vector reduces to convolution, as shown in the following proposition.

Proposition 3.10. *The product of an $n \times n$ Hankel matrix with an n-dimensional vector reduces to the problem of computing the convolution of two $(2n-1)$-dimensional vectors.*

Proof. Suppose we compute the production of a Hankel matrix with some vector:

$$\begin{bmatrix} x_1 & x_2 & \cdots & x_n \\ x_2 & x_3 & \cdots & x_{n+1} \\ \vdots & \vdots & \ddots & \vdots \\ x_n & x_{n+1} & \cdots & x_{2n-1} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}. \tag{3.94}$$

This can be computed from the convolution of $[x_1, \ldots, x_{2n-1}]$ and $[0, \ldots, 0, y_n, \ldots, y_1]$: this convolution is a $(4n-2)$-dimensional vector that is the vector in Eq. (3.94) padded with $2n-2$ components on the left and $n-1$ components on the right. □

To compute Eq. (3.91), we first compute $e_1, \ldots, e_n \in \mathbb{Z}_4$ satisfying

$$\begin{bmatrix} e_1 \\ \vdots \\ e_n \end{bmatrix} = W \begin{bmatrix} c_1 \\ \vdots \\ c_n \end{bmatrix}, \tag{3.95}$$

with a fast algorithm for polynomial multiplication over $\mathbb{Z}_4$ using $O(n \log n \log\log n)$ gates (e.g., [vzGG13, Theorem 8.23]). Then the value of Eq. (3.91) for the exponent for i in Eq. (3.65) can be obtained from the $2n$ ancilla qubits containing $e_1, \ldots, e_n$ (as each e_j is a two-bit string) and the n qubits containing $c_1, \ldots, c_n$ as follows. For each $j \in \{1, \ldots, n\}$, apply a controlled-Z gate between the high order bit of e_j and c_j, and apply a controlled-S gate between the low order bit of e_j and c_j.

Note that this construction uses controlled-S gates, which are not in the Clifford group. It is not straightforward to circumvent as the underlying ring is $\mathbb{Z}_4$ and addition mod 4 requires non-Clifford gates. The construction uses polynomial multiplications. Therefore, using the similar circuit depth analysis in Subsection 3.4.1, the circuit depth of this construction is $O(\log n)$.

Implementation with $O(n \log^2 n \log\log n)$ Clifford gates

Now, we show how to implement the generator $\left(\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right)$ using $O(n \log^2 n \log\log n)$ Clifford gates with circuit depth $O(\log^2 n)$. In the previous construction, the computation is reduced to a convolution in mod 4 arithmetic, which can be computed efficiently with non-Clifford

gates. In this construction, we use a recursive procedure that is based on convolutions in mod 2 arithmetic, which can be computed efficiently with Clifford gates.

Without loss of generality, we assume that n is a power of 2. (In general, an arbitrary n is divided unevenly in the recursive step, as $n = \lfloor \frac{n}{2} \rfloor + \lceil \frac{n}{2} \rceil$.) We divide W into four $\frac{n}{2} \times \frac{n}{2}$ blocks:

$$W = \begin{bmatrix} W^{(11)} & W^{(12)} \\ W^{(21)} & W^{(22)} \end{bmatrix}, \tag{3.96}$$

where $W^{(11)}, W^{(12)}, W^{(21)}$, and $W^{(22)}$ are $\frac{n}{2} \times \frac{n}{2}$ Hankel matrices, and $W^{(12)} = W^{(21)}$. Define block matrices A, B, and C as

$$A = \begin{bmatrix} 0 & W^{(12)} \\ W^{(21)} & 0 \end{bmatrix}, B = \begin{bmatrix} W^{(11)} & 0 \\ 0 & 0 \end{bmatrix}, \text{ and } C = \begin{bmatrix} 0 & 0 \\ 0 & W^{(22)} \end{bmatrix}. \tag{3.97}$$

Clearly, $W = A + B + C$. If we treat A, B, and C as basis conversion matrices, we define Γ_A, Γ_B, and Γ_C as in Eq. (3.65) (with W replaced to A, B, and C, respectively). Then Γ_W can be implemented with $\Gamma_A \Gamma_B \Gamma_C$ as

$$\begin{aligned} \Gamma_A \Gamma_B \Gamma_C X^{\lceil a \rceil} Z^{\lfloor b \rfloor} \Gamma_C^\dagger \Gamma_B^\dagger \Gamma_A^\dagger &= \Gamma_A \Gamma_B X^{\lceil a \rceil} Z^{C\lceil a \rceil + \lfloor b \rfloor} \Gamma_B^\dagger \Gamma_A^\dagger && (3.98) \\ &= \Gamma_A X^{\lceil a \rceil} Z^{(B+C)\lceil a \rceil + \lfloor b \rfloor} \Gamma_A^\dagger && (3.99) \\ &= X^{\lceil a \rceil} Z^{(A+B+C)\lceil a \rceil + \lfloor b \rfloor} && (3.100) \\ &= X^{\lceil a \rceil} Z^{W\lceil a \rceil + \lfloor b \rfloor} && (3.101) \\ &= \Gamma_W X^{\lceil a \rceil} Z^{\lfloor b \rfloor} \Gamma_W^\dagger. && (3.102) \end{aligned}$$

We first show how to implement Γ_A using $O(n \log n \log\log n)$ gates. Similar to the definition of Γ_W in Eq. (3.65), we have

$$\Gamma_A |c\rangle = (-1)^{\sum_{j=1}^{n/2} \sum_{k=n/2+1}^{n} (W)_{j,k} c_j c_k} |c\rangle. \tag{3.103}$$

To implement Γ_W, it suffices to compute the exponent of -1 using mod 2 arithmetic, and the exponent can be written as

$$\begin{bmatrix} c_1 & \cdots & c_{\frac{n}{2}} \end{bmatrix} W^{(12)} \begin{bmatrix} c_{\frac{n}{2}+1} \\ \vdots \\ c_n \end{bmatrix}. \tag{3.104}$$

Again, by Proposition 3.10, the multiplication of a Hankel matrix with a vector reduces to computing a convolution. We first compute $e_{\frac{n}{2}+1}, \ldots, e_n \in \mathbb{Z}_2$ satisfying

$$\begin{bmatrix} e_{\frac{n}{2}+1} \\ \vdots \\ e_n \end{bmatrix} = W^{(12)} \begin{bmatrix} c_{\frac{n}{2}+1} \\ \vdots \\ c_n \end{bmatrix} \tag{3.105}$$

with a fast algorithm for polynomial multiplication over $\mathbb{Z}_2$ using $O(n \log n \log\log n)$ gates. Since the convolution is with respect to the entries of W, which are constants in our setting, the multiplication can be implemented using Clifford gates (in fact, CNOT gates). Then the phase in Eq. (3.65) can be computed by applying $O(n)$ controlled-Z gates between the bits $e_{\frac{n}{2}+1}, \ldots, e_n$ and $c_1, \ldots, c_{\frac{n}{2}}$, respectively.

The implementation of Γ_B and Γ_C are equivalent to the implementations of the original instance of size $n/2$. For the base case of the recursion (where W is a 1×1 matrix), a single S gate implements Γ_W. The gate cost $G(n)$ of the recursive implementation satisfies the recurrence relation:

$$G(n) = 2G(n/2) + O(n \log n \log\log n). \tag{3.106}$$

It follows that

$$G(n) \in O(n \log^2 n \log\log n). \tag{3.107}$$

This recursive construction uses polynomial multiplication in each recursion step. Therefore, it follows from the circuit depth analysis for polynomial multiplication in Subsection 3.4.1 that circuit depth of this construction is $O(\log n + \log \frac{n}{2} + \cdots + 1) = O(\log^2 n)$.

Pauli mixing from $\Delta_2(\mathrm{GF}(2^n))$ and $\nabla_2(\mathrm{GF}(2^n))$ in different bases

Now, we show how to achieve Pauli mixing by implementing U_M that induces $M \in \Delta_2(\mathrm{GF}(2^n))$ and $\widehat{U}_M$ that induces $M \in \nabla_2(\mathrm{GF}(2^n))$. In the following, we first show how to generate and construct an element of an ensemble of unitaries, which no longer corresponds to $\mathrm{SL}_2(\mathrm{GF}(2^n))$. Then, we prove that this ensemble is Pauli mixing, and hence a unitary 2-design.

The construction is based on the following decomposition of elements of $\mathrm{SL}_2(\mathrm{GF}(2^n))$,

which follows from Eq. (3.108):

$$\begin{pmatrix} \alpha & \gamma \\ \beta & \delta \end{pmatrix} = \begin{cases} \begin{pmatrix} 1 & 0 \\ \beta/\alpha & 1 \end{pmatrix} \begin{pmatrix} \alpha & \gamma \\ 0 & \alpha^{-1} \end{pmatrix} & \text{if } \alpha \neq 0 \\ \begin{pmatrix} \gamma & 0 \\ \delta & \gamma^{-1} \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} & \text{if } \alpha = 0. \end{cases} \tag{3.108}$$

All matrices in this decomposition are lower-triangular, upper-triangular, or $\left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$. Unitaries that induce lower-triangular matrices can be implemented in the primal basis; unitaries that induce upper-triangular matrices can be implemented in the dual basis, and the unitary that induces $\left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$ can be implemented in any self-dual basis (by $H^{\otimes n}$).

The procedure to generate an element of the ensemble is as follows.

Procedure 3.1: Generation procedure

1 Sample $\left(\begin{smallmatrix} \alpha & \gamma \\ \beta & \delta \end{smallmatrix}\right) \in \mathrm{SL}_2(\mathrm{GF}(2^n))$ uniformly at random;
2 **if** $\alpha \neq 0$ **then**
3 $\quad M_1 \leftarrow \left(\begin{smallmatrix} \alpha & \gamma \\ 0 & \alpha^{-1} \end{smallmatrix}\right)$;
4 $\quad M_2 \leftarrow \left(\begin{smallmatrix} 1 & 0 \\ \beta/\alpha & 1 \end{smallmatrix}\right)$;
5 $\quad$ Construct the Clifford unitary $U_{M_2}\widehat{U}_{M_1}$ (composition of two circuits);
6 **end**
7 **else if** $\alpha = 0$ **then**
8 $\quad M \leftarrow \left(\begin{smallmatrix} \gamma & 0 \\ \delta & \gamma^{-1} \end{smallmatrix}\right)$;
9 $\quad$ Construct the Clifford unitary $U_M H^{\otimes n}$ (composition of two circuits);
10 **end**

Note that the construction in steps 2-5 of Procedure 3.1 corresponds to the first case of Eq. (3.108), and the construction in steps 7-9 corresponds to the second case of Eq. (3.108). Either case results in a Clifford unitary that can be implemented with $O(n \log n \log\log n)$ gates (including non-Clifford gates), or with $O(n \log^2 n \log\log n)$ Clifford gates. Although the subset of all Clifford unitaries that can be generated by Procedure 3.1 does not have the structure of $\mathrm{SL}_2(\mathrm{GF}(2^n))$ because different bases are used for the components, we show in the following that this ensemble is indeed Pauli mixing.

First, we consider the mixing property over the Paulis that result from conjugating by U_M for a random $M \in \Delta_2(\mathrm{GF}(2^n))$. Partition the nonzero elements of $\mathrm{GF}(2^n) \times \mathrm{GF}(2^n)$

into the following disjoint subsets:

$$\mathrm{R}_1 = \left\{ \binom{a}{b} \in \mathrm{GF}(2^n) \times \mathrm{GF}(2^n) : a = 0 \text{ and } b \neq 0 \right\}, \tag{3.109}$$

$$\mathrm{R}_2 = \left\{ \binom{a}{b} \in \mathrm{GF}(2^n) \times \mathrm{GF}(2^n) : a \neq 0 \right\}. \tag{3.110}$$

A random element $M \in \Delta_2(\mathrm{GF}(2^n))$ uniformly mixes elements in R_1 as well as in R_2 in the following sense. (A similar result holds for $\nabla_2(\mathrm{GF}(2^n))$ with a and b switched in the definitions of R_1 and R_2.)

Lemma 3.11. *Let $M \in \Delta_2(\mathrm{GF}(2^n)$ be chosen uniformly at random. Then, for any $\binom{a}{b} \in \mathrm{R}_1$, the distribution $M\binom{a}{b}$ is uniform over R_1; for any $\binom{a}{b} \in \mathrm{R}_2$, the distribution $M\binom{a}{b}$ is uniform over R_2.*

Proof. First, we consider R_1. Note that $\binom{d^{-1}\ 0}{0\ \ d}\binom{0}{1} = \binom{0}{d}$. It follows that any element $\binom{0}{b} \in \mathrm{R}_1$ can be mapped to $\binom{0}{d_1}$ and then mapped to any $\binom{0}{d_2} \in \mathrm{R}_1$ by elements in $\Delta_2(\mathrm{GF}(2^n))$. Suppose there exist distinct $\binom{0}{d_1}$ and $\binom{0}{d_2}$ such that $\Pr_M\{M\binom{0}{b} = \binom{0}{d_1}\} > \Pr_M\{M\binom{0}{b} = \binom{0}{d_2}\}$. There exists $M' \in \mathrm{R}_1$ such that $M'\binom{0}{d_1} = \binom{0}{d_2}$. Then, we have $\Pr_M\{M'M\binom{0}{b} = \binom{0}{d_2}\} \geq \Pr_M\{M\binom{0}{b} = \binom{0}{d_1}\}$. Since the distribution over $M'M$ is the same as the distribution over M, the last inequality implies that $\Pr_M\{M\binom{0}{b} = \binom{0}{d_1}\} \leq \Pr_M\{M\binom{0}{b} = \binom{0}{d_2}\}$, which is a contradiction.

Similarly, we consider R_2. Note that $\binom{c\ \ 0}{d\ c^{-1}}\binom{1}{0} = \binom{c}{d}$. It follows that any element $\binom{a}{b} \in \mathrm{R}_1$ can be mapped to $\binom{c_1}{d_1}$ and then mapped to any $\binom{c_2}{d_2} \in \mathrm{R}_2$ by elements in $\Delta_2(\mathrm{GF}(2^n))$. Suppose there exist distinct $\binom{c_1}{d_1}$ and $\binom{c_2}{d_2}$ such that $\Pr_M\{M\binom{a}{b} = \binom{c_1}{d_1}\} > \Pr_M\{M\binom{a}{b} = \binom{c_2}{d_2}\}$. There exists $M' \in \mathrm{R}_2$ such that $M'\binom{c_1}{d_1} = \binom{c_2}{d_2}$. Then, we have $\Pr_M\{M'M\binom{a}{b} = \binom{c_2}{d_2}\} \geq \Pr_M\{M\binom{a}{b} = \binom{c_1}{d_1}\}$. Since the distribution over $M'M$ is the same as the distribution over M, the last inequality implies that $\Pr_M\{M\binom{a}{b} = \binom{c_1}{d_1}\} \leq \Pr_M\{M\binom{a}{b} = \binom{c_2}{d_2}\}$, which is a contradiction. □

It is fruitful to organize the n-qubit Paulis in $\mathrm{Q}(\mathbb{C}^{2^n})$ into rows and columns where $X^{\lceil a \rceil}Z^{\lfloor b \rfloor}$ is in column a and row b. The first column and row are labeled by $a = 0$ and $b = 0$, and they are referred to as the *zero column* and the *zero row*, respectively. The remaining rows and columns are referred to as the *nonzero rows* and the *nonzero columns*. The relative ordering of the nonzero rows and nonzero columns does not affect the analysis. An example of such an organization for $n = 2$ is shown as the following:

$$\begin{array}{cccc} & IX & XI & XX \\ IZ & IY & XZ & XY \\ ZI & ZX & YI & YX \\ ZZ & ZY & YZ & YY \end{array} \tag{3.111}$$

where the identity Pauli (in the upper-left corner) is excluded. In this example, we need the uniform distribution on the 15 items to achieve Pauli mixing.

According to Lemma 3.11, conjugating by U_M for a uniformly sampled $M \in \Delta_2(\mathrm{GF}(2^n))$ results in a uniform mixing of the zero column (as illustrated in Figure 3.6) and a uniform mixing of all nonzero columns (as illustrated in Figure 3.7). We refer to this mixing as the *lower-triangular Pauli mixing*. Similarly, we define the *upper-triangular Pauli mixing*, which is corresponding to a transposed versions of Figures 3.6 and 3.7. The upper-triangular Pauli mixing can be achieved by conjugating $\widehat{U}_M$ for uniformly sampled $M \in \nabla_2(\mathrm{GF}(2^n))$.

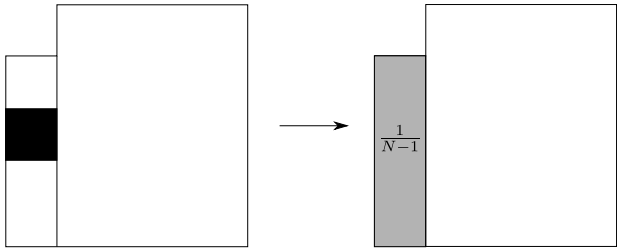

Figure 3.6: Illustration of the lower-triangular Pauli mixing within the zero column ($N = 2^n$).

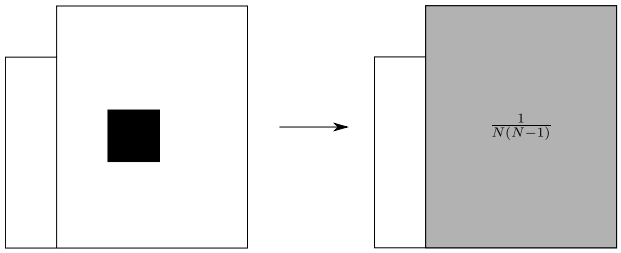

Figure 3.7: Illustration of the lower-triangular Pauli mixing within the nonzero columns ($N = 2^n$).

Now, we consider another form of mixing, which is referred to as the *column Pauli mixing*, that is achieved by choosing $M = \left(\begin{smallmatrix} 1 & 0 \\ \beta & 1 \end{smallmatrix}\right)$ for a uniformly sampled $\beta \in \mathrm{GF}(2^n)$, and then conjugating by the U_M. The effect of the column Pauli mixing is that the Paulis in the zero column do not change (as illustrated in Figure 3.8), and any Pauli in a nonzero column mixes within its column (as illustrated in Figure 3.9).

It follows from Eq. (3.108) that Procedure 3.1 is equivalent to applying a probabilistic mixture of the two procedures below: with probability $\frac{2^n}{2^n+1}$, Procedure 3.2 is applied; with

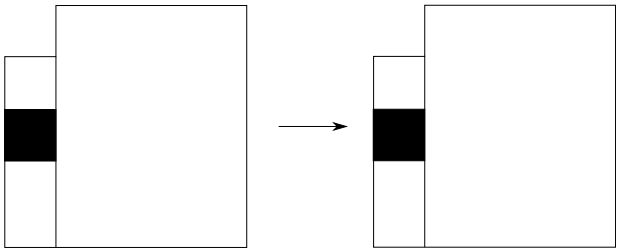

Figure 3.8: Illustration of the column Pauli mixing for the zero column ($N = 2^n$).

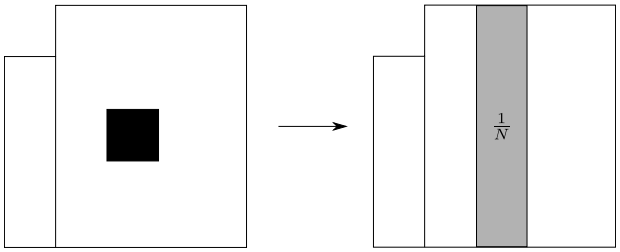

Figure 3.9: Illustration of the column Pauli mixing for the nonzero columns ($N = 2^n$).

probability $\frac{1}{2^n+1}$, Procedure 3.3 is applied. The probability $\frac{1}{2^n+1}$ is the probability that $\alpha = 0$ for a random $\left(\begin{smallmatrix} \alpha & \gamma \\ \beta & \delta \end{smallmatrix}\right) \in \mathrm{SL}_2(\mathrm{GF}(2^n))$.

Procedure 3.2:

1 Apply an upper-triangular Pauli mixing operation;
2 Apply a column Pauli mixing operation (independently from the first step);

We prove that the above probabilistic mixture of Procedure 3.2 and Procedure 3.3 results in Pauli mixing as in the following lemma.

Lemma 3.12. *The stochastic process of applying either Procedure 3.2 with probability* $\frac{2^n}{2^n+1}$ *or Procedure 3.3 with probability* $\frac{1}{2^n+1}$ *is Pauli mixing.*

Proof. To simplify the presentation, assume $N = 2^n$. First, consider the case where an initial Pauli is in the zero row (i.e., $b = 0$ and the Pauli is of the form $X^{\lceil a \rceil}$ for some $a \neq 0$). Then, as illustrated in Figure 3.10, if Procedure 3.2 is applied, it results in a uniform distribution over the Paulis in the nonzero columns, and the probability for each Pauli is $\frac{1}{N(N-1)}$; if Procedure 3.3 is applied, it results in a uniform distribution of the Paulis in the zero column, and the probability for each Pauli is $\frac{1}{N-1}$. Consider the probabilistic mixture

Procedure 3.3:

1 Apply $H^{\otimes n}$ (to transpose the layout of Paulis);
2 Apply a lower-triangular Pauli mixing operation;

of the two procedures. Since $\frac{N}{N+1} \cdot \frac{1}{N(N-1)} = \frac{1}{N^2-1}$ and $\frac{1}{N+1} \cdot \frac{1}{N-1} = \frac{1}{N^2-1}$, it implies the uniform distribution.

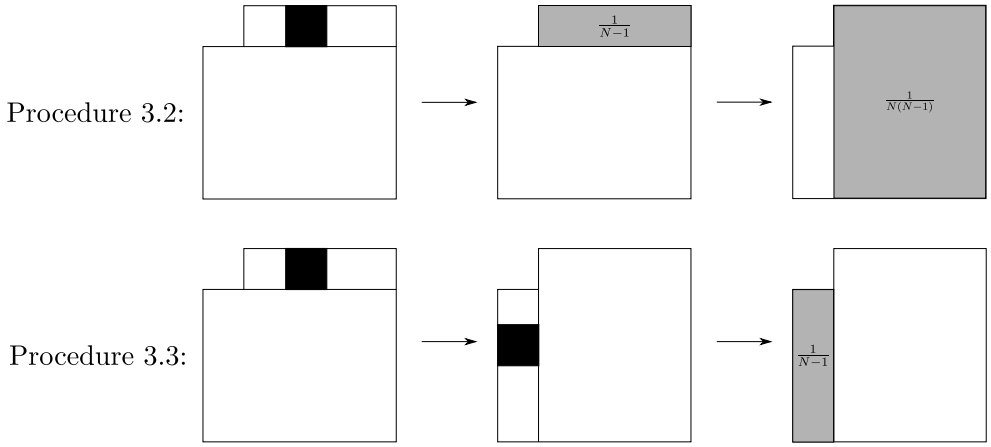

Figure 3.10: Illustration of the mixing procedure starting in the zero row ($N = 2^n$).

Next, consider the case where an initial Pauli is not in the zero row (i.e., $b \neq 0$ and the Pauli is of the form $X^{\lceil a \rceil} Z^{\lfloor b \rfloor}$). Then, as illustrated in Figure 3.11, if Procedure 3.2 is applied, it results in a two-level distribution of Paulis: either a uniform distribution over the Paulis in the zero column where the probability of each Pauli is $\frac{1}{N(N-1)}$, or a uniform distribution over the Paulis in the nonzero columns where the probability of each Pauli is $\frac{1}{N^2}$; if Procedure 3.3 is applied, it results in a uniform distribution over the nonzero columns, and the probability of each Pauli is $\frac{1}{N(N-1)}$. Consider the probabilistic mixture of the two procedures. Since $\frac{N}{N+1} \cdot \frac{1}{N(N-1)} = \frac{1}{N^2-1}$ and $\frac{N}{N+1} \cdot \frac{1}{N^2} + \frac{1}{N+1} \cdot \frac{1}{N(N-1)} = \frac{1}{N^2-1}$, it also implies the uniform distribution.

□

3.4.3 Lower bounds for the size and depth of unitary 2-designs

Let $\mathcal{E} = \{p_j, U_j\}_{j=1}^k$ be any exact unitary 2-design on n qubits. In the following, we show that with constant probability, the set of unitaries has circuit size $\Omega(n)$ and depth $\Omega(\log n)$,

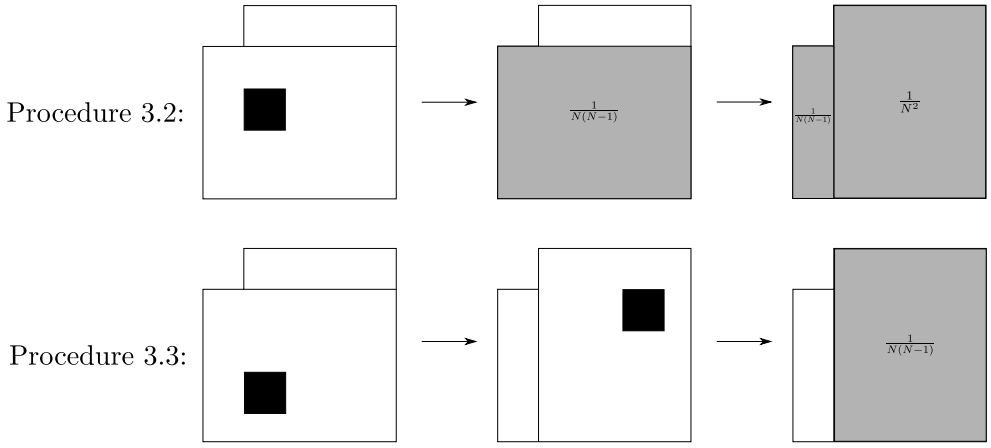

Figure 3.11: Illustration of the mixing procedure starting in a nonzero row ($N = 2^n$).

assuming a universal gate set consisting of 1- and 2-qubit gates. This proof also applies to unitary 2-designs that approximate the exact operation under Definition 3.2 or 3.3 in terms of the diamond norm.

