Approaches to Practical Quantum Cryptography

Peter Yuen

Thesis submitted to the University of Ottawa
in partial fulfillment of the requirements for the degree of
Doctorate in Philosophy Mathematics and Statistics[1]

Department of Mathematics and Statistics
Faculty of Science
University of Ottawa

[1]The Ph.D. program is a joint program with Carleton University, administered by the Ottawa-Carleton Institute of Mathematics and Statistics

Abstract

There are various practical obstacles in quantum cryptography. These include the problems that arise in implementing quantum cryptographic protocols with current and near-term technology, and those that come with the development and security analysis of new protocols. In this work, we examine a few of these obstacles and, for each one, we highlight an approach that aims to mitigate it.

We start by examining the setting of device-independence, with oblivious transfer in the bounded-quantum-storage-model as the primitive of interest. The typical approach to enabling device-independence is the use of a Bell-inequality violation, which requires a strict non-communication assumption that is practically difficult to enforce. We develop and analyze the security of a device-independent oblivious transfer protocol that replaces the Bell-inequality violation approach with a computational assumption, thereby relaxing the non-communication assumption.

Our second topic examines the problem of noise in a quantum money scheme. We show how to achieve noise-tolerance in the public-key setting. Our scheme can be seen as a generalization of the [AC12] scheme: a valid banknote is now a subspace state possibly affected by noise, and verification is performed by using classical oracles to check for membership in "larger spaces." Additionally, a banknote in our scheme is minted by preparing conjugate coding states and applying a unitary.

For our last topic, we consider the general task of proving security against a specious adversary, which is the quantum analogue of the classical semi-honest adversary. A useful feature of specious adversaries is that security against them can guarantee security against stronger adversaries [DNS12]. This motivates the search for a canonical specious adversary, where proving security against such an adversary guarantees security against the entire class of specious adversaries. Within a restricted setting, we define a particular specious adversary and show that it satisfies our definition of canonical.

Acknowledgements

I would first like to thank my supervisor, Dr. Anne Broadbent. I have learned a considerable amount under her mentorship, and the opportunities she gave me have deeply enriched my studies. Additionally, her patience and unwavering support have been invaluable in some of the more difficult periods of this work.

I also thank everyone in our research group, both past and present. The list is long, but they have each contributed towards making the past several years some of the most enjoyable of my education. I will fondly remember the academic events we attended together, our group activities, and the discussions we had in the office (and, for a couple years, online).

I would like to acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), the US Air Force Office of Scientific Research under award number FA9550-20-1-0375, and the University of Ottawa's Research Chairs program.

I thank the examiners, Dr. Michel Barbeau, Dr. Arthur Mehta, Dr. Monica Nevins, and Dr. Or Sattath; their valuable feedback has improved the quality of this thesis.

I thank my partner, Kelly, for always being there for me. It is difficult to imagine how any of this could be possible without her support, which in itself is impossible to articulate.

Lastly, I would like to thank my family. I thank my mother for supporting me at every step and for always being ready to answer the phone. I thank my brother for always showing a keen interest in my work. And I thank my father, who encouraged me to start this journey and would have been happy to see me complete it.

List of publications

A list of publications completed during the author's PhD. Chapter 4 of this thesis is based on [BY23], and Chapter 5 is based on [Yue25].

[BY23] A. Broadbent and P. Yuen. Device-independent oblivious transfer from the bounded-quantum-storage-model and computational assumptions. *New Journal of Physics*, 25: 053019, 2023.
DOI: 10.1088/1367-2630/accf32

[Yue25] P. Yuen. Noise-tolerant public-key quantum money from a classical oracle. *Quantum*, 9: 1691, 2025.
DOI: 10.22331/q-2025-04-07-1691

Contents

List of Figures

Chapter 1

Introduction

The fields of quantum computing and quantum cryptography are relatively young, with their beginnings approximately set in the 1980s, though it has taken little time for them to garner immense intrigue that encompasses not only academics but also governments, the tech industry, and the public. The source for this interest can be attributed to various reasons. In some cases, it is the promise that quantum computing can be used as a powerful tool to solve difficult problems in other sciences like biology [LCC+13, KBD+21] and finance [HGL+23]; in other cases, it stems from pure fascination; and in perhaps the most highlighted case, it is the implications that quantum technologies have on every level of security, from the day-to-day privacy on which we personally depend to the national security of any given country. In this latter case, the motivation to pursue quantum technologies is almost dire.

Regardless of the motivation, it is clear that there is a strong effort to realize viable quantum technologies in the near future, but in working towards this goal, it has become apparent that there are difficult obstacles to overcome. Quantum protocols for accomplishing tasks must now account for the factors that come with operating in the real world and, in some instances, they must be tailored to account for our current technological limitations. Such modifications must be done carefully; any change to a cryptographic protocol, for instance, can affect the security guarantee, and thus careful analysis is required.

To understand these obstacles, it is first necessary to get a better sense of how quantum physics distinguishes itself from classical physics. At the simplest level, quantum computing and quantum cryptography often rely on the idea of encoding information into a *quantum state*. Quantum states can exhibit several bizarre features that can be exploited in remarkably useful ways. As an example, quantum states can be *measured* as a means of extracting the information encoded in them; however, to successfully retrieve this information, one must know the "right way" to measure the state, since measuring in the wrong way can cause the state to *collapse*, resulting in an irreversible loss of information.

Another fundamental feature is the *no-cloning theorem* [Die82, WZ82] which, informally, states that a quantum state cannot be cloned. This is radically different from classical physics where it is always possible to make a copy of some classical information. Then there is the remarkable phenomenon where two quantum states may become *entangled* in the sense that a change to one quantum state *instantaneously* affects the state entangled with it. This behaviour famously puzzled Einstein, coining the almost commonplace phrase "spooky action at a distance" [EBB71].

All of these oddities can be viewed as limitations to using quantum physics for accomplishing typical tasks but, from another perspective, these oddities are rather useful features. Each of them can be cleverly exploited to accomplish tasks that are *impossible* with classical physics. We see the first exploitation of the no-cloning theorem in Wiesner's paper [Wie83] (which had actually been written almost a decade earlier) where he proposed the idea of *quantum money*. Intuitively, by treating a quantum state as a banknote, it becomes unforgeable as a result of the no-cloning theorem. In [BB84], similar techniques are used to construct a protocol for *quantum key distribution*: a quantum analog of the cryptographic task of distributing a secret key for the purpose of secure communication. In this latter application, the use of quantum physics gives us a security guarantee that is stronger than what we get from any classical protocol.

In the time that followed [Wie83, BB84], a wealth of exciting quantum cryptographic protocols were developed. Now, as experimentalists seek to realize these protocols, and as theorists seek to develop and prove the security of new protocols, we find, among the various obstacles, problems that have a practical nature to them. The goal of this thesis is to highlight just a few of these practical problems alongside a technique for mitigating them. We do this by examining each problem-and-technique through the lens of some topic: device-independence, quantum money, and specious adversaries.

For the remainder of this section, we discuss each of these topics at a high level so that we can describe our contributions without requiring prerequisite material. A detailed introduction to each topic can be found at the beginning of the associated chapter.

1.1 Device-independence

We start by examining the topic of *device-independence* where the goal is to execute a protocol on a device without having to "trust it." This concept is motivated by the following question. If a device is sold to a user, how can they verify that it is functioning correctly without opening it up? It is reasonable to want to verify the proper functionality of a device, especially if we suspect that the device may be faulty or that the manufacturer has maliciously tampered with the device. For simple devices, or for an advanced user, this can easily be done by opening up a

device to inspect its components, but for a complex, delicately designed device (like a quantum device) the average user cannot take this approach. Thus, the task is to verify the device through classical interactions alone; that is, by using classical inputs and outputs (*i.e.*, pressing buttons and reading off outputs from a display). It is impossible to verify a classical device in this way, but with a quantum device, there are several strategies available.

For a quantum device, what we actually seek to verify, broadly, is that the device is genuinely using entangled quantum states to achieve its task, rather than deceiving us with a classical simulation. The typical technique employed to verify this is to test the device by playing a "game" with it. The device is said to win the game if some condition involving the classical inputs and outputs is met. For certain games, it has been shown that a quantum device's winning probability is provably higher than the winning probability of a classical device, where the advantage comes from using entangled states in its strategy. This approach is typically formulated as the violation of an inequality, called a *Bell inequality*. When a Bell inequality is violated, we can, in theory, conclude that entangled quantum states are present, which is precisely what we sought to verify.

In practice, drawing this final conclusion (that a Bell inequality violation implies the presence of something quantum) is host to a variety of problems referred to as loopholes. Unless these loopholes are closed, one cannot confidently say that an entangled quantum state was used by the device, but closing these loopholes can come with their own challenges. One loophole which we focus on is the *locality loophole*, which describes the situation where a classical device with two components *can* violate a Bell inequality if the two components are able to communicate. To close the loophole, one must enforce non-communication by either separating the components with a large distance (so that it takes less time to interact with the device than for a signal of light to travel between components) or by perfectly shielding the components of the device. Either approach is practically challenging to achieve, and so it is natural to ask if there is an entirely different approach to testing a device which circumvents these loopholes.

If we utilize a post-quantum computational assumption, then such an approach does, in fact, exist. By this, we mean a mathematical problem that is believed to be computationally difficult even for quantum computers. With this assumption, we can say that the device cannot solve the problem during the execution of the protocol; then, by leveraging this fact, one can test the device. This line of work began in the foundational work of [Mah18, BCM+18] and was then further developed in [MDCA21, MV21], the latter of which we rely on in this thesis. The major advantage here is that this computational assumption replaces the Bell inequality violation approach, thus eliminating the locality loophole right from the start. In fact, the components of the device can communicate, to some degree.

In this thesis, we apply this approach to a cryptographic primitive known as *obliv-*

ious transfer. This simple primitive is powerful in that any two-party cryptographic functionality can be constructed from secure oblivious transfer [Kil88]. Unfortunately, secure oblivious transfer is actually impossible, even with quantum resources [May97, LC97]. To construct oblivious transfer, some assumption must be made, and so we follow [DFR+07] by making the reasonable assumption that one of the two parties has bounded quantum storage. Given the trajectory of quantum technology, this assumption is highly reasonable and it achieves a powerful notion of security known as *everlasting security* (if the security of the protocol is not broken *during* the protocol's execution, it is secure for all time afterwards). So, within this model of oblivious transfer, we apply this testing technique, based off a post-quantum computational assumption, to achieve a protocol for device-independent oblivious transfer without a strict non-communication assumption.

1.2 Quantum money

We use the topic of quantum money to discuss what is arguably the most well-known practical problem in developing quantum technologies: noise. Quantum states are highly sensitive to their environment; factors like heat and vibrations can alter them in unexpected, undesirable ways. When this occurs, we say that the quantum state has been affected by noise (*i.e.*, the state has incurred some errors). In the context of this thesis, we care less about the source of noise and more so on how noise is handled in general.

Noise is also present in classical computing, though its manifestation and the way it is handled is simpler than the quantum case. Without loss of generality, suppose we have a single bit of data, $x = 0$. The only type of classical error that can occur is a "bit-flip" which turns a 0 to a 1 and vice versa. To protect against such errors, we could make several copies of our data, $x_L \coloneqq (x, x, x) = (0, 0, 0)$. Now let some error E affect x_L which causes the first bit to flip, $Ex_L = (1, 0, 0)$. By taking a "majority vote," we can then infer that the first bit should be 0 and so we can correct the error by flipping the first bit back to 0. This process is a simple example of an *error correcting code*. A crucial observation here is that, by virtue of the data being classical, the error correcting code can heavily rely on the ability to make copies of the data. In the quantum case, the no-cloning theorem immediately prohibits this approach, and thus quantum error correction, though possible, requires a much more delicate approach.

The goal of this section is to apply quantum error correction to the task of quantum money. As noted earlier, the goal of quantum money is to produce verifiable, unforgeable banknotes. Noise-tolerance is especially relevant here because in any practical real-world version of quantum money, a banknote would need to be stored for some period of time (like how money sits in an account). But with a quantum state acting as a banknote, it is natural that errors will occur as time is spent in storage.

With enough errors, a valid banknote would fail a verification check, resulting in a loss of value. Thus, a noise-tolerant scheme is a necessity.

There are, in fact, already noise-tolerant quantum money schemes, though these all fall into a category of quantum money called *private-key* quantum money. In private-key schemes, like Wiesner's original scheme [Wie83], only banks are capable of verifying the validity of banknotes. A much stronger version of quantum money is to allow *anybody* to verify a banknote; this is known as the *public-key* setting.

In this thesis, we present the first noise-tolerant public-key quantum money scheme. Our approach is to use the foundational framework developed by Aaronson and Christiano [AC12] and build upon the fact that their scheme resembles a well-known quantum error correcting code. This results in a scheme for public-key quantum money that is inherently noise-tolerant, as opposed to applying a layer of quantum error correction on top of an existing scheme.

1.3 Specious adversaries

Practical problems are often thought of as the problems that arise when theoretical protocols are translated to real-world implementations. While this is the source of most practical problems, there are also those that arise in the theoretical development of protocols, and thus must be tackled by theorists alone. In the development of any quantum cryptographic protocol, a necessary component is the associated security analysis/proof, which is often the most difficult and time-consuming part of the development process. A valuable asset to a cryptographer is any technique for efficiently proving the security of a protocol. Apart from speeding up the development process, awareness of such techniques also contributes towards intuition on what protocols will or will not work. The aim of this section is to contribute towards the development of such a technique for the task of secure two-party computation.

In secure two-party computation, we have two parties, Alice and Bob, who interact to accomplish some task, though they may not necessarily trust each other. In the classical setting, Alice and Bob, who have respective inputs x and y, seek to jointly evaluate some function f on their inputs $f(x, y)$. Since they do not necessarily trust each other, they want to accomplish this in such a way that Alice does not learn anything of Bob's input y beyond what can be learned from the output of the function, and vice versa. The quantum setting is similar except that Alice's and Bob's inputs are parts of a quantum state, and the function f is replaced by an operation that evolves quantum states.

The security of a cryptographic protocol can be considered against different classes of adversaries, where a class is determined by the strength of the adversaries. While it is desirable to prove security against the strongest type of adversary, it is not always obvious how this can be done. When we consider a weaker adversary (they

are restricted, in some well-defined way, in how malicious they can act), we are often afforded a simpler security analysis.

Along these lines, it is interesting to consider the weak class of adversaries that was formally defined in [DNS10] for quantum two-party computation. An adversary within this class is called a *specious adversary* and is weak in the sense that they they must "appear" honest throughout the protocol. This means that, at *any* step of the protocol, a check can be performed to confirm that the current state of the protocol is what we would expect from a protocol executed with two honest parties. As a consequence, this means that, at the end of the protocol, the expected output is always produced. So, although specious adversaries have unbounded computational power and memory, this requirement of appearing honest restricts how malicious they can act, though there are still many things they could do to deviate from the protocol. The classical version of this adversary (the *semi-honest adversary*) is well-motivated: this is the adversary that executes a protocol honestly but records everything that occurs throughout with the hope of using this data to break privacy at a later point. As a result of the no-cloning theorem, a quantum analogue of the semi-honest adversary is not obviously defined, but the specious adversary defined in [DNS10] evokes the same intuition.

By being a weak adversary, the specious adversary can be used as a tool for checking if a proposed protocol satisfies a minimum level of security. A potentially greater utility appeared in [DNS12] where the authors showed that, in a particular instance, if one proves the security of a protocol against specious adversaries, then the protocol can be compiled into one that is secure against stronger adversaries. Thus, in certain cases, it may suffice to only prove security against specious adversaries. This naturally leads to the following question: how can we easily prove security against the class of specious adversaries? Answering this question forms the basis of the technique we seek to develop.

Our technique is the formulation of a canonical specious adversary, where canonical means that if we prove security against such an adversary, then, as a consequence, we have proved security against an entire set of specious adversaries. In this thesis, we formally define the notion of a canonical specious adversary. We then define a particular specious adversary, which we call the *greedy adversary*, and show that it satisfies our definition of being canonical within a restricted setting. Although our greedy adversary is canonical, it is defined in such a way that proving security against it is not obviously simpler than proving security against the entire set of specious adversaries. Put simply, the greedy adversary relies on the information-theoretic setting, where we allow unbounded computational power and quantum memory, to perform "all" dishonest behaviour available to it, at each step of the protocol. Thus, the greedy adversary mostly serves to prove that the canonical definition can be satisfied, though it does not satisfy our initial aim of utilizing a canonical specious adversary to simplify security analyses. Following our results, we discuss several follow-up questions, such

as how one might define a canonical specious adversary that is easier to work with, and the existence of canonical specious adversaries in less restrictive settings.

1.4 Outline

We now describe the structure of this thesis. As much as reasonably possible, this thesis is self-contained. When reviewing preliminary material, we omit proofs and instead refer the reader to the sources that are cited in the relevant section.

In Chapter 2, we introduce our notation for frequently used sets and review some basic concepts from linear algebra and probability theory.

In Chapter 3, we review the necessary ideas of quantum theory. This includes the foundational postulates of quantum theory and how we represent quantum states as vectors or density operators. We then review specialized topics that will be referenced later, such as quantum error correction.

In Chapter 4, we discuss the first of our three main topics: device-independent oblivious transfer. We start with a review of mathematical preliminaries that are specific to this chapter and then proceed to formally review the notions of oblivious transfer and self-testing with a computational assumption. We then combine these building blocks to form our main protocol for device-independent oblivious transfer and analyze its security.

Next, in Chapter 5, we discuss noise-tolerant public-key quantum money. We recall the public-key quantum money scheme from [AC12] and, after drawing connections to quantum error correction, we show how the scheme can be made inherently noise-tolerant. We then analyze our scheme and examine the trade-off between security and noise-tolerance.

In Chapter 6, we study specious adversaries as our final topic. We recall the formalism introduced in [DNS10] to analyze the privacy of two-party protocols and specious adversaries. We then define the notion of a canonical specious adversary and, after characterizing the behaviour of specious adversaries in a restricted setting, we define the greedy adversary and prove that it satisfies our definition of canonical.

Lastly, in Chapter 7, we conclude the thesis with a brief summary and a selection of questions that could be examined for future research.

Chapter 2

Preliminaries

2.1 Sets and notation

We use the standard notation for the following frequently used sets:

- Natural numbers: $\mathbb{N} = \{1, 2, 3, \ldots\}$.
- Integers: $\mathbb{Z}$.
- Real numbers: $\mathbb{R}$.
- Complex numbers: $\mathbb{C}$.

The set of natural numbers up to and including n is denoted as $[n] = \{1, 2, \ldots, n\}$.

The set of integers modulo n, where $n \in \mathbb{N}$, is denoted by $\mathbb{Z}_n = \mathbb{Z}/n\mathbb{Z}$. Each element of this set forms a congruence class. With modular addition $\oplus$, the set $\mathbb{Z}_n$ forms an abelian group. With multiplication, the set of all congruence classes in $\mathbb{Z}_n$ with multiplicative inverses forms an abelian group. A particularly important example of $\mathbb{Z}_n$ is $\mathbb{Z}_2$, which is equivalent to the space of a single bit; under this equivalence, addition is the XOR operation and multiplication is the AND operation.

We frequently consider the notion of an n-bit string $x \in \mathbb{Z}_2^n$. We concatenate the entries of x in the following sense $x = x_1 \cdots x_n$.

2.1.1 Asymptotic notation

Here, we mainly use [CLRS22] to review various notations that are used to characterize the asymptotic behaviour of a function. This notation can be particularly useful for studying the asymptotic efficiency of an algorithm, *i.e.*, when the input sizes become so large that we are only concerned with the order of growth of the running time.

Throughout this section, let $f, g : \mathbb{N} \to [0, \infty)$.

We denote by $O(g(n))$ the set of functions such that $f(n) \in O(g(n))$ if there exists $N \in \mathbb{N}$ and $k > 0$ such that

$$f(n) \leq kg(n), \quad \text{for all} \quad n \geq N. \tag{2.1.1}$$

This notation is frequently used and often referred to as *big-O* notation.

Similarly, $f(n) \in o(g(n))$ if, for *any* $k > 0$, there exists $N \in \mathbb{N}$ such that

$$f(n) \leq kg(n), \quad \text{for all} \quad n \geq N. \tag{2.1.2}$$

We say $f(n) \in \Omega(g(n))$ if there exists $k \geq 0$ and $N \in \mathbb{N}$ such that

$$kg(n) \leq f(n), \quad \text{for all} \quad n \geq N. \tag{2.1.3}$$

Equivalently, $f(n) \in \Omega(g(n))$ if and only if $g(n) \in O(f(n))$.

The notation $O(g(n))$ is frequently shortened to $O(g)$, and likewise for the other notations.

The function f is said to be *polynomial* if $f \in O(n^k)$ for some $k \in \mathbb{N}$. The set of all polynomial functions is denoted as $\mathsf{poly}(n)$.

In the context of cryptography, we are often interested in "small success probabilities" for proving the security of schemes against adversaries. This notion of "small" is formalized through *negligible functions*. From [KL15], the function f is *negligible* if, for every positive polynomial p, there exists an $N \in \mathbb{N}$ such that for all $n > N$, it holds that $f(n) < 1/p(n)$. This is equivalent to saying that for all $k \in \mathbb{N}$, there exists an N such that for all $n \geq N$, it holds that $f(n) < n^{-k}$. It can be shown that f is negligible if and only if $\lim_{n\to\infty} f(n)p(n) = 0$ for any positive polynomial p. We denote an arbitrary negligible function by $\mathsf{negl}(n)$.

2.2 Vector spaces

In this section, we briefly review some concepts from linear algebra, though we assume some basic familiarity. We mostly rely on [NC10] throughout this section.

A *vector space* V over a field $\mathbb{F}$ is a set that is closed under addition and scalar multiplication. A *subspace* of a vector space V is a non-empty subset of $W \subseteq V$ such that W is closed under addition and scalar multiplication.

Let V be a vector space over a field $\mathbb{F}$, and let $v_1, \ldots, v_n$ be elements of V. The set of elements $\{v_1, \ldots, v_n\}$ is said to be *linearly dependent* if there exist elements $c_1, \ldots, c_n \in \mathbb{F}$, not all equal to 0, such that

$$\sum_{i=1}^{n} c_i v_i = 0, \tag{2.2.1}$$

else the collection is said to be *linearly independent*. Furthermore, if $\{v_1, \ldots, v_n\}$ are linearly independent and generate the vector space V, then the collection forms a *basis* of V. The dimension of V, denoted as $\dim(V)$, is the cardinality of the basis set (all bases of V have the same cardinality). A vector space is *finite-dimensional* if $\dim(V) < \infty$, which will be the case for all vector spaces we work with. Given a set of vectors $S = \{v_1, \ldots, v_k\}$, the *span* of S is defined as the set which consists of all linear combinations formed from the vectors in S.

At this point, it is useful to introduce "bra-ket" (or Dirac) notation which is ubiquitous in quantum mechanics. Under this notation, an element of the vector space $\mathbb{C}^n$ is denoted as $|v\rangle$, *i.e.,* $|v\rangle \in \mathbb{C}^n$. As we shall see momentarily, this enables a particularly convenient notation for the inner product.

Definition 2.2.1. *Let V be a vector space over $\mathbb{C}$. An **inner product** on V is a map $(\cdot,\cdot) : V \times V \to \mathbb{C}$ such that, for arbitrary $|u\rangle, |v\rangle, |w\rangle \in V$ and $\alpha, \beta \in \mathbb{C}$, the following conditions are satisfied:*

- *Positive-definiteness:*
$$(|u\rangle, |u\rangle) \geq 0, \tag{2.2.2}$$
with equality if and only if $u = 0$.

- *Conjugate symmetry:*
$$(|u\rangle, |v\rangle) = (|v\rangle, |u\rangle)^*. \tag{2.2.3}$$

- *Linearity:*
$$(|u\rangle, \alpha|v\rangle + \beta|w\rangle) = \alpha(|u\rangle, |v\rangle) + \beta(|u\rangle, |w\rangle). \tag{2.2.4}$$

*When V is equipped with an inner product, it is called an **inner product space**.*

Two vectors $|u\rangle, |v\rangle$ are *orthogonal* if $(|u\rangle, |v\rangle) = 0$.

Definition 2.2.2. *Let V be a vector space over $\mathbb{C}$. A **norm** is a function $\|\cdot\| : V \to \mathbb{R}$ such that, for arbitrary $|u\rangle, |v\rangle \in V$ and $\lambda \in \mathbb{C}$, the following conditions are satisfied:*

- *Positive-definiteness: $\|u\| = 0$ if and only if $u = 0$.*
- *Homogeneity: $\|\lambda u\| = |\lambda|\|u\|$.*
- *Triangle inequality: $\|u + v\| \leq \|u\| + \|v\|$.*

*The pair $(V, \|\cdot\|)$ is called a **normed vector space**.*

If a vector satisfies $\||v\rangle\| = 1$, it is a *unit vector*.

Definition 2.2.3. *Let X be a set. A* ***metric*** *is a function $d : X \times X \to \mathbb{R}$ such that, for arbitrary $x, y, z \in X$, the following conditions are satisfied:*

- *Positive-definiteness: $d(x, y) = 0$ if and only if $x = y$.*
- *Symmetry: $d(x, y) = d(y, x)$.*
- *Triangle inequality: $d(x, y) \leq d(x, z) + d(z, y)$.*

The pair (X, d) is called a ***metric space****.*

Example 2.2.4. Consider the vector space of n-bit strings, $\mathbb{Z}_2^n$. The *Hamming distance* $d(x, y)$ is defined as the number of places where the bit strings x and y differ; mathematically,

$$d(x, y) = \sum_{i=1}^{n} |x_i - y_i|. \tag{2.2.5}$$

It is straightforward to verify that the Hamming distance indeed defines a metric on $\mathbb{Z}_2^n$. A related concept is the *Hamming weight* of an element $x \in \mathbb{Z}_2^n$ which is denoted as $\mathrm{wt}(x)$ and is defined as the number of places where x differs from the all-zero bit string 0^n. So, for example, $\mathrm{wt}(1001) = 2$. Then the Hamming distance $d(x, y)$ can be formulated as $d(x, y) = \mathrm{wt}(x + y)$.

The inner product we most frequently use is given by

$$(|u\rangle, |v\rangle) = \sum_i u_i^* v_i. \tag{2.2.6}$$

Given a vector $|u\rangle$, its *dual* is a linear operator denoted by $\langle u|$ and defined by

$$\langle u| : |v\rangle \longmapsto (|u\rangle, |v\rangle). \tag{2.2.7}$$

The inner product in equation (2.2.6) can then be conveniently denoted as

$$\langle u|v\rangle \equiv (|u\rangle, |v\rangle), \tag{2.2.8}$$

where the dual of $|u\rangle$ is given by its conjugate transpose.

Note that any inner product on a vector space induces a norm via

$$\||v\rangle\| := \sqrt{\langle v|v\rangle}. \tag{2.2.9}$$

Using the inner product in equation (2.2.6), we get the Euclidean norm on $\mathbb{C}^n$,

$$\||v\rangle\| = \sqrt{\langle v|v\rangle} = \sqrt{\sum_{i=1}^{n} |v_i|^2}. \tag{2.2.10}$$

Additionally, any norm on a vector space induces a metric via $d(x, y) \coloneqq \|x - y\|$.

If we have a basis $\{|v_1\rangle, \ldots, |v_n\rangle\}$ for an inner product space V with the property that $\langle v_i|v_j\rangle = 0$ for $i \neq j$, then we say that it is an *orthogonal basis*. If we have the additional property that $\||v_i\rangle\| = 1$ for each basis vector, then we say that it is an *orthonormal basis*.

For the vector space $\mathbb{C}^2$, we denote the standard basis with the following notation,

$$|0\rangle \coloneqq \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad |1\rangle \coloneqq \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{2.2.11}$$

This notation will naturally generalize when we introduce tensor products.

When a finite-dimensional complex vector space is equipped with an inner product, we refer to it as a *Hilbert space* and denote it as $\mathcal{H}$.

Given any subspace S of an inner product vector space V, we can define its *orthogonal complement* $S^\perp$ which consists of all elements orthogonal to every element of S,

$$S^\perp = \{|w\rangle \in V : \langle w|v\rangle = 0, \text{for all } |v\rangle \in V\}. \tag{2.2.12}$$

Furthermore, V is then the direct sum of S and $S^\perp$, and we have the following relation $\dim(V) = \dim(S) + \dim(S^\perp)$.

2.2.1 Linear operators

Given two vector spaces V, W, a *linear operator* (or, *linear map*) is a function $A : V \to W$ which is linear in its inputs. When we say that A is a linear operator on the vector space V, we mean $A : V \to V$.

We can represent each linear operator as a matrix (and each matrix can be viewed as a linear operator). If the vector spaces V, W have respective bases $\{|v_1\rangle, \ldots, |v_m\rangle\}$ and $\{|w_1\rangle, \ldots, |w_n\rangle\}$, then for each $j = 1, \ldots, m$, there exist complex numbers A_{ij} such that

$$A|v_j\rangle = \sum_{i=1}^{n} A_{ij}|w_i\rangle. \tag{2.2.13}$$

The $n \times m$ matrix with entries A_{ij} is then the matrix representation of A.

Expanding upon this point, suppose that $\{|v_1\rangle, \ldots, |v_m\rangle\}$ is an orthonormal basis. Then this basis satisfies the *completeness relation*

$$\sum_i |v_i\rangle\langle v_i| = I \tag{2.2.14}$$

where I is the identity and $\{\langle v_i|\}$ forms a basis for the dual space. This follows from the fact that an arbitrary vector $|z\rangle$ can be written as $|z\rangle = c_i|v_i\rangle$ so that

$$\Big(\sum_i |v_i\rangle\langle v_i|\Big)|z\rangle = \sum_i |v_i\rangle\langle v_i|z\rangle = \sum_i c_i|v_i\rangle = |z\rangle. \tag{2.2.15}$$

If both bases $\{|w_i\rangle\}, \{|v_i\rangle\}$ are orthonormal, then for a linear operator $A : V \to W$, we have

$$\begin{aligned} A &= I_W A I_V \\ &= \sum_{ij} |w_i\rangle\langle w_i| \, A \, |v_j\rangle\langle v_j| \\ &= \sum_{ij} \langle w_i|A|v_j\rangle \, |w_i\rangle\langle v_j| \end{aligned}$$

so that $A_{ij} = \langle w_i|A|v_j\rangle$.

In certain cases, the operator A may be *diagonalizable*, meaning it has a *diagonal representation*

$$A = \sum_i \lambda_i \, |v_i\rangle\langle v_i| \tag{2.2.16}$$

where $\{|v_i\rangle\}_i$ forms an orthonormal set of eigenvectors for A, and λ_i is the eigenvalue corresponding to the eigenvector $|v_i\rangle$.

Given a linear operator A on a Hilbert space $\mathcal{H}$, the *adjoint* $A^\dagger$ is the unique linear operator which satisfies the following property for all $|v\rangle, |w\rangle \in \mathcal{H}$,

$$(|v\rangle, A\,|w\rangle) = (A^\dagger \, |v\rangle, |w\rangle). \tag{2.2.17}$$

In the matrix representation of A, the adjoint is precisely given by the conjugate transpose of A, *i.e.,* $A^\dagger = (A^*)^T$. The operator A is *Hermitian* (or *self-adjoint*) if $A = A^\dagger$.

A *projection* P on a vector space V is a linear operator $P : V \to V$ such that $P \circ P = P$ (*i.e.*, $P^2 = P$). When V is an inner product space, P is called an *orthogonal projection* if it satisfies the additional condition of being Hermitian. As an example, if $\{|v_i\rangle\}_{i=1,\ldots,n}$ forms an orthonormal basis for V, then the operator defined as

$$P = \sum_{i=1}^{k} |v_i\rangle\langle v_i| \tag{2.2.18}$$

is a projector onto the subspace with orthonormal basis $\{|v_i\rangle\}_{i=1,\ldots,k}$. In fact, this is an orthogonal projector since $|v_i\rangle\langle v_i|$ is Hermitian for each i. Unless otherwise specified, a projector will always mean an orthogonal projector and so we use the terms interchangeably.

An operator A is *normal* if it satisfies $A^\dagger A = AA^\dagger$. Notably, any Hermitian operator is automatically normal.

Theorem 2.2.5 (Spectral decomposition)**.** *For any normal operator A on a vector space V, there exists an orthonormal basis of V such that A is diagonal with respect to the basis. Conversely, any diagonalizable operator is normal.*

A linear operator (or matrix) U is unitary if $U^\dagger U = UU^\dagger = I$. Notably, if U is unitary and acts on an inner product space V, then U preserves the inner product,

$$(U|v\rangle, U|w\rangle) = \langle v|U^\dagger U|w\rangle = \langle v|w\rangle, \quad \text{for all } v, w \in V. \tag{2.2.19}$$

Furthermore, U preserves the norm of a vector $|\psi\rangle$.

Definition 2.2.6. *The* ***trace*** $\mathrm{tr}(A)$ *of a square matrix A is defined as the sum of its diagonal elements,*

$$\mathrm{tr}(A) \coloneqq \sum_i A_{ii}, \tag{2.2.20}$$

With an orthonormal basis $\{|i\rangle\}$ for a Hilbert space $\mathcal{H}$, we can equivalently write the trace as $\mathrm{tr}(A) = \sum_i \langle i|A|i\rangle$.

The trace of a linear operator A is defined as the trace of any matrix representation of A. The invariance of the trace under matrix representations of A follows from its cyclic property $\mathrm{tr}(AB) = \mathrm{tr}(BA)$. Thus, for a unitary U, we have $\mathrm{tr}\big(UAU^\dagger\big) = \mathrm{tr}\big(U^\dagger UA\big) = \mathrm{tr}(A)$.

2.2.2 Tensor products

Let V, W be Hilbert spaces of dimension m, n, respectively. Then we can define a new vector space, $V \otimes W$, of dimension mn. The elements of $V \otimes W$ are of the form $|v\rangle \otimes |w\rangle$, where $|v\rangle \in V$ and $|w\rangle \in W$. In the case where $\{|i\rangle\}_i$ and $\{|j\rangle\}_j$ are orthonormal bases of V and W, respectively, the set $\{|i\rangle \otimes |j\rangle\}_{ij}$ forms an orthonormal basis of $V \otimes W$.

Suppose A and B are linear operators on V and W, respectively. Then we can define a linear operator $A \otimes B$ which acts on an element of $V \otimes W$ as

$$(A \otimes B)(|v\rangle \otimes |w\rangle) \coloneqq A|v\rangle \otimes B|w\rangle. \tag{2.2.21}$$

We can define an inner product on $V \otimes W$ by using the inner products on V and W. Such an inner product is defined as

$$\Big(\sum_i a_i |v_i\rangle \otimes |w_i\rangle, \sum_j b_j |v'_j\rangle \otimes |w'_j\rangle\Big) \coloneqq \sum_{ij} a_i^* b_j \langle v_i|v'_j\rangle \langle w_i|w'_j\rangle. \tag{2.2.22}$$

We now introduce the *Kronecker product*, which is particularly useful for computing a matrix representation of a tensor product. Let A be an $m \times n$ matrix, and B be a $p \times q$ matrix. Then $A \otimes B$ is an $(mp) \times (nq)$ matrix,

$$A \otimes B \coloneqq \begin{bmatrix} A_{11}B & \cdots & A_{1n}B \\ \vdots & \vdots & \vdots \\ A_{m1}B & \cdots & A_{mn}B \end{bmatrix}. \tag{2.2.23}$$

2.3 Probability theory

In this section, we briefly recall some basic probability theory at a level that is sufficient for our needs. Further details can be found in [Bil95, Dur19, Mit05].

Consider a discrete probability space $(\Omega, \mathcal{F}, P)$, where Ω (the "set of outcomes") is a countable set and $\mathcal{F}$ (the "set of events") is the set of all subsets of Ω, and $P : \mathcal{F} \to [0, 1]$ is a probability measure which assigns probabilities to events and which has the property $P(\Omega) = 1$.

A (discrete) random variable X is a function $X : \Omega \to \mathcal{X}$, where $\mathcal{X} \subset \mathbb{R}$ is a finite set. The probability that X takes on a value $x \in \mathcal{X}$ is given by $P(\{\omega \in \Omega | X(\omega) = x\})$. We denote this probability by $\Pr(X = x)$ or by $p_X(x)$. In this way, X induces a probability measure which we refer to as its probability distribution.

The *expectation value* (or *mean value*) of the random variable X is defined as

$$\mathbb{E}[X] \coloneqq \sum_{x \in \mathcal{X}} x \Pr(X = x). \tag{2.3.1}$$

Lemma 2.3.1 (Chernoff bounds)**.** *Let X be a random variable. For any $a \in \mathbb{R}$, we have*

$$\Pr(X \geq a) \leq \mathbb{E}[e^{tX}]e^{-ta}, \quad \textit{for any } t > 0, \tag{2.3.2}$$

and

$$\Pr(X \leq a) \leq \mathbb{E}[e^{tX}]e^{-ta}, \quad \textit{for any } t < 0. \tag{2.3.3}$$

In addition to the bounds in Lemma 2.3.1, there are a variety of bounds referred to as Chernoff bounds that can be obtained by considering different distributions. A particularly relevant such bound is given below.

Lemma 2.3.2. *Let $X_1, \ldots, X_n$ be independent random variables taking values in the set $\{0, 1\}$. Let $X = X_1 + \cdots + X_n$ and $\mu = \mathbb{E}[X]$. Then, for $0 < \delta < 1$,*

$$\Pr(|X - \mu| \geq \delta\mu) \leq 2e^{-\mu\delta^2/3}. \tag{2.3.4}$$

Chapter 3

Quantum theory

In this chapter, we review the basic notions of quantum theory while also covering particularly relevant topics and results that will be used in later chapters. In Section 3.1, we introduce the formalism of quantum mechanics where quantum states are treated as vectors. In Section 3.2, we introduce an equivalent formalism where states are viewed as density matrices. Then, in Section 3.3, we provide background on various notions such as entanglement, quantum channels, entropy, and quantum error correction. We largely follow [NC10, Wil13] throughout this chapter.

3.1 State vector formalism

Postulate 1. *Any closed (i.e., isolated) quantum system is described by a unit vector (which we refer to as a quantum state) within a Hilbert space.*

The simplest system we can work with is that of a *qubit* which lives inside a two-dimensional Hilbert space. Consider some orthonormal basis, such as the standard basis $\{|0\rangle, |1\rangle\}$ (which we often refer to as the *computational basis*). In this basis, an arbitrary qubit can be written as

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle , \tag{3.1.1}$$

where $\alpha, \beta \in \mathbb{C}$ are such that $|\alpha|^2 + |\beta|^2 = 1$. This latter condition, often referred to as the *normalization condition*, enforces the requirement that $|\psi\rangle$ is a unit vector.

More generally, a quantum state $|\psi\rangle$ could be represented by a *superposition* of the states $\{|\psi\rangle_i\}_i$ in the following sense

$$|\psi\rangle = \sum_i \alpha_i |\psi_i\rangle , \tag{3.1.2}$$

where the coefficients α_i are such that the state $|\psi\rangle$ is normalized; these coefficients are sometimes referred to as *amplitudes*.

To handle multiple systems, we use the tensor product:

Postulate 2. *If we have multiple Hilbert spaces $\mathcal{H}_1, \ldots, \mathcal{H}_n$, then the overall space is the tensor product $\mathcal{H}_1 \otimes \cdots \otimes \mathcal{H}_n$. If, additionally, we have the state $|\psi_i\rangle \in \mathcal{H}_i$ for each i, then the joint state of the whole system is $|\psi_1\rangle \otimes \cdots \otimes |\psi_n\rangle$.*

Remark 3.1.1. We often use the word *register* to evoke the idea of a "place" where quantum information can be stored. To denote a register, we typically use cursive letters like $\mathcal{A}$ or $\mathcal{B}$. As an example, if we have some party, say Alice, we would denote their register as $\mathcal{A}$ and understand that it is associated to a Hilbert space $\mathcal{H}(\mathcal{A})$. The contents of the register is then described by a state $|\psi\rangle \in \mathcal{H}(\mathcal{A})$ or, when the context is clear, we may write $|\psi\rangle \in \mathcal{A}$. With another party Bob, whose register is $\mathcal{B}$, the joint Hilbert space is $\mathcal{H}(\mathcal{A} \otimes \mathcal{B})$. In the context of this composite system, we may refer to $\mathcal{A}$ and $\mathcal{B}$ as *subregisters* of $\mathcal{A} \otimes \mathcal{B}$. See [Wat18] for further details.

When we use the computational basis and work with composite systems, we often use the notation

$$|i\rangle \otimes |j\rangle \equiv |ij\rangle\,, \quad i, j \in \{0, 1\}, \tag{3.1.3}$$

which generalizes in a natural way.

Postulate 3. *Given a closed quantum system, the evolution of a state $|\psi\rangle \in \mathcal{H}$ is described by a unitary matrix U,*

$$|\psi'\rangle = U\,|\psi\rangle\,, \tag{3.1.4}$$

in the sense that the state $|\psi\rangle$ at time t_0 is related to the state $|\psi'\rangle$ at time t_1 by the unitary U.

Let $\mathsf{U}(\mathcal{H})$ denote the set of all unitaries on the Hilbert space $\mathcal{H}$. We now list some of the commonly used unitary matrices. Note that, in the context of quantum computations, we often refer to these matrices as *gates*. The single-qubit Pauli gates are:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}; \qquad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}; \qquad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{3.1.5}$$

The Hadamard gate H, the phase gate S, and the T gate (or $\pi/8$), are:

$$\mathsf{H} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}; \qquad S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}; \qquad T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}. \tag{3.1.6}$$

An important example of a two-qubit gate is the controlled-not gate CNOT,

$$\mathsf{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \tag{3.1.7}$$

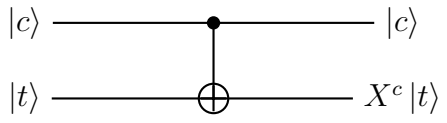

Figure 3.1: The CNOT gate.

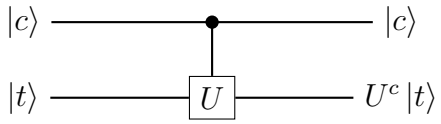

Figure 3.2: The application of the unitary U is controlled by the qubit $|c\rangle$.

which, in the computational basis, has the effect $|c\rangle\,|t\rangle \to |c\rangle\,|t \oplus c\rangle$ where we interpret $|c\rangle$ as the control qubit and $|t\rangle$ as the target qubit. Notably, the CNOT gate can be interpreted as the application of the Pauli-X gate, controlled by $|c\rangle$ (see Figure 3.1). This idea can be generalized to control the application of an arbitrary unitary U (see Figure 3.2). When this is done, we denote the controlled-U operation as CU.

Postulate 4. *The act of measuring some quantum state $|\psi\rangle$ (i.e., observing something) is formalized through a collection of measurement operators $\{M_m\}$ that act on the Hilbert space under consideration, where m runs over the set of possible measurement outcomes. The probability of observing m when measuring the state $|\psi\rangle$ is given by*

$$\Pr(m) = \langle\psi|\, M_m^\dagger M_m\, |\psi\rangle = \|M_m\,|\psi\rangle\|^2, \tag{3.1.8}$$

and the post-measurement state (i.e., the state that remains after observing m) is given by

$$\frac{M_m\,|\psi\rangle}{\sqrt{\langle\psi|\, M_m^\dagger M_m\,|\psi\rangle}}. \tag{3.1.9}$$

Additionally, the measurement operators satisfy the ***completeness relation****,*

$$I = \sum_m M_m^\dagger M_m. \tag{3.1.10}$$

Note that the intuition behind the completeness relation is that the probabilities should all sum to one,

$$1 = \sum_m \Pr(m) = \sum_m \langle\psi|\, M_m^\dagger M_m\,|\psi\rangle\,. \tag{3.1.11}$$

Remark 3.1.2. Consider two quantum states $|\psi\rangle$ and $e^{i\theta}\,|\psi\rangle$, where $\theta \in \mathbb{R}$. Observe that if we were to measure these quantum states (compute the probability of observing m for some measurement operator M_m), then

$$\Pr(m) = \langle\psi|e^{-i\theta} M_m^\dagger M_m e^{i\theta}|\psi\rangle = \langle\psi|M_m^\dagger M_m|\psi\rangle\,. \tag{3.1.12}$$

The conclusion is that these two states produce the same measurement statistics, and so we say they are identical up to the *global phase factor* $e^{i\theta}$. Thus, when a quantum

system is described by a vector $|\psi\rangle$, it more accurate to say that the quantum system is described by an equivalence class of vectors, where $|\psi\rangle$ is equivalent to $|\psi'\rangle$ if the two are related by a global phase factor. In the density matrix formalism, Section 3.2, this "ambiguity" is removed.

Projective measurements

A frequently used special case of the measurement formalism is that of *projective measurements*. A projective measurement is described by an *observable* M which is a Hermitian operator with spectral (*i.e.*, diagonal) decomposition

$$M = \sum_m m P_m \tag{3.1.13}$$

where P_m is a projector onto the eigenspace of M with eigenvalue m. The eigenvalues m correspond to the measurement outcomes of the observable. The probability of observing m is given by

$$\Pr(m) = \langle\psi| P_m |\psi\rangle \tag{3.1.14}$$

and the post-measurement state is

$$\frac{P_m |\psi\rangle}{\sqrt{\langle\psi| P_m |\psi\rangle}}. \tag{3.1.15}$$

If a collection of measurement operators $\{M_m\}$ satisfy the additional condition of the M_m being orthogonal projectors (M_m are Hermitian and $M_m M'_m = \delta_{m,m'} M_m$), then $\{M_m\}$ describes a projective measurement. It is in this sense that projective measurements can be viewed as a special case of the more general measurement formalism.

Remark 3.1.3. Instead of giving an observable M to describe a projective measurement, one often gives a set of orthogonal projectors P_m satisfying $I = \sum_m P_m$ and $P_m P_{m'} = \delta_{mm'} P_m$ instead, with the observable $M = \sum m P_m$ being implicit. This relates to the common phrase, *measures in the basis* $\{|m\rangle\}$. In this context, $|m\rangle$ forms an orthonormal basis, and the measurement being done is a projective measurement with projectors $P_m = |m\rangle\langle m|$. This is most widely used for the *computational basis* (*i.e.*, the standard basis) $\{|0\rangle, |1\rangle\}$ and the *Hadamard basis* (also known as the *diagonal basis*) which is denoted as

$$|+\rangle := \frac{|0\rangle + |1\rangle}{\sqrt{2}}, \qquad |-\rangle := \frac{|0\rangle - |1\rangle}{\sqrt{2}}. \tag{3.1.16}$$

In the case of the computational basis, the projectors are $|0\rangle\langle 0|$ and $|1\rangle\langle 1|$ which indeed satisfy the completeness relation. Note that the corresponding observable is the Pauli Z operator: $Z = |0\rangle\langle 0| - |1\rangle\langle 1|$.

3.2 Density matrix formalism

In addition to the state vector formalism introduced in Section 3.1, there is an equivalent formalism where a quantum state is represented by a *density matrix* (or *density operator*). Depending on the context, it will sometimes be more convenient to use one formalism over the other.

It is worth noting that, in the state vector formalism, we may have an ensemble of states $\{p_i, |\psi_i\rangle\}$ (*i.e.*, we only know that we have the state $|\psi_i\rangle$ with probability p_i), and in this case, it is especially convenient to use the density matrix formalism. In this instance, the density matrix is given by

$$\rho \equiv \sum_i p_i \, |\psi_i\rangle\langle\psi_i| \, , \tag{3.2.1}$$

and is called a *mixed state*. In the simpler case where we know that the quantum system is exactly given by $|\psi\rangle$, the associated density matrix is given by $\rho \equiv |\psi\rangle\langle\psi|$ and is said to be a *pure state* (note that when it is clear from context, we sometimes refer to $|\psi\rangle$ as a pure state).

It is important to note that we need not always come from the state vector formalism to define a density matrix.

Definition 3.2.1. *A **density operator** ρ on a Hilbert space $\mathcal{H}$ is a positive semi-definite operator on $\mathcal{H}$ (i.e., $\langle\phi|\rho|\phi\rangle$ is real and non-negative for every $|\phi\rangle \in \mathcal{H}$) with trace one, $\mathrm{tr}(\rho) = 1$. The space of all density operators on $\mathcal{H}$ is denoted as $\mathcal{D}(\mathcal{H})$.*

Under this definition, a density operator can always be associated to some ensemble $\{p_i, |\psi_i\rangle\}$. This follows from the fact that density operators are Hermitian (since they are positive semi-definite) and thus have a spectral decomposition (Theorem 2.2.5); with the $\mathrm{tr}(\rho) = 1$ condition, the decomposition corresponds to an ensemble with well-defined probabilities. As for whether the density matrix is pure or mixed, we have the criterion that ρ is a pure state if and only if $\mathrm{tr}(\rho^2) = 1$, otherwise it is a mixed state with $\mathrm{tr}(\rho^2) < 1$.

We can also consider the more general situation where we have the density matrix ρ_i with probability p_i, *i.e.*, an ensemble of ensembles. In this case, the overall system can be described by the density matrix $\rho = \sum_i p_i \rho_i$.

The postulates of quantum mechanics, rephrased in terms of density matrices, are as follows:

Postulate 1. *We associate, to any closed quantum system, a Hilbert space. The system itself is described by a density operator.*

Postulate 2. *If we have multiple Hilbert spaces, then the joint space is the tensor product of the Hilbert spaces. In the case where we have ρ_1 acting on $\mathcal{H}_1$, and ρ_2 acting on $\mathcal{H}_2$, then $\rho_1 \otimes \rho_2$ is the joint density operator acting on the joint system $\mathcal{H}_1 \otimes \mathcal{H}_2$.*

Note that a density operator which is a tensor product of other density operators is called a *product state*.

Postulate 3. *Given a closed quantum systen, the evolution of a density operator $\rho \in \mathcal{D}(\mathcal{H})$ is described by a unitary operator U,*

$$\rho' = U\rho U^\dagger, \tag{3.2.2}$$

in the sense that the state ρ at time t_0 is related to the state ρ' at time t_1 by the unitary U.

When it is clear from context, we sometimes denote the action of a unitary on a density matrix as $U \cdot \rho \equiv U\rho U^\dagger$.

Postulate 4. *Quantum measurements are described by a collection of measurement operators $\{M_m\}$ that act on the Hilbert space under consideration, where m refers to the possible measurement outcomes. The probability of observing m when measuring the density operator ρ is given by*

$$\Pr(m) = \mathrm{tr}\big(M_m^\dagger M_m \rho\big) \tag{3.2.3}$$

and the post-measurement state is given by

$$\frac{M_m \rho M_m^\dagger}{\mathrm{tr}\Big(M_m^\dagger M_m \rho\Big)}. \tag{3.2.4}$$

Additionally, the measurement operators satisfy the completeness relation

$$\sum_m M_m^\dagger M_m = I. \tag{3.2.5}$$

We are often interested in the case where we have a joint state $\rho_{AB} \in \mathcal{H}_A \otimes \mathcal{H}_B$ that describes the system A held by one party, Alice, and the system B held by another party, Bob. In these situations, we often wish to analyze the local state of a party, *i.e.*, in the case of Alice, a density operator ρ_A that describes her local system when she lacks access to Bob's system. To do this, we employ the *partial trace operation* which has the effect of "tracing out" Bob's system, leaving us with a local density operator that is sometimes called the *reduced density operator*.

Definition 3.2.2. *Let X_{AB} be a square operator acting on the Hilbert space $\mathcal{H}_A \otimes \mathcal{H}_B$, and let $\{|k\rangle\}$ be an orthonormal basis for $\mathcal{H}_B$. Then the* ***partial trace*** *over the Hilbert space $\mathcal{H}_B$ is defined as*

$$\mathrm{tr}_B(X_{AB}) \coloneqq \sum_k (\mathbb{1}_A \otimes \langle k|_B) X_{AB} (\mathbb{1}_A \otimes |k\rangle_B). \tag{3.2.6}$$

Note that, like the trace operation, the partial trace is a linear operation.

So, in the scenario we described, Alice's local density operator would be given by $\rho_A = \mathrm{tr}_B(\rho_{AB})$. Observe that in the case of a product state $\rho_A \otimes \sigma_B$, tracing out $\mathcal{H}_B$ has the effect $\mathrm{tr}_B(\rho_A \otimes \sigma_B) = \rho_A$.

Classical-quantum states. A classical random variable X can be encoded as a quantum state as

$$\sum_{x\in\mathcal{X}} p_X(x)\,|x\rangle\langle x|\,. \tag{3.2.7}$$

In some cases, a quantum state may depend on some classical random variable X in the sense that it is described by the density matrix ρ_A^x with probability $p_X(x)$. The joint state, consisting of both the classical random variable X and the quantum register A, is called a *classical-quantum* state; the ensemble is $\{p_X(x), |x\rangle\langle x|_X \otimes \rho_A^x\}_{x\in\mathcal{X}}$ and the corresponding density matrix is

$$\rho_{XA} = \sum_{x\in\mathcal{X}} p_X(x)\,|x\rangle\langle x|_X \otimes \rho_A^x. \tag{3.2.8}$$

A classical-quantum state is sometimes abbreviated as a cq-state. For more classical random variables, this convention naturally generalizes as ccq, cccq, etc. If a party, Alice, does not have access to the random variable X in equation (3.2.8), then their ensemble is $\{p_X(x), \rho_A^x\}_{x\in\mathcal{X}}$ and their state is given by

$$\rho_A = \mathrm{tr}_X(\rho_{XA}) = \sum_{x\in\mathcal{X}} p_X(x)\rho_A^x. \tag{3.2.9}$$

Purification. An important fact of density operators is that they can always be purified, where purification is understood in the sense of the following definition.