For the circuit size, suppose that the circuit for U_j acts nontrivially on s_j qubits. We show that $\sum_{j=1}^{k} p_j s_j \geq n/2$, so the circuit size is at least $n/2$ on average. Since $\mathcal{E}$ is a unitary 2-design, it implements the full bilateral twirl. Therefore, the linear map $\rho \mapsto \sum_{j=1}^{k} p_j U_j \rho U_j^\dagger = \frac{I_{\mathbb{C}^{2^n}}}{2^n}$ is the completely depolarizing channel on n qubits. For each ℓ, consider the input state $|0\rangle\langle 0|_{\mathbb{C}^2}$ on the ℓ-th qubit, and the output on this qubit is $\frac{I_{\mathbb{C}^2}}{2}$. Then, the cumulated probability of one of the unitaries U_j's acting on this qubit is at least $1/2$. Define a matrix M with rows labeled by $j = 1, \ldots, k$ and columns labeled by $\ell = 1, \ldots, n$, and $(M)_{j,\ell} = p_j$ if U_j acts nontrivially on the ℓ-th qubit. It implies that the sum of each column of M is at least $1/2$. Also, the sum of the j-th row is $s_j p_j$. The total of the row sums is equal to the total of column sums, and thus we have that $\sum_{j=1}^{k} p_j s_j \geq n/2$. Furthermore, if $\sum_{j:s_j<n/4} p_j > 2/3$, then $\sum_{j=1}^{k} p_j s_j \not\geq n/2$. Therefore, with probability at least $1/3$, the circuit has size at least $n/4$.

For the circuit depth, consider the bilateral twirl $\mathcal{T}_\mathcal{E}$ on $Z \otimes I^{\otimes n-1} \otimes Z \otimes I^{\otimes n-1}$:

$$\mathcal{T}_\mathcal{E}[Z \otimes I^{\otimes n-1} \otimes Z \otimes I^{\otimes n-1}] = \sum_{j=1}^{k} p_j (U_j(Z \otimes I^{\otimes n-1})U_j^\dagger) \otimes (U_j(Z \otimes I^{\otimes n-1})U_j^\dagger). \quad (3.112)$$

Express each $U_j(Z \otimes I^{\otimes n-1})U_j^\dagger$ as a linear combination of Paulis, and define t_j as the number of qubits that are acted on nontrivially by at least one of the terms in this lin-

ear combination of Paulis. Intuitively, t_j is the number of qubits that $U_j(Z \otimes I^{\otimes n-1})U_j^\dagger$ nontrivially acts on. Since each gate interacts with at most two qubits, the depth of the circuit for U_j is at least $\log t_j$. This is because $Z \otimes I^{\otimes n-1}$ nontrivially acts on only one qubit, and conjugating by each layer of the circuit for U_j can at most double the qubits it acts on. Now we show that for most $U_j(Z \otimes I^{\otimes n-1})U_j^\dagger$ it holds that $t_j \geq n/2$. By the proof of Lemma 3.3, we have

$$\mathcal{T}_{\mathcal{E}}[Z \otimes I^{\otimes n-1} \otimes Z \otimes I^{\otimes n-1}] = \frac{1}{2^{2n}-1} \sum_{R_\ell \in \mathrm{Q}(\mathbb{C}^{2^n}) \setminus \{I\}} R_\ell \otimes R_\ell. \tag{3.113}$$

The fraction of R_ℓ's that nontrivially act on less than $n/2$ qubits is equal to

$$4^{-n} \sum_{\ell=0}^{\lfloor n/2 \rfloor} \binom{n}{\ell} 3^\ell \leq 4^{-n} \sum_{\ell=0}^{\lfloor n/2 \rfloor} \binom{n}{\ell} 3^{n/2} \leq 4^{-n} \cdot \frac{1}{2} \cdot 2^n \cdot 3^{n/2} \approx \frac{1}{2} \cdot 0.866^n. \tag{3.114}$$

Therefore, we have $\sum_{j:t_j \geq n/2} p_j \to 1$. In particular, we have $\sum_{j:t_j \geq n/2} p_j \geq 1/2$, since otherwise, the RHS of Eqns. (3.112) and (3.113) cannot be equal.

Chapter 4

Continuous-Time Evolution of Markovian Open Quantum Systems

In this chapter, we study a description of the continuous-time evolution of Markovian open quantum systems. A quantum system is *closed* when it has no interaction with the environment (i.e., some external system). Consider the Hilbert space $\mathbb{C}^N$. The continuous-time evolution of a closed quantum system can be described by the Schrödinger equation:

$$\frac{d}{dt}|\psi_t\rangle = -iH|\psi_t\rangle, \tag{4.1}$$

where $H \in \mathrm{L}(\mathbb{C}^N)$, referred to as the *Hamiltonian*, is a Hermitian matrix. In the following, we assume that H is time-independent. The solution to the Schrödinger equation is

$$|\psi_t\rangle = e^{-iHt}|\psi_0\rangle, \tag{4.2}$$

which is a time-dependent unitary operation. If a quantum system has interaction with the environment, it is called an *open quantum system*. In this thesis, we consider a class of open quantum systems where $\rho_{t+\delta}$ is completely determined by ρ_t for any $\delta > 0$. These open quantum systems are often referred to as *Markovian open quantum systems*. It should be noted that, technically, these are open systems in an idealized sense, that assumes there is no transfer of information from the environment back to the system (which can lead to good approximations if the backward flow of information is very small). Markovian open quantum system dynamics are expressed (without any reference to an environment) in terms of dynamical semigroups that are generated by the following elegant generalization of the Schrödinger equation

$$\frac{d}{dt}\rho_t = \mathcal{L}[\rho_t], \tag{4.3}$$

where $\mathcal{L} \in \mathrm{T}(\mathbb{C}^N)$, called the *Lindbladian*, is a superoperator defined as

$$\mathcal{L}[\rho] = -i[H, \rho] + \sum_{j=1}^{m} \left(L_j \rho L_j^\dagger - \frac{1}{2} L_j^\dagger L_j \rho - \frac{1}{2} \rho L_j^\dagger L_j \right) \tag{4.4}$$

for all $\rho \in \mathrm{D}(\mathbb{C}^N)$, where $H \in \mathrm{L}(\mathbb{C}^N)$ is a Hamiltonian, $L_1, \ldots, L_m \in \mathrm{L}(\mathbb{C}^N)$ are linear operators, and recall that $[H, \rho]$ denotes the operator $H\rho - \rho H$. Eq. (4.4) is called the *Lindblad equation*. The solution to the Lindblad equation is

$$\rho_t = e^{\mathcal{L}t}[\rho_0], \tag{4.5}$$

which is a time-dependent quantum channel (again, assuming that $H, L_1, \ldots, L_m$ are time-independent). Note that in the special case where $L_1 = \cdots = L_m = \mathbf{0}$, Eqns. (4.3) and (4.4) become $\frac{d}{dt}\rho = -i[H, \rho]$, which is equivalent to the Schrödinger equation.

Hamiltonian evolution can be viewed as a semigroup of unitaries $\{U_t \in \mathrm{U}(\mathbb{C}^N) : t \geq 0\}$ generated by H in the sense that $U_t = e^{-iHt}$. Similarly, Lindblad evolution can be viewed as a semigroup of quantum channels $\{\mathcal{M}_t \in \mathrm{C}(\mathbb{C}^N) : t \geq 0\}$ generated by $\mathcal{L}$ in the sense that $\mathcal{M}_t = e^{\mathcal{L}t}$.

In the remainder of this chapter, we first show a derivation of Lindblad equation and then present some examples of Lindblad evolution. Finally, we show that Lindblad evolution can be simulated as Hamiltonian evolution in a larger Hilbert space, and we prove a lower bound for these simulation methods. In this chapter, Sections 4.1 and 4.2 contain introductory material. In Section 4.3, Theorem 4.1 is based on [CW17]; however, Theorem 4.3 is not previously published.

4.1 Macroscopic derivation of the Lindblad equation

There exist microscopic derivations of the Lindblad equation based on the Schrödinger equation in a larger Hilbert space and imposing the Born-Markov approximation and the weak-coupling limit, for example, in [BP07]. In this section, we present a macroscopic derivation of the Lindblad equation, which is closely connected to the Kraus operators of quantum channels. This derivation is based on Preskill's lecture notes [Pre, Chapter 3].

The goal is to obtain an expression of the following linear map:

$$\frac{d}{dt}\rho_t = \mathcal{L}[\rho_t]. \tag{4.6}$$

The definition of the derivative implies that

$$\lim_{\delta \to 0} \frac{\rho_{t+\delta} - \rho_t}{\delta} = \mathcal{L}[\rho_t], \tag{4.7}$$

which implies that, after small time δ, the state ρ_t evolves to

$$\rho_{t+\delta} = \rho_t + \delta \mathcal{L}[\rho_t] + o(\delta), \tag{4.8}$$

where the error $o(\delta)$ is with respect to the trace norm. In the following, we derive an expression of $\mathcal{L}$ by looking at the Kraus operators of the linear map that is an approximation of the above equation (by omitting the $o(\delta)$ term).

To begin with, let us consider the following Kraus representation:

$$\rho_{t+\delta} = \sum_{j=0}^{m+1} E_j \rho_t E_j^\dagger, \tag{4.9}$$

where $E_0, \ldots, E_m \in \mathrm{L}(\mathbb{C}^N)$ are linear operators. The linear map specified in Eq. (4.8) suggests that one of the Kraus operators can be in the form of $I + O(\delta)$ (with respect to the spectral norm) and the spectral norms of the others Kraus operators are in the order of $O(\sqrt{\delta})$. In the following, we argue that any Kraus representation for Eq. (4.8) can be converted to this form. First consider a general Kraus representation with m Kraus operators $\{F_0, \ldots, F_{m-1}\}$. We express F_j as a polynomial in $\sqrt{\delta}$:

$$F_j = F_{j0} + \sqrt{\delta} F_{j1} + \delta F_{j2}, \tag{4.10}$$

and we have

$$F_j \rho F_j^\dagger = F_{j0} \rho F_{j0}^\dagger + \sqrt{\delta}(F_{j0} \rho F_{j1}^\dagger + F_{j1} \rho F_{j0}^\dagger) + \delta(F_{j1} \rho F_{j1}^\dagger + F_{j0} \rho F_{j2}^\dagger + F_{j2} \rho F_{j0}^\dagger). \tag{4.11}$$

Compare the above equation to Eq. (4.8), we have

$$\sum_{j=0}^{m-1} F_{j0} \rho F_{j0}^\dagger = \rho. \tag{4.12}$$

By [NC00, Theorem 8.2], there exist $\alpha_0, \ldots, \alpha_{m-1} \in \mathbb{C}$ such that

$$F_{j0} = \alpha_j I, \tag{4.13}$$

for all $j \in \{0, \ldots, m-1\}$. Moreover, it holds that $\sum_{j=0}^{m-1} |\alpha_j|^2 = 1$. Invoking [NC00, Theorem 8.2] again, this Kraus representation can be converted to another Kraus representation with Kraus operators $\{E_0, \ldots, E_{m-1}\}$ such that

$$E_{00} = I \text{ and} \tag{4.14}$$
$$E_{j0} = \mathbf{0} \text{ for all } j \in \{1, \ldots, m-1\}. \tag{4.15}$$

Now, Eq (4.11) can be rewritten as

$$E_0 \rho E_0^\dagger = \rho + \sqrt{\delta}(\rho E_{01}^\dagger + E_{01}\rho) + \delta(E_{01}\rho E_{01}^\dagger + \rho E_{02}^\dagger + E_{02}\rho), \text{ and} \tag{4.16}$$
$$E_j \rho E_j^\dagger = \delta(E_{j1}\rho E_{j1}^\dagger) \text{ for all } j \in \{1, \ldots, m-1\}. \tag{4.17}$$

Comparing the above two equations to Eq. (4.8), we have that

$$\rho E_{01}^\dagger + E_{01}\rho = \mathbf{0}, \tag{4.18}$$

for all $\rho \in \mathrm{D}(\mathbb{C}^N)$. There are two cases for the above equation to hold. First, $E_{01} = \mathbf{0}$ and we are done. Second, $E_{01} = \alpha i I$ for some nonzero $\alpha \in \mathbb{R}$. In this case, the action of the Kraus operator $E_0 = I + \sqrt{\delta}\alpha i I$ is equivalent to the action of two Kraus operators I and $\alpha\sqrt{\delta} I$.

Therefore, let $L_0, \ldots, L_m \in \mathrm{L}(\mathbb{C}^N)$ arbitrary linear operators acting on $\mathbb{C}^N$, and we write

$$E_0 = I + \delta L_0, \tag{4.19}$$
$$E_j = \sqrt{\delta} L_j, \text{ for } j \in \{1, \ldots, m\}. \tag{4.20}$$

It is fruitful to write L_0 in the form $L_0 = -iH + K$ for Hermitian operators H and K. We can determine K by the normalization condition for Kraus operators. In fact, we have

$$I = \sum_{j=0}^{m} E_j^\dagger E_j \tag{4.21}$$
$$= (I + \delta(iH + K))(I + \delta(-iH + K)) + \sum_{j=1}^{m} \delta L_j^\dagger L_j \tag{4.22}$$
$$= I + \delta(2K + \sum_{j=1}^{m} L_j^\dagger L_j) + O(\delta^2). \tag{4.23}$$

This implies that

$$K = -\frac{1}{2}\sum_{j=1}^{m} L_j^\dagger L_j + O(\delta). \tag{4.24}$$

The big-O notations in the above equations are with respect to the spectral norm. Then Eq. (4.19) can be written as

$$E_0 = I + \delta\left(-iH - \frac{1}{2}\sum_{j=1}^{m} L_j^\dagger L_j\right) + O(\delta^2). \tag{4.25}$$

Now we are ready to derive the Lindblad equation. Consider the expressions of $\rho_{t+\delta}$ in Eq. (4.8), and the Kraus operators in the form of Eqns. (4.20) and (4.25). We have

$$\rho_{t+\delta} = \sum_{j=0}^{m} E_j \rho_t E_j^\dagger \tag{4.26}$$

$$= \rho_t + \delta\left(-i[H,\rho_t] + \sum_{j=1}^{m}\left(L_j\rho_t L_j^\dagger - \frac{1}{2}L_j^\dagger L_j\rho_t - \frac{1}{2}\rho_t L_j^\dagger L_j\right)\right) + O(\delta^2). \tag{4.27}$$

It follows that

$$\frac{d}{dt}\rho_t = \lim_{\delta\to 0}\frac{\rho_{t+\delta}-\rho_t}{\delta} = -i[H,\rho_t] + \sum_{j=1}^{m}\left(L_j\rho_t L_j^\dagger - \frac{1}{2}L_j^\dagger L_j\rho_t - \frac{1}{2}\rho_t L_j^\dagger L_j\right), \tag{4.28}$$

for all $t \geq 0$. This is the Lindblad equation as in Eq. (4.4)

The first term of the Lindblad equation, $\frac{d}{dt}\rho_0 = -i[H,\rho_0]$, is a generalization of the Schrödinger equation to apply to density operators, and it describes the unitary part of the Lindblad evolution. Intuitively, the other terms correspond to the interaction with the environment. Each L_j is called a *jump operator* or a *Lindblad operator*. The $L_j\rho L_j^\dagger$ terms can be interpreted as one of the possible jumps. The $-\frac{1}{2}L_j^\dagger L_j\rho - \frac{1}{2}\rho L_j^\dagger L_j$ terms can be interpreted as the (non-unitary) evolution when no jump occurs.

The Lindblad equation can be viewed as an idealization of the frequently occurring physical scenario where a quantum system evolves jointly with a large external environment in a manner where information dissipates from the system into the environment. In quantum information theoretic terms, Lindblad evolution is a continuous-time process that, for any evolution time, is a quantum channel. Moreover, Lindblad evolution is *Markovian* in the sense that, for any $\delta > 0$, the state at time $t+\delta$ is a function of the state at time t alone (i.e., is independent of the state before time t).

4.2 Examples of Lindblad evolution

In this section, we present the Lindblad evolution for three quantum channels: the depolarizing channel, the phase damping channel, and the amplitude damping channel.

The *depolarizing channel* can be used to model a system that undergoes some noisy process. It can be described as the following process. With probability $1-p$, the state remains the same; with probability p, one of the three errors will occur with equal probability: the bit flip error (applying X gate), the phase flip error (applying Z gate), and both bit flip error and phase flip error (applying Y gate). This quantum channel can be described by the Kraus representation:

$$\rho \mapsto (1-p)\rho + \frac{p}{3}X\rho X + \frac{p}{3}Y\rho Y + \frac{p}{3}Z\rho Z, \tag{4.29}$$

for all $\rho \in \mathrm{D}(\mathbb{C}^2)$. It has four Kraus operators, $\sqrt{1-p}I$, $\sqrt{p/3}X$, $\sqrt{p/3}Y$, and $\sqrt{p/3}Z$.

When $p \to 0$, repeatedly applying this quantum channel yields a continuous-time process, which we call the *depolarizing process*, and it can be described by the Lindblad evolution. Following the steps in Section 4.1 (setting $\delta = p$), we obtain the Lindblad operators $L_1 = X/\sqrt{3}$, $L_2 = Y/\sqrt{3}$, and $L_3 = Z/\sqrt{3}$, and the Lindblad equation of the depolarizing process can be written as

$$\frac{d}{dt}\rho = \sum_{j=1}^{3}\left(L_j\rho L_j^\dagger - \frac{1}{2}L_j^\dagger L_j\rho - \frac{1}{2}\rho L_j^\dagger L_j\right) = \frac{1}{3}(X\rho X^\dagger + Y\rho Y^\dagger + Z\rho Z^\dagger) - \rho. \tag{4.30}$$

The *phase damping channel* (a.k.a. the *dephasing channel*) can be used to model a system that is probabilistically losing coherence (its density operator becomes diagonal): with probability $1-p$, the state remains intact; with probability p, the state loses coherence (as an effect of measurement). This process can be described by the Kraus representation:

$$\rho \mapsto (1-p)\rho + pA_1\rho A_1^\dagger + pA_2\rho A_2^\dagger, \tag{4.31}$$

where $A_1 = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 0 \end{smallmatrix}\right)$ and $A_2 = \left(\begin{smallmatrix} 0 & 0 \\ 0 & 1 \end{smallmatrix}\right)$.

Similarly, when $p \to 0$, repeatedly applying this quantum channel yields a continuous-time process, which we call the *phase damping process*. It can be described by a Lindblad equation. Following the steps in Section 4.1 (setting $\delta = p$), we obtain the Lindblad equation for the phase damping process as:

$$\frac{d}{dt}\rho = \sum_{j=1}^{2}\left(L_j\rho L_j^\dagger - \frac{1}{2}L_j^\dagger L_j\rho - \frac{1}{2}\rho L_j^\dagger L_j\right), \tag{4.32}$$

where $L_1 = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 0 \end{smallmatrix}\right)$ and $L_2 = \left(\begin{smallmatrix} 0 & 0 \\ 0 & 1 \end{smallmatrix}\right)$.

The *amplitude damping channel* can be used to model a physical process called "spontaneous emission" (where a system transitions from a higher energy state to a lower energy state and emits a photon). In this thesis, we focus on its information-theoretical point of view and computational applications. This channel can be described by the Kraus representation:

$$\rho \mapsto A_1 \rho A_1^\dagger + A_2 \rho A_2^\dagger, \tag{4.33}$$

where $A_1 = \left(\begin{smallmatrix} 1 & 0 \\ 0 & \sqrt{1-p} \end{smallmatrix}\right)$ and $A_2 = \left(\begin{smallmatrix} 0 & \sqrt{p} \\ 0 & 0 \end{smallmatrix}\right)$ are the Kraus operators[1].

Again, when $p \to 0$, repeatedly applying this quantum channel yields a continuous-time process, which we call the *amplitude damping process*. It can be described by a Lindblad equation. Following the steps in Section 4.1 (setting $\delta = p$), we obtain the Lindblad equation for the phase damping process as:

$$\frac{d}{dt}\rho = L\rho L^\dagger - \frac{1}{2}L^\dagger L\rho - \frac{1}{2}\rho L^\dagger L, \tag{4.34}$$

where $L = \left(\begin{smallmatrix} 0 & 1 \\ 0 & 0 \end{smallmatrix}\right)$.

Note that the depolarizing channel defined in Eq. (4.29) is a *mixed unitary channel*: it can be written as a probability distribution of unitary operations. The amplitude damping channel defined in Eq. (4.33) is not unital[2], and thus is not a mixed unitary channel. Actually, it cannot be written as *any* linear combination of other quantum channels. To see why this is so, first observe that for A_1 and A_2 defined in Eq. (4.33), the set $\{A_1^\dagger A_1, A_1^\dagger A_2, A_2^\dagger A_1, A_2^\dagger A_2\}$ is linearly independent. Then, the statement follows from [Wat18, Theorem 2.31]. Such quantum channels are called *extremal channels*.

4.3 Lower-bound of simulation as Hamiltonian evolution in a larger Hilbert space

Lindblad evolution can be intuitively thought of as Hamiltonian evolution in a larger system that includes an ancilla register, but where the ancilla register is being continually reset

[1] In Chapter 6, we use another form of Kraus operators $\left(\begin{smallmatrix} 1 & 0 \\ 0 & e^{-\lambda/2} \end{smallmatrix}\right)$ and $\left(\begin{smallmatrix} 0 & \sqrt{1-e^{-\lambda}} \\ 0 & 0 \end{smallmatrix}\right)$ for the amplitude damping channel. Note that these two forms of Kraus operators are equivalent by setting $p = 1 - e^{-\lambda}$. In particular, when $\lambda \to 0$, $\lambda \approx 1 - e^{-\lambda}$.

[2] A linear map $\mathcal{M} \in \mathrm{T}(\mathbb{C}^N, \mathbb{C}^M)$ is *unital* if $\mathcal{M}[I_{\mathbb{C}^N}] = I_{\mathbb{C}^M}$.

to its initial state. To make this more precise, consider a time interval $[0, t]$, and divide it into N subintervals of length $\delta = \frac{t}{N}$ each. At the beginning of each subinterval, reset the state of the ancilla register to its initial state, and then let the joint system-ancilla evolve under a Hamiltonian J and the system itself evolve under H. Let the evolution time for J be $\sqrt{\delta} = \sqrt{t/N}$ and the evolution time for H be $\delta = t/N$. This process, illustrated in Figure 4.1, converges to the true Lindblad evolution as N approaches ∞. For the specific evolution described by Eq. (4.4), it suffices to set the ancilla register to $\mathbb{C}^{m+1}$ (with initial state $|0\rangle$) and the Hamiltonian J to the block matrix

$$J = \begin{pmatrix} \mathbf{0} & L_1^\dagger & \cdots & L_m^\dagger \\ L_1 & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ L_m & \mathbf{0} & \cdots & \mathbf{0} \end{pmatrix}. \tag{4.35}$$

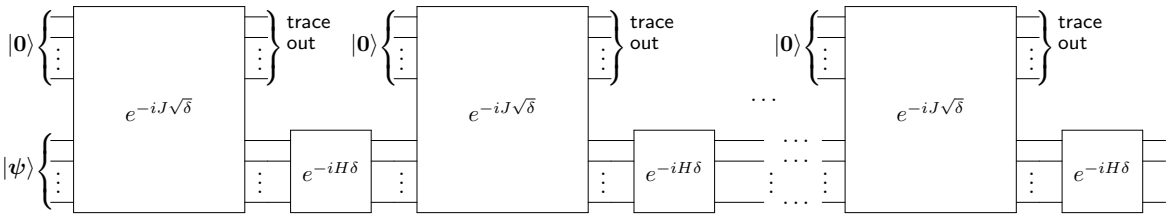

Figure 4.1: Lindblad evolution for time t approximated by unitary operations. There are N iterations and $\delta = t/N$. This converges to Lindblad evolution as $N \to \infty$.

A remarkable property of this way of representing Lindblad evolution is that the rate at which the Hamiltonian J evolves is effectively infinite: Lindblad evolution for time t/N is simulated by a process that includes evolution by J for time $\sqrt{t/N}$, so the rate of the evolution scales as

$$\frac{\sqrt{t/N}}{t/N} = \sqrt{\frac{N}{t}}, \tag{4.36}$$

which diverges as $N \to \infty$. Moreover, the total Hamiltonian evolution time of J in Figure 4.1 is $N\sqrt{t/N} = \sqrt{Nt}$, which also diverges. We are interested in whether much more efficient simulations of Lindblad evolution are possible, such as $O(t \operatorname{polylog}(t/\epsilon))$. This problem is addressed in Chapter 5.

In the following, we prove that, in general, the above scaling phenomenon is necessary for simulating time-independent Lindblad evolution in terms of time-independent Hamiltonian evolution along the lines of the overall structure of Figure 4.1. In this sense, *exact*

Lindblad evolution for finite time does not directly correspond to Hamiltonian evolution for *any* finite time. On the other hand, it can be shown that if the scaling of N is at least t^2/ϵ then the final state is an *approximation* within ϵ. Note that then the corresponding total evolution time for J scales as $\sqrt{(t^2/\epsilon)t} = t^{1.5}/\sqrt{\epsilon}$. Therefore, quantum algorithms that simulate Lindblad evolution by first applying the above reduction to Hamiltonian evolution and then efficiently simulating the Hamiltonian evolution are likely to incur scaling that is at least $t^{1.5}/\sqrt{\epsilon}$.

Let $\mathcal{L} \in \mathrm{T}(\mathbb{C}^{2^n})$ be a Lindbladian acting on n qubits over a time interval $[0, T]$. For each initial state, $\mathcal{L}$ associates a *trajectory*, consisting of a density operator ρ_t for all $t \in [0, T]$. Here we show that if this is simulated by Hamiltonian evolution in a larger system with an ancilla register that is continually reset (expressed as a limiting case when $N \to \infty$ in the process illustrated in Figure 4.2) then the total evolution time for this Hamiltonian can be necessarily infinite.