Definition 3.2.3. *A **purification** of a density operator $\rho_A \in \mathcal{D}(\mathcal{H}_A)$ is a pure bipartite state $|\psi\rangle_{RA} \in \mathcal{H}_R \otimes \mathcal{H}_A$ on a reference system R and the original system A, such that the reduced state on register A is equal to ρ_A,*

$$\rho_A = \mathrm{tr}_R(|\psi\rangle\langle\psi|_{RA}). \tag{3.2.10}$$

The purified state $|\psi\rangle_{RA}$ is sometimes referred to as the *global state*. From a physical point of view, a mixed state ρ_A can be interpreted as being entangled with some reference system R which we do not have access to. Adding to this, we can interpret noise that is local to our system A as being due to entanglement with the environment R.

To briefly see that a density operator $\rho_A \in \mathcal{D}(\mathcal{H}_A)$ can always be purified, recall that ρ_A always has a spectral decomposition and suppose it is of the form

$$\rho_A = \sum_x p(x)\,|x\rangle\langle x|_A\,. \tag{3.2.11}$$

Then by introducing a system $\mathcal{H}_R$, which is a copy of $\mathcal{H}_A$ with orthonormal basis $\{|x\rangle_R\}$, we can define a purification as

$$\rho_{RA} = \sum_x \sqrt{p(x)}\,|x\rangle_A\,|x\rangle_R\,. \tag{3.2.12}$$

Proposition 3.2.4. *A quantum state $\rho_A \in \mathcal{D}(\mathcal{H}_A)$ is a pure state if and only if the global state $\rho_{RA} \in \mathcal{D}(\mathcal{H}_R \otimes \mathcal{H}_A)$ is a pure product state, $\rho_R \otimes \rho_A$.*

An important note about purifications is that they are all equivalent up to some isometry.

Theorem 3.2.5. *Let ρ_A be a density operator, and let $|\psi\rangle_{R^1A}$ and $|\phi\rangle_{R^2A}$ be purifications of ρ_A such that $\dim(\mathcal{H}_{R^1}) \leq \dim(\mathcal{H}_{R^2})$. Then there exists an isometry $V_{R^1\to R^2}$ such that*

$$|\phi\rangle_{R^2A} = (V_{R^1\to R^2} \otimes I_A) |\psi\rangle_{R^1A} . \tag{3.2.13}$$

Corollary 3.2.6. *If $|\psi\rangle_{RA}, |\phi\rangle_{RA} \in \mathcal{H}_R \otimes \mathcal{H}_A$ are two purifications of a density operator ρ_A, then there exists a unitary U_R on $\mathcal{H}_R$ such that $|\psi\rangle_{RA} = (U_R \otimes I_A) |\phi\rangle_{RA}$.*

3.3 Quantum information

3.3.1 Entanglement, coset states, and no-cloning

Given a pure quantum state $|\psi\rangle_{AB} \in \mathcal{H}_A \otimes \mathcal{H}_B$, it may be impossible to write it as a tensor product. That is, $|\psi\rangle_{AB} \neq |\psi\rangle_A \otimes |\varphi\rangle_B$ for any $|\psi\rangle_A \in \mathcal{H}_A$ and $|\varphi\rangle_B \in \mathcal{H}_B$. When this is the case, we say that the quantum state $|\psi\rangle$ is *entangled*.

The most common entangled quantum states are the following two-qubit states,

$$\begin{aligned} |\phi^{(0,0)}\rangle &= \frac{|00\rangle + |11\rangle}{\sqrt{2}}, \qquad |\phi^{(0,1)}\rangle = \frac{|10\rangle + |01\rangle}{\sqrt{2}}, \\ |\phi^{(1,0)}\rangle &= \frac{|00\rangle - |11\rangle}{\sqrt{2}}, \qquad |\phi^{(1,1)}\rangle = \frac{|01\rangle - |10\rangle}{\sqrt{2}}. \end{aligned} \tag{3.3.1}$$

These states are referred to as *Bell* or *EPR states* (or, *pairs*), where EPR stands for Einstein, Podolsky, and Rosen. Observe that we can write these states succinctly as

$$|\phi^{(\alpha,\beta)}\rangle = (Z^\alpha X^\beta \otimes \mathbb{1}) \frac{|00\rangle + |11\rangle}{\sqrt{2}}. \tag{3.3.2}$$

We can directly verify that these states are entangled by attempting to write them as a tensor product of two single-qubit states: $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$ and $|\varphi\rangle = \gamma |0\rangle + \delta |1\rangle$. Their tensor product is

$$|\psi\rangle \otimes |\varphi\rangle = \alpha\gamma |00\rangle + \alpha\delta |01\rangle + \beta\gamma |10\rangle + \beta\delta |11\rangle . \tag{3.3.3}$$

Comparing this with, say, $|\phi^{(0,0)}\rangle$, we see that $\alpha\gamma = \beta\delta = 1/\sqrt{2}$ and $\alpha\delta = \beta\gamma = 0$ which is a contradiction and thus $|\phi^{(0,0)}\rangle$ is indeed entangled.

Entanglement is an especially useful resource in quantum computing, as the measurement outcomes of entangled states may be correlated and consequently exploited

to complete certain tasks that are impossible classically. As an example of such a correlation, consider the Bell state $|\phi^{(0,0)}\rangle$ and suppose one party, Alice, measures the first qubit in the computational basis. Alice may observe 0 or 1, each with equal probability. Supposing she observes $a = 0$, the post-measurement state is $|00\rangle$. Thus, if the second qubit (held by a different party, Bob) is also measured in the computational basis, he is guaranteed to observe $b = 0$.

Generalizing, if x and y are the bases used to measure, respectively, the first and second qubit of the state $|\phi^{(\alpha,\beta)}\rangle$, with respective outcomes a and b, then the following holds:

$$\begin{cases} a \oplus b = \beta, & \text{if } x = y = \texttt{Computational} \\ a \oplus b = \alpha, & \text{if } x = y = \texttt{Hadamard} \end{cases} \tag{3.3.4}$$

To describe a single-qubit measurement in either the `Computational` or `Hadamard` basis, we sometimes denote the measurement operators as $\{M_c^a\}_{a\in\{0,1\}}$, where c is a bit that denotes the measurement basis ($c = 0$ for `Computational`, and $c = 1$ for `Hadamard`). Along these lines, we may use the following notation to emphasize how the bit c determines the measurement basis,

$$[\texttt{Computational, Hadamard}]_c = \begin{cases} \texttt{Computational}, & \text{if } c = 0 \\ \texttt{Hadamard}, & \text{if } c = 1. \end{cases} \tag{3.3.5}$$

Then, for example, with $c \equiv \texttt{Hadamard}$, we have $M_1^0 = |+\rangle\langle+|$ and $M_1^1 = |-\rangle\langle-|$.

We now introduce the Hadamard transform. Let $x \in \{0,1\}^n$ and recall that H denotes the Hadamard gate. Then the n-qubit Hadamard transfom $\mathsf{H}^{\otimes n}$ is given by

$$\mathsf{H}^{\otimes n} |x\rangle = \frac{1}{\sqrt{2^n}} \sum_{z\in\{0,1\}^n} (-1)^{x\cdot z} |z\rangle . \tag{3.3.6}$$

There are some particular types of quantum states that will frequently appear in this thesis. Given this, we review these states now, starting with *conjugate-coding states*, which are ubiquitous in quantum cryptography. With $x, \theta \in \{0,1\}^n$, an n-tensor product of conjugate-coding states is

$$|x\rangle_\theta = |x_1\rangle_{\theta_1} \cdots |x_n\rangle_{\theta_n} , \tag{3.3.7}$$

where $|x_i\rangle_{\theta_i} = \mathsf{H}^{\theta_i} |x_i\rangle$. Note that in some literature, conjugate-coding states are referred to as BB84 states [Wie83, BB84].

We also need *subspace states* and *coset states*, where the latter is simply a generalization of the former. For a linear subspace A of $\mathbb{F}_2^n$ and $t, t' \in \{0,1\}^n$, a subspace state $|A\rangle$ and a coset state $|A_{t,t'}\rangle$ are defined as

$$|A\rangle = \frac{1}{\sqrt{|A|}} \sum_{v\in A} |v\rangle \quad \text{and} \quad |A_{t,t'}\rangle = X^t Z^{t'} |A\rangle = \frac{1}{\sqrt{|A|}} \sum_{v\in A} (-1)^{v\cdot t'} |v + t\rangle , \tag{3.3.8}$$

where $X^t = X^{t_1} \otimes \cdots \otimes X^{t_n}$, $Z^{t'} = Z^{t'_1} \otimes \cdots \otimes Z^{t'_n}$. The subspace state is a superposition over all the elements in the subspace A. In the computational basis, the coset state is a superposition over all elements in the coset $A + t$, while in the Hadamard basis (obtained by applying the Hadamard transform), it is a superposition over all elements in the coset $A^\perp + t'$.

The following lemma formally establishes a relationship between coset states and conjugate-coding states. For this lemma, let $\mathcal{B} = \{u_1, \ldots, u_n\}$ be a basis of $\mathbb{F}_2^n$ and let $U_\mathcal{B}$ be a unitary of $(\mathbb{C}^2)^{\otimes n}$ which acts as a change-of-basis from the standard basis to $\mathcal{B}$:

$$\forall x \in \{0,1\}^n, \qquad U_\mathcal{B} |x\rangle = \left| \sum_i x_i u_i \right\rangle . \tag{3.3.9}$$

Lemma 3.3.1 ([CV22])**.** *Let $\{u_1, \ldots, u_n\}$ be a basis of $\mathbb{F}_2^n$ and $U_\mathcal{B}$ be defined as in equation (3.3.9). Let $\{u^1, \ldots, u^n\}$ be its dual basis, i.e., $u^i \cdot u_j = \delta_{i,j}$. Let $T \subseteq \{1, \ldots, n\}$ be such that $|T| = n/2$ and $A = Span\{u_i : i \in \overline{T}\}$. Let $\theta \in \{0,1\}^n$ be the indicator of $\overline{T}$ (i.e., $\theta_i = 1$ if and only if $i \notin T$), and $x \in \{0,1\}^n$. Let $t = \sum_{i \in T} x_i u_i$ and $t' = \sum_{i \in \overline{T}} x_i u^i$. Then*

$$|A_{t,t'}\rangle = U_\mathcal{B} |x\rangle_\theta . \tag{3.3.10}$$

We now review the fundamental *no-cloning theorem* which, informally, says that quantum states cannot be cloned. While the no-cloning theorem is not directly used in this thesis, the concept fundamentally motivates topics we discuss in detail, like quantum money and the design of quantum error correction codes. For these reasons, we give the precise statement of the theorem (for perfect cloning of qubits) along with a proof, which we have included for the sake of completeness. For a generalization of the no-cloning theorem to mixed states, see [BCF+96].

Theorem 3.3.2 (No-cloning [Die82, WZ82])**.** *There is no unitary U such that*

$$U(|\psi\rangle |0\rangle) = (|\psi\rangle |\psi\rangle), \tag{3.3.11}$$

for all quantum states $|\psi\rangle$.

Proof: Suppose, for the sake of contradiction, that such a U exists. Then for any two quantum states $|\psi\rangle, |\phi\rangle$ we have

$$U(|\psi\rangle |0\rangle) = (|\psi\rangle |\psi\rangle), \tag{3.3.12}$$
$$U(|\phi\rangle |0\rangle) = (|\phi\rangle |\phi\rangle). \tag{3.3.13}$$

Now consider the following inner product $\langle\psi| \langle\psi| |\phi\rangle |\phi\rangle$ which we may compute as

$$\langle\psi| \langle\psi| |\phi\rangle |\phi\rangle = \langle\psi|\phi\rangle \langle\psi|\phi\rangle = (\langle\psi|\phi\rangle)^2. \tag{3.3.14}$$

Alternatively, we may compute this inner product as

$$\begin{aligned}\langle\psi|\langle\psi|\,|\phi\rangle\,|\phi\rangle &= \langle\psi|\langle 0|\,U^\dagger U\,|\phi\rangle\,|0\rangle\,, &(3.3.15)\\ &= \langle\psi|\langle 0|\,I\,|\phi\rangle\,|0\rangle\,, &(3.3.16)\\ &= \langle\psi|\phi\rangle\,\langle 0|0\rangle\,, &(3.3.17)\\ &= \langle\psi|\phi\rangle\,. &(3.3.18)\end{aligned}$$

Comparing these two computations, we find that $\langle\psi|\phi\rangle = (\langle\psi|\phi\rangle)^2$. The only solutions to this equation are $\langle\psi|\phi\rangle = 1$ (the states are identical) or $\langle\psi|\phi\rangle = 0$ (the states are orthogonal). Thus, we cannot in general clone an arbitrary state; we can only clone orthogonal states. ∎

3.3.2 Classical oracles

Let $f : \{0,1\}^* \to \{0,1\}$ be some function. Then a classical oracle U is a unitary transformation of the form

$$\begin{aligned}U\,|x\rangle\left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right) &= |x\rangle\left(\frac{|0\oplus f(x)\rangle - |1\oplus f(x)\rangle}{\sqrt{2}}\right)\\ \Leftrightarrow U\,|x\rangle\,|-\rangle &= (-1)^{f(x)}\,|x\rangle\,|-\rangle\,.\end{aligned}$$

Since the ancillary qubit $|-\rangle$ is left untouched by this operation, we refrain from writing it and simply say $U\,|x\rangle = (-1)^{f(x)}\,|x\rangle$. Frequently, we use a classical oracle U_A as a means for checking membership in some subset A in the following sense

$$U_A\,|x\rangle = \begin{cases} -\,|x\rangle\,, & \text{if } x \in A\\ |x\rangle\,, & \text{otherwise.}\end{cases} \tag{3.3.19}$$

As noted in [AC12], we can implement U_A as a projector $\mathbb{P}_A$. To do this, we initialize a control qubit $|+\rangle$ to control the application of U_A to $|x\rangle$. After this, we measure the control qubit in the Hadamard basis. If the outcome is $|-\rangle$, then $x \in A$, otherwise $x \notin A$. Indeed,

$$CU_A(|x\rangle\,,|+\rangle) = \frac{|x\rangle\,|0\rangle + U_A\,|x\rangle\,|1\rangle}{\sqrt{2}} = \begin{cases}|x\rangle\,|-\rangle & \text{if } x \in A\\ |x\rangle\,|+\rangle & \text{if } x \notin A.\end{cases} \tag{3.3.20}$$

Note that, unless otherwise stated, a classical oracle can be queried in a quantum superposition.

3.3.3 Quantum circuits

In this section, we follow [Wat08] to briefly review the quantum circuit model.

It is common to associate a quantum computation with the application of a *quantum circuit*, which is a unitary operator that acts on some number of qubits k. The k qubits can be divided into input qubits and ancillary (or auxiliary) qubits, where the ancilliary qubits are initially set to $|0\rangle$ and have the purpose of facilitating the computation. The unitary operator is a composition of unitary gates, taken from some fixed set, and it acts nontrivially on the input qubits. A classical outcome can be obtained by measuring some of the qubits after the application of the unitary operator. As an example of a quantum circuit, see Figure 3.4.

The fixed set of unitary gates is often taken to be a *universal gate set* where the term universal means that *every* unitary operation can be approximated, with arbitrary accuracy, via a quantum circuit constructed with gates from this set. Accuracy, here, is measured in terms of a metric that is defined via the *diamond norm* (see [Wil13]), though the details are not needed for this thesis. As an example, the following set is universal [NC10],

$$\{\mathsf{H}, \mathsf{S}, \mathsf{CNOT}, \mathsf{T}\}. \tag{3.3.21}$$

3.3.4 Quantum channels

The evolution of a quantum state through unitary transformations and measurements can be viewed as special cases of the more general notion of quantum channels. In this section, we define quantum channels according to a set of axioms. Before proceeding, we note that each quantum channel, defined relative to density operators alone, can be associated with a unique linear extension that maps linear operators to linear operators (see Appendix B of [Wil13] for details). For this reason, and for mathematical convenience, we define quantum channels in this section relative to linear operators.

Recall that $\mathcal{D}(\mathcal{H})$ denotes the space of density operators acting on the Hilbert space $\mathcal{H}$. Let $\mathcal{L}(\mathcal{H})$ denote the space of linear operators that act on $\mathcal{H}$ and have output in $\mathcal{H}$. We denote by $\mathcal{L}(\mathcal{H}_1, \mathcal{H}_2)$ as the space of all linear operators that take input from a Hilbert space $\mathcal{H}_1$ and output to the Hilbert space $\mathcal{H}_2$.

Axiom 1. *A quantum channel $\mathcal{N} : \mathcal{L}(\mathcal{H}_A) \to \mathcal{L}(\mathcal{H}_B)$ is **linear**.*

Naturally, we would like a quantum channel to send density operators to density operators. To this point, quantum channels must preserve the class of positive semi-definite operators, which leads us to the requirement that quantum channels should be positive maps.

Definition 3.3.3. *A linear map $\mathcal{N} : \mathcal{L}(\mathcal{H}_A) \to \mathcal{L}(\mathcal{H}_B)$ is **positive** if $\mathcal{N}(X_A)$ is positive semi-definite for all positive semi-definite $X_A \in \mathcal{L}(\mathcal{H}_A)$.*

Being a positive map alone is not quite sufficient. In general, a given quantum state on some system $\mathcal{H}_A$ could be part of a larger quantum state $\rho_{RA} \in \mathcal{D}(\mathcal{H}_R \otimes \mathcal{H}_A)$. In this case, a map $\mathcal{N}_{A\to B}$ acting strictly on the system $\mathcal{H}_A$ with output $\mathcal{H}_B$ is actually described by the evolution $\mathbb{1}_R \otimes \mathcal{N}_{A\to B}$, where $\mathbb{1}_R$ is the identity operator on the system $\mathcal{H}_R$. In this instance, we want the map $\mathbb{1}_R \otimes \mathcal{N}_{A\to B}$ to be positive, not just $\mathcal{N}_{A\to B}$, which leads us to the following definition.

Definition 3.3.4. *A linear map $\mathcal{N} : \mathcal{L}(\mathcal{H}_A) \to \mathcal{L}(\mathcal{H}_B)$ is **completely positive** if $\mathbb{1}_R \otimes \mathcal{N}$ is a positive map for a reference system R of arbitrary size.*

Axiom 2. *A quantum channel $\mathcal{N} : \mathcal{L}(\mathcal{H}_A) \to \mathcal{L}(\mathcal{H}_B)$ is **completely positive**.*

Lastly, we recall that density operators have trace one, and thus a quantum channel should preserve this quality. This quality, which we refer to as *trace-preserving*, formally means that $\mathrm{tr}(X_A) = \mathrm{tr}(\mathcal{N}(X_A))$ for all $X_A \in \mathcal{L}(\mathcal{H}_A)$.

Axiom 3. *A quantum channel $\mathcal{N} : \mathcal{L}(\mathcal{H}_A) \to \mathcal{L}(\mathcal{H}_B)$ is **trace-preserving**.*

With these three axioms, we have the following definition for quantum channels.

Definition 3.3.5. *A **quantum channel** (or, **quantum operation**) is a linear, completely positive, trace-preserving (CPTP) map.*

A convenient characterization of quantum channels is given by the following theorem. This representation of a quantum channel is also called the *operator-sum* representation.

Theorem 3.3.6. *A map $\mathcal{N} : \mathcal{L}(\mathcal{H}_A) \to \mathcal{L}(\mathcal{H}_B)$ is a linear CPTP map if and only if it has a **Choi-Kraus decomposition** in the sense that there exist $K_j \in \mathcal{L}(\mathcal{H}_A, \mathcal{H}_B)$ for all $j \in \{0, \ldots, d-1\}$ such that*

$$\sum_{l=0}^{d-1} K_j^\dagger K_j = I_A, \tag{3.3.22}$$

and for all $X_A \in \mathcal{L}(\mathcal{H}_A)$ we have

$$\mathcal{N}_{A\to B}(X_A) = \sum_{j=0}^{d-1} K_j X_A K_j^\dagger, \tag{3.3.23}$$

*where d need not be any larger than $\dim(\mathcal{H}_A)\dim(\mathcal{H}_B)$. The operators K_j are called **Kraus operators**.*

Remark 3.3.7. Equation (3.3.22) is a completeness relation, similar to the ones we have seen before, though this relation stems from the trace-preserving property of quantum channels.

We now show how a measurement can be viewed as a quantum channel. Suppose we have a density operator ρ and a set of measurement operators $\{M_m\}$. We know that the probability of observing the outcome m is given by $\Pr(m) = \operatorname{tr}\big(M_m^\dagger M_m \rho\big)$ and that the post-measurement state is given by $M_m \rho M_m^\dagger / \Pr(m)$. If the measurement is performed and we do not know the outcome (say, we did not record the outcome), then we have the resulting ensemble

$$\left\{ \Pr(m), \frac{M_m \rho M_m^\dagger}{\Pr(m)} \right\}_m \tag{3.3.24}$$

where the density operator is given by

$$\sum_m \Pr(m) \frac{M_m \rho M_m^\dagger}{\Pr(m)} = \sum_m M_m \rho M_m^\dagger. \tag{3.3.25}$$

From the above equation, we see that this evolution can be described by the quantum channel $\mathcal{N} = \sum_m M_m \rho M_m^\dagger$ where the Kraus operators are just the measurement operators $\{M_m\}$.

Evolution through a unitary operator also constitutes a quantum channel. In this case, there is only a single Kraus operator: the unitary $U \in \mathcal{L}(\mathcal{H})$. The unitary channel for the unitary U is denoted as

$$\mathcal{U}(\rho) = U \rho U^\dagger. \tag{3.3.26}$$

Lastly, it is useful to observe that if we have a state $\rho \in \mathcal{D}(\mathcal{H}_A \otimes \mathcal{H}_B)$ and quantum channels $\mathcal{V}$ and $\mathcal{N}$ that act locally on different registers, then these quantum channels commute, which is what we would intuitively expect. Indeed, suppose $\mathcal{V}$ acts on $\mathcal{H}_A$ in the sense that it has Kraus operators $\{V_j \otimes I_B\}$, and $\mathcal{N}$ has Kraus operators $\{I_A \otimes N_k\}$. Then from equation (3.3.23), we see that

$$\begin{aligned}
(\mathcal{V} \circ \mathcal{N})(\rho) &= \sum_{j,k} (V_j \otimes I_B)(I_A \otimes N_k) \rho (I_A \otimes N_k^\dagger)(V_j^\dagger \otimes I_B) \\
&= \sum_{j,k} (V_j \otimes N_k) \rho (V_j^\dagger \otimes N_k^\dagger) \\
&= \sum_{j,k} (I_A \otimes N_k)(V_j \otimes I_B) \rho (V_j^\dagger \otimes I_B)(I_A \otimes N_k^\dagger) \\
&= (\mathcal{N} \circ \mathcal{V})(\rho)
\end{aligned}$$

This observation will be particularly useful for us in the case where one of these quantum channels is taken to be the partial trace operation. Note that the partial trace operation $\operatorname{tr}_A(\rho_{AB})$ is indeed a quantum channel; its Kraus operators are $\{|x\rangle_A \otimes I_B\}$ where the set $\{|x\rangle_A\}$ forms an orthonormal basis of $\mathcal{H}_A$.

Extending quantum channels

In general, quantum channels as we have described so far are not unitary. However, if we consider the evolution of both the system of interest *and* the environment, then a quantum channel can be viewed as a unitary on this joint system followed by tracing out the environment. To understand this extension, we start by recalling the definition of an isometry.

Definition 3.3.8 (Isometry)**.** *Let $\mathcal{H}$ and $\mathcal{H}'$ be Hilbert spaces such that $\dim(\mathcal{H}) \leq \dim(\mathcal{H}')$. An isometry is a linear map $V : \mathcal{H} \to \mathcal{H}'$ such that $V^\dagger V = I_\mathcal{H}$. Equivalently, an isometry V is a linear, norm-preserving operator, in the sense that*

$$\| |\psi\rangle \| = \| V |\psi\rangle \| \tag{3.3.27}$$

for all $|\psi\rangle \in \mathcal{H}$.

Notable to isometries is the fact that, in general, they are maps between Hilbert spaces of different dimensions, meaning the property $VV^\dagger = \mathbb{1}_{\mathcal{H}'}$ does not necessarily hold. Instead, we have $VV^\dagger = \Pi_{\mathcal{H}'}$, where $\Pi_{\mathcal{H}'}$ is some projection onto $\mathcal{H}'$.

Definition 3.3.9 (Isometric channel)**.** *A quantum channel $\mathcal{V} : \mathcal{L}(\mathcal{H}) \to \mathcal{L}(\mathcal{H}')$ is an **isometric channel** if there exists a linear isometry $V : \mathcal{H} \to \mathcal{H}'$ such that*

$$\mathcal{V}(X) = VXV^\dagger, \tag{3.3.28}$$

for $X \in \mathcal{L}(\mathcal{H})$.

Observe that both unitary channels and isometric channels have a single Kraus operator. Since unitary channels are reversible, we intuitively expect isometric channels to be reversible as well, especially since $V^\dagger V = I_\mathcal{H}$. However, the channel $\mathcal{V}^\dagger$ does not constitute a quantum channel that reverses $\mathcal{V}$, as it is not trace-preserving. Instead, the reversing channel is given by

$$\mathcal{R}(Y) = \mathcal{V}^\dagger(Y) + \mathrm{tr}((I_{\mathcal{H}'} - \Pi_{\mathcal{H}'})Y)\sigma, \tag{3.3.29}$$

where $\sigma \in \mathcal{D}(\mathcal{H})$.

Before we can proceed further, we must take a moment to introduce the *Choi operator*.

Definition 3.3.10 (Choi operator)**.** *Let $\mathcal{H}_R$ and $\mathcal{H}_A$ be isomorphic Hilbert spaces, and let $\{|i\rangle_R\}$ and $\{|i\rangle_A\}$ be orthonormal bases for $\mathcal{H}_R$ and $\mathcal{H}_A$, respectively. Let $\mathcal{H}_B$ be some other Hilbert space, and let $\mathcal{N} : \mathcal{L}(\mathcal{H}_A) \to \mathcal{L}(\mathcal{H}_B)$ be a linear map. The **Choi operator** corresponding to $\mathcal{N}_{A\to B}$ and the bases $\{|i\rangle_R\}$ and $\{|i\rangle_A\}$ is defined as the following operator:*

$$(\mathbb{1}_R \otimes \mathcal{N}_{A\to B}) \cdot (|\Gamma\rangle\langle\Gamma|_{RA}) = \sum_{i,j=0}^{d_A-1} |i\rangle\langle j|_R \otimes \mathcal{N}_{A\to B}(|i\rangle\langle j|_A), \tag{3.3.30}$$

where $d_A \coloneqq \dim(\mathcal{H}_A)$ *and* $|\Gamma\rangle_{RA}$ *is an unnormalized maximally entangled vector defined as*

$$|\Gamma\rangle_{RA} \coloneqq \sum_{i=0}^{d_A-1} |i\rangle_R \otimes |i\rangle_A . \tag{3.3.31}$$

The rank of the Choi operator is called the ***Choi rank****.*

One useful feature of the Choi operator is that it completely specifies how the associated quantum channel $\mathcal{N}_{A\to B}$ acts on each of its inputs. To see this, observe that the Choi operator can be represented as a matrix of matrices, where the (i, j) entry is given by $\mathcal{N}_{A\to B}(|i\rangle\langle j|)$,

$$\begin{bmatrix} \mathcal{N}_{A\to B}\left(|0\rangle\langle 0|\right) & \mathcal{N}_{A\to B}\left(|0\rangle\langle 1|\right) & \cdots & \mathcal{N}_{A\to B}\left(|0\rangle\langle d^A-1|\right) \\ \mathcal{N}_{A\to B}\left(|1\rangle\langle 0|\right) & \mathcal{N}_{A\to B}\left(|1\rangle\langle 1|\right) & \cdots & \mathcal{N}_{A\to B}\left(|1\rangle\langle d^A-1|\right) \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{N}_{A\to B}\left(|d^A-1\rangle\langle 0|\right) & \mathcal{N}_{A\to B}\left(|d^A-1\rangle\langle 1|\right) & \cdots & \mathcal{N}_{A\to B}\left(|d^A-1\rangle\langle d^A-1|\right) \end{bmatrix}$$

Isometries enable us to extend quantum channels in a way that is analogous to the purification of density operators. In the case of density operators, a mixed state can always be purified by considering the environment system; then the mixed state is recovered by simply tracing out the environment. In the case of quantum channels, we can introduce the environment to extend a quantum channel to an isometric channel; and we can recover the original quantum channel by also tracing out the environment.

Definition 3.3.11 (Isometric extension)**.** *Let* $\mathcal{H}_A$ *and* $\mathcal{H}_B$ *be Hilbert spaces, and let* $\mathcal{N} : \mathcal{L}(\mathcal{H}_A) \to \mathcal{L}(\mathcal{H}_B)$ *be a quantum channel. Let* $\mathcal{H}_E$ *be a Hilbert space with dimension no smaller than the Choi rank of the channel* $\mathcal{N}$*. An* ***isometric extension*** *(or* ***Stinespring dilation****)* $U : \mathcal{H}_A \to \mathcal{H}_B \otimes \mathcal{H}_E$ *of the channel* $\mathcal{N}$ *is a linear isometry such that*

$$\operatorname{tr}_E(UX_AU^\dagger) = \mathcal{N}_{A\to B}(X_A) \tag{3.3.32}$$

for $X_A \in \mathcal{L}(\mathcal{H}_A)$*.*

By Definition 3.3.9, the quantum channel associated with the isometric extension is given by

$$\mathcal{U}^{\mathcal{N}}_{A\to BE}(X_A) = UX_AU^\dagger \tag{3.3.33}$$

for $X_A \in \mathcal{L}(\mathcal{H}_A)$. This notion is depicted in Figure 3.3.

To see that every quantum channel has an isometric extension, suppose we are given a quantum channel $\mathcal{N}_{A\to B}$ with the Kraus representation

$$N_{A\to B}(\rho_A) = \sum_j K_j \rho_A K_j^\dagger. \tag{3.3.34}$$

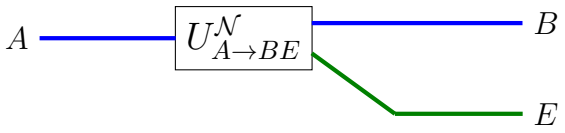

Figure 3.3: An isometric extension of the channel $\mathcal{N}_{A\to B}$. Here, we see that the environment E can receive some quantum information. The original channel $\mathcal{N}_{A\to B}$ can be obtained by tracing out E.

Then it can be shown that an isometric extension of this channel is given by

$$U^{\mathcal{N}}_{A\to BE} = \sum_j K_j \otimes |j\rangle_E \,. \tag{3.3.35}$$

At this point, we can nearly see how a quantum channel can be viewed as a unitary on a larger system. Since $U^\dagger U = I_A$, the extended channel in equation (3.3.33) can be reversed with the channel in equation (3.3.29), but the property $UU^\dagger = \Pi_{BE}$ prevents the channel from being unitary. However, if we extend the domain of our channel to also include the environment (so that we have a mapping of the form $\mathcal{H}_A \otimes \mathcal{H}_E \to \mathcal{H}_B \otimes \mathcal{H}_E$), then we can indeed view this extended channel as a unitary acting on the environment and our system of interest.

Remark 3.3.12. This perspective of quantum channels enables an interesting interpretation of the noise that arises in quantum channels. If $\mathcal{N}$ is some quantum channel being applied to our state ρ_A, then we can equivalently consider the associated unitary U_{AE} acting on our system A and the environment E. Supposing the environment starts in $|0\rangle\langle 0|_E$, the joint system evolves as $U_{AE}(\rho_A \otimes |0\rangle\langle 0|_E)U^\dagger_{AE}$. Generally, we only have access to our system A, and so the state we observe is obtained by tracing out the environment. This can result in a loss of information which we interpret as noise.

3.3.5 Closeness of quantum states

In this section, we review the notions of *trace distance* and *fidelity*, which are measures to quantify how "close" two quantum states are. Although these measures can be related to each other (Theorem 3.3.23), we sometimes find that it can be more convenient to use one measure over the other; hence, it is beneficial to discuss both measures.

Definition 3.3.13. *The* ***trace distance*** *between two density operators $\rho, \sigma \in \mathcal{D}(\mathcal{H})$ is defined as*

$$\Delta(\rho, \sigma) \coloneqq \frac{1}{2}\|\rho - \sigma\|. \tag{3.3.36}$$

where $\|\gamma\| = \mathrm{tr}\left(\sqrt{\gamma^\dagger \gamma}\right)$ for $\gamma \in \mathcal{D}(\mathcal{H})$.

Remark 3.3.14. Note that some literature adopts a convention of dropping the $\frac{1}{2}$ factor in the above definition (so that the trace distance is simply the norm of $\rho - \sigma$) but in this thesis we retain the $\frac{1}{2}$ factor.

We sometimes use the notation $\rho \approx_\varepsilon \sigma$ to mean that ρ and σ are ε-close in the sense that $\Delta(\rho, \sigma) \leq \varepsilon$.

The trace distance defines a metric on the space of density operators $\mathcal{D}(\mathcal{H})$ because it satisfies the following conditions:

- $\Delta(\rho, \sigma) = 0$ if and only if $\rho = \sigma$, where $\rho, \sigma \in \mathcal{D}(\mathcal{H})$.
- $\Delta(\rho, \sigma) = \Delta(\sigma, \rho)$, for all $\rho, \sigma \in \mathcal{D}(\mathcal{H})$.
- $\Delta(\rho, \tau) \leq \Delta(\rho, \sigma) + \Delta(\sigma, \tau)$, where $\rho, \sigma, \tau \in \mathcal{D}(\mathcal{H})$.

Note that the trace distance is upper bounded by 1, so that $0 \leq \Delta(\rho, \sigma) \leq 1$.

Lemma 3.3.15. *For any density operators $\rho_1, \rho_2, \sigma_1, \sigma_2$, the trace distance satisfies,*

$$\Delta(\rho_1 \otimes \rho_2, \sigma_1 \otimes \sigma_2) \leq \Delta(\rho_1, \sigma_1) + \Delta(\rho_2, \sigma_2). \tag{3.3.37}$$

As a special case of the above lemma, we have that

$$\Delta(\rho \otimes \omega, \sigma \otimes \omega) = \Delta(\rho, \sigma), \tag{3.3.38}$$

for any density operators ρ, σ, ω.

It is useful to note that the trace distance is invariant under isometric quantum channels; hence, it is invariant under unitary transformations,

$$\Delta(U\rho U^\dagger, U\sigma U^\dagger) = \Delta(\rho, \sigma). \tag{3.3.39}$$

More generally, we have the following theorem.

Theorem 3.3.16. *Let $\mathcal{N}$ be a quantum channel and $\rho, \sigma \in \mathcal{D}(\mathcal{H})$. Then*

$$\Delta(\mathcal{N}(\rho), \mathcal{N}(\sigma)) \leq \Delta(\rho, \sigma). \tag{3.3.40}$$

A related notion to trace distance is that of *fidelity*, which can also be used as a means of quantifying the "closeness" of quantum states. Though, unlike the trace distance, the fidelity does not define a metric on $\mathcal{D}(\mathcal{H})$.

Definition 3.3.17. *The **fidelity** between two density operators ρ, σ is defined as*

$$F(\rho, \sigma) \coloneqq \operatorname{tr}\left(\sqrt{\rho^{1/2}\sigma\rho^{1/2}}\right) \tag{3.3.41}$$

Proposition 3.3.18. *The fidelity between some density operator ρ and a pure state $|\psi\rangle$ is given by*

$$F(\rho, |\psi\rangle) = \sqrt{\langle\psi|\rho|\psi\rangle}. \tag{3.3.42}$$

We now introduce Uhlmann's theorem which, in some texts, is used to actually define the fidelity. It is also worth pointing out that in some of these cases, the fidelity may be defined as the square of the right-hand-side of equation (3.3.43).

Theorem 3.3.19 (Uhlmann's theorem). *Suppose $\rho, \sigma \in \mathcal{D}(\mathcal{H})$. Introduce a purifying system $\mathcal{R}$, which is a copy of $\mathcal{H}$. Then*

$$F(\rho, \sigma) = \max_{|\psi\rangle, |\phi\rangle} |\langle\psi|\phi\rangle|, \tag{3.3.43}$$

where the maximization is over all purifications $|\psi\rangle, |\phi\rangle$ of ρ, σ, respectively, into $\mathcal{H} \otimes \mathcal{R}$.

Equivalently,

$$F(\rho, \sigma) = \max_{|\phi\rangle} |\langle\psi|\phi\rangle| \tag{3.3.44}$$

where $|\psi\rangle$ is any fixed purification of ρ, and the maximization is over all purifications $|\phi\rangle$ of σ.

From Uhlmann's theorem, we can learn several useful properties about the fidelity. For instance, it reveals that the fidelity is symmetric in its inputs. Additionally, $0 \leq F(\rho, \sigma) \leq 1$, where equality with 1 is obtained when $\rho = \sigma$.

Uhlmann's theorem can be expanded upon to define the fidelity of a density operator with a subspace.

Definition 3.3.20. *The fidelity of $\rho \in \mathcal{D}(\mathcal{H})$ with a subspace S of $\mathcal{H}$ is defined as,*

$$F(\rho, S) = \max_{|\psi\rangle, |\phi\rangle} |\langle\psi|\phi\rangle| \tag{3.3.45}$$

where the maximization is over all purifications $|\psi\rangle$ of ρ and all unit vectors $|\phi\rangle \in S$.

As with the trace distance, the fidelity is invariant under isometries and hence unitaries. More generally, we have the following result.

Theorem 3.3.21. *Let $\mathcal{N}$ be a quantum channel and $\rho, \sigma \in \mathcal{D}(\mathcal{H})$. Then*

$$F(\rho, \sigma) \leq F(\mathcal{N}(\rho), \mathcal{N}(\sigma)). \tag{3.3.46}$$

Although the fidelity does not satisfy a true triangle inequality, we have the following result which will prove useful.

Lemma 3.3.22 ("Triangle inequality" for fidelity [AC12])**.** *Suppose $\langle\psi|\rho|\psi\rangle \geq 1 - \epsilon$ and $\langle\psi|\sigma|\psi\rangle \geq 1 - \epsilon$. Then $F(\rho, \sigma) \leq |\langle\psi|\phi\rangle| + 2\epsilon^{1/4}$.*

Lastly, the relationship between the trace distance and the fidelity is formalized by the following theorem.

Theorem 3.3.23 (Fuchs-van de Graaf inequalities)**.** *Let $\rho, \sigma \in \mathcal{D}(\mathcal{H})$ be two density operators. Then,*

$$1 - F(\rho, \sigma) \leq \Delta(\rho, \sigma) \leq \sqrt{1 - F(\rho, \sigma)^2}. \tag{3.3.47}$$

3.3.6 Entropy

Broadly, the notion of entropy in classical and quantum information theory characterizes the amount of uncertainty a party has about the state of some system; in other words, it characterizes randomness.

There are various measures of entropies and, depending on the problem at hand, it may be useful to consider one measure over another. In this section, we follow [Ren05, RW05, Sch07] to briefly review classical (smooth) Rényi entropy.

Rényi entropy

Definition 3.3.24 (**Rényi entropy** [Rén61])**.** *Let X be a random variable over a finite set $\mathcal{X}$ with probability distribution P_X. Let $\alpha \in [0, \infty]$. Then the **Rényi entropy of order** α is defined as*

$$H_\alpha(P_X) \coloneqq -\log\left(\left(\sum_{x \in \mathcal{X}} P_X(x)^\alpha\right)^{\frac{1}{\alpha - 1}}\right). \tag{3.3.48}$$

When the context is clear, we often write $H_\alpha(X)$ instead of $H_\alpha(P_X)$. We can obtain various entropy measures of interest by taking the limit of α. In the limit $\alpha \to \infty$, we get the *min-entropy*,

$$H_{\min}(X) \equiv H_\infty(X) = -\log\left(\max_{x \in \mathcal{X}} P_X(x)\right). \tag{3.3.49}$$

In the limit $\alpha \to 0$, we get the *max-entropy*,

$$H_{\max}(X) \equiv H_0(X) = \log\left|\{x \in \mathcal{X} : P_X(x) > 0\}\right| \tag{3.3.50}$$

In the limit $\alpha \to 1$, we recover the standard *Shannon entropy*,

$$H(X) = -\sum_x P_X(x) \log P_X(x). \tag{3.3.51}$$

The *conditional Rényi entropy* is defined as,

$$H_\alpha(X|Y) \coloneqq \frac{1}{1 - \alpha} \log\left(\max_y \sum_x P_{X|Y=y}(x)^\alpha\right). \tag{3.3.52}$$

For a fixed event $Y = y$, we have

$$H_\alpha(X|Y=y) \coloneqq \frac{1}{1-\alpha}\log\left(\sum_x P_{X|Y=y}(x)^\alpha\right). \tag{3.3.53}$$

As before, we can take the limit of α to get the conditional min-entropy ($\alpha \to \infty$) and conditional max-entropy ($\alpha \to 0$), which are, respectively, given as,

$$H_{\min}(X|Y) \coloneqq \min_y H_{\min}(X|Y=y) = \min_{x,y}\left(-\log P_{X|Y=y}(x)\right), \tag{3.3.54}$$

$$H_{\max}(X|Y) \coloneqq \max_y H_{\max}(X|Y=y) = \max_y \log|\{x \in \mathcal{X} : P_{X|Y=y}(x) > 0\}|. \tag{3.3.55}$$

Smooth Rényi entropy

Related to Rényi entropy is the notion of *smooth Rényi entropy*, which was introduced in [RW04]. Put simply, the smooth Rényi entropy H_α^ε is a family of entropies parametrized by a real number $\varepsilon \geq 0$, where $\varepsilon = 0$ corresponds to equation (3.3.48).

In [Ren05, RW05], smooth Rényi entropy is used to generalize the notion of conditional Rényi entropy. By once again considering the limits of α, one can obtain the *conditional smooth min-* and *max-entropy* which are, respectively, given by

$$H_{\min}^\varepsilon(X|Y) \equiv H_\infty^\varepsilon(X|Y) \coloneqq \max_{\mathcal{E}} \min_{x,y}\left(-\log\left(\frac{P_{XY\mathcal{E}}(x,y)}{P_Y(y)}\right)\right), \tag{3.3.56}$$

$$H_{\max}^\varepsilon(X|Y) \equiv H_0^\varepsilon(X|Y) \coloneqq \min_{\mathcal{E}} \max_y \log|\{x \in \mathcal{X} : \frac{P_{XY\mathcal{E}}(x,y)}{P_Y(y)} > 0\}|, \tag{3.3.57}$$

where the maximum/minimum ranges over all events $\mathcal{E}$ with probability $\Pr[\mathcal{E}] \geq 1-\varepsilon$, and $P_{XY\mathcal{E}}(x,y)$ is the probability that $\mathcal{E}$ occurs and X, Y take values x, y, respectively.

We refrain from defining $H_\alpha^\varepsilon(X|Y)$ here and instead note that it can be approximated by the conditional smooth min- and max-entropy. Indeed, in [RW05] it was shown that for $\alpha > 1$,

$$H_{\min}^{\varepsilon+\varepsilon'}(X|Y) + \frac{\log(1/\varepsilon')}{1-\alpha} \geq H_\alpha^\varepsilon(X|Y) \geq H_{\min}^\varepsilon(X|Y). \tag{3.3.58}$$

For $\alpha < 1$,

$$H_{\max}^{\varepsilon+\varepsilon'}(X|Y) - \frac{\log(1/\varepsilon')}{1-\alpha} \leq H_\alpha^\varepsilon(X|Y) \leq H_{\max}^\varepsilon(X|Y). \tag{3.3.59}$$

We now list some useful results, starting with the following chain rule for the conditional smooth min-entropy.

Lemma 3.3.25 (**Chain rule** [RW05])**.**

$$H_{min}^{\varepsilon+\varepsilon'}(X|Y) > H_{min}^{\varepsilon}(XY) - H_{max}(Y) - \log\left(\frac{1}{\varepsilon'}\right)$$

for all $\varepsilon, \varepsilon' > 0$.

The following lemma is a special case of a more general entropic uncertainty relation introduced in [DFR+07].

Lemma 3.3.26 ([DFR+07])**.** *Let* $\rho \in \mathcal{D}(\mathcal{H}_2^{\otimes n})$ *be an arbitrary* n*-qubit quantum state. Let* $\Theta = (\Theta_1, \ldots, \Theta_n)$ *be uniformly distributed over* $\{\texttt{Computational, Hadamard}\}^n$*, and let* $X = (X_1, \ldots X_n)$ *be the outcome when measuring* ρ *in basis* Θ*. Then for any* $0 < \lambda < 1/2$*,*

$$H_{min}^{\varepsilon}(X|\Theta) \geq \left(\frac{1}{2} - 2\lambda\right)n$$

where $\varepsilon = \exp\left(-\frac{\lambda^2 n}{32(2-\log(\lambda))^2}\right)$.

The following lemma is presented as a corollary in [DFR+07] to the *Min-Entropy-Splitting Lemma* (which essentially states that if the joint entropy of two random variables is large, then at least one of the random variables must have at least half of the original entropy in a randomized sense).

Lemma 3.3.27 ([DFR+07])**.** *Let* $\varepsilon \geq 0$*, and let* X_0, X_1, Z *be random variables (over possibly different alphabets) such that* $H_{min}^{\varepsilon}(X_0X_1|Z) \geq \alpha$*. Then, there exists a binary random variable* C *over* $\{0,1\}$ *such that* $H_{min}^{\varepsilon+\varepsilon'}(X_{1-C}|ZC) \geq \alpha/2 - 1 - \log(1/\varepsilon')$ *for any* $\varepsilon' > 0$.

We end this section by describing the meaning of the conditional smooth min-entropy. To do this, consider the following task. We have a classical random variable X and some additional variable R. We extract a binary string S from (X, R) such that, with probability $1-\varepsilon$, the string S looks uniformly random to an adversary who knows Y and R. A natural follow-up question is, what is the maximum length of S? This is answered by the conditional smooth min-entropy $H_{\min}^{\varepsilon}(X|Y)$. A formalization of this idea appears in Theorem 3.3.28 below.

Before we state the result, we must briefly introduce the notion of *hash functions* (for the full formal definition, see [KL15]). Informally, a hash function F takes bit strings of some length $\{0,1\}^n$ as input and produces bit strings that are shorter in length as output $\{0,1\}^\ell$, where $\ell < n$. This can be thought of as "compressing" the inputs. A *collision* is when we have distinct inputs x, x' that are mapped to the same output $F(x) = F(x')$. Let $\mathcal{F}$ be a class of hash functions from $\{0,1\}^n$ to $\{0,1\}^\ell$. The class $\mathcal{F}$ is said to be *two-universal* [CW79] if, for any distinct $x, x' \in \{0,1\}^n$ and for F uniformly distributed over $\mathcal{F}$, it holds that $\Pr[F(x) = F(x')] \leq 1/2^\ell$.

Theorem 3.3.28 (**Privacy amplification** [DFR+07])**.** *Let $\varepsilon \geq 0$. Let ρ_{XUE} be a ccq-state, where X takes values in $\{0,1\}^n$, U in the finite domain $\mathcal{U}$, and register E contains q qubits. Let F be the random and independent choice of a member of a two-universal class of hash functions $\mathcal{F}$ from $\{0,1\}^n$ to $\{0,1\}^\ell$. Then*

$$\Delta(\rho_{F(X)FUE}, \tfrac{1}{2^\ell}\mathbb{1} \otimes \rho_{FUE}) \leq \frac{1}{2} 2^{-\frac{1}{2}(H_{min}^{\varepsilon}(X|U)-q-\ell)} + 2\varepsilon.$$

3.4 Quantum error correction

To understand the general ideas of error correction and the challenges that arise in the quantum setting, we start by briefly discussing error correction in the classical setting. Suppose we have a classical bit $x \in \{0,1\}$ which we want to send through a communication channel to another party. In reality, the communication channel is noisy in the sense that, with some probability $p > 0$, an error occurs as x gets transmitted (and with probability $1-p$ no error occurs). In this classical setting, the only possible error that can occur is a *bit-flip*: if $x = 1$, the bit is changed to 0, and vice versa.

Protecting classical data against the noise in a communication channel largely relies on the fact that classical information can be arbitrarily copied. For example, consider the following simple error correction procedure. The first step is to *encode* the bit x, which can be done by copying it multiple times:

$$0 \to 000, \qquad 1 \to 111. \tag{3.4.1}$$

These encoded bits, 000 and 111, are known as the *logical codewords*. We now send the encoding of x through the noisy communication channel. At the other end of the channel, we can *detect* errors by taking a majority vote. If an error occurs which causes one of the bits to flip, say 000 becomes 010, then a majority vote tells us that the second bit should be 0. We can *correct* the error (or, *recover* the codeword) by flipping the second bit back to 0, and then *decode* the resulting codeword 000 to 0. However, this process can fail when two bit-flips occur, as it can lead us to incorrectly infer what the original codeword was; and if all three bits were flipped, so that 000 became 111, then the error goes undetected because 111 is itself a codeword.

In the quantum setting, there are a number of difficulties which hinder attempts to design a close analogue of any classical error correction procedure. For instance, the no-cloning theorem (Theorem 3.3.2) immediately rules out the classical approach of arbitrarily copying information to protect it. However, we can still use the basic idea of equation (3.4.1), where the data was distributed across extra resources, thereby introducing redundancy. This will not violate the no-cloning theorem, as the resulting encoded quantum states are entangled, not in a tensor product.

Another issue that is inherently quantum is the fact that a quantum state can collapse and erase information encoded in it. Consequently, quantum error correcting

codes must be carefully designed so that any measurements in the error correction procedure do not erase the information we are trying to protect.

In the sections that follow, we review the formalism of quantum error correction. We place particular emphasis on the stabilizer formalism and an especially useful quantum error correcting code called the Calderbank-Shor-Steane code. To do this, we primarily follow [Got97, NC10, Fuj15, Rof19].

3.4.1 Discretization of quantum errors

Suppose we have a qubit $|\psi\rangle$ which we send through a noisy quantum communication channel $\mathcal{N}$ which has Kraus operators $\{E_i\}$. Then,

$$\mathcal{N}(|\psi\rangle\langle\psi|) = \sum_i E_i |\psi\rangle\langle\psi| E_i^\dagger. \tag{3.4.2}$$

where the $\{E_i\}$ are interpreted as single-qubit errors.

As a special case of equation (3.4.2), we can consider the channel which, with probability $p > 0$, the qubit is affected by the Pauli X operator, and with probability $1 - p$ it is unaffected. Observe that an X error is the quantum analogue of a bit-flip error,

$$X |x\rangle = |x \oplus 1\rangle , \tag{3.4.3}$$

where $x \in \{0, 1\}$, and so this special case is an analogue of the classical noisy communication channel that we saw earlier.

In general, an arbitrary single-qubit error E_i in equation (3.4.2) can be written as a linear combination of the single-qubit Pauli gates,

$$E = \alpha_I I + \alpha_X X + \alpha_Z Z + \alpha_Y Y. \tag{3.4.4}$$

From equation (3.4.4), we see that there are an infinite number of errors that a qubit could be subject to, as opposed to the singular bit-flip error in the classical regime. Despite this continuum of errors, we shall see that it is sufficient to know how to correct a discrete set of errors. To start, we use the fact that $Y = iXZ$ to rewrite equation (3.4.4) as

$$E = \alpha_I I + \alpha_X X + \alpha_Z Z + \alpha_{XZ} XZ, \tag{3.4.5}$$

where the i factor has been absorbed in the α_{XZ} coefficient. We already saw how the X error is the bit-flip error. For the Z error, we refer to it as a phase-flip error, as it has the following effect:

$$Z |0\rangle = |0\rangle \quad \text{and} \quad Z |1\rangle = - |1\rangle . \tag{3.4.6}$$

So, from equation (3.4.5) we see that an arbitrary error can be written as a sum from the set $\{I, X, Z, XZ\}$. Informally, the quantum error correction procedure

involves measurements which collapse the superposition in equation (3.4.5) to either an X, Z, or XZ error; thus, correcting X and Z errors alone is sufficient. This is known as the *discretization* (or, *digitization*) of quantum errors. This process will be made clearer in Section 3.4.2 below, where we discuss the error correction procedure in detail.

For the n-qubit case, we make the common assumption that noise affects each qubit independently. To formally describe this setting, we introduce some useful notation. Let $\mathcal{E}_X$ ($\mathcal{E}_Z$) denote the set of error vectors that corresponds to bit-flip (phase-flip) errors affecting up to q out of n qubits:

$$\mathcal{E}_X = \{e \in \{0,1\}^n : \mathrm{wt}(e) \leq q\} \quad \text{and} \quad \mathcal{E}_Z = \{e' \in \{0,1\}^n : \mathrm{wt}(e') \leq q\}, \tag{3.4.7}$$

where $\mathrm{wt}(\cdot)$ is the Hamming weight. So, for example, a bit-flip error affecting up to q out of n qubits is of the form X^e where $e \in \mathcal{E}_X$. Let $\mathcal{E}_q$ denote the following set, which we refer to as the set of all correctable error vectors,

$$\mathcal{E}_q \coloneqq \{(e, e') : e, e' \in \{0,1\}^n \text{ and } \mathrm{wt}(e) \leq q \text{ and } \mathrm{wt}(e') \leq q\}. \tag{3.4.8}$$

We can easily see that,

$$|\mathcal{E}_q| = |\mathcal{E}_X| \cdot |\mathcal{E}_Z| = \left(\sum_{j=0}^{q} \binom{n}{j} \right)^2. \tag{3.4.9}$$

When it is clear from context, we will sometimes drop the q subscript on $\mathcal{E}_q$.

Now when we say that an n-qubit state has been affected by q arbitrary errors, we mean that q of the qubits have been affected by an error of the form in equation (3.4.5). Then the operator describing q arbitrary errors on an n-qubit state $|\psi\rangle$ is a superposition of Pauli products in the following sense,

$$\sum_{(e,e') \in A} \alpha_{e,e'} X^e Z^{e'} |\psi\rangle , \tag{3.4.10}$$

where A is some subset of $\mathcal{E}_q$ which contains at least one pair (e, e') that corresponds to a weight q error.

3.4.2 The stabilizer formalism

The stabilizer formalism [Got97] can be used to describe a wide and important class of quantum error correcting codes.

To start, we recall that the n-Pauli group $\mathcal{P}_n$ is defined as

$$\mathcal{P}_n \coloneqq \{\pm 1, \pm i\} \times \{I, X, Y, Z\}^{\otimes n}. \tag{3.4.11}$$

An n-qubit stabilizer $\mathcal{S}$ is an abelian (*i.e.*, commutative) subgroup of $\mathcal{P}_n$ such that $-I \notin \mathcal{S}$.

Given a stabilizer, the codespace $C_\mathcal{S}$ is the vector space that is stabilized by $\mathcal{S}$ in the sense that it consists of all n-qubit states which are fixed by every element of $\mathcal{S}$. Equivalently, $C_\mathcal{S}$ is the common $+1$ eigenspace of all elements of $\mathcal{S}$,

$$C_\mathcal{S} = \{|\psi\rangle : S_i |\psi\rangle = |\psi\rangle \,,\ \forall S_i \in \mathcal{S}\}. \tag{3.4.12}$$

When the context is clear, we sometimes drop the $\mathcal{S}$ subscript on $C_\mathcal{S}$. Note that the reason we require $-I \notin \mathcal{S}$ and that the elements of $\mathcal{S}$ commute is to ensure that the vector space stabilized by $\mathcal{S}$ is non-trivial.

A stabilizer is often described by its generators g_i, which are elements of its maximum independent set $\mathcal{S}_g$. Independence is understood in the sense that no element of $\mathcal{S}_g$ can be expressed as a product of other elements in $\mathcal{S}_g$. The generators may be denoted as $\mathcal{S}_g = \{g_1, \ldots, g_m\}$. Alternatively, the generators can be described by a *check matrix*. This is an $m \times 2n$ matrix where the ith row is constructed according to the following rules:

- If g_i contains an I on the jth qubit, then the (i, j) and $(i, n + j)$ entries are 0.
- If g_i contains an X on the jth qubit, then the (i, j) entry is 1 and the $(i, n + j)$ entry is 0.
- If g_i contains a Z on the jth qubit, then the (i, j) entry is 0 and the $(i, n + j)$ entry is 1.
- If g_i contains a Y on the jth qubit, then the (i, j) and $(i, n + j)$ entries are 1.

Since elements of a stabilizer have ± 1 eigenvalues, each generator divides the 2^n-dimensional Hilbert space into two orthogonal subspaces (the $+1$ eigenspace and the -1 eigenspace). Intuitively, we then expect that with $n - k$ generators, the common $+1$ eigenspace (the codespace $C_\mathcal{S}$) has dimension $2^k = 2^{n-(n-k)}$, which is indeed the case.

Proposition 3.4.1. *Suppose a stabilizer $\mathcal{S}$ has $n - k$ generators, so that $\mathcal{S}_g = \{g_1, \ldots, g_{n-k}\}$. Then the vector space $C_\mathcal{S}$ stabilized by $\mathcal{S}$ is 2^k-dimensional.*

Given a stabilizer $\mathcal{S}$, we may have a set of operators that do not belong to $\mathcal{S}$ and yet commute with every element of $\mathcal{S}$. These operators, known as *logical operators*, can be interpreted as a means of computing on encoded states, as they move elements of the codespace around without moving them "out" of $C_\mathcal{S}$. Formally, the logical operators are elements of $N(\mathcal{S}) \setminus \mathcal{S}$, where $N(\mathcal{S})$ is the *normalizer*,

$$N(\mathcal{S}) \coloneqq \{L \in \mathcal{P}_n : LS_iL^\dagger \in \mathcal{S} \ \ \forall S_i \in \mathcal{S}\}. \tag{3.4.13}$$

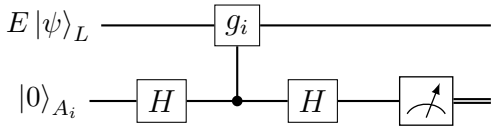

Figure 3.4: Circuit to measure a generator g_i

With $n-k$ generators, there are k pairs of logical operators (or rather, $2k$ logical operators).

We now describe how to perform quantum error correction using a stabilizer $\mathcal{S}$. Since the common starting point for most quantum computations is the $|0\rangle$ state, it is standard to first prepare the codeword $|0\rangle_L \equiv |0^{\otimes k}\rangle_L \in C_{\mathcal{S}}$ by projecting $|0\rangle \equiv |0^{\otimes n}\rangle$ onto the $+1$ eigenspace of all generators (see [NC10, Rof19] for details on how to do this). Note that the other codewords in $C_{\mathcal{S}}$ can be prepared by applying the logical operators to $|0\rangle_L$.

Now suppose that an error E has occurred on our encoded state $|\psi\rangle_L$, leaving us with the state $E|\psi\rangle_L$. The next step is to measure each stabilizer generator g_i, $i = 1, \ldots, n-k$ through the circuit in Figure 3.4. Before measurement, this circuit has the effect of mapping the overall state $E|\psi\rangle_L |0\rangle_{A_i}$ in the following way

$$E|\psi\rangle_L |0\rangle_{A_i} \longrightarrow \frac{1}{2}(I^{\otimes n} + g_i)E|\psi\rangle_L |0\rangle_{A_i} + \frac{1}{2}(I^{\otimes n} - g_i)E|\psi\rangle_L |1\rangle_{A_i}. \tag{3.4.14}$$

We say that a stabilizer generator will detect the error E if it anti-commutes with E, in which case a measurement of the ancilla qubit yields 1; if they commute, the measurement will yield 0. We can equivalently interpret this as: the generator g_i detects the error E if $E|\psi\rangle_L$ is in the -1 eigenspace of g_i (the error has moved the state "out" of the codespace), and the error will not be detected if $E|\psi\rangle_L$ is in the $+1$ eigenspace of g_i. From this perspective, logical operators can be viewed as undetectable errors.

After each of the $n-k$ ancilla qubits $|0\rangle_{A_i}$ have been measured, the results form an $(n-k)$-bit string called the *syndrome*. At this point the syndrome is processed (or, decoded) to determine an appropriate recovery operation R so that $RE|\psi\rangle_L = |\psi\rangle_L$.

At this stage, we can elaborate on a point we raised earlier in Section 3.4.1: the error correction procedure collapses an arbitrary error to a bit-flip or phase-flip error (or a combination of the two), and so it suffices to correct X and Z errors. For simplicity, suppose we have the error $E = \alpha_I I + \alpha_X X + \alpha_Z Z + \alpha_{XZ} XZ$. Further suppose that $|\psi\rangle_L, X|\psi\rangle_L$ are in the $+1$ eigenspace of g_i while $Z|\psi\rangle_L, XZ|\psi\rangle_L$ are in its -1 eigenspace. Then applying the circuit in Figure 3.4, we get the following overall state immediately before measurement,

$$(\alpha_I I^{\otimes n} + \alpha_X X)|\psi\rangle_L |0\rangle_{A_i} + (\alpha_Z Z + \alpha_{XZ} XZ)|\psi\rangle_L |1\rangle_{A_i}. \tag{3.4.15}$$

Then measuring the ancilla qubit has the effect of collapsing the original superposition E. By repeating this procedure with more generators, we end up with $|\psi\rangle_L, X|\psi\rangle_L, Z|\psi\rangle_L$, or $XZ|\psi\rangle_L$, and thus the procedure has effectively collapsed the arbitrary error E into an X, Z or XZ error, where the syndrome can be thought of as indicating which of these errors we have collapsed to. This also applies to the more general case of q arbitrary errors, where the overall error operator is a superposition of Pauli products of the form $X^e Z^{e'}$ where $(e, e') \in \mathcal{E}_q$.

Informally, the *distance* d of a code is the minimum number of errors required to transform one codeword into another. For the formal definition, recall that $N(\mathcal{S}) - \mathcal{S}$ is the set of logical operators, and that the weight of an error $\mathsf{wt}(E)$ is the number qubits being acted upon by a Pauli operator (excluding the identity). Then the distance is defined as,

$$d = \min_{E \in N(\mathcal{S}) - \mathcal{S}} \mathsf{wt}(E). \tag{3.4.16}$$

The distance of a code plays an important role in quantifying how many errors it can correct. If $d \geq 2q + 1$, then the code is capable of correcting q arbitrary errors.

Quantum error correcting codes are often described in the format $[[n, k, d]]$, where n is the number of qubits, k is the number logical qubits, and d is the code's distance.

3.4.3 Calderbank-Shor-Steane (CSS) codes

We now review a well-known category of quantum error correcting codes known as Calderbank-Shor-Steane (CSS) codes. We start by describing their construction from classical linear codes; after this, we examine CSS codes within the stabilizer formalism.

A classical linear code C that encodes k bits into an n-bit codespace is specified by an $n \times k$ matrix G called the *generator matrix*. The entries of G are elements of $\mathbb{Z}_2 = \{0, 1\}$ and its columns are linearly independent and span the codespace. So, for a k-bit string x, represented as a column vector, we get the corresponding codeword by computing Gx.

Equivalently, a classical code can be defined as the kernel of an $(n-k) \times n$ matrix H whose entries are elements of $\mathbb{Z}_2$. That is, the codespace consists of all $x \in \{0, 1\}^n$ such that $Hx = 0$. The matrix H is known as a *parity check* matrix.

The *distance* of a code C is defined as

$$d(C) \coloneqq \min_{x, y \in C, x \neq y} d(x, y),$$

where $d(x, y)$ is the Hamming distance. Since $d(x, y) = \mathrm{wt}(x + y)$, where $\mathrm{wt}(\cdot)$ is the Hamming weight, we equivalently have

$$d(C) = \min_{x \in C, x \neq 0} \mathrm{wt}(x). \tag{3.4.17}$$

If a code has distance $d \geq 2q+1$ for some integer q, then the code is able to correct up to q errors (for a noisy codeword y', there is a unique codeword y that is within Hamming distance q of y').

A classical linear code is denoted as $[n, k, d]$, or simply $[n, k]$ when it is not necessary to specify d. Observe that this resembles the notation for quantum codes, except that the latter case uses double square brackets (to distinguish from the classical case).

Given an $[n, k]$ classical linear code C with generator matrix G and parity check matrix H, one can define the *dual code*, denoted as $C^\perp$, as the code with generator matrix H^T and parity check matrix G^T.

Now suppose C_1 and C_2 are $[n, k_1]$ and $[n, k_2]$ classical linear codes with the property that $C_2 \subseteq C_1$, and C_1 and $C_2^\perp$ both correct q errors. The CSS code, constructed from C_1 and C_2, is then an $[[n, k_1 - k_2]]$ quantum code that can correct up to q bit-flip and q phase-flip errors (*i.e.*, up to q arbitrary errors), and thus it corrects the set $\mathcal{E}_q$, defined in equation (3.4.8).

The CSS code is an example of a stabilizer code where its check matrix is of the form

$$\left[\begin{array}{c|c} H_{C_2^\perp} & 0 \\ 0 & H_{C_1} \end{array}\right]. \tag{3.4.18}$$

So, from the parity check matrices H_{C_1} and $H_{C_2^\perp}$ we can obtain the stabilizer generators, and vice versa. The code C_1 then contributes $n - k_1$ generators, and the code $C_2^\perp$ contributes k_2 generators, for a total of $n - k_1 + k_2$ generators. By Proposition 3.4.1, the codespace has dimension $2^k = 2^{n-(n-k_1+k_2)} = 2^{k_1-k_2}$ which is exactly what we saw earlier.

Codewords in the CSS code are coset states of the form

$$|w + C_2\rangle \equiv \frac{1}{\sqrt{|C_2|}} \sum_{v \in C_2} |w + v\rangle\,, \tag{3.4.19}$$

where $w \in C_1$. Note that since $C_2 \subseteq C_1$, $w + v \in C_1$. The codespace is then the vector space spanned by these coset states. If $w_1, w_2 \in C_1$ are distinct in the sense that $w_1 - w_2 \notin C_2$, then the coset states $|w_1 + C_2\rangle\,, |w_2 + C_2\rangle$ are orthonormal. The number of such cosets is given by $|C_1|/|C_2| = 2^{k_1-k_2}$ which is precisely the dimension of the codespace.

Intuitively, error correction for a CSS code works by using the error correcting properties of its classical linear codes; C_1 is used to correct bit-flip errors and $C_2^\perp$ is used to correct phase-flip errors. To understand this better, we start by examining a noisy codeword; bit-flip errors are described be an n-bit vector e_1 with 1s where bit-flips have occurred and 0s otherwise. That is, $e_1 = (e_{1,1}, \ldots, e_{1,n})$ corresponds to the error $X^{e_1} = X^{e_{1,1}} \otimes \cdots \otimes X^{e_{1,n}}$. Similarly, phase-flip errors are described by a vector e_2. Then the noisy state is

$$X^{e_1} Z^{e_2} |w + C_2\rangle = \frac{1}{\sqrt{|C_2|}} \sum_{v \in C_2} (-1)^{(w+v)\cdot e_2} |w + v + e_1\rangle\,. \tag{3.4.20}$$

First, bit-flips are dealt with by applying the parity check matrix H_{C_1} for the code C_1 and storing the result in ancilla qubits; measuring the ancilla qubits yields an error syndrome $H_{C_1}e_1$ that can be classically processed to determine an appropriate correction operation. After correction, the state is

$$\frac{1}{\sqrt{|C_2|}}\sum_{v\in C_2}(-1)^{(w+v)\cdot e_2}\left|w+v\right\rangle. \tag{3.4.21}$$

By moving to the Hadamard basis, phase-flip errors can be handled in the same way. Indeed, by applying the Hadamard transform, it can be shown that the resulting state takes the form

$$\frac{1}{\sqrt{|C_2^{\perp}|}}\sum_{z\in C_2^{\perp}}(-1)^{w\cdot z}\left|z+e_2\right\rangle. \tag{3.4.22}$$

As in the case of bit-flip errors, we now apply the parity check matrix $H_{C_2^{\perp}}$ for $C_2^{\perp}$ and obtain the error syndrome $H_{C_2^{\perp}}e_2$. After correction, another Hadamard transform is applied to return the state to the original codeword.

For an $[[n,k]]$ CSS code that can correct up to q arbitrary errors, the relationship between the parameters n, k, and q can be formalized with the Gilbert-Varshamov bound for CSS codes. This bound says that an $[[n,k]]$ CSS code which corrects up to q arbitrary errors exists for some k such that

$$\frac{k}{n}\geq 1-2H\left(\frac{2q}{n}\right), \tag{3.4.23}$$

where $H(x)=-x\log(x)-(1-x)\log(1-x)$ is the binary Shannon entropy.

Chapter 4

Device-independent oblivious transfer

In this chapter, we study the oblivious transfer primitive in the bounded-quantum-storage-model. We show how a post-quantum computational assumption can be used to obtain device-independence. This chapter is based on [BY23].

4.1 Introduction

Oblivious transfer (OT), described in terms of its well-known variant *one-out-of-two oblivious transfer* (1-2 OT), involves a sender inputting two bits, and a receiver who is then able to select and receive only one of these bits (see Figure 4.1). For this two-party primitive to be secure, the sender must be unable to learn which of the two bits the receiver chose, and the receiver must be unable to learn both of the bits that were sent. OT is both intriguing and important for the fact that it is universal: a secure implementation of OT enables the implementation of any cryptographic functionality between two parties [Kil88].

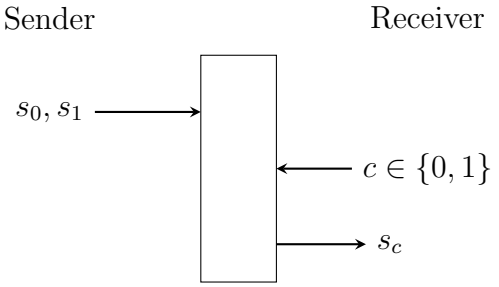

Figure 4.1: 1-2 OT. The sender inputs two bits s_0, s_1. The receiver's choice bit c determines which of the two bits they keep.

However, the search for an unconditionally-secure implementation of oblivious transfer has, unfortunately, been put to rest. In [May97, LC97], it was shown that, even with the power of quantum communication, it is impossible to realize an unconditionally secure implementation of bit commitment—another two-party primitive—thereby also showing the impossibility of unconditionally secure OT. Thus, any secure OT must necessarily be realized with the aid of additional assumptions.

A strong alternative notion of security is *everlasting security*, where we only require our assumptions to hold *during* the execution of the protocol; even if an adversary gains unlimited power after the protocol is over, its security still holds. In the classical setting, [LH19, PVW08] present protocols for OT that are secure by a post-quantum computational assumption, though these classical protocols are not everlastingly secure, as shown in [Unr13]. A different approach can be found in [DFR+07], where it is shown that the *bounded-quantum-storage-model* (BQSM) is sufficient for realizing secure OT. In this setting, it is assumed that the receiver's quantum storage is bounded during the execution of the protocol. Intuitively, what this enables is the following: a dishonest receiver is limited in its capacity to store qubits during the execution of the protocol, thereby forcing a measurement; this disables a large family of attacks and is formally shown to limit the probability of the adversary determining both of the sender's bits to be negligible. Using BQSM to realize OT in this way gives everlasting security because a dishonest receiver who has unlimited quantum storage and processing power after the execution of the protocol does not gain any advantage. In [DFR+07], a protocol for Rand 1-2 OT^{ℓ} (a *randomized* variant of OT that uses ℓ-bit strings, described in Section 4.4.1) is shown to have perfect receiver-security and ε-sender-security. Informally, ε-receiver-security says that the real state of the dishonest sender is ε-close in trace distance to a state that is independent of the choice bit. Similarly, ε-sender-security says that the real state of the dishonest receiver is ε-close to a state that is independent of the information not specified by the choice bit and, additionally, that information appears random. When $\varepsilon = 0$, the security is said to be *perfect*.

The protocol for Rand 1-2 OT^{ℓ} in [DFR+07] assumes that the device is behaving as intended. In reality, a device may be faulty or may have been manufactured by a party with malicious intent, and the resulting behaviour can lead to a security breach in a protocol whose security proof assumes that the device behaves as intended. The assumption that a device is behaving as intended can be relaxed by lifting to the *device-independent* (DI) setting, where no assumptions on the inner workings of the device are made. That is, the device is treated as a black box with which the parties classically interact, and through this classical interaction alone, they can *test* the device to determine if it is behaving as intended.

Device-independent two-party cryptography was explored in [SCA+11, AMPS16] in the form of bit commitment and coin flipping, though it should be noted that these works necessarily focus on protocols with weak security, since they make no additional

assumptions, and thus the impossibility of bit commitment applies. For the device-independent setting, security of the two-party primitive known as *weak string erasure* (from which OT can be constructed) has been proved in the works [KW16, RTK+18] by considering the bounded quantum storage model (and, more generally, a model where the quantum storage may be noisy), though they do not achieve true device-independence, as it is assumed that the devices behave independently and identically in each round (this is the *IID assumption).* The authors in [KW16] do prove security for the non-IID setting, though they assume that, in this case, the dishonest party only makes sequential attacks. Full device-independence (*i.e.*, without the IID assumption) in two-party cryptography is, in general, quite difficult to prove. Despite this, there does exist a fully device-independent protocol for a variant of oblivious transfer, called XOR oblivious transfer, in [KST22]. Soundness of the protocol in [KST22] is quantified through the cheating probabilities of the dishonest parties; that is, the authors show that the cheating probability of each party can be bounded away from 1. As a result of the general difficulty with full device-independence, there exist weaker notions of device-independence such as measurement-device-independence (MDI) where only the measurement devices are treated as black boxes. Security of a protocol for MDI OT has been proved in [RW20].

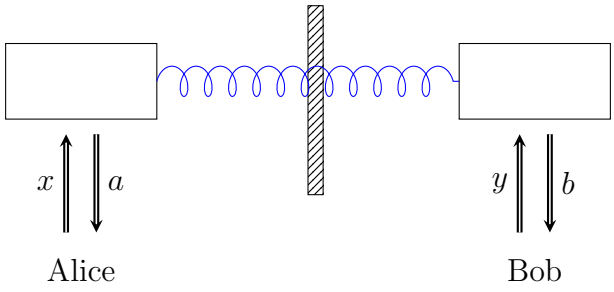

Figure 4.2: Alice and Bob each hold a component of a device. To test the device, they supply their components with classical questions, x, y, and receive classical answers a, b. If the components cannot communicate (indicated by the wall that separates them), and the classical data (x, y, a, b) violates a Bell inequality, then it can be concluded that quantum entanglement is present in the device (indicated by the blue coiled line).

To prove security of device-independent protocols, one often relies on *Bell inequalities*. For instance, the Bell inequality known as the CHSH inequality was used in [KW16, RTK+18] to enable device-independence, while the Mermin-Peres magic square game was used in [KST22]. Simply put, the parties are able to point to the presence of an entangled quantum state in the device by observing that their input and output data violate a Bell inequality, for only by measuring an entangled quantum state is such a violation possible [Bel64, CHSH69]. However, such conclusions

can only hold if all loopholes have been closed. The locality loophole, for instance, describes the fact that a purely classical device, consisting of two components, can violate a Bell inequality if the two components are allowed to communicate. Thus, if one hopes to use a violation of a Bell inequality as an indicator of something quantum in the device, this loophole must obviously be closed, *i.e.*, one must assume non-communication between the components. See Figure 4.2.

To enforce non-communication, there are two options. The first option is to separate the two components with enough distance such that interacting with the device takes less time than a signal of light to travel from one component to the other. The second option is to perfectly shield the components of the device, thus preventing any signal from one component to reach the other. Implementation can then be done by distributing entanglement "on-the-fly" (*i.e.*, un-shield the components after a round, distribute an EPR pair, and then re-shield for the next round). Alternatively, all Bell pairs that are to be used in the protocol could be prepared and distributed prior to the start of the protocol, though this requires that each component be able to store a large number of qubits, and it should be pointed out that OT in the BQSM precisely relies on the fact that a dishonest receiver is unable to store all qubits being used in the protocol.

4.1.1 Contributions

We present a protocol for DI Rand 1-2 OT^{ℓ} and analyze its security. Our protocol is everlastingly secure (because we work in the BQSM) and relies on the difficulty of the Learning with Errors problem (see Section 4.2) instead of a non-communication assumption.

4.1.2 Model

As mentioned above, relying on a violation of a Bell inequality requires the non-communication assumption. For our DI Rand 1-2 OT^{ℓ} protocol, we rely on a single-device self-testing protocol from [MDCA21, MV21] which assumes that the device cannot solve the Learning with Errors (LWE) problem during the execution of the protocol (instead of the non-communication assumption). Note that the assumption that the LWE problem is difficult is a standard computational assumption in post-quantum cryptography [Pei16, Reg05]. This approach of using a post-quantum computational assumption in single-device self-testing was initiated in the groundbreaking work of [Mah18] and [BCM+18]. Our device is then modelled as consisting of two components connected by a quantum channel, making it equivalent to a single device. While the device can then implement any non-local operation on its two components, we shall require that our protocol has an honest implementation which only requires local operations and EPR pairs that are distributed on-the-fly (one

component prepares an EPR pair and sends one qubit of the pair to the other component). We use this post-quantum computational assumption in combination with the BQSM to get everlasting security for our device-independent protocol. The exact assumptions that we make for our DI Rand 1-2 OT^{ℓ} protocol are:

1. *The quantum storage of the receiver is bounded during the execution of the protocol.*

2. *The device is computationally bounded in the sense that it cannot solve the Learning with Errors (LWE) problem during the execution of the protocol.*

3. *The device behaves in an IID manner, i.e., it behaves independently and identically in each round of the protocol.*[1]

4. *For each device component, the input corresponding to the choice of measurement basis, and the resulting output, is communicated only with the party holding that component.*

The first assumption is for the sake of achieving everlastingly secure OT. The second assumption is for achieving a form of device-independence that is more tolerant of communication between the device components. We believe that these first two assumptions are reasonable, as the ability to store qubits is very difficult with current technology, and it is standard to assume that the LWE problem is quantum-computationally hard. The third assumption is made to simplify the security analysis; we discuss the more general setting as a future direction in Section 4.7. In our context, the fourth assumption, discussed in greater detail in Section 4.6, is required for retaining everlasting security. That is, because the sender and receiver do not trust each other, allowing arbitrary communication between the two components of the device makes it possible to honestly execute the protocol but use leaked information to break security at a later time. Assumption 4 remedies this problem while being less strict than the usual non-communication assumption.[2]

To realize this assumption, the device components could initially be given to the sender so that they can verify the validity of the assumption; note that initially

[1] In particular, the device's behaviour cannot be changed between rounds by some party; this is especially relevant for when we estimate the device's winning probability (see Remark 4.5.2). Additionally, for the portion of the main protocol where the device is tested, this IID assumption extends to include the receiver (see Remark 4.6.1). We thank one of the thesis examiners for pointing out this correction.

[2] A similar problem arises in [MDCA21] where the single-device self-testing protocol, which allows arbitrary quantum communication between the device components, is used to obtain a device-independent quantum key distribution (DIQKD) protocol; the problem in the DIQKD setting is that if an eavesdropper can access information sent via the communication channel, then the device could use the channel to signal to the eavesdropper. To remedy this, the authors of [MDCA21] assume that the eavesdropper cannot access the communication channel.

giving the components to the sender will already be required so that they can estimate the device's winning probability (see the end of Section 4.5.2). When the receiver's component is returned to them, the protocol can proceed with the requirement that the receiver shields their component (as opposed to requiring that both components be shielded).

4.1.3 Techniques

We show how to modify the Rand 1-2 OT^{ℓ} protocol from [DFR+07], that was proved to be secure in the BQSM, and the single-device self-testing protocol for a computationally bounded device in [MDCA21] to ensure their compatibility with each other. Then, in our main technical contribution, we show how the two modified protocols can be combined to create a device-independent protocol for Rand 1-2 OT^{ℓ}. It should be noted that the reason the single-device self-testing protocol must be modified is that, in its form in [MDCA21], it assumes that the two parties (Alice and Bob) trust each other, and thus they can work together to test and verify the untrusted device. While this is suitable for the application to device-independent quantum key distribution (DIQKD) in [MDCA21] where the two parties trust each other, it is not suitable for the setting of OT, where the sender and receiver distrust each other and the device.

4.1.4 Outline

We start by reviewing the Learning with Errors problem and trapdoor functions (in Section 4.2 and Section 4.3, respectively) as these concepts are used in the self-testing protocol. The details of the Learning with Errors problem are not crucially used here, but understanding Section 4.3 on trapdoor functions is necessary for seeing how the self-testing protocol works. In Section 4.4 and Section 4.5, we introduce our building blocks: a protocol for oblivious transfer in the BQSM and the self-testing protocol. Then, in Section 4.6, we use these protocols to construct our DI Rand 1-2 OT^{ℓ} protocol (Protocol 4) and then analyze its security.

4.2 Learning with Errors

In this section, we briefly review the Learning with Errors problem. For our purposes, it suffices to only know that the LWE problem is quantum-computationally hard and can be used to construct the trapdoor functions described in Section 4.3, but in the interest of completeness, we introduce the problem here and discuss its hardness. For this short review, we follow [Pei16].

The Learning with Errors problem, introduced in [Reg05], is parametrized by positive integers $n, q \in \mathbb{Z}_+$, and a distribution χ over $\mathbb{Z}$ (the samples drawn from χ are referred to as errors, and thus χ may be referred to as an error distribution).

Definition 4.2.1 (LWE distribution)**.** *For a vector $|s\rangle \in \mathbb{Z}_q^n$, which is called the **secret**, the **LWE distribution** $A_{s,\chi}$ over $\mathbb{Z}_q^n \times \mathbb{Z}_q$ is sampled by choosing $|a\rangle \in \mathbb{Z}_q^n$ uniformly at random, sampling e from χ, and outputting the pair $(|a\rangle, b)$ where*

$$b = \langle a|s\rangle + e \pmod{q}. \tag{4.2.1}$$

There are two versions of the LWE problem: one is to *search* to find the secret $|s\rangle$, given LWE samples; the other is a *decision* type problem, where one must distinguish between LWE samples and uniformly random ones. Both problems require an additional parameter m to specify the number of available samples. While we specify both problems here, it is an important fact that both problems are actually equivalent (up to a polynomial factor of the number of samples).

Definition 4.2.2 (Search-LWE$_{n,q,\chi,m}$)**.** *Given m independent samples*

$$(|a_i\rangle, b_i) \in \mathbb{Z}_q^n \times \mathbb{Z}_q, \tag{4.2.2}$$

drawn from $A_{s,\chi}$ for a uniformly random $|s\rangle \in \mathbb{Z}_q^n$ (fixed for all samples), find $|s\rangle$.

Definition 4.2.3 (Decision-LWE$_{n,q,\chi,m}$)**.** *Given m independent samples*

$$(|a_i\rangle, b_i) \in \mathbb{Z}_q^n \times \mathbb{Z}_q, \tag{4.2.3}$$

where every sample is distributed according to either

(1) $A_{s,\chi}$ for a uniformly random $|s\rangle \in \mathbb{Z}_q^n$ (fixed for all samples),

(2) or, the uniform distribution,

distinguish, with non-negligible advantage, which is the case.

A convenient way of viewing LWE samples is to form a matrix $A \in \mathbb{Z}_q^{n\times m}$ (where the columns are the vectors $|a_i\rangle \in \mathbb{Z}_q^n$) and form a vector $|b\rangle \in \mathbb{Z}_q^m$ whose entries are the $b_i \in \mathbb{Z}_q$, so that

$$|b\rangle = A^T |s\rangle + |e\rangle \mod q \tag{4.2.4}$$

where $|e\rangle$ is sampled from χ^m. Note that for the uniform case of the decision LWE problem, $|b\rangle$ is uniformly random and independent of A. In this formulation of LWE samples, it is particularly easy to see that if we do not have error terms coming from χ, then the LWE problems are easily solved: Gaussian elimination can be used to solve for $|s\rangle$, and in the uniform case of the decision LWE problem, no solution exists with high probability.

We now discuss the hardness of the LWE problem, which is related to two other problems known as the GapSVP and SIVP problems. We do not discuss these latter problems here and instead refer the reader to [Pei16]; we only note, informally, that

no known classical or quantum algorithm exists for efficiently solving these problems. The relation of the LWE problem to the GapSVP and SIVP problems is formalized in [Reg05], where Regev constructed an efficient quantum algorithm that solves the GapSVP and SIVP problems by using an oracle for the LWE problem. Then, through this reduction and the believed difficulty of the GapSVP and SIVP problems, the LWE problem is believed to be difficult.

4.3 Trapdoor functions

In Section 4.5, we shall introduce the single-device self-testing protocol that we will be utilizing. The underlying cryptographic primitive of this protocol is an *extended noisy trapdoor claw-free function family* (ENTCF family) which was introduced, and constructed from the LWE problem, in [BCM+18, Mah18]. We can think of an ENTCF family as an approximation (or "noisy") version of an *extended trapdoor claw-free function family* (ETCF family). The formal definition of an ENTCF family is quite involved and so we refrain from giving it here. Instead, we will follow [Mah18, BCM+18, MV21] to review ETCF families and their properties which will be sufficient for our purposes. When we come to discussing the the self-testing protocol, we will mention the effect of using ENTCF families over ETCF families.

Let $\mathcal{X}$ and $\mathcal{Y}$ be some finite sets that depend on a security parameter λ. For simplicity, take $\mathcal{X} = \{0,1\}^w$ for some $w \in \mathbb{Z}_+$. A *trapdoor claw-free function family* (TCF family) is a collection of function pairs, where each pair is indexed by a key belonging to a finite set $\mathcal{K}_\mathcal{F}$,

$$\mathcal{F} = \{f_{k,b} : \mathcal{X} \to \mathcal{Y} \mid b \in \{0,1\}, k \in \mathcal{K}_\mathcal{F}\}.$$

This function family satisfies the following properties:

(i) For each $k \in \mathcal{K}_\mathcal{F}$, the functions $f_{k,0}$ and $f_{k,1}$ are injective with the same image.

(ii) For a fixed $k \in \mathcal{K}_\mathcal{F}$, every y in the image of the function pair has a unique pair of pre-images (x_0, x_1), called a *claw*, such that

$$f_{k,0}(x_0) = f_{k,1}(x_1) = y. \tag{4.3.1}$$

(iii) It is quantum-computationally difficult to find a claw, but with a y in the image and a trapdoor t_k, which can be sampled together with k, finding a claw is computationally easy. This is the meaning behind "claw-free" (which should not be misunderstood as meaning that a claw does not exist).

(iv) It is quantum-computationally difficult, without access to a trapdoor, to simultaneously compute a preimage x_b and a non-trivial bit string $d \in \{0,1\}^w \setminus \{0^w\}$ such that $d \cdot (x_0 \oplus x_1)$ is a bit, where $\cdot$ is the inner product between bit strings and (x_0, x_1) forms a valid claw. Similarly, it is quantum-computationally difficult to find just a bit of the form $d \cdot (x_0 \oplus x_1)$. We refer to these properties as a whole as the *adapative hardcore bit* property.

To define an *extended trapdoor claw-free function family* (ETCF family), we will need a *trapdoor injective function family* $\mathcal{G} = \{g_{k,b} : \mathcal{X} \to \mathcal{Y} \mid b \in \{0,1\}, k \in \mathcal{K}_{\mathcal{G}}\}$. This function family has the following properties:

(i) For a fixed $k \in \mathcal{K}_{\mathcal{G}}$, the functions $g_{k,0}, g_{k,1}$ are injective but their images are disjoint.

(ii) With each $k \in \mathcal{K}_{\mathcal{G}}$, we can sample a trapdoor t_k such that, given $y = g_{k,b}(x)$, we can easily recover both b and x.

Then an ETCF family consists of $\mathcal{F}, \mathcal{G}$ such that the following property is satisfied:

(i) The two families are computationally indistinguishable in the sense that given $k \in \mathcal{K}_{\mathcal{F}} \cup \mathcal{K}_{\mathcal{G}}$, it is quantum-computationally difficult, without access to the trapdoor, to determine whether $k \in \mathcal{K}_{\mathcal{F}}$ or $k \in \mathcal{K}_{\mathcal{G}}$. This is known as **injective invariance**

It is not yet known how to exactly construct an ETCF family, or a TCF family, as described above. Instead, one can define an approximation called an *extended noisy trapdoor claw-free function family* (ENTCF family) which *can* be constructed from the difficulty of the LWE problem. By approximation, we mean that the range $\mathcal{Y}$ for the functions is now a set of probability densities over $\mathcal{Y}$ (so that the output of a function is a density). This difference, and the effect it has on the above properties, is what leads to the formal definition of an ENTCF family in [Mah18, BCM+18].

4.4 Oblivious transfer in the bounded quantum storage model

In this section, we introduce the first of our two building blocks that will be used to create our DI Rand 1-2 OT^ℓ protocol in Section 4.6. In Section 4.4.1, we describe Rand 1-2 OT^ℓ and present a security definition that comes almost directly from [DFR+07], though it is modified to account for protocol aborts. Then, in Section 4.4.2, we present Protocol 1, which is our modified version of the protocol in [DFR+07] for Rand 1-2 OT^ℓ.

4.4.1 Background of Rand 1-2 OT$^\ell$

We follow [DFR+07] in discussing oblivious transfer in the bounded-quantum-storage model. First, consider 1-out-of-2 oblivious transfer (1-2 OT$^\ell$) which is described in the context of two parties: the sender and the receiver. The sender sends two ℓ-bit strings S_0 and S_1 to the receiver such that the receiver can choose which of the two strings they want to receive; the receiver should be unable to learn anything about the other string, and the sender should be unable to learn which string the receiver chose.

We will be concerned with Rand 1-2 OT$^\ell$ which operates in the same manner as 1-2 OT$^\ell$ except that the two strings S_0 and S_1 are generated uniformly at random during the protocol and output to the sender at the end of the protocol, as opposed to the sender inputting the strings (see Figure 4.3). From Rand 1-2 OT$^\ell$, one can then construct 1-2 OT$^\ell$ [DFSS06].

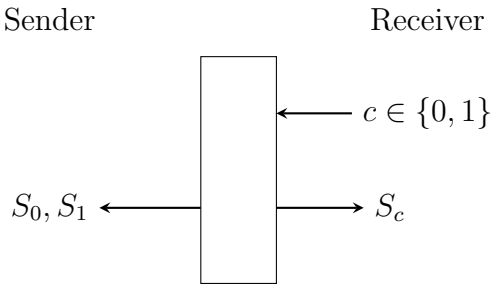

Figure 4.3: Rand 1-2 OT$^\ell$. The strings $S_0, S_1 \in \{0,1\}^\ell$ are generated uniformly at random during the protocol. The receiver's choice bit c determines which of the two strings they get.

Let us now formally define what a Rand 1-2 OT$^\ell$ protocol is and what security for such a protocol means. Definition 4.4.1 below comes almost directly from [DFR+07]; the only change we have made is the inclusion of a way to account for protocol aborts.[3] Regarding notation in this definition, an honest sender is denoted as S and a dishonest sender as $\widetilde{S}$. Similarly, R for an honest receiver and $\widetilde{R}$ for a dishonest receiver. Additionally, C denotes the binary random variable which describes R's choice bit; S_0, S_1 denote the ℓ-bit long random variables describing S's output strings; Y denotes the ℓ-bit long random variable describing R's output string (which should be S_C when both are honest); and Z denotes the binary random variable which describes whether the protocol aborts ($Z = 1$ corresponds to the protocol not aborting). Given a protocol for Rand 1-2 OT$^\ell$, the overall quantum state in the case of a dishonest sender $\widetilde{S}$ is given by the cccq-state $\rho_{CS_0S_1\widetilde{S}}$, and in the case of a dishonest receiver $\widetilde{R}$, the overall quantum state is given by the cccq-state $\rho_{S_{1-C}S_C C\widetilde{R}}$.

[3] The authors would like to thank an anonymous reviewer for suggesting a way of handling aborts in our main protocol.

Definition 4.4.1 (Rand 1-2 OT$^\ell$). *An ε-secure Rand 1-2 OT$^\ell$ is a quantum protocol between a sender S and a receiver R. The sender has no input but the receiver has input $C \in \{0,1\}$ such that for any distribution of C, if S and R follow the protocol, then S gets uniformly random output $S_0, S_1 \in \{0,1\}^\ell$ and R gets $Y = S_C$, except with probability ε, and, with $Z \in \{0,1\}$ describing whether the protocol aborts ($Z = 0$) or does not abort ($Z = 1$), the following two properties hold:*

*ε-**Receiver-security**: If R is honest, then for any $\widetilde{S}$, there exist random variables S'_0, S'_1 such that $Pr[Y = S'_C] \geq 1 - \varepsilon$ and, with $\rho^{Z=1}_{CS'_0S'_1\widetilde{S}}$ being the state when the protocol does not abort,*

$$\Pr(Z = 1) \cdot \Delta(\rho^{Z=1}_{CS'_0S'_1\widetilde{S}}, \rho^{Z=1}_{C} \otimes \rho^{Z=1}_{S'_0S'_1\widetilde{S}}) \leq \varepsilon. \tag{4.4.1}$$

*ε-**Sender-security**: If S is honest, then for any $\widetilde{R}$, there exists a random variable C' such that, with $\rho^{Z=1}_{S_{1-C'}S_{C'}C'\widetilde{R}}$ being the state when the protocol does not abort,*

$$\Pr(Z = 1) \cdot \Delta\big(\rho^{Z=1}_{S_{1-C'}S_{C'}C'\widetilde{R}}, \frac{1}{2^\ell}\mathbb{1} \otimes \rho^{Z=1}_{S_{C'}C'\widetilde{R}}\big) \leq \varepsilon. \tag{4.4.2}$$

***Completeness**: When both parties and the device are honest, the probability of aborting is small,*

$$\Pr(Z = 0) \leq \varepsilon. \tag{4.4.3}$$

Additionally, if any of the above holds for $\varepsilon = 0$, then the corresponding property is said to hold perfectly. If one of the properties only holds with respect to a restricted class $\mathfrak{S}$ of $\widetilde{S}$'s (respectively $\mathfrak{R}$ of $\widetilde{R}$'s), then this property is said to hold and the protocol is said to be secure against $\mathfrak{S}$ (respectively $\mathfrak{R}$).

Receiver-security says that whatever the actions of a dishonest sender, they are just as good as the following actions: cause the protocol to abort or generate the ccq-state $\rho_{S'_0S'_1\widetilde{S}}$ independently of C, inform R of S'_C, and output $\rho_{\widetilde{S}}$. Sender-security says that whatever the actions of a dishonest receiver, they are just as good as causing the protocol to abort or having a ccq-state $\rho_{S_{C'}C'\widetilde{R}}$, informing S of $S_{C'}$ and an independent uniformly distributed ℓ-bit string, and outputting $\rho_{\widetilde{R}}$.

A protocol that satisfies Definition 4.4.1 is then a secure implementation of Rand 1-2 OT$^\ell$ with the exception that a dishonest sender may influence the distribution of S_0 and S_1, and a dishonest receiver may influence the distribution of the string of their choice, though as pointed out in [DFR+07], this is acceptable for a straightforward construction of the standard 1-2 OT$^\ell$. For an explanation of why the existence of S'_0 and S'_1 in the statement of receiver-security is necessary, the reader is referred to the discussion after Definition 3 in [DFR+07].

4.4.2 Modified Rand 1-2 OT$^\ell$ protocol

We now present a quantum protocol for Rand 1-2 OT$^\ell$ that has been slightly modified from [DFR+07] for the purpose of compatibility with the self-testing protocol that will be introduced in Section 4.5. This modified protocol, given as Protocol 1 below, differs from the Rand 1-2 OT$^\ell$ protocol in [DFR+07] in that their protocol has the sender use conjugate coding to send quantum states to the receiver; they then show that to prove sender security, it suffices to prove sender security for an EPR-based version of their protocol, and thus do so for receivers with bounded quantum storage. Our protocol, however, explicitly uses Bell pairs; furthermore, our protocol allows for the use any of the four possible Bell pairs (recall equation (3.3.1)) while the EPR-based version of the protocol in [DFR+07] only needed to use the $|\phi^{(0,0)}\rangle$ Bell pair.

These modifications are minor and so the proofs in [DFR+07] for perfect receiver-security and ε-sender-security against quantum-memory-bounded receivers largely carries over to our protocol. Nevertheless, for the sake of being explicit, we shall give the proofs here and note how our modifications affect them.

In Protocol 1, we let $\mathcal{F}$ be a fixed two-universal class of hash functions from $\{0,1\}^n$ to $\{0,1\}^\ell$, where ℓ is to be determined later. Note that we can apply a function $f \in \mathcal{F}$ to a n'-bit string with $n' < n$ by padding it with zeros. For example, suppose $x \in \{0,1\}^n$ and $I \subseteq \{1,\dots,n\}$ is such that $|I| = n' < n$. Then $x|_I$ denotes the n'-bit string that consists of x_i where $i \in I$. So, if we write $f(x|_I)$, it will be assumed that $x|_I$ has been padded with zeros to form an n-bit string.

Protocol 1: Rand 1-2 OT$^\ell$ with Bell pairs

1. A device prepares n uniformly random Bell pairs $|\phi^{(\alpha_i,\beta_i)}\rangle$, $i = 1,\dots,n$, where the first qubit of each pair goes to S along with the string α, and the second qubit of each pair goes to R along with the string β.

2. R measures all qubits in the basis $y = [\texttt{Computational, Hadamard}]_c$ where c is R's choice bit. Let $b \in \{0,1\}^n$ be the outcome. R then computes $b \oplus w^\beta$, where the i-th entry of w^β is defined by

$$w_i^\beta := \begin{cases} 0, & \text{if } y = \texttt{Hadamard} \\ \beta_i, & \text{if } y = \texttt{Computational} \end{cases}$$

3. S picks uniformly random $x \in \{\texttt{Computational, Hadamard}\}^n$, and measures the i-th qubit in basis x_i. Let $a \in \{0,1\}^n$ be the outcome. S then computes $a \oplus w^\alpha$, where the i-th entry of w^α is defined by

$$w_i^\alpha := \begin{cases} \alpha_i, & \text{if } x_i = \texttt{Hadamard} \\ 0, & \text{if } x_i = \texttt{Computational} \end{cases}$$

4. S picks two uniformly random hash functions $f_0, f_1 \in \mathcal{F}$, announces x and f_0, f_1 to R, and outputs $s_0 \coloneqq f_0(a \oplus w^\alpha|_{I_0})$ and $s_1 \coloneqq f_1(a \oplus w^\alpha|_{I_1})$ where $I_r \coloneqq \{i : x_i = [\texttt{Computational, Hadamard}]_r\}$.

5. R outputs $s_c = f_c(b \oplus w^\beta|_{I_c})$.

The purpose in defining w^α, w^β is primarily for compact notation. Recalling equation (3.3.4),

$$\begin{cases} a \oplus b = \beta, & \text{if } x = y = \texttt{Computational} \\ a \oplus b = \alpha, & \text{if } x = y = \texttt{Hadamard} \end{cases}$$

we see that, in the honest case of Protocol 1, we have $a \oplus w^\alpha = b \oplus w^\beta$ when $x = y$.

Observe that without the bounded quantum storage assumption, the receiver can easily break the security of Protocol 1 by choosing to not measure their qubits in step 2 and instead store them until step 4, where the sender publishes their measurement bases. At this point, the receiver can copy the sender's measurement bases, thus allowing the receiver to learn both s_0 and s_1.

Note that there is no step in Protocol 1 where an abort can occur, and so the probability of not aborting is one, $\Pr(Z = 1) = 1$. Hence, in the case that both parties are honest, we easily get $\Pr(Z = 0) = 0$ which satisfies the completeness condition of Definition 4.4.1.

Although we use Bell pairs in our protocol, and conjugate coding is used in the protocol in [DFR+07], the intuition behind perfect receiver-security in both protocols is the same, as is the strategy for proving it: the non-interactivity of both protocols means that a dishonest sender is unable to learn the receiver's choice bit. To prove perfect receiver-security, we consider the scenario where a dishonest sender executes the protocol with a receiver that has unbounded quantum memory and thus can compute S'_0, S'_1. Consequently, the only modification that must be made to the proof of Proposition 4.5 (perfect receiver-security) in [DFR+07] is that we must take equation (3.3.4) into account.

Proposition 4.4.2. *Protocol 1 is perfectly receiver-secure.*

Proof: Since the probability of not aborting is one in Protocol 1, we can drop the superscript $Z = 1$ on our overall quantum state (since the superscript $Z = 1$ denotes the state when the protocol does not abort; see Definition 4.4.1).

The ccq-state $\rho_{CY\widetilde{S}}$ is defined by the experiment where $\widetilde{S}$ interacts with an honest memory-bounded R. Now, in a new Hilbert space, we define the ccccq-state $\hat{\rho}_{\hat{C}\hat{Y}\hat{S}'_0\hat{S}'_1\widetilde{S}}$ according to a different experiment.

In this different experiment, we let $\widetilde{S}$ interact with a receiver that has unbounded quantum memory. Let V^α, V^β be the strings that describe the random and independent choices of α, β which, together, describe which of the four Bell states are used in each round (see equation (3.3.2)). Let A be the string the sender gets after measuring the i-th qubit in the basis x_i for $i = 1, \ldots, n$. Define the string W^α in terms of V^α in the same way w^α is defined in terms of α in step 3.

Now, the receiver waits to receive x and then also measures the i-th qubit in the basis x_i for $i = 1, \ldots, n$. Let B be resulting string, and define the string W^β in terms of V^β in the same way w^β is defined in terms of β in step 2.

Define $\hat{S}'_0 = f_0(A \oplus W^\alpha|_{I_0})$ and $\hat{S}'_1 = f_1(A \oplus W^\alpha|_{I_1})$, where we recall that

$$A \oplus W^\alpha = B \oplus W^\beta. \tag{4.4.4}$$

Sample $\hat{C}$ according to P_C and set $\hat{Y} = \hat{S}'_{\hat{C}}$. It follows by construction that $\hat{\rho}_{\hat{C}}$ is independent of $\hat{\rho}_{\hat{S}'_0 \hat{S}'_1 \widetilde{S}}$ and that $\Pr[\hat{Y} \neq \hat{S}'_{\hat{C}}] = 0$.

It now remains to argue that $\hat{\rho}_{\hat{C}\hat{Y}\widetilde{S}} = \rho_{CY\widetilde{S}}$ so that the corresponding S'_0 and S'_1 also exist in the original experiment. But, this is satisfied since the only difference between the two experiments is when and in what basis the qubits at position $i \in I_{1-C}$ are measured, which does not affect $\rho_{CY\widetilde{S}}$ respectively $\hat{\rho}_{\hat{C}\hat{Y}\widetilde{S}}$. ∎

Now, for security against dishonest receivers, we restrict ourselves to receivers whose quantum storage is bounded during the execution of the protocol.

Definition 4.4.3 ([DFR+07])**.** *Let $\mathfrak{R}_\gamma$ denote the set of all possible quantum dishonest receivers $\widetilde{R}$ in Protocol 1 (or Protocol 4) which have quantum memory of size at most γn when step 4 of Protocol 1 (or $\gamma n'$ when step 8 of Protocol 4) is reached.*

The proof of ε-sender-security against $\mathfrak{R}_\gamma$, given below, is nearly identical to the proof of ε-sender-security in [DFR+07]. However, because we allow for the use of all four Bell states, we need to show that

$$H^\varepsilon_{\min}(K|X) = H^\varepsilon_{\min}(A|X)$$

where A is the random variable describing the outcome of the sender measuring their part of the quantum state in the random basis X, and $K \coloneqq A \oplus W^\alpha$. Observe that if we only used Bell pairs of the form $|\phi^{(0,0)}\rangle$, then $W^\alpha = 0$, and so our proof would fully reduce to the proof in [DFR+07], as expected.

Proposition 4.4.4. *Protocol 1 is ε-sender-secure against $\mathfrak{R}_\gamma$ for a negligible (in n) ε if there exists a $k > 0$ such that $kn \leq n/4 - 2\ell - \gamma n$.*

Proof: Since the probability of not aborting is one in Protocol 1, we can drop the superscript $Z = 1$ on our overall quantum state (since the superscript $Z = 1$ denotes the state when the protocol does not abort; see Definition 4.4.1).

Now consider the quantum state in Protocol 1 after $\widetilde{R}$ has measured all but γn of their qubits. Let V^α, V^β be the random variables that describe the random and independent choices of α, β which, together, describe which of the four Bell states are used in each round (see equation (3.3.2)). Define W^α in terms of V^α in the same way w^α is defined in terms of α in Item 3. Let A be the random variable that describes the outcome of the sender measuring their part of the state in the random basis X, and let E be the random state that describes $\widetilde{R}$'s part of the state. Let F_0 and F_1 be the random variables that describe the random and independent choices of $f_0, f_1 \in \mathcal{F}$.

Choose $\lambda, \lambda', \kappa$ all positive, but small enough such that

$$(1/4 - \lambda - 2\lambda' - \kappa)n - 1 - 2\ell \geq \gamma n, \tag{4.4.5}$$

so that

$$(1/4 - \lambda - 2\lambda')n - 1 - \ell \geq \gamma n + \ell + \kappa n. \tag{4.4.6}$$

From the uncertainty relation Lemma 3.3.26, we know that

$$H^\varepsilon_{\min}(A|X) \geq (1/2 - 2\lambda)n, \tag{4.4.7}$$

for ε exponentially small in n. Let $A_r = A|_{I_r}$ and $W^\alpha_r = W^\alpha|_{I_r}$ where

$$I_r = \{i : X_i = [\texttt{Computational, Hadamard}]_r\}.$$

Define $K = A \oplus W^\alpha$ and let K_r be $K|_{I_r}$ padded with zeros so that it will make sense to apply F_r. Now, to see that $H^\varepsilon_{\min}(K|X) = H^\varepsilon_{\min}(A|X)$, it suffices to observe that, for a single round, $P_{AX\mathcal{E}}(a, x) = P_{KX\mathcal{E}}(k, x)$ (recall equation (3.3.56)).

$$P_{AX\mathcal{E}}(a, x) = P_{A\mathcal{E}|X}(a|x = 0) \cdot P_X(x = 0) + P_{A\mathcal{E}|X}(a|x = 1) \cdot P_X(x = 1), \tag{4.4.8}$$

For the first term, note that $x = 0$ corresponds to $w^\alpha = 0$, and so,

$$\begin{aligned} P_{A\mathcal{E}|X}(a|x = 0) &= P_{KW^\alpha\mathcal{E}|X}(k \oplus w^\alpha = a|x = 0), && (4.4.9)\\ &= P_{K\mathcal{E}|X}(k = a \oplus w^\alpha|w^\alpha, x = 0) \cdot P_{W^\alpha}(w^\alpha|x = 0), && (4.4.10)\\ &= P_{K\mathcal{E}|X}(k|x = 0). && (4.4.11) \end{aligned}$$

For the second term, $x = 1$ corresponds to $w^\alpha = \alpha$ where α is a uniformly random bit, and so,

$$\begin{aligned} P_{A\mathcal{E}|X}(a|x = 1) &= P_{KW^\alpha\mathcal{E}|X}(k \oplus w^\alpha = a|x = 1), && (4.4.12)\\ &= P_{K\mathcal{E}|X}(k = a \oplus w^\alpha|w^\alpha, x = 1) \cdot P_{W^\alpha}(w^\alpha|x = 1), && (4.4.13) \end{aligned}$$

$$= \left(\sum_{j=1}^{2} P_{K\mathcal{E}|X}(k=a|x=1) \cdot \frac{1}{2} \right), \tag{4.4.14}$$

$$= P_{K\mathcal{E}|X}(k|x=1). \tag{4.4.15}$$

Thus, we indeed have $P_{AX\mathcal{E}}(a,x) = P_{KX\mathcal{E}}(k,x)$ and hence

$$H_{\min}^{\varepsilon}(K|X) = H_{\min}^{\varepsilon}(A|X) \geq (1/2 - 2\lambda)n. \tag{4.4.16}$$

Furthermore, we have $H_{\min}^{\varepsilon}(K_0K_1|X) \geq (1/2 - 2\lambda)n$.