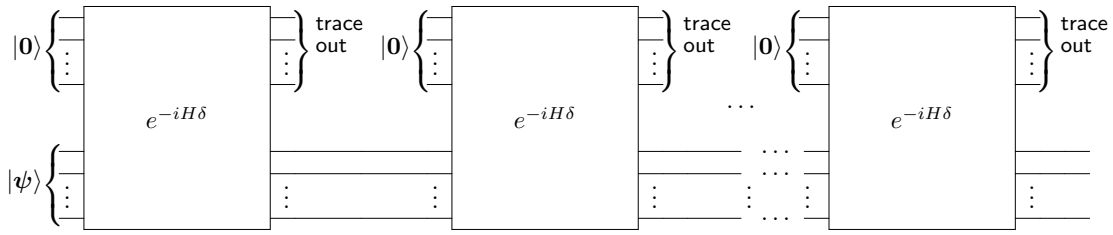

Figure 4.2: N-stage ϵ-precision discretization of the trajectory resulting from $\mathcal{L}$. For each $k \in \{1, \ldots, N\}$, after k stages, the channel should be within ϵ of $\exp\left(\frac{kT}{N}\mathcal{L}\right)$.

Definition 4.1. *Define an* N-stage ϵ-precision discretization of $\mathcal{L}$ for interval $[0, T]$ *as an ancilla register (Hilbert space* $\mathbb{C}^M$*), referred to as* E*, a Hamiltonian* H *acting on the joint system* $\mathbb{C}^M \otimes \mathbb{C}^{2^n}$*, and* $\delta \geq 0$ *such that the channel* $\mathcal{N}_{H\delta}$*, defined as*

$$\mathcal{N}_{H\delta}[\rho] = \mathrm{Tr}_{\mathrm{E}}\left[e^{-iH\delta}(|0\rangle\langle 0| \otimes \rho)e^{iH\delta}\right] \tag{4.37}$$

for all $\rho \in \mathrm{D}(\mathbb{C}^{2^n})$*, has the following property.* $\mathcal{N}_{H\delta}$ *approximates evolution under* $\mathcal{L}$ *in the sense that, for each* $k \in \{1, \ldots, N\}$*,*

$$\left\|(\mathcal{N}_{H\delta})^k - \exp\left(\tfrac{kT}{N}\mathcal{L}\right)\right\|_\diamond \leq \epsilon. \tag{4.38}$$

That is, the N *points generated by* $\mathcal{N}_{H\delta}, (\mathcal{N}_{H\delta})^2, \ldots, (\mathcal{N}_{H\delta})^N$ *approximate the corresponding points on the trajectory determined by* $\mathcal{L}$*.*

Our lower bound is for the *amplitude damping process* on $\mathbb{C}^2$ which is the time-evolution described by the Lindbladian $\mathcal{L}$, where

$$\mathcal{L}[\rho] = L\rho L^\dagger - \tfrac{1}{2}(L^\dagger L\rho + \rho L^\dagger L), \tag{4.39}$$

and $L = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$.

Theorem 4.1. *Any $\frac{1}{4}$-precision N-stage discretization of the amplitude damping process over the time interval $[0, \ln 2]$ has the property that the total evolution time t of H satisfies $t\|H\| \in \Omega(\sqrt{N})$. (Note that this lower bound is independent of the dimension of the ancillary system.)*

Although the above theorem is stated specifically for the amplitude damping process, one can prove similar results for many other Lindblad processes.

To prove Theorem 4.1, we first prove the following *Local Hamiltonian Approximation* lemma. This concerns a scenario where H is a Hamiltonian acting on a joint system of two registers, a system register (the Hilbert space $\mathbb{C}^{2^n}$) and an ancilla register (the Hilbert space $\mathbb{C}^M$) referred to as E, and where E is traced out after this evolution. Informally, the lemma states that, if the initial state is a product state and the evolution time is short, then this process can be approximated by the evolution of another Hamiltonian G that acts on $\mathbb{C}^{2^n}$ alone. This is illustrated in Figure 4.3.

Figure 4.3: The Local Hamiltonian Approximation lemma. The first register is M-dimensional, the second register contains n qubits, and the approximation is within $O(\delta^2)$ (independent of M and n).

Lemma 4.2 (Local Hamiltonian Approximation)**.** *Consider a joint system $\mathbb{C}^M \otimes \mathbb{C}^{2^n}$. We refer to the first register $\mathbb{C}^M$ as* E*. Let $H \in \mathrm{L}(\mathbb{C}^M \otimes \mathbb{C}^{2^n})$ be a Hamiltonian. Define the quantum channel $\mathcal{N}_{H\delta} \in \mathrm{C}(\mathbb{C}^{2^n})$ as*

$$\mathcal{N}_{H\delta}[\rho] = \mathrm{Tr}_{\mathrm{E}}\big[e^{-iH\delta}(|0\rangle\langle 0| \otimes \rho)e^{iH\delta}\big]. \tag{4.40}$$

Then there exists a Hamiltonian $G \in \mathrm{L}(\mathbb{C}^{2^n})$ (with $\|G\| \leq \|H\|$), such that $\mathcal{N}_{G\delta} \in \mathrm{C}(\mathbb{C}^{2^n})$ defined as

$$\mathcal{N}_{G\delta}[\rho] = e^{-iG\delta} \rho \, e^{iG\delta} \tag{4.41}$$

satisfies $\|\mathcal{N}_{H\delta} - \mathcal{N}_{G\delta}\|_1 \in O(\delta^2 \|H\|^2)$.

Proof. Viewing H as a $d \times d$ block matrix (where $d = 2^n$), we have

$$H = \sum_{j=0}^{d-1} \sum_{k=0}^{d-1} |j\rangle\langle k| \otimes H_{jk} \tag{4.42}$$

and we refer to H_{jk} as the (j,k)-*block*. Define D as the diagonal blocks of H, namely

$$D = \sum_{j=0}^{d-1} |j\rangle\langle j| \otimes H_{jj}, \tag{4.43}$$

and set $J = H - D$ (the off-diagonal blocks). Note that $\|D\| \leq \|H\|, \|J\| \leq 2\|H\|$, and $\|e^{-iH\delta} - e^{-iD\delta} e^{-iJ\delta}\| \leq \delta^2 \|H\|^2$, for $\delta > 0$, which permits us to consider the effect of J and D separately.

Now consider the state $e^{-iJ\delta}(|0\rangle \otimes |\psi\rangle)$. We will show that, if the measurement corresponding to projectors $|0\rangle\langle 0|$ and $I - |0\rangle\langle 0|$ is performed on E, then the residual state has trace distance $O(\delta^2)$ from $|0\rangle \otimes |\psi\rangle$. Since the $(0,0)$-block of J is $\mathbf{0}$,

$$J\delta \, (|0\rangle \otimes |\psi\rangle) = \|J\| \delta' |\Psi^{\perp}\rangle, \tag{4.44}$$

where $|\Psi^{\perp}\rangle$ is a state such that $(|0\rangle\langle 0| \otimes I)|\Psi^{\perp}\rangle = 0$ and $0 \leq \delta' \leq \delta$. Therefore,

$$e^{-iJ\delta}(|0\rangle \otimes |\psi\rangle) = \sum_{r=0}^{\infty} \frac{(-iJ\delta)^r}{r!} (|0\rangle \otimes |\psi\rangle) \tag{4.45}$$

$$= |0\rangle \otimes |\psi\rangle - i\|J\|\delta' |\Psi^{\perp}\rangle + \delta'' |\Phi\rangle, \tag{4.46}$$

where $0 \leq \delta'' \leq e^{\delta\|J\|} - 1 - \delta\|J\| \in O(\delta^2 \|J\|^2)$. It follows that, if the above measurement is performed on register E, then the probability of measurement outcome corresponding to $I - |0\rangle\langle 0|$ is at most $(\delta'\|J\|)^2 + (\delta'')^2 \in O(\delta^2\|J\|^2) \in O(\delta^2\|H\|^2)$. This implies that the state after E is traced out from $e^{-iJ\delta}(|0\rangle \otimes |\psi\rangle)$, namely

$$\mathrm{Tr}_{\mathrm{E}}\big[e^{-iJ\delta}(|0\rangle\langle 0| \otimes |\psi\rangle\langle\psi|) e^{iJ\delta}\big], \tag{4.47}$$

has trace distance $O(\delta^2\|H\|^2)$ from the original state $|\psi\rangle\langle\psi|$.

Therefore, for states of the form $|0\rangle \otimes |\psi\rangle$, the operation $e^{-iH\delta}$ can be approximated by $e^{-iD\delta}$ at the cost of an error of $O(\delta^2\|H\|^2)$ in trace distance. The result follows by setting $G = H_{00}$ (the $(0,0)$-block of D). □

Now, we are ready to prove Theorem 4.1.

Proof of Theorem 4.1. It is straightforward to check that, starting with the initial state $|1\rangle\langle 1|$ and evolving by the amplitude process for time $T = \ln 2$ produces the maximally mixed state.

Consider any $\frac{1}{4}$-precision N-stage discretization of this process, with Hamiltonian H and $\delta > 0$. We can apply the Local Hamiltonian Approximation lemma (Lemma 4.2) to approximate each of the N evolutions of H with evolution by a Hamiltonian G that is local to the register $\mathbb{C}^{2^n}$. The result is unitary evolution on $\mathbb{C}^{2^n}$ that approximates the amplitude damping process within trace distance error at most $O(N\delta^2\|H\|^2)$.

Unitary evolution applied to $|1\rangle\langle 1|$ results in a pure state, and the trace distance between any pure state and the maximally mixed state is $\frac{1}{2}$. Therefore, to avoid a contradiction, we must have $N\delta^2\|H\|^2 \in \Omega(1)$, which implies that $\delta \in \Omega\big(\frac{1}{\sqrt{N}\|H\|}\big)$. Thus, the total evolution time of H is $t = N\delta \in \Omega(\sqrt{N}/\|H\|)$, and we conclude that $t\|H\| \in \Omega(\sqrt{N})$. □

Note that $\mathrm{Tr}_E[e^{-iHt}(|0\rangle\langle 0| \otimes \rho)e^{iHt}]$ defines a continuous-time output for all $t \geq 0$. It is natural to consider a simulation of $\mathcal{L}$ by a Hamiltonian H in a larger Hilbert space such that this continuous-time output is always close to $e^{\mathcal{L}t}[\rho]$ for all $t \geq 0$. We refer to this type of simulation as the *trajectory simulation* of $\mathcal{L}$. More precisely, we have the following definition.

Definition 4.2. *A Hamiltonian H is an ϵ-precision* trajectory simulation *of a Lindbladian $\mathcal{L}$ for time interval $[0,T]$, if for any state ρ and $t \in [0,T]$, it holds that*

$$\left\|\mathrm{Tr}_E\left[e^{-iHt}(|0\rangle\langle 0| \otimes \rho)e^{iHt}\right] - e^{\mathcal{L}t}[\rho]\right\|_1 \leq \epsilon. \tag{4.48}$$

A lower bound corresponding to the trajectory simulation is given by the following theorem.

Theorem 4.3. *Any ϵ-precision trajectory simulation H of the amplitude damping process for time interval $[0,T]$ for any $T \geq \ln 2$ has the property that $\|H\| \in \Omega(\sqrt{1/\epsilon})$.*

Again, the above theorem is stated specifically for the amplitude damping process, but one can prove similar results for many other Lindblad processes.

Proof of Theorem 4.3. Let N be the smallest power of 2 that is larger than $\frac{1}{8\epsilon}$. We have $\frac{1}{8\epsilon} \leq N \leq \frac{1}{4\epsilon}$. Given ϵ, let H be an ϵ-precision trajectory simulation for the amplitude damping process for time interval $[0, T]$, as shown in Figure 4.4.

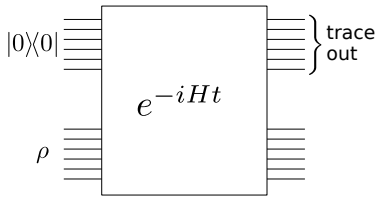

Figure 4.4: A demonstration of a trajectory simulation H of some Lindbladian $\mathcal{L}$.

Note that by Definition 4.2, H is also an ϵ-precision trajectory simulation for the amplitude damping process for the subinterval $[0, \ln 2]$. This trajectory simulation process for time interval $[0, \ln 2]$ can be approximated by a 2-stage discretization as shown in Figure 4.5, where the evolution time for each piece is $\ln 2/2$. In Figure 4.5, the state at ① is

$$\mathrm{Tr}_E \left[e^{-iH \ln 2/2} (|0\rangle\langle 0| \otimes \rho) e^{iH \ln 2/2} \right]. \tag{4.49}$$

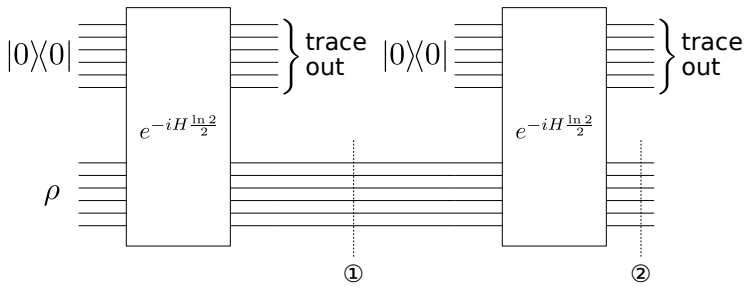

Figure 4.5: Approximating a trajectory simulation H as two stages.

By Definition 4.2, the trace distance between this state and $e^{\mathcal{L} \ln 2/2}[\rho]$ is at most ϵ. Then, at ②, the accumulated error is bounded by 2ϵ. Continue this approximation recursively until there are $N = \frac{1}{4\epsilon}$ pieces, where the evolution time for each piece is at most $\ln 2/N$, as shown in Figure 4.6. By Definition 4.2, the error for each piece is at most ϵ. It follows

that the accumulated approximation error is upper-bounded by $\epsilon N \leq \epsilon \cdot \frac{1}{4\epsilon} = \frac{1}{4}$, and the total evolution time is $\ln 2$. This is a $\frac{1}{4}$-precision N-stage approximation of the amplitude damping process. By Theorem 4.1, $\|H\| \ln 2 \in \Omega(\sqrt{N}) \in \Omega(\sqrt{1/\epsilon})$, and therefore $\|H\| \in \Omega(\sqrt{1/\epsilon})$.

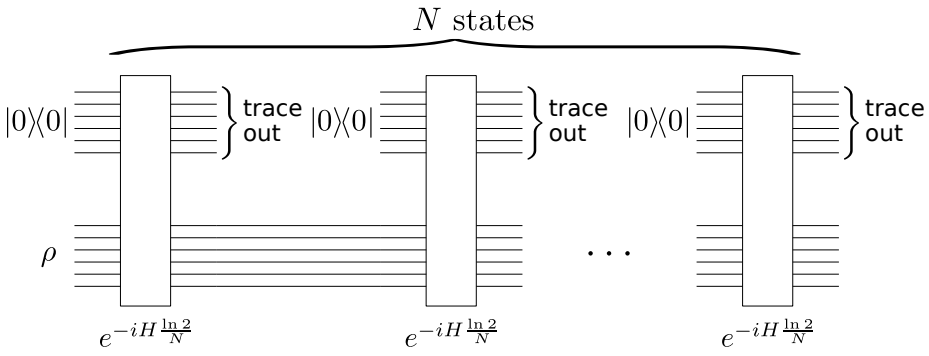

Figure 4.6: Approximating a trajectory simulation H as a $\frac{1}{4}$-precision N-stage discretization ($N = \frac{1}{4\epsilon}$).

□

Chapter 5

Quantum Algorithms for Simulating Markovian Open Quantum Systems

The dynamics of Markovian open quantum systems is captured by Lindblad evolution, which is discussed in Chapter 4. In the problem of simulating a Lindbladian $\mathcal{L}$ for time t, the objective is to build quantum circuits to simulate the quantum channel $e^{\mathcal{L}t}$ with small error. This problem is briefly discussed in Section 1.2, and a lower bound is given in Section 4.3 if this Lindblad evolution is simulated as Hamiltonian evolution in a larger Hilbert space. In this chapter, we give more efficient quantum algorithms for simulating Lindblad evolution. This chapter is based on [CW17] (not including the appendix, whose content is in the previous chapter).

5.1 Previous work and main results

5.1.1 Previous work

As presented in Section 4.1, Hamiltonian evolution is the special case of Lindblad evolution where $L_j = \mathbf{0}$ for all j. Feynman [Fey82] proposed the Hamiltonian simulation problem as a motivation for building quantum computers. Since then, the problem has received considerable attention [ATS03, BCC+15, BCC+17, BCK15, BCG14, BN16, Chi04, Kot14, Llo96, LC17, PP17]. A series of recent algorithms has been discovered that achieve a scaling that is $O(t\,\mathrm{polylog}(t/\epsilon)\mathrm{poly}(n))$, thereby outperforming what can be accomplished by the longstanding Trotter-Suzuki methods [Suz91].

The problem of simulating Lindblad evolution, which is the natural generalization from closed systems to Markovian open systems has received much less attention. Kliesch *et al.* [KBG+11] gave a quantum algorithm for simulating Lindblad evolution in the case where each of H, $L_1, \ldots, L_m$ can be expressed as a sum of local operators (i.e., acting on a constant number of qubits). The cost of this algorithm with respect to t and ϵ (omitting factors of $\mathrm{poly}(n)$ and $\mathrm{poly}(m)$) is $O(t^2/\epsilon)$. Recently, Childs and Li [CL17] improved this to $O(t^{1.5}/\sqrt{\epsilon})$ and also gave an $O((t^2/\epsilon)\,\mathrm{polylog}(t/\epsilon))$ query algorithm for the case where each of H and $L_1, \ldots, L_m$ is sparse and represented in terms of an oracle (i.e., a black-box that accepts a row index j and an integer k, and outputs the position and value of the k-th nonzero entry on row j of a matrix). Another result in [CL17] is an $\Omega(t)$ lower bound for the query complexity for time t when $H = \mathbf{0}$ and $m = 1$.

To the best of our knowledge, none of the previous algorithms for simulating Lindblad evolution has cost $O(t\,\mathrm{polylog}(t/\epsilon)\mathrm{poly}(m, n))$, which is the performance of the algorithm presented in this chapter.

Note that there are simulation algorithms that solve related problems, for example, in [CPdC+15], where the final state is not produced; instead, it simulates the expectation of an observable applied to the final state. There are also interesting classical algorithmic techniques for simulating Lindblad evolution that are feasible when the dimension of the Hilbert space (which is 2^n for n qubits) is not too large. In the classical setting, since the state is known (and stored) explicitly, some "unravellings" of the process (that are state-dependent in general) can be simulated. For example, the random variable corresponding to "the next jump time" (which is highly state-dependent) can be simulated. In the context of quantum algorithms, the current state (even the input state) is unknown and cannot be measured without affecting it.

5.1.2 Main results

Evolution under Eq. (4.4) for time t corresponds to the linear map $e^{\mathcal{L}t}$ (which is a quantum channel for any $t \geq 0$).

Each of the operators $H, L_1, \ldots, L_m \in \mathrm{L}(\mathbb{C}^{2^n})$ corresponds to a $2^n \times 2^n$ matrix. The simulation algorithm is based on a succinct specification of these matrices. Our succinct

specification is as a *linear combination of q Paulis*, defined as

$$H = \sum_{k=0}^{q-1} \beta_{0k} V_{0k} \tag{5.1}$$

$$L_j = \sum_{k=0}^{q-1} \beta_{jk} V_{jk}, \tag{5.2}$$

where, for each $j \in \{0, \ldots, m\}$ and $k \in \{0, \ldots, q-1\}$, V_{jk} is an n-fold tensor product of Paulis (I, X, Y, Z) and a scalar phase $e^{i\theta}$ ($\theta \in [0, 2\pi]$), and $\beta_{jk} \geq 0$.

In the evolution $e^{\mathcal{L}t}$, it is possible to scale up $\mathcal{L}$ by some factor while reducing t by the same factor, i.e., $e^{\mathcal{L}t}[\rho] = e^{(c\mathcal{L})\frac{t}{c}}[\rho]$ for any $c > 0$, where $c\mathcal{L}$ denotes the linear map obtained from $\mathcal{L}$ with H multiplied by c and each L_j multiplied by $\sqrt{c}$. This reduces the simulation time but transfers the cost into the magnitude of $\mathcal{L}$. To normalize this cost, we define a norm based on the specification of $\mathcal{L}$.

Define the norm of a specification of a Lindbladian $\mathcal{L}$ as a linear product of Paulis as

$$\|\mathcal{L}\|_{\mathsf{pauli}} = \sum_{k=0}^{q-1} \beta_{0k} + \sum_{j=1}^{m} \Big(\sum_{k=0}^{q-1} \beta_{jk} \Big)^2. \tag{5.3}$$

For simplicity we use the terminology $\|\mathcal{L}\|_{\mathsf{pauli}}$ even though the quantity is not directly a function of the linear map $\mathcal{L}$. However, $\|c\mathcal{L}\|_{\mathsf{pauli}} = c\|\mathcal{L}\|_{\mathsf{pauli}}$ if $c\mathcal{L}$ denotes the expression in Eq. (4.4) with the factor c multiplied through.

Our main result is the following theorem.

Theorem 5.1. *Let $\mathcal{L}$ be a Lindbladian presented as a linear combination of q Paulis. Then, for any $t > 0$ and $\epsilon > 0$, there exists a quantum circuit of size*

$$O\left(m^2 q^2 \tau \frac{(\log(mq\tau/\epsilon) + n)\log(\tau/\epsilon)}{\log\log(\tau/\epsilon)} \right) \tag{5.4}$$

that implements a quantum channel $\mathcal{N}$, such that $\left\| \mathcal{N} - e^{\mathcal{L}t} \right\|_\diamond \leq \epsilon$, where $\tau = t\,\|\mathcal{L}\|_{\mathsf{pauli}}$ and m is the number of jump operators in $\mathcal{L}$.

Remarks:

1. The proof of Theorem 5.1 is in Section 5.4. A main novel ingredient of the proof is Lemma 5.3, concerning a variant of the "linear combination of unitaries" construction that is suitable for quantum channels (explained in Sections 5.2 and 5.3).

2. The factor $\|\mathcal{L}\|_{\mathsf{pauli}}$ corresponding to the coefficients of the specification as a linear combination of Paulis is a natural generalization to the case of Lindbladians of a similar factor for Hamiltonians that appears in [BCC+15].

3. When $m, q \in \mathrm{poly}(n)$, the gate complexity in Theorem 5.1 simplifies to

$$O\left(\tau \frac{\log(\tau/\epsilon)^2}{\log\log(\tau/\epsilon)} \mathrm{poly}(n)\right). \tag{5.5}$$

4. A Lindbladian $\mathcal{L}$ is *local* if

$$H = \sum_{j=1}^{m'} H_j, \tag{5.6}$$

where $H_1, \ldots, H_{m'}$ and also $L_1, \ldots, L_m$ are local (i.e., they each act on a constant number of qubits). A *local specification* of $\mathcal{L}$ is as $H_1, \ldots, H_{m'}, L_1, \ldots, L_m$ and we define its norm as

$$\|\mathcal{L}\|_{\mathsf{local}} = \sum_{j=1}^{m'} \|H_j\| + \sum_{j=1}^{m} \|L_j\|^2. \tag{5.7}$$

For local Lindbladians, Theorem 5.1 reduces to the following.

Corollary 5.2. *If $\mathcal{L}$ is a local Lindbladian then the gate complexity for simulating $e^{\mathcal{L}t}$ with precision ϵ is*

$$O\left((m+m')^2 \tau \frac{\log((m+m')\tau/\epsilon)\log(\tau/\epsilon)}{\log\log(\tau/\epsilon)}\right), \tag{5.8}$$

where $\tau = t\,\|\mathcal{L}\|_{\mathsf{local}}$.

5. We also consider *sparse* Lindbladians (see [CL17] for various definitions, extending definitions and specifications of sparse Hamiltonians [ATS03]). Here, we define a Lindbladian to have d-sparse operators if $H, L_1, \ldots, L_m$ each have at most d nonzero entries in each row/column. A *sparse specification* of such a Lindbladian $\mathcal{L}$ is as a black-box that provides the positions and values of the nonzero entries of each row/column of $H, L_1, \ldots, L_m$ via queries.

Define the norm of any specification of a Lindbladian in terms of the norm of operators $H, L_1, \ldots, L_m$ as

$$\|\mathcal{L}\|_{\mathsf{ops}} = \|H\| + \sum_{j=1}^{m} \|L_j\|^2. \tag{5.9}$$

The query complexity and gate complexity for simulating d-sparse Lindbladians $\mathcal{L}$ are

$$O\left(\tau \,\mathrm{polylog}(mq\tau/\epsilon)\mathrm{poly}(d,n)\right), \tag{5.10}$$

where $\tau = t\|\mathcal{L}\|_{\mathsf{ops}}$. We sketch the analysis in Section 5.5.

6. We expect some of the methodologies in [BCC+15, BCK15, LC17, PP17] to be adaptable to the Lindblad evolution simulation problem (in conjunction with our variant of the LCU construction and oblivious amplitude amplification), but have not investigated this.

5.2 Novel techniques

As noted in Section 5.1, for the Hamiltonian simulation problem, several recent methods have achieved the scaling that is $O(t\,\mathrm{polylog}(1/\epsilon)\,\mathrm{poly}(n))$ which improve on what has been accomplished using the longstanding Trotter-Suzuki decomposition [Suz91]. One of the main tools of these algorithms is a remarkable circuit construction to implement certain unitary operators (or near-unitary operators) that can be decomposed into a linear combination of unitaries. We refer to this construction as the *standard LCU method* [BCC+15, Kot14]. For a unitary V that is a linear combination of unitaries as $V = \alpha_0 U_0 + \cdots + \alpha_{m-1} U_{m-1}$, the standard LCU method is a circuit construction W (consisting of the implementations of $U_0, \ldots, U_{m-1}$) that performs the following mapping:

$$W|0\rangle|\psi\rangle = \sqrt{p}V|\psi\rangle + \sqrt{1-p}|\Phi^{\perp}\rangle, \tag{5.11}$$

where $|\Phi^{\perp}\rangle$ is some state such that $(|0\rangle\langle 0| \otimes I)|\Phi^{\perp}\rangle = 0$ and

$$p = \frac{1}{\left(\sum_{j=0}^{m-1} \alpha_j\right)^2} \tag{5.12}$$

is the success probability (that arises if the first register is measured).

Another technique used in previous Hamiltonian simulation algorithms is *oblivious amplitude amplification for unitaries* (introduced in [BCC+15] and originally inspired by [MW05]), which is a generalization of *amplitude amplification for unitaries* [BH97, BHMT02]. Suppose a unitary V produces the desired state $|\psi\rangle$ with probability p in the following sense

$$V|0\rangle = \sqrt{p}|\psi\rangle + \sqrt{1-p}|\Phi^{\perp}\rangle, \tag{5.13}$$

where $|\Phi^{\perp}\rangle$ is some state that is orthogonal to $|\psi\rangle$. Amplitude amplification for unitaries can be viewed as a generalization of Grover's algorithm [Gro96] in the sense that when the reflection about $|\psi\rangle$ (i.e., $2|\psi\rangle\langle\psi| - I$) and the reflection about the input state (i.e., $2|0\rangle\langle 0| - I$) are available, only $O(\sqrt{1/p})$ applications of V is sufficient to obtain the desired state $|\psi\rangle$ with almost certainty. In a more general case, the reflection about the input state is unavailable. Then the technique of oblivious amplitude amplification for unitaries can be applied to boost the success probability with the same performance as amplitude amplification for unitaries.

In the remainder of this section, we first demonstrate that the standard LCU method performs poorly on Stinespring dilations. Then, we briefly summarize the two novel techniques, namely, LCU for channels, and oblivious amplitude amplification for isometries.

5.2.1 The performance of the standard LCU method on Stinespring dilations

For the case of Lindblad evolution, the operations that arise are quantum channels that are not generally unitary. One method to implement a quantum channel is to consider a larger Hilbert space by Stinespring dilations and use the standard LCU method to implement the evolution in the larger Hilbert space. In this subsection, we show in some technical detail why the standard LCU method performs poorly for Stinespring dilations of quantum channels.

Let us consider the amplitude damping channel defined by Eq. (4.33), whose two Kraus operators have the following LCU decompositions:

$$E_0 = \begin{pmatrix} 1 & 0 \\ 0 & \sqrt{1-\delta} \end{pmatrix} = \alpha_{00} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \alpha_{01} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{5.14}$$

$$E_1 = \begin{pmatrix} 0 & \sqrt{\delta} \\ 0 & 0 \end{pmatrix} = \alpha_{10} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} + \alpha_{11} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \tag{5.15}$$

where $\alpha_{00} = \frac{1+\sqrt{1-\delta}}{2}, \alpha_{01} = \frac{1-\sqrt{1-\delta}}{2}$, and $\alpha_{10} = \alpha_{11} = \frac{\sqrt{\delta}}{2}$. Evolving the amplitude damping process defined by Eq. (4.34) for time t yields this quantum channel with $\delta = 1 - e^{-t}$. When $t \ll 1$, we have $\delta \approx t, \alpha_{00} \approx 1 - t/4$, and $\alpha_{10} \approx t/4$.

A Stinespring dilation of this quantum channel (denoted by V) and its LCU decomposition can be derived from the above LCU decompositions of E_0 and E_1 as

$$V = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \sqrt{1-\delta} & -\sqrt{\delta} & 0 \\ 0 & \sqrt{\delta} & \sqrt{1-\delta} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \alpha_{00} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} + \alpha_{01} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{5.16}$$

$$+ \alpha_{10} \begin{pmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} + \alpha_{11} \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{pmatrix}. \tag{5.17}$$

Applying the standard LCU method and using Eq. (5.12) , the success probability is

$$\frac{1}{(\alpha_{00} + \alpha_{01} + \alpha_{10} + \alpha_{11})^2} = \frac{1}{(1+\sqrt{\delta})^2} = 1 - 2\sqrt{\delta} + \Theta(\delta). \tag{5.18}$$

It implies that for small time evolution t, the failure probability is $\Theta(\sqrt{t})$, which is prohibitively expensive: the process can be repeated at most $\Theta(1/\sqrt{t})$ times until the cumulative failure probability becomes a constant. This process corresponds to the amplitude damping process of total evolution time

$$\Theta\left(\frac{1}{\sqrt{t}}\right) \cdot t = \Theta(\sqrt{t}), \tag{5.19}$$

which is subconstant as $t \to 0$. This causes a problem in the general Lindblad simulation.

5.2.2 Brief summary of novel techniques

Some quantum channels are *mixed unitary channels*, which means they can be expressed as a probability distribution of unitary operations (e.g., with probability $p_0, \ldots, p_{m-1}$ on unitaries $U_0, \ldots, U_{m-1}$). To simulate such channels, one can first randomly sample $j \in \{0, \ldots, m-1\}$ and then apply the standard LCU method to each unitary U_j. However, there are quantum channels that are *not* mixed unitary – and such channels can arise from the Lindblad evolution, such as the amplitude damping process we examined in Section 4.2.

As demonstrated in Subsection 5.2.1, a simple reductionist approach, namely, expressing these quantum channels in the Stinespring form and then applying the standard LCU method in the larger Hilbert space, performs poorly. Instead, we take a different approach that does not involve a reduction to the unitary case: we develop a new variant of the LCU method for channels. Using this new LCU method, to implement the amplitude damping process with the Kraus operators specified in Eqns. (5.14) and (5.15), a higher success probability

$$\frac{1}{(\alpha_{00}+\alpha_{01})^2+(\alpha_{10}+\alpha_{11})^2} = \frac{1}{1+\delta} = 1-\delta+\Theta(\delta^2) \tag{5.20}$$

can be achieved compared to the success probability in Eq. (5.18). For small time evolution t, the failure probability is $\Theta(t)$. Now, the process can be repeated $\Theta(1/t)$ times until the cumulative failure probability becomes a constant. This implementation corresponds to the amplitude damping process of total evolution time

$$\Theta\left(\frac{1}{t}\right)\cdot t = \Theta(1), \tag{5.21}$$

which is constant as $t \to 0$. This is consistent with the algorithms of simulating Hamiltonian evolution in [BCC+15, BCC+17]. Therefore the methodologies used therein can be used with various adjustments to obtain the similar simulation bounds.

Another new technique that we employ is an oblivious amplitude amplification algorithm for isometries (as opposed to unitaries), which is noteworthy because a reductionist approach based on extending isometries to unitaries does not work. Intuitively, this is because the new LCU construction for channels turns out to produce an isometry that corresponds to a purification of the quantum channel, and it does not produce a unitary extension of that isometry.

5.3 New LCU method for channels and completely positive maps

Let $A_0, \ldots, A_{m-1} \in \mathrm{L}(\mathbb{C}^{2^n})$ be the Kraus operators of a quantum channel. Suppose that, for each $j \in \{0, \ldots, m-1\}$, we have a decomposition of A_j as a linear combination of unitaries in the form

$$A_j = \sum_{k=0}^{q-1} \alpha_{jk} U_{jk}, \tag{5.22}$$

where, for each $j \in \{0, \ldots, m-1\}$ and $k \in \{0, \ldots, q-1\}$, $\alpha_{jk} \geq 0$ and U_{jk} is unitary.

The objective is to implement the quantum channel in terms of the implementations of U_{jk}'s. We will describe a circuit W and a fixed state $|\mu\rangle$ such that, for any n-qubit state $|\psi\rangle$,

$$W|0\rangle|\mu\rangle|\psi\rangle = \sqrt{p}|0\rangle \left(\sum_{j=0}^{m-1} |j\rangle A_j|\psi\rangle \right) + \sqrt{1-p}|\Phi^{\perp}\rangle, \tag{5.23}$$

where $(|0\rangle\langle 0| \otimes I \otimes I)|\Phi^{\perp}\rangle = 0$ and

$$p = \frac{1}{\sum_{j=0}^{m-1} (\sum_{k=0}^{q-1} \alpha_{jk})^2} \tag{5.24}$$

is called the *success probability parameter* (which is realized if the first register is measured). Note that the isometry $|\psi\rangle \mapsto \sum_{j=0}^{m-1} |j\rangle A_j|\psi\rangle$ is the quantum channel in purified form.

The circuit W is in terms of two gates. One gate is a *multiplexed-U gate*, denoted by multi-U such that, for all $j \in \{0, \ldots, m-1\}$ and $k \in \{0, \ldots, q-1\}$,

$$\text{multi-}U|k\rangle|j\rangle|\psi\rangle = |k\rangle|j\rangle U_{jk}|\psi\rangle. \tag{5.25}$$

The other gate is a *multiplexed-B gate*, denoted by multi-B. It performs the following mapping for all $j \in \{0, \ldots, m-1\}$:

$$\text{multi-}B|0\rangle|j\rangle = \left(\frac{1}{\sqrt{s_j}} \sum_{k=0}^{q-1} \sqrt{\alpha_{jk}}|k\rangle \right) |j\rangle, \tag{5.26}$$

where

$$s_j = \sum_{k=0}^{q-1} \alpha_{jk}. \tag{5.27}$$

Define the state $|\mu\rangle$ (in terms of $s_0, \ldots, s_{m-1}$ from Eq. (5.27)) as

$$|\mu\rangle = \frac{1}{\sqrt{\sum_{j=1}^{m-1} s_j^2}} \sum_{j=0}^{m-1} s_j|j\rangle. \tag{5.28}$$

Define the unitary $W \in \mathrm{U}(\mathbb{C}^q \otimes \mathbb{C}^m \otimes \mathbb{C}^{2^n})$ as

$$W = (\text{multi-}B^{\dagger} \otimes I)\text{multi-}U(\text{multi-}B \otimes I). \tag{5.29}$$

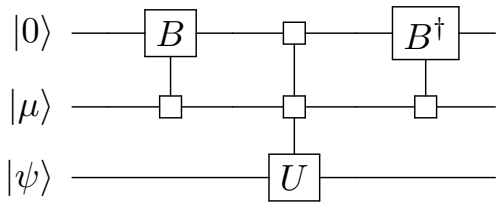

Figure 5.1: The circuit W for simulating a quantum channel using the new LCU method.

The LCU construction with the circuit of W with its initial state $|0\rangle\otimes|\mu\rangle\otimes|\psi\rangle$ is illustrated in Figure 5.1. In this figure, we refer to the first register as the *indicator register* (as it indicates whether the computation succeeds or not at the end of this operation), the second register as the *purifier register* (as it is used to purify the quantum channel when the computation succeeds), and the third register as the *system register* (as it contains the state being evolved).

In the following lemma, Eq. (5.23) is shown to apply where $A_0,\ldots,A_{m-1}\in \mathrm{L}(\mathbb{C}^{2^n})$ are arbitrary linear operators (i.e., Kraus operators of a completely positive map that is not necessarily trace-preserving). If the map is also trace-preserving, then it holds that $\sum_{j=0}^{m-1}|j\rangle A_j|\psi\rangle$ and $|\Phi^\perp\rangle$ are normalized states, and the success probability parameter p is the actual success probability realized if the first register is measured; otherwise, these need not be the case. In subsequent subsections, we will apply this lemma in a context where the trace-preserving condition is *approximately* satisfied.

Lemma 5.3. *Let $A_0,\ldots,A_{m-1}$ be the Kraus operators of a completely positive map. Suppose that each A_j can be written in the form of Eq. (5.22). Let multi-U, multi-B, W, and $|\mu\rangle$ be defined as above. Then applying the unitary W on any state of the form $|0\rangle|\mu\rangle|\psi\rangle$ produces the state*

$$\sqrt{p}|0\rangle\left(\sum_{j=0}^{m-1}|j\rangle A_j|\psi\rangle\right)+\sqrt{1-p}|\Phi^\perp\rangle,$$

where $(|0\rangle\langle 0|\otimes I\otimes I)|\Phi^\perp\rangle=0$, and

$$p=\frac{1}{\sum_{j=0}^{m-1}\left(\sum_{k=0}^{q-1}\alpha_{jk}\right)^2}\,.$$

Proof. First consider the state $|0\rangle|j\rangle|\psi\rangle$ for any $j\in\{0,\ldots,m-1\}$. Applying W on this

state is corresponding to the standard LCU method [Kot14]:

$$W|0\rangle|j\rangle|\psi\rangle = (\text{multi-}B^\dagger \otimes I)\text{multi-}U(\text{multi-}B \otimes I)|0\rangle|j\rangle|\psi\rangle \tag{5.30}$$

$$= \frac{1}{\sqrt{s_j}}(\text{multi-}B^\dagger \otimes I)\text{multi-}U\left(\sum_{k=0}^{q-1}\sqrt{\alpha_{jk}}|k\rangle\right)|j\rangle|\psi\rangle \tag{5.31}$$

$$= \frac{1}{\sqrt{s_j}}(\text{multi-}B^\dagger \otimes I)\left(\sum_{k=0}^{q-1}\sqrt{\alpha_{jk}}|k\rangle|j\rangle U_{jk}|\psi\rangle\right) \tag{5.32}$$

$$= \frac{1}{s_j}|0\rangle|j\rangle\left(\sum_{k=0}^{q-1}\alpha_{jk}U_{jk}|\psi\rangle\right) + \sqrt{\gamma_j}|\Phi_j^\perp\rangle \tag{5.33}$$

$$= \frac{1}{s_j}|0\rangle|j\rangle A_j|\psi\rangle + \sqrt{\gamma_j}|\Phi_j^\perp\rangle, \tag{5.34}$$

where $|\Phi_j^\perp\rangle$ is a state satisfying $(|0\rangle\langle 0|\otimes I\otimes I)|\Phi_j^\perp\rangle = 0$ and γ_j is some normalization factor.

Up to this point, if the indicator register were measured and $|0\rangle$ were observed as the "success" case as in the standard LCU method, then the state of the purifier and the system register collapses to $|j\rangle A_j|\psi\rangle$. However, this is not a meaningful quantum state, as it only captures one Kraus operator of a linear map. Now we use this specially designed quantum state $|\mu\rangle$ to obtain the desired purification state. We use the superposition $|\mu\rangle$ instead of $|j\rangle$ in the purifier register then, by linearity, we have

$$W|0\rangle|\mu\rangle|\psi\rangle = \sqrt{p}|0\rangle\left(\sum_{j=0}^{m-1}|j\rangle A_j|\psi\rangle\right) + \sqrt{1-p}|\Phi^\perp\rangle, \tag{5.35}$$

where $(|0\rangle\langle 0|\otimes I\otimes I)|\Phi^\perp\rangle = 0$ and $p = \frac{1}{\sum_{j=0}^{m-1}s_j^2}$. □

5.4 Overview of the algorithm

In this section we show how to apply our new LCU method in order to prove the main result, Theorem 5.1.

We first show that, for Lindbladians specified by Eqns. (4.4), (5.1) and (5.2), the quantities $\|\mathcal{L}\|_{\mathsf{pauli}}$ (defined in Eq. (5.3)) and $\|\mathcal{L}\|_{\mathsf{ops}}$ (defined in Eq. (5.9)) satisfy

$$\|\mathcal{L}\|_\diamond \leq 2\|\mathcal{L}\|_{\mathsf{ops}} \leq 2\|\mathcal{L}\|_{\mathsf{pauli}}. \tag{5.36}$$

For the first inequality in Eq. (5.36), note that $\|\mathcal{L}\|_1 \leq 2\|\mathcal{L}\|_{\mathsf{ops}}$ holds by the triangle inequality and the fact that, for any $M \in \mathrm{L}(\mathbb{C}^{2^n})$ such that $\|M\|_1 = 1$,

$$\|[H, M]\|_1 \leq 2\|H\|, \tag{5.37}$$

$$\left\|L_j M L_j^\dagger\right\|_1 \leq \left\|L_j\right\|\left\|M\right\|_1\left\|L_j^\dagger\right\| = \left\|L_j\right\|^2. \tag{5.38}$$

Then, since $\|M \otimes I_{\mathbb{C}^{2^n}}\| = \|M\|$ for any $M \in \mathrm{L}(\mathbb{C}^{2^n})$, the first inequality in Eq. (5.36) follows. The second inequality in Eq. (5.36) follows from the fact that, if H and $L_0, \ldots, L_{m-1}$ are specified as in Eqns. (5.1) and (5.2), then

$$\|H\| \leq \sum_{k=0}^{q-1} \beta_{0k} \quad \text{and,} \quad \|L_j\| \leq \sum_{k=0}^{q-1} \beta_{jk}, \quad \text{for all } j \in \{1, \ldots, m\}. \tag{5.39}$$

Now, we are ready to present the details of our construction for the proof of Theorem 5.1. The overall structure is similar to that in [BCC+15] and [BCC+17], with the main novel ingredient being our variant of the LCU construction (explained in Section 5.3) and also a variant of oblivious amplitude amplification for isometries. For clarity, the details are organized into the following subsections, whose content is summarized as:

1. In subsection 5.4.1, we describe a simple linear map $\mathcal{M}_\delta$ in terms of Kraus operators that are based on the operators in $\mathcal{L}$. For small δ, $\mathcal{M}_\delta$ is a good approximation of $e^{\mathcal{L}\delta}$.

2. In subsection 5.4.2, we show how to simulate the linear map $\mathcal{M}_\delta$ in the sense of Lemma 5.3, with success probability parameter $1 - O(\delta)$.

3. In subsection 5.4.3, we show how to combine r simulations of $\mathcal{M}_{O(1/r)}$ so as to obtain cumulative success probability parameter $1/4$. Conditioned on success, this produces a good approximation of constant-time Lindblad evolution.

4. In subsection 5.4.4, we show how to apply a modified version of oblivious amplitude amplification to unconditionally simulate an approximation of constant-time Lindblad evolution.

5. In subsection 5.4.5, we show how to reduce the number of multiplexed Pauli gates by a concentration bound on the amplitudes associated with nontrivial Pauli gates.

6. In subsection 5.4.6, we bound the total number of gates and combine the simulations for constant-time evolution to simulate arbitrary time evolution, which completes the proof of Theorem 5.1.

5.4.1 A linear map that approximates infinitesimal Lindblad evolution

In this subsection, we show how to approximate Lindblad evolution for small time δ, namely $e^{\mathcal{L}\delta}$, by a linear map $\mathcal{M}_\delta$ that can be described in terms of $m+1$ Kraus operators, where the precision of the approximation is $O(\delta^2)$.

We first bound the error of an approximation of $e^{\mathcal{L}\delta}$ for small δ. In particular, we prove the following lemma.

Lemma 5.4. *Let $\mathcal{L} \in \mathrm{T}(\mathbb{C}^N)$ be a Lindbladian. Then, the following holds:*

$$\left\|(\mathcal{I}_{\mathrm{L}(\mathbb{C}^N)} + \delta\mathcal{L}) - e^{\mathcal{L}\delta}\right\|_\diamond \leq (\delta \left\|\mathcal{L}\right\|_\diamond)^2. \tag{5.40}$$

Proof. Assume that $0 \leq \delta \left\|\mathcal{L}\right\|_\diamond \leq 1$. Then, for any $M \in \mathrm{L}(\mathbb{C}^N)$ such that $\left\|M\right\|_1 \leq 1$, we have

$$\begin{aligned}
\left\|(e^{\delta\mathcal{L}} - (\mathcal{I}_{\mathrm{L}(\mathbb{C}^N)} + \delta\mathcal{L}))[M]\right\|_1 &= \left\|\sum_{s=2}^{\infty} \frac{\delta^s}{s!}\mathcal{L}^s[M]\right\|_1 && (5.41)\\
&\leq \sum_{s=2}^{\infty} \frac{\delta^s}{s!} \left\|\mathcal{L}^s[M]\right\|_1 && (5.42)\\
&\leq \sum_{s=2}^{\infty} \frac{\delta^s}{s!} (\left\|\mathcal{L}[M]\right\|_1)^s && (5.43)\\
&\leq (\delta \left\|\mathcal{L}[M]\right\|_1)^2 && (5.44)\\
&\leq (\delta \left\|\mathcal{L}\right\|_1)^2, && (5.45)
\end{aligned}$$

where we are using the fact that $e^z - (1+z) \leq z^2$ when $0 \leq z \leq 1$.

To extend this from the induced trace norm to the diamond norm, we consider an arbitrary Hilbert space $\mathbb{C}^{N'}$ with $N' \geq N$. Note that

$$\begin{aligned}
&(e^{\delta\mathcal{L}} - (\mathcal{I}_{\mathrm{L}(\mathbb{C}^N)} + \delta\mathcal{L})) \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^{N'})} \\
&\qquad = \exp\left(\delta\left(\mathcal{L} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^{N'})}\right)\right) - \left(\mathcal{I}_{\mathrm{L}(\mathbb{C}^N \otimes \mathbb{C}^{N'})} + \delta\left(\mathcal{L} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^{N'})}\right)\right). \qquad (5.46)
\end{aligned}$$

Also, $\mathcal{L} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^{N'})}$ is a Lindbladian with $\left\|\mathcal{L} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^{N'})}\right\|_1 = \left\|\mathcal{L}\right\|_\diamond$ when $N' \geq N$. Therefore,

we have

$$\begin{aligned}
&\left\|e^{\delta\mathcal{L}} - (\mathcal{I}_{\mathrm{L}(\mathbb{C}^N)} + \delta\mathcal{L})\right\|_\diamond \\
&\qquad = \left\|(e^{\delta\mathcal{L}} - (\mathcal{I}_{\mathrm{L}(\mathbb{C}^N)} + \delta(\mathcal{L})) \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^{N'})}\right\|_1 && (5.47)\\
&\qquad = \left\|\exp\left(\delta\left(\mathcal{L} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^{N'})}\right)\right) - \left(\mathcal{I}_{\mathrm{L}(\mathbb{C}^N \otimes \mathbb{C}^{N'})} + \delta\left(\mathcal{L} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^{N'})}\right)\right)\right\|_1 && (5.48)\\
&\qquad \leq \left(\delta\left\|\mathcal{L} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^{N'})}\right\|_1\right)^2 && (5.49)\\
&\qquad = \left(\delta\left\|\mathcal{L}\right\|_\diamond\right)^2. && (5.50)
\end{aligned}$$

□

Following the approach described in Section 4.1, define the linear map $\mathcal{M}_\delta$ as

$$\mathcal{M}_\delta[Q] = \sum_{j=0}^{m} A_j Q A_j^\dagger, \tag{5.51}$$

where

$$A_0 = I - \frac{\delta}{2}\sum_{j=1}^{m} L_j^\dagger L_j - i\delta H \quad \text{and, for } j \in \{1,\ldots,m\}, \quad A_j = \sqrt{\delta} L_j. \tag{5.52}$$

Note that, in general, $\mathcal{M}_\delta$ does not satisfy the trace-preserving condition for a quantum channel; however, it satisfies an approximate version of it:

$$\begin{aligned}
\left\|\sum_{j=0}^{m} A_j^\dagger A_j - I\right\| &= \left\|\frac{\delta^2}{4}\left(\sum_{j=1}^{m} L_j^\dagger L_j\right)^2 + \delta^2 H^2\right\| && (5.53)\\
&\leq \frac{\delta^2}{4}\left\|\left(\sum_{j=1}^{m} L_j^\dagger L_j\right)^2\right\| + \delta^2\left\|H^2\right\| && (5.54)\\
&\leq \delta^2\left\|\sum_{j=1}^{m} L_j^\dagger L_j\right\|^2 + \delta^2\left\|H\right\|^2 && (5.55)\\
&\leq \delta^2\left(\sum_{j=1}^{m}\left\|L_j^\dagger L_j\right\| + \left\|H\right\|\right)^2 && (5.56)\\
&= \left(\delta\left\|\mathcal{L}\right\|_{\mathsf{ops}}\right)^2. && (5.57)
\end{aligned}$$

Now, we show that

$$\|\mathcal{M}_\delta - e^{\mathcal{L}\delta}\|_\diamond \leq 5(\delta \|\mathcal{L}\|_{\mathsf{ops}})^2. \tag{5.58}$$

To do this, we introduce an intermediate linear map, $\mathcal{I} + \delta\mathcal{L}$ (mapping ρ to $\rho + \delta\mathcal{L}[\rho]$), and show that

$$\|\mathcal{M}_\delta - (\mathcal{I} + \delta\mathcal{L})\|_\diamond \leq (\delta \|\mathcal{L}\|_{\mathsf{ops}})^2 \tag{5.59}$$

and then Eq. (5.58) follows from the fact that

$$\begin{aligned} \left\|(\mathcal{I} + \delta\mathcal{L}) - e^{\mathcal{L}\delta}\right\|_\diamond &\leq (\delta \|\mathcal{L}\|_\diamond)^2 && (5.60)\\ &\leq (2\delta \|\mathcal{L}\|_{\mathsf{ops}})^2, && (5.61) \end{aligned}$$

where the first inequality follows from Lemma 5.4. In order to show Eq. (5.59), note that for any operator $Q \in \mathrm{L}(\mathbb{C}^N \otimes \mathbb{C}^{N'})$ with $\|Q\|_1 = 1$ and $N' \geq N$, we have

$$\begin{aligned} &\left\|(\mathcal{M}_\delta \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^{N'})} - (\mathcal{I}_{\mathrm{L}(\mathbb{C}^N)} + \delta\mathcal{L}) \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^{N'})})[Q]\right\|_1 \\ &\qquad = \left\|\sum_{j=0}^m (A_j \otimes I)Q(A_j \otimes I)^\dagger - (Q + \delta(\mathcal{L} \otimes \mathcal{I}_{\mathrm{L}(\mathbb{C}^{N'})})[Q])\right\|_1 && (5.62)\\ &\qquad = \left\|\frac{\delta^2}{4}\Big(\sum_{j=1}^m L_j^\dagger L_j \otimes I\Big)Q\Big(\sum_{j'=1}^m L_{j'}^\dagger L_{j'} \otimes I\Big) - \frac{\delta^2}{2} i \sum_{j=1}^m (L_j^\dagger L_j \otimes I)Q(H \otimes I)\right. && (5.63)\\ &\qquad\qquad \left. + \frac{\delta^2}{2} i (H \otimes I)Q \sum_{j=1}^m (L_j^\dagger L_j \otimes I) + \delta^2 (H \otimes I)Q(H \otimes I)\right\|_1 && (5.64)\\ &\qquad \leq \delta^2 \left(\left\|\sum_{j=1}^m L_j^\dagger L_j \otimes I\right\|^2 + 2\left\|H \otimes I\right\| \left\|\sum_{j=1}^m L_j^\dagger L_j \otimes I\right\| + \left\|H \otimes I\right\|^2 \right) && (5.65)\\ &\qquad \leq \delta^2 \left(\left\|\sum_{j=1}^m L_j^\dagger L_j \otimes I\right\| + \left\|H \otimes I\right\| \right)^2 && (5.66)\\ &\qquad \leq \delta^2 \left(\sum_{j=1}^m \|L_j\|^2 + \|H\| \right)^2 && (5.67)\\ &\qquad \leq \delta^2 \|\mathcal{L}\|_{\mathsf{ops}}^2 . && (5.68) \end{aligned}$$

This completes the proof of Eq. (5.58).