Therefore, by Lemma 3.3.27, there exists a binary random variable C' such that for $\varepsilon' = 2^{-\lambda' n}$, it holds that

$$H_{\min}^{\varepsilon+\varepsilon'}(K_{1-C'}|X,C') \geq \frac{(1/2-2\lambda)n}{2} - 1 - \log\left(\frac{1}{2^{-\lambda' n}}\right), \tag{4.4.17}$$

$$\geq (1/4 - \lambda - \lambda')n - 1. \tag{4.4.18}$$

It is clear that we can condition on the independent $F_{C'}$ and use the chain rule Lemma 3.3.25 to obtain

$$H_{\min}^{\varepsilon+2\varepsilon'}(K_{1-C'}|XF_{C'}(K_{C'})F_{C'},C') \tag{4.4.19}$$

$$\geq H_{\min}^{\varepsilon+\varepsilon'}(K_{1-C'}F_{C'}(K_{C'})|XF_{C'}C') - H_{\max}(F_{C'}(K_{C'})|XF_{C'}C') - \lambda' n \tag{4.4.20}$$

$$\geq (1/4 - \lambda - 2\lambda')n - 1 - H_{\max}(F_{C'}(K_{C'})|XF_{C'}C') \tag{4.4.21}$$

$$\geq (1/4 - \lambda - 2\lambda')n - 1 - \ell \tag{4.4.22}$$

$$\geq \gamma n + \ell + \kappa n, \tag{4.4.23}$$

where we used equation (3.3.55) in the second-last line, and the last line follows by the choice of $\lambda, \lambda', \kappa$.

We write $S_0 = F_0(K_0)$ and $S_1 = F_1(K_1)$. Now, by setting $U = XS_{C'}F_{C'}C'$ and then applying Theorem 3.3.28, we get

$$\begin{aligned}
&\Delta(\rho_{S_{1-C'}F_{1-C'}XS_{C'}F_{C'}C'E}, \tfrac{1}{2^\ell}\mathbb{1} \otimes \rho_{F_{1-C'}XS_{C'}F_{C'}C'E}) \\
&\qquad \leq \frac{1}{2} 2^{-\frac{1}{2}(H_{\min}^{\varepsilon+2\varepsilon'}(K_{1-C'}|XS_{C'}F_{C'}C') - \gamma n - \ell)} + (2\varepsilon + 4\varepsilon') \\
&\qquad \leq \frac{1}{2} 2^{-\frac{1}{2}\kappa n} + 2\varepsilon + 4\varepsilon'
\end{aligned}$$

which is negligible, and thus we have ε-sender security. ∎

4.5 Device-independence from computational assumptions

We now discuss our second building block: a self-testing protocol for a single (quantum) device that relies on a computational assumption. We start in Section 4.5.1 with a discussion of the single-device self-testing protocol from [MDCA21], Protocol 2, to outline how it works and to present the self-testing guarantee of the protocol, Theorem 4.5.1. Then, in Section 4.5.2 we present our single-device self-testing protocol, Protocol 3, which is a slightly modified version of Protocol 2 that is appropriate for the setting where there are two parties that distrust each other, as in the case of OT.

4.5.1 Original single-device self-testing protocol

In short, the single-device self-testing protocol from [MDCA21, MV21] uses computational assumptions to certify that a device, consisting of two components but connected by a quantum channel, has correctly prepared a quantum state and measured it according to the bases specified by the verifier(s). This protocol was first presented in [MV21] as a protocol for a single verifier. Then, in [MDCA21], the protocol was stated in a form that involves two verifiers so that it could be used to formulate a device-independent quantum key distribution protocol that generates an information-theoretically secure key; this latter form of the protocol, presented as Protocol 2 below, will be our starting point.

The computational assumption made in this protocol is that the device cannot solve the Learning with Errors problem during the execution of the protocol; specifically, the underlying cryptographic primitive is an ENTCF family (see Section 4.3). This computational assumption replaces the non-communication assumption that is necessary for typical self-testing protocols which rely on the violation of a Bell inequality. While non-local operations are then possible, there is an honest implementation which only requires local operations and EPR pairs that are distributed on-the-fly.

A brief description of this honest implementation is outlined throughout Protocol 2, and is further described in the discussion of Protocol 2's winning condition. For a detailed description of the device's honest behaviour, the reader is referred to the appendix of [MDCA21]; for further details, the reader is referred to [MV21].

Protocol 2: Single-device self-testing with two verifiers

1. Alice chooses a basis, called the *state basis*, $\theta^A \in \{\texttt{Computational}, \texttt{Hadamard}\}$ uniformly at random and generates a key k^A together with a trapdoor t^A, where the generation procedure for k^A and t^A depends on θ^A and a security parameter η. Likewise, Bob generates θ^B, k^B, and t^B. The keys are such that the device

cannot efficiently compute the state bases θ^A, θ^B from the keys k^A, k^B. Alice and Bob send the keys k^A, k^B to the device.

2. Alice and Bob receive strings c^A and c^B, respectively, from the device.
Honest behaviour: Prepare a product state $|\psi^A\rangle |\psi^B\rangle$, where $|\psi^A\rangle$ and $|\psi^B\rangle$ are, respectively, functions of k^A and k^B. Measure part of $|\psi^A\rangle$ in the computational basis to obtain a string c^A and send it to Alice, keeping the remainder of the state. Similarly, obtain c^B from $|\psi^B\rangle$ and send it to Bob.

3. Alice and Bob independently choose a *challenge type* $CT \in \{\texttt{a},\texttt{b}\}$ uniformly at random and send them to the device.

4. If $CT = \texttt{a}$ for Alice (Bob):

   i) Alice (Bob) receives string z^A (z^B) from the device.
   Honest behaviour: Measure the remainder of the state $|\psi^A\rangle$ ($|\psi^B\rangle$) in the computational basis and send the resulting string z^A (z^B) to Alice (Bob).

5. If $CT = \texttt{b}$ for Alice (Bob):

   i) Alice (Bob) receives string d^A (d^B) from the device.
   Honest behaviour: Measure the remainder of the state $|\psi^A\rangle$ ($|\psi^B\rangle$), except for one qubit of the state, in the Hadamard basis and send the resulting string d^A (d^B) to Alice (Bob).

   ii) Alice (Bob) chooses a uniformly random measurement basis $x \in \{\texttt{Computational, Hadamard}\}$ ($y \in \{\texttt{Computational, Hadamard}\}$) and sends it to the device.

   iii) Alice (Bob) receives an answer bit a (b) from the device. Alice (Bob) also receives the bit h^A (h^B) from the device.
   Honest behaviour: when $CT = \texttt{b}$ for Alice and Bob, the remaining state has two qubits. In place of a controlled-Z operation, apply the circuit in Figure 4.4 by using a single EPR pair that has been distributed on-the-fly, and return the bits h^A and h^B to Alice and Bob, respectively. Then apply a Hadamard gate on the second qubit. Then measure the first qubit in the basis x and the second qubit in the basis y, obtaining outcomes $a, b \in \{0, 1\}$, respectively. Send a to Alice and b to Bob.

Before discussing the winning condition of Protocol 2, we note that the purpose of using the circuit in Figure 4.4 at step 5(iii) instead of a controlled-Z gate is to replace the need for non-local operations. By using the circuit in Figure 4.4, an honest device can succeed in Protocol 2 with only local operations and EPR pairs

that are distributed-on-the-fly. Note that if $|\psi\rangle_{AB}$ is the state just before the circuit is applied, then after the circuit we have

$$X^{h^A} Z^{h^B} \otimes X^{h^B} Z^{h^A} CZ \, |\psi\rangle_{AB} \,, \tag{4.5.1}$$

where CZ is the controlled-Z gate. The operator $X^{h^A} Z^{h^B} \otimes X^{h^B} Z^{h^A}$ can be undone by having the device's components communicate the bits h^A and h^B with each other and then applying the appropriate local operations, though this is undesirable as the purpose in using the circuit in Figure 4.4 is to allow an honest device to succeed without communication between its components. So, rather than having the device undo the operator $X^{h^A} Z^{h^B} \otimes X^{h^B} Z^{h^A}$, the checks that Alice and Bob need to perform are modified to account for the presence of the operator. It is also worth noting that by using the circuit in Figure 4.4, the actions of the component held by Alice and the component held by Bob are independent of each other. This means that the honest implementation described above can be extended to the case where Alice and Bob choose different challenge types CT; that is, each component acts according to the challenge type it received.

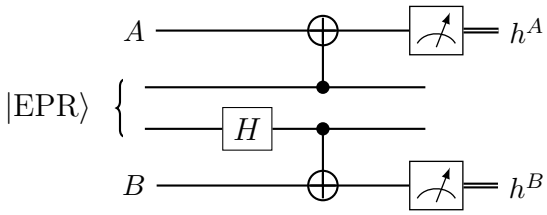

Figure 4.4: Circuit from [MDCA21] to replace controlled-Z gate

We now describe the winning conditions of Protocol 2. In doing this, we briefly discuss the honest behaviour of the device that was outlined in Protocol 2. Although the self-testing protocol formally relies on ENTCF families, our discussion will only make reference to ETCF families for the sake of simplicity. The only consequence of this, as noted in [MDCA21], is that an honest device cannot perfectly satisfy the winning conditions; it can only do so with probability $1 - \mathsf{negl}(\eta)$ for some security parameter η, which is reflected in the self-testing guarantee (Theorem 4.5.1). We remind the reader that a detailed description of the device's honest behaviour, and thus why the winning condition is as it is, can be found in the appendix of [MDCA21] and, for further details, one should see [MV21].

For the device to win a given round of Protocol 2, several checks must be passed. If these checks pass, then Alice and Bob set a variable W to `pass`; otherwise, $W = \texttt{fail}$.

We focus on Alice's component of the device for now; the honest behaviour of Bob's component follows similarly. Alice's state basis θ^A determines which family her key k^A will belong to: $k^A \in \mathcal{K}_{\mathcal{G}}$ if $\theta^A = \texttt{Computational}$, and $k^A \in \mathcal{K}_{\mathcal{F}}$ if

$\theta^A = \texttt{Hadamard}$. When the device receives k^A, it evaluates the corresponding function pair $(f_{k^A,0}, f_{k^A,1})$ over the uniform superposition of the domain $\mathcal{X}$, resulting in the state,

$$|\psi_0^A\rangle = \sum_{b\in\{0,1\}} \sum_{x\in\mathcal{X}} |b\rangle\,|x\rangle\,|f_{k^A,b}(x)\rangle\,, \tag{4.5.2}$$

where we have dropped the normalization factor. The string c^A, which is returned to Alice, is then obtained by measuring the last register of this state in the computational basis.

In the case where $k^A \in \mathcal{K}_{\mathcal{F}}$, the post-measurement state is, up to normalization,

$$|\psi_1^A\rangle = |0\rangle\,|x_0^A\rangle + |1\rangle\,|x_1^A\rangle\,, \tag{4.5.3}$$

where (x_0^A, x_1^A) forms a claw such that $f_{k^A,0}(x_0^A) = f_{k^A,1}(x_1^A) = c^A$.

In the other case, where $k^A \in \mathcal{K}_{\mathcal{G}}$, there is a unique pair $(\hat{b}^A, \hat{x}^A) \in \{0,1\} \times \mathcal{X}$ such that $f_{k^A,\hat{b}^A}(\hat{x}^A) = c^A$, and so the post-measurement state is

$$|\psi_1^A\rangle = |\hat{b}^A\rangle\,|\hat{x}^A\rangle\,. \tag{4.5.4}$$

If $CT = \texttt{a}$: The device measures $|\psi_1^A\rangle$ in the computational basis with output z^A. Let z_1^A be the first bit of z^A, and z_r^A be the remainder of the string. Regardless of whether $k^A \in \mathcal{K}_{\mathcal{F}}$ or $k^A \in \mathcal{K}_{\mathcal{G}}$, Alice checks if $f_{k^A,z_1^A}(z_r^A) = c^A$. If this check passes for both Alice and Bob, set $W = \texttt{pass}$.

If $CT = \texttt{b}$: The honest behaviour of the device in step 5(i) is to measure the second register of $|\psi_1^A\rangle$ in the Hadamard basis to obtain the outcome d^A. If $k^A \in \mathcal{K}_{\mathcal{F}}$, then the post-measurement state is

$$|\psi_2^A\rangle = |0\rangle + (-1)^{d^A\cdot(x_0^A \oplus x_1^A)}\,|1\rangle\,. \tag{4.5.5}$$

If $k^A \in \mathcal{K}_{\mathcal{G}}$, then the post-measurement state is simply

$$|\psi_2^A\rangle = |\hat{b}^A\rangle\,. \tag{4.5.6}$$

When $CT = \texttt{b}$ for Alice *and* Bob, the overall state held by the device in step 5(iii), just before measurement is, up to a global phase, given by

$$|\psi_3\rangle = X^{h^A} Z^{h^B} \otimes X^{h^B} Z^{h^A} (\mathbb{1} \otimes \mathsf{H}) CZ\,|\psi_2^A\rangle\,|\psi_2^B\rangle\,, \tag{4.5.7}$$

where the Hadamard gate has been commuted past the operator $X^{h^A} Z^{h^B} \otimes X^{h^B} Z^{h^A}$.

Unless $k^A \in \mathcal{K}_{\mathcal{F}}$ and $k^B \in \mathcal{K}_{\mathcal{F}}$, the state $|\psi_3\rangle$ is a product state. Together, Alice and Bob can determine precisely what product state the honest device has prepared. Indeed, with the trapdoor t^A, Alice can easily compute (x_0^A, x_1^A) or $\hat{x}^A$ from c^A (by

the properties of the ENTCF family), and likewise for Bob. Then, with h^A, h^B, d^A, and d^B, Alice and Bob have everything they need to determine $|\psi_3\rangle$. Knowing $|\psi_3\rangle$, Alice and Bob now determine what answers an honest device would have returned to their measurement basis questions x and y, and then check if the answers a and b returned by the device are the same; if they are, they set $W = \texttt{pass}$.

Now if $k^A \in \mathcal{K}_{\mathcal{F}}$ and $k^B \in \mathcal{K}_{\mathcal{F}}$ (*i.e.*, $\theta^A = \theta^B = \texttt{Hadamard}$), then the state held by the device in step 5(iii), just before measurement, turns out to be one of the four Bell states, up to a global phase,

$$|\phi^{(d^A\cdot(x_0^A\oplus x_1^A),\, d^B\cdot(x_0^B\oplus x_1^B))}\rangle\,. \tag{4.5.8}$$

Recalling equation (3.3.2),

$$|\phi^{(\alpha,\,\beta)}\rangle = (Z^\alpha X^\beta \otimes \mathbb{1})\frac{|00\rangle + |11\rangle}{\sqrt{2}}, \tag{4.5.9}$$

we see that,

$$\alpha = d^A \cdot (x_0^A \oplus x_1^A),$$
$$\beta = d^B \cdot (x_0^B \oplus x_1^B).$$

Recall that, with the trapdoor t^A, Alice can easily compute (x_0^A, x_1^A) from c^A, and likewise for Bob, and thus they can compute α and β. Recalling relation (3.3.4), Alice and Bob perform the following checks:

- If $x = y = \texttt{Computational}$, check if $a \oplus b = d^B \cdot (x_0^B \oplus x_1^B)$.
- If $x = y = \texttt{Hadamard}$, check if $a \oplus b = d^A \cdot (x_0^A \oplus x_1^A)$.

If one of the above checks pass, or if $x \neq y$, then set $W = \texttt{pass}$.

Now that the winning condition of Protocol 2 has been described, let us give the self-testing guarantee, stated as Theorem 4.5.1 below. Note that the guarantee is essentially stating that any computationally bounded device that wins in Protocol 2 must have performed single qubit measurements on a Bell state to obtain the results returned to the verifier. Recall that $\{M_x^a\}_{a\in\{0,1\}}$ denotes the single-qubit measurement in the basis x (see Section 3.3.1), and note that for questions x, y, we denote the 4-outcome measurement used by the device to obtain answers a, b by $\{P_{x,y}^{(a,b)}\}_{a,b\in\{0,1\}}$. We denote the state held by the device by $\sigma^{(\alpha,\beta)}$, where α, β are the bits that label which of the Bell states the device should have prepared, as in equation (3.3.2).

Theorem 4.5.1 ([MDCA21]). *Consider a device that wins Protocol 2 with probability* $1-\delta$ *and make the LWE assumption. Let* η *be the security parameter used in the protocol,* $\alpha, \beta \in \{0,1\}$ *be the bits that label the Bell state as in equation (3.3.2),* $\mathcal{H}$ *be the device's physical Hilbert space, and* $\mathcal{H}'$ *be some ancillary Hilbert space. Then there exists an isometry* $V : \mathcal{H} \to \mathbb{C}^4 \otimes \mathcal{H}'$ *and some state* $\zeta_{\mathcal{H}'}^{(\alpha,\beta)}$ *such that, in the case* $\theta^A = \theta^B = \texttt{Hadamard}$*, the following holds:*

$$V P_{x,y}^{(a,b)} \sigma^{(\alpha,\beta)} P_{x,y}^{(a,b)} V^\dagger \approx_{\mathcal{O}(\delta^r)+\mathsf{negl}(\eta)} \left((M_x^a \otimes M_y^b) \left|\phi^{(\alpha,\beta)}\right\rangle \left\langle\phi^{(\alpha,\beta)}\right| (M_x^a \otimes M_y^b) \right) \otimes \zeta_{\mathcal{H}'}^{(\alpha,\beta)},$$

where r *is some small constant arising in the proof.*

In accordance with the honest implementation described in the discussion of the winning condition, rounds where $\theta^A = \theta^B = \texttt{Hadamard}$ and $CT = \texttt{b}$ are referred to as `Bell` rounds, while all other rounds are referred to as `Product` rounds.

It is worth mentioning why Theorem 4.5.1 only makes a statement about `Bell` rounds. The crucial point is that Alice and Bob will always know whether it is a `Bell` round or a `Product` round, but the computationally bounded device, which does not have access to θ^A and θ^B, cannot determine what round it is in. This stems from the fact that the device is never given the trapdoor information t^A, t^B and thus, by injective invariance of the ENTCF family (property (i)), it is quantum-computationally hard for the device to determine which families the keys k^A, k^B come from, and hence the type of round cannot be determined by the device.

Furthermore, Alice and Bob will always know precisely what Bell state or what product state the honest device should have prepared and, consequently, what answers should be returned in response to their measurement questions. So, to succeed and pass the checks of Alice and Bob, the device is forced to behave honestly. Thus, the self-testing guarantee only mentions `Bell` rounds.

Since `Bell` rounds are the ones where an honest device will prepare a Bell state, they are of primary interest. Recall that, in such a round, the state held by the device in step 5(iii), just before measurement, is

$$\left|\phi^{(d^A \cdot (x_0^A \oplus x_1^A),\, d^B \cdot (x_0^B \oplus x_1^B))}\right\rangle . \tag{4.5.10}$$

As noted earlier, Alice and Bob can determine precisely which of the four Bell states have been prepared in a given `Bell` round. But, by the adaptive hardcore bit property of the ENTCF family (property (iv)), the device cannot efficiently compute α, β and hence cannot determine precisely what Bell state it has prepared.

4.5.2 Modified single-device self-testing protocol

It is important to note that Protocol 2 relies upon the verifiers, Alice and Bob, both behaving honestly. Specifically, after Protocol 2 has been executed, one of the two

parties must publish their stored data so that the other can use it, along with their own stored data, to determine whether the device is behaving honestly or not. If, for instance, Bob is the one publishing his data for Alice to test the device, and he is dishonest, he could publish data that is different from what he received from the device, thereby giving Alice a false impression of the device's behaviour.

This reliance on the honesty of Alice and Bob is a point of concern in the setting of OT. Given this, we now present a variation of Protocol 2 where Alice is the sole verifier; Bob is still present in the protocol except that he is now being modelled as part of the device. The checks that must be performed are the same as the checks for Protocol 2, except that they are all done by Alice now. The behaviour of an honest device in Protocol 3 is the same as the honest behaviour in Protocol 2.

Protocol 3: Single-device self-testing with a single verifier

1. Alice chooses the state bases $\theta^A, \theta^B \in \{\texttt{Computational, Hadamard}\}$ uniformly at random and generates key-trapdoor pairs $(k^A, t^A), (k^B, t^B)$, where the generation procedure for k^A and t^A depends on θ^A and a security parameter η, and likewise for k^B and t^B. Alice supplies Bob with k^B. Alice and Bob, respectively, then send the keys k^A, k^B to the device.

2. Alice and Bob receive strings c^A and c^B, respectively, from the device.

3. Alice chooses a *challenge type* $CT \in \{\texttt{a,b}\}$ uniformly at random and sends it to Bob. Alice and Bob then send CT to each component of their device.

4. If $CT = \texttt{a}$:

   i) Alice and Bob receive strings z^A and z^B, respectively, from the device.

5. If $CT = \texttt{b}$:

   i) Alice and Bob receive strings d^A and d^B, respectively, from the device.

   ii) Alice chooses uniformly random measurement bases $x, y \in \{\texttt{Computational, Hadamard}\}$ and sends y to Bob. Alice and Bob then, respectively, send x and y to the device.

   iii) Alice and Bob receive answer bits a and b, respectively, from the device. Alice and Bob also receive bits h^A and h^B, respectively, from the device.

The only role Bob has in Protocol 3 is to act as a relay between Alice and the component of the device held by Bob. It is clear that since Bob is not supplied with the state bases θ^A, θ^B or the trapdoors t^A, t^B, any malicious behaviour from Bob can be folded into the device, and so without loss of generality, we can assume that Bob acts honestly in Protocol 3. Then in order for Alice's checks to pass, the device must act honestly in Protocol 3 for the same reason that it must act honestly in Protocol 2. Thus, Theorem 4.5.1 applies to Protocol 3 as well.

It should also be noted that to call Theorem 4.5.1, we must know the probability $1-\delta$ with which the device wins Protocol 3. Similar to the IID case in [KW16], the IID assumption enables Alice to estimate δ ahead of time; in our case, this can be done by having Bob temporarily give Alice his component of the device (so that Bob cannot influence the sample that Alice uses to estimate δ).

Remark 4.5.2. A natural concern is if Bob's component has a "secret backdoor" which can be used to change its behaviour for this estimation phase, thus allowing Bob to influence the sample used by Alice despite the fact that she holds his component. However, this type of behaviour would violate the IID assumption, and thus we can safely rule it out.

Now suppose Alice uses N rounds to estimate δ. Let F_i be a binary random variable for whether or not the device fails the i-th round. Then $F_1, \ldots, F_N$ are independent random variables, each of which is equal to 1 with probability δ. If $F = F_1 + \cdots + F_N$, then the expected value is $\mathbb{E}[F] = N\delta$. Alice's estimate of δ is then $\delta' \coloneqq F/N$. If she wants her estimate to be within τ of δ, then from the Chernoff bound Lemma 2.3.2, we have that,

$$\begin{aligned} P(|\delta' - \delta| \geq \tau) &= P(|F - N\delta| \geq N\tau), \\ &= P(|F - N\delta| \geq (\tau/\delta)N\delta), \\ &\leq 2e^{-\frac{(\tau/\delta)^2 N\delta}{3}}, \\ &= 2e^{-\frac{\tau^2 N}{3\delta}}, \\ &\leq 2e^{-\frac{\tau^2 N}{3}}. \end{aligned}$$

Thus, $1 - 2e^{-\frac{\tau^2 N}{3}} \leq P(\delta' \in (\delta - \tau, \delta + \tau))$. That is, Alice can improve her estimate δ' by choosing τ to be small and taking a large sample N.

4.6 Device-independent oblivious transfer

The goal of this section is to use the single-device self-testing protocol (Protocol 3) to make the protocol for Rand 1-2 OT^ℓ (Protocol 1) device-independent. The result of

this is Protocol 4. It should be noted that Protocol 4 only considers the case where the sender is the verifier. Although it seems natural to require another protocol to allow the receiver to be the verifier, we find that such a protocol is entirely unnecessary due to the fact that we already have perfect-receiver-security for Protocol 4 (see Proposition 4.6.2).

Let us now describe Protocol 4. The first five steps of Protocol 4 can be summarized as executing n rounds of Protocol 3 (with the sender playing the role of Alice and the receiver playing the role of Bob), processing the data, and then checking if the device has behaved honestly for a subset of the rounds. In fact, these first five steps look very similar to the first six steps of the DIQKD protocol in [MDCA21]. However, there are some key differences which we now discuss.

Firstly, the DIQKD protocol uses Protocol 2 for single-device self-testing while we use Protocol 3. As noted earlier, the setting of OT is such that the sender and receiver do not necessarily trust each other. It is natural, then, that in making a device-independent version of Protocol 1, we should not utilize Protocol 2 to verify the device, as this protocol relies on two cooperating parties. Instead, we should use Protocol 3, which only requires one verifier.

Furthermore, it is assumed that the probability with which the device wins Protocol 3, $1-\delta$, has been estimated by the sender prior to Protocol 4, as discussed at the end of Section 4.5.2. The result of this is that with probability at least $1-2e^{-\frac{\tau^2 N}{3}}$, we have $\delta-\tau<\delta'<\delta+\tau$ and thus $\delta'-\tau<\delta$. We then use $\delta'-\tau$ as the threshold to check against in step 5.

Secondly, we require that, with some probability, the receiver ignore the measurement basis question supplied by the sender and instead ask the device to measure in the basis specified by the choice bit. The purpose of this modification is to remedy the following problem. Since the sender is testing the device, they are supplying the receiver with all inputs for their component of the device. This gives the sender precise knowledge of what measurement basis the receiver is using for every round. Receiver-security is then compromised when the receiver tells the sender the indices of all rounds where they have retained, amongst the useable rounds, those where the measurement basis coincided with their choice bit (the sender can immediately learn the choice bit from this). But, with our modification, we end up with a set of rounds I where the sender is ignorant to the receiver's measurement basis questions, and thus ignorant to the choice bit. The sender remains ignorant to the choice bit even after the step where the sender tests the device's honesty; this is because when the receiver is required to send their stored data to the sender, for the sake of testing, the receiver excludes data for rounds from I.

We now return to our description of Protocol 4. In step 6, the sender identifies a subset $\widetilde{I}\subseteq I$, which is the set of all rounds in I where the device has prepared a Bell pair and measured the sender's and receiver's half of the pair in the basis specified by each of them. Then for each round in $\widetilde{I}$, the sender publishes the trapdoor that

corresponds to the receiver's key (the trapdoors are needed for the next step). Then, in step 7, the sender and receiver correct their output in accordance with relation (3.3.4). At this point, we will have completed the first three steps of Protocol 1 in a device-independent manner.

The last two steps of Protocol 4, step 8 and step 9, are then identical to the last two steps of Protocol 1, with slightly different notation.

Protocol 4: DI Rand 1-2 OT^{ℓ}

Data generation:

1. The sender and receiver execute n rounds of Protocol 3 with the sender as Alice and the receiver as Bob, and with the following modification:
   If $CT_i = \texttt{b}$, the receiver makes a uniformly random choice on whether to use the measurement basis question supplied by the sender or

   $$y_i = [\texttt{Computational, Hadamard}]_c$$

   where c is the receiver's choice bit.[4] Let I be the set of indices marking the rounds where this has been done.
   For each round $i \in \{1, \ldots, n\}$, the receiver stores:

   - c_i^B
   - z_i^B if $CT_i = \texttt{a}$
   - or (d_i^B, y_i, b_i, h_i^B) if $CT_i = \texttt{b}$

   The sender stores $\theta_i^A, \theta_i^B, (k_i^A, t_i^A), (k_i^B, t_i^B), c_i^A, CT_i$; and z_i^A if $CT_i = \texttt{a}$, or (d_i^A, x_i, a_i, h_i^A) and y_i if $CT_i = \texttt{b}$.

2. For every $i \in \{1, \ldots, n\}$, the sender stores the variable RT_i (round type), defined as follows:

   - if $CT_i = \texttt{b}$ and $\theta_i^A = \theta_i^B = \texttt{Hadamard}$, then $RT_i = \texttt{Bell}$
   - else, set $RT_i = \texttt{Product}$

3. For every $i \in \{1, \ldots, n\}$, the sender chooses T_i, indicating a test round or generation round, as follows:

   - if $RT_i = \texttt{Bell}$, choose $T_i \in \{\texttt{Test, Generate}\}$ uniformly at random
   - else, set $T_i = \texttt{Test}$.

[4] We thank Daochen Wang, Honghao Fu, and Qi Zhao for pointing out a correction related to the probability with which the receiver makes this measurement choice.

The sender sends $(T_1, \ldots, T_n)$ to the receiver.

Testing:

4. The receiver sends the set of indices I to the sender. The receiver publishes their output for all $T_i = \texttt{Test}$ rounds where $i \notin I$. Using this published data, the sender sets a variable W_i to `pass` if the checks for Protocol 3 are passed; otherwise, $W_i = \texttt{fail}$.

5. The sender computes the fraction of test rounds (for which the receiver has published data for) for which $W_i = \texttt{fail}$. If this exceeds the threshold $\delta' - \tau$ estimated by the sender prior to the protocol, then the protocol aborts.

Preparing data:

6. Let $\widetilde{I} \coloneqq \{i : i \in I \text{ and } T_i = \texttt{Generate}\}$ and $n' = |\widetilde{I}|$. The sender publishes $\widetilde{I}$ and, for each $i \in \widetilde{I}$, the trapdoor t_i^B that corresponds to the key k_i^B that was given by the sender in the execution of Protocol 3, step 1.

7. For each $i \in \widetilde{I}$, the sender calculates α_i and defines w_i^α by

$$w_i^\alpha = \begin{cases} \alpha_i, & \text{if } x_i = \texttt{Hadamard} \\ 0, & \text{if } x_i = \texttt{Computational} \end{cases}$$

and the receiver calculates β_i and defines w_i^β by

$$w_i^\beta = \begin{cases} 0, & \text{if } y_i = \texttt{Hadamard} \\ \beta_i, & \text{if } y_i = \texttt{Computational} \end{cases}$$

Obtaining output:

8. The sender picks two uniformly random hash functions $f_0, f_1 \in \mathcal{F}$, announces f_0, f_1 and x_i for each $i \in \widetilde{I}$, and outputs $s_0 = f_0(a \oplus w^\alpha|_{\widetilde{I}_0})$ and $s_1 = f_1(a \oplus w^\alpha|_{\widetilde{I}_1})$, where $\widetilde{I}_r \coloneqq \{i \in \widetilde{I} : x_i = [\texttt{Computational, Hadamard}]_r\}$.

9. Receiver outputs $s_c = f_c(b \oplus w^\beta|_{\widetilde{I}_c})$.

Note that if step 5 is passed, then the fraction of failed test rounds does not exceed $\delta' - \tau$. Additionally, with probability at least $1 - 2e^{-\frac{\tau^2 N}{3}}$, we have that the sender's estimate $1 - \delta'$ of the device's winning probability $1 - \delta$ satisfies $\delta' - \tau < \delta$. With the occurrence of these two events, the sender can use Theorem 4.5.1 and the IID assumption to say that for each `Generate` round, one of the four Bell states has been prepared. This results in our statement of sender-security being conditioned on the high probability event that $\delta' - \tau < \delta$.

Remark 4.6.1. Notably, the use of the IID assumption here applies to both the device *and* the receiver. This stems from the fact that, in the single-verifier self-testing protocol, Protocol 3, any malicious behaviour from Bob was folded into the device. Thus, when the receiver holds a component of the device and we use Protocol 3, it is necessary to extend the IID assumption to the receiver for the following security analysis to hold.

The rest of the protocol, which operates only on the rounds $i \in \widetilde{I}$, is then almost identical to Protocol 1. The main difference here is that, for the relevant rounds, the sender is supplying the receiver with the trapdoor t^B in step 6 to allow the receiver to compute w^β. Intuitively, this action does not give a dishonest receiver any advantage at this point, as the device has already been verified and the key-trapdoor pair only allows the receiver to learn one of the two bits that, collectively, indicates which of the four Bell states has been used in a given round.

Additionally, a dishonest sender also has no advantage in this device-independent setting when compared to Protocol 1. Although the receiver must interact with the sender in Protocol 4, while there was previously no need to in Protocol 1, this interaction does not give the sender any information on the receiver's choice bit. To see this, observe that this interaction occurs in step 4 where the receiver publishes the set of indices I and all outputs for `Test` rounds so long as $i \notin I$. Thus, the output that the sender gets for `Test` rounds says nothing about the receiver's actions since the input for these rounds was completely specified by the sender. It is only in the rounds where $i \in I$ that the receiver measures according to their choice bit, and for these rounds, only the set of indices I is published, which says nothing of the actual choice bit.

It should also be noted why assumption 4 is necessary. The use of Protocol 3 means that we can certify that the device has prepared a quantum state and measured it according to the prescribed measurement bases, while allowing arbitrary communication. Arbitrary communication poses a problem, though, to everlasting security in this setting where the sender and receiver do not trust each other. For instance, if the component held by the receiver leaked their measurement basis questions y, then the sender can immediately learn the receiver's choice bit by looking at what y was in the rounds $i \in I$. Conversely, if the component held by the sender leaked their inputs and outputs, then a dishonest receiver could execute Protocol 4 honestly and still compromise sender-security. Indeed, suppose that, in executing the protocol honestly, the receiver obtained the string s_0 but stored the leaked inputs and outputs from the sender's component. To then learn $s_1 = f_1(a \oplus w^\alpha|_{\widetilde{I}_1})$ after the protocol is over, the receiver must learn the sender's measurement outcomes a and the bits α for the $\widetilde{I}_1$ rounds. The measurement outcomes a would be amongst the leaked data, and so the task reduces to determining α from the following leaked data:

- the key k^A which indexes the function pair $(f_{k^A,0}, f_{k^A,1})$
- the string c^A which satisfies $f_{k^A,0}(x_0^A) = f_{k^A,1}(x_1^A) = c^A$
- the string d^A which satisfies $\alpha = d^A \cdot (x_0^A \oplus x_1^A)$

If the receiver can find the claw (x_0^A, x_1^A), then they can compute α. Finding the claw (x_0^A, x_1^A) is quantum-computationally hard without access to the trapdoor t^A (which was never given to the device), but with enough time and computational power, this could be done, and thus everlasting security for the sender is compromised. Consequently, assumption 4 is necessary in this context for everlasting security.

The proof of Proposition 4.4.2 largely carries over to the proof of perfect receiver-security for Protocol 4. As for sender-security for Protocol 4, the proof is similar to the proof of Proposition 4.4.4, though we will now have to use Theorem 4.5.1 and analyze the probability of not aborting. Note that the case where the sender, receiver, and the device behave honestly is analyzed in the proof of sender-security (see Case 1 of Proposition 4.6.3).

Proposition 4.6.2. *Protocol 4 is perfectly receiver-secure.*

Proof: The ccq-state $\rho_{CY\widetilde{S}}^{Z=1}$ is defined by the experiment where $\widetilde{S}$ interacts with an honest memory-bounded R and the protocol does not abort. Now, in a new Hilbert space, we define the ccccq-state $\hat{\rho}_{\hat{C}\hat{Y}\hat{S}_0'\hat{S}_1'\widetilde{S}}^{Z=1}$ according to a different experiment.

In this different experiment, we let $\widetilde{S}$ interact with a receiver that has unbounded quantum memory. Suppose the receiver has not actually input any measurement questions y_i into their component of the device for rounds where $i \in I$. Let A be the string the sender gets after inputting x_i for $i \in \widetilde{I}$, and let W^α be the string where the i-th entry is defined as

$$W_i^\alpha = \begin{cases} \alpha_i, & \text{if } x_i = \texttt{Hadamard}, \\ 0, & \text{if } x_i = \texttt{Computational}. \end{cases}$$

Now, the receiver waits for step 8 to receive x for the $i \in \widetilde{I}$ rounds. Let B be the string the receiver gets after inputting x for the $i \in \widetilde{I}$ rounds. By assumption 4, the sender is oblivious to the measurement basis questions the receiver has given to their component of the device, along with the answer bits returned to the receiver. Note that at this point the receiver will also have, from step 6, t_i^B for each $i \in \widetilde{I}$. The receiver uses t_i^B to calculate β_i for each $i \in \widetilde{I}$ and defines W^β to be the string where the i-th entry is

$$W_i^\beta = \begin{cases} 0, & \text{if } x_i = \texttt{Hadamard}, \\ \beta_i, & \text{if } x_i = \texttt{Computational}. \end{cases}$$

Define $\hat{S}'_0, \hat{S}'_1$ as

$$\hat{S}'_0 = f_0(A \oplus W^\alpha|_{\widetilde{I}_0}) \quad \text{and} \quad \hat{S}'_1 = f_1(A \oplus W^\alpha|_{\widetilde{I}_1}), \tag{4.6.1}$$

and note that $A \oplus W^\alpha = B \oplus W^\beta$. Sample $\hat{C}$ according to P_C and set $\hat{Y} = \hat{S}'_{\hat{C}}$. It follows by construction that

$$\Pr[\hat{Y} \neq \hat{S}'_{\hat{C}}] = 0, \tag{4.6.2}$$

and that $\hat{\rho}^{Z=1}_{\hat{C}}$ is independent of $\hat{\rho}^{Z=1}_{\hat{S}'_0\hat{S}'_1\widetilde{S}}$.

It now remains to argue that,

$$\hat{\rho}^{Z=1}_{\hat{C}\hat{Y}\widetilde{S}} = \rho^{Z=1}_{CY\widetilde{S}}$$

so that the corresponding S'_0 and S'_1 also exist in the original experiment. But, this is satisfied since the only difference between the two experiments is when and what x_i the receiver inputs for $i \in \widetilde{I}_{1-C}$ rounds, which does not affect $\rho^{Z=1}_{CY\widetilde{S}}$ respectively $\hat{\rho}^{Z=1}_{\hat{C}\hat{Y}\widetilde{S}}$. ∎

For the following proposition, recall Definition 4.4.3 which defines $\mathfrak{R}_\gamma$ as the set of all possible quantum dishonest receivers in Protocol 4 which have quantum memory of size at most $\gamma n'$ when step 8 of Protocol 4 is reached.

Also note that if we were not dealing with finite statistics, then the sender's initial estimate $1-\delta'$ of the device's winning probability $1-\delta$ could be done with arbitrary precision and so the following proposition would no longer be conditional on $\delta' - \tau < \delta$.

Proposition 4.6.3. *Protocol 4 is $(\varepsilon + (\delta'-\tau)^r)$-sender-secure against $\mathfrak{R}_\gamma$ for a negligible (in n' and η) $\varepsilon + (\delta'-\tau)^r$ if $\delta' - \tau < \delta$ (which occurs with probability at least $1 - 2e^{-\frac{\tau^2 N}{3}}$) and if there exists a $k > 0$ such that $kn' \leq n'/4 - 2\ell - \gamma n'$, where $n' = |\widetilde{I}|$, η is the security parameter used in Protocol 3, $(\delta' - \tau)$ and N is arising from the sender's estimate of δ, and r is from the application of Theorem 4.5.1.*

Proof:

We consider the different cases regarding the probability of not aborting. Let δ_F denote the fraction of failed test rounds in Protocol 4. If δ_F exceeds the threshold $\delta' - \tau$, Protocol 4 aborts. Given this, let Z be the binary random variable which describes whether Protocol 4 aborts or not. That is,

$$Z := \begin{cases} 1, & \text{if } \delta_F \leq \delta' - \tau, \\ 0, & \text{if } \delta_F > \delta' - \tau. \end{cases} \tag{4.6.3}$$

Case 1, honest behaviour: When both parties and the device behave honestly, the fraction of failed test rounds δ_F is small; by this, we mean that

$$\Pr(\delta_F \leq \delta' - \tau) \geq 1 - (\varepsilon + (\delta' - \tau)^r). \tag{4.6.4}$$

Since $\mathbb{E}[Z] = \Pr(\delta_F \leq \delta' - \tau)$, we then have $1 - \mathbb{E}[Z] \leq (\varepsilon + (\delta' - \tau)^r)$. We can now use the Chernoff bound in Lemma 2.3.1 to show that the probability of aborting is small,

$$\begin{aligned}
\Pr(Z = 0) &= \Pr(-Z \geq 0), \\
&\leq \mathbb{E}[e^{-Zt}], \\
&= e^{-t} \Pr(Z = 1) + (1 - \Pr(Z = 1)), \\
&= e^{-t} \Pr(\delta_F \leq \delta' - \tau) + (1 - \Pr(\delta_F \leq \delta' - \tau)), \\
&= e^{-t}\mathbb{E}[Z] + (1 - \mathbb{E}[Z]), \\
&\leq e^{-t}\mathbb{E}[Z] + (\varepsilon + (\delta' - \tau)^r), \quad \forall t \geq 0.
\end{aligned} \tag{4.6.5}$$

Then taking the limit $t \to \infty$ in equation (4.6.5), we see that the probability of aborting is small, $\Pr(Z = 0) \leq (\varepsilon + (\delta' - \tau)^r)$, which satisfies the completeness condition of Definition 4.4.1.

Case 2, dishonest behaviour and large δ_F: We now show that when the behaviour is dishonest and the fraction of failed test rounds δ_F is large, in the sense that

$$\Pr(\delta_F \leq \delta' - \tau) \leq (\varepsilon + (\delta' - \tau)^r), \tag{4.6.6}$$

the probability of not aborting is small. For this, we again use the Chernoff bound in Lemma 2.3.1,

$$\begin{aligned}
\Pr(Z = 1) &= \Pr(Z \geq 1), \\
&\leq \mathbb{E}[e^{tZ}]e^{-t}, \\
&= \Pr(Z = 1) + e^{-t} \Pr(Z = 0), \\
&= \mathbb{E}[Z] + e^{-t}(1 - \mathbb{E}[Z]), \\
&\leq (\varepsilon + (\delta' - \tau)^r) + e^{-t}(1 - \mathbb{E}[Z]), \quad \forall t \geq 0.
\end{aligned} \tag{4.6.7}$$

Then taking the limit $t \to \infty$ in equation (4.6.7), the probability of not aborting is small,

$$\Pr(Z = 1) \leq (\varepsilon + (\delta' - \tau)^r).$$

Thus, the overall expression in equation (4.4.2) is satisfied.

Case 3, dishonest behaviour and small δ_F: In this case, we bound the overall expression in equation (4.4.2). To do this, we first consider a single round $i \in \widetilde{I}$. In the context of this single round, we let α, β denote, respectively, the bits computed by the sender and the receiver in step 7, and letting X, Y and A, B be the classical random variables describing the sender's and receiver's questions and answers, respectively. Let $\sigma^{(\alpha,\beta)}$ be the joint state of the device in step 1 right before the device performs the measurements $P_{x,y}^{(a,b)}$. Then the state after step 1 is

$$\sum_{x,y,a,b} P_X(x)P_Y(y)\,\mathrm{tr}\big[P_{x,y}^{(a,b)}\sigma^{(\alpha,\beta)}P_{x,y}^{(a,b)}\big] \otimes |x,y,a,b\rangle\langle x,y,a,b|_{XYAB}\,.$$

Now we consider the event $\delta' - \tau < \delta$, which, from the end of Section 4.5.2, occurs with probability at least $1 - 2e^{-\frac{\tau^2 N}{3}}$. Observe that if the protocol did not abort at step 5, then $\delta_F \leq \delta' - \tau$, and hence, $1 - \delta < 1 - (\delta' - \tau) \leq 1 - \delta_F$, meaning that the winning condition of Protocol 3 is satisfied with probability at least $1 - (\delta' - \tau)$ in the `Test` rounds; but after step 1, it has not yet been decided whether a round will be a `Test` round or a `Generate` round, and so, we can use the IID assumption (see Remark 4.6.1) to apply Theorem 4.5.1 to `Generate` rounds. Using Theorem 4.5.1 in our $i \in \widetilde{I}$ round, the continuous and cyclical properties of the trace, and that $V^\dagger V = \mathbb{1}$, we find that the state for this round $\rho_{XYAB}^{Z=1}$ must be within trace distance $\mathcal{O}((\delta' - \tau)^r) + \mathsf{negl}(\eta)$ of the ideal state,

$$\xi_{XYAB} = \sum_{\alpha,\beta,x,y,a,b} P_X(x)P_Y(y)\,\langle\phi^{(\alpha,\beta)}|\,M_x^a \otimes M_y^b\,|\phi^{(\alpha,\beta)}\rangle \otimes |x,y,a,b\rangle\langle x,y,a,b|_{XYAB}\,. \tag{4.6.8}$$

This confirms that in each round $i \in \widetilde{I}$, the device has prepared a Bell state and measured it according to the measurement basis choices of the sender and the receiver.

Now, let us consider the state $\rho_{XAOE}^{Z=1}$ in the scenario where the protocol has not aborted and $\widetilde{R}$ has measured all but $\gamma n'$ of their qubits from the $\widetilde{I}$ rounds, where $n' = |\widetilde{I}|$. For the rest of the proof, we drop the $Z = 1$ superscript on the state for the sake of readability. Now, since we are making the IID assumption, this state is an n'-fold tensor product and the following are n'-tuples with each entry representing one of the $i \in \widetilde{I}$ rounds:

- X is the classical random variable describing the random choice of bases of the sender.
- A is the classical random variable describing the sender's results after measuring their part of the state in the random bases X.
- O is the classical random variable which contains the random choices of the key-trapdoor pairs (k^B, t^B) and the measurement basis questions y supplied by the sender. It will also contain any other information that the sender's component

of the device may have leaked to the receiver; this may include k^A, c^A, d^A, but by assumption 4, it cannot include the sender's measurement basis questions x or their answer bits a.

- E is the random state that describes $\widetilde{R}$'s part of the state.

The state ρ_{XAOE} then satisfies

$$\Delta(\rho_{XAOE}, \xi_{XAOE}) \leq n' \Big(\mathcal{O}((\delta' - \tau)^r) + \mathsf{negl}(\eta) \Big) = \mathcal{O}((\delta' - \tau)^r) + \mathsf{negl}(\eta), \quad (4.6.9)$$

where ξ_{XAOE} is the ideal state.

Analyzing the ideal state, the proof is now similar to the proof of Proposition 4.4.4. We start by lower bounding the smooth min-entropy for the state that is not conditioned on the event $\{Z = 1\}$ (*i.e.*, the event where the protocol does not abort).

At Equation (4.4.5), we choose $\lambda, \lambda', \kappa$ such that

$$(1/4 - \lambda - 2\lambda' - \kappa)n' - 1 - 2\ell \geq \gamma n'. \quad (4.6.10)$$

Then at Equation (4.4.17), we have

$$H_{\min}^{\varepsilon+\varepsilon'}(K_{1-C'}|X, C') \geq (1/4 - \lambda - \lambda')n' - 1,$$

where ε is exponentially small in n' and comes from Lemma 3.3.26, and $\varepsilon' = 2^{-\lambda' n'}$.

Then at Equation (4.4.19), in addition to conditioning on $F_{C'}$, we also condition on the random variable O because it too is independent. It is easy to see this for the measurement questions y supplied by the sender since they are generated uniformly randomly. Regarding the key-trapdoor pairs, observe that knowledge of (k^B, t^B) makes the adaptive hardcore bit property (iv) and the injective invariance property (i) of the ETCF family computationally easy. That is,

- With (k^B, t^B), the receiver can overcome the injective invariance property of the ETCF family (property (i)) and determine what θ^B is. However, the sender only publishes (k^B, t^B) for `Bell` rounds, as these are the only types of rounds that are useable for accomplishing Rand 1-2 OT^ℓ, and so the ability to overcome the injective invariance property and learn θ^B is redundant at this point.

- With (k^B, t^B), property (iv) of the ETCF family becomes computationally easy. That is, the unique claw (x_0^B, x_1^B) can be calculated with the function pair indexed by the key k^B, the trapdoor t^B, and the string c^B. This means the receiver can determine the precise form of the state $|\psi_2^B\rangle$ (this is the analogous state of $|\psi_2^A\rangle$ in Equation (4.5.5)), but $|\psi_2^B\rangle$ is independent of $|\psi_2^A\rangle$ until the entangling operation. After the entangling operation, knowledge of $|\psi_2^B\rangle$ is equivalent to learning β, but this is necessary and accounted for with the random variable K in the proof of Proposition 4.4.4.

As for information leaked from the sender's component of the device, knowledge of k^A, c^A, d^A makes it possible for the receiver to compute the bit α (see the discussion immediately before Proposition 4.6.2), though they cannot do this efficiently since they do not have access to the sender's trapdoor t^A. Given that the receiver will also know β, the receiver could eventually learn precisely what Bell state was used in each of the $\widetilde{I}$ rounds. To then learn the other string s_{1-c}, the receiver needs $a|_{\widetilde{I}_{1-c}}$ which appears uniformly random to the receiver since the sender and receiver chose different measurement bases for the $\widetilde{I}_{1-c}$ rounds. Thus, the receiver can do no better than guess, correctly, the other string s_{1-c} with probability $1/2^\ell$.

So, conditioning on O as well, and using the chain rule (Lemma 3.3.25) as in Equation (4.4.19), we obtain

$$H_{\min}^{\varepsilon+2\varepsilon'}(K_{1-C'}|XF_{C'}(K_{C'})F_{C'}OC') \geq \gamma n' + \ell + \kappa n'. \tag{4.6.11}$$

We now consider the smooth min-entropy of the state conditioned on not aborting (recall that $\{Z=1\}$ denotes the event of not aborting), which is lower bounded by the unconditioned smooth min-entropy (see Lemma 10 of [TL17]). Conditioning the min-entropy on this event decreases the min-entropy by at most $\log(1/\Pr(Z=1))$. So,

$$\begin{aligned} &H_{\min}^{\varepsilon+2\varepsilon'}(K_{1-C'}|XF_{C'}(K_{C'})F_{C'}OC')_{\{Z=1\}} \\ &\qquad \geq H_{\min}^{\varepsilon+2\varepsilon'}(K_{1-C'}|XF_{C'}(K_{C'})F_{C'}OC') - 2\log\left(\frac{1}{\Pr(Z=1)}\right), \\ &\qquad \geq \gamma n' + \ell + \kappa n' - 2\log\left(\frac{1}{\Pr(Z=1)}\right). \end{aligned}$$

Additionally, let $\varepsilon'' := (\varepsilon + 2\varepsilon')/\Pr(Z=1)$. Then,

$$\begin{aligned} H_{\min}^{\varepsilon''}(K_{1-C'}|XF_{C'}(K_{C'})F_{C'}OC')_{\{Z=1\}} &\geq H_{\min}^{\varepsilon+2\varepsilon'}(K_{1-C'}|XF_{C'}(K_{C'})F_{C'}OC')_{\{Z=1\}}, \\ &\geq \gamma n' + \ell + \kappa n' - 2\log\left(\frac{1}{\Pr(Z=1)}\right). \end{aligned}$$

Letting $S_0 = F_0(K_0)$ and $S_1 = F_1(K_1)$, and setting $U = XS_{C'}F_{C'}OC'$, we get, after applying Theorem 3.3.28,

$$\begin{aligned} &\Delta(\xi_{S_{1-C'}F_{1-C'}XS_{C'}F_{C'}OC'E}, \tfrac{1}{2^\ell}\mathbb{1}\otimes\xi_{F_{1-C'}XS_{C'}F_{C'}OC'E}) \\ &\qquad \leq \frac{1}{2}2^{-\frac{1}{2}(H_{\min}^{\varepsilon''}(K_{1-C'}|XS_{C'}F_{C'}OC')_{\{Z=1\}}-\gamma n'-\ell)} + 2\varepsilon'', \\ &\qquad \leq \left(\frac{1}{2}2^{-\frac{1}{2}\kappa n'} + 2\varepsilon + 4\varepsilon'\right)\frac{1}{\Pr(Z=1)}. \end{aligned}$$

Now, using the triangle inequality twice,

$$
\begin{aligned}
&\Delta(\rho_{S_{1-C'}F_{1-C'}UE},\, \tfrac{1}{2^\ell}\mathbb{1}\otimes\rho_{F_{1-C'}UE}) \\
&\qquad\leq \Delta(\rho_{S_{1-C'}F_{1-C'}UE}, \xi_{S_{1-C'}F_{1-C'}UE}) + \Delta(\xi_{S_{1-C'}F_{1-C'}UE},\, \tfrac{1}{2^\ell}\mathbb{1}\otimes\rho_{F_{1-C'}UE}), \\
&\qquad\leq \mathcal{O}((\delta'-\tau)^r) + \mathsf{negl}(\eta) + \Delta(\xi_{S_{1-C'}F_{1-C'}UE},\, \tfrac{1}{2^\ell}\mathbb{1}\otimes\xi_{F_{1-C'}UE}) \\
&\qquad\qquad + \Delta(\tfrac{1}{2^\ell}\mathbb{1}\otimes\xi_{F_{1-C'}UE}, \tfrac{1}{2^\ell}\mathbb{1}\otimes\rho_{F_{1-C'}UE}), \\
&\qquad\leq 2(\mathcal{O}((\delta'-\tau)^r) + \mathsf{negl}(\eta)) + \left(\frac{1}{2}2^{-\frac{1}{2}\kappa n'} + 2\varepsilon + 4\varepsilon'\right)\frac{1}{\Pr(Z=1)}.
\end{aligned}
$$

Thus,

$$
\begin{aligned}
&\Pr(Z=1)\Delta(\rho_{S_{1-C'}F_{1-C'}UE},\, \tfrac{1}{2^\ell}\mathbb{1}\otimes\rho_{F_{1-C'}UE}) \\
&\qquad\leq 2(\mathcal{O}((\delta'-\tau)^r) + \mathsf{negl}(\eta)) + \frac{1}{2}2^{-\frac{1}{2}\kappa n'} + (2\varepsilon + 4\varepsilon').
\end{aligned}
$$

∎

4.7 Summary and future directions

We have developed and proved the security of a protocol for DIOT. To enable OT, we relied on the reasonable assumption of bounded-quantum-storage. To achieve device-independence, we relied on a post-quantum computational assumption; the benefit of this is that our protocol allows some communication between its components, whereas approaches that rely on a violation of a Bell-inequality requires strict non-communication (which can be practically difficult to enforce).

Can we prove security of our DIOT protocol in a more general setting? When we proved the security of our scheme, we made the IID assumption: that the device behaves independently and identically in each round. This assumption significantly simplifies the analysis, but it comes at the cost of weakening the meaning of device-independence; for true device-independence, we must prove security in the non-IID setting. Naturally, we then ask if one can prove the security of our DIOT protocol in the non-IID setting? The technique of reducing the analysis of the non-IID setting to the IID setting via the entropy accumulation theorem seems like a promising approach for achieving such a task. For more on this technique, the reader is referred to [ADF+18, Arn18, ARV19, DFR20].

Chapter 5

Noise-tolerance for public-key quantum money

In this chapter, we study quantum money in the public-key setting. We develop a scheme in the classical oracle model that tolerates noise and then analyze its security. This chapter is based on [Yue25].

5.1 Introduction

The history of quantum money is essentially as long as quantum information itself. At its core, quantum money aims to address the issue of verifying the validity of banknotes while ensuring that they cannot be counterfeited (*i.e.*, *cloned*). Famously, this problem was first addressed by Wiesner in [Wie83] where, put simply, a single banknote consisted of n conjugate coding states, and verification involved measuring each qubit in the basis that it was prepared in (thus, this verification procedure requires the verifier to know how a given banknote was prepared). Without knowledge of how the banknote was prepared, a counterfeiter attempting to produce a second copy of a valid banknote, without the ability to access the verification procedure during the counterfeiting process, has a success probability of at most $(3/4)^n$, as shown in [MVW13].

Crucially, in Wiesner's scheme, only banks are capable of verifying banknotes (since the ability to verify enables the minting of new, valid banknotes). Quantum money where *anyone* can perform verification of a banknote is known as *public-key quantum money*. Public-key quantum money was first addressed in [Aar09] where it was proved that there exists a quantum oracle relative to which secure public-key quantum money is possible. Then, in the foundational work of [AC12], it was proven that public-key quantum money relative to a classical oracle is secure. The scheme involves a banknote as a subspace state $|A\rangle$ (a uniform superposition of the vectors in a random $n/2$-dimensional subspace $A \subseteq \mathbb{F}_2^n$) and verification is performed through

classical oracles that check for membership in A and the dual subspace $A^{\perp}$. The technical result from [AC12] is that any quantum algorithm that maps $|A\rangle$ to $|A\rangle^{\otimes 2}$ must make exponentially many queries, implying that a quantum polynomial-time adversary with access to the verification procedure would not be able to counterfeit a banknote.

Noise-tolerant schemes. While much work has been devoted to public-key quantum money, that work has been focused on instantiating the oracles of [AC12] or devising entirely new schemes. Regardless of how public-key quantum money is obtained in the plain model (without oracles), it will be necessary for such a scheme to be noise-tolerant (*i.e.*, robust) for it to be practically meaningful. Indeed, one of the most pressing practical issues of quantum money is the necessity of quantum storage for the banknote (for a review of quantum storage, see [HEH+16]; for more recent work, see [WLQ+21] where the authors maintain coherence of a qubit for over one hour). Due to the inherently sensitive nature of quantum information, any amount of time in quantum storage almost guarantees the incursion of quantum errors. If noise affects an honest banknote to the point that it does not pass verification, then it has lost its prescribed value. In the classical oracle scheme of [AC12], a single error will, in general, cause an honest banknote to no longer pass verification (see Section 5.2.1). To the best of our knowledge, we are unaware of any previous work that focuses on noise-tolerant public-key quantum money.

In the private-key setting of quantum money, there has indeed been work towards noise-tolerance (see [PYJ+12, AA17, Kum19]). Notably, the scheme in [AA17] is based off *hidden matching* quantum retrieval games; in this model, the authors prove that, for $n = 14$, the maximum achievable noise-tolerance is 23.3%, which makes their protocol nearly optimal since it can tolerate up to 23.03% noise. In [BSS21], the authors construct a noise-tolerant private tokenized signature scheme (from which one can obtain private-key quantum money; see Section 5.1.3). To simply describe their approach, a token is a tensor product of conjugate coding states, signing the bit 0 (1) involves measuring all qubits in the computational (Hadamard) basis, and verification amounts to comparing the measured string with a bank's secret string to check for agreement; noise-tolerance comes from allowing inconsistencies in a fraction of the string. Unfortunately, their scheme is not secure against an adversary that can query the classical verification oracle in quantum superposition.

5.1.1 Contributions

We present a scheme for noise-tolerant public-key quantum money and analyze its completeness and soundness error. Intuitively, completeness error quantifies the correctness of the scheme while soundness error quantifies the success probability of a counterfeiter. More precisely, completeness error ε means that valid banknotes (those

minted by the bank and then possibly affected by tolerable noise) are accepted by the verification procedure with probability at least $1 - \varepsilon$. Soundness error δ means that a counterfeiter, who possesses a valid banknote and whose quantum circuit is polynomial in size, can produce another banknote such that both banknotes are accepted with probability at most δ. We show that our scheme has perfect completeness error and has $|\mathcal{E}_q|^2/\mathsf{exp}(n)$ soundness error, where $\mathcal{E}_q$ represents the set of tolerated Pauli X and Z errors, each affecting up to q of the n qubits in our banknote (*i.e.*, this corresponds to tolerating q arbitrary errors). Intuitively, perfect completeness error stems from the fact that noise-tolerance is built into our verification procedure (a freshly minted banknote affected by tolerable noise will always be accepted). This verification procedure uses classical oracles to check for membership in spaces that are determined by q and the chosen subspace. As a result, increasing q correlates to increasing the size of the space being checked; intuitively, this improves the success probability of a counterfeiter and is formally reflected in our $|\mathcal{E}_q|^2/\mathsf{exp}(n)$ soundness error. A freshly minted banknote in our scheme is a subspace state, as in [AC12], though we use a relation formalized in [CV22] to show that such a banknote can be prepared by generating conjugate coding states and applying a unitary that acts as a change-of-basis.

5.1.2 Techniques

Our approach to noise-tolerance starts with the fact that the banknote state used in [AC12] can be viewed as a codeword for a quantum error correcting code; specifically, a Calderbank-Shor-Steane (CSS) code. With this connection, we ask: is it possible to securely use the corresponding CSS code to detect and tolerate errors? Ignoring security, momentarily, the answer is yes. In the stabilizer formalism [Got97], error detection involves measuring a set of *stabilizer generators*; the resulting classical string, known as a *syndrome*, can be processed to tell us what errors occurred and on which qubits. This information can be used to correct the noise or, more simply, to accept a state that has been afflicted by correctable noise. When it comes to security, we find that direct application of this error detection procedure, without "hiding" its properties, can compromise security, as the generators reveal information about the underlying subspace (see Section 5.3.1).

The next and perhaps most obvious approach is to apply a layer of error correction on top of the oracle scheme of [AC12]. With this approach, one would encode the subspace state $|A\rangle$ and then proceed in one of two ways: perform verification in a fault-tolerant manner; or, error correct, decode, and then apply the normal verification procedure. In Section 5.3.2, we briefly discuss this approach and outline some of its obstacles, though we do not rule it out as a possibility and instead leave it as one of the open questions listed in Section 5.7.

In this work, the direction we take is to design a scheme that is inherently

noise-tolerant (although we focus on tolerating noise, we discuss in Remark 5.6.11 how our scheme could be easily adapted to also correct the errors). We present two approaches for a noise-tolerant scheme, where one approach can be viewed as a generalization of the other. While the two approaches achieve the same completeness and soundness error, we include both of them here since only one approach seems to have an obvious realization in the plain model with existing techniques, though it is the less general approach (see Section 5.7 where we discuss this as a direction for future research). Notably, both approaches can be seen as natural extensions of the original classical oracle scheme of [AC12]. The difference between the two approaches, described simply, is that one approach checks for membership in a union of subspaces while the other approach checks membership in a union of subsets. Note that in both approaches, we tolerate up to q arbitrary errors by tolerating up to q Pauli-X errors and up to q Pauli-Z errors, as is done in CSS codes.

Conceptually, the subspace approach closely resembles error detection in the stabilizer formalism. It starts by recognizing that each tolerable syndrome (*i.e.*, tolerable error) corresponds to a subspace. We can then form two unions of disjoint subspaces (one for Pauli-X errors and one for Pauli-Z errors). We then perform verification by using classical oracles to check if the banknote is a member of these two unions.

In the subset approach, we do essentially the same thing except that we check membership in subsets. We recognize that, in the computational basis, a Pauli-X error causes a subspace state to become a superposition of the vectors in a coset (*i.e.*, it becomes a coset state), where each Pauli-X error corresponds to a unique coset. Under this relation, the set of tolerated Pauli-X errors corresponds to a collection of cosets. To then accept a state corrupted by a Pauli-X error, we use a classical oracle to check membership in the subset defined as the union of all cosets from this collection. We tolerate Pauli-Z errors by checking membership in a similarly defined subset.

We focus primarily on the subset approach, as it is conceptually simpler and the security proof for the subspace approach is almost the same (throughout the security analysis, we remark on any changes that would be necessary for the subspace approach). As one might expect, for a banknote with a fixed size n, the more noise we tolerate the easier it will be for a counterfeiter to succeed. This is captured by the $|\mathcal{E}_q|^2/\mathsf{exp}(n)$ soundness error, where the quantity $|\mathcal{E}_q|$ (formally given in equation (3.4.9)) approaches exponential when q is close to n, and thus the soundness error approaches 1 in this case. On the other hand, if $q = kn$ and $0 < k \leq 1/2$, then from [FG06] we see that

$$|\mathcal{E}_q| \leq 2^{2H(k)n}, \tag{5.1.1}$$

where $H(\cdot) \in [0, 1]$ is the binary Shannon entropy. Thus, if we tolerate only a few errors, then $|\mathcal{E}_q|$ will be sufficiently small to get a "good" soundness error. This trade-off between security and noise-tolerance is visualized in Figure 5.2.

On top of the above construction, we make use of an observation (formalized in [CV22]) on the relation between conjugate coding states and coset states in the generation of our banknotes. Since conjugate coding states are practically simple to prepare, we use this relation with the intent of making banknote generation similarly simple. This relation (Lemma 3.3.1), allows the bank to generate the subspace state banknote by preparing conjugate coding states and then applying a unitary that acts as a change-of-basis. Intuitively, this does not affect the security of our scheme, as an adversary still only receives a subspace state from the bank.

5.1.3 Further background on quantum money

As noted earlier, Wiesner's scheme is secure in a model where the counterfeiter cannot access the verification procedure during the counterfeiting process. In a model where the verification procedure returns accept/reject *and* the post-measurement state, there is a simple attack on Wiesner's scheme where, by repeatedly submitting banknotes for verification, a counterfeiter can produce a second banknote [Lut10, Aar09]. More generally, the technique of quantum state restoration [FGH+10] can be used to break the scheme. In a slightly different model, where the post-measurement state is only returned if the verification procedure accepts, Wiesner's scheme remains insecure, as shown in [NSBU16].

There have been several attempts to obtain public-key quantum money in the plain model, though this has proven to be a highly non-trivial task. Most attempts at instantiating the oracles have subsequently been broken. In the quantum oracle model of [Aar09], Aaronson proposed an explicit scheme based on random stabilizer states, though this was later broken in [LAF+10]. When the authors of [AC12] introduced the classical oracle model, they also proposed an instantiation that was later broken in [PFP15, CDF+19, BDG23]. In [Zha19], Zhandry proposed a new primitive called quantum lightning from which one can obtain public-key quantum money; informally, quantum lightning is like public-key quantum money except that even banknote generation is public, though an adversary is still unable to produce two banknotes with the same serial number. Zhandry also gave a construction of quantum lightning in [Zha19], though, as noted in a later version of the paper [Zha21], a weakness in the underlying hardness assumption was detected in [Rob21]. A full attack on the scheme was eventually found in [BDG23]. Also in [Zha19], Zhandry instantiated the oracles in [AC12] by using indistinguishability obfuscation (iO) [BGI+12, GGH+13, SW14]. The existence of post-quantum iO remains unsolved, as candidate constructions proposed in [WW21, GP21] have been subject to attacks [HJL21], but results such as [BGMZ18] suggest that it could still exist.

Apart from the previously mentioned schemes that focus on instantiating the oracles, there is a category of proposed public-key quantum money schemes that rely on mostly untested hardness assumptions. A particularly well-known example

is a scheme from [FGH+12] that relies on knot theory; while the scheme remains unbroken, it is difficult to analyze and lacks a security proof. Also in this category are the schemes presented in [Kan18, KSS21], though some analysis on [KSS21] has been done in [BDG23].

In [LMZ23, AHY23], the authors push us towards a better understanding of suitable problems for instantiating public-key quantum money: in [LMZ23], the authors rule out a natural class of quantum money schemes that rely on lattice problems; and in [AHY23], the authors rule out a class of money schemes that are constructed from black box access to collision-resistant hash functions (where the verification procedure only makes classical queries to the hash function).

It is also worth mentioning the notion of quantum tokenized signature schemes. The relationship of this primitive to quantum money is characterized in [BS23] where the authors show that a testable public (private) tokenized signature scheme can give public (private) quantum money. The testable tokenized signature scheme constructed in [BS23] is based on the oracle scheme in [AC12]. In [CLLZ21], the authors construct a variant of the [BS23] scheme (coset states are used instead of subspace states) and show its security in the plain model by using post-quantum iO. The schemes in [BS23] and [CLLZ21] are not noise-tolerant, and we are unaware of any noise-tolerant public tokenized signature scheme.

5.1.4 Outline

We start in Section 5.2 with recalling how public-key quantum money was defined in [AC12]. Then, in Section 5.2.1, we extend this definition to the noisy setting. Before describing our approaches to designing a noise-tolerant scheme (in Section 5.4 and Section 5.5), we briefly outline alternative approaches in Section 5.3, though these are not the main focus of our work. In Section 5.6, we present the security analysis of our scheme. In Section 5.7, we list a few open questions for future research.

5.2 Definitions

In this section, we recall how public-key quantum money is defined in [AC12]. Specifically, we focus on what Aaronson and Christiano call *mini-schemes*, since they show that a secure mini-scheme can be combined with a digital signature scheme to construct a full public-key quantum money scheme that is also secure. It is worth noting that the construction uses a mini-scheme in a straightforward manner, so that if we handle noise at the level of the mini-scheme, we will retain the same level of noise-tolerance in the full scheme (from another perspective, the construction does not involve any transformations of the quantum money state that could result in further errors). For additional details on this construction, the reader is referred to [AC12].

Notably, a *valid banknote* in the context of Definition 5.2.1 refers to a banknote as output by the algorithm Bank. For the noise-tolerant setting, we define a valid banknote differently (see definition Definition 5.2.3).

Definition 5.2.1 (Mini-schemes [AC12])**.** *A (public-key)* ***mini-scheme*** $\mathcal{M}$ *consists of two polynomial-time quantum algorithms:*

- Bank*, which takes as input a security parameter* 0^n*, and probabilistically generates a banknote* $\$ = (s, \rho_s)$ *where* s *is a classical* ***serial number****, and* ρ_s *is a quantum money state.*
- Ver*, which takes as input an alleged banknote* $\tilde{\$}$*, and either accepts or rejects.*

We say $\mathcal{M}$ *has* ***completeness error*** ε *if* $\mathsf{Ver}(\$)$ *accepts with probability at least* $1 - \varepsilon$ *for all valid banknotes* $\$$*. If* $\varepsilon = 0$*, then* $\mathcal{M}$ *has perfect completeness. If, furthermore,* $\rho_s = |\psi_s\rangle\langle\psi_s|$ *is always a pure state, and* Ver *simply consists of a projective measurement onto the rank-1 subspace spanned by* $|\psi_s\rangle$*, then we say that* $\mathcal{M}$ *is* ***projective****.*

Let Ver_2 *(the* ***double verifier****) take as input a single serial number* s *as well as two (possibly entangled) states* σ_1 *and* σ_2*, and accept if and only if* $\mathsf{Ver}(s, \sigma_1)$ *and* $\mathsf{Ver}(s, \sigma_2)$ *both accept. We say that* $\mathcal{M}$ *has* ***soundness error*** δ *if, given any quantum circuit* $\mathscr{C}$ *of size* $\mathsf{poly}(n)$ *(the* ***counterfeiter****),* $\mathsf{Ver}_2(s, \mathscr{C}(\$))$ *accepts with probability at most* δ*, where* s *is the honest serial number generated by* Bank *during the minting of* $\$$*. Here, the probability is over the banknote* $\$$ *output by* $\mathsf{Bank}(0^n)$*, as well as the behaviour of* Ver_2 *and* $\mathscr{C}$*.*

We call $\mathcal{M}$ ***secure*** *if it has completeness error* $\leq 1/3$ *and negligible soundness error.*

It is worth noting that the purpose of the classical serial number is to index the quantum money state. This point will be made clearer in Section 5.4.

5.2.1 Defining a noise-tolerant mini-scheme

Put simply, the banknote in the money scheme of [AC12] is a subspace state $|A\rangle$ and verification is performed by projecting onto the subspace spanned by $|A\rangle$. If the state $|A\rangle$ is corrupted by a Pauli error, then this state will, in general, be orthogonal to $|A\rangle$ (see the end of the proof for Lemma 5.5.1) and hence rejected by this verification procedure. To extend the ideas of [AC12] to tolerate noise, we propose that verification of a banknote amounts to projecting onto the subspace spanned by the banknote and all its tolerated noisy variants (*i.e.* a subspace spanned by coset states, where the coset states are indexed by the set of tolerated error vectors). This idea, which is a natural extension of projective mini-schemes from Definition 5.2.1, is formalized by the following definition.

Definition 5.2.2 (Noisy-projective mini-schemes)**.** *Let $\mathcal{M} = (\mathsf{Bank}, \mathsf{Ver})$ be a mini-scheme. We say that $\mathcal{M}$ is **noisy-projective** if the money state output by Bank is a pure state $\rho_s = |\psi_s\rangle\langle\psi_s|$ and Ver consists of a projective measurement onto the subspace spanned by $\{X^e Z^{e'} |\psi_s\rangle\}_{(e,e')\in\mathcal{E}_q}$, where $\mathcal{E}_q$ is the set of tolerated error vectors for a scheme that can tolerate up to q arbitrary errors on an n-qubit banknote (see equation (3.4.8)).*

Definition 5.2.3 (Valid banknotes for a noise-tolerant scheme)**.** *Let $\mathcal{M}$ be a mini-scheme. In a noisy-projective $\mathcal{M}$, we say that a **valid banknote** is one that has been minted by the bank and then possibly affected by a tolerable error; that is, a pair consisting of a valid serial number s and a state from the set $\{X^e Z^{e'} |\psi_s\rangle\}_{(e,e')\in\mathcal{E}_q}$.*

Definition 5.2.4 (Security of a noisy-projective mini-scheme)**.** *Let $\mathcal{M}$ be a mini-scheme. We say that a noisy-projective $\mathcal{M}$ has **completeness error** ε if $\mathsf{Ver}(\$)$ accepts with probability at least $1-\varepsilon$ for all valid banknotes $\$$ (where valid is understood in the sense of Definition 5.2.3). If $\varepsilon = 0$, then $\mathcal{M}$ has perfect completeness.*

*Let Ver_2 (the **double verifier**) take as input a single serial number s as well as two (possibly entangled) states σ_1 and σ_2, and accept if and only if $\mathsf{Ver}(s, \sigma_1)$ and $\mathsf{Ver}(s, \sigma_2)$ both accept. We say that $\mathcal{M}$ has **soundness error** δ if, given any quantum circuit $\mathscr{C}$ of size $\mathsf{poly}(n)$ (the **counterfeiter**), $\mathsf{Ver}_2(s, \mathscr{C}(\$))$ accepts with probability at most δ. Here, the probability is over the banknote $\$$ output by $\mathsf{Bank}(0^n)$, as well as the behaviour of Ver_2 and $\mathscr{C}$.*

*When $q \geq 1$, we say that a noisy-projective $\mathcal{M}$ is **ϵ-noise-tolerant**, where $\epsilon := q/n$. We say that a noisy-projective $\mathcal{M}$ is **δ-secure** if the scheme has soundness error δ and negligible completeness error.*

5.3 Initial approaches to noise-tolerance

In this section, we briefly discuss two approaches to noise-tolerance that are at first intuitive but then prove to present obstacles. In Section 5.3.1, we treat the banknote of the [AC12] scheme as already being a codeword of a CSS code. In Section 5.3.2, we consider a different direction by adding a layer of error correction onto the [AC12] scheme.

5.3.1 Direct error correction

We can view the banknote subspace state in [AC12] as a codeword in a CSS code. Furthermore, from the perspective of a CSS code (with only one codeword), the verification procedure of [AC12] (checking membership in A and $A^\perp$) can be thought of as error detection where one only checks for membership in the codespace (the common $+1$ eigenspace of the generators), as both procedures amount to projecting onto the

subspace spanned by $|A\rangle$. If we use this CSS code and the error correction procedure outlined in Section 3.4 for noise-tolerance, then we must measure the generators of the code to determine whether the codeword has been affected by a correctable error vector. The immediate difficulty with this approach is that publishing the generators reveals the subspace used for constructing the banknote.

To formally see this, we first need to elaborate on why our CSS codes can only have a single codeword. If a CSS code has more than one codeword, then the verification procedure where one checks membership in the codespace allows all codewords to be accepted; in the noise-free quantum money setting, this is advantageous for a counterfeiter, as multiple states can now be accepted, as opposed to the single state $|A\rangle$. In a noise-tolerant setting, this problem would only worsen.

To be precise in our characterization of applicable CSS codes, we are asking for a subspace C such that $\dim(C) = n/2$ and that both C and $C^\perp$ can correct up to q errors. Then, setting $C_1 = C$ and $C_2 = C$ in the CSS code framework, we get a code that can correct up to q arbitrary errors and where the codespace has $2^{k_1-k_2} = 1$ element, as desired. The single codeword is then the subspace state

$$|C\rangle = \frac{1}{\sqrt{|C|}} \sum_{v \in C} |v\rangle \,.$$

Returning to the task of noise-tolerance, we recall that measuring the generators of a code is how we determine whether a codeword has been affected by a correctable error vector. For an applicable CSS code, the check matrix (which describes the generators), is

$$\left[\begin{array}{c|c} H_{C^\perp} & 0 \\ 0 & H_C \end{array}\right]. \tag{5.3.1}$$

Thus, if we were to publish a description of the generators for a banknote $|C\rangle$, an adversary could determine the parity check matrices which in turn reveals the codespace C. It then seems that to use the generators for tolerating noise, one must "hide" a full description of them.

5.3.2 A layer of error correction

Now suppose that we take a more traditional approach by applying a layer of error correction on top of the oracle scheme of [AC12]. The process for this starts by choosing a stabilizer code and then encoding the banknote subspace state $|A\rangle$. Given a state $|\psi\rangle$, let $|\psi\rangle_L$ denote the encoding of $|\psi\rangle$. The natural way to prepare $|A\rangle_L$ is to construct it from $|0\rangle_L$, just as $|A\rangle$ would be constructed from $|0\rangle$.

The standard encoding procedure for a stabilizer code is to start by encoding the $|0\rangle$ state via the following operation

$$|0\rangle_L \equiv |0^{\otimes k}\rangle_L = \frac{1}{N} \prod_{g_i \in \mathcal{S}_g} \left(I^{\otimes n} + g_i\right) |0^{\otimes n}\rangle \,, \tag{5.3.2}$$

where we recall that $\mathcal{S}_g$ denotes the set of stabilizer generators, and N is for normalization. Note that with the codeword $|0\rangle_L$, one can prepare the other codewords by applying logical operators to $|0\rangle_L$. Now, given the circuit which transforms $|0\rangle$ to $|A\rangle$, one must now translate this circuit into one that operates on the encoded state $|0\rangle_L$ to prepare $|A\rangle_L$.

Assuming that $|A\rangle_L$ can be prepared, one can take two directions. The first is to perform error correction on $|A\rangle_L$, decode it to $|A\rangle$, and then apply the verification procedure. The other approach is to design a fault-tolerant version of the verification procedure which includes the queries to the classical oracle. The former approach is simpler but assumes that the verification procedure is noise-free. In the latter approach, a formal proof of security does not seem to immediately follow from the original security proof of [AC12]. A main difficulty is that a codeword may be quite different from the state it encodes, and so when a layer of error correction is applied to the [AC12] scheme, the adversary would hold the encoded state $|A\rangle_L$, while the original security proof of [AC12] assumes that the adversary holds the state $|A\rangle$. Thus, to prove security for this approach, it must be shown that $|A\rangle_L$ provides no additional advantage over holding $|A\rangle$. More formally, the proof of security uses the inner product adversary method (Section 5.6.1) where the number of queries needed by the adversary crucially relies on the state that the adversary starts with, and thus giving $|A\rangle_L$ as the starting point instead of $|A\rangle$ could reduce the number of queries needed.

A natural question is if applying a layer of error correction would be preferable to the approach taken in this work. Applying a layer of error correction often comes with the practical difficulty of requiring many additional qubits, though it could also come with a favourable trade-off (between noise-tolerance and soundness error). To make a formal comparison, we would need a proof of security for a scheme that accounts for a layer of error correction. We return to this question as a future direction in Section 5.7.

5.4 Formal specification of the scheme

In this section, we give the formal specification of our noisy-projective mini-scheme. The specification given here is made in reference to the subset approach, which relies on classical oracles, denoted as $U_{C_{\mathcal{E}_X}}$ and $U_{C_{\mathcal{E}_Z}}$, to perform verification. The formal specification for the subspace approach is similar except that $U_{Q_{\mathfrak{S}^X}}$ and $U_{Q_{\mathfrak{S}^Z}}$ are used, respectively, in place of $U_{C_{\mathcal{E}_X}}$ and $U_{C_{\mathcal{E}_Z}}$. We define these oracles and show how they actually verify a banknote in Section 5.5, where we describe the two approaches in detail.

To start, we must clarify how the bank actually provides access to the oracles $U_{C_{\mathcal{E}_X}}$ and $U_{C_{\mathcal{E}_Z}}$ in the subset approach. In the final noisy-projective mini-scheme $\mathcal{M} = (\mathsf{Bank}_{\mathcal{M}}, \mathsf{Ver}_{\mathcal{M}})$, the bank, verifier, and counterfeiter will all have access to a

single classical oracle U, which consists of the four components described below. In Section 5.4.1, we show how this specification can be amended for the purpose of using conjugate coding states.

Note that $\mathcal{W}$, in the specification below, corresponds to the set of applicable CSS codes (described in Section 5.3.1) and is formally defined in equation (5.5.1).

- A **banknote generator** $\mathcal{G}(r)$ which takes as input a random string $r \in \{0,1\}^n$, and outputs a set of linearly independent generators $\langle C_r \rangle = \{x_1, \ldots, x_{n/2}\}$ for a subspace $C_r \in \mathcal{W}$, as well as a unique $3n$-bit serial number $z_r \in \{0,1\}^{3n}$. The function $\mathcal{G}$ is chosen uniformly at random, subject to the constraint that the serial numbers are all distinct.[1]

- A **serial number checker** $\mathcal{Z}(z)$, which outputs 1 if $z = z_r$ is a valid serial number for some $\langle C_r \rangle$, and 0 otherwise.

- A **primal subset tester** $\mathcal{T}_{\text{primal}}$ which takes an input of the form $|z\rangle\,|x\rangle$, applies $U_{C_{\mathcal{E}_X}}$ to $|x\rangle$ if $z = z_r$ is a valid serial number for some $\langle C_r \rangle$, and does nothing otherwise.

- A **dual subset tester** $\mathcal{T}_{\text{dual}}$, identical to $\mathcal{T}_{\text{primal}}$ except that it applies $U_{C_{\mathcal{E}_Z}}$ instead of $U_{C_{\mathcal{E}_X}}$.

Then $\mathcal{M} = (\mathsf{Bank}_{\mathcal{M}}, \mathsf{Ver}_{\mathcal{M}})$ is defined as follows:

- $\mathsf{Bank}_{\mathcal{M}}(0^n)$ chooses $r \in \{0,1\}^n$ uniformly at random. Following this, it queries $\mathcal{G}(r) = (z_r, \langle C_r \rangle)$, and outputs the banknote $|\$_r\rangle = |z_r\rangle\,|C_r\rangle$.

- $\mathsf{Ver}_{\mathcal{M}}(\tilde{\$})$ first uses $\mathcal{Z}$ to check that $\tilde{\$}$ has the form (z, ρ), where $z = z_r$ is a valid serial number. If so, it then uses $\mathcal{T}_{\text{primal}}$ and $\mathcal{T}_{\text{dual}}$ to apply $V_{C_r} = \mathsf{H}^{\otimes n}\mathbb{P}_{C_r^\perp}\mathsf{H}^{\otimes n}\mathbb{P}_{C_r}$, and accepts if and only if $V_{C_r}(\rho)$ accepts.

Recall that the purpose of the serial number is to index the money state. In the above formal specification, this is made clearer, as we see how z_r is used to determine which classical oracle to use for verification (for example, see the primal subset tester $\mathcal{T}_{\text{primal}}$).

[1] As noted in [AC12], the banknote generator $\mathcal{G}$ could be implemented with a random oracle. In this implementation, the random oracle is queried with the input r and outputs a fixed but random pair $(\langle C_r \rangle, z_r)$. The constraint that the serial numbers are all distinct would be satisfied with probability $1 - \mathcal{O}(2^{-n})$.

5.4.1 Using conjugate coding states

Given that conjugate coding states are practically simple to prepare, it would be ideal if the money state of our scheme could be prepared with conjugate coding states, rather than directly preparing a subspace state $|A\rangle$. In this section, we show how this can be done with Lemma 3.3.1 and, consequently, alter the banknote generator $\mathcal{G}(r)$ and $\mathsf{Bank}_{\mathcal{M}}$.

Recalling Lemma 3.3.1, we see that generating a uniformly random coset state $|A_{t,t'}\rangle$ over subspaces $A \subseteq \mathbb{F}_2^n$ with dimension $n/2$ is equivalent to generating a basis $\mathcal{B} = \{u_1, \ldots, u_n\}$ and $x, \theta \in \{0,1\}^n$, such that θ has Hamming weight $n/2$, uniformly at random. This leads to the following procedure:

1. Choose $n \geq 2$ to be even. Select $x \in \{0,1\}^n$ uniformly at random and select $\theta \in \{0,1\}^n$, such that θ has Hamming weight $n/2$, uniformly at random. Then form the state $|x\rangle_\theta$.
2. Select a uniformly random basis $\mathcal{B} = \{u_1, \ldots, u_n\}$ of $\mathbb{F}_2^n$.
3. Set $T = \{i : \theta_i = 0\}$ and then set $A = \text{Span}\{u_i : \theta_i = 1\} = \text{Span}\{u_i : i \in \overline{T}\}$. Then A is a uniformly random subspace of $\mathbb{F}_2^n$ of dimension $n/2$.
4. Set $t = \sum_{i \in T} x_i u_i$ and $t' = \sum_{i \in \overline{T}} x_i u^i$.
5. By Lemma 3.3.1, applying $U_{\mathcal{B}}$ to $|x\rangle_\theta$ yields $|A_{t,t'}\rangle$.

In our scheme, a freshly minted money state (not yet corrupted by noise) is a subspace state $|C\rangle$ with $C \in \mathcal{W}$. To generate a uniformly random subspace state $|C\rangle$, we need $t = t' = 0$ in the above construction. A non-trivial choice of x cannot achieve this without contradicting the fact that $\{u_1, \ldots, u_n\}$ forms a basis, thus we must have $x = 0$. So, to generate a uniformly random subspace state under the above construction, we must generate $|0\rangle_\theta$.

To then incorporate conjugate coding states into the formal specification of our scheme, we must alter the behaviour of the banknote generator $\mathcal{G}(r)$ and $\mathsf{Bank}_{\mathcal{M}}$.

- The **banknote generator** $\mathcal{G}(r)$ takes as input a random string $r \in \{0,1\}^n$, and outputs a basis $\mathcal{B}_r = \{u_1, \ldots, u_n\}$ and a $\theta_r \in \{0,1\}^n$ with Hamming weight $n/2$ such that the subspace specified by θ_r belongs to $\mathcal{W}$, as well as a unique $3n$-bit serial number $z_r \in \{0,1\}^{3n}$. The function $\mathcal{G}$ is chosen uniformly at random, subject to the constraint that the serial numbers are all distinct.
- $\mathsf{Bank}_{\mathcal{M}}(0^n)$ chooses $r \in \{0,1\}^n$ uniformly at random. It then queries $\mathcal{G}(r) = (z_r, \theta_r, \mathcal{B}_r)$, and outputs the banknote $|\$_r\rangle = |z_r\rangle\, U_{\mathcal{B}_r}\, |0\rangle_{\theta_r}$.

Intuitively, using conjugate coding states in the minting of a banknote does not affect the security of the scheme since a counterfeiter in both variants of the scheme only receives a subspace state and access to the oracles. Formally, this change is handled in Theorem 5.6.10.

5.5 Verification

In this section, we describe our noise-tolerant quantum money scheme. We present two approaches which we refer to as the the subset approach (Section 5.5.1) and the subspace approach (Section 5.5.2). In each section, we characterize the banknote and the verification procedure. As noted in Section 5.1.2, we cover both approaches here because the subset approach is more general while the subspace approach seems to have an obvious instantiation (Zhandry's technique of using iO to instantiate the oracle from [AC12] could be used); we discuss this latter point further in Section 5.7 as a future direction. Since the subset approach is more general, and conceptually simpler, it will be the primary focus for the rest of this chapter. In Section 5.6, we analyze the security of our scheme.

5.5.1 The subset approach

We now describe the verification procedure that we use throughout the rest of the chapter. To start, let $\mathcal{K}$ be the set of all subspaces $C \subseteq \mathbb{F}_2^n$ such that $\dim C = n/2$. Recalling equation (3.4.17), let $\mathcal{W}$ be the subset of $\mathcal{K}$ defined as

$$\mathcal{W} \coloneqq \{C \in \mathcal{K} : d(C) \geq 2q+1 \text{ and } d(C^\perp) \geq 2q+1\}. \tag{5.5.1}$$

In other words, $\mathcal{W}$ consists of applicable CSS codes: $n/2$-dimensional subspaces C of $\mathbb{F}_2^n$ such that, as classical linear codes, both C and $C^\perp$ can each correct q errors.

A freshly minted money state of our scheme will be a subspace state

$$|C\rangle = \frac{1}{\sqrt{|C|}} \sum_{v \in C} |v\rangle\,, \tag{5.5.2}$$

where $C \in \mathcal{W}$. This is similar to [AC12], where a freshly minted money state is also a subspace state except with $C \in \mathcal{K}$. The reason we must select our subspaces from $\mathcal{W}$ is to ensure that our verification operator, constructed below, is a projector; this is made clear in the proof of Lemma 5.5.1. We discuss the existence of such subspaces in the subspace approach (Section 5.5.2), which also requires the $C \in \mathcal{W}$ condition.

When the state in equation (5.5.2) is corrupted by noise (a bit-flip error X^e and a phase-flip error $Z^{e'}$), it becomes a coset state

$$|C_{e,e'}\rangle = X^e Z^{e'} |C\rangle = \frac{1}{\sqrt{|C|}} \sum_{v \in C} (-1)^{v \cdot e'} |v + e\rangle\,,$$

where e, e' are n-bit vectors. Suppose we wish to tolerate up to q arbitrary errors. We know that, for quantum error correction (and CSS codes, in particular), we can achieve this by tolerating up to q bit-flip and q phase-flip errors (*i.e.*, tolerating the set $\mathcal{E}_q$). Taking inspiration from this, we design our scheme to tolerate the set $\mathcal{E}_q$ and

then show that, as a result, our scheme indeed tolerates q arbitrary errors (though the reasoning for this is not exactly the same as in quantum error correction). We discuss this further after Lemma 5.5.1.

When we say that our scheme should tolerate the set $\mathcal{E}_q$, we mean that we should tolerate the sets $\mathcal{E}_X, \mathcal{E}_Z$ defined by equation (3.4.7). To tolerate these two sets, we use classical oracles $U_{C_{\mathcal{E}_X}}$ and $U_{C_{\mathcal{E}_Z}}$ to check for membership in the subsets $C_{\mathcal{E}_X}$ and $C_{\mathcal{E}_Z}$, respectively, where these subsets are defined as

$$C_{\mathcal{E}_X} \coloneqq \bigcup_{e \in \mathcal{E}_X} C + e \quad \text{and} \quad C_{\mathcal{E}_Z} \coloneqq \bigcup_{e' \in \mathcal{E}_Z} C^{\perp} + e'. \tag{5.5.3}$$

Formally, tolerable bit-flip errors are checked by our first projector $\mathbb{P}_C$ which is constructed from the classical oracle $U_{C_{\mathcal{E}_X}}$. This construction uses the same techniques of [AC12] for implementing a classical oracle as a projector (recall Section 3.3.2 for details). In this context, the construction of $\mathbb{P}_C$ consists of the following steps:

1. For the oracle $U_{C_{\mathcal{E}_X}}$, we introduce a control qubit $|+\rangle$ and apply the classical oracle controlled on $|+\rangle$,

$$CU_{C_{\mathcal{E}_X}}(|x\rangle, |+\rangle) = \frac{|x\rangle|0\rangle + U_{C_{\mathcal{E}_X}}|x\rangle|1\rangle}{\sqrt{2}} = \begin{cases} |x\rangle|-\rangle & \text{if } x \in C_{\mathcal{E}_X}, \\ |x\rangle|+\rangle & \text{if } x \notin C_{\mathcal{E}_X}. \end{cases} \tag{5.5.4}$$

2. We then measure the control qubit and accept if the outcome $|-\rangle$ is observed, and reject otherwise.

The projector $\mathbb{P}_{C^{\perp}}$, which will check for tolerable phase-flip errors, is constructed in exactly the same way except that the classical oracle $U_{C_{\mathcal{E}_Z}}$ is used and, for this projector to work, it will need to be preceded and followed by the Hadamard transform.

Verification is then performed by the following operator

$$V = \mathsf{H}^{\otimes n}\mathbb{P}_{C^{\perp}}\mathsf{H}^{\otimes n}\mathbb{P}_C. \tag{5.5.5}$$

The following lemma is a generalization of lemma 19 from [AC12]. It shows that our verification operator is a projector onto the subspace spanned by the money state $|C\rangle$ and all its tolerated noisy variants. By linearity, the proof of this result extends to arbitrary errors (see the discussion that follows the proof), and so perfect completeness follows from this lemma and the specification given in Section 5.4.

Lemma 5.5.1. *The verification operator V, defined in equation (5.5.5), is the projector $V = \sum_{(e,e') \in \mathcal{E}} |C_{e,e'}\rangle\langle C_{e,e'}|$, i.e., it projects onto the subspace S spanned by $\{|C_{e,e'}\rangle\}_{(e,e') \in \mathcal{E}}$. In particular, the probability of acceptance is given by*

$$\Pr[V(|\psi\rangle) = \textit{accepts}] = \sum_{(e,e') \in \mathcal{E}} |\langle \psi | C_{e,e'} \rangle|^2.$$

Proof: We first show that $V\left|C_{e,e'}\right\rangle = \left|C_{e,e'}\right\rangle$ for any $(e,e') \in \mathcal{E}$,

$$\begin{aligned}
V\left|C_{e,e'}\right\rangle &= V\frac{1}{\sqrt{|C|}}\sum_{v\in C}(-1)^{v\cdot e'}\left|v+e\right\rangle, \\
&= \mathsf{H}^{\otimes n}\mathbb{P}_{C^\perp}\mathsf{H}^{\otimes n}\mathbb{P}_C\frac{1}{\sqrt{|C|}}\sum_{v\in C}(-1)^{v\cdot e'}\left|v+e\right\rangle, \\
&= \mathsf{H}^{\otimes n}\mathbb{P}_{C^\perp}\mathsf{H}^{\otimes n}\frac{1}{\sqrt{|C|}}\sum_{v\in C}(-1)^{v\cdot e'}\left|v+e\right\rangle, \\
&= \mathsf{H}^{\otimes n}\mathbb{P}_{C^\perp}\frac{1}{\sqrt{|C|}}\sum_{v\in C}(-1)^{v\cdot e'}\frac{1}{\sqrt{2^n}}\sum_{z\in 2^n}(-1)^{(v+e)\cdot z}\left|z\right\rangle, \\
&= \mathsf{H}^{\otimes n}\mathbb{P}_{C^\perp}\frac{(-1)^{e\cdot e'}}{\sqrt{2^n|C|}}\sum_{v\in C}\sum_{z\in 2^n}(-1)^{(v+e)\cdot(z+e')}\left|z\right\rangle.
\end{aligned}$$

Setting $z' \equiv z + e'$ in the above expression, we get

$$= \mathsf{H}^{\otimes n}\mathbb{P}_{C^\perp}\frac{(-1)^{e\cdot e'}}{\sqrt{2^n|C|}}\sum_{z'\in 2^n}\sum_{v\in C}(-1)^{(v+e)\cdot z'}\left|z'+e'\right\rangle.$$

To complete the calculation, we use the result that if $z' \in C^\perp$ then $\sum_{v\in C}(-1)^{v\cdot z'} = |C|$ and if $z' \notin C^\perp$ then $\sum_{v\in C}(-1)^{v\cdot z'} = 0$,

$$\begin{aligned}
&= \mathsf{H}^{\otimes n}\mathbb{P}_{C^\perp}\frac{(-1)^{e\cdot e'}}{\sqrt{2^n/|C|}}\sum_{z'\in C^\perp}(-1)^{z'\cdot e}\left|z'+e'\right\rangle, \\
&= \mathsf{H}^{\otimes n}\frac{(-1)^{e\cdot e'}}{\sqrt{|C^\perp|}}\sum_{z'\in C^\perp}(-1)^{z'\cdot e}\left|z'+e'\right\rangle, \\
&= \frac{1}{\sqrt{|C|}}\sum_{v\in C}(-1)^{v\cdot e'}\left|v+e\right\rangle.
\end{aligned}$$

We now show that $V\left|\psi\right\rangle = 0$ for all $\left|\psi\right\rangle$ that are orthogonal to *every* tolerable noisy state $\left|C_{e,e'}\right\rangle$. We start by writing $\left|\psi\right\rangle$ as

$$\left|\psi\right\rangle = \sum_{x\in 2^n} c_x\left|x\right\rangle.$$

Then the orthogonality condition gives

$$\begin{aligned}
0 &= \left\langle\psi\,\middle|\,C_{e,e'}\right\rangle, \\
&= \left(\sum_{x\in 2^n} c_x^*\left\langle x\right|, \frac{1}{\sqrt{|C|}}\sum_{v\in C}(-1)^{v\cdot e'}\left|v+e\right\rangle\right),
\end{aligned}$$

$$\iff 0 = \sum_{x \in C+e} c_x^* (-1)^{(x+e)\cdot e'},$$
$$\iff 0 = \sum_{v \in C} c_{v+e} (-1)^{v \cdot e'}, \tag{5.5.6}$$

for all $(e, e') \in \mathcal{E}$. Then,

$$\begin{aligned} V|\psi\rangle &= \mathsf{H}^{\otimes n} \mathbb{P}_{C^\perp} \mathsf{H}^{\otimes n} \mathbb{P}_C \sum_{x \in 2^n} c_x |x\rangle, \\ &= \mathsf{H}^{\otimes n} \mathbb{P}_{C^\perp} \mathsf{H}^{\otimes n} \sum_{e \in \mathcal{E}_X} \left(\sum_{v \in C} c_{v+e} |v+e\rangle \right), \\ &= \mathsf{H}^{\otimes n} \mathbb{P}_{C^\perp} \sum_{e \in \mathcal{E}_X} \left(\sum_{v \in C} c_{v+e} \left(\frac{1}{\sqrt{2^n}} \sum_{y \in 2^n} (-1)^{(v+e)\cdot y} |y\rangle \right) \right). \end{aligned}$$

Setting $y \equiv y' + e'$ in the above expression, we get

$$\begin{aligned} &= \mathsf{H}^{\otimes n} \mathbb{P}_{C^\perp} \sum_{e \in \mathcal{E}_X} \left(\sum_{v \in C} c_{v+e} \left(\frac{1}{\sqrt{2^n}} \sum_{y' \in 2^n} (-1)^{(v+e)\cdot(y'+e')} |y'+e'\rangle \right) \right), \\ &= \mathsf{H}^{\otimes n} \mathbb{P}_{C^\perp} \sum_{e \in \mathcal{E}_X} \left(\sum_{v \in C} c_{v+e} \left(\frac{1}{\sqrt{2^n}} (-1)^{(v+e)\cdot e'} \sum_{y' \in 2^n} (-1)^{(v+e)\cdot y'} |y'+e'\rangle \right) \right), \\ &= \mathsf{H}^{\otimes n} \sum_{(e,e') \in \mathcal{E}} \left(\sum_{v \in C} c_{v+e} \left(\frac{1}{\sqrt{2^n}} (-1)^{(v+e)\cdot e'} \sum_{z \in C^\perp} |z+e'\rangle \right) \right), \end{aligned}$$

where $e' \in \mathcal{E}_Z$. By rearranging the above expression, we can use the orthogonality condition, equation (5.5.6), to complete the calculation,

$$\begin{aligned} &= \mathsf{H}^{\otimes n} \sum_{(e,e') \in \mathcal{E}} \left(\left(\sum_{v \in C} c_{v+e} (-1)^{v \cdot e'} \right) \left(\frac{(-1)^{e \cdot e'}}{\sqrt{2^n}} \sum_{z \in C^\perp} |z+e'\rangle \right) \right), \\ &= 0. \end{aligned}$$

Then V is a sum of projectors, $V = \sum_{(e,e') \in \mathcal{E}} |C_{e,e'}\rangle \langle C_{e,e'}|$, which is itself a projector. Indeed, V is a projector if and only if $|C_{e,e'}\rangle$ and $|C_{s,s'}\rangle$ are orthogonal whenever $(e, e') \neq (s, s')$.

$$\langle C_{e,e'} | C_{s,s'} \rangle = \left(\frac{1}{\sqrt{|C|}} \sum_{v \in C} (-1)^{v \cdot e'} \langle v+e|, \frac{1}{\sqrt{|C|}} \sum_{z \in C} (-1)^{z \cdot s'} |z+s\rangle \right)$$

The above expression is equal to zero for a fixed $C \in \mathcal{W}$ and when $(e, e') \neq (s, s')$. To see this, first observe that for a fixed v, $\langle v+e \,|\, z+s \rangle = 0$ unless $v + e = z + s$

for some $z \in C$, or rather, $z = v + e + s$, but this is impossible for $e \neq s$, since $v, z \in C$ and, because $d(C) \geq 2q + 1$, we have $e, s, e + s \notin C$, and hence $z \in C$ while $v + e + s \notin C$. So, suppose that $e = s$ but, by the assumption $(e, e') \neq (s, s')$, we now have $e' \neq s'$. In the Hadamard basis, we get

$$\left(\frac{(-1)^{e \cdot e'}}{\sqrt{|C^{\perp}|}} \sum_{v \in C^{\perp}} (-1)^{v \cdot e} \langle v + e' | , \frac{(-1)^{e \cdot e'}}{\sqrt{|C^{\perp}|}} \sum_{z \in C^{\perp}} (-1)^{z \cdot s} | z + s' \rangle \right).$$

Again, we get zero unless $z = v + e' + s'$ but $v, z \in C^{\perp}$ and, because $d(C^{\perp}) \geq 2q + 1$, we have $e', s', e' + s' \notin C^{\perp}$ and hence $z \in C^{\perp}$ while $v + e' + s' \notin C^{\perp}$. Thus, $|C_{e,e'}\rangle$ and $|C_{s,s'}\rangle$ are orthogonal because their error vectors are distinct and because $C \in \mathcal{W}$. ∎

Now recall from Section 3.4.1 that for q arbitrary errors, the error operator E is the product of q single-qubit errors of the form

$$E_i = \alpha_I I + \alpha_X X + \alpha_Z Z + \alpha_{XZ} XZ. \tag{5.5.7}$$

Thus, after relabelling the coefficients, the error operator E takes the form

$$E = \sum_{(e,e') \in A} \alpha_{e,e'} X^e Z^{e'}, \tag{5.5.8}$$

where A is some subset of $\mathcal{E}_q$ which contains at least one pair (e, e') that corresponds to an error of weight q. Then our noisy money state is of the form

$$E | C \rangle = \sum_{(e,e') \in A} \alpha_{e,e'} | C_{e,e'} \rangle . \tag{5.5.9}$$

Now suppose we apply the projector $\mathbb{P}_C$ (the process is similar for $\mathbb{P}_{C^{\perp}}$). That is, we apply the oracle $U_{C_{\mathcal{E}_X}}$ controlled by $|+\rangle$,

$$CU_{C_{\mathcal{E}_X}}(E | C \rangle , |+\rangle) = \sum_{(e,e') \in A} \alpha_{e,e'} | C_{e,e'} \rangle | - \rangle , \tag{5.5.10}$$

$$= \left(\sum_{(e,e') \in A} \alpha_{e,e'} | C_{e,e'} \rangle \right) | - \rangle , \tag{5.5.11}$$

where the first equality follows by the fact that $|C_{e,e'}\rangle \in C_{\mathcal{E}_X}$ for each $(e, e') \in A$. We will observe $|-\rangle$ when we measure the control qubit, and thus we accept the state. In general, linearity and Lemma 5.5.1 ensures that a state affected by q arbitrary errors will always be accepted. Notably, this process does not cause a "collapse" of the arbitrary error as in the typical error correction procedure (see Section 3.4.2).

We end this section with the following lemma, which we will need to prove Theorem 5.6.8. The need for this lemma is a consequence of the $C \in \mathcal{W}$ condition.

Lemma 5.5.2. *Let $f : \mathbb{F}_2^n \to \mathbb{F}_2^n$ be an invertible linear isometry. Then for any $C \in \mathcal{W}$, we have $f(C) \in \mathcal{W}$.*

Proof: A classical linear code with distance at least $2q + 1$ is able to correct up to q errors. Recall that the distance of a code C is defined as

$$d(C) \coloneqq \min_{x,y \in C, x \neq y} d(x, y),$$

where $d(x, y)$ is the Hamming distance between x and y. Since f is an isometry (*i.e.*, distance-preserving map) we have $d(f(x), f(y)) = d(x, y)$ for all $x, y \in C$ and hence $d(C) = d(f(C))$. Hence, $f(C)$ has the same distance and is able to correct the same number of errors. ∎

5.5.2 The subspace approach

In this approach to verification, let $C \in \mathcal{W}$, where $\mathcal{W}$ is defined by equation (5.5.1) (*i.e.*, C is an applicable CSS code, as described in Section 5.3.1). Then observe that an error syndrome of all zeros indicates a codeword that has been unaffected by noise (see Remark 5.5.3); put another way, the codeword belongs to the common $+1$ eigenspace of all the generators. If the error syndrome contains, for instance, two 1's, then we can interpret this as the banknote state belonging to the intersection of all $+1$ eigenspaces and two -1 eigenspaces. From this perspective, each valid syndrome corresponds to a particular intersection of eigenspaces.

Remark 5.5.3. In general, an error syndrome of all zeros indicates that the codeword has been unaffected by noise *or* affected by undetectable noise. To elaborate, let L be a logical operator of a stabilizer code and let $|\psi\rangle$ be a codeword. Recall that because L commutes with all elements of the stabilizer, $L|\psi\rangle$ will also produce a syndrome of all zeros (*i.e.*, $L|\psi\rangle$ is still in the codespace). So, it is not necessarily true that a syndrome of all zeros correlates to a noiseless codeword. However, in the case of an applicable CSS code where there is only a single codeword (*i.e.*, no logical operators), we can say that the all-zero syndrome implies a noiseless codeword.

Formally, let $\mathfrak{S}$ denote the set of good syndromes (*i.e.*, if $s \in \mathfrak{S}$, then the error that produced s is correctable). Under the stabilizer formalism, every $s \in \mathfrak{S}$ corresponds to a particular intersection of eigenspaces of the generators. Denote such a subspace as Q_s, where $s \in \mathfrak{S}$ indexes the subspace. Then for each $s \in \mathfrak{S}$, we want to check membership in Q_s.

In the context of a CSS code, the syndrome is a concatenation $s = (s^X, s^Z)$, where s^X (s^Z) denotes the syndrome that arises from bit-flip (phase-flip) errors. So, for a given $s \in \mathfrak{S}$, instead of checking membership in the single subspace Q_s, one checks for

membership in two subspaces, Q_{s^X} and Q_{s^Z}. Let $\mathfrak{S}^X, \mathfrak{S}^Z$ denote, respectively, the set of syndromes that correspond to correctable bit-flip and phase-flip errors. Then let $U_{Q_{\mathfrak{S}^X}}, U_{Q_{\mathfrak{S}^Z}}$ denote the classical oracles that check for membership in the subspaces $Q_{\mathfrak{S}^X}, Q_{\mathfrak{S}^Z}$, respectively, where these subspaces are defined as,

$$Q_{\mathfrak{S}^X} \coloneqq \bigcup_{s^X \in \mathfrak{S}^X} Q_{s^X} \quad \text{and} \quad Q_{\mathfrak{S}^Z} \coloneqq \bigcup_{s^Z \in \mathfrak{S}^Z} Q^{\mathsf{H}}_{s^Z}.$$

where $Q^{\mathsf{H}}_{s^Z}$ denotes Q_{s^Z} in the Hadamard basis. In a similar manner to the subset approach, the first projector $\mathbb{P}_C$ is constructed as follows:

1. For the oracle $U_{Q_{\mathfrak{S}^X}}$, we introduce a control qubit $|+\rangle$ and apply the classical oracle controlled on $|+\rangle$.

2. We then measure the control qubit and accept if the outcome $|-\rangle$ is observed, and reject otherwise.

The process for phase-flip errors is only slightly different: repeat the above steps using $Q_{\mathfrak{S}^Z}$ to construct the second projector $\mathbb{P}_{C^\perp}$. The final verification operator would then be given by $V = \mathsf{H}^{\otimes n}\mathbb{P}_{C^\perp}\mathsf{H}^{\otimes n}\mathbb{P}_C$.