5.4.2 Implementing the approximation map by the new LCU method

Here we show how to construct a quantum circuit that computes an approximation of $\mathcal{M}_\delta$ along the lines of Eq. (5.23) using the new LCU method.

By substituting Eqns. (5.1) and (5.2) into Eq. (5.52), we have

$$A_j = \sqrt{\delta} \sum_{k=0}^{q-1} \beta_{jk} V_{jk}, \quad \text{for } j \in \{1, \ldots, m\}, \text{ and} \tag{5.69}$$

$$A_0 = I - \frac{\delta}{2} \sum_{j=1}^{m} \left(\sum_{k=0}^{q-1} \beta_{jk} V_{jk} \right)^\dagger \left(\sum_{l=0}^{q-1} \beta_{jl} V_{jl} \right) - i\delta \sum_{k=0}^{q-1} \beta_{0k} V_{0k} \tag{5.70}$$

$$= I + \frac{\delta}{2} \sum_{j=1}^{m} \sum_{k=0}^{q-1} \sum_{l=0}^{q-1} \beta_{jk} \beta_{jl} \left(-V_{jk}^\dagger V_{jl} \right) + \delta \sum_{k=0}^{q-1} \beta_{0k} \left(-i V_{0k} \right). \tag{5.71}$$

Note that Eqns. (5.69) and (5.71) are expressing the Kraus operators $A_0, \ldots, A_m$ as tensor products of Paulis (i.e., they are of the form of Eqns. (5.1) and (5.2)). Therefore, by Lemma 5.3, the circuit construction of W in Figure 5.1 and the state $|\mu\rangle$ from Eq. (5.28) satisfy the following property. For any state $|\psi\rangle$,

$$W|0\rangle|\mu\rangle|\psi\rangle = \sqrt{p}|0\rangle \left(\sum_{j=0}^{m} |j\rangle A_j |\psi\rangle \right) + \sqrt{1-p}|\Phi^\perp\rangle, \tag{5.72}$$

where $|\Phi^\perp\rangle$ satisfies $(|0\rangle\langle 0| \otimes I \otimes I)|\Phi^\perp\rangle = 0$ and

$$p = \frac{1}{\sum_{j=0}^{m} s_j^2}, \tag{5.73}$$

where

$$s_j = \sqrt{\delta} \sum_{k=0}^{q-1} \beta_{jk}, \quad \text{for } j \in \{1, \ldots, m\}, \quad \text{and} \tag{5.74}$$

$$s_0 = 1 + \frac{\delta}{2} \sum_{j=1}^{m} \sum_{k=0}^{q-1} \sum_{l=0}^{q-1} \beta_{jk} \beta_{jl} + \delta \sum_{k=0}^{q-1} \beta_{0k}. \tag{5.75}$$

(The values of $s_0, \ldots, s_m$ are directly from Eqns. (5.69) and (5.71).)

To simplify the expression for the success probability parameter, define

$$c_j = \sum_{k=0}^{q-1} \beta_{jk}, \tag{5.76}$$

for $j \in \{0, \ldots, m\}$. Then we can rewrite Eqns. (5.74) and (5.75) as

$$s_j = \sqrt{\delta} c_j, \quad \text{for } j \in \{1, \ldots, m\}, \quad \text{and} \tag{5.77}$$

$$s_0 = 1 + \frac{\delta}{2} \sum_{j=1}^{m} c_j^2 + \delta c_0 \tag{5.78}$$

and

$$p = \frac{1}{\sum_{j=0}^{m} s_j^2} \tag{5.79}$$

$$= \frac{1}{\left(1 + \frac{\delta}{2} \sum_{j=1}^{m} c_j^2 + \delta c_0\right)^2 + \sum_{j=1}^{m} c_j^2 \delta} \tag{5.80}$$

$$= \frac{1}{1 + 2\delta \sum_{j=1}^{m} c_j^2 + 2\delta c_0 + \delta^2 \left(\frac{1}{2} \sum_{j=1}^{m} c_j^2 + c_0\right)^2} \tag{5.81}$$

$$= \frac{1}{1 + 2\delta\left(\sum_{j=1}^{m} c_j^2 + c_0\right) + \frac{\delta^2}{4}\left(\sum_{j=1}^{m} c_j^2 + 2c_0\right)^2} \tag{5.82}$$

$$= \frac{1}{1 + 2\delta \left\|\mathcal{L}\right\|_{\mathsf{pauli}} + \frac{\delta^2}{4}\left(\left\|\mathcal{L}\right\|_{\mathsf{pauli}} + c_0\right)^2} \tag{5.83}$$

$$= 1 - 2\delta \left\|\mathcal{L}\right\|_{\mathsf{pauli}} - O\left(\delta^2 \left\|\mathcal{L}\right\|_{\mathsf{pauli}}^2\right). \tag{5.84}$$

Note that, since $\mathcal{M}_\delta$ is only an approximate channel, the success probability parameter p does not correspond to the actual probability of outcome 0 if the indicator register is measured; however, it can be shown that p is within $O(\delta^2)$ of the actual success probability. We do not show this here; our analysis will be in terms of the cumulative error arising in circuit constructions in the subsequent subsections (which consist of several instances of the construction from this subsection).

5.4.3 Simulation with constant success probability

In this subsection we iterate the construction from the previous subsection r times, with $\delta = O(1/r)$.

The resulting success probability parameter associated with $\mathcal{M}_\delta^r$ is $p^r = (1 - O(1/r))^r$, which converges to a constant. We can tune the parameter δ so that $p^r = 1/4$ holds exactly. This is accomplished by setting $p = 4^{-1/r}$ and then solving for δ in Eq. (5.83), yielding the positive solution

$$\delta = \frac{-\left\|\mathcal{L}\right\|_{\mathsf{pauli}} + \sqrt{\left\|\mathcal{L}\right\|_{\mathsf{pauli}}^2 + \frac{1}{4}\left(\left\|\mathcal{L}\right\|_{\mathsf{pauli}} + c_0\right)^2 (4^{1/r} - 1)}}{\frac{1}{4}\left(\left\|\mathcal{L}\right\|_{\mathsf{pauli}} + c_0\right)^2} \tag{5.85}$$

$$= \left(\frac{\ln(2)}{\left\|\mathcal{L}\right\|_{\mathsf{pauli}}}\right)\frac{1}{r} + O\Big(\frac{1}{r^2}\Big). \tag{5.86}$$

The circuit that implements $\mathcal{M}_\delta^r$ uses an initial state $(|0\rangle|\mu\rangle)^{\otimes r}|\psi\rangle$, which can be reordered to $|0\rangle^{\otimes r}|\mu\rangle^{\otimes r}|\psi\rangle$. It consists of r instances of W, each with *separate* indicator and purifier registers, but with *the same* system register. Let $\widehat{W}$ denote this unitary operator (consisting of r applications of W on different indicator and purifier registers). For each $\hat{\jmath} = j_0 \cdots j_{r-1} \in \{0, \ldots, m\}^r$, define $\widehat{A}_{\hat{\jmath}}$ as

$$\widehat{A}_{\hat{\jmath}} = A_{j_0} \cdots A_{j_{r-1}}. \tag{5.87}$$

We can conclude that

$$\widehat{W}\Big(|0\rangle^{\otimes r}|\mu\rangle^{\otimes r}|\psi\rangle\Big) = \sqrt{p^r}|0\rangle^{\otimes r}\left(\sum_{\hat{\jmath}\in\{0,\ldots,m\}^r} |\hat{\jmath}\rangle\widehat{A}_{\hat{\jmath}}|\psi\rangle\right) + \sqrt{1-p^r}|\widehat{\Phi}^\perp\rangle \tag{5.88}$$

$$= \frac{1}{2}|0\rangle^{\otimes r}\left(\sum_{\hat{\jmath}\in\{0,\ldots,m\}^r} |\hat{\jmath}\rangle\widehat{A}_{\hat{\jmath}}|\psi\rangle\right) + \frac{\sqrt{3}}{2}|\widehat{\Phi}^\perp\rangle, \tag{5.89}$$

where $|\widehat{\Phi}^\perp\rangle$ satisfies $(|0\rangle\langle 0|^{\otimes r} \otimes I^{\otimes r} \otimes I)|\widehat{\Phi}^\perp\rangle = 0$.

Note that this conditionally simulates $\mathcal{M}_\delta^r$, and $\mathcal{M}_\delta^r$ approximates $e^{\mathcal{L}t}$, for

$$t = r\delta = \frac{\ln(2)}{\left\|\mathcal{L}\right\|_{\mathsf{pauli}}} + O\Big(\frac{1}{r}\Big). \tag{5.90}$$

The approximation is in the sense that

$$\left\|\mathcal{M}_\delta^r - e^{\mathcal{L}t}\right\|_\diamond = O\Big(\frac{1}{r}\Big). \tag{5.91}$$

If the desired evolution time is such that $t\left\|\mathcal{L}\right\|_{\mathsf{pauli}} < \ln(2)$ then the success probability parameter resulting from this approach is larger than $1/4$; however, it can be diluted to be exactly $1/4$ using a method described in [BCC+17] that employs an additional qubit as part of the indicator register.

In the next subsection, we show how to use oblivious amplitude amplification to achieve perfect success probability.

5.4.4 Oblivious amplitude amplification for isometries

There are two hurdles for applying oblivious amplitude amplification in our construction. First, the purified quantum state corresponding to the success case is not a normalized quantum state, as the Kraus operators of $\mathcal{M}_\delta^r$ do not satisfy the trace-preserving condition. Second, the operation corresponding to the success case is an isometry (rather than a unitary), because part of the registers in the initial state is restricted to $(|\mu_0\rangle \cdots |\mu_{m-1}\rangle)^{\otimes r}$.

The second hurdle is resolved by using different projectors in the amplitude amplification operator. For the first hurdle, we show that it only causes a small error. To begin with, we examine how far it is for the Kraus operators to satisfy the trace-preserving condition, and this quantity will be used later in the proof. By repeatedly applying Eq. (5.57), we have

$$\left\| \sum_{\hat{\jmath}\in\{0,\ldots,m\}^r} \widehat{A}_{\hat{\jmath}}^\dagger \widehat{A}_{\hat{\jmath}} - I \right\| = \left\| \sum_{j_0\cdots j_{r-1}\in\{0,\ldots,m\}^r} (A_{j_{r-1}}^\dagger \cdots A_{\hat{j}_0}^\dagger)(A_{j_0}\cdots A_{j_{r-1}}) - I \right\| \tag{5.92}$$

$$\leq r\left(\delta\left\|\mathcal{L}\right\|_{\mathsf{pauli}}\right)^2 \tag{5.93}$$

$$= (\ln(2))^2/r + O(1/r^2), \tag{5.94}$$

where the second equality follows from substituting the value of δ from Eq. (5.85).

Before we present the oblivious amplitude amplification construction, we introduce more notations for convenience. For any $|\psi\rangle$, let $|\Psi\rangle$ denote the initial state

$$|\Psi\rangle := |\widehat{0}\rangle|\widehat{\mu}\rangle|\psi\rangle, \tag{5.95}$$

where $|\widehat{0}\rangle = |0\rangle^{\otimes r}$, and $|\widehat{\mu}\rangle = |\mu\rangle^{\otimes r}$. Let $|\Phi\rangle$ denote the desired purification state, i.e.,

$$|\Phi\rangle = |\widehat{0}\rangle \left(\sum_{\hat{\jmath}\in\{0,\ldots,m\}^r} |\hat{\jmath}\rangle \widehat{A}_{\hat{\jmath}} |\psi\rangle \right). \tag{5.96}$$

Let $P_0 := |\widehat{0}\rangle\langle\widehat{0}| \otimes I \otimes I$ and $P_1 := |\widehat{0}\rangle\langle\widehat{0}| \otimes |\widehat{\mu}\rangle\langle\widehat{\mu}| \otimes I$ be two projectors. By Eq. (5.89), we have

$$\widehat{W}|\Psi\rangle = \frac{1}{2}|\Phi\rangle + \frac{\sqrt{3}}{2}|\Phi^{\perp}\rangle, \tag{5.97}$$

for some state $|\Phi^{\perp}\rangle$ satisfying $P_0|\Phi^{\perp}\rangle = 0$. Define the unitary operator

$$F = -\widehat{W}(I - 2P_1)\widehat{W}^{\dagger}(I - 2P_0)\widehat{W} \tag{5.98}$$

as the oblivious amplitude amplification operator. We summarize the result in the following lemma.

Lemma 5.5. *For any state $|\psi\rangle$, Let $|\Psi\rangle$, $|\Phi\rangle$, and F be defined as above. Then*

$$\||F|\Psi\rangle - |\Phi\rangle\| = O(1/r).$$

To prove this lemma, we need the following lemma, which slightly extends the results of [Kot14, Lemma 2.3].

Lemma 5.6. *For any $|\psi\rangle$, let $|\Psi\rangle$, $|\Phi\rangle$, $|\Phi^{\perp}\rangle$, P_0, and P_1 be defined as above. Let $|\Psi^{\perp}\rangle$ be a state satisfying the equation*

$$\widehat{W}|\Psi^{\perp}\rangle = \frac{\sqrt{3}}{2}|\Phi\rangle - \frac{1}{2}|\Phi^{\perp}\rangle. \tag{5.99}$$

Then $P_1|\Psi^{\perp}\rangle = O(1/r)$.

Proof. Define the operator

$$Q = \left(\langle\widehat{0}|\langle\widehat{\mu}| \otimes I\right) \widehat{W}^{\dagger} P_0 \widehat{W} \left(|\widehat{0}\rangle|\widehat{\mu}\rangle \otimes I\right). \tag{5.100}$$

For any state $|\psi\rangle$,

$$\langle\psi|Q|\psi\rangle = \left\| P_0 \widehat{W} \left(|\widehat{0}\rangle|\widehat{\mu}\rangle|\psi\rangle\right)\right\|^2 = \left\| P_0 \left(\frac{1}{2}|\Phi\rangle + \frac{\sqrt{3}}{2}|\Phi^{\perp}\rangle\right)\right\|^2 = \left\|\frac{1}{2}|\Phi\rangle\right\|^2 = \frac{1}{4} + O(1/r). \tag{5.101}$$

The last equality holds because $\||\Phi\rangle\|^2 = 1 + O(1/r)$, which follows from Eq. (5.94). Therefore, all the eigenvalues of Q are $\frac{1}{4} + O(1/r)$, and we can write

$$Q = \frac{1}{4} I + O\Big(\frac{1}{r}\Big), \tag{5.102}$$

where the big-O notation is with respect to the spectral norm. Now, for any $|\psi\rangle$, we have

$$\begin{aligned} Q|\psi\rangle &= \Big(\langle\widehat{0}|\langle\widehat{\mu}| \otimes I\Big)\, \widehat{W}^\dagger P_0 \widehat{W}\, \Big(|\widehat{0}\rangle|\widehat{\mu}\rangle|\psi\rangle\Big) && (5.103)\\ &= \frac{1}{2}\Big(\langle\widehat{0}|\langle\widehat{\mu}| \otimes I\Big)\, \widehat{W}^\dagger|\Phi\rangle && (5.104)\\ &= \frac{1}{2}\Big(\langle\widehat{0}|\langle\widehat{\mu}| \otimes I\Big)\, \Big(\frac{1}{2}|\Psi\rangle + \frac{\sqrt{3}}{2}|\Psi^\perp\rangle\Big) && (5.105)\\ &= \frac{1}{4}|\psi\rangle + \frac{\sqrt{3}}{4}\Big(\langle\widehat{0}|\langle\widehat{\mu}| \otimes I\Big)\, |\Psi^\perp\rangle. && (5.106) \end{aligned}$$

The third equality follows from Eqns. (5.97) and (5.99). On the other hand, by Eq. (5.102), we have

$$Q|\psi\rangle = \frac{1}{4}|\psi\rangle + O\Big(\frac{1}{r}\Big). \tag{5.107}$$

By Eqns. (5.106) and (5.107), we have

$$\Big(\langle\widehat{0}|\langle\widehat{\mu}| \otimes I\Big)\, |\Psi^\perp\rangle = O\Big(\frac{1}{r}\Big), \tag{5.108}$$

which implies $P_1|\Psi^\perp\rangle = O(1/r)$. □

Now we are ready to prove Lemma 5.5. The proof uses the methods in [BCC+15].

Proof of Lemma 5.5. First consider the operator $P_1\widehat{W}^\dagger P_0 \widehat{W}$. We have

$$P_1\widehat{W}^\dagger P_0\widehat{W}|\Psi\rangle = \frac{1}{2}P_1\widehat{W}^\dagger|\Phi\rangle = \frac{1}{2}P_1\left(\frac{1}{2}|\Psi\rangle + \frac{\sqrt{3}}{2}|\Psi^\perp\rangle\right) = \frac{1}{4}|\Psi\rangle + O\Big(\frac{1}{r}\Big), \tag{5.109}$$

where the second equality follows from Eqns. (5.97) and (5.99) and the last equality follows from Lemma 5.6. Then we have

$$\begin{aligned} F|\Psi\rangle &= (-\widehat{W}(I - 2P_1)\widehat{W}^\dagger(I - 2P_0)\widehat{W}|\Psi\rangle && (5.110)\\ &= (2P_0\widehat{W} + \widehat{W} - 4\widehat{W}P_1\widehat{W}^\dagger P_0\widehat{W})|\Psi\rangle && (5.111)\\ &= |\Phi\rangle + O(1/r). && (5.112) \end{aligned}$$

Therefore, $\|F|\Psi\rangle - |\Phi\rangle\| = O(1/r)$. □

Since $|\Phi\rangle$ is a purification of $\mathcal{M}_\delta^r[|\psi\rangle\langle\psi|]$, Lemma 5.5 implies that the circuit for F simulates a Stinespring dilation of $\mathcal{M}_\delta^r$ with error $O(1/r)$. This further implies that

$$\left\|\mathcal{N} - \mathcal{M}_\delta^r\right\|_\diamond = O(1/r), \tag{5.113}$$

where $\mathcal{N}$ is the quantum channel that F implements by tracing out indicator and purifier registers.

5.4.5 Concentration bound and encoding scheme

From the previous subsections, r is a parameter that determines the precision, which is $O(1/r)$. Up to this point, to simulate constant-time Lindblad evolution, the number of occurrences of the multiplexed-U gate in our construction is $O(r)$. In this subsection, we show how to reduce this to $O\big(\frac{\log(1/\epsilon)}{\log\log(1/\epsilon)}\big)$ while only introducing an additional error of ϵ.

It is important to note that, in light of Eqns. (5.69) and (5.71), there are $O(m)$ Kraus operators for $\mathcal{M}_\delta$ and each can be expressed as an LCU of $O(mq^2)$ terms.

Consider the initial state $(|0\rangle|\mu\rangle)^{\otimes r}$ of the indicator and purifier registers. At the beginning of the algorithm, the multiplexed-B gates applied on this state are multi-$B^{\otimes r}$. Note that the first term in Eq. (5.71) corresponds to the unitary I, which need not be performed. The circuit can be rearranged to bypass these operations, as in earlier papers on Hamiltonian evolution (see, for example, [BCC+17]).

More precisely, we compute the amplitude associated with this I being performed. For each instance of W acting on $|0\rangle|\mu\rangle|\psi\rangle$, consider the state of indicator and purifier registers which control the multiplexed-U gates (i.e., the state multi-$B|0\rangle|\mu\rangle$). The state $|0\rangle|0\rangle$ corresponds to unitary I. The amplitude of $|0\rangle|0\rangle$ is

$$\frac{s_0}{\sqrt{\sum_{j=0}^m s_j^2}}\frac{1}{\sqrt{s_0}} = \sqrt{\frac{s_0}{\sum_{j=0}^m s_j^2}} \tag{5.114}$$

$$= \sqrt{\frac{1+\delta/2\sum_{j=1}^m c_j^2 + \delta c_0}{1+2\delta\sum_{j=1}^m c_j^2 + 2\delta c_0 + \Theta(\delta^2(\sum_{j=1}^m c_j^2 + c_0)^2)}} \tag{5.115}$$

$$= \sqrt{1 - \frac{3\delta}{2}\sum_{j=1}^m c_j^2 - \delta c_0 + \Theta\left(\delta^2\Big(\sum_{j=1}^m c_j^2 + c_0\Big)^2\right)}, \tag{5.116}$$

where s_j and c_j are defined in Eqns. (5.74), (5.75), and (5.76) ($j \in \{0, \ldots, m\}$), and δ is defined in Eq. (5.85).

If this indicator and purifier registers are measured in the computational basis then the probability that the outcome is *not* $(0, 0)$ is

$$\frac{3\delta}{2}\sum_{j=1}^{m} c_j^2 + \delta c_0 + \Theta\left(\delta^2\left(\sum_{j=1}^{m} c_j^2 + c_0\right)^2\right) \leq \frac{3}{2}\delta \left\|\mathcal{L}\right\|_{\mathsf{pauli}} + \Theta\left(\delta^2 \left\|\mathcal{L}\right\|_{\mathsf{pauli}}^2\right) \tag{5.117}$$

$$= \frac{3}{2r} + \Theta\left(\frac{1}{r^2}\right). \tag{5.118}$$

Therefore, after the multi-B acting on $|0\rangle|\mu\rangle|\psi\rangle$, if the indicator and purifier registers are measured, then the probability that the outcome is not $(0, 0)$ is upper-bounded by $\frac{3}{2r}$.

Roughly speaking, this is qualitatively the same scaling that arises in Hamiltonian evolution simulation [BCC+17], hence the same so-called Hamming weight cut-off applies. Below is a more precise explanation of this.

In the indicator and purifier registers, after applying multi-B, the computational basis states of the indicator and purifier registers are of the form $|k_0, l_0\rangle \cdots |k_{r-1}, l_{r-1}\rangle$. Define the *Hamming weight* of such a state as the number of $j \in \{0, \ldots, r-1\}$ such that $(k_j, l_j) \neq (0, 0)$. If the indicator and purifier registers are restricted to states that have Hamming weight at most h then the circuit can be restructured so that there are only h occurrences of the multiplexed-U gates.

Let $X_1, \ldots, X_r$ be r independent random variables with $\Pr[X_j = 1] = \frac{3}{2r}$ and $\Pr[X_j = 0] = 1 - \frac{3}{2r}$ for all $j \in \{1, \ldots, r\}$. Consider the state of the indicator and purifier registers right before multiplexed-U gates are applied (i.e., the state (multi-$B|0\rangle|\mu\rangle)^{\otimes r}$). We are interested in how much amplitude is associated with the low Hamming weight states. This is related to the Chernoff bound (see [MR95]), i.e., for all $\delta > 0$, it holds that

$$\Pr\left[\sum_{j=1}^{r} X_j > (1+\delta)\mu\right] < \frac{e^{\delta\mu}}{(1+\delta)^{(1+\delta)\mu}}, \tag{5.119}$$

where $\mu = \sum_{j=1}^{r} \Pr[X_j = 1] = \frac{3}{2}$. Letting $h = (1+\delta)\mu$, we have

$$\Pr\left[\sum_{j=1}^{r} X_j > h\right] < \frac{e^{h-\mu}\mu^h}{h^h} \leq \frac{(e\mu)^h}{h^h} = \frac{(3e/2)^h}{h^h}. \tag{5.120}$$

Therefore, the probability of the Hamming weight being larger than h is upper bounded by ϵ^2 provided

$$h \in O\left(\frac{\log(1/\epsilon)}{\log\log(1/\epsilon)}\right). \tag{5.121}$$

From this, we conclude that the occurrences of the multiplexed-U gates can be reduced to $O\left(\frac{\log(1/\epsilon)}{\log\log(1/\epsilon)}\right)$ with error ϵ.

Now, the number of qubits for indicator and purifier registers in a segment is still $O(r\log(mq))$. We use the similar compression scheme as in [BCC+17] to reduce the number of qubits for indicator and purifier registers. The intuition is to only store the positions of components with nonzero Hamming weight, and we also need two other registers to store the actual state in this position.

The compression scheme works as follows. First note that the multiplex-B gate acts on $|0\rangle|\mu\rangle$ as

$$\text{multi-}B|0\rangle|\mu\rangle = \frac{1}{\sqrt{\sum_{j=0}^{m-1} s_j^2}} \sum_{j=0}^{m-1}\sum_{k=0}^{mq^2} \sqrt{s_j}\sqrt{\alpha_{jk}}|k\rangle|j\rangle. \tag{5.122}$$

We consider the initial sate $(|0\rangle|\mu\rangle)^{\otimes r}$. After applying the multiplexed-B gates (before applying the multiplexed-U gates), the state becomes $(\text{multi-}B|0\rangle|\mu\rangle)^{\otimes r}$, which is

$$(\text{multi-}B|0\rangle|\mu\rangle)^{\otimes r} = \left(\frac{\sqrt{s_0}\sqrt{\alpha_{00}}|0\rangle|0\rangle + \sum_{j=1}^{m-1}\sum_{k=1}^{mq^2}\sqrt{s_j}\sqrt{\alpha_{jk}}|k\rangle|j\rangle}{\sqrt{\sum_{j=0}^{m-1} s_j^2}}\right)^{\otimes r} \tag{5.123}$$

$$= \left(\kappa|0\rangle|0\rangle + \sigma\frac{\sum_{j=1}^{m-1}\sum_{k=1}^{mq^2}\sqrt{s_j}\sqrt{\alpha_{jk}}}{\sqrt{\sum_{j=0}^{m-1} s_j^2 - s_0\alpha_{00}}}|k\rangle|j\rangle\right)^{\otimes r}, \tag{5.124}$$

where

$$\kappa = \sqrt{\frac{s_0\alpha_{00}}{\sum_{j=0}^{m-1} s_j^2}}, \text{ and } \sigma = \sqrt{\frac{\sum_{j=0}^{m-1} s_j^2 - s_0\alpha_{00}}{\sum_{j=0}^{m-1} s_j^2}}. \tag{5.125}$$

The compressed encoding is performed in two steps. First, we prepare the state

$$\sum_{j\in \mathrm{G}} \kappa^{r-|g|}\sigma^{|g|}|g\rangle + \eta|\perp\rangle, \tag{5.126}$$

where $|\cdot|$ denotes the Hamming weight, $\mathrm{G} = \{(g_1 \ldots, g_{h'}) : 1 \leq h' \leq h, g_1 + \cdots + g_{h'} \leq r - h'\}$, and $|\perp\rangle$ (with amplitude η) is some state that is orthogonal to all terms in the first sum. It is shown in [BCG14, Section 4.2–4.4] that this state can be prepared with $O(h(\log(r) + \log\log(1/\epsilon)))$ 1- and 2-qubit gates so that $|\eta|^2 \leq \epsilon$. Second, we add one register to prepare the state

$$\left(\sum_{j\in \mathrm{G}} \kappa^{r-|g|}\sigma^{|g|}|g\rangle + \eta|\perp\rangle\right)\left(\frac{\sum_{j=1}^{m-1}\sum_{k=1}^{mq^2}\sqrt{s_j}\sqrt{\alpha_{jk}}|k\rangle|j\rangle}{\sqrt{\sum_{j=0}^{m-1} s_j^2 - s_0\alpha_{00}}}\right), \tag{5.127}$$

where the state in the second register can be prepared by a slightly modified multiplex-B gate and state $|\mu\rangle$, and it can be implemented with $O(m^2q^2)$ 1- and 2-qubit gates.