With this V, Lemma 5.5.1 also holds, and for the same reasons as in the subset approach, the verification operator V can be used to accept arbitrary errors. It is worth noting that if we do not use an applicable CSS code, then Lemma 5.5.1 does not necessarily hold, since we can no longer guarantee that V only accepts a codeword and its tolerated noisy variants. A codeword affected by a logical operator (*i.e.*, a highly corrupted codeword) could be accepted, but this possibility is eliminated by using an applicable CSS code.

Remark 5.5.4. The subset approach in Section 5.5.1 can be viewed as a generalization of this subspace approach. To see this, we can compare $C_{\mathcal{E}_X}$, as defined in equation (5.5.3), with $Q_{\mathfrak{S}^X}$. The set $C_{\mathcal{E}_X}$ is a union of subsets where each subset in the union corresponds to a tolerated bit-flip error; likewise, each subspace in the union $Q_{\mathfrak{S}^X}$ is also a subset which corresponds to tolerating a bit-flip error. Additionally, the verification operator V is constructed in the same way for both cases and corresponds to a projector onto the same subspace. The subspace approach is then an interesting case that closely resembles error correction and offers the possibility of being instantiated in the plain model (see Section 5.7).

To end this section, we remark that the existence of applicable CSS codes, $C \in \mathcal{W}$, is guaranteed by the Gilbert-Varshamov bound for CSS codes, given by equation (3.4.23). In our case, we have $k = 0$, and so the bound becomes

$$0 \geq 1 - 2H\left(\frac{2q}{n}\right). \tag{5.5.12}$$

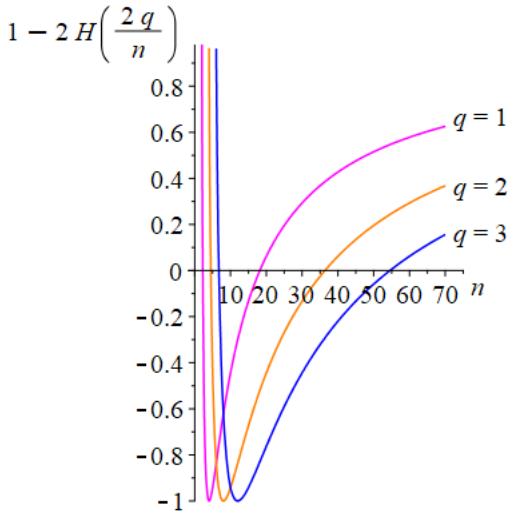

Figure 5.1: A plot of the right-hand-side of equation (3.4.23) for various choices of q. For a fixed choice of q, equation (3.4.23) guarantees the existence of an applicable CSS codes as long as n is chosen such that the corresponding point on the plot is below the horizontal axis.

If we now fix a value for q, we can find an appropriate n such that equation (5.5.12) is satisfied. This relationship is visualized in Figure 5.1.

As an example of an applicable CSS code, consider the following $[6, 3]$ classical linear code C,

$$G_C = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}, \quad H_C = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}.$$

Since the distance of this code is $d = 3$ and the inequality $d \geq 2q + 1$ is satisfied for $q = 1$, this code can correct a single error. The dual code $C^{\perp}$ has the following generator and parity check matrix,

$$G_{C^{\perp}} = H_C^T = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad H_{C^{\perp}} = G_C^T = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}.$$

The $C^\perp$ code also has a distance of $d = 3$ and can thus also correct a single error. Hence, as a CSS code with $C_1, C_2 = C$ we can correct a single arbitrary error. The single codeword of this CSS code is given by

$$|C\rangle = \frac{1}{\sqrt{8}}\big(\,|000000\rangle + |100011\rangle + |010110\rangle + |001101\rangle + |110101\rangle + |101110\rangle + |011011\rangle + |111000\rangle\,\big).$$

Remark 5.5.5. Although the existence of applicable CSS codes is easily determined (with the Gilbert-Varshamov bound, equation (5.5.12)), explicitly constructing them for arbitrary n and q could pose a difficult challenge. It would be useful, for implementation purposes, to have a method for constructing such codes, though we are unaware of any such techniques.

5.6 Security

To prove security, we follow the same approach of [AC12], changing the proofs as necessary. To start, we look at one of the crucial components from [AC12] that was used for proving security: the amplification of counterfeiters theorem. Intuitively, the theorem shows that a counterfeiter that can copy a banknote with small success can be used to copy a banknote almost perfectly; hence, it suffices to prove security against counterfeiters that can copy almost perfectly. The analogous statement for noisy-projective mini-schemes is given as Theorem 5.6.1 below. The proof of this theorem in [AC12] essentially starts with the following equality

$$\Pr[V(\rho) = \text{accepts}] = F(\rho, S)^2, \tag{5.6.1}$$

where $F(\rho, S)$ is the fidelity of ρ with the subspace S that the verification algorithm V projects to. Unlike [AC12], our space S is not 1-dimensional (see Lemma 5.5.1), though we can still rely on the proof in [AC12] to get the same complexity bound. However, we will later see how having a "larger" S will lead to a trade-off between security and noise-tolerance.

Theorem 5.6.1 (Amplification of counterfeiters)**.** *Let $\mathcal{M} = (\mathsf{Bank}, \mathsf{Ver})$ be a noisy-projective mini-scheme, let S be the subspace that the verification operator V projects to, and let $\$ = (s, \rho)$ be a valid banknote (recall Definition 5.2.3). Suppose there exists a counterfeiter $\mathscr{C}$ that produces a counterfeit state $\mathscr{C}(\$)$ such that*

$$F(\mathscr{C}(\$), S^{\otimes 2}) \geq \sqrt{\varepsilon}$$

and hence

$$\Pr\big[V^{\otimes 2}(\mathscr{C}(\$)) = \textit{accepts}\big] \geq \varepsilon.$$

Then for all $\delta > 0$, there is also a modified counterfeiter $\mathscr{C}'$ (depending only on ε and δ, not \$), which makes

$$\mathcal{O}\left(\frac{\log 1/\delta}{\sqrt{\varepsilon}(\sqrt{\varepsilon}+\delta^2)}\right) \tag{5.6.2}$$

queries to $\mathscr{C}, \mathscr{C}^{-1}$, and V and which satisfies

$$F(\mathscr{C}'(\$), S^{\otimes 2}) \geq 1-\delta$$

which implies

$$\Pr\left[V^{\otimes 2}(\mathscr{C}'(\$)) = \textit{accepts}\right] \geq (1-\delta)^2.$$

Proof: We first note that

$$\begin{aligned}
\Pr[V(\rho) = \text{accepts}] &= \operatorname{tr}(V\rho), \\
&= \operatorname{tr}\left(\sum_{(e,e')\in\mathcal{E}} |C_{e,e'}\rangle\langle C_{e,e'}|\,\rho\right), \\
&= \sum_{(e,e')\in\mathcal{E}} \operatorname{tr}\left(|C_{e,e'}\rangle\langle C_{e,e'}|\,\rho\right), \\
&= \sum_{(e,e')\in\mathcal{E}} \langle C_{e,e'}|\,\rho\,|C_{e,e'}\rangle, \\
&= F(\rho, S)^2.
\end{aligned}$$

Then, for the case of verifying two banknotes supplied by the counterfeiter $\mathscr{C}$, we have,

$$\begin{aligned}
\Pr\left[V^{\otimes 2}(\mathscr{C}(\rho)) = \text{accepts}\right] &= \sum_{(e,e'),(t,t')\in\mathcal{E}} \langle C_{e,e'}| \otimes \langle C_{t,t'}|\,\mathscr{C}(\rho)\,|C_{e,e'}\rangle \otimes |C_{t,t'}\rangle, \\
&= \sum_{(e,e'),(t,t')\in\mathcal{E}} F(\mathscr{C}(\rho), |C_{e,e'}\rangle \otimes |C_{t,t'}\rangle)^2, \\
&= F(\mathscr{C}(\rho), S^{\otimes 2})^2.
\end{aligned}$$

By assumption of the theorem,

$$F(\mathscr{C}(\rho), S^{\otimes 2}) \geq \sqrt{\varepsilon},$$

and thus we indeed have

$$\Pr\left[V^{\otimes 2}(\mathscr{C}(\rho)) = \text{accepts}\right] \geq \varepsilon.$$

Put simply, one can now apply a fixed-point Grover search with $\mathscr{C}(\rho)$ as the initial state and $S^{\otimes 2}$ as the target subspace to produce a state ρ' such that $F(\rho', S^{\otimes 2}) \geq 1-\delta$.

As shown in [AC12] (see Section 2.3 and the proof of Theorem 13 for details), this can be done in

$$\mathcal{O}\left(\frac{\log 1/\delta}{\sqrt{\varepsilon}(\sqrt{\varepsilon}+\delta^2)}\right) \tag{5.6.3}$$

Grover iterations. Each iteration can be implemented using $\mathcal{O}(1)$ queries to $\mathscr{C}, \mathscr{C}^{-1}$, and V. Thus, the total number of queries to $\mathscr{C}, \mathscr{C}^{-1}$, and V is also given by equation (5.6.3). Then the probability of ρ' being accepted is

$$\begin{aligned}\Pr\left[V^{\otimes 2}(\rho') = \text{accepts}\right] &= \sum_{(e,e'),(t,t')\in\mathcal{E}} F(\rho', |C_{e,e'}\rangle \otimes |C_{t,t'}\rangle)^2, \\ &= F(\rho', S^{\otimes 2})^2, \\ &\geq (1-\delta)^2.\end{aligned}$$

∎

5.6.1 The inner-product adversary method

Put simply, proving security amounts to proving that a counterfeiter, who possesses a copy of $|\psi\rangle$, needs to make exponentially many queries to the verification algorithm to prepare $|\psi_e\rangle\,|\psi_t\rangle$, where $|\psi_e\rangle$ and $|\psi_t\rangle$ are possibly different noisy copies of $|\psi\rangle$. For simplicity, let us forget about noise-tolerance so that the counterfeiter is now trying to prepare $|\psi\rangle\,|\psi\rangle$. Suppose that $|\psi\rangle$ and $|\phi\rangle$ are two possible quantum money states. Consider a counterfeiting algorithm $\mathscr{C}$ that is executed in parallel to clone $|\psi\rangle$ and $|\phi\rangle$. If, say, $\langle\psi|\phi\rangle = 1/2$, and if the counterfeiting algorithm $\mathscr{C}$ succeeds perfectly (*i.e.*, it maps $|\psi\rangle$ to $|\psi\rangle^{\otimes 2}$ and $|\phi\rangle$ to $|\phi\rangle^{\otimes 2}$), then $\langle\psi|^{\otimes 2}\,|\phi\rangle^{\otimes 2} = 1/4$, meaning that $\mathscr{C}$ must decrease the inner product by at least $1/4$. But the analysis of [AC12] shows that the *average* inner product can decrease by at most $1/\mathsf{exp}(n)$ from a single query, which results in $2^{\Omega(n)}$ queries being needed.

This analysis is done using the inner-product adversary method developed in [AC12] (which is an adaptation of the quantum adversary method in [Amb02]). Let us now recall the basic ideas of the method (for further details, see [AC12]). The idea is to get an upper bound on the *progress* a quantum algorithm $\mathscr{C}$ can make at distinguishing pairs of oracles, as a result of a single query. To make this a bit more formal, let $|\Psi_t^U\rangle$ be $\mathscr{C}$'s state after t queries. For now, assume that $|\Psi_0^U\rangle = |\Psi_0^V\rangle$ for all oracles U and V. After the final query T, we must have, say, $|\langle\Psi_T^U|\Psi_T^V\rangle| \leq 1/2$. Thus, if we are able to show that $|\langle\Psi_t^U|\Psi_t^V\rangle|$ can decrease by at most ε as the result of a single query, then it follows that $\mathscr{C}$ must make $\Omega(1/\varepsilon)$ queries.

The difficulty, overcome in [AC12], is that in the quantum money framework, $\mathscr{C}$ starts with a valid banknote and hence it is not generally true that $|\Psi_0^U\rangle = |\Psi_0^V\rangle$.

This starting point is advantageous for $\mathscr{C}$ since it can allow them to do much better in decreasing $|\langle\Psi_t^U|\Psi_t^V\rangle|$ as a result of a single query. The solution in [AC12] is to choose a distribution $\mathcal{D}$ over oracle pairs (U,V) and then analyze the expected inner product

$$\mathop{\mathbb{E}}_{(U,V)\sim\mathcal{D}}\left[|\langle\Psi_t^U|\Psi_t^V\rangle|\right]$$

and how much it can decrease as the result of a single query to U or V. Analysis shows that, although $\mathscr{C}$ can succeed in significantly reducing $|\langle\Psi_t^U|\Psi_t^V\rangle|$ for some oracle pairs (U,V), they cannot do so for most pairs.

Formal description

This section, which gives the formal description of the inner-product adversary method, is taken almost directly from [AC12]. Let $\mathcal{O}$ denote a set of quantum oracles acting on n qubits each. For each $U\in\mathcal{O}$, assume there exists a subspace $S_U\subseteq\mathbb{C}^{2n}$ such that

(i) $U|\psi\rangle = -|\psi\rangle$ for all $|\psi\rangle\in S_U$

(ii) $U|\eta\rangle$ for all $|\eta\rangle\in S_U^{\perp}$.

Let $R\subset\mathcal{O}\times\mathcal{O}$ be a symmetric binary relation on $\mathcal{O}$, with the properties that

(i) $(U,U)\notin R$ for all $U\in\mathcal{O}$, and

(ii) for every $U\in\mathcal{O}$ there exists a $V\in\mathcal{O}$ such that $(U,V)\in R$.

Suppose that for all $U\in\mathcal{O}$ and all $|\eta\rangle\in S_U^{\perp}$, we have

$$\mathop{\mathbb{E}}_{V:(U,V)\in R}\left[F(|\eta\rangle, S_V)^2\right]\leq\varepsilon,$$

where $F(|\eta\rangle, S_V) = \max_{|\psi\rangle\in S_V}|\langle\eta|\psi\rangle|$ is the fidelity between $|\eta\rangle$ and S_V. Let $\mathscr{C}$ be a quantum oracle algorithm, and let $\mathscr{C}^U$ denote $\mathscr{C}$ run with the oracle $U\in\mathcal{O}$. Suppose $\mathscr{C}^U$ begins in the state $|\Psi_0^U\rangle$ (possibly dependent on U). Let $|\Psi_t^U\rangle$ denote the state of $\mathscr{C}^U$ immediately after the t^{th} query. Also, define a *progress measure* p_t by

$$p_t \coloneqq \mathop{\mathbb{E}}_{V:(U,V)\in R}\left[|\langle\Psi_t^U|\Psi_t^V\rangle|\right].$$

The following lemma bounds how much p_t can decrease as the result of a single query. See [AC12] for a proof.

Lemma 5.6.2 (Bound on progress rate [AC12])**.**

$$p_t\geq p_{t-1}-4\sqrt{\varepsilon}.$$

As a result of Lemma 5.6.2, we have the following result.

Lemma 5.6.3 (Inner-product adversary method [AC12])**.** *Suppose that we initially have* $|\langle\Psi_0^U|\Psi_0^V\rangle| \geq c$ *for* $(U,V) \in R$*, whereas by the end we need* $|\langle\Psi_T^U|\Psi_T^V\rangle| \leq d$ *for all* $(U,V) \in R$*. Then* $\mathscr{C}$ *must make*

$$T = \Omega\left(\frac{c-d}{\sqrt{\varepsilon}}\right)$$

oracle queries.

5.6.2 Applying the method

For much of the security proof, we suppose that the counterfeiter only has access to the oracles $(U_{C_{\mathcal{E}_X}}, U_{C_{\mathcal{E}_Z}})$ as opposed to the oracles in the noisy-projective mini-scheme specified in Section 5.4. The generalization to the full scheme will be done later in Theorem 5.6.10.

We start the analysis by thinking of the pair of oracles $(U_{C_{\mathcal{E}_X}}, U_{C_{\mathcal{E}_Z}})$ as a single oracle. In the case of [AC12], they have the pair of oracles $(U_A, U_{A^\perp})$ and the corresponding single oracle is constructed in the following way. First, they consider a subset $A^* \subset \{0,1\}^{n+1}$ defined by

$$A^* \coloneqq (0, A) \cup (1, A^\perp).$$

Then they let S_{A^*} denote the subspace of $\mathbb{C}^{2n+1}$ that is spanned by basis states $|x\rangle$ such that $x \in A^*$. This allows their collection of oracles $(U_A, U_{A^\perp})$ to be thought of as a single oracle U_{A^*} which satisfies

$$U_{A^*}|\psi\rangle = \begin{cases} -|\psi\rangle, & \text{if } |\psi\rangle \in S_{A^*} \\ |\psi\rangle, & \text{if } |\psi\rangle \in S_{A^*}^\perp \end{cases}.$$

To generalize this to our setting, first suppose that our scheme only tolerates the errors $\mathcal{E} = \{(e,e'),(t,t')\}$. Then the subset $C^* \subset \{0,1\}^{n+2}$, defined by

$$C^* \coloneqq (00, C+e) \cup (01, C^\perp + e') \cup (10, C+t) \cup (11, C^\perp + t'), \tag{5.6.4}$$

would be a generalization of A^*. Let us also introduce the following notation which naturally generalizes for a larger set of error vectors,

$$\begin{aligned} C^*_{\mathcal{E}_X} &\coloneqq (00, C+e) \cup (10, C+t), \\ C^*_{\mathcal{E}_Z} &\coloneqq (01, C^\perp + e') \cup (11, C^\perp + t'). \end{aligned}$$

In general, every error vector from $\mathcal{E}_X$ and $\mathcal{E}_Z$ corresponds to a term in the union that defines C^*. So, we would have $2|\mathcal{E}_X|$ (*i.e.*, $|\mathcal{E}_X| + |\mathcal{E}_Z|$) terms in the union, meaning $C^* \subset \{0,1\}^{n+k}$ where k is such that

$$2^k = 2|\mathcal{E}_X| \Leftrightarrow k = 1 + \log(|\mathcal{E}_X|). \tag{5.6.5}$$

Now let S_{C^*} be the subspace of $\mathbb{C}^{2^{n+k}}$ that is spanned by basis states $|x\rangle$ such that $x \in C^*$. Then our pair of oracles $(U_{C_{\mathcal{E}_X}}, U_{C_{\mathcal{E}_Z}})$ corresponds to the single oracle U_{C^*} which satisfies

$$U_{C^*}|\psi\rangle = \begin{cases} -|\psi\rangle, & \text{if } |\psi\rangle \in S_{C^*} \\ |\psi\rangle, & \text{if } |\psi\rangle \in S_{C^*}^{\perp} \end{cases}. \tag{5.6.6}$$

A notable consequence of this generalization to the noisy setting is that a single query to $U_{C_{\mathcal{E}_X}}$ equates to $|\mathcal{E}_X|$ queries to U_{C^*}, and likewise for $U_{C_{\mathcal{E}_Z}}$. To see this, observe the example of C^* in equation (5.6.4) where a query to $U_{C_{\mathcal{E}_X}}$ is achieved by preparing ancilla qubits set to $|00\rangle$ and $|10\rangle$ and then querying U_{C^*} with each of these ancilla qubits. For security, this means that the lower bound for the number of queries to $(U_{C_{\mathcal{E}_X}}, U_{C_{\mathcal{E}_Z}})$ is the lower bound for queries to U_{C^*} divided by $2|\mathcal{E}_X|$. We return to this point in Corollary 5.6.9.

The following theorem shows that, for the case of perfect counterfeiting, exponentially many queries are needed. In [AC12], perfect counterfeiting meant that the counterfeiter, who possessed a copy of $|C\rangle$, had to prepare $|C\rangle \otimes |C\rangle$. In our noise-tolerant setting, the counterfeiter has more flexibility by only needing to prepare noisy copies of $|C\rangle$. This difference adds some extra steps to the proof but the lower bound on the needed queries remains the same as the analogous result in [AC12].

Theorem 5.6.4 (Lower bound for perfect counterfeiting)**.** *Given one copy of $|C\rangle$, as well as oracle access to U_{C^*}, a counterfeiter needs $\Omega\left(2^{n/4}\right)$ queries to prepare $|C_{e,e'}\rangle \otimes |C_{t,t'}\rangle$, where $(e,e'),(t,t') \in \mathcal{E}$, with certainty (for a worst-case $|C\rangle$).*

Proof: Let the set $\mathcal{O}$ contain U_{C^*} for every possible subspace $C \subseteq \mathbb{F}_2^n$ such that $\dim(C) = n/2$. Let $(U_{C^*}, U_{D^*}) \in R$ if and only if $\dim(C \cap D) = n/2 - 1$. Then given $U_{C^*} \in \mathcal{O}$ and $|\eta\rangle \in S_{C^*}^{\perp}$, let

$$|\eta\rangle = \sum_{x \in \{0,1\}^{n+k} \setminus C^*} \alpha_x |x\rangle.$$

We have

$$\mathop{\mathbb{E}}_{U_{D^*}:(U_{C^*},U_{D^*})\in R} \left[F(|\eta\rangle, S_{D^*})^2\right] = \mathop{\mathbb{E}}_{\substack{D:\dim(D)=n/2,\\ \dim(C\cap D)=n/2-1}} \left[\sum_{x \in D^* \setminus C^*} |\alpha_x|^2\right], \tag{5.6.7}$$

$$\leq \max_{x \in \{0,1\}^{n+k} \setminus C^*} \left(\Pr_{\substack{D:\dim(D)=n/2,\\ \dim(C\cap D)=n/2-1}} [x \in D^*]\right), \tag{5.6.8}$$

$$= \max_{x \in \{0,1\}^{n+k-1} \setminus C^*_{\mathcal{E}_X}} \left(\Pr_{\substack{D\,:\dim(D)=n/2,\\ \dim(C\cap D)=n/2-1}} [x \in D^*_{\mathcal{E}_X}] \right), \tag{5.6.9}$$

$$= \frac{|D^*_{\mathcal{E}_X} \setminus C^*_{\mathcal{E}_X}|}{|\{0,1\}^{n+k-1} \setminus C^*_{\mathcal{E}_X}|}, \tag{5.6.10}$$

$$= \frac{|D^*_{\mathcal{E}_X} \setminus C^*_{\mathcal{E}_X}|}{|\mathcal{E}_X| 2^n - |\mathcal{E}_X| 2^{n/2}}, \tag{5.6.11}$$

$$\leq \frac{|\mathcal{E}_X| 2^{n/2} - 2^{n/2-1}}{|\mathcal{E}_X| 2^n - |\mathcal{E}_X| 2^{n/2}}, \tag{5.6.12}$$

$$= \frac{|\mathcal{E}_X| - 2^{-1}}{|\mathcal{E}_X| \left(2^{n/2} - 1\right)}, \tag{5.6.13}$$

$$\leq \frac{1}{2^{n/2} - 1}, \tag{5.6.14}$$

$$\leq \frac{1}{2^{n/2-1}}. \tag{5.6.15}$$

Equation (5.6.7) uses the definition of the fidelity. As done in [AC12], equation (5.6.8) uses the easy direction of the minimax theorem. Equation (5.6.9) uses the symmetry between $C^*_{\mathcal{E}_X}$ and $C^*_{\mathcal{E}_Z}$. Equation (5.6.11) uses equation (5.6.5) and the following facts: each coset has the same number of elements as the group; two cosets for the same group are either disjoint or identical; and that there is a coset for each error vector. Equation (5.6.12) uses the fact that $|D^*_{\mathcal{E}_X} \setminus C^*_{\mathcal{E}_X}| = |D^*_{\mathcal{E}_X}| - |D^*_{\mathcal{E}_X} \cap C^*_{\mathcal{E}_X}|$ and that the intersection of two cosets from D and C are either empty or a coset of $D \cap C$; while we do not know how many distinct coset intersections we have, we know that there is at least one such intersection: the trivial coset $C \cap D$ which has $2^{n/2-1}$ elements. The conclusion here is that $\varepsilon := 2^{-n/2+1}$.

Now fix $(U_{C^*}, U_{D^*}) \in R$. Then initially $|\langle C | D \rangle| = 1/2$. The counterfeiter must map $|C\rangle$ to some state $|f_C\rangle := |C_{e,e'}\rangle |C_{t,t'}\rangle |\text{garbage}_C\rangle$, where $(e, e'), (t, t') \in \mathcal{E}$, and likewise for $|D\rangle$. Thus, $|\langle f_C | f_D \rangle| \leq 1/4$. To see this, observe the following:

$$\begin{aligned}
\langle C_{e,e'} | D_{t,t'} \rangle &= \left(\frac{1}{\sqrt{|C|}} \sum_{v \in C} (-1)^{v \cdot e'} \langle v + e | , \frac{1}{\sqrt{|D|}} \sum_{z \in D} (-1)^{z \cdot t'} |z + t\rangle \right), \\
&= \frac{1}{2^{n/2}} \left(\sum_{v \in C} (-1)^{v \cdot e'} \langle v + e | , \sum_{z \in D} (-1)^{z \cdot t'} |z + t\rangle \right), \\
&\leq \frac{1}{2^{n/2}} \sum_{v \in C} \sum_{z \in D} \langle v + e | z + t \rangle .
\end{aligned}$$

Now suppose that the intersection of the two cosets, $C + e \cap D + t$ is non-empty. Then we can interpret the above sum of inner-products as the number of elements in

a coset of $C \cap D$, which is equal to $|C \cap D| = 2^{n/2-1}$. Hence,

$$\langle C_{e,e'}|D_{t,t'}\rangle \leq \frac{2^{n/2-1}}{2^{n/2}} = \frac{1}{2},$$

which indeed implies $|\langle f_C|f_D\rangle| \leq 1/4$.

Then by Lemma 5.6.3, with $c = 1/2, d = 1/4$, and $\varepsilon = 2^{-n/2+1}$,

$$\Omega\left(\frac{c-d}{\sqrt{\varepsilon}}\right) = \Omega\left(2^{(n-2)/4}\right) = \Omega\left(2^{n/4}\right).$$

∎

Remark 5.6.5. It would be practically interesting to study the situation where the counterfeiter starts with an *imperfect* copy of $|C\rangle$. This could model the scenario where the initial transfer from the bank is done through a noisy channel, and so a counterfeiter would start with a money state that has already incurred errors. Intuitively, this difference would not result in a substantial change to the analysis. To formally analyze this case, one could apply the inner product adversary method, as done in Theorem 5.6.4, but the "starting point" of $|\langle C|D\rangle| = 1/2$ would now be different on account of the initial noise.

The following corollary shows that, for the case where the counterfeiter succeeds almost perfectly, exponentially many queries are still needed.

Corollary 5.6.6 (Lower bound for small-error counterfeiting)**.** *Given one copy of $|C\rangle$, as well as oracle access to U_{C^*}, a counterfeiter needs $\Omega(2^{n/4})$ queries to prepare a state ρ such that*

$$\Big(\langle C_{e,e'}| \otimes \langle C_{t,t'}|\Big)\rho\Big(|C_{e,e'}\rangle \otimes |C_{t,t'}\rangle\Big) \geq 1 - \epsilon$$

for small ϵ (say $\epsilon = 0.0001$) and some $(e,e'),(t,t') \in \mathcal{E}$, for a worst case $|C\rangle$. The probability that this state passes verification is

$$\Pr\left[V^{\otimes 2}(\rho) = \textit{accepts}\right] \geq 1 - \epsilon.$$

Proof: Let $|\langle C|D\rangle| = c$. If the counterfeiter succeeds, it must map $|C\rangle$ to some state ρ_C such that

$$\Big(\langle C_{\tilde{e},\tilde{e}'}| \otimes \langle C_{\tilde{t},\tilde{t}'}|\Big)\rho_C\Big(|C_{\tilde{e},\tilde{e}'}\rangle \otimes |C_{\tilde{t},\tilde{t}'}\rangle\Big) \geq 1 - \epsilon,$$

for some $(\tilde{e},\tilde{e}'),(\tilde{t},\tilde{t}') \in \mathcal{E}$. Likewise, it must map $|D\rangle$ to some state ρ_D such that

$$\Big(\langle C_{\tilde{s},\tilde{s}'}| \otimes \langle C_{\tilde{r},\tilde{r}'}|\Big)\rho_D\Big(|C_{\tilde{s},\tilde{s}'}\rangle \otimes |C_{\tilde{r},\tilde{r}'}\rangle\Big) \geq 1 - \epsilon,$$

for some $(\tilde{s}, \tilde{s}'), (\tilde{r}, \tilde{r}') \in \mathcal{E}$

Let $|f_C\rangle, |f_D\rangle$ be purifications of ρ_C, ρ_D respectively. Then by Lemma 3.3.22

$$\begin{aligned} |\langle f_C|f_D\rangle| &\leq F(\rho_C, \rho_D), \\ &\leq \left| \langle C_{\tilde{e},\tilde{e}'}| \otimes \langle C_{\tilde{t},\tilde{t}'}| \, |D_{s,s'}\rangle \otimes |D_{r,r'}\rangle \right| + 2\epsilon^{1/4}, \\ &\leq c^2 + 2\epsilon^{1/4}, \end{aligned}$$

where the upper bound is achieved by assuming that the above coset intersections are non-empty. Thus, with $d := c^2 + 2\epsilon^{1/4}$ in Lemma 5.6.3,

$$\Omega\left(\frac{c-d}{\sqrt{\varepsilon}}\right) = \Omega\left(\left(c - c^2 - 2\epsilon^{1/4}\right)\left(2^{(n-2)/4}\right)\right).$$

Fixing $c = 1/2$ gives $\Omega\left(2^{n/4}\right)$

The probability that such a state ρ passes verification is

$$\begin{aligned} \Pr\left[V^{\otimes 2}(\rho) = \text{accepts}\right] &= \sum_{(e,e')',(t,t')\in\mathcal{E}} \langle C_{e,e'}| \otimes \langle C_{t,t'}| \, \rho \, |C_{e,e'}\rangle \otimes |C_{t,t'}\rangle, \\ &\geq \Big(\langle C_{\tilde{e},\tilde{e}'}| \otimes \langle C_{\tilde{t},\tilde{t}'}| \Big)\rho\Big(|C_{\tilde{e},\tilde{e}'}\rangle \otimes |C_{\tilde{t},\tilde{t}'}\rangle\Big), \\ &\geq 1 - \epsilon. \end{aligned}$$

∎

Recall the implication of Theorem 5.6.1: since a counterfeiter with small success probability can be amplified to one with high success probability, it suffices to prove security against the latter. We now formally use this theorem, and the previous result on counterfeiters that succeed with high probability (Corollary 5.6.6), to provide a lower bound to the number of queries needed by a counterfeiter that has small success probability.

While the proof technique for this result is practically the same as the analogous result in [AC12], our result contains a noticeable difference. Since the space S which our verification operator projects to is not 1-dimensional, the probability of a counterfeiter's state being accepted can be notably higher in comparison to the noise-free setting, even if that state has a low fidelity with the tensor product of two tolerable noisy states. This difference marks the trade-off between security and noise-tolerance.

Corollary 5.6.7 (Lower bound for high-error counterfeiting)**.** *Let $1/\varepsilon = o(2^{n/2})$. Given one copy of $|C\rangle$, and oracle access to U_{C^*}, a counterfeiter needs $\Omega\left(\sqrt{\varepsilon}2^{n/4}\right)$ queries to prepare a state ρ such that*

$$\max_{(e,e'),(t,t')\in\mathcal{E}} \Big(\langle C_{e,e'}| \otimes \langle C_{t,t'}| \Big)\rho\Big(|C_{e,e'}\rangle \otimes |C_{t,t'}\rangle\Big) \geq \varepsilon$$

for a worst-case $|C\rangle$. Furthermore, the greatest lower bound on the probability of accepting this state is

$$\Pr\left[V^{\otimes 2}(\rho) = \textit{accepts}\right] \geq |\mathcal{E}|^2\varepsilon.$$

Proof: Suppose we have a counterfeiter $\mathscr{C}$ that makes $o\left(\sqrt{\varepsilon}2^{n/4}\right)$ queries to U_{C^*}, and prepares a state σ such that

$$\max_{(e,e'),(t,t')\in\mathcal{E}} \Big(\langle C_{e,e'}| \otimes \langle C_{t,t'}| \Big)\sigma\Big(|C_{e,e'}\rangle \otimes |C_{t,t'}\rangle \Big) \geq \varepsilon.$$
$$\Leftrightarrow F(\sigma, S^{\otimes 2})^2 \geq \varepsilon.$$

Then

$$\Pr\left[V^{\otimes 2}(\sigma) = \text{accepts}\right] = F(\sigma, S^{\otimes 2})^2 \geq \varepsilon.$$

Let δ be small, say $\delta = 0.00001$. Then by Theorem 5.6.1, there exists an amplified counterfeiter $\mathscr{C}'$ that makes

$$\mathcal{O}\left(\frac{\log 1/\delta}{\sqrt{\varepsilon}\,(\sqrt{\varepsilon}+\delta^2)}\right) = \mathcal{O}\left(\frac{1}{\sqrt{\varepsilon}}\right)$$

calls to $\mathscr{C}$ and V_C, and that prepares a state ρ such that $F(\rho, S^{\otimes 2}) \geq 1-\delta$ which implies

$$\Pr\left[V^{\otimes 2}(\rho) = \text{accepts}\right] \geq (1-\delta)^2.$$

Counting the $o\left(\sqrt{\varepsilon}2^{n/4}\right)$ queries from each use of $\mathscr{C}$, and $\mathcal{O}(1)$ queries from each use of V_C, the total number of queries that $\mathscr{C}'$ makes to U_{C^*} is

$$\left[o\left(\sqrt{\varepsilon}2^{n/4}\right) + \mathcal{O}(1)\right] \cdot \mathcal{O}\left(\frac{1}{\sqrt{\varepsilon}}\right) = o\left(2^{n/4}\right).$$

But this contradicts Corollary 5.6.6.

Now suppose a counterfeiter uses $\Omega\left(\sqrt{\varepsilon}2^{n/4}\right)$ queries to prepare a state ρ such that

$$\max_{(e,e'),(t,t')\in\mathcal{E}} \Big(\langle C_{e,e'}| \otimes \langle C_{t,t'}| \Big)\rho\Big(|C_{e,e'}\rangle \otimes |C_{t,t'}\rangle \Big) \geq \varepsilon.$$

If $\mathscr{C}$ submits this state for verification, their probability of acceptance will be higher if

$$\Big(\langle C_{e,e'}| \otimes \langle C_{t,t'}| \Big)\rho\Big(|C_{e,e'}\rangle \otimes |C_{t,t'}\rangle \Big) > 0,$$

for all $(e,e'),(t,t') \in \mathcal{E}$. The best scenario for $\mathscr{C}$ is that each term obtains the maximum value, in which case we have

$$\Pr\left[V^{\otimes 2}(\rho) = \text{accepts}\right] = \sum_{(e,e'),(t,t')\in\mathcal{E}} \langle C_{e,e'}| \otimes \langle C_{t,t'}|\,\rho\,|C_{e,e'}\rangle \otimes |C_{t,t'}\rangle\,,$$

$$= |\mathcal{E}|^2 \cdot \max_{(e,e'),(t,t')\in\mathcal{E}} \Big(\langle C_{e,e'}| \otimes \langle C_{t,t'}| \Big)\rho\Big(|C_{e,e'}\rangle \otimes |C_{t,t'}\rangle \Big),$$
$$\geq |\mathcal{E}|^2\varepsilon.$$

∎

The following theorem shows that it suffices for a bank to generate uniformly random banknotes. Indeed, as the proof shows, if a counterfeiter is able to clone a uniformly random banknote, then they can clone any banknote. Note that in the noise-free case of [AC12], the proof of this theorem used an invertible linear map f; in our case, we additionally require f to be an isometry (*i.e.*, a distance-preserving map) to ensure that the noise-tolerance is, in a sense, "preserved" under the mapping f.

Theorem 5.6.8 (Lower bound for average-case counterfeiting)**.** *Let C be a uniformly random element of $\mathcal{W}$. Then given one copy of $|C\rangle$, as well as oracle access to U_{C^*}, a counterfeiter $\mathscr{C}$ needs $\Omega(\sqrt{\varepsilon}2^{n/4})$ queries to prepare a $2n$-qubit state ρ that $V_C^{\otimes 2}$ accepts with probability at least $|\mathcal{E}|^2\varepsilon$, for all $1/\varepsilon = o(2^{n/2})$. Here the probability is taken over the choice C, as well as the behaviour of $\mathscr{C}$ and $V_C^{\otimes 2}$.*

Proof: Suppose we had a counterfeiter $\mathscr{C}$ that violated the above. Using $\mathscr{C}$ as a black box, we construct a new counterfeiter $\mathscr{C}'$ that violates Corollary 5.6.7.

Given a (deterministically-chosen) banknote $|C\rangle$ and oracle access to U_{C^*}, first choose an invertible linear isometry $f : \mathbb{F}_2^n \to \mathbb{F}_2^n$ uniformly at random. Then $f(C)$ is a uniformly random element of $\mathcal{W}$ by Lemma 5.5.2. Furthermore, the state $|C\rangle$ can be transformed into $|f(C)\rangle$ straightforwardly.

Using the oracle U_{C^*} and the map f, we can simulate the corresponding oracle for the money state $|f(C)\rangle$. Let U_f be the unitary that acts as $U_f|x\rangle = |f(x)\rangle$. Then the corresponding oracle is given by $(I^k \otimes U_f)U_{C^*}(I^k \otimes U_f^\dagger)$. To see that this works, observe that,

$$\begin{aligned} x \in C_{\mathcal{E}_X} &\Leftrightarrow x \in C + e, \text{ for some } e \in \mathcal{E}_X \\ &\Leftrightarrow f(x) \in f(C) + f(e), \text{ where } f(e) \in \mathcal{E}_X \\ &\Leftrightarrow f(x) \in f(C)_{\mathcal{E}_X} \end{aligned}$$

where $f(e) \in \mathcal{E}_X$ follows because f is an isometry and thus preserves the weight of e.

So, by using $\mathscr{C}$ for $n/2$-dimensional states that are drawn uniformly randomly from $\mathcal{W}$, the adversary $\mathscr{C}'$ can produce a state ρ_f that $V_{f(C)}^{\otimes 2}$ accepts with probability at least $|\mathcal{E}|^2\varepsilon$. Applying f^{-1} to both registers of ρ_f, $\mathscr{C}'$ can obtain a state ρ that $V_C^{\otimes 2}$ accepts with probability at least $|\mathcal{E}|^2\varepsilon$, thereby contradicting Corollary 5.6.7. ∎

We now rephrase Theorem 5.6.8 in terms of the oracle pair $(U_{C_{\mathcal{E}_X}}, U_{C_{\mathcal{E}_Z}})$ as opposed to the single oracle U_{C^*}. In the noise-free case of [AC12], this is immediate as their single oracle U_{A^*} exactly corresponds to the pair $(U_A, U_{A^\perp})$. In our case, we recall (see Section 5.6.2) that the query complexity for the pair $(U_{C_{\mathcal{E}_X}}, U_{C_{\mathcal{E}_Z}})$ is the query complexity for U_{C^*} divided by $2|\mathcal{E}_X|$. Under this correspondence, we get the following corollary of Theorem 5.6.8.

Corollary 5.6.9. *Let C be a uniformly random element of $\mathcal{W}$. Then given one copy of $|C\rangle$, as well as oracle access to $(U_{C_{\mathcal{E}_X}}, U_{C_{\mathcal{E}_Z}})$, a counterfeiter $\mathscr{C}$ needs $\Omega(\sqrt{\varepsilon}2^{n/4}/|\mathcal{E}_X|)$ queries to prepare a $2n$-qubit state ρ that $V_C^{\otimes 2}$ accepts with probability at least $|\mathcal{E}|^2\varepsilon$, for all $1/\varepsilon = o(2^{n/2})$. Here the probability is taken over the choice C, as well as the behaviour of $\mathscr{C}$ and $V_C^{\otimes 2}$.*

In the noise-free case, Corollary 5.6.9 implies that ε is $1/\mathsf{exp}(n)$, which constitutes the soundness error in this case. In the noisy case, we need $|\mathcal{E}|^2$ to be constant in n to get the same soundness error, *i.e.*, for $|\mathcal{E}|^2\varepsilon$ to be $1/\mathsf{exp}(n)$. More errors can be tolerated at the cost of increasing the soundness error, though we remark that it is not possible to tolerate too many errors. Recall from equation (3.4.9) that the size of $|\mathcal{E}_q|$ is given by

$$|\mathcal{E}_q| = |\mathcal{E}_X| \cdot |\mathcal{E}_Z| = \left(\sum_{j=0}^{q} \binom{n}{j}\right)^2.$$

When q is large (close to n), the sum in the parentheses approaches exponential, and so the soundness error approaches 1. Intuitively, this makes sense since a high noise-tolerance allows highly imperfect copies to be accepted, which is advantageous for a counterfeiter. In addition to increasing the soundness error, Corollary 5.6.9 shows that tolerating more errors has the effect of reducing the number of queries needed by an adversary. On the other hand, when $q = kn$, with $0 < k \leq 1/2$, we have the following bound from [FG06],

$$|\mathcal{E}_q| \leq 2^{2H(k)n}, \tag{5.6.16}$$

where $H(\cdot)$ is the binary Shannon entropy. Then for a small number of arbitrary errors q (*i.e.*, small k), we see that $|\mathcal{E}_q|$ stays small enough to ensure that we can get a reasonably low soundness error.

This trade-off between noise-tolerance and soundness is visualized in Figure 5.2. We see that if we fix a desirable value for the soundness error, then increasing q necessitates an increase of n to maintain the same soundness error. Given this, it is interesting to consider what an optimal choice of n would be for a fixed soundness error. If, for example, we consider a soundness error of 0.03, inspection of Figure 5.2 reveals that the ratio q/n is approximately 1.75% for $(n, q) = (57, 1)$, but 1.81% for $(n, q) = (166, 3)$, and thus increasing n can improve the noise-tolerance. For larger soundness error, it is less clear that this still holds true. We discuss this further in Section 5.7 as a direction for future research.

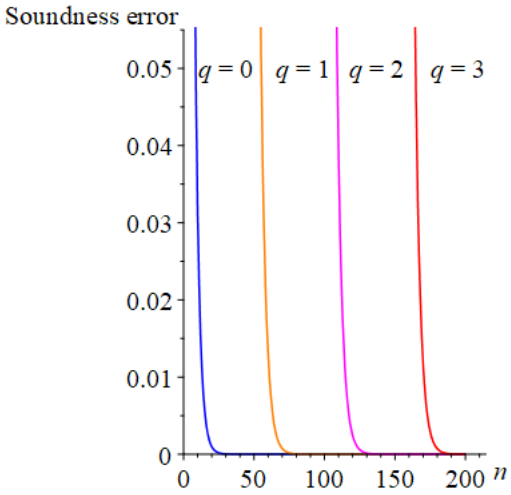

Figure 5.2: A plot of the soundness error $|\mathcal{E}_q|^2\varepsilon$ for various choices of q (the number of tolerated arbitrary errors) against the number of qubits n. Here, we take $\varepsilon = 2^{-n/2}$.

We now return to proving security for the noisy-projective mini-scheme $\mathcal{M}$ specified in Section 5.4. To do this, we use the same proof technique as the analogous result in [AC12]; the notable difference here is that we must account for the fact that the formal specification of our scheme uses conjugate coding states, though this does not affect security.

Theorem 5.6.10 (Security of the noisy-projective mini-scheme)**.** *The noisy-projective mini-scheme $\mathcal{M} = (\mathsf{Bank}_{\mathcal{M}}, \mathsf{Ver}_{\mathcal{M}})$, which is defined relative to the classical oracle U, has perfect completeness and $|\mathcal{E}|^2/\mathsf{exp}(n)$ soundness error.*

Proof: Perfect completeness follows from how $\mathcal{M}$ is defined (see Section 5.4) and from Lemma 5.5.1.

Corollary 5.6.9 almost gives us the soundness result. To formally complete the proof, we must show that if a quantum polynomial-time counterfeiter $\mathscr{C}$ is given a banknote of the form $|\$_r\rangle = |z_r\rangle\, U_{\mathcal{B}_r}\, |0\rangle_{\theta_r}$, then querying the oracles $\mathcal{G}, \mathcal{Z}, \mathcal{T}_{\text{primal}}, \mathcal{T}_{\text{dual}}$ provides no additional advantage over querying $(U_{C_{\mathcal{E}_X},r}, U_{C_{\mathcal{E}_Z},r})$, as done in Corollary 5.6.9.

The goal is to show that any successful attack on $\mathcal{M}$ can be converted into successful counterfeiting of $|C\rangle$ given oracle access to $(U_{C_{\mathcal{E}_X},r}, U_{C_{\mathcal{E}_Z},r})$. This would then contradict Corollary 5.6.9, thus proving the result. This can be achieved by observing that an adversary trying to counterfeit $|C\rangle$, given $(U_{C_{\mathcal{E}_X},r}, U_{C_{\mathcal{E}_Z},r})$, can essentially simulate the oracles $\mathcal{G}, \mathcal{Z}, \mathcal{T}_{\text{primal}}, \mathcal{T}_{\text{dual}}$.

For the oracle $\mathcal{G}$, let $r \in \{0,1\}^n$ be the random string chosen by the bank, so that $\mathcal{G}(r) = (z_r, \theta_r, \mathcal{B}_r)$. Observe that the string r remains uniformly random, even

when we condition on z_r, θ_r, and $\mathcal{B}_r$, as well as complete descriptions of $\mathcal{T}_{\text{primal}}, \mathcal{T}_{\text{dual}}$ and $\mathcal{Z}$. Even if we query $\mathcal{G}(r')$ for $r' \neq r$, no information about r is revealed because the values of $\mathcal{G}$ are generated independently. So, for the simulation of $\mathcal{G}$, suppose we instead modify it by setting $\mathcal{G}(r) \coloneqq (z'_r, \theta'_r, \mathcal{B}'_r)$ for some new $3n$-bit serial number z'_r and $\theta'_r, \mathcal{B}'_r$ (such that θ'_r has Hamming weight $n/2$) chosen uniformly at random. Then the BBBV hybrid argument [BBBV97] says that, in expectation over r, this can alter the final state output by the counterfeiter $\mathscr{C}(\$_r)$ by at most $\mathsf{poly}(n)/2^{n/2}$ in trace distance. Thus, if $\mathscr{C}$ succeeded with non-negligible probability before, then $\mathscr{C}$ must still succeed with non-negligible probability after we set $\mathcal{G}(r) \coloneqq (z'_r, \theta'_r, \mathcal{B}'_r)$.

Given this simulation of $\mathcal{G}$, an adversary can now easily simulate $\mathcal{Z}$ for the new serial number z'_r. For $\mathcal{T}_{\text{primal}}$, recall that its behaviour is $\mathcal{T}_{\text{primal}} |z\rangle |v\rangle = |z\rangle U_{C_{\mathcal{E}_X}} |v\rangle$, which can be simulated with knowledge of z and $U_{C_{\mathcal{E}_X}}$. Similarly, $\mathcal{T}_{\text{dual}}$ can be simulated with $U_{C_{\mathcal{E}_Z}}$. Note that the security guarantees we consider are query complexity bounds, thus we do not need to be concerned with the computational complexity of these simulations.

Thus, with these simulated oracles, any successful attack on $\mathcal{M}$ can indeed be converted into successful counterfeiting of $|C\rangle$ with oracle access to only $(U_{C_{\mathcal{E}_X},r}, U_{C_{\mathcal{E}_Z},r})$. This contradicts Corollary 5.6.9. ∎

Remark 5.6.11. The scheme we have presented tolerates noise but does not correct it. Consequently, a valid banknote will eventually accumulate too many errors to be accepted. It is straightforward to adapt the subset approach to correct tolerable noise. Instead of checking bit-flip errors with a single oracle $U_{C_{\mathcal{E}_X}}$, we would use a collection of oracles $\{U_{C+e}\}_{e \in \mathcal{E}_X}$ where U_{C+e} checks for membership in the coset $C + e$, and similarly for phase-flip errors. Verification would involve querying these oracles sequentially; by doing this, we can determine if the money state is in one of the cosets (in which case we accept it) and, based off which coset it belongs to, we learn which error has occurred (which allows for correction). The security proof would proceed in a similar manner. In Section 5.6.2, the construction of C^* would remain the same, but now a single query to U_{C+e} corresponds to a single query to U_{C^*}, and so the query complexity in Corollary 5.6.9 would be $\Omega(\sqrt{\varepsilon} 2^{n/4})$. The downside of this approach is that verification now involves the oracles $(\{U_{C+e}\}_{e \in \mathcal{E}_X}, \{U_{C+e}\}_{e \in \mathcal{E}_Z})$, whereas the scheme we presented for tolerating noise only involved $(U_{C_{\mathcal{E}_X}}, U_{C_{\mathcal{E}_Z}})$, and so this scheme for error correction could involve many additional queries as part of the verification algorithm.

5.7 Summary and future directions

We have developed a noise-tolerant public-key quantum money scheme and analyzed its security within the classical oracle model. Our scheme is built upon the [AC12] scheme, where we have obtained noise-tolerance by drawing connections to quantum error correction and enlarging the space that is used for verifying banknotes.

In Section 5.5, we presented two approaches to verification. Is one preferable over the other? Both approaches achieve the same completeness and soundness error, but is one easier to realize in the plain model? Towards answering this question, we note that in [Zha19], Zhandry instantiated the oracle scheme of [AC12] by using iO to construct what he called a *subspace hiding obfuscator*. This technique seems to naturally apply to our subspace approach, thus implying that it can also be realized in the plain model. Is it any more difficult to use iO to instantiate the classical oracle that checks membership in subsets?

How does adding a layer of error correction compare to our approach? In Section 5.3.2, we briefly discussed how one could take a different approach to noise-tolerance by applying a layer of error correction on top of the [AC12] oracle scheme. The version of this approach where the verification algorithm is made to operate fault-tolerantly has no known security proof, though we have identified the following obstacles: translating any necessary computations to versions that operate on encoded states, and determining if/how the original security proof changes when the adversary holds an encoded state instead of a subspace state. Given a realization of this approach, it would be interesting to make a comparison with our scheme. Notably, a layer of error correction typically requires many additional qubits (*i.e.*, large overhead), so one may ask questions like: for a fixed n, how does the noise-tolerance and soundness error compare?

What is the optimal noise-tolerance for a fixed soundness error? In the discussion that followed Corollary 5.6.9, we observed that for a fixed soundness error, the noise-tolerance q/n could be improved by increasing n. Can this observation be formalized in general? Specifically, for a fixed, arbitrary soundness error, what choice of n maximizes the quantity q/n? Answering this exactly requires analysis of the expression for $|\mathcal{E}_q|^2\varepsilon$. For low noise, the bound in equation (5.6.16) could be used to perhaps simplify the analysis.

Is it possible to obtain noise-tolerance for other public-key quantum money schemes? In Section 5.1.3, we mentioned some of the other public-key quantum money schemes that rely on hardness assumptions instead of oracles. Is it possible to make any of these schemes noise-tolerant?

Chapter 6

Finding a canonical specious adversary

In this chapter, we study the specious adversary, which is the quantum analogue of the classical semi-honest adversary. Within a restricted setting, we identify a particular specious adversary that is canonical.

6.1 Introduction

The setting of secure two-party computation can often be described in the following way: Two parties, Alice and Bob, each have their own inputs which are, respectively, given by x and y. They would like to jointly evaluate some function f on their inputs $f(x, y)$ such that Alice cannot learn anything more of Bob's input y than from what she can learn from the output of the protocol, and vice versa. In classical two-party computation, there exists a well-known adversary that is called *semi-honest*. The semi-honest adversary executes a protocol honestly to ensure that the task is completed (*i.e.*, correctness) but records everything that occurs throughout the protocol with the hope of using this information to learn something about the other party's input. Such an adversary is said to be weak, since correctness places a limitation on their possible deviations.

It is naturally interesting to study the quantum version of the classical semi-honest adversary. In the quantum setting, we have Alice and Bob with respective registers $\mathcal{A}, \mathcal{B}$ who share an initial state $\rho_{\text{in}} \in \mathcal{D}(\mathcal{A} \otimes \mathcal{B})$ and wish to privately evaluate some unitary that acts on both registers, *i.e.*, compute $\rho_{\text{out}} = U \cdot \rho_{\text{in}}$ where $U \in \mathsf{U}(\mathcal{A} \otimes \mathcal{B})$. Privacy, in this context, means that Alice should be unable to learn anything of Bob's register, and vice versa. This task is known as quantum two-party evaluation of unitaries. Defining a quantum analogue of the semi-honest adversary in this setting is not immediately obvious since the no-cloning theorem does not permit copying in general, which is one of the defining traits of the classical semi-honest

adversary. The most natural quantum analogue was defined in [DNS10] as a class of adversaries called *specious*.

Put simply, a quantum adversary is specious if, at any step of the protocol, it is capable of producing some state such that when it is combined with the state held by the honest party, the overall state is close to the case where both parties are honest. Phrased differently, a specious adversary can deviate from the protocol description but, at any step, must be able to use its current state to approximately reconstruct the joint honest state. Intuitively, this captures what we expect in a quantum analogue of the classical semi-honest adversary.

A remark on terminology: Since the specious adversary is reminiscent of the classical semi-honest adversary, one may wonder why we refrain from simply saying "quantum semi-honest." The reason is that when the formal definition of the specious adversary (Definition 6.2.3) is translated to the classical setting, it results in an adversary that is strictly stronger than the classical semi-honest adversary (see appendix B in the arXiv version of [DNS10]). Thus, the adoption of the name "semi-honest" for the quantum setting is not exactly natural.

While a specious adversary may not be able to clone states (as a consequence of the no-cloning theorem), their definition affords them a wide range of behaviour. For instance, such an adversary could defer a prescribed measurement to a later point of the protocol in an indistinguishable manner (this is the deferred measurement principle). As another example, the *purified adversary*, who is defined by purifying all of their actions, also satisfies the specious definition. The purified adversary is especially well-known and has appeared in literature long before the notion of specious was formally defined (see [BB84], where the purified adversary is viewed as using entangled states to delay input choices).

Our main motivation for studying the specious adversary stems from a line of work that was carried out in [DNS10, DNS12]. At the simplest level, this line of work showed how to transform a two-party protocol that is secure against specious adversaries into a protocol that is secure against a stronger class of adversaries. Given that specious adversaries are weak quantum adversaries, this result highlights their potential utility. If, for instance, we are developing a protocol to accomplish some task, proving its security against a strong adversary may be a daunting task. A much simpler approach might be to first prove security against a weak adversary, like the specious adversary, and to then compile the protocol in a similar manner to [DNS12] to get security against stronger adversaries.

Following this direction, one natural question to ask is how can we easily prove security against specious adversaries? The most desirable way would be to determine a specious adversary that is *canonical* in the sense that if we prove security against the canonical specious adversary, then we have proved security against the entire set

of specious adversaries. In the classical case, the dishonest behaviour of a semi-honest adversary is the act of copying information, and so it is straightforward to see that a semi-honest adversary is canonical if they copy *everything*. In the quantum case, finding a specious adversary that is canonical is less obvious, as there may be an infinite number of specious adversaries to account for, and so to simplify the analysis, our definition of canonical is relative to a finite set of specious adversaries.

6.1.1 Contributions

We formally define the notion of a canonical specious adversary for a finite set of specious adversaries. We then prove that if the finite set consists of specious adversaries that can perfectly reconstruct the honest state at each step, and if we consider a restricted set of two-party protocols (defined by only using isometries), then there exists a canonical specious adversary which we call the *greedy adversary*. Notably, the greedy adversary only partially fulfills our goal: it satisfies the definition of canonical but, as defined, it is not necessarily any easier to prove security against it over a finite set of specious adversaries.

6.1.2 Model

We study the problem of finding a canonical specious adversary within the framework of two-party protocols that [DNS10] defined when they studied the problem of quantum two-party evaluation of unitaries. In this framework, a two-party protocol for evaluating a unitary U on some input state ρ_{in} is described as a series of steps involving two parties, with the actions of each party described as a sequence of quantum channels. Additionally, such a protocol can make calls to some oracle, where the oracle calls can be viewed as a reliance on an ideal functionality to obtain privacy, or as a way to model communication between the two parties.

The *correctness* of a two-party protocol is quantified by how close, in terms of trace distance, the real state produced at the end of the protocol is to the ideal output (U applied to ρ_{in}).

The security, or *privacy*, of such a protocol is defined in terms of simulation. If a simulator with access to only the adversary's input and the ideal functionality is able to construct the adversary's state at *any* step of the protocol, then we say the protocol is private. Intuitively, if the simulator is unable to do this, then the adversary's state must rely on some additional information gathered during the protocol's execution, thereby violating privacy.

For specious adversaries, the formal definition allows us to consider different "classes" of specious adversaries which differ by how well they can reconstruct the honest state. More precisely, a ε-specious adversary is defined as one who is able to produce a state, at *any* step, such that when it is combined with the honest party's

state, the resulting joint state is ε-close in trace distance to the joint state where both parties are honest. Apart from this restriction, the specious adversary has access to an arbitrarily large quantum memory and has unbounded computational power.

In this work, we focus on the $\varepsilon = 0$ class of specious adversaries (*i.e.*, those who are able to perfectly reconstruct the honest state) and perfectly correct protocols that only use isometries (as opposed to the more general case where quantum channels may be used). These restrictions are done for the sake of obtaining a better characterization of specious behaviour (see Section 6.3 for details).

6.1.3 Methods

Intuitively, a protocol with perfect correctness requires that the ideal output is produced at the last step, thus forcing a specious adversary to execute any dishonest behaviour prior to this. This intuition was formalized in [DNS10] as the *Rushing Lemma* which, mathematically, says that the dishonest register of the specious adversary is in a tensor product with the ideal output at the end of the protocol.

In this work, the restriction to 0-specious adversaries and protocols that only use isometries allows us to prove a result that resembles the Rushing Lemma applied to *all* steps. That is, at each step of the protocol, the joint state is isometric to the specious adversary's dishonest register in tensor product with the honest joint state. This characterizes the specious behaviour within our model. To define a specious adversary that is canonical for a finite set of specious adversaries, we define an adversary who, with unbounded computational power and memory, performs "all" dishonest actions of the adversaries within the set under consideration (see Section 6.4 and Definition 6.4.6). We call this adversary the *greedy adversary*.

To prove that the greedy adversary is canonical (within the same restrictions), we suppose that a simulator for it exists and then show that this simulator can be transformed into a simulator for an arbitrary specious adversary with at least the same level of privacy. Intuitively, an arbitrary specious adversary's register can be viewed as a subregister of the dishonest register belonging to the greedy adversary (by virtue of the fact that the greedy adversary does everything). Then this transformation from one simulator to another roughly translates to tracing out everything that the arbitrary specious adversary did not execute.

6.1.4 Further background

Prior to the formal definition of specious adversaries in [DNS10], the idea of an adversary who appears honest had been utilized in various works, usually in the form of the purified adversary. As we noted earlier, there is the example of the purified adversary in [BB84]. In addition to this, the purified adversary was used in [LC97, May97] to show the impossibility of bit commitment with quantum communication,

and in [SSS09] to obtain a more general impossibility result which applies to any non-trivial two-party classical function. These impossibility results stem from the fact that if the adversary purifies their actions, information will leak. Note that in [SSS09], this purifying behaviour is referred to as *honest-but-curious*.

These impossibility results do not apply to the quantum primitive studied in [DNS10]: two-party evaluation of a unitary on a joint input state. Nevertheless, the authors found that this task is also impossible. More precisely, they found that no protocol in the *bare model* (no oracle calls to an ideal functionality) can be statistically private (see Definition 6.2.5) against specious adversaries (the swap gate is the simple counterexample). The consequence of this result is that cryptographic assumptions are needed for the private evaluation of unitaries. The authors then give a protocol for the private evaluation of any unitary by using two ideal functionalities for classical computation: one which they call an AND-box, and the other is an ideal swap.

Several years later, specious adversaries were explicitly used in [BB15], where the authors studied the task of quantum private information retrieval (QPIR) with a single server: using quantum resources for the task of retrieving information from a database in a private manner. Specifically, the authors were concerned with the amount of communication required to accomplish PIR, and if using quantum resources could reduce the communication complexity. It had been previously suggested, in [LG12], that this was the case, as the QPIR protocol in [LG12] had sublinear communication complexity. However, in [BB15], the authors showed that any QPIR protocol that is at least secure against specious adversaries has linear communication complexity. Thus, the protocol in [LG12] is not secure against specious adversaries. Later, in [ABC+19], the authors defined an adversarial model, called *anchored-specious*, that is weaker than the specious model in the sense that an adversary can act speciously but cannot perform superposition attacks; the authors showed that the protocol in [LG12] is secure within this weaker model.

6.1.5 Outline

We start in Section 6.2 by recalling, from [DNS10], the formal definition of a two-party protocol, the definition of specious adversaries, and how privacy is defined in terms of simulators. In Section 6.3, we formally define what it means for a specious adversary to be canonical; then, within a restricted setting, we characterize the behaviour of 0-specious adversaries. In Section 6.4, we define a 0-specious adversary, which we call the greedy adversary, and in Section 6.5, we prove that this adversary indeed satisfies our definition of canonical. We conclude in Section 6.6 by outlining a few questions for future research.

6.2 Definitions

6.2.1 Formalizing two-party protocols

In this section, we recall the definitions from [DNS10] that are needed to formally analyze two-party protocols for the evaluation of unitaries. We start with Definition 6.2.1, which defines a two-party strategy $\Pi^{\mathscr{O}}$ as a sequence of quantum operations for each party that may include oracle calls $\mathscr{O}$ (which are also modelled as quantum operations). Then, Definition 6.2.2 defines a two-party protocol for the evaluation of a unitary $\Pi_U^{\mathscr{O}}$ as a two-party strategy for some specified unitary U.

Definition 6.2.1 ([DNS10])**.** *An n-step **two-party strategy** with oracle calls, denoted as $\Pi^{\mathscr{O}} = (\mathscr{A}, \mathscr{B}, \mathscr{O}, n)$, consists of*

1. *Input spaces $\mathcal{A}_0$ and $\mathcal{B}_0$ for parties $\mathscr{A}$ and $\mathscr{B}$, respectively.*
2. *Memory spaces $\mathcal{A}_1, \ldots, \mathcal{A}_n$ for $\mathscr{A}$ and $\mathcal{B}_1, \ldots, \mathcal{B}_n$ for $\mathscr{B}$. These memory spaces can be written as $\mathcal{A}_i = \mathcal{A}_i^{\mathscr{O}} \otimes \mathcal{A}_i'$ and $\mathcal{B}_i = \mathcal{B}_i^{\mathscr{O}} \otimes \mathcal{B}_i'$ for $1 \leq i \leq n$.*
3. *An n-tuple of quantum operations $(\mathscr{A}_1, \ldots, \mathscr{A}_n)$ for $\mathscr{A}$, where*
$$\mathscr{A}_i : \mathcal{L}(\mathcal{A}_{i-1}) \mapsto \mathcal{L}(\mathcal{A}_i), \tag{6.2.1}$$
with $1 \leq i \leq n$.
4. *An n-tuple of quantum operations $(\mathscr{B}_1, \ldots, \mathscr{B}_n)$ for $\mathscr{B}$ where*
$$\mathscr{B}_i : \mathcal{L}(\mathcal{B}_{i-1}) \mapsto \mathcal{L}(\mathcal{B}_i), \tag{6.2.2}$$
with $1 \leq i \leq n$.
5. *An n-tuple of quantum operations $\mathscr{O} = (\mathscr{O}_1, \ldots, \mathscr{O}_n)$ where*
$$\mathscr{O}_i : \mathcal{L}(\mathcal{A}_i^{\mathscr{O}} \otimes \mathcal{B}_i^{\mathscr{O}}) \mapsto \mathcal{L}(\mathcal{A}_i^{\mathscr{O}} \otimes \mathcal{B}_i^{\mathscr{O}}), \tag{6.2.3}$$
with $1 \leq i \leq n$.

If $\Pi^{\mathscr{O}} = (\mathscr{A}, \mathscr{B}, \mathscr{O}, n)$ is an n-turn two-party strategy, then the final state of the interaction upon input state $\rho_{in} \in \mathcal{D}(\mathcal{A}_0 \otimes \mathcal{B}_0 \otimes \mathcal{R})$, where $\mathcal{R}$ is a system of dimension $\dim \mathcal{R} = \dim \mathcal{A}_0 \dim \mathcal{B}_0$, is:

$$\begin{aligned}[\mathscr{A} \circledast \mathscr{B}](\rho_{in}) \coloneqq &\left(\mathbb{1}_{\mathcal{L}(\mathcal{A}_n' \otimes \mathcal{B}_n' \otimes \mathcal{R})} \otimes \mathscr{O}_n\right)(\mathscr{A}_n \otimes \mathscr{B}_n \otimes \mathbb{1}_{\mathcal{R}}) \\ &\cdots \left(\mathbb{1}_{\mathcal{L}(\mathcal{A}_1' \otimes \mathcal{B}_1' \otimes \mathcal{R})} \otimes \mathscr{O}_1\right)(\mathscr{A}_1 \otimes \mathscr{B}_1 \otimes \mathbb{1}_{\mathcal{R}})(\rho_{in}).\end{aligned}$$

Note that the inclusion of a reference system $\mathcal{R}$ in Definition 6.2.1 allows us to consider not only pure input states but also states that are entangled with some other system.

Definition 6.2.1 allows us to model communication between two parties as a *communication oracle*. As an example, consider the case of Alice sending a state to Bob. Take $\mathcal{A}_i^{\mathscr{O}} \approx \mathcal{B}_i^{\mathscr{O}}$ and then let $\mathscr{O}_i$ move the state stored in $\mathcal{A}_i^{\mathscr{O}}$ to $\mathcal{B}_i^{\mathscr{O}}$, and erase $\mathcal{A}_i^{\mathscr{O}}$. The *bare model* is defined as a protocol which only uses communication oracles.

For the following definition, recall that $\mathsf{U}(\mathcal{H})$ denotes the set of unitaries on the Hilbert space $\mathcal{H}$.

Definition 6.2.2 ([DNS10])**.** *A* ***two-party protocol*** $\Pi_U^{\mathscr{O}} = (\mathscr{A}, \mathscr{B}, \mathscr{O}, n)$ *for a unitary* $U \in \mathsf{U}(\mathcal{A}_0 \otimes \mathcal{B}_0)$ *is an* n*-step two-party strategy with oracle calls, where* $\mathcal{A}_n \approx \mathcal{A}_0$ *and* $\mathcal{B}_n \approx \mathcal{B}_0$*. It is said to be* ε***-correct*** *if*

$$\Delta\left([\mathscr{A} \circledast \mathscr{B}](\rho_{in}), (U \otimes \mathbb{1}_{\mathcal{R}}) \cdot \rho_{in}\right) \leq \varepsilon, \tag{6.2.4}$$

for all $\rho_{in} \in \mathcal{D}(\mathcal{A}_0 \otimes \mathcal{B}_0 \otimes \mathcal{R})$*.*

A two-party protocol in the bare model is denoted as Π_U *and, without loss of generality, we assume that* $\mathscr{O}_{2i+1}$ $(0 \leq i \leq \lfloor \frac{n}{2} \rfloor)$ *implements a communication channel from* $\mathscr{A}$ *to* $\mathscr{B}$*, and* $\mathscr{O}_{2i}$ $(1 \leq i \leq \lfloor \frac{n}{2} \rfloor)$ *implements a communication channel from* $\mathscr{B}$ *to* $\mathscr{A}$*. Communication oracles are said to be* ***trivial****.*

Let us now establish the following notation:

$$\begin{aligned}
\rho_1(\rho_{\text{in}}) &\coloneqq \left(\mathbb{1}_{\mathcal{L}(\mathcal{A}_1' \otimes \mathcal{B}_1' \otimes \mathcal{R})} \otimes \mathscr{O}_1\right)(\mathscr{A}_1 \otimes \mathscr{B}_1 \otimes \mathbb{1}_{\mathcal{R}})(\rho_{\text{in}}), \\
\rho_{i+1}(\rho_{\text{in}}) &\coloneqq \left(\mathbb{1}_{\mathcal{L}(\mathcal{A}_{i+1}' \otimes \mathcal{B}_{i+1}' \otimes \mathcal{R})} \otimes \mathscr{O}_{i+1}\right)(\mathscr{A}_{i+1} \otimes \mathscr{B}_{i+1} \otimes \mathbb{1}_{\mathcal{R}})(\rho_i(\rho_{\text{in}})),
\end{aligned} \tag{6.2.5}$$

for $1 \leq i < n$. The final state of the protocol $\Pi_U^{\mathscr{O}}$ with input state ρ_{in} is denoted as $\rho_n(\rho_{\text{in}}) = [\mathscr{A} \circledast \mathscr{B}](\rho_{\text{in}})$.

6.2.2 Specious adversaries

Specious adversaries are reminiscent of the semi-honest adversary. Informally, their behaviour is indistinguishable from the honest behaviour of the protocol. This means that at any step of the protocol, we can "audit" the adversary by requiring them to provide a state such that, when combined with the honest party's state, the joint state is close to the case where both parties are honest. The requirement of passing an audit at any given step can be seen as a restriction on how malicious the adversary can behave, and thus specious adversaries form a weak class of adversaries (though we note that no restriction is placed on their computational power and the size of their quantum memory).

We now review the formal definition of a specious adversary. We denote an adversary in the protocol $\Pi_U^{\mathscr{O}}$ by $\tilde{\mathscr{A}}$ or $\tilde{\mathscr{B}}$. When the adversary $\tilde{\mathscr{A}}$ is involved in the protocol

$\Pi_U^{\mathscr{O}}$, we denote the n-step two-party strategy as $\Pi_U^{\mathscr{O}}(\tilde{\mathscr{A}}) = (\tilde{\mathscr{A}}, \mathscr{B}, \mathscr{O}, n)$ and we denote the overall state at step i as $\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{\text{in}})$, which is defined as in equation (6.2.5), and likewise for $\tilde{\mathscr{B}}$.

Definition 6.2.3 (Specious adversary [DNS10])**.** *Let $\Pi_U^{\mathscr{O}} = (\mathscr{A}, \mathscr{B}, \mathscr{O}, n)$ be an n-step two-party protocol with oracle calls for $U \in \mathsf{U}(\mathcal{A}_0 \otimes \mathcal{B}_0)$. We say that*

- *$\tilde{\mathscr{A}}$ is ε-**specious** if $\Pi_U^{\mathscr{O}}(\tilde{\mathscr{A}}) = (\tilde{\mathscr{A}}, \mathscr{B}, \mathscr{O}, n)$ is an n-step two-party strategy with $\tilde{\mathcal{A}}_0 = \mathcal{A}_0$ and there exists a sequence of quantum operations $(\mathscr{F}_1, \ldots, \mathscr{F}_n)$ such that:*
  1. *For every $1 \leq i \leq n$, $\mathscr{F}_i : \mathcal{L}(\tilde{\mathcal{A}}_i) \mapsto \mathcal{L}(\mathcal{A}_i)$.*
  2. *For every input state $\rho_{in} \in \mathcal{D}(\mathcal{A}_0 \otimes \mathcal{B}_0 \otimes \mathcal{R})$, and for all $1 \leq i \leq n$,*

$$\Delta\left(\left(\mathscr{F}_i \otimes \mathbb{1}_{\mathcal{L}(\mathcal{B}_i \otimes \mathcal{R})}\right)\left(\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{in})\right), \rho_i(\rho_{in})\right) \leq \varepsilon. \tag{6.2.6}$$

- *$\tilde{\mathscr{B}}$ is ε-**specious** if $\Pi_U^{\mathscr{O}}(\tilde{\mathscr{B}}) = (\mathscr{A}, \tilde{\mathscr{B}}, \mathscr{O}, n)$ is an n-step two-party strategy with $\tilde{\mathcal{B}}_0 = \mathcal{B}_0$ and there exists a sequence of quantum operations $(\mathscr{F}_1, \ldots, \mathscr{F}_n)$ defined as before with $\mathcal{B}_i, \tilde{\mathcal{B}}_i$ and $\tilde{\rho}_i(\tilde{\mathcal{B}}, \rho_{in})$ replacing $\mathcal{A}_i, \tilde{\mathcal{A}}_i$, and $\tilde{\rho}_i(\tilde{\mathcal{A}}_i, \rho_{in})$ respectively.*

*If a party is $\varepsilon(m)$-specious with $\varepsilon(m)$ negligible for a security parameter m, then we say that this party is **statistically specious**.*

Note that the sequence of quantum operations $(\mathscr{F}_1, \ldots, \mathscr{F}_n)$ model the transformation that a specious adversary may need to do to recover a state that is close to the honest state. See Figure 6.1 for a simple visualization of a specious adversary. Additionally, observe that the above definition allows for various types of specious adversaries: with $\varepsilon = 0$, we get specious adversaries that can perfectly recover the honest behaviour, while with $\varepsilon > 0$ we get specious adversaries that are ε-close to the honest behaviour.

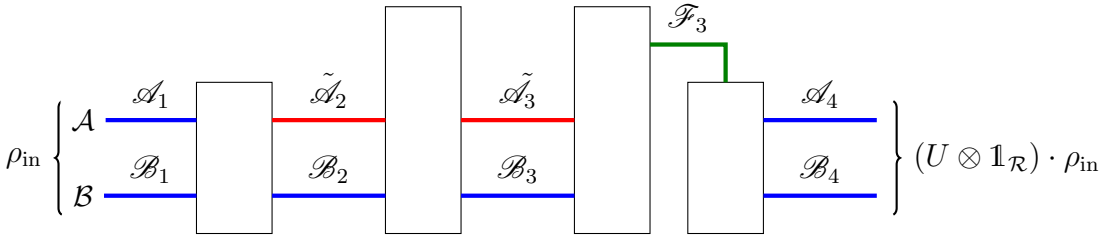

Figure 6.1: A two-party protocol with a specious adversary $\tilde{\mathscr{A}}$ for a unitary U. At the second and third step, $\tilde{\mathscr{A}}$ executes some dishonest behaviour. After the third step, an audit is called which requires the adversary to apply their recovery operation $\mathscr{F}_3$ to obtain the honest state.

6.2.3 Privacy

We now review how privacy for the protocol $\Pi_U^{\mathscr{O}}$ is defined. Consider a simulator which only has access to an adversary's input and the ideal functionality U. Informally, if the simulator is able to construct the adversary's state at any step of the protocol, then we say that the protocol is private. This concept is formalized by considering a sequence of quantum operations $(\mathscr{S}_1, \ldots, \mathscr{S}_n)$ where $\mathscr{S}_i$ constructs the adversary's view after step i. The simulator can use the adversary's input or the ideal functionality U (applied to the joint input state) to construct the adversary's view, though, by the no-cloning theorem, the simulator cannot use both (a call to the ideal functionality results in the expenditure of the input state); to control whether or not $\mathscr{S}_i$ calls U, a bit $q_i \in \{0,1\}$ is used ($q_i = 1$ means that $\mathscr{S}_i$ calls U while $q_i = 0$ means it does not).

Definition 6.2.4 (Simulator [DNS10])**.** *Let $\Pi_U^{\mathscr{O}} = (\mathscr{A}, \mathscr{B}, \mathscr{O}, n)$ be an n-step two-party protocol for $U \in \mathsf{U}(\mathcal{A}_0 \otimes \mathcal{B}_0)$. Then,*

- *$\mathscr{S}(\tilde{\mathscr{A}}) = \langle (\mathscr{S}_1, \ldots, \mathscr{S}_n), q \rangle$ is a **simulator** for adversary $\tilde{\mathscr{A}}$ in $\Pi_U^{\mathscr{O}}$ if it consists of:*
  1. *A sequence of quantum operations $(\mathscr{S}_1, \ldots, \mathscr{S}_n)$ where for $1 \leq i \leq n$, $\mathscr{S}_i : \mathcal{L}(\mathcal{A}_0) \mapsto \mathcal{L}(\tilde{\mathcal{A}}_i)$.*
  2. *A sequence of bits $q \in \{0,1\}^n$ determining if the simulator calls the ideal functionality at step i: $q_i = 1$ if and only if the simulator calls the ideal functionality.*
- *Similarly, $\mathscr{S}(\tilde{\mathscr{B}}) = \langle (\mathscr{S}_1, \ldots, \mathscr{S}_n), q' \rangle$ is a simulator for adversary $\tilde{\mathscr{B}}$ in $\Pi_U^{\mathscr{O}}$ if it satisfies the same conditions above with $q', \mathcal{B}_0, \mathcal{B}_i$, and $\tilde{\mathcal{B}}_i$ replacing $q, \mathcal{A}_0, \mathcal{A}_i$, and $\tilde{\mathcal{A}}_i$ respectively.*

For the next definition, we establish some additional notation. Given an input state $\rho_{\text{in}} \in \mathcal{D}(\mathcal{A}_0 \otimes \mathcal{B}_0 \otimes \mathcal{R})$, the simulated view of the adversaries $\tilde{\mathscr{A}}$ and $\tilde{\mathscr{B}}$ is, respectively, defined as,

$$\nu_i(\tilde{\mathscr{A}}, \rho_{\text{in}}) \coloneqq \text{tr}_{\mathcal{B}_0}\left(\left(\mathscr{S}_i \otimes \mathbb{1}_{\mathcal{L}(\mathcal{B}_0 \otimes \mathcal{R})}\right)\left((U^{q_i} \otimes \mathbb{1}_{\mathcal{R}}) \cdot \rho_{\text{in}}\right)\right),$$
$$\nu_i(\tilde{\mathscr{B}}, \rho_{\text{in}}) \coloneqq \text{tr}_{\mathcal{A}_0}\left(\left(\mathbb{1}_{\mathcal{L}(\mathcal{A}_0 \otimes \mathcal{R})} \otimes \mathscr{S}_i\right)\left(\left(U^{q_i'} \otimes \mathbb{1}_{\mathcal{R}}\right) \cdot \rho_{\text{in}}\right)\right).$$

Definition 6.2.5 (Privacy [DNS10])**.** *Let $\Pi_U^{\mathscr{O}} = (\mathscr{A}, \mathscr{B}, \mathscr{O}, n)$ be a two-party protocol for $U \in \mathsf{U}(\mathcal{A}_0 \otimes \mathcal{B}_0)$ and let $0 \leq \delta \leq 1$. We say that $\Pi_U^{\mathscr{O}}$ is δ-**private against** ε-**specious** $\tilde{\mathscr{A}}$ if there exists a simulator $\mathscr{S}(\tilde{\mathscr{A}})$ such that for all input states $\rho_{in} \in \mathcal{D}(\mathcal{A}_0 \otimes \mathcal{B}_0 \otimes \mathcal{R})$ and for all $1 \leq i \leq n$, it holds that*

$$\Delta\left(\nu_i(\tilde{\mathscr{A}}, \rho_{in}), \text{tr}_{\mathcal{B}_i}(\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{in}))\right) \leq \delta. \tag{6.2.7}$$

Similarly, we say that $\Pi_U^{\mathscr{O}}$ *is* δ*-private against* ε*-specious* $\tilde{\mathscr{B}}$ *if there exists a simulator* $\mathscr{S}(\tilde{\mathscr{B}})$ *such that for all input states* $\rho_{in} \in \mathcal{D}(\mathcal{A}_0 \otimes \mathcal{B}_0 \otimes \mathcal{R})$ *and for all* $1 \leq i \leq n$*,*

$$\Delta\left(\nu_i(\tilde{\mathscr{B}}, \rho_{in}), \mathrm{tr}_{\mathcal{A}_i}(\tilde{\rho}_i(\tilde{\mathscr{B}}, \rho_{in}))\right) \leq \delta. \tag{6.2.8}$$

Protocol $\Pi_U^{\mathscr{O}}$ *is* δ*-private against* ε*-specious adversaries if it is* δ*-private against both* $\bar{\mathscr{A}}$ *and* $\tilde{\mathscr{B}}$*. For* $\gamma > 0$*, if* $\Pi_U^{\mathscr{O}}$ *is* $2^{-\gamma m}$*-private for a security parameter* $m \in \mathbb{N}^+$*, then we say that* $\Pi_U^{\mathscr{O}}$ *is* ***statistically private****.*

6.3 Characterizing 0-specious adversaries

While specious adversaries are, by definition, restricted in how malicious they can act, there are still many different actions within this restricted space that are available to a specious adversary. For instance, consider the class of 0-specious adversaries. The most notable example in this class is the *purified adversary* $\bar{\mathscr{A}}$ who purifies their actions. More precisely, $\bar{\mathscr{A}}$ is defined by a sequence of unitaries $(\bar{\mathscr{A}}_1, \ldots, \bar{\mathscr{A}}_n)$, where $\bar{\mathscr{A}}_i : \mathcal{L}(\mathcal{A}_{i-1} \otimes \bar{\mathcal{A}}_{i-1}) \to \mathcal{L}(\mathcal{A}_i \otimes \bar{\mathcal{A}}_i)$; the space $\bar{\mathcal{A}}_0$ is of sufficiently large dimension, initialized to the zero state, and the remaining spaces $\bar{\mathcal{A}}_1, \ldots, \bar{\mathcal{A}}_n$ are purifying spaces which, if traced out, turns the overall state back into the honest state. That is, $\mathcal{F}_i = \mathrm{tr}_{\bar{\mathcal{A}}_i}(\cdot)$.

We can obtain another example by considering an adversary who utilizes the deferred measurement principle (see [NC10] for details): in certain protocols, it may be possible for an adversary to defer a measurement, that was specified by the protocol, to a later point in such a way that the situation is indistinguishable from the honest case.

We can also consider the adversary that applies some unitary, not specified by the protocol, to their register at some step i. Then the honest state can be recovered by simply applying the inverse of the unitary (if the application of the inverse is delayed to a later step, the adversary must ensure that the inverse commutes with any behaviour that occurred between). Immediately, we see that by taking various combinations of different unitaries, we can define a vast number of 0-specious adversaries through this behaviour.

Overall, we observe that with such a large set of specious actions, we can define an even larger set of specious adversaries by considering various subsets of such actions. Following this observation, a natural question to ask is whether there exists a canonical specious adversary. Our notion of canonical is relative to a finite set of specious adversaries, in the sense that proving privacy against the canonical adversary would automatically yield privacy against all specious adversaries within the set. Notably, the requirement that the set be finite means that not all specious adversaries are accounted for; we discuss this point further as a future direction in Section 6.6.

Formally, we require that a simulator for a canonical specious adversary can be transformed into a simulator for an arbitrary specious adversary within the set in such a way that we get, at a minimum, the same level of privacy. In other words, the transformation should not degrade the initial amount of privacy.

Definition 6.3.1 (Canonical specious adversary)**.** *Let $\Pi_U^{\mathscr{O}} = (\mathscr{A}, \mathscr{B}, \mathscr{O}, n)$ be a two-party protocol for $U \in \mathsf{U}(\mathcal{A}_0 \otimes \mathcal{B}_0)$. Let $\hat{\mathscr{A}}$ be a ε-specious adversary such that $\Pi_U^{\mathscr{O}}$ is δ-private against $\hat{\mathscr{A}}$ for some $0 \leq \delta \leq 1$, meaning there exists a simulator $\mathscr{S}(\hat{\mathscr{A}}) = (\hat{\mathscr{S}}_1, \ldots, \hat{\mathscr{S}}_n)$ which satisfies Definition 6.2.5. Let $\mathcal{S}^\varepsilon$ be a finite set of arbitrary ε-specious adversaries. If, for each $\tilde{\mathscr{A}} \in \mathcal{S}^\varepsilon$, $1 \leq i \leq n$, and input state $\rho_{in} \in \mathcal{D}(\mathcal{A}_0 \otimes \mathcal{B}_0 \otimes \mathcal{R})$, there exists a map $\mathcal{M}_i : \mathcal{L}(\hat{\mathcal{A}}_i) \to \mathcal{L}(\tilde{\mathcal{A}}_i)$ such that the state*

$$\mathcal{M}_i \cdot \nu_i(\hat{\mathscr{A}}, \rho_{in}) \equiv \mathrm{tr}_{\mathcal{B}_0}\left(\left(\mathcal{M}_i \circ \hat{\mathscr{S}}_i \otimes \mathbb{1}_{\mathcal{B}_0 \otimes \mathcal{R}}\right)\left((U^{q_i} \otimes \mathbb{1}_{\mathcal{R}}) \cdot \rho_{in}\right)\right), \tag{6.3.1}$$

satisfies

$$\Delta\left(\mathcal{M}_i \cdot \nu_i(\hat{\mathscr{A}}, \rho_{in}), \mathrm{tr}_{\mathcal{B}_i}(\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{in}))\right) \leq \delta, \tag{6.3.2}$$

*then we say that $\hat{\mathscr{A}}$ is a $\mathcal{S}^\varepsilon$-**canonical** ε-**specious adversary**.*

We now turn to the task of finding a canonical specious adversary. For simplicity, we focus on the class of 0-specious adversaries, as they have the most restricted behaviour out of all specious adversaries, thereby making it easier to characterize them.

Intuitively, we expect that to satisfy the definition of canonical, an adversary must be sufficiently "powerful." By this logic, a natural candidate would be the purified adversary $\bar{\mathscr{A}}$. While it is easy enough to see that the purified behaviour of $\bar{\mathscr{A}}$ could be mapped to many other specious adversaries, there is some specious behaviour which draws the existence of this mapping into question, as illustrated by the following toy example.

Example 6.3.2.

1. *Bob chooses a bit $\theta \in \{0, 1\}$ uniformly at random. If $\theta = 0$, he randomly selects a qubit from the set $\{|0\rangle, |1\rangle\}$ to send to Alice. Similarly, if $\theta = 1$, he uses the set $\{|+\rangle, |-\rangle\}$. Let x be the bit encoded in the selected qubit.*

   From Alice's perspective, she holds the state

   $$\rho_{\mathcal{A}} = \frac{1}{2}|0\rangle\langle 0|_{\mathcal{A}} + \frac{1}{2}|1\rangle\langle 1|_{\mathcal{A}},$$

   or

   $$\rho_{\mathcal{A}} = \frac{1}{2}|+\rangle\langle +|_{\mathcal{A}} + \frac{1}{2}|-\rangle\langle -|_{\mathcal{A}}.$$

   These states are identical and so there is nothing a specious Alice can do to determine the bit θ at this step.

2. *Bob then sends θ to Alice.*

   With θ (say $\theta = 0$), Alice now knows that she has

   $$|x\rangle\langle x|_{\mathcal{A}},$$

   where $x \in \{0, 1\}$. At this point, there is nothing more a purified adversary would do. But, with knowledge of θ, a different specious adversary $\tilde{\mathscr{A}}$ could clone their register without being detected,

   $$\mathsf{CNOT}(|x\rangle_{\mathcal{A}} |0\rangle_{\tilde{\mathcal{A}}}) = |x\rangle_{\mathcal{A}} |x\rangle_{\tilde{\mathcal{A}}}.$$

While the purified adversary $\bar{\mathscr{A}}$ would not clone the state in Example 6.3.2, it is clear that $\bar{\mathscr{A}}$'s view could still be mapped to the view of the adversary $\tilde{\mathscr{A}}$ who does clone. The difficulty lies in more complex protocols, where the ability to perform some specious behaviour $\tilde{\mathscr{A}}_{i+k}$ *depends* on previous specious behaviour $\tilde{\mathscr{A}}_i$, such as the cloning of a state. It is conceivable that a protocol could be designed such that if $\tilde{\mathscr{A}}_i$ was not performed, then it is not possible to do it at a later time, thereby ruling out the ability to perform $\tilde{\mathscr{A}}_{i+k}$. As an example, consider the following extension of Example 6.3.2.

Example 6.3.3.

1. *Consider a modified version of Example 6.3.2 where Alice must return $|x\rangle\langle x|$ at the end. Bob executes this modified version k times until the the string $(x_1, \ldots, x_k)$ represents a classical description of some unitary U which has $+1, -1$ eigenvalues. Bob then prepares one of the two eigenvectors $\{|\psi^0\rangle, |\psi^1\rangle\}$, uniformly at random, and sends it to Alice.*

   Alice's perspective of the state is

   $$\rho_{\mathcal{A}} = \frac{1}{2} |\psi^0\rangle\langle\psi^0| + \frac{1}{2} |\psi^1\rangle\langle\psi^1|.$$

   Observe that the purified adversary $\bar{\mathscr{A}}$ can do nothing at this stage, since they have not retained any of the x_i values. On the other hand, the adversary $\tilde{\mathscr{A}}$, who knows $(x_1, \ldots, x_k)$ and hence U, can learn which eigenvector they hold and produce a copy of it.

   $$\begin{aligned} |0\rangle_{\tilde{\mathcal{A}}} |\psi\rangle_{\mathcal{A}} &\mapsto H |0\rangle_{\tilde{\mathcal{A}}} |\psi\rangle_{\mathcal{A}}, \\ &\equiv |+\rangle_{\tilde{\mathcal{A}}} |\psi\rangle_{\mathcal{A}}, \\ &\mapsto \mathsf{CU}(|+\rangle_{\tilde{\mathcal{A}}}, |\psi\rangle_{\mathcal{A}}), \end{aligned}$$

$$
\begin{aligned}
&\equiv \frac{1}{\sqrt{2}} |0\rangle_{\tilde{\mathcal{A}}} |\psi\rangle_{\mathcal{A}} + \frac{1}{\sqrt{2}} |1\rangle_{\tilde{\mathcal{A}}} U |\psi\rangle_{\mathcal{A}} , \\
&\equiv \frac{1}{\sqrt{2}} |0\rangle_{\tilde{\mathcal{A}}} |\psi\rangle_{\mathcal{A}} + \frac{1}{\sqrt{2}} |1\rangle_{\tilde{\mathcal{A}}} (-1)^b |\psi\rangle_{\mathcal{A}} , \\
&\equiv \begin{cases} |+\rangle_{\tilde{\mathcal{A}}} |\psi^0\rangle_{\mathcal{A}} , & \text{if } b = 0, \\ |-\rangle_{\tilde{\mathcal{A}}} |\psi^1\rangle_{\mathcal{A}} , & \text{if } b = 1. \end{cases}
\end{aligned}
$$

So, by measuring the $\tilde{\mathcal{A}}$ register in the Hadamard basis, $\tilde{\mathscr{A}}$ can learn which eigenvector she was given without actually disturbing the state. Supposing she has the $+1$ eigenvector ($b = 0$), her post-measurement state is $|+\rangle_{\tilde{\mathcal{A}}} |\psi^0\rangle_{\mathcal{A}}$. Knowing this, she can essentially clone her $\mathcal{A}$ register by preparing a second copy of it:

$$(|+\rangle |\psi\rangle)_{\tilde{\mathcal{A}}} \otimes |\psi\rangle_{\mathcal{A}} .$$

Example 6.3.3 seems to indicate that the purified adversary's view cannot be mapped to the view of $\tilde{\mathscr{A}}$. The purified adversary returned the states $|x_i\rangle\langle x_i|$ without cloning them, thereby forfeiting their ability to learn which eigenvector they received; thus, their view cannot be mapped to the view of the adversary $\tilde{\mathscr{A}}$ who *does* know which eigenvector was sent.

There is, however, a slight ambiguity in this argument. Since $\bar{\mathscr{A}}$ purifies their actions, we might ask: what is the purification of sending a qubit from specious Alice to honest Bob? In the context of our toy examples, can we view the communication of a classical state in such a way that the purifying register contains information on the classical state being sent? If so, we could possibly argue that $\bar{\mathscr{A}}$ has inadvertently cloned the classical state through the act of purifying the communication channel.

On the other hand, this ambiguity is quickly resolved if we recall that communication between the two parties is modelled by a communication oracle $\mathscr{O}$ (recall Definition 6.2.1). The purified adversary has no control over this oracle and hence cannot purify it. With this perspective, the above toy examples indeed rule out the purified adversary as a canonical specious adversary.

As a result of this ambiguity, we will consider a different 0-specious adversary as our candidate for being canonical. Before defining this specious adversary (in Section 6.4), we start by generally characterizing the behaviour of 0-specious adversaries. To do this, we recall the Rushing Lemma from [DNS10]. Informally, the Rushing Lemma states that for a correct protocol and a ε-specious adversary $\tilde{\mathscr{A}}$, the overall state at the end of the protocol $[\tilde{\mathscr{A}} \circledast \mathscr{B}](\rho_{\text{in}})$ is a tensor product of the ideal output $(U \otimes \mathbb{1}_{\mathcal{R}}) \cdot \rho_{\text{in}}$ with some state in the adversary's dishonest register. Intuitively, correctness of the protocol constrains the adversary by requiring them to produce the ideal output at the end of the protocol, and thus $\tilde{\mathscr{A}}$ must "rush" to break privacy before the final step.

We develop a variation of the Rushing Lemma for the class of 0-specious adversaries. Notably, this variation (Lemma 6.3.4) is restricted to protocols that only use isometries; we denote such protocols by $\bar{\Pi}_U^{\mathscr{O}}$. The result we obtain is that at *each step* of such a protocol, a 0-specious adversary's dishonest state is in a tensor product with the honest joint state (up to some isometry).

While the proof of this variation largely uses the same techniques as the Rushing Lemma, there are some steps that require careful attention. Most notably, the original proof uses the following crucial fact: for a pure input state ρ_{in}, the ideal state $(U \otimes \mathbb{1}_{\mathcal{R}}) \cdot \rho_{\text{in}}$ is also pure, and so this state in an enlarged system takes a tensor product form (recall Proposition 3.2.4). For an arbitrary two-party protocol $\Pi_U^{\mathscr{O}}$ that starts with a pure input state ρ_{in}, it is not true in general that the state remains pure throughout the protocol, meaning the state $\rho_i(\rho_{\text{in}})$ does not necessarily take a tensor product form in an enlarged system. The only way to guarantee that $\rho_i(\rho_{\text{in}})$ remains pure throughout the protocol is to require that the protocol only uses isometries in its honest execution, hence the restriction to protocols of the type $\bar{\Pi}_U^{\mathscr{O}}$.

While Lemma 6.3.4 is stated for specious $\tilde{\mathscr{A}}$, the statement for specious $\tilde{\mathscr{B}}$ is symmetrical, as is the proof.

Lemma 6.3.4. *Let $\bar{\Pi}_U^{\mathscr{O}}$ be a correct two-party protocol for $U \in \mathsf{U}(\mathcal{A}_0 \otimes \mathcal{B}_0)$. Consider any 0-specious adversary $\tilde{\mathscr{A}}$. Then at each step i of the protocol, there exists an isometry $T_i : \tilde{\mathcal{A}}_i \to \mathcal{A}_i \otimes \tilde{\mathcal{A}}_i^*$ and a mixed state $\tilde{\gamma}_i \in \mathcal{D}(\tilde{\mathcal{A}}_i^*)$ such that for all joint input states $\rho_{in} \in \mathcal{D}(\mathcal{A}_0 \otimes \mathcal{B}_0 \otimes \mathcal{R})$,*

$$\Delta((T_i \otimes \mathbb{1}_{\mathcal{B}_i \otimes \mathcal{R}}) \cdot (\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{in})), \tilde{\gamma}_i \otimes \rho_i(\rho_{in})) = 0. \tag{6.3.3}$$

Thus, for every $1 \leq i \leq n$, the map $\mathscr{F}_i : \mathcal{L}(\tilde{\mathcal{A}}_i) \to \mathcal{L}(\mathcal{A}_i)$ is of the form

$$\mathscr{F}_i\left(\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{in})\right) = \text{tr}_{\tilde{\mathcal{A}}_i^*}((T_i \otimes \mathbb{1}_{\mathcal{B}_i \otimes \mathcal{R}}) \cdot \tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{in})). \tag{6.3.4}$$

Proof: To start, consider any pair of pure input states $|\psi_1\rangle$ and $|\psi_2\rangle$ in $\mathcal{A}_0 \otimes \mathcal{B}_0 \otimes \mathcal{R}$. Let $\mathcal{R}' \coloneqq \mathcal{R} \otimes \mathcal{R}_2$ where $\mathcal{R}_2 = \text{span}\{|1\rangle, |2\rangle\}$ represents a single qubit, and define the state

$$|\psi\rangle \coloneqq \frac{1}{\sqrt{2}}(|\psi_1\rangle |1\rangle + |\psi_2\rangle |2\rangle) \in \mathcal{A}_0 \otimes \mathcal{B}_0 \otimes \mathcal{R}'. \tag{6.3.5}$$

Observe that $\text{tr}_{\mathcal{R}_2}(|\psi\rangle\langle\psi|) = \frac{1}{2}|\psi_1\rangle\langle\psi_1| + \frac{1}{2}|\psi_2\rangle\langle\psi_2|$.

By the 0-speciousness of $\tilde{\mathscr{A}}$, there exists a quantum operation $\mathscr{F}_i : \mathcal{L}(\tilde{\mathcal{A}}_i) \mapsto \mathcal{L}(\mathcal{A}_i)$ such that

$$\Delta\Big((\mathscr{F}_i \otimes \mathbb{1}_{\mathcal{B}_i \otimes \mathcal{R}'}) \cdot (\tilde{\rho}_i(\tilde{\mathscr{A}}, |\psi\rangle\langle\psi|)), \rho_i(|\psi\rangle\langle\psi|)\Big) = 0. \tag{6.3.6}$$

Now consider any isometry $T_i : \tilde{\mathcal{A}}_i \to \mathcal{A}_i \otimes \tilde{\mathcal{A}}_i^*$ such that

$$\mathscr{F}_i(\sigma) = \text{tr}_{\tilde{\mathcal{A}}_i^*}(T_i \cdot \sigma), \tag{6.3.7}$$

for every $\sigma \in \mathcal{L}(\tilde{\mathcal{A}}_i)$. With this in mind, we compute the following trace distance:

$$\begin{aligned}
&\Delta\Big((T_i \otimes \mathbb{1}_{\mathcal{B}_i\otimes\mathcal{R}'}) \cdot (\tilde{\rho}_i(\tilde{\mathscr{A}}, |\psi\rangle\langle\psi|)), \tilde{\gamma}_i \otimes \rho_i(|\psi\rangle\langle\psi|)\Big) \\
&\leq \sqrt{1 - F\Big((T_i \otimes \mathbb{1}_{\mathcal{B}_i\otimes\mathcal{R}'}) \cdot (\tilde{\rho}_i(\tilde{\mathscr{A}}, |\psi\rangle\langle\psi|)), \tilde{\gamma}_i \otimes \rho_i(|\psi\rangle\langle\psi|)\Big)^2}, && (6.3.8) \\
&= \sqrt{1 - F\Big((\mathscr{F}_i \otimes \mathbb{1}_{\mathcal{B}_i\otimes\mathcal{R}}) \cdot (\tilde{\rho}_i(\tilde{\mathscr{A}}, |\psi\rangle\langle\psi|)), \rho_i(|\psi\rangle\langle\psi|)\Big)^2}, && (6.3.9) \\
&= 0. && (6.3.10)
\end{aligned}$$

Equation (6.3.8) follows from an application of the Fuchs-van de Graaf inequality (Theorem 3.3.23). Equation (6.3.9) follows from an application of Uhlmann's theorem (Theorem 3.3.19) and the fact that $\rho_i(|\psi\rangle\langle\psi|)$ is pure for all $1 \leq i \leq n$ (because we consider protocols of the type $\bar{\Pi}^{\mathscr{O}}_U$) and hence there must exist a state $\tilde{\gamma}_i \in \mathcal{D}(\tilde{\mathcal{A}}^*_i)$.

By Theorem 3.3.16, if we apply the projector $P_1 = \mathbb{1}_{\mathcal{A}_i\otimes\mathcal{B}_i\otimes\mathcal{R}} \otimes |1\rangle\langle 1|$ to both states in the above trace distance expression, we still get equality with zero. Put another way, we project both states onto $|1\rangle$ on $\mathcal{R}_2$ which has the effect of turning $|\psi\rangle\langle\psi|$ into $\frac{1}{2}|\psi_1\rangle\langle\psi_1|$. Factoring out the $\frac{1}{2}$, we get

$$\Delta\Big((T_i \otimes \mathbb{1}_{\mathcal{B}_i\otimes\mathcal{R}}) \cdot (\tilde{\rho}_i(\tilde{\mathscr{A}}, |\psi_1\rangle\langle\psi_1|)), \tilde{\gamma}_i \otimes \rho_i(|\psi_1\rangle\langle\psi_1|)\Big) = 0. \tag{6.3.11}$$

Similarly, we can project onto $|2\rangle$ on $\mathcal{R}_2$ to get

$$\Delta((T_i \otimes \mathbb{1}_{\mathcal{B}_i\otimes\mathcal{R}}) \cdot (\tilde{\rho}_i(\tilde{\mathscr{A}}, |\psi_2\rangle\langle\psi_2|)), \tilde{\gamma}_i \otimes \rho_i(|\psi_2\rangle\langle\psi_2|)) = 0. \tag{6.3.12}$$

We have now shown that the lemma is true for the states $|\psi_1\rangle, |\psi_2\rangle \in \mathcal{A}_0 \otimes \mathcal{B}_0 \otimes \mathcal{R}$. To show that the lemma holds for *all* input states, we must prove that the state $\tilde{\gamma}_i$ does not depend on $|\psi_1\rangle$ and $|\psi_2\rangle$. To do this, we can repeat the above argument with $|\psi_1\rangle$ and $|\psi_3\rangle$, for any $|\psi_3\rangle$, to yield a $\tilde{\gamma}'_i$ with

$$\Delta\Big((T_i \otimes \mathbb{1}_{\mathcal{B}_i\otimes\mathcal{R}}) \cdot (\tilde{\rho}_i(\tilde{\mathscr{A}}, |\psi_1\rangle\langle\psi_1|)), \tilde{\gamma}'_i \otimes \rho_i(|\psi_1\rangle\langle\psi_1|)\Big) = 0. \tag{6.3.13}$$

Using the triangle inequality with equation (6.3.11) and equation (6.3.13), we get $\Delta(\tilde{\gamma}_i, \tilde{\gamma}'_i) = 0$. Thus, for *any* state $|\phi\rangle \in \mathcal{A}_0 \otimes \mathcal{B}_0 \otimes \mathcal{R}$, we have

$$\Delta((T_i \otimes \mathbb{1}_{\mathcal{B}_i\otimes\mathcal{R}}) \cdot (\tilde{\rho}_i(\tilde{\mathscr{A}}, |\phi\rangle\langle\phi|)), \tilde{\gamma}_i \otimes \rho_i(|\phi\rangle\langle\phi|)) = 0. \tag{6.3.14}$$

∎

Lemma 6.3.4 characterizes the behaviour of 0-specious adversaries in the sense that at any given step, the state $\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{\text{in}})$ is isometric to $\tilde{\gamma}_i \otimes \rho_i(\rho_{\text{in}})$ for some $\tilde{\gamma}_i$. Observe that this result captures the notion of a specious adversary which "copies." Indeed, if we recall Example 6.3.2, the cloned state is in tensor product with the honest state. The isometry T_i in this case would simply be the identity map.

6.4 The greedy 0-specious adversary

In the previous section, we found that for the class of protocols $\bar{\Pi}_U^{\mathscr{O}}$, a 0-specious adversary $\tilde{\mathscr{A}}$ is restricted in the sense that despite their dishonest behaviour, the overall state at some step i is always isometric to $\tilde{\gamma}_i \otimes \rho_i(\rho_{\text{in}})$ for some $\tilde{\gamma}_i$. In this section, we use this result to define a particular 0-specious adversary, which we call the *greedy adversary* $\hat{\mathscr{A}}$, and prove that it is canonical.

We start with a toy example to motivate the behaviour of the greedy adversary. We then justify, in Lemma 6.4.3, how this behaviour can be generalized. Using this result, we formally define the greedy adversary in Definition 6.4.6.

Example 6.4.1.

(i) *Bob $\mathscr{B}$ chooses two strings $x, y \in \{0,1\}^n$ uniformly at random. He prepares one of them in the computational basis $|x\rangle$ and the other in the Hadamard basis $H^n|y\rangle$. Bob then sends $|x\rangle$ followed by $H^n|y\rangle$ to Alice $\mathscr{A}$.*

Consider the protocol with three different 0-specious adversaries $\hat{\mathscr{A}}$, $\tilde{\mathscr{A}}_1$, and $\tilde{\mathscr{A}}_2$. The protocol description does not say that Alice should copy these states, thus doing so is specious behaviour. $\tilde{\mathscr{A}}_1$ clones $|x\rangle$ only, and $\tilde{\mathscr{A}}_2$ clones $H^n|y\rangle$ only. Clearly, since $\hat{\mathscr{A}}$ knows the encoding basis of both strings, there is nothing stopping them from cloning both $|x\rangle$ and $|y\rangle$; put another way, $\hat{\mathscr{A}}$ performs the specious behaviour of $\tilde{\mathscr{A}}_1$ *and* $\tilde{\mathscr{A}}_2$.

Intuitively, a generalization of the specious behaviour of the above example is that $\hat{\mathscr{A}}$ uses unbounded computational power and quantum memory to perform every 0-specious action available to them at each step. This generalization will form the definition of the greedy adversary. We expect that by performing every such action, we should be able to view an arbitrary 0-specious adversary's view as a subregister which can be obtained by tracing out everything else.