Note that there is a natural one-to-one correspondence between the set G and the set of binary strings with Hamming weight at most h as

$$g_1 \ldots, g_{h'} \leftrightarrow 0^{g_1}10^{g_2}10^{g_3}\ldots 0^{g_{h'}}10^{r-h-g_1-\ldots-g_{h'}}. \tag{5.128}$$

To see that Eq. (5.127) encodes the state (multi-$B|0\rangle|\mu\rangle)^{\otimes r}$ for the terms with Hamming weight at most h, observe that the state

$$|(g_1, \ldots, g_{h'})\rangle\left(\frac{\sum_{j=1}^{m-1}\sum_{k=1}^{mq^2}\sqrt{s_j}\sqrt{\alpha_{jk}}|k\rangle|j\rangle}{\sqrt{\sum_{j=0}^{m-1} s_j^2 - s_0\alpha_{00}}}\right) \tag{5.129}$$

corresponds to

$$(|0\rangle|0\rangle)^{\otimes g_1}|\nu\rangle\ldots(|0\rangle|0\rangle)^{\otimes g_2}|\nu\rangle(|0\rangle|0\rangle)^{\otimes(r-h-g_1-\ldots-g_{h'})} \tag{5.130}$$

where

$$|\nu\rangle = \frac{\sum_{j=1}^{m-1}\sum_{k=1}^{mq^2}\sqrt{s_j}\sqrt{\alpha_{jk}}|k\rangle|j\rangle}{\sqrt{\sum_{j=0}^{m-1} s_j^2 - s_0\alpha_{00}}}. \tag{5.131}$$

It follows that Eq. (5.127) has the desired amplitudes for the terms with nonzero Hamming weight in Eq. (5.124).

In this encoding scheme, the register for $|g\rangle$ requires $O(\log(r)h)$ qubits. The two additional registers require $O(h\log(mq))$ qubits. If we prepare the indicator and purifier registers in this encoded representation, the number of qubits is

$$O((\log(r)h + \log(mq)h)) = O((\log(1/\epsilon) + \log(mq))h). \tag{5.132}$$

We summarize this encoding scheme as follows. In the original representation, the initial state of the indicator and purifier registers is $(|0\rangle|\mu\rangle)^{\otimes r}$, and we apply multiplexed-B gates multi-$B^{\otimes r}$ on this state before applying multiplexed-U gates. In the encoded representation, the initial state is $|0^a\rangle|0^b\rangle|0^c\rangle$, where $a = O(\log(r)h)$, $b = O(\log(mq)h)$, and $c = O(\log(m)h)$; the first and second registers correspond to the indicator register in the original representation, and the third register corresponds to the purifier register in the original representation. We denote the encoding operator by E. The operator E corresponds to the multiplexed-B gates in the original representation, as we apply E on the encoded initial state before applying multiplexed-U gates.

5.4.6 Total number of gates and proof of the main theorem

In this subsection, we count the number of 1- and 2-qubit gates in our construction. There are three parts that we need to consider: the implementation of the encoding operator E, the implementation of the reflections in the oblivious amplitude amplification operator, and the implementation of the multiplexed-U gates. To complete the proof of the main theorem, all that remains is to bound the number of these gates.

Proof of Theorem 5.1. We first consider the case where t is as defined in Eq. (5.90), so $t\,\|\mathcal{L}\|_{\mathsf{pauli}} = \ln(2) + O(1/r)$. The quantum circuit is based on the oblivious amplitude amplification operator F, whose correctness is shown by Lemma 5.5. We modify the quantum circuit of F by applying a concentration bound and the encoding scheme on the indicator and purifier registers as shown in Subsection 5.4.5. In the following, we show that this quantum circuit achieves the desired gate complexity.

For the encoding operator E, we first apply the techniques in [BCG14] for $(|0\rangle|\mu\rangle)^{\otimes r}$ to prepare the state as in Eq. (5.126). This can be done with $O\left(h(\log(r) + \log\log(1/\epsilon))\right) \in O(\log(r)h)$ gates. In addition, we need prepare a superposition of the basis states with nonzero Hamming weight in the second and third registers as in the second register in Eq. (5.127). This can be done with a slightly modified multiplex-B gate and state $|\mu\rangle$ with gate cost $O(m^2q^2)$. Thus, the number of 1- and 2-qubit gates required for the encoding operator E is $O(\log(r)h + m^2q^2) = O(\log(1/\epsilon)h + m^2q^2)$.

In the oblivious amplitude amplification operator F, there are two reflections, $I - 2P_0$ and $I - 2P_1$, between $\widehat{W}$ and $\widehat{W}^\dagger$. If we look into the constructions for $\widehat{W}$, the two reflections are between multiplexed-B gates. To translate the operator $(\text{multi-}B^{\otimes r})\,(I - 2P_1)\left(\text{multi-}B^{\dagger\otimes r}\right)$ to the encoded representation, note that the multiplexed-B gates correspond to the encoding operator E in the encoded representation, and the reflection $I - 2P_1$

is the reflection about the initial state $|0\rangle^{\otimes r}|\mu\rangle^{\otimes r}$. Hence in the encoded representation, the corresponding operation is first applying $E^\dagger$, reflecting about the encoded initial state $|0^a\rangle|0^b\rangle|0^c\rangle$, where a, b, and c are defined in the last paragraph of Subsection 5.4.5, and then applying E.

A similar method applies to the operation $(\text{multi-}B^{\otimes r})\,(I-2P_0)\left(\text{multi-}B^{\dagger\otimes r}\right)$. The only difference is that the reflection $I-2P_0$ is reflecting about the subspace where the state of the indicator register is $|0\rangle^{\otimes r}$. In the encoded representation, the corresponding reflection in the encoded representation should be about the subspace where the first two registers is in the state $|0^a\rangle|0^b\rangle$. Therefore, the corresponding operation in the encoded representation is first applying $E^\dagger$, then applying the reflection about the encoded state $|0^a\rangle|0^b\rangle$ on the first two registers, and last applying E.

The number of 1- and 2-qubit gates involved in the two reflections consists of the implementation of the encoding operator E, and two reflections. The number of gates for the reflections is of the same order of the number of qubits for the encoded representation. Therefore the number of 1- and 2-qubit gates in this part is $O(\log(1/\epsilon)h + m^2q^2)$.

Each multiplexed-U gate costs $O(mq^2(\log(mq)+n))$ of 1- and 2-qubit gates, as each controlled-U requires $\log(mq)$ qubits for multiplexing and $O(n)$ Paulis, and we have to implement $O(mq^2)$ these controlled-U gates. Since the number of occurrences of multiplexed-U gates is h, the gate cost for this part is $O(mq^2h(\log(mq)+n))$

Therefore, the total number of 1- and 2-qubit gates is

$$O(m^2q^2 + \log(1/\epsilon)h + mq^2(\log(mq)+n)h) \in O\left(m^2q^2\frac{(\log(mq/\epsilon)+n)\log(1/\epsilon)}{\log\log(1/\epsilon)}\right). \tag{5.133}$$

For arbitrary evolution time t, let $\tau := t\,\|\mathcal{L}\|_{\text{pauli}}$. Divide the normalized evolution time into $O(\tau)$ segments. Then run this quantum circuit for a segment with precision ϵ/τ and trace out the indicator and purifier registers. Repeat this $O(\tau)$ times and this evolution is simulated with total number of 1- or 2-qubit gates $O\left(m^2q^2\tau\frac{(\log(mq\tau/\epsilon)+n)\log(\tau/\epsilon)}{\log\log(\tau/\epsilon)}\right)$.

The distance between $\mathcal{N}$ and $e^{\mathcal{L}t}$ in terms of the diamond norm is established by Eqns. (5.91) and (5.113). Choosing r large enough (i.e., $r = 1/\epsilon$), the error of the simulation is within ϵ. Note that the concentration bound and encoding scheme only cause $O(\epsilon)$ error. □

5.5 Lindbladians with sparse Hamiltonian and Lindblad operators

In this subsection, we sketch the analysis of the simulation of Lindbladians with d-sparse Hamiltonian and Lindblad operators. Without loss of generality, we assume $\|H\| \geq 1$ and $\|L_j\| \geq 1$ for $j \in \{1, \ldots, m\}$. We first describe a method to approximate H and L_j as a linear combination of unitaries. Then we sketch the analysis of two key quantities: the normalized evolution time $t \|\mathcal{L}\|_{\mathsf{pauli}}$, and the number of unitaries q in this approximation

We consider the case where H is d-sparse and each L_j is both column and row d-sparse given by an oracle. Each L_j can be decomposed as $L_j = \frac{L_j + L_j^\dagger}{2} + i\frac{L_j - L_j^\dagger}{2i}$, where $\frac{L_j + L_j^\dagger}{2}$ and $\frac{L_j - L_j^\dagger}{2i}$ are Hermitian. Let the *max norm* of an operator $A \in \mathrm{L}(\mathbb{C}^{2^n})$, denoted by $\|A\|_{\mathsf{max}}$, be defined as $\|A\|_{\mathsf{max}} = \max_{j,k} |\langle j|A|k\rangle|$. For each $\frac{L_j + L_j^\dagger}{2}$, $\frac{L_j - L_j^\dagger}{2i}$, and H, we use the methods in [BCC+15] to approximate them as a linear combination of unitaries with equal coefficient γ. In particular, for any $H \in \mathrm{L}(\mathbb{C}^N)$ that is Hermitian and is d-sparse, this approximation can be accomplished in the following two steps:

1. Obtain a decomposition $H = \sum_{j=1}^{d^2} H_j$, where each H_j is 1-sparse and a query to H_j can be simulated with $O(1)$ queries to H. [BCC+15, Lemma 4.4]

2. Further decompose each 1-sparse Hamiltonian H_j into $O(\|H_j\|/\gamma)$ unitary Hamiltonians H_{jk} such that $\|H_j - \gamma \sum_k H_{jk}\|_{\mathsf{max}} \leq \sqrt{2}\gamma$. [BCC+15, Lemma 4.3]

The above approximation also works for each $\frac{L_j + L_j^\dagger}{2}$ and $\frac{L_j - L_j^\dagger}{2i}$. The error of the approximation is $O(d^2\gamma)$ in terms of the max norm, and the number of unitaries in the approximation is $O\big(d^2 \|H\|/\gamma\big)$ for H and $O\left(d^2 \|L_j\|/\gamma\right)$ for L_j. Let c_0 be the sum of coefficients in the LCU approximation of H, and c_j be the sum of coefficients in the LCU approximation of L_j for $j \in \{1, \ldots, m\}$. It is easy to see that $c_0 = O\left(d^2 \|H\|\right)$ and $c_j = O\left(d^2 \|L_j\|\right)$ for $j \in \{1, \ldots, m\}$. We have $t \|\mathcal{L}\|_{\mathsf{pauli}} = t\big(c_0 + \sum_{j=1}^m c_j^2\big) \in O\big(td^4 \|\mathcal{L}\|_{\mathsf{ops}}\big)$.

To bound q, which is the number of terms in the LCU decomposition for each of H and L_j, we consider the error of this approximation. As each of H and L_j can be approximated with error $O(d^2\gamma)$ in terms of the max norm, $\mathcal{L}$ can be approximated with error $O\big(d^3 \|\mathcal{L}\|_{\mathsf{ops}}\, \gamma\big)$ in terms of the diamond norm. To see this, for all $\rho \in \mathrm{D}(\mathbb{C}^{2^n})$, let

$$\mathcal{L}'[\rho] = -i[H', \rho] + \sum_{j=1}^{m} \left(L_j' \rho L_j'^\dagger - \frac{1}{2} L_j'^\dagger L_j' \rho - \frac{1}{2} \rho L_j'^\dagger L_j' \right), \tag{5.134}$$

with $\|H - H'\|_{\mathsf{max}} \in O(d^2\gamma)$ and $\left\|L_j - L'_j\right\|_{\mathsf{max}} \in O(d^2\gamma)$ for all $j \in \{1, \ldots, m\}$. For all d-sparse operator $A \in \mathrm{L}(\mathbb{C}^{2^n})$, it holds that $\|A\| \leq d \|A\|_{\mathsf{max}}$ [CK10]. Hence, $\|H - H'\| \in O(d^3\gamma)$ and $\left\|L_j - L'_j\right\| \in O(d^3\gamma)$ for all $j \in \{1, \ldots, m\}$. We have

$$(\mathcal{L}' - \mathcal{L})[\rho] = -i[(H' - H), \rho] + \sum_{j=1}^{m} \left(L'_j \rho L_j'^{\dagger} - L_j \rho L_j^{\dagger} - \frac{1}{2}\left(L_j'^{\dagger} L'_j \rho - L_j^{\dagger} L_j \rho \right) - \frac{1}{2}\left(\rho L_j'^{\dagger} L'_j - \rho L_j^{\dagger} L_j \right) \right), \quad (5.135)$$

for all $\rho \in \mathrm{D}(\mathbb{C}^{2^n})$. Furthermore,

$$\|(\mathcal{L}' - \mathcal{L})[\rho]\|_1 \leq 2\|H' - H\| + \sum_{j=1}^{m} \left(2\|L_j\| \left\|L'_j - L_j\right\| + 2\left\|L'_j\right\| \left\|L'_j - L_j\right\| \right) \quad (5.136)$$

$$\in O\left(d^3 \gamma \|\mathcal{L}\|_{\mathsf{ops}} \right). \quad (5.137)$$

The bound $\|\mathcal{L}' - \mathcal{L}\|_{\diamond} \in O(d^3\gamma \|\mathcal{L}\|_{\mathrm{ops}})$ follows from an extension of the above inequality to a larger Hilbert space by considering the operator $H' \otimes I_{\mathbb{C}^{2^n}}$, $H \otimes I_{\mathbb{C}^{2^n}}$, $L'_j \otimes I_{\mathbb{C}^{2^n}}$ and $L_j \otimes I_{\mathbb{C}^{2^n}}$ for all $j \in \{1, \ldots, m\}$. To restrict the simulation error within ϵ for evolution time t, γ can be chosen so that $\gamma = O\big(\epsilon/(td^3 \|\mathcal{L}\|_{\mathsf{ops}})\big)$. Therefore, the number of unitaries in the decomposition is bounded by $q = O\big(td^5 \|\mathcal{L}\|_{\mathsf{ops}}^2 / \epsilon\big)$.

In the implementation of the multiplexed-U gates, we no longer need to implement all the $O(mq^2)$ unitaries, since the oracles for H and L_j are given. Also, the cost for implementing the encoding scheme becomes $O(\log(1/\epsilon)h + \mathrm{poly}(n))$ as the coefficients in the LCU for each H and L_j are the same, which saves the $O(m^2q^2)$ factor. Thus the m and q factor in the gate complexity can be eliminated. (The $\log(m)$ and $\log(q)$ factor will be preserved.) Let $\tau = t\|\mathcal{L}\|_{\mathsf{ops}}$. By our construction, the gate complexity is

$$O\left(\tau \,\mathrm{polylog}(mq\tau/\epsilon)\,\mathrm{poly}(n, d)\right).$$

The query complexity is the number of occurrences of the multiplexed-U gates, which is

$$O\left(\tau \frac{\log(\tau/\epsilon)}{\log\log(\tau/\epsilon)} \mathrm{poly}(d) \right).$$

Chapter 6

Harnessing Open Quantum Systems: Dissipative Quantum Search

In this chapter, we provide methods for solving the "over-cooking" problem of Grover's algorithm (briefly discussed in Section 1.3): if the number of marked items M is unknown, how to produce a marked state while preserving the quadratic speedup.

A search problem can be modeled as in a search space of N items with a boolean function f to partition a set of N items. Without loss of generality, assume N is a power of 2. This set can be labeled by natural numbers so that we can refer to this set as $\{0, \ldots, N-1\}$. We call an item *marked* if $f(j) = 1$. Otherwise, we call it *non-marked*. The goal is to find a marked item with as few queries to f as possible. All classical algorithms need to make $\Omega(N/M)$ queries to f, where $M = |\{j : f(j) = 1\}|$ is the number of marked items. Remarkably, quantum algorithms permit a quadratic speedup: $O(\sqrt{N/M})$ queries are sufficient. This quadratic speedup is achieved by Grover's algorithm. We give a brief overview of this algorithm in Section 6.2. In this chapter, we consider a more general search problem where only *one* copy of a non-trivial initial state is given, which is discussed in detail in Subsection 6.1.2. The results in this chapter are not previously published.

6.1 Previous work and main results

6.1.1 Previous work

Methods have been proposed to address this over-cooking problem. One representative of these methods is shown in [BBHT98], where a trial-and-error approach is used: one first apply a small number of iterations; if a marked item is not found, keep increasing the number of iterations. As presented in [BBHT98], this trial-and-error approach uses multiple copies of the initial state. It is possible to use only one copy of the initial state: if a marked item is not found, the state collapses to the superposition of the unmarked states, where the next trial can start with, and the query complexity is still $O(\sqrt{N/M})$.

Other techniques such as quantum counting [BHT98] and quantum amplitude estimation [BHMT02] can also be used to address this over-cooking problem. These approaches consist of two steps: first estimate M; then apply Grover's algorithm for a desired number of iterations. With these two-step approaches, at least two copies of the initial state are required.

Although the above approaches perform well, there is some aesthetic appeal in the possibility of an algorithm that naturally converges to the target stage through its iterative process, without extra intermediate steps such as measurements and classical randomness. This is usually called the *fixed-point property*. Grover *et al.* [Gro05, GPT06] proposed the $\pi/3$-search algorithm where a sequence of operators (which are more generalized version of the Grover iterator) is built in a recursive manner. After applying these operators, the state becomes monotonically closer to the target state but never passes the target state. This algorithm perfectly solves the over-cooking problem. However, the quadratic speedup is lost [CRR05].

Yoder *et al.* [YLC14] proposed the first fixed-point quantum search algorithm which preserves the quadratic speedup. The fixed-point property of this algorithm is presented in the following sense: for any N, M_{min}, and a desired error tolerance ϵ, a sequence of $O(\sqrt{N/M_{\mathsf{min}}}\log(1/\epsilon))$ unitaries (which are more generalized version of the Grover iterator) is constructed such that for any instance of the search problem with N items and M marked items (with $M \geq M_{\mathsf{min}}$), this sequence of unitaries yields a state that is ϵ-close to the target state (in terms of the Euclidean norm). As each unitary in this sequence uses $O(1)$ queries to f, the query complexity of this algorithm is $O(\sqrt{N/M_{\mathsf{min}}}\log(1/\epsilon))$. This sequence of operators can be extended to obtain better results (in terms of ϵ). However, the complexity of this sequence extension is multiplicative, and the quadratic speedup is likely to be lost.

6.1.2 Main results

In this chapter, we present a novel fixed-point approach for solving the search problem that preserves the quadratic speedup against classical algorithms. Our algorithm works in a natural fixed-point manner: the error (in terms of the trace-distance between the current state and the target state) converges with more iterations, though not necessarily monotonically. Moreover, only one copy of the initial state suffices and no intermediate measurement or classical randomness is required.

Instead of working on the search problem for N items, we define the search problem abstractly in terms of amplitude amplification, and in a context where the initial state can be nontrivial.

Definition 6.1. *In this* general search problem*, let* $\mathrm{G} \subseteq \mathbb{C}^N$ *be an unknown subspace of* $\mathbb{C}^N$*. We are given* one *copy of an initial state* $|\psi_{\mathsf{init}}\rangle \in \mathbb{C}^N$ *and the following oracles:*

- *An* initial state verification oracle $Q_{\mathsf{init}} \in \mathrm{U}(\mathbb{C}^2)$ *that performs the following mapping*

$$Q_{\mathsf{init}}|\psi_{\mathsf{init}}\rangle = -|\psi_{\mathsf{init}}\rangle, \text{ and} \tag{6.1}$$
$$Q_{\mathsf{init}}|\psi_{\mathsf{init}}^{\perp}\rangle = |\psi_{\mathsf{init}}^{\perp}\rangle, \tag{6.2}$$

  where $|\psi_{\mathsf{init}}^{\perp}\rangle \in \mathbb{C}^N$ *is a state that is orthogonal to* $|\psi_{\mathsf{init}}\rangle$*. A controlled-*$Q_{\mathsf{init}}$ *is also available.*

- *A* solution verification oracle $Q_{\mathsf{good}} \in \mathrm{U}(\mathbb{C}^N \otimes \mathbb{C}^2)$ *that performs the following mapping*

$$Q_{\mathsf{good}}|\psi\rangle|b\rangle = \begin{cases} |\psi\rangle|b \oplus 1\rangle, & \text{if } |\psi\rangle \in \mathrm{G}, \\ |\psi\rangle|b\rangle, & \text{if } |\psi\rangle \text{ is orthogonal to } \mathrm{G}. \end{cases} \tag{6.3}$$

  A controlled-Q_{good} *is also available.*

- *A* guessing oracle R*, which is a unitary that, starting in state* $|\psi_{\mathsf{init}}\rangle$*, yields a guess of a state in the target space with probability* p *in the sense that*

$$\max_{|\psi\rangle \in \mathrm{G}} |\langle\psi|R|\psi_{\mathsf{init}}\rangle|^2 = p. \tag{6.4}$$

  We assume that p *is unknown.*

Let $|\psi_{\mathsf{good}}\rangle$ *be the projection of* $|\psi_{\mathsf{init}}\rangle$ *on* G*. The goal is to create the state* $|\psi_{\mathsf{good}}\rangle$ *with as few queries to* Q_{good} *and* Q_{init} *as possible.*

One of the main results of this chapter is stated in the following theorem.

Theorem 6.1. *Consider the general search problem defined in Definition 6.1. Assume $p \leq 1/4$. Fix an $\epsilon > 0$. There exists a quantum algorithm acting on two registers $\mathbb{C}^2 \otimes \mathbb{C}^{2^n}$, where the state in the second register ($\mathbb{C}^{2^n}$) after a total number of j steps starting from $|\psi_{\mathsf{init}}\rangle\langle\psi_{\mathsf{init}}|$ is denoted by ρ_j, such that $\|\rho_j - |\psi_{\mathsf{good}}\rangle\langle\psi_{\mathsf{good}}|\|_1 \leq \epsilon$ for all $j \geq c \cdot \frac{1}{\sqrt{p}} \log(1/\epsilon)$, where c is some constant independent of p and ϵ. Moreover, each step of this quantum algorithm can be implemented using a constant number of queries to Q_{good} and Q_{init}.*

We prove Theorem 6.1 in Section 6.4.

6.2 Review of Grover's algorithm

In this section, we give a brief review of Grover's algorithm [Gro96] for solving the search problems in a more general model. For a more comprehensive review of these algorithms, readers may refer to the book by Nielsen and Chuang [NC00].

We first consider the more general search problem defined in Definition 6.1. Here, we assume that p is known. Recall the solution verification oracle $Q_{\mathsf{good}} \in \mathrm{U}(\mathbb{C}^N \otimes \mathbb{C}^2)$ defined in Eq. (6.3). It is convenient to conjugate the second register by the Hadamard gate H, and set $b = 1$. This yields a unitary $(I \otimes H)Q_{\mathsf{good}}(I \otimes H)$ that performs the following mapping

$$(I \otimes H)Q_{\mathsf{good}}(I \otimes H)|\psi_{\mathsf{good}}\rangle|1\rangle = -|\psi_{\mathsf{good}}\rangle|1\rangle, \text{ and} \tag{6.5}$$

$$(I \otimes H)Q_{\mathsf{good}}(I \otimes H)|\psi^{\perp}_{\mathsf{good}}\rangle|1\rangle = |\psi^{\perp}_{\mathsf{good}}\rangle|1\rangle, \tag{6.6}$$

where $|\psi^{\perp}_{\mathsf{good}}\rangle$ is a state that is orthogonal to G. Effectively, this is equivalent to a unitary $\widehat{Q}_{\mathsf{good}} \in \mathrm{U}(\mathbb{C}^N)$ that flips the phase of marked states as

$$\widehat{Q}_{\mathsf{good}}|\psi_{\mathsf{good}}\rangle = -|\psi_{\mathsf{good}}\rangle, \text{ and} \tag{6.7}$$

$$\widehat{Q}_{\mathsf{good}}|\psi^{\perp}_{\mathsf{good}}\rangle = |\psi^{\perp}_{\mathsf{good}}\rangle. \tag{6.8}$$

The Grover iterator is defined as

$$G = \left(RQ_{\mathsf{init}}R^{\dagger}\right)\widehat{Q}_{\mathsf{good}}. \tag{6.9}$$

To view Grover's algorithm in a geometric way, let θ be the angle such that $\sin(\theta/2) = \sqrt{p}$ and $\cos(\theta/2) = \sqrt{1-p}$. Write $R|\psi_{\mathsf{init}}\rangle = \cos(\theta/2)|\psi^{\perp}_{\mathsf{good}}\rangle + \sin(\theta/2)|\psi_{\mathsf{good}}\rangle$, where $|\psi^{\perp}_{\mathsf{good}}\rangle$

is some state orthogonal to G. Then, we conduct the analysis with respect to the basis $\{|\psi_{\mathsf{good}}^{\perp}\rangle, |\psi_{\mathsf{good}}\rangle\}$. The operator G can be represented as

$$G = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}. \tag{6.10}$$

Starting with the state $R|\psi_{\mathsf{init}}\rangle$, each application of G rotates the angle by θ. After $O(1/\sqrt{p})$ applications of G, the state will be close to $|\psi_{\mathsf{good}}\rangle$.