We now introduce some formalism to justify the generalization of this behaviour. To start, let $\hat{\mathscr{A}}$ be a specious adversary such that the isometry in Lemma 6.3.4 is always the identity. Thus, Lemma 6.3.4 reduces to

$$\hat{\rho}_i(\hat{\mathscr{A}}, \rho_{\text{in}}) = \hat{\gamma}_i \otimes \rho_i(\rho_{\text{in}}). \tag{6.4.1}$$

The implies that at each step, $\hat{\mathscr{A}}$ is performing some combination of honest behaviour prescribed by the protocol and dishonest behaviour (which results in the state $\hat{\gamma}_i$). This reveals a small ambiguity: does $\hat{\mathscr{A}}$ perform its dishonest behaviour before or

after its honest behaviour? Consider the case where the action of $\hat{\mathscr{A}}$ at some step of a protocol is of the form

$$\begin{aligned}\hat{\mathscr{A}_i} :&\mathcal{L}(\mathcal{A}_{i-1}) \to \mathcal{L}(\hat{\mathcal{A}}_i),\\ &\rho_{i-1}(\rho_{\text{in}}) \mapsto \hat{\gamma}_i \otimes \rho_i(\rho_{\text{in}}),\end{aligned}$$

where the application of the honest map $\mathscr{B}_i$ is implicit. Formally, the ambiguity is whether the dishonest behaviour, which produces $\hat{\gamma}_i$, depends on $\rho_i(\rho_{\text{in}})$ or $\rho_{i-1}(\rho_{\text{in}})$. If we let $\mathcal{Z}_i$ be the map which produces $\hat{\gamma}_i$ from $\rho_i(\rho_{\text{in}})$, then with $1 \leq i \leq n$ our possibilities are,

$$\hat{\mathscr{A}_i} = \mathcal{Z}_i \circ \mathscr{A}_i : \rho_{i-1}(\rho_{\text{in}}) \mapsto \rho_i(\rho_{\text{in}}) \mapsto \hat{\gamma}_i \otimes \rho_i(\rho_{\text{in}}), \tag{6.4.2}$$

or

$$\hat{\mathscr{A}_i} = \mathscr{A}_i \circ \mathcal{Z}_{i-1} : \rho_{i-1}(\rho_{\text{in}}) \mapsto \hat{\gamma}_{i-1} \otimes \rho_{i-1}(\rho_{\text{in}}) \mapsto \hat{\gamma}_{i-1} \otimes \rho_i(\rho_{\text{in}}). \tag{6.4.3}$$

Note that both of the above scenarios differ at only two points. Equation (6.4.2) allows $\hat{\mathscr{A}}$ to potentially clone something at the final step, $i = n$, while equation (6.4.3) does not; on the other hand, equation (6.4.3) allows potential cloning of something at the beginning of the protocol while equation (6.4.2) does not. Without loss of generality, we adopt the convention of equation (6.4.2) in combination with the additional map

$$\begin{aligned}\mathcal{Z}_0 :\ &\mathcal{L}(\mathcal{A}_0) \to \mathcal{L}(\hat{\mathcal{A}}_0^* \otimes \mathcal{A}_0),\\ &\rho_{\text{in}} \mapsto \hat{\gamma}_0 \otimes \rho_{\text{in}}\,.\end{aligned} \tag{6.4.4}$$

In general, the dishonest behaviour at any given step could additionally depend on previous dishonest behaviour. To capture this, we can extend our convention in the following way:

$$\left.\begin{aligned}&\mathcal{Z}_{i-1} : \rho_{i-1}(\rho_{\text{in}}) \mapsto \hat{\gamma}_{i-1} \otimes \rho_{i-1}(\rho_{\text{in}}),\\ \left(\mathbb{1}_{\mathcal{A}_i' \otimes \mathcal{B}_i' \otimes \mathcal{R}} \otimes \mathscr{O}_i\right)&\left(\mathscr{A}_i \otimes \mathscr{B}_i \otimes \mathbb{1}_{\mathcal{R}}\right) : \hat{\gamma}_{i-1} \otimes \rho_{i-1}(\rho_{\text{in}}) \mapsto \hat{\gamma}_{i-1} \otimes \rho_i(\rho_{\text{in}}),\\ &\mathcal{Z}_i : \hat{\gamma}_{i-1} \otimes \rho_i(\rho_{\text{in}}) \mapsto \hat{\gamma}_i \otimes \rho_i(\rho_{\text{in}}).\end{aligned}\right\} \tag{6.4.5}$$

This covers the situation where the adversary's action

$$\begin{aligned}\mathcal{Z}_i : \mathcal{L}(\hat{\mathcal{A}}_{i-1}^* \otimes \mathcal{A}_i) &\to \mathcal{L}(\hat{\mathcal{A}}_i^* \otimes \mathcal{A}_i),\\ \hat{\gamma}_{i-1} \otimes \rho_i(\rho_{\text{in}}) &\mapsto \hat{\gamma}_i \otimes \rho_i(\rho_{\text{in}}),\end{aligned} \tag{6.4.6}$$

may also utilize both $\rho_i(\rho_{\text{in}})$ *and* $\hat{\gamma}_{i-1}$ in some way. At any step $1 \leq i \leq n$, let C_i denote the set of all quantum channels $\mathcal{Z}_i$ that take the form of equation (6.4.6) and are executed by an adversary in the finite set $\mathcal{S}^0$. Observe that since $\mathcal{S}^0$ is finite, the sets C_i are also finite.

Now that we have clarified how an adversary $\hat{\mathscr{A}}$ (whose isometry in Lemma 6.3.4 is always the identity) behaves, we can formalize the intuition introduced in Example 6.4.1: at a given step i, such an adversary *can* execute all specious actions in the set C_i. The main point to prove is that, for such an adversary, there are no "forks in the road" in the sense that there is not some 0-specious behaviour which, if performed, eliminates the possibility of performing some other 0-specious behaviour.

Remark 6.4.2. Notably, there exists specious behaviour that does not respect the tensor product structure in equation (6.4.1), and hence is not contained in C_i. However, we will see in Section 6.5 that defining the greedy adversary as executing everything in the sets C_i is sufficient for proving that they are canonical.

Lemma 6.4.3. *Let $\bar{\Pi}_U^{\mathscr{O}}$ be a two-party protocol for $U \in \mathsf{U}(\mathcal{A}_0 \otimes \mathcal{B}_0)$. Let $\hat{\mathscr{A}}$ denote a 0-specious adversary whose isometry in Lemma 6.3.4 is always the identity. If, at some step i of the protocol, there exist two dishonest actions $\mathcal{Z}_{i,1}, \mathcal{Z}_{i,2} \in C_i$ available to such an adversary, then $\hat{\mathscr{A}}$ can perform both actions.*

Proof: Consider some step $j < i - 1 < i$ where $\hat{\mathscr{A}}$ performed their first specious action

$$\mathcal{Z}_j : |0\rangle\langle 0| \otimes \rho_j(\rho_{\text{in}}) \mapsto \hat{\gamma}_j \otimes \rho_j(\rho_{\text{in}}). \tag{6.4.7}$$

The adversary $\hat{\mathscr{A}}$ can introduce another ancillary state $|0\rangle\langle 0|$ to *repeat* this specious behaviour,

$$\mathbb{1} \otimes \mathcal{Z}_j : \hat{\gamma}_j \otimes (|0\rangle\langle 0| \otimes \rho_j(\rho_{\text{in}})) \mapsto \hat{\gamma}_j \otimes \hat{\gamma}_j \otimes \rho_j(\rho_{\text{in}}). \tag{6.4.8}$$

Now suppose that $\hat{\mathscr{A}}$ progresses in the protocol while evolving both copies of $\hat{\gamma}_j$ in an identical manner. Consider a later step, i, of the protocol where $\hat{\mathscr{A}}$ has applied the honest map $\mathscr{A}_i$ but has not yet applied one of the two dishonest maps $\mathcal{Z}_{i,1}$ or $\mathcal{Z}_{i,2}$. Then the overall state of the protocol at this point is $\hat{\gamma}_{i-1} \otimes \hat{\gamma}_{i-1} \otimes \rho_i(\rho_{\text{in}})$. The map $\mathcal{Z}_{i,1}$ may depend on $\hat{\gamma}_{i-1}$, and likewise for $\mathcal{Z}_{i,2}$. Since $\hat{\mathscr{A}}$ has two copies of $\hat{\gamma}_{i-1}$, we observe that both maps can be applied. Indeed,

$$\mathcal{Z}_{i,1} \otimes \mathbb{1} : (\hat{\gamma}_{i-1} \otimes \rho_i(\rho_{\text{in}})) \otimes \hat{\gamma}_{i-1} \mapsto (\hat{\gamma}_{i,1} \otimes \rho_i(\rho_{\text{in}})) \otimes \hat{\gamma}_{i-1}, \tag{6.4.9}$$

followed by

$$\mathcal{Z}_{i,2} \otimes \mathbb{1} : (\hat{\gamma}_{i-1} \otimes \rho_i(\rho_{\text{in}})) \otimes \hat{\gamma}_{i,1} \mapsto (\hat{\gamma}_{i,2} \otimes \rho_i(\rho_{\text{in}})) \otimes \hat{\gamma}_{i,1}, \tag{6.4.10}$$

which proves that $\hat{\mathscr{A}}$ can perform both specious actions. ∎

To guarantee that our greedy adversary $\hat{\mathscr{A}}$ can execute, at each step i, every action in the set C_i, we can generalize the technique employed in the proof of Lemma 6.4.3, where the action $\mathcal{Z}_j$ was repeated twice to ensure that both $\mathcal{Z}_{i,1}$ and

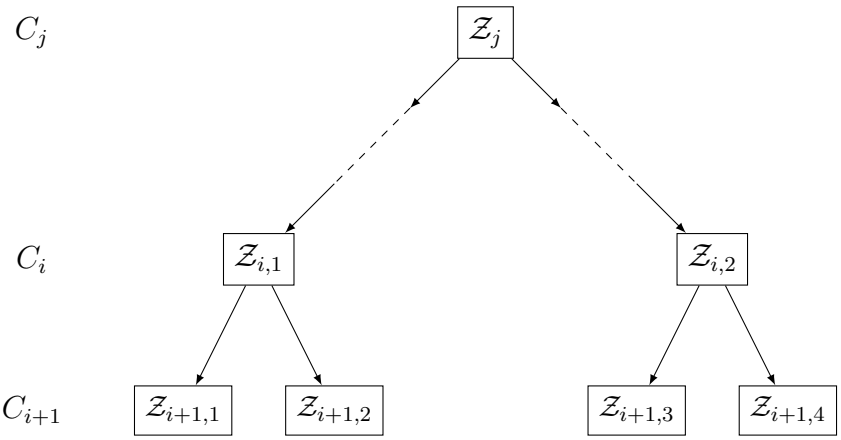

Figure 6.2: A tree graph where each level of the graph corresponds to a set of 0-specious actions C_i. The directed edges from $\mathcal{Z}_{i,1}$ to $\mathcal{Z}_{i+1,1}, \mathcal{Z}_{i+1,2}$ indicate that these latter actions are dependent on $\mathcal{Z}_{i,1}$.

$\mathcal{Z}_{i,2}$ could be executed later in the protocol. To explain this generalization, we represent the sets of specious behaviour $\{C_i\}_i$ as a tree graph. Level j of the graph corresponds to the set C_j, and each vertex in this level corresponds to a specious action $\mathcal{Z}_j \in C_j$. If a specious action at a later step $\mathcal{Z}_i, i > j$ depends upon $\mathcal{Z}_j$, then there is a directed path from $\mathcal{Z}_j$ to $\mathcal{Z}_i$. See Figure 6.2 for a simple example of such a tree. Once we have constructed this tree graph, we can use Algorithm 6.4.4 below to determine how many times some specious action should be repeated. The application of this algorithm to the tree graph in Figure 6.2 is visualized in Figure 6.3.

Algorithm 6.4.4.

1. Construct a tree graph from the sets of 0-specious actions $C_i, 0 \leq i \leq n$.

2. Starting at the root vertex of the tree, count the number of leaves that are connected to the root vertex by a path. If there are k leaves, then execute the specious behaviour associated to the root vertex k-times. Each leaf can now be associated to its own copy of the root vertex.

3. For a fixed path from a copy of the root vertex to a leaf, progressing the protocol corresponds to moving down the path towards the leaf. At each vertex encountered along the way, execute the associated specious behaviour at the appropriate step. Do this simultaneously for all other paths.

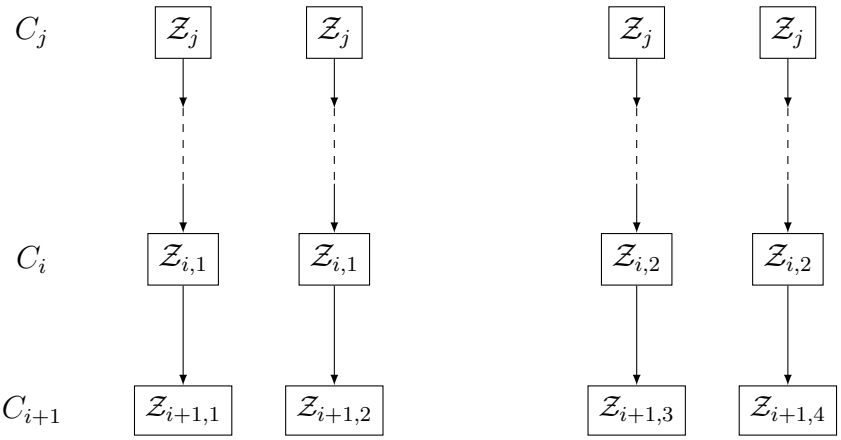

Figure 6.3: The effect of applying Algorithm 6.4.4 to the tree graph in Figure 6.2. Since there were four leaves $\{\mathcal{Z}_{i+1,1}, \mathcal{Z}_{i+1,2}, \mathcal{Z}_{i+1,3}, \mathcal{Z}_{i+1,4}\}$, the specious action $\mathcal{Z}_j$ was repeated four times at step j. The result is four registers, where each register corresponds to one of the above paths and evolves accordingly.

Recall that the sets C_i are finite and that we consider the information-theoretic setting, where we place no restriction on the quantum memory or computational power of a specious adversary, and so we are not concerned with the ability of the greedy adversary to execute Algorithm 6.4.4. The case where the sets C_i are infinite is discussed further as a future direction in Section 6.6.

With Lemma 6.4.3 and Algorithm 6.4.4, we have the following corollary.

Corollary 6.4.5. *Let $\bar{\Pi}_U^{\mathscr{O}}$ be a two-party protocol for $U \in \mathsf{U}(\mathcal{A}_0 \otimes \mathcal{B}_0)$. A 0-specious adversary, whose isometry in Lemma 6.3.4 is always the identity, can execute every action in the set C_i, for $0 \leq i \leq n$ by following Algorithm 6.4.4.*

Definition 6.4.6 (The greedy 0-specious adversary)**.** *Let $\bar{\Pi}_U^{\mathscr{O}}$ be a two-party protocol for $U \in \mathsf{U}(\mathcal{A}_0 \otimes \mathcal{B}_0)$. The **greedy** 0-specious adversary $\hat{\mathscr{A}}$ is defined by the following characteristics:*

(i) The isometry in Lemma 6.3.4 is always the identity.

(ii) $\hat{\mathscr{A}}$ executes all 0-specious behaviour in C_i, for $0 \leq i \leq n$, by following Algorithm 6.4.4.

Remark 6.4.7. Instead of Algorithm 6.4.4, we could consider a different algorithm which does not immediately repeat the root vertex of the tree graph as many times as there are leaves. Instead, the algorithm could be such that each action is repeated as needed, meaning that as the protocol progresses (descending down the tree), each vertex/action is repeated only as many times as there are directed edges leaving it. Since

this achieves the same result as Algorithm 6.4.4, we only consider Algorithm 6.4.4 since it is conceptually simpler.

6.5 The greedy adversary is canonical

The goal of this section is to prove that the greedy adversary $\hat{\mathscr{A}}$ is canonical in the sense of Definition 6.3.1. Informally, this means that we can map the simulated view of $\hat{\mathscr{A}}$ to the simulated view of an arbitrary 0-specious adversary $\tilde{\mathscr{A}} \in \mathcal{S}^0$. Since privacy is defined in terms of simulators, the ability to construct such a mapping then allows us to obtain privacy against $\tilde{\mathscr{A}}$ by proving privacy against $\hat{\mathscr{A}}$.

Recall that the intuition which led to the greedy adversary was that if $\hat{\mathscr{A}}$ does all specious behaviour, then $\tilde{\mathscr{A}}$'s dishonest register can be viewed as a subregister of $\hat{\mathscr{A}}$'s dishonest register. Then mapping the view of $\hat{\mathscr{A}}$ to $\tilde{\mathscr{A}}$ roughly corresponds to tracing out the "complement." It is not immediately obvious that this is always possible, since $\hat{\mathscr{A}}$ does not really perform *all* specious behaviour, only everything contained in the sets C_i. However, we show that even if $\tilde{\mathscr{A}}$ executes specious behaviour not contained in C_i, the resulting register can still be viewed as subregister of $\hat{\mathscr{A}}$'s view under the isometry from Lemma 6.3.4.

We start in Section 6.5.1 by considering perfect simulators ($\delta = 0$) and then, in Section 6.5.2, we consider the case of imperfect simulators ($\delta > 0$).

6.5.1 A perfect simulator

Recall the notation

$$\nu_i(\tilde{\mathscr{A}}, \rho_{\text{in}}) \coloneqq \operatorname{tr}_{\mathcal{B}_0}\left(\left(\tilde{\mathscr{S}}_i \otimes \mathbb{1}_{\mathcal{B}_0 \otimes \mathcal{R}}\right) \cdot ((U^{q_i} \otimes \mathbb{1}_{\mathcal{R}}) \cdot \rho_{\text{in}})\right), \tag{6.5.1}$$

where $\tilde{\mathscr{S}}$ is the simulator for $\tilde{\mathscr{A}}$. Now suppose we have a perfect simulator for an arbitrary 0-specious $\tilde{\mathscr{A}}$, meaning the protocol is 0-private ($\delta = 0$). By Definition 6.2.5, this means that for all $1 \leq i \leq n$, we have,

$$\Delta\left(\nu_i(\tilde{\mathscr{A}}, \rho_{\text{in}}), \operatorname{tr}_{\mathcal{B}_i}(\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{\text{in}}))\right) = 0. \tag{6.5.2}$$

Our task is to use a perfect simulator $\hat{\mathscr{S}}$ for some other 0-specious adversary $\hat{\mathscr{A}}$ and construct a map $\mathcal{M}_i : \mathcal{L}(\hat{\mathcal{A}}_i) \to \mathcal{L}(\tilde{\mathcal{A}}_i)$ such that

$$\Delta\left(\mathcal{M}_i \cdot \nu_i(\hat{\mathscr{A}}, \rho_{\text{in}}), \operatorname{tr}_{\mathcal{B}_i}(\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{\text{in}}))\right) = 0. \tag{6.5.3}$$

In other words, the $\delta = 0$ condition means that we must construct $\operatorname{tr}_{\mathcal{B}_i}(\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{\text{in}}))$ from $\nu_i(\hat{\mathscr{A}}, \rho_{\text{in}})$.

We start by recalling the effect of the isometry $\tilde{T}_i : \tilde{\mathcal{A}}_i \to \mathcal{A}_i \otimes \tilde{\mathcal{A}}_i^*$ from Lemma 6.3.4.

$$(\tilde{T}_i \otimes \mathbb{1}_{\mathcal{R}}) \cdot \mathrm{tr}_{\mathcal{B}_i}(\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{\mathrm{in}})) = \mathrm{tr}_{\mathcal{B}_i}\left((\tilde{T}_i \otimes \mathbb{1}_{\mathcal{B}_i\mathcal{R}}) \cdot (\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{\mathrm{in}}))\right), \tag{6.5.4}$$

$$= \mathrm{tr}_{\mathcal{B}_i}\left(\tilde{\gamma}_i \otimes \rho_i(\rho_{\mathrm{in}})\right), \tag{6.5.5}$$

$$= \tilde{\gamma}_i \otimes \mathrm{tr}_{\mathcal{B}_i}(\rho_i(\rho_{\mathrm{in}})). \tag{6.5.6}$$

In the special case of a 0-specious adversary $\hat{\mathscr{A}}$ whose isometry $\hat{T}_i$ is the identity, we have

$$\mathrm{tr}_{\mathcal{B}_i}(\hat{\rho}_i(\hat{\mathscr{A}}, \rho_{\mathrm{in}})) = \hat{\gamma}_i \otimes \mathrm{tr}_{\mathcal{B}_i}(\rho_i(\rho_{\mathrm{in}})), \tag{6.5.7}$$

and hence, by the $\delta = 0$ condition, $\nu_i(\hat{\mathscr{A}}, \rho_{\mathrm{in}}) = \hat{\gamma}_i \otimes \mathrm{tr}_{\mathcal{B}_i}(\rho_i(\rho_{\mathrm{in}}))$.

From these observations, we claim that we can construct the quantum channel $\mathcal{M}_i : \mathcal{L}(\hat{\mathcal{A}}_i) \to \mathcal{L}(\tilde{\mathcal{A}}_i)$ from a quantum channel of the form

$$\begin{aligned} \mathcal{V}_i : \hat{\mathcal{A}}_i^* &\longrightarrow \tilde{\mathcal{A}}_i^* \\ \hat{\gamma}_i &\longmapsto \tilde{\gamma}_i, \end{aligned}$$

and the quantum channel $\tilde{T}_i^{-1}$ that reverses the isometry $\tilde{T}_i$ (recall equation (3.3.29)).

Indeed, given these quantum channels, and starting from $\hat{\mathscr{A}}$'s simulated state $\nu_i(\hat{\mathscr{A}}, \rho_{\mathrm{in}}) = \hat{\gamma}_i \otimes \mathrm{tr}_{\mathcal{B}_i}(\rho_i(\rho_{\mathrm{in}}))$, we could perform the following sequence of operations:

$$\mathcal{V}_i \otimes \mathbb{1}_{\mathcal{A}_i\mathcal{R}} : \hat{\gamma}_i \otimes \mathrm{tr}_{\mathcal{B}_i}(\rho_i(\rho_{\mathrm{in}})) \longmapsto \tilde{\gamma}_i \otimes \mathrm{tr}_{\mathcal{B}_i}(\rho_i(\rho_{\mathrm{in}})), \tag{6.5.8}$$

$$\tilde{T}_i^{-1} \otimes \mathbb{1}_{\mathcal{A}_i\mathcal{R}} : \tilde{\gamma}_i \otimes \mathrm{tr}_{\mathcal{B}_i}(\rho_i(\rho_{\mathrm{in}})) \longmapsto \mathrm{tr}_{\mathcal{B}_i}(\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{\mathrm{in}})). \tag{6.5.9}$$

See Figure 6.4 for a visualization of this construction.

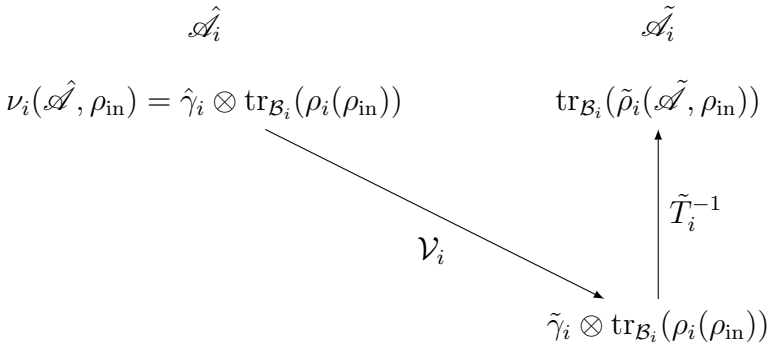

Figure 6.4: The map $\mathcal{M}_i = (\tilde{T}_i^{-1} \circ \mathcal{V}_i \otimes \mathbb{1}_{\mathcal{A}_i\mathcal{R}})$, which turns a simulator for $\hat{\mathscr{A}}$ into a simulator for $\tilde{\mathscr{A}}$, applied to the $\delta = 0$ setting. We think of $\mathcal{V}_i$ as operating "under" the isometries $\hat{T}_i$ and $\tilde{T}_i$ where, by choice of $\hat{\mathscr{A}}$, the isometry $\hat{T}_i$ is the identity.

Thus, we have shown that the composition $\tilde{T}_i^{-1} \circ \mathcal{V}_i$ with the simulator $\hat{\mathscr{S}}_i$ is a construction of a perfect simulator for $\tilde{\mathscr{A}}$. The last thing we need to address is the construction of the map $\mathcal{V}_i$, which is handled in the proof of Lemma 6.5.1 by taking $\hat{\mathscr{A}}$ to be the greedy adversary.

Lemma 6.5.1. *Let $\bar{\Pi}_U^{\mathscr{O}}$ be a two-party protocol for $U \in \mathsf{U}(\mathcal{A}_0 \otimes \mathcal{B}_0)$. Let $\mathcal{S}^0$ be a finite set of arbitrary 0-specious adversaries, and let $\hat{\mathscr{A}}$ be the greedy adversary for the set $\mathcal{S}^0$. Then a perfect simulator for $\tilde{\mathscr{A}} \in \mathcal{S}^0$ can be constructed from a perfect simulator for $\hat{\mathscr{A}}$.*

Proof: Recall that to construct the simulator $\tilde{\mathscr{S}}_i$, it suffices to construct a mapping $\mathcal{V}_i : \hat{\gamma}_i \mapsto \tilde{\gamma}_i$, where $\hat{\gamma}_i \in \hat{\mathcal{A}}_i^*$ and $\tilde{\gamma}_i \in \tilde{\mathcal{A}}_i^*$. Also recall that $C_i = \{\mathcal{Z}_i\}$ denotes the set of all specious behaviour available to an adversary, at step i, whose isometry in Lemma 6.3.4 is the identity; that is, $\mathcal{Z}_i \in C_i$ is a dishonest action that results in the following tensor product,

$$\hat{\rho}_i(\hat{\mathscr{A}}, \rho_{\text{in}}) = \hat{\gamma}_i \otimes \rho_i(\rho_{\text{in}}). \tag{6.5.10}$$

By Corollary 6.4.5, the greedy adversary performs every action in the sets $\{C_i\}_i$ via Algorithm 6.4.4. As part of this algorithm, an action $\mathcal{Z}_i \in C_i$ may be repeated m times, resulting in m dishonest registers in tensor product with $\rho_i(\rho_{\text{in}})$. Thus, we rewrite $\hat{\gamma}_i$ as $\hat{\gamma}_i = \hat{\gamma}_{i,1} \otimes \hat{\gamma}_{i,2} \otimes \cdots \otimes \hat{\gamma}_{i,m}$.

Now suppose that at this step i, and all previous steps, the specious behaviour performed by $\tilde{\mathscr{A}}$ forms a subset of $C_0, \ldots, C_i$ (so $\tilde{\mathscr{A}}$ only performed specious actions that respected equation (6.5.10)). Then, without loss of generality, we can write $\tilde{\mathscr{A}}$'s dishonest state as $\tilde{\gamma}_i = \hat{\gamma}_{i,1} \otimes \cdots \otimes \hat{\gamma}_{i,k} \in \tilde{\mathcal{A}}_i^*$ for some $1 \leq k < m$. Then let $\tilde{\gamma}_i^{\perp} \coloneqq \hat{\gamma}_{i,k+1} \otimes \cdots \otimes \hat{\gamma}_{i,m} \in \tilde{\mathcal{A}}_i^{*\perp}$. Thus,

$$\hat{\gamma}_i \equiv \tilde{\gamma}_i \otimes \tilde{\gamma}_i^{\perp}, \tag{6.5.11}$$

and so $\tilde{\mathscr{A}}$'s dishonest state is a subregister of $\hat{\mathscr{A}}$'s dishonest state. As a consequence, we can define our map as tracing out $\tilde{\gamma}_i^{\perp}$, that is, $\mathcal{V}_i \coloneqq \text{tr}_{\tilde{\mathcal{A}}_i^{*\perp}}(\cdot)$.

The other case to consider is when $\tilde{\mathscr{A}}$'s specious behaviour cannot be described as a subset of C_i. More precisely, there exists a quantum channel $\tilde{\mathscr{A}}_i$ that is not exactly equal to a pair $(\mathscr{A}_i, \mathcal{Z}_i)$ where $\mathcal{Z}_i \in C_i$. For this case, we must determine if the state $\tilde{\gamma}_i$, produced under application of the isometry $\tilde{T}_i$, could also have been produced by $\hat{\mathscr{A}}$. If so, then equation (6.5.11) would still be satisfied and we could still define our mapping as $\mathcal{V}_i \coloneqq \text{tr}_{\tilde{\mathcal{A}}_i^{*\perp}}(\cdot)$. To answer this, consider the composition $\tilde{T}_i \circ \tilde{\mathscr{A}}_i \circ \tilde{T}_{i-1}^{-1}$,

$$\begin{aligned}
\tilde{T}_{i-1}^{-1} \otimes \mathbb{1}_{\mathcal{B}_{i-1} \otimes \mathcal{R}} &: \tilde{\gamma}_{i-1} \otimes \rho_{i-1}(\rho_{\text{in}}) \mapsto \tilde{\rho}_{i-1}(\tilde{\mathscr{A}}, \rho_{\text{in}}), \\
\left(\mathbb{1}_{\mathcal{A}_i' \otimes \mathcal{B}_i' \otimes \mathcal{R}} \otimes \mathscr{O}_i\right)\left(\tilde{\mathscr{A}}_i \otimes \mathscr{B}_i \otimes \mathbb{1}_{\mathcal{R}}\right) &: \tilde{\rho}_{i-1}(\tilde{\mathscr{A}}, \rho_{\text{in}}) \mapsto \tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{\text{in}}), \\
\tilde{T}_i \otimes \mathbb{1}_{\mathcal{B}_i \otimes \mathcal{R}} &: \tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{\text{in}}) \mapsto \tilde{\gamma}_i \otimes \rho_i(\rho_{\text{in}}).
\end{aligned}$$

Thus, the composition $\tilde{T}_i \circ \tilde{\mathscr{A}}_i \circ \tilde{T}_{i-1}^{-1}$ corresponds to a pair $(\mathscr{A}_i, \mathcal{Z}_i)$ for some $\mathcal{Z}_i \in C_i$ (see equation (6.4.5) for comparison). Then, by definition of the greedy adversary, this $\mathcal{Z}_i$ would have been executed, and so $\tilde{\gamma}_i$ still satisfies equation (6.5.11). Thus, our mapping $\mathcal{V}_i$ is still defined as $\mathcal{V}_i \coloneqq \mathrm{tr}_{\tilde{A}_i^{*\perp}}(\cdot)$. With $\mathcal{V}_i$ defined, we can apply the channel $\tilde{T}_i^{-1} \circ \mathcal{V}_i \otimes \mathbb{1}_{\mathcal{R}}$ to the simulated state $\nu_i(\hat{\mathscr{A}}, \rho_{\text{in}})$ to get a state that perfectly simulates the view of $\tilde{\mathscr{A}}$. ∎

6.5.2 An imperfect simulator

In this section, we consider the case where the simulator is not perfect (*i.e.*, we suppose that the protocol is δ-private against an arbitrary 0-specious adversary $\tilde{\mathscr{A}} \in \mathcal{S}^0$, for some fixed $\delta > 0$). This means that for all $1 \leq i \leq n$, we have,

$$\Delta\left(\nu_i(\tilde{\mathscr{A}}, \rho_{\text{in}}), \mathrm{tr}_{\mathcal{B}_i}(\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{\text{in}}))\right) \leq \delta. \tag{6.5.12}$$

Similar to the $\delta = 0$ case in Section 6.5.1, our task is to use a simulator $\hat{\mathscr{S}}$, for some 0-specious adversary $\hat{\mathscr{A}}$, and then construct a map $\mathcal{M}_i$ such that

$$\Delta\left(\mathcal{M}_i \cdot \nu_i(\hat{\mathscr{A}}, \rho_{\text{in}}), \mathrm{tr}_{\mathcal{B}_i}(\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{\text{in}}))\right) \leq \delta. \tag{6.5.13}$$

We claim that the approach taken in Section 6.5.1 also works in this case. We know that for the special case of a 0-specious adversary $\hat{\mathscr{A}}$ whose isometry $\hat{T}_i$ is the identity, we have $\mathrm{tr}_{\mathcal{B}_i}(\hat{\rho}_i(\hat{\mathscr{A}}, \rho_{\text{in}})) = \hat{\gamma}_i \otimes \mathrm{tr}_{\mathcal{B}_i}(\rho_i(\rho_{\text{in}}))$. Thus, by equation (6.5.8) and equation (6.5.9),

$$(\tilde{T}_i^{-1} \circ \mathcal{V}_i \otimes \mathbb{1}_{\mathcal{A}_i\mathcal{R}}) \cdot \left(\mathrm{tr}_{\mathcal{B}_i}(\hat{\rho}_i(\hat{\mathscr{A}}, \rho_{\text{in}}))\right) = \mathrm{tr}_{\mathcal{B}_i}(\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{\text{in}})). \tag{6.5.14}$$

Then,

$$\begin{aligned} \delta &\geq \Delta\left(\nu_i(\hat{\mathscr{A}}, \rho_{\text{in}}), \mathrm{tr}_{\mathcal{B}_i}(\hat{\rho}_i(\hat{\mathscr{A}}, \rho_{\text{in}}))\right), && (6.5.15)\\ &\geq \Delta\left((\tilde{T}_i^{-1} \circ \mathcal{V}_i \otimes \mathbb{1}_{\mathcal{A}_i\mathcal{R}}) \cdot \nu_i(\hat{\mathscr{A}}, \rho_{\text{in}}), \mathrm{tr}_{\mathcal{B}_i}(\tilde{\rho}_i(\tilde{\mathscr{A}}, \rho_{\text{in}}))\right), && (6.5.16) \end{aligned}$$

where in the second line we used equation (6.5.14) and the fact that the trace distance contracts under quantum operations. Thus, $\mathcal{M}_i = (\tilde{T}_i^{-1} \circ \mathcal{V}_i \otimes \mathbb{1}_{\mathcal{A}_i\mathcal{R}})$. See Figure 6.5 for a visualization of this technique.

Notably, the construction of the quantum channel $\mathcal{V}_i$, given in the proof of Lemma 6.5.1, relied on taking $\hat{\mathscr{A}}$ to be the greedy adversary but it did not depend on the choice of δ, and thus the same construction can be used in this case. With this in mind, we have the following lemma.

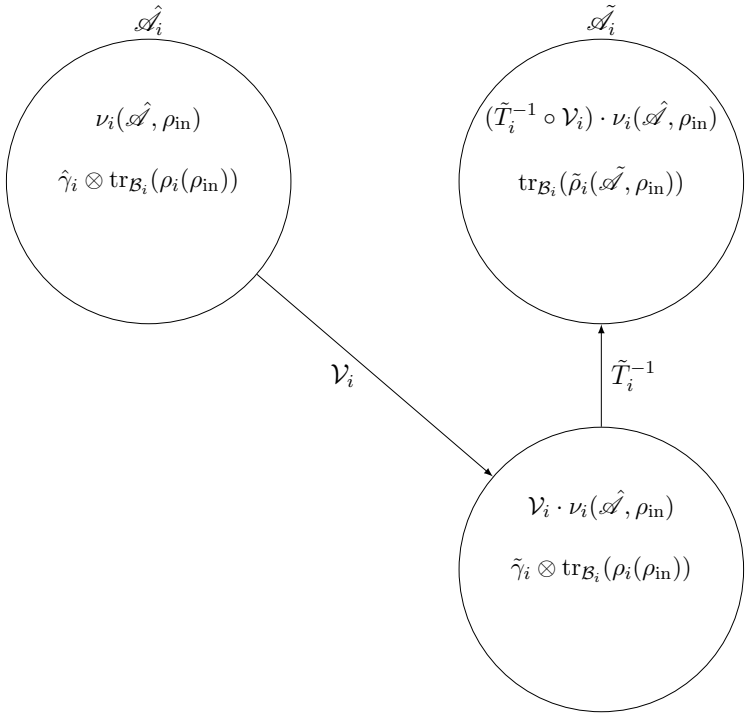

Figure 6.5: The map $\mathcal{M}_i = (\tilde{T}_i^{-1} \circ \mathcal{V}_i \otimes \mathbb{1}_{\mathcal{A}_i\mathcal{R}})$ applied to the $\delta > 0$ setting. Here, we start with a state $\nu_i(\hat{\mathscr{A}}, \rho_{\text{in}})$ that is δ-close to the real view of $\hat{\mathscr{A}}$; this is visualized as the state being contained within the ball, of radius δ, that is centered on the real state $\hat{\gamma}_i \otimes \text{tr}_{\mathcal{B}_i}(\rho_i(\rho_{\text{in}}))$.

Lemma 6.5.2. *Let $\bar{\Pi}_U^{\mathscr{O}}$ be a two-party protocol for $U \in \mathsf{U}(\mathcal{A}_0 \otimes \mathcal{B}_0)$. Let $\mathcal{S}^0$ be a finite set of arbitrary 0-specious adversaries, and let $\hat{\mathscr{A}}$ be the greedy adversary for the set $\mathcal{S}^0$. Then for a fixed $\delta > 0$, an imperfect simulator for $\tilde{\mathscr{A}} \in \mathcal{S}^0$ can be constructed from an imperfect simulator for $\hat{\mathscr{A}}$.*

Combining Lemma 6.5.1 and Lemma 6.5.2, we get our main result.

Theorem 6.5.3. *Let $\bar{\Pi}_U^{\mathscr{O}}$ be a two-party protocol for $U \in \mathsf{U}(\mathcal{A}_0 \otimes \mathcal{B}_0)$. Then the greedy adversary is a $\mathcal{S}^0$-canonical 0-specious adversary.*

6.6 Summary and future directions

In this chapter, we have examined the class of adversaries known as specious and the notion of a canonical specious adversary. Specious adversaries are weak in the sense that they must always appear honest; if an audit is called at any step of a two-party protocol, they must ensure that their state, when combined with the honest party's state, is close to the joint state where both parties are honest. We defined a specious

adversary as canonical if a proof of security against them implies security against a finite set of arbitrary specious adversaries.

Practically, specious adversaries are useful for testing the security of some candidate two-party protocol; if a protocol is insecure against specious adversaries, then it does not satisfy the minimum level of reasonable security. Additionally, security against specious adversaries may imply security against stronger adversaries, as suggested in [DNS12]. The notion of a canonical specious adversary then promises to simplify this latter technique, as checking security against specious adversaries amounts to checking security against a single specious adversary.

We have shown that the well-known purified adversary does not obviously satisfy our definition of canonical. As a result, we have formally defined a specious adversary, called the greedy adversary, which can use unbounded computational power and memory to perform all dishonest actions of the adversaries it is trying to simulate. We then proved that, within a restricted setting, the greedy adversary is indeed canonical.

Although the greedy adversary proves the existence of a canonical specious adversary, it does not clearly satisfy our initial aim of using a canonical specious adversary to simplify the security analysis of protocols. The way we have defined the greedy adversary (as executing "all" dishonest actions) makes it cumbersome to work with, and proving security against it is not obviously simpler than proving security against an arbitrary specious adversary.

Following our results, there are numerous future directions that one could take.

Can we define a specious adversary that is canonical for an entire class of specious adversaries? Recall that our notion of canonical is defined relative to a finite set of specious adversaries (see Definition 6.3.1). The reason we restrict to a finite set is to ensure that our greedy adversary can actually execute "all" specious behaviour; without this restriction, the set of dishonest actions C_i available to the greedy adversary becomes infinite. At this point, we would need more sophisticated techniques to handle these infinite sets, or it may be possible to prove that a specious adversary cannot satisfy our definition of canonical for an entire class of specious adversaries. Alternatively, it may be possible to show that simulating a finite set of specious adversaries is actually sufficient, under some form of approximation; that is, by simulating a finite set, we can approximately simulate *any* specious adversary within the class. To enable this approach, it may be necessary to slightly relax our requirement that, to be canonical, the level of privacy cannot degrade.

In comparison to the greedy adversary, can we define a canonical 0-specious adversary that is easier to work with? Ideally, a canonical specious adversary could be defined according to something like an information measure in the following sense: at each step, the dishonest behaviour of the canonical specious adversary

is characterized by maximizing some information measure. Then, proving security against this adversary amounts to just working with this measure, which would be simpler than proving security against the greedy adversary.

To prove that such an adversary is canonical, the approach is roughly as follows: if $\hat{\mathscr{A}}$ maximizes the quantity, while $\tilde{\mathscr{A}}$ does not, then there exists a quantum channel from $\hat{\mathscr{A}}$'s simulated view to a simulated view of $\tilde{\mathscr{A}}$, such that the level of privacy is preserved. The difficulty, here, is this latter point: if, under the quantum channel, an "extra epsilon" is introduced, then we will not preserve the level of privacy, and thus will not satisfy our definition of canonical.

In a less restricted setting, how can we prove that a specious adversary is canonical? We have focused on 0-specious adversaries and protocols that only use isometries. These restrictions enabled Lemma 6.3.4, which was crucially used in proving that the greedy adversary was canonical.

If we generalize to two-party protocols where the honest parties may apply quantum channels, or to ε-specious adversaries where $\varepsilon > 0$, then it is unlikely that the lemma will hold exactly. As a consequence, our technique for proving that a specious adversary is canonical is also unlikely to hold in the general setting, and thus a different approach would be needed.

Furthermore, the class of specious adversaries with $\varepsilon > 0$ are able to do far more than the $\varepsilon = 0$ class. Notably, $\varepsilon > 0$ means that they do not need to perfectly reconstruct the honest state at each step, so their dishonest behaviour could even include measurements that only slightly perturb the state (see [Wil13] for the notion of *gentle measurements*). Thus, a canonical specious adversary for this setting would need to be defined in such a way that it captures this larger set of dishonest behaviour.

Chapter 7

Conclusion

We have identified three practical problems in quantum cryptography and, for each problem, we have presented a possible solution. By using a post-quantum computational assumption, we have obtained a device-independent protocol for oblivious transfer that relaxes the non-communication assumption for the device's components. We have generalized the [AC12] public-key quantum money scheme so that the verification algorithm inherently tolerates noise. Lastly, with the aim of simplifying security proofs against specious adversaries, we have defined the notion of a canonical specious adversary and identified an adversary that satisfies our definition, within a restricted setting.

While each of these problems have a practical flavour, they are noticeably distinct. The point of this observation is to highlight the great variation in the practical problems facing quantum cryptography. Furthermore, it emphasizes that to solve such problems, a wide breadth of techniques are needed.

There are, of course, many other practical problems that remain to be solved, but even those discussed in this thesis can be further improved upon. To this latter point, we conclude by recalling several questions that were raised throughout the thesis and may serve as directions for future research. For additional open questions, and detailed discussions on the questions listed here, see the end of the appropriate chapter (Section 4.7, Section 5.7, and Section 6.6).

Can we prove security of our device-independent oblivious transfer protocol in a more general setting? When we proved the security of our scheme, we made the IID assumption: that the device behaves independently and identically in each round. This assumption significantly simplifies the analysis, though at the cost of weakening the meaning of device-independence; for true device-independence, we must prove security in the non-IID setting.

We presented two approaches to verifying a quantum money state: the subset approach and the subspace approach. Is one preferable over the other? Both approaches (see Section 5.5) achieve the same completeness and soundness error, but is one easier to realize in the plain model? Towards answering this question, we note that in [Zha19], Zhandry instantiated the oracle scheme of [AC12]. The technique used seems to naturally apply to our subspace approach, thus implying that it can also be realized in the plain model. Is it any more difficult to use this technique to instantiate the oracle in our subset approach?

How does adding a layer of error correction compare to our approach for designing a noise-tolerant money scheme? In Section 5.3.2, we briefly discussed how one could take a different approach to noise-tolerance by applying a layer of error correction on top of the [AC12] oracle scheme. Given a realization of this approach, it would be interesting to make a formal comparison with our scheme.

Is it possible to obtain noise-tolerance for other public-key quantum money schemes? In Section 5.1.3, we mentioned some of the other public-key quantum money schemes that rely on hardness assumptions instead of oracles. Is it possible to make any of these schemes noise-tolerant?

In comparison to the greedy adversary, can we define a canonical 0-specious adversary that is easier to work with? Ideally, a canonical specious adversary could be defined according to something like an information measure in the following sense: at each step, the dishonest behaviour of the canonical specious adversary is characterized by maximizing some information measure. Then, proving security against this adversary amounts to just working with this measure, which would be simpler than proving security against the greedy adversary.

What can we say about a canonical specious adversary in a more general setting? In Chapter 6, we focused on finite sets of 0-specious adversaries and protocols that only use isometries. These restrictions were crucial to proving that the greedy adversary is canonical. Naturally, we can ask what happens if we relax these restrictions. To consider the entire class of specious adversaries, rather than a finite set, there will be infinitely many specious adversaries to account for. If we generalize to two-party protocols where the honest parties may apply quantum channels, or to ε-specious adversaries where $\varepsilon > 0$ (and, as a consequence, have access to a larger set of dishonest actions), then it is unlikely that our proof techniques will continue to hold.

Bibliography

[AA17] R. Amiri and J. M. Arrazola. Quantum money with nearly optimal error tolerance. *Physical Review A*, 95(6): 062334, 2017.
DOI: 10.1103/PhysRevA.95.062334.

[Aar09] S. Aaronson. Quantum copy-protection and quantum money. In *24th Annual Conference on Computational Complexity (CCC 2009)*, pages 229–242, 2009.
DOI: 10.1109/CCC.2009.42.

[ABC+19] D. Aharonov, Z. Brakerski, K.-M. Chung, A. Green, C.-Y. Lai, and O. Sattath. On quantum advantage in information theoretic single-server PIR. In *Advances in Cryptology — EUROCRYPT 2019*, volume 3, pages 219–246, 2019.
DOI: 10.1007/978-3-030-17659-4_8.

[AC12] S. Aaronson and P. Christiano. Quantum money from hidden subspaces. In *STOC '12: Proceedings of the fourty-fourth ACM symposium on Theory of computing*, pages 41–60, 2012.
DOI: 10.1145/2213977.2213983.

[ADF+18] R. Arnon-Friedman, F. Dupuis, O. Fawzi, R. Renner, and T. Vidick. Practical device-independent quantum cryptography via entropy accumulation. *Nature Communications*, 9(1): 459:1–459:11, 2018.
DOI: 10.1038/s41467-017-02307-4.

[AHY23] P. Ananth, Z. Hu, and H. Yuen. On the (im)plausibility of public-key quantum money from collision-resistant hash functions. In *Advances in Cryptology — ASIACRYPT 2023*, pages 39–72, 2023.
DOI: 10.1007/978-981-99-8742-9_2.

[Amb02] A. Ambainis. Quantum lower bounds by quantum arguments. *Journal of Computer and System Sciences*, 64(4): 750–767, 2002.
DOI: 10.1006/jcss.2002.1826.

[AMPS16] N. Aharon, S. Massar, S. Pironio, and J. Silman. Device-independent bit commitment based on the CHSH inequality. *New Journal of Physics*, 18(2): 25014, 2016.
DOI: 10.1088/1367-2630/18/2/025014.

[Arn18] R. Arnon-Friedman. *Reductions to IID in Device-independent Quantum Information Processing*. PhD thesis, ETH Zurich, 2018.
DOI: 10.3929/ethz-b-000298420.

[ARV19] R. Arnon-Friedman, R. Renner, and T. Vidick. Simple and tight device-independent security proofs. *SIAM Journal on Computing*, 48(1): 181–225, 2019.
DOI: 10.1137/18M1174726.

[BB84] C. H. Bennett and G. Brassard. Quantum cryptography: Public key distribution and coin tossing. In *International Conference on Computers, Systems and Signal Processing*, pages 175–179, 1984.

[BB15] Ä. Baumeler and A. Broadbent. Quantum private information retrieval has linear communication complexity. *Journal of Cryptology*, 28(1): 161–175, 2015.
DOI: 10.1007/s00145-014-9180-2.

[BBBV97] C. H. Bennett, E. Bernstein, G. Brassard, and U. Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5): 1510–1523, 1997.
DOI: 10.1137/S0097539796300933.

[BCF+96] H. Barnum, C. M. Caves, C. A. Fuchs, R. Jozsa, and B. Schumacher. Noncommuting mixed states cannot be broadcast. *Physical Review Letters*, 76(15): 2818–2821, 1996.
DOI: 10.1103/PhysRevLett.76.2818.

[BCM+18] Z. Brakerski, P. Christiano, U. Mahadev, U. Vazirani, and T. Vidick. A cryptographic test of quantumness and certifiable randomness from a single quantum device. In *2018 59th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2018)*, pages 320–331, 2018.
DOI: 10.1109/FOCS.2018.00038.

[BDG23] A. Bilyk, J. Doliskani, and Z. Gong. Cryptanalysis of three quantum money schemes. *Quantum Information Processing*, 22(4): 177:1–177:27, 2023.
DOI: 10.1007/s11128-023-03919-0.

[Bel64] J. S. Bell. On the Einstein-Podolsky-Rosen paradox. *Physics Physique Fizika*, 1(3): 195–200, 1964.
DOI: 10.1103/PhysicsPhysiqueFizika.1.195.

[BGI+12] B. Barak, O. Goldreich, R. Impagliazzo, S. Rudich, A. Sahai, S. Vadhan, and K. Yang. On the (im)possibility of obfuscating programs. *Journal of the ACM*, 59(2): 6, 2012.
DOI: 10.1145/2160158.2160159.

[BGMZ18] J. Bartusek, J. Guan, F. Ma, and M. Zhandry. Return of GGH15: provable security against zeroizing attacks. In *Theory of Cryptography (TCC 2018)*, volume 2, pages 544–574, 2018.
DOI: 10.1007/978-3-030-03810-6_20.

[Bil95] P. Billingsley. *Probability and Measure*. John Wiley & Sons, 3rd edition, 1995.

[BS23] S. Ben-David and O. Sattah. Quantum tokens for digital signatures. *Quantum*, 7: 901, 2023.
DOI: 10.22331/q-2023-01-19-901.

[BSS21] A. Behera, O. Sattath, and U. Shinar. Noise-tolerant quantum tokens for MAC. E-print arXiv:2105.05016 [quant-ph], 2021.
arXiv: 2105.05016.

[BY23] A. Broadbent and P. Yuen. Device-independent oblivious transfer from the bounded-quantum-storage-model and computational assumptions. *New Journal of Physics*, 25: 053019, 2023.
DOI: 10.1088/1367-2630/accf32.

[CDF+19] M. Conde Pena, R. Durán Díaz, J.-C. Faugère, L. Hernández Encinas, and L. Perret. Non-quantum cryptanalysis of the noisy version of Aaronson-Christiano's quantum money scheme. *IET Information Security*, 13(4): 362–366, 2019.
DOI: 10.1049/iet-ifs.2018.5307.

[CHSH69] J. F. Clauser, M. A. Horne., A. Shimony, and R. A. Holt. Proposed experiment to test local hidden-variable theories. *Physical Review Letters*, 23(15): 880–884, 1969.
DOI: 10.1103/PhysRevLett.23.880.

[CLLZ21] A. Coladangelo, J. Liu, Q. Liu, and M. Zhandry. Hidden cosets and applications to unclonable cryptography. In *Advances in Cryptology — CRYPTO 2021*, volume 1, pages 556–584, 2021.
DOI: 10.1007/978-3-030-84242-0_20.

[CLRS22] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. The MIT Press, 4th edition, 2022.

[CV22] E. Culf and T. Vidick. A monogamy-of-entanglement game for subspace coset states. *Quantum*, 6: 791, 2022.
DOI: 10.22331/q-2022-09-01-791.

[CW79] J. L. Carter and M. N. Wegman. Universal classes of hash functions. *Journal of Computer and System Sciences*, 18(2): 143–154, 1979.
DOI: 10.1016/0022-0000(79)90044-8.

[DFR+07] I. B. Damgård, S. Fehr, R. Renner, L. Salvail, and C. Schaffner. A tight high-order entropic quantum uncertainty relation with applications. In *Advances in Cryptology — CRYPTO 2007*, pages 360–378, 2007.
DOI: 10.1007/978-3-540-74143-5_20.

[DFR20] F. Dupuis, O. Fawzi, and R. Renner. Entropy accumulation. *Communications in Mathematical Physics*, 379(3): 867–913, 2020.
DOI: 10.1007/s00220-020-03839-5.

[DFSS06] I. Damgård, S. Fehr, L. Salvail, and C. Schaffner. Oblivious transfer and linear functions. In *Advances in Cryptology — CRYPTO 2006*, pages 427–444, 2006.
DOI: 10.1007/11818175_26.

[Die82] D. Dieks. Communication by EPR devices. *Physics Letters A*, 92(6): 271–272, 1982.
DOI: 10.1016/0375-9601(82)90084-6.

[DNS10] F. Dupuis, J. B. Nielsen, and L. Salvail. Secure two-party quantum evaluation of unitaries against specious adversaries. In *Advances in Cryptology — CRYPTO 2010*, pages 685–706, 2010.
DOI: 10.1007/978-3-642-14623-7_37.

[DNS12] F. Dupuis, J. B. Nielsen, and L. Salvail. Actively secure two-party evaluation of any quantum operation. In *Advances in Cryptology — CRYPTO 2012*, pages 794–811, 2012.
DOI: 10.1007/978-3-642-32009-5_46.

[Dur19] R. Durrett. *Probability: Theory and Examples*. Cambridge University Press, 5th edition, 2019.

[EBB71] A. Einstein, M. Born, and H. Born. *The Born-Einstein letters: Correspondence between Albert Einstein and Max and Hedwig Born from 1916 to 1955*. Walker, 1971.

[FG06] J. Flum and M. Grohe. *Parametrized Complexity Theory*. Springer, 2006.

[FGH+10] E. Farhi, D. Gosset, A. Hassidimand, A. Lutomirski, D. Nagaj, and P. Shor. Quantum state restoration and single-copy tomography for ground states of Hamiltonians. *Physical Review Letters*, 105(19): 190503, 2010.
DOI: 10.1103/physrevlett.105.190503.

[FGH+12] E. Farhi, D. Gosset, A. Hassidim, A. Lutomirski, and P. W. Shor. Quantum money from knots. In *3rd Conference on Innovations in Theoretical Computer Science—ITCS 2012*, pages 276–289, 2012.
DOI: 10.1145/2090236.2090260.

[Fuj15] K. Fujii. *Quantum computation with topological codes: from qubit to topological fault-tolerance*. Springer Singapore, 2015.
DOI: 10.1007/978-981-287-996-7.

[GGH+13] S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *2013 54th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2013)*, pages 40–49, 2013.
DOI: 10.1109/FOCS.2013.13.

[Got97] D. Gottesman. *Stabilizer Codes and Quantum Error Correction*. PhD thesis, California Institute of Technology, 1997.
DOI: 10.7907/rzr7-dt72.

[GP21] R. Gay and R. Pass. Indistinguishability obfuscation from circular security. In *STOC 2021: Proceedings of the 53rd ACM SIGACT Symposium on Theory of Computing*, pages 736–749, 2021.
DOI: 10.1145/3406325.3451070.

[HEH+16] K. Heshami, D. G. England, P. C. Humphreys, P. J. Bustard, V. M. Acosta, J. Nunn, and B. J. Sussman. Quantum memories: emerging applications and recent advances. *Journal of Modern Optics*, 63(20): 2005–2028, 2016.
DOI: 10.1080/09500340.2016.1148212.

[HGL+23] D. Herman, C. Googin, X. Liu, Y. Sun, A. Galda, I. Safro, M. Pistoia, and Y. Alexeev. Quantum computing for finance. *Nature Reviews Physics*, 5: 450–465, 2023.
DOI: 10.1038/s42254-023-00603-1.

[HJL21] S. Hopkins, A. Jain, and H. Lin. Counterexamples to new circular security assumptions underlying iO. In *Advances in Cryptology — CRYPTO 2021*, volume 2, pages 673–700, 2021.
DOI: 10.1007/978-3-030-84245-1_23.

[Kan18] D. M. Kane. Quantum money from modular forms. E-print arXiv:1809.05925 [quant-ph], 2018.
arXiv: 1809.05925.

[KBD+21] Y. Kim, F. Bertagna, E. M. D'Souza, D. J. Heyes, L. O. Johannissen, E. T. Nery, A. Pantelias, A. Sanchez-Pedreño Jimenez, L. Slocombe, M. G. Spencer, J. Al-Khalili, G. S. Engel, S. Hay, S. M. Hingley-Wilson, K. Jeevaratnam, A. R. Jones, D. R. Kattnig