Remark. The search problem that was considered by Grover [Gro96] is a special case of the more general search problem defined in Definition 6.1 in the following sense. Let the set $\{0, \ldots, N-1\}$ be a search space of N items. Assume $N = 2^n$ is a power of 2. A boolean function $f : \{0, \ldots, N-1\} \to \{0, 1\}$ is used to determine whether an item j is marked ($f(j) = 1$) or not ($f(j) = 0$). The target state is defined as

$$|\psi_{\mathsf{good}}\rangle = \frac{1}{\sqrt{M}} \sum_{j:f(j)=1} |j\rangle, \tag{6.11}$$

where $M = |\{j : f(j) = 1\}|$. The initial state is $|0\rangle_N$, and the guessing oracle is the n-fold Hadamard gate $H^{\otimes n}$. The initial state verification oracle is simply the unitary $I_{\mathbb{C}^N} - 2|0\rangle\langle 0|_N$. The solution verification oracle is defined as

$$Q_{\mathsf{good}}|j\rangle|b\rangle = |j\rangle|b \oplus f(j)\rangle, \tag{6.12}$$

for $b \in \{0, 1\}$. We have that $|\langle\psi_{\mathsf{good}}|H^{\otimes n}|\psi_{\mathsf{init}}\rangle|^2 = M/N$. Therefore, the number of queries is $O(\sqrt{N/M})$.

6.3 The dissipative query model

In this section, we present a gadget for the dissipative quantum search algorithm, namely, the dissipative query model. In Section 6.2, we have reviewed the standard query model, which performs the maps $|\psi_{\mathsf{good}}\rangle|b\rangle \mapsto |\psi_{\mathsf{good}}\rangle|b \oplus 1\rangle$ and $|\psi_{\mathsf{good}}^{\perp}\rangle|b\rangle \mapsto |\psi_{\mathsf{good}}^{\perp}\rangle|b\rangle$. For the dissipative query model, the intuition is to non-reversibly change the state of the second register via the amplitude damping process.

To begin with, we first define a parameterized amplitude damping channel.

Definition 6.2. *The* amplitude damping channel *with* damping strength $\lambda \geq 0$*, denoted by* $\mathcal{AD}_\lambda$*, is a quantum channel acting on* $\mathrm{L}(\mathbb{C}^2)$ *that performs the following mapping.*

$$\mathcal{AD}_\lambda[\rho] = A_0\rho A_0^\dagger + A_1\rho A_1^\dagger, \tag{6.13}$$

for all $\rho \in \mathrm{D}(\mathbb{C}^2)$ *where*

$$A_0 = \begin{pmatrix} 1 & 0 \\ 0 & e^{-\lambda/2} \end{pmatrix}, \text{ and } A_1 = \begin{pmatrix} 0 & \sqrt{1-e^{-\lambda}} \\ 0 & 0 \end{pmatrix}. \tag{6.14}$$

Note that when $\lambda \to \infty$, $\mathcal{AD}_\infty$ is the *completely amplitude damping channel*, and when $\lambda = 0$, $\mathcal{AD}_0$ is the identity channel. The amplitude damping channel with damping strength λ can be viewed as a continuous-time process in terms of λ, as

$$\mathcal{AD}_{\lambda_1} \circ \mathcal{AD}_{\lambda_0} = \mathcal{AD}_{\lambda_0+\lambda_1}, \tag{6.15}$$

for all $\lambda_0, \lambda_1 \geq 0$. Moreover, this continuous-time process is Markovian and thereby can be described by a Lindblad equation specified as follows:

$$\mathcal{L}_{\mathrm{ad}}[\rho] = L_{\mathrm{ad}} \rho L_{\mathrm{ad}}^\dagger - \frac{1}{2}\left(L_{\mathrm{ad}}^\dagger L_{\mathrm{ad}} \rho + \rho L_{\mathrm{ad}}^\dagger L_{\mathrm{ad}}\right), \tag{6.16}$$

where the jump operator L_{ad} is defined as

$$L_{\mathrm{ad}} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}. \tag{6.17}$$

The Lindbladian $\mathcal{L}_{\mathrm{ad}}$ generates $\mathcal{AD}_\lambda$ in the sense that

$$\mathcal{AD}_\lambda = e^{\mathcal{L}_{\mathrm{ad}}\lambda}. \tag{6.18}$$

Next, we extend the amplitude damping channel with damping strength λ to a controlled channel. More formally, we have the following definition:

Definition 6.3. *The* controlled amplitude damping channel *with* damping strength $\lambda \geq 0$*, denoted by c-*$\mathcal{AD}_\lambda$*, is a quantum channel acting on* $\mathrm{L}(\mathbb{C}^2 \otimes \mathbb{C}^2)$ *that performs the following mapping:*

$$c\text{-}\mathcal{AD}_\lambda[\rho] = B_0 \rho B_0^\dagger + B_1 \rho B_1^\dagger, \tag{6.19}$$

for all $\rho \in \mathrm{D}(\mathbb{C}^2 \otimes \mathbb{C}^2)$*, where*

$$B_0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{-\lambda/2} \end{pmatrix}, \text{ and } B_1 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sqrt{1-e^{-\lambda}} \\ 0 & 0 & 0 & 0 \end{pmatrix}. \tag{6.20}$$

It can be verified that the controlled amplitude damping channel with damping strength λ performs the following mapping

$$|b\rangle\langle b| \otimes \rho \mapsto \begin{cases} |b\rangle\langle b| \otimes \mathcal{AD}_\lambda[\rho], & \text{when } b = 1 \\ |b\rangle\langle b| \otimes \rho, & \text{when } b = 0. \end{cases} \tag{6.21}$$

We use the circuit in Figure 6.1 to represent such a channel.

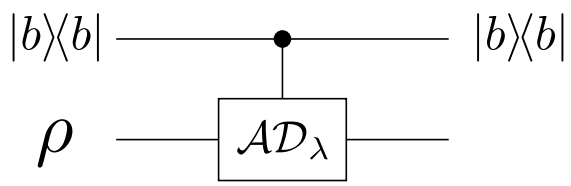

Figure 6.1: The circuit representation of the controlled amplitude damping channel with damping strength λ.

Now, we use a standard query oracle, instead of a single qubit, to control the amplitude damping channel. To do this, we consider a 2-dimensional subspace for the n-qubit register $\mathbb{C}^N$ (where $N = 2^n$). Define the state $|r\rangle$ as

$$|r\rangle = \frac{1}{\sqrt{1-p}}(|\psi_{\mathsf{init}}\rangle - \sqrt{p}|\psi_{\mathsf{good}}\rangle). \tag{6.22}$$

Then the set $\{|r\rangle, |\psi_{\mathsf{good}}\rangle\}$ forms an orthonormal basis for the 2-dimensional subspace spanned by $|\psi_{\mathsf{init}}\rangle$ and $|\psi_{\mathsf{good}}\rangle$. In the following definition, we only consider the case where the state of the first register is in the 2-dimensional subspace spanned by $|\psi_{\mathsf{init}}\rangle$ and $|\psi_{\mathsf{good}}\rangle$.

Definition 6.4. *Provided the state in $\mathbb{C}^{2^n}$ is in the 2-dimensional subspace spanned by $|\psi_{\mathsf{init}}\rangle$ and $|\psi_{\mathsf{good}}\rangle$, the* query-controlled amplitude damping channel *with* damping strength *$\lambda \geq 0$, denoted by q-$\mathcal{AD}_\lambda$, is a quantum channel acting on $\mathrm{L}(\mathbb{C}^{2^n} \otimes \mathbb{C}^2)$ that performs the following mapping:*

$$q\text{-}\mathcal{AD}_\lambda[\rho] = E_0 \rho E_0^\dagger + E_1 \rho E_1^\dagger, \tag{6.23}$$

for all $\rho \in \mathrm{D}(\mathbb{C}^{2^n} \otimes \mathbb{C}^2)$. With respect to the basis $\{|r\rangle, |\psi_{\mathsf{good}}\rangle\}$ for the $\mathbb{C}^{2^n}$, E_0 and E_1 are defined as

$$E_0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{-\lambda/2} \end{pmatrix}, \text{ and } E_1 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sqrt{1-e^{-\lambda}} \\ 0 & 0 & 0 & 0 \end{pmatrix}. \tag{6.24}$$

It is easy to verify that the query-controlled amplitude damping channel with damping strength λ performs the following mapping:

$$|\psi\rangle\langle\psi| \otimes \rho \mapsto \begin{cases} |\psi\rangle\langle\psi| \otimes \mathcal{AD}_\lambda[\rho], & \text{when } |\psi\rangle \in \mathrm{G} \\ |\psi\rangle\langle\psi| \otimes \rho, & \text{when } |\psi\rangle \text{ is orthogonal to G.} \end{cases} \tag{6.25}$$

We use the circuit shown in Figure 6.2 to denote the query-controlled amplitude damping channel with damping strength λ.

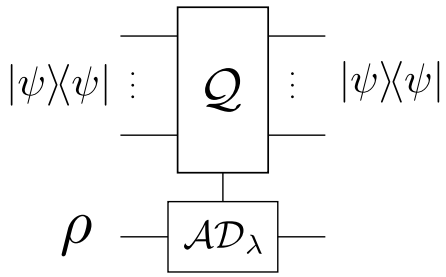

Figure 6.2: The circuit representation of the query-controlled amplitude damping channel with damping strength λ.

In this thesis, we refer to an application of q-$\mathcal{AD}_\lambda$ for any $\lambda > 0$ as a *dissipative query*. The dissipative query and the standard query are closely related: one dissipative query can be implemented by two standard queries together with one controlled amplitude damping channel. We have the following proposition, which is straightforward.

Proposition 6.2. *For all $\lambda > 0$, one application of q-$\mathcal{AD}_\lambda$ can be implemented by two standard queries to Q_{good} and one c-$\mathcal{AD}_\lambda$ by the circuit shown in Figure 6.3.*

A notable property for both c-$\mathcal{AD}_\lambda$ and q-$\mathcal{AD}_\lambda$ is the continuous-time property, which is inherited from $\mathcal{AD}_\lambda$. In particular, it holds that

$$\text{c-}\mathcal{AD}_{\lambda_1} \circ \text{c-}\mathcal{AD}_{\lambda_0} = \text{c-}\mathcal{AD}_{\lambda_0+\lambda_1}, \text{ and} \tag{6.26}$$

$$\text{q-}\mathcal{AD}_{\lambda_1} \circ \text{q-}\mathcal{AD}_{\lambda_0} = \text{q-}\mathcal{AD}_{\lambda_0+\lambda_1}, \tag{6.27}$$

for all $\lambda_0, \lambda_1 \geq 0$. When $\lambda \to 0$, repeatedly applying q-$\mathcal{AD}_\lambda$ yields a continuous-time process, which is Markovian and can be described by the following Lindblad equation

$$\mathcal{L}[\rho] = L\rho L^\dagger - \frac{1}{2}\left(L^\dagger L\rho + \rho L^\dagger L\right). \tag{6.28}$$

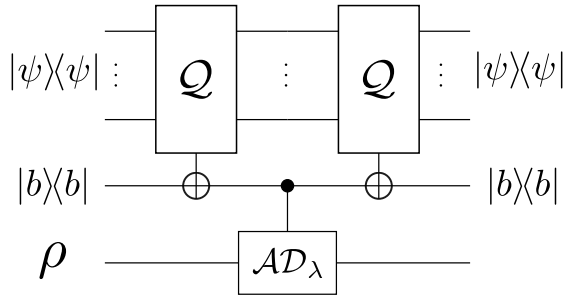

Figure 6.3: Implementing one dissipative query with two standard queries to Q_{good} and one c-$\mathcal{AD}_\lambda$.

When the state of the first register is in the 2-dimensional subspace spanned by $|\psi_{\mathsf{init}}\rangle$ and $|\psi_{\mathsf{good}}\rangle$, with respect to the basis $\{|r\rangle, |\psi_{\mathsf{good}}\rangle\}$ for the first register, the jump operator L is defined as

$$L = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \tag{6.29}$$

Evolving by this Lindblad equation for time λ is equivalent to applying a dissipative query with damping strength λ, i.e.,

$$e^{\mathcal{L}\lambda} = \text{q-}\mathcal{AD}_\lambda. \tag{6.30}$$

6.4 Dissipative quantum search algorithm

The building block of this algorithm is a quantum channel $\mathcal{M}_\lambda$, which is demonstrated in Figure 6.4. The channel $\mathcal{M}_\lambda$ is a composition of two quantum channels q-$\mathcal{AD}_\lambda$, which is defined in Definition 6.4, and c-$\mathcal{G}$, which is the controlled-Grover iterator in the form of a quantum channel. Note that in Definition 6.4, the query-controlled amplitude damping channel with damping strength λ is acting on $\mathrm{L}(\mathbb{C}^{2^n} \otimes \mathbb{C}^2)$, whereas in this section, we swap the two Hilbert spaces so that q-$\mathcal{AD}_\lambda$ is acting on $\mathrm{L}(\mathbb{C}^2 \otimes \mathbb{C}^{2^n})$. This results in a

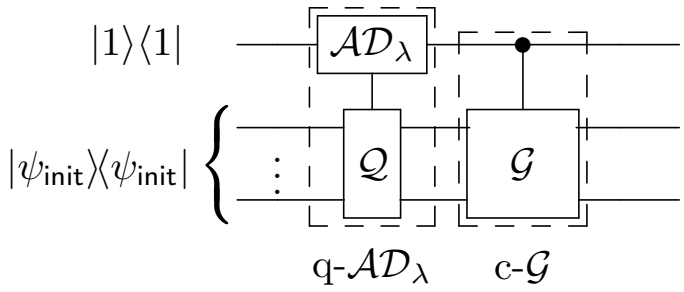

Figure 6.4: The quantum circuit for $\mathcal{M}_\lambda$.

slight change of the Kraus operators:

$$E_{\lambda,0} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{-\lambda/2} \end{pmatrix}, \text{ and } E_{\lambda,1} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sqrt{1-e^{-\lambda}} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \tag{6.31}$$

again, assuming that the state in $\mathbb{C}^{2^n}$ is in the 2-dimensional subspace spanned by $|\psi_{\mathsf{init}}\rangle$ and $|\psi_{\mathsf{good}}\rangle$. Under the same assumption, the matrix representation of the controlled-Grover iterator is

$$\text{c-}G = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos\theta & -\sin\theta \\ 0 & 0 & \sin\theta & \cos\theta \end{pmatrix}. \tag{6.32}$$

The channel c-$\mathcal{G}$ is specified as

$$\text{c-}\mathcal{G}[\rho] = \text{c-}G\rho\text{c-}G^\dagger, \tag{6.33}$$

for all $\rho \in \mathrm{D}(\mathbb{C}^2 \otimes \mathbb{C}^{2^n})$. It follows that the channel $\mathcal{M}_\lambda$ can be specified as

$$\mathcal{M}_\lambda[\rho] = \text{c-}G(E_{\lambda,0}\rho E_{\lambda,0}^\dagger + E_{\lambda,1}\rho E_{\lambda,1}^\dagger)\text{c-}G^\dagger \tag{6.34}$$

$$= \text{c-}G\, E_{\lambda,0}\rho E_{\lambda,0}^\dagger\, \text{c-}G^\dagger + \text{c-}G\, E_{\lambda,1}\rho E_{\lambda,1}^\dagger\, \text{c-}G^\dagger. \tag{6.35}$$

Let $|\psi_{\mathsf{init}}\rangle$ and $|\psi_{\mathsf{good}}\rangle$ be defined as in Definition 6.1. Now, we present the algorithm in Algorithm 6.1.

To show that Algorithm 6.1 has the desired property for Theorem 6.1, we need the following lemma.

Algorithm 6.1: Dissipative quantum search procedure

$\rho \leftarrow |1\rangle\langle 1| \otimes |\psi_{\mathsf{init}}\rangle\langle\psi_{\mathsf{init}}|$;
$\lambda \leftarrow 1$;
$k \leftarrow 1$;
Loop
  $\lambda \leftarrow \lambda/2$;
  for $j \leftarrow k$ *to* $k + 2\lceil \frac{1}{\lambda} \log(\frac{1}{\epsilon}) \rceil$ **do**
    $\rho_{j+1} \leftarrow \mathcal{M}_\lambda[\rho_j]$;
  end
  $k \leftarrow j$;
EndLoop

Lemma 6.3. *Assume $p \leq 1/4$. Let $|\psi\rangle$ be any state in the 2-dimensional subspace spanned by $|\psi_{\mathsf{good}}\rangle$ and $|\psi_{\mathsf{init}}\rangle$ with nonzero overlap with $|\psi_{\mathsf{good}}\rangle$, and let σ be a mixture of $|1\rangle\langle 1| \otimes |\psi\rangle\langle\psi|$ and $|0\rangle\langle 0| \otimes |\psi_{\mathsf{good}}\rangle\langle\psi_{\mathsf{good}}|$. Let $\mathcal{M}_\lambda$ be a quantum channel parameterized by λ defined in Eq. (6.35). If $2\sqrt{p} \leq \lambda \leq 4\sqrt{p}$, then for any $\epsilon > 0$, it holds that*

$$\left\| \mathcal{M}_\lambda^j[\sigma] - |0\rangle\langle 0| \otimes |\psi_{\mathsf{good}}\rangle\langle\psi_{\mathsf{good}}| \right\|_1 \leq \epsilon, \tag{6.36}$$

for all $j > 2\lceil \frac{1}{\sqrt{p}} \log(1/\epsilon) \rceil$.

Before proving Lemma 6.3, we first analyze $\mathcal{M}_\lambda$. Let us first focus on the operator c-$G \cdot E_{\lambda,0}$. To simplify the analysis, we only consider the lower-right block (as the operator c-$G \cdot E_{\lambda,0}$ keeps the first register intact if the state in this register is either $|0\rangle$ or $|1\rangle$). Define the operator A_λ as

$$A_\lambda = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & e^{-\lambda/2} \end{pmatrix}. \tag{6.37}$$

For an angle φ, we use the shorthand $|\varphi\rangle$ to denote the vector $\binom{\cos\varphi}{\sin\varphi}$. For the operator A_λ, we have the following lemma.

Lemma 6.4. *Let $a|\varphi_1\rangle = A_\lambda|\varphi_0\rangle$, where we choose the amplitude $a \geq 0$ to be non-negative, and the angle $\varphi_1 \geq \varphi_0$ to be the smallest possible such angle no smaller than φ_0. Then, the following holds.*

1. *If $0 \leq \lambda \leq 1$, then $a^2 \leq 1 - \frac{\lambda}{4}$ whenever $\frac{\pi}{4} \leq (\varphi_0 \mod \pi) \leq \frac{3\pi}{4}$.*

2. *If* $0 \le \lambda \le 4 \sin \frac{\theta}{2}$, *then* $\frac{\theta}{2} \le \varphi_1 - \varphi_0 \le \frac{3\theta}{2}$.

The main idea of this lemma is to show that the amplitude of the state is contracted by the applications of A_λ. In particular, there exists an interval of φ that permits this contraction (statement 1), and the angle φ rotates with certain speed (statement 2).

Proof of Lemma 6.4. Consider statement 1. Suppose $\frac{\pi}{4} \le (\varphi_0 \mod \pi) \le \frac{3\pi}{4}$. Then we have $|\cos\varphi_0| \le \frac{1}{\sqrt{2}} \le |\sin\varphi_0|$, and hence $a^2 = \cos^2\varphi_0 + e^{-\lambda}\sin^2\varphi_0 \le \frac{1}{2}(1+e^{-\lambda})$. It is easy to verify that the function

$$g(\lambda) = 1 - \frac{\lambda}{4} - \frac{1}{2}(1+e^{-\lambda}) \tag{6.38}$$

is monotonically decreasing in λ. In the interval $[0,1]$, the function g takes its minimum value when $\lambda = 1$, where $g(1) = 1 - \frac{1}{4} - \frac{1}{2}(1+e^{-1}) > 0$. We have

$$\frac{1}{2}(1+e^{-\lambda}) \le 1 - \frac{\lambda}{4}, \tag{6.39}$$

for all $\lambda \in [0,1]$. It follows that $a^2 \le 1 - \frac{\lambda}{4}$, which proves statement 1.

Now, consider statement 2. We first analyze the action of the operator of A_λ. Let φ' be an angle such that

$$\begin{pmatrix} 1 & 0 \\ 0 & e^{-\lambda/2} \end{pmatrix} \begin{pmatrix} \cos\varphi_0 \\ \sin\varphi_0 \end{pmatrix} = \begin{pmatrix} \cos\varphi_0 \\ e^{-\lambda/2}\sin\varphi_0 \end{pmatrix} = a' \begin{pmatrix} \cos\varphi' \\ \sin\varphi' \end{pmatrix}, \tag{6.40}$$

for some $a' > 0$. The last equality implies that $\tan\varphi' = e^{-\lambda/2}\tan\varphi_0$, and therefore

$$\frac{\tan\varphi'}{\tan\varphi_0} = e^{-\lambda/2}. \tag{6.41}$$

By the laws of tangents and sines, we have

$$\frac{\sin(\varphi_0 - \varphi')}{\sin(\varphi_0 + \varphi')} = \frac{1 - e^{-\lambda/2}}{1 + e^{-\lambda/2}} = \tanh\left(\frac{\lambda}{4}\right) \ge 0, \tag{6.42}$$

for all $\lambda \ge 0$. It follows that

$$|\sin(\varphi_0 - \varphi')| \le \tanh\left(\frac{\lambda}{4}\right) |\sin(\varphi_0 + \varphi')| \le \tanh\left(\frac{\lambda}{4}\right). \tag{6.43}$$

By the fact that $\tanh(\lambda) \leq \sin(\lambda)$ for all $\lambda \in [0, 1)$, as shown in [KVV10, Eq. (2.13)], we have

$$|\sin(\varphi_0 - \varphi')| \leq \sin\left(\frac{\lambda}{4}\right). \tag{6.44}$$

By the monotonicity of sine, we obtain

$$|\varphi_0 - \varphi'| \leq \frac{\lambda}{4}. \tag{6.45}$$

Therefore, when $0 \leq \lambda \leq 4\sin(\frac{\theta}{2})$, we have

$$|\varphi_0 - \varphi'| \leq \frac{\theta}{2}. \tag{6.46}$$

Now, let us analyze the action of the second operator of A_λ, which rotates the state by an angle θ. It implies that $\varphi_1 - \varphi' = \theta$. Together with Eq. (6.46), we have

$$\frac{\theta}{2} \leq \varphi_1 - \varphi_0 \leq \frac{3\theta}{2}, \tag{6.47}$$

which concludes statement 2. □

In the following, we use Lemma 6.4 to prove Lemma 6.3.

Proof of Lemma 6.3. To understand the impact of repeatedly applying $\mathcal{M}_\lambda$, we consider the impact of applying the operators c-$G \cdot E_{\lambda,0}$ and c-$G \cdot E_{\lambda,1}$ separately.

For any state σ specified in the lemma, applying $E_{\lambda,1}$ on it yields $|0\rangle\langle 0| \otimes |\psi_{\mathsf{good}}\rangle\langle\psi_{\mathsf{good}}|$, which will be invariant for any subsequent operations such as c-G, $E_{\lambda,0}$, and E_{λ_1}. Applying c-$G \cdot E_{\lambda,0}$ on σ results in some state σ_1, which is a mixture of $|1\rangle\langle 1| \otimes |\psi_1\rangle\langle\psi_1|$ and $|0\rangle\langle 0| \otimes |\psi_{\mathsf{good}}\rangle\langle\psi_{\mathsf{good}}|$, where $|\psi_1\rangle$ is some state in the 2-dimensional subspace spanned by $|\psi_{\mathsf{good}}\rangle$ and $|\psi_{\mathsf{init}}\rangle$. After j application of $\mathcal{M}_\lambda$, the resulting state σ_j will always be a mixture of $|1\rangle\langle 1| \otimes |\psi_j\rangle\langle\psi_j|$ and $|0\rangle\langle 0| \otimes |\psi_{\mathsf{good}}\rangle\langle\psi_{\mathsf{good}}|$, where $|\psi_j\rangle$ is a state in the 2-dimensional subspace spanned by $|\psi_{\mathsf{good}}\rangle$ and $|\psi_{\mathsf{init}}\rangle$. To bound the distance between $\mathcal{M}_\lambda^j[\sigma]$ and $|0\rangle\langle 0| \otimes |\psi_{\mathsf{good}}\rangle\langle\psi_{\mathsf{good}}|$, it suffices to bound the amplitude of $(\text{c-}G \cdot E_{\lambda,0})^j|1\rangle|\psi\rangle$, where $|\psi\rangle$ can be any state in the 2-dimensional subspace spanned by $|\psi_{\mathsf{good}}\rangle$ and $|\psi_{\mathsf{init}}\rangle$. This can be demonstrated in Figure 6.5.

The impact of the operator c-$G \cdot E_{\lambda,0}$ is shown in Lemma 6.4. Note that both c-G and $E_{\lambda,0}$ do not affect the first register if the state in this register is either $|0\rangle$ or $|1\rangle$, so it

suffices to consider the operator A_λ defined in Eq. (6.37). Take $2\sqrt{p} = 2\sin(\theta/2) \leq \lambda \leq 4\sin(\theta/2) = 4\sqrt{p}$. By statement 1 of Lemma 6.4, it is guaranteed that the amplitude of the state will be contracted by $1 - \lambda/2 = 1 - \sqrt{p}/2$ when $\frac{\pi}{4} \leq (\varphi_0 \mod \pi) \leq \frac{3\pi}{4}$. When $p \leq 1/4$, we have

$$0.3 < (1 - \sqrt{p}/2)^{\frac{2}{\sqrt{p}}} \leq \frac{1}{e}. \tag{6.48}$$

As the interval $[\frac{\pi}{4}, \frac{3\pi}{4}]$ is half of the interval $[0, \pi]$, to ensure the desired constant-contraction, the number of iterations should be $4/\sqrt{p}$. Statement 2 of Lemma 6.4 ensures that the rotation speed of φ is at least $\sqrt{p}/2$, which implies that even if the algorithm is started with some $\varphi < \frac{\pi}{4}$ or $\varphi > \frac{3\pi}{4}$, with an additional $\sqrt{p}/2$ iterations, the angle φ will be in the interval $[\frac{\pi}{4}, \frac{3\pi}{4}]$. Therefore, when $j > 4\lceil \frac{1}{\sqrt{p}} \log(1/\epsilon) \rceil$, the state is contracted by at least $\sqrt{\epsilon}$, and the trace-distance will be upper-bounded by ϵ, which proves the lemma. □

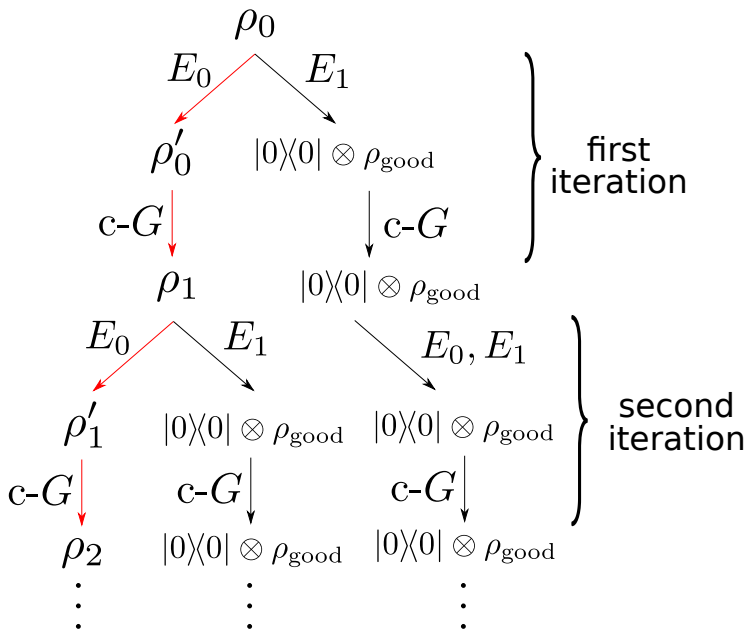

Figure 6.5: Demonstration of the impact of E_0 and E_1 in each iteration.

In Figure 6.6, we show a numerical comparison between the algorithm in Lemma 6.3 and Grover's algorithm. This numerical comparison illustrates that the error converges (though not monotonically) with more iterations of $\mathcal{M}_\lambda$.

Now, we are ready to prove Theorem 6.1.

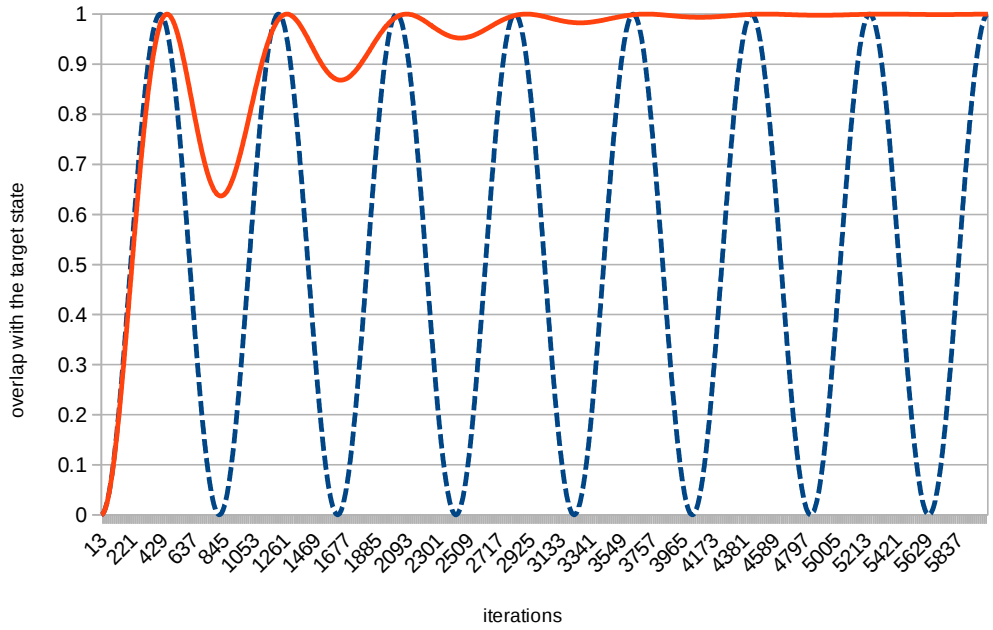

Figure 6.6: Numerical comparison between the algorithm given in Lemma 6.3 (solid) and Grover's algorithm (dashed). The horizontal axis represents the number of iterations (of $\mathcal{M}_\lambda$ and the Grover iterator, respectively), and the vertical axis represents the overlap of the current state with the target state. The size of search space is $N = 2^{18}$ and there is only one marked item.

Proof of Theorem 6.1. We use Algorithm 6.1. For each trial of λ, if $2\sqrt{p} \le \lambda \le 4\sqrt{p}$, then by Lemma 6.3, the error bound is satisfied. Otherwise, the resulting state is in the form of σ specified in Lemma 6.3, and this state can be the input state of the iterations for the next trial of λ. To obtain error bound ϵ, the total number of iterations is

$$4\lceil 2\log(1/\epsilon)\rceil + 4\lceil 4\log(1/\epsilon)\rceil + \cdots + 4\lceil \frac{1}{\lambda'}\log(1/\epsilon)\rceil \le \frac{8}{\lambda'}\log(1/\epsilon). \tag{6.49}$$

Since $2\sqrt{p} \le \lambda' \le 4\sqrt{p}$, the above quantity is upper-bounded by $\frac{4}{\sqrt{p}}\log(1/\epsilon)$. The fact that each application of $\mathcal{M}_\lambda$ can be implemented using a constant number of queries to Q_{good} and Q_{init} follows from Proposition 6.2. □

The numerical comparison between Algorithm 6.1 and Grover's algorithm is shown in Figure 6.7. Note that the error convergence is not monotonic as it appears in the figure. There exits tiny increasing of error during the applications of $\mathcal{M}_\lambda$ that cannot be observed due to the limited scale of this plot.

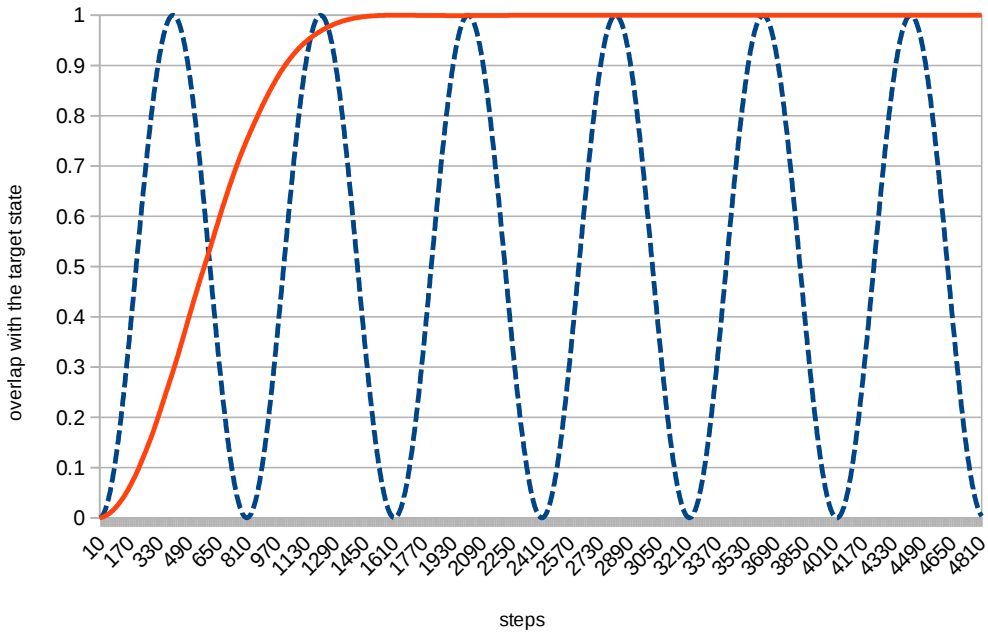

Figure 6.7: Numerical comparison between Algorithm 6.1 (solid) and Grover's algorithm (dashed). The horizontal axis represents the number of applications of step 6 in Algorithm 6.1 and the Grover iterator in Grover's algorithm, respectively, and the vertical axis represents the overlap of the current state with the target state. The size of search space is $N = 2^{18}$ and there is only one marked item.

Chapter 7

Conclusion

In this thesis, we studied computational problems related to open quantum systems. In Chapter 2, we provided the necessary background for this thesis, including the terminologies and notations for quantum information and some properties of finite fields. In Chapter 3, we studied an example of open quantum systems – unitary 2-designs. We gave three constructions of unitary 2-designs on n qubits. The first construction requires $O(n \log n \log\log n)$ Clifford gates, and it works for infinitely many n under the extended Riemann Hypothesis. The second construction requires $O(n \log n \log\log n)$ gates (including non-Clifford gates), unconditionally for all n. The third construction requires $O(n \log^2 n \log\log n)$ Clifford gates, unconditionally for all n. In Chapter 4, we studied the Lindblad evolution, which captures the dynamics of Markovian open quantum systems. We first gave a macroscopic derivation of the Lindblad equation. Then, we showed a lower bound for simulating Lindblad evolution as Hamiltonian evolution in a larger Hilbert space. In Chapter 5, we presented an efficient quantum algorithm for simulating Lindblad evolution for time t and error ϵ with gate complexity (in a simplified form) $O(t \operatorname{polylog}(t/\epsilon))$, which cannot be achieved if the Lindblad evolution is simulated as Hamiltonian evolution in a larger Hilbert space (because of the lower bound shown in Chapter 4). In Chapter 6, we demonstrated that open quantum systems can be used as building blocks for other quantum algorithms. In particular, we introduced a novel query model – the dissipative query model, which can be implemented by the standard query model with the amplitude damping process. With this dissipative query model, we showed a fixed-point quantum search algorithm that preserves the quadratic speedup.

References

[AG04] Scott Aaronson and Daniel Gottesman. Improved simulation of stabilizer circuits. *Physical Review A*, 70(5), 2004.

[ATS03] Dorit Aharonov and Amnon Ta-Shma. Adiabatic quantum state generation and statistical zero knowledge. In *Proceedings of the Thirty-fifth ACM Symposium on Theory of Computing - STOC '03*. ACM Press, 2003.

[BBHT98] Michel Boyer, Gilles Brassard, Peter Høyer, and Alain Tapp. Tight bounds on quantum searching. *Fortschritte der Physik: Progress of Physics*, 46(4-5):493–505, 1998.

[BBP+96] Charles H. Bennett, Gilles Brassard, Sandu Popescu, Benjamin Schumacher, John A. Smolin, and William K. Wootters. Purification of noisy entanglement and faithful teleportation via noisy channels. *Physical Review Letters*, 76(5):722–725, 1996.

[BCC+15] Dominic W. Berry, Andrew M. Childs, Richard Cleve, Robin Kothari, and Rolando D. Somma. Simulating Hamiltonian dynamics with a truncated Taylor series. *Physical Review Letters*, 114(9), 2015.

[BCC+17] Dominic W. Berry, Andrew M. Childs, Richard Cleve, Robin Kothari, and Rolando D. Somma. Exponential improvement in precision for simulating sparse Hamiltonians. *Forum of Mathematics, Sigma*, 5, 2017.

[BCG14] Dominic W. Berry, Richard Cleve, and Sevag Gharibian. Gate-efficient discrete simulations of continuous-time quantum query algorithms. *Quantum Information & Computation*, 14(1-2):1–30, 2014.

[BCK15] Dominic W. Berry, Andrew M. Childs, and Robin Kothari. Hamiltonian simulation with nearly optimal dependence on all parameters. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*. IEEE, 2015.

[BDSW96] Charles H. Bennett, David P. DiVincenzo, John A. Smolin, and William K. Wootters. Mixed-state entanglement and quantum error correction. *Physical Review A*, 54(5):3824–3851, 1996.

[BF13] Winton Brown and Omar Fawzi. Short random circuits define good quantum error correcting codes. In *2013 IEEE International Symposium on Information Theory*. IEEE, 2013.

[BF15] Winton Brown and Omar Fawzi. Decoupling with random quantum circuits. *Communications in Mathematical Physics*, 340(3):867–900, 2015.

[BH97] Gilles Brassard and Peter Høyer. An exact quantum polynomial-time algorithm for Simon's problem. In *Theory of Computing and Systems, 1997., Proceedings of the Fifth Israeli Symposium on*, pages 12–23. IEEE, 1997.

[BHMT02] Gilles Brassard, Peter Høyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. *Contemporary Mathematics*, 305:53–74, 2002.

[BHT98] Gilles Brassard, Peter Høyer, and Alain Tapp. Quantum counting. In *Proceedings of the 25th International Colloquium on Automata, Languages, and Programming - ICALP '98*, pages 820–831. Springer, 1998.

[BN16] Dominic W. Berry and Leonardo Novo. Corrected quantum walk for optimal Hamiltonian simulation. *arXiv preprint arXiv:1606.03443*, 2016.

[BP07] Heinz-Peter Breuer and Francesco Petruccione. *The Theory of Open Quantum Systems*. Oxford University Press, 2007.

[Cha05] Hoi Fung Chau. Unconditionally secure key distribution in higher dimensions by depolarization. *IEEE Transactions on Information Theory*, 51(4):1451–1468, 2005.

[Chi04] Andrew M. Childs. *Quantum Information Processing in Continuous Time*. PhD thesis, Massachusetts Institute of Technology, 2004.

[Cho75] Man-Duen Choi. Completely positive linear maps on complex matrices. *Linear Algebra and its Applications*, 10(3):285–290, 1975.

[CK10] Andrew M. Childs and Robin Kothari. Limitations on the simulation of non-sparse Hamiltonians. *Quantum Information & Computation*, 10(7&8):0669–0684, 2010.

[CL17] Andrew M. Childs and Tongyang Li. Efficient simulation of sparse markovian quantum dynamics. *Quantum Information & Computation*, 17(11&12):0901–0947, 2017.

[CLLW16] Richard Cleve, Debbie W. Leung, Li Liu, and Chunhao Wang. Near-linear constructions of exact unitary 2-designs. *Quantum Information & Computation*, 16(9-10):721–756, 2016.

[CPdC+15] Roberto Di Candia, Julen S. Pedernales, Adolfo del Campo, Enrique Solano, and Jorge Casanova. Quantum simulation of dissipative processes without reservoir engineering. *Scientific Reports*, 5(1), 2015.

[CRR05] Sourav Chakraborty, Jaikumar Radhakrishnan, and Nandakumar Raghunathan. Bounds for error reduction with few quantum queries. In *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques*, pages 245–256. Springer, 2005.

[CW79] John L. Carter and Mark N. Wegman. Universal classes of hash functions. *Journal of Computer and System Sciences*, 18(2):143–154, 1979.

[CW17] Richard Cleve and Chunhao Wang. Efficient quantum algorithms for simulating Lindblad evolution. In *Proceedings of the 44th International Colloquium on Automata, Languages, and Programming - ICALP '17*, pages 17:1–17:14. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2017.

[DCEL09] Christoph Dankert, Richard Cleve, Joseph Emerson, and Etera Livine. Exact and approximate unitary 2-designs and their application to fidelity estimation. *Physical Review A*, 80(1), 2009.

[Dev05] Igor Devetak. The private classical capacity and quantum capacity of a quantum channel. *IEEE Transactions on Information Theory*, 51(1):44–55, 2005.

[DGV12] Ross Dorner, John Goold, and Vlatko Vedral. Towards quantum simulations of biological information flow. *Interface Focus*, 2(4):522–528, 2012.

[DLT02] David P. DiVincenzo, Debbie W. Leung, and Barbara M. Terhal. Quantum data hiding. *IEEE Transactions on Information Theory*, 48(3):580–598, 2002.

[EWS+03] Joseph Emerson, Yaakov S. Weinstein, Marcos Saraceno, Seth Lloyd, and David G. Cory. Pseudo-random unitary operators for quantum information processing. *Science*, 302(5653):2098–2100, 2003.

[Fey82] Richard P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6-7):467–488, 1982.

[FG98] Edward Farhi and Sam Gutmann. Analog analogue of a digital quantum computation. *Physical Review A*, 57(4):2403–2406, 1998.

[Fin47] Nathan J. Fine. Binomial coefficients modulo a prime. *The American Mathematical Monthly*, 54(10):589, 1947.

[GAE07] David Gross, Koenraad Audenaert, and Jens Eisert. Evenly distributed unitaries: on the structure of unitary designs. *Journal of Mathematical Physics*, 48(5):052104, 2007.

[GKS76] Vittorio Gorini, Andrzej Kossakowski, and Ennackal C. G. Sudarshan. Completely positive dynamical semigroups of n-level systems. *Journal of Mathematical Physics*, 17(5):821–825, 1976.

[Got97] Daniel Gottesman. *Stabilizer Codes and Quantum Error Correction*. PhD thesis, California Institute of Technology, 1997. arXiv preprint quant-ph/9705052.

[Got98] Daniel Gottesman. Theory of fault-tolerant quantum computation. *Physical Review A*, 57(1):127–137, 1998.

[Got99] Daniel Gottesman. The Heisenberg representation of quantum computers. In *Proceedings of the XXII International Colloquium on Group Theoretical Methods in Physics*, pages 32–43, Cambridge, MA, USA, 1999. International Press.

[GPT06] Lov K. Grover, Apoorva Patel, and Tathagat Tulsi. Quantum algorithms with fixed points: the case of database search, 2006. arXiv preprint quant-ph/0603132.

[Gro96] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-eighth ACM Symposium on Theory of Computing - STOC '96*. ACM Press, 1996.

[Gro05] Lov K. Grover. Fixed-point quantum search. *Physical Review Letters*, 95(15), 2005.

[GvzGPS00] Shuhong Gao, Joachim von zur Gathen, Daniel Panario, and Victor Shoup. Algorithms for exponentiation in finite fields. *Journal of Symbolic Computation*, 29(6):879–889, 2000.

[Has09] Matthew B. Hastings. Superadditivity of communication capacity using entangled inputs. *Nature Physics*, 5(4):255–257, 2009.

[Haz96] Michiel Hazewinkel. *Handbook of Algebra*, volume 1. Elsevier, 1996.

[HHWY08] Patrick Hayden, Michał Horodecki, Andreas Winter, and Jon Yard. A decoupling approach to the quantum capacity. *Open Systems & Information Dynamics*, 15(01):7–19, 2008.

[HL09] Aram W. Harrow and Richard A. Low. Random quantum circuits are approximate 2-designs. *Communications in Mathematical Physics*, 291(1):257–302, 2009.

[HLSW04] Patrick Hayden, Debbie W. Leung, Peter W. Shor, and Andreas Winter. Randomizing quantum states: constructions and applications. *Communications in Mathematical Physics*, 250(2):371–391, 2004.

[HLW06] Patrick Hayden, Debbie W. Leung, and Andreas Winter. Aspects of generic entanglement. *Communications in Mathematical Physics*, 265(1):95–117, 2006.

[HP13] Susana F. Huelga and Martin B. Plenio. Vibrations, quanta and biology. *Contemporary Physics*, 54(4):181–207, 2013.

[HW08] Patrick Hayden and Andreas Winter. Counterexamples to the maximal p-norm multiplicativity conjecture for all p > 1. *Communications in Mathematical Physics*, 284(1):263–280, 2008.

[KB16] Michael J. Kastoryano and Fernando G. S. L. Brandão. Quantum Gibbs samplers: the commuting case. *Communications in Mathematical Physics*, 344(3):915–957, 2016.

[KBD+08] Barbara Kraus, Hans P. Büchler, Sebastian Diehl, Adrian Kantian, Andrea Micheli, and Peter Zoller. Preparation of entangled states by quantum Markov processes. *Physical Review A*, 78(4), 2008.

[KBG+11] Martin Kliesch, Thomas Barthel, Christian Gogolin, Michael J. Kastoryano, and Jens Eisert. Dissipative quantum Church-Turing theorem. *Physical Review Letters*, 107(12), 2011.

[Kot14] Robin Kothari. *Efficient Algorithms in Quantum Query Complexity*. PhD thesis, University of Waterloo, 2014.

[KRS11] Michael J. Kastoryano, Florentin Reiter, and Anders S. Sørensen. Dissipative preparation of entanglement in optical cavities. *Physical Review Letters*, 106(9), 2011.

[KVV10] Riku Klén, M. Visuri, and Matti Vuorinen. On Jordan type inequalities for hyperbolic functions. *Journal of Inequalities and Applications*, 2010(1):362548, 2010.

[LC17] Guang Hao Low and Isaac L. Chuang. Optimal Hamiltonian simulation by quantum signal processing. *Physical Review Letters*, 118(1), 2017.

[LCD+87] Anthony J. Leggett, Sudip Chakravarty, Alan T. Dorsey, Matthew P. A. Fisher, Anupam Garg, and Wilhelm Zwerger. Dynamics of the dissipative two-state system. *Reviews of Modern Physics*, 59(1):1–85, 1987.

[Leu03] Debbie W. Leung. Choi's proof as a recipe for quantum process tomography. *Journal of Mathematical Physics*, 44(2):528, 2003.

[Lin76] Goran Lindblad. On the generators of quantum dynamical semigroups. *Communications in Mathematical Physics*, 48(2):119–130, 1976.

[Llo96] Seth Lloyd. Universal quantum simulators. *Science*, 273(5278):1073–1078, 1996.

[Llo97] Seth Lloyd. Capacity of the noisy quantum channel. *Physical Review A*, 55(3):1613–1622, 1997.

[LN94] Rudolf Lidl and Harald Niederreiter. *Introduction to Finite Fields and their Applications*. Cambridge University Press, 1994.

[Low09] Richard A. Low. *Pseudo-randomness and Learning in Quantum Computation*. PhD thesis, University of Bristol, 2009. arXiv preprint arXiv:1006.5227.

[MK08] Volkhard May and Oliver Kühn. *Charge and Energy Transfer Dynamics in Molecular Systems*. John Wiley & Sons, 2008.

[MPGC13] Easwar Magesan, Daniel Puzzuoli, Christopher E. Granade, and David G. Cory. Modeling quantum noise for efficient testing of fault-tolerant circuits. *Physical Review A*, 87(1), 2013.

[MR95] Rajeev Motwani and Prabhakar Raghavan. *Randomized Algorithms*. Cambridge University Press, 1995.

[MRE+12] Sarah Mostame, Patrick Rebentrost, Alexander Eisfeld, Andrew J Kerman, Dimitris I. Tsomokos, and Alán Aspuru-Guzik. Quantum simulator of an open quantum system using superconducting qubits: exciton transport in photosynthetic complexes. *New Journal of Physics*, 14(10):105013, 2012.

[MW05] Chris Marriott and John Watrous. Quantum Arthur–Merlin games. *Computational Complexity*, 14(2):122–152, 2005.

[NC00] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

[Nit06] Abraham Nitzan. *Chemical Dynamics in Condensed Phases: Relaxation, Transfer and Reactions in Condensed Molecular Systems*. Oxford University Press, 2006.

[PP17] Apoorva Patel and Anjani Priyadarsini. Optimization of quantum Hamiltonian evolution: from two projection operators to local Hamiltonians. *International Journal of Quantum Information*, 15(02):1650027, 2017.

[Pre] John Preskill. Lecture notes for Quantum Computation. Available online at http://theory.caltech.edu/people/preskill/ph229/, visited on 2018-06-15.

[Rid] Larry Riddle. Unpublished manuscript. Available online at http://ecademy.agnesscott.edu/~lriddle/ifs/siertri/Pascalmath.htm, visited on 2018-06-15.

[RRS16] Florentin Reiter, David Reeb, and Anders S Sørensen. Scalable dissipative preparation of many-body entanglement. *Physical Review Letters*, 117(4), 2016.

[RS09] Aidan Roy and A. J. Scott. Unitary designs and codes. *Designs, Codes and Cryptography*, 53(1):13–31, 2009.

[Sch77] Arnold Schönhage. Schnelle multiplikation von polynomen über körpern der charakteristik 2. *Acta Informatica*, 7(4):395–398, 1977.

[SDTR13] Oleg Szehr, Frédéric Dupuis, Marco Tomamichel, and Renato Renner. Decoupling with unitary approximate two-designs. *New Journal of Physics*, 15(5):053022, 2013.

[Sho] Peter W. Shor. The quantum channel capacity and coherent information. Lecture notes, MSRI Workshop on Quantum Computation, 2002. Available online at http://www.msri.org/realvideo/ln/msri/2002/quantumcrypto/shor/1/, visited on 2018-06-15.

[Suz91] Masuo Suzuki. General theory of fractal path integrals with applications to many-body theories and statistical physics. *Journal of Mathematical Physics*, 32(2):400–407, 1991.

[VWC09] Frank Verstraete, Michael M. Wolf, and Juan Ignacio Cirac. Quantum computation and quantum-state engineering driven by dissipation. *Nature Physics*, 5(9):633–636, 2009.

[vzGG13] Joachim von zur Gathen and Jürgen Gerhard. *Modern Computer Algebra*. Cambridge University Press, 2013.

[vzGP01] Joachim von zur Gathen and Francesco Pappalardi. Density estimates related to Gauß periods. In *Cryptography and Computational Number Theory*, pages 33–41. Birkhuser Basel, 2001.

[vzGSS07] Joachim von zur Gathen, Amin Shokrollahi, and Jamshid Shokrollahi. Efficient multiplication using type 2 optimal normal bases. In *Arithmetic of Finite Fields*, pages 55–68. Springer Berlin Heidelberg, 2007.

[Wat18] John Watrous. *The Theory of Quantum Information*. Cambridge University Press, 2018.

[Wei12] Ulrich Weiss. *Quantum Dissipative Systems*. World Scientific, 2012.

[YLC14] Theodore J. Yoder, Guang Hao Low, and Isaac L. Chuang. Fixed-point quantum search with an optimal number of queries. *Physical Review Letters*, 113(21), 2014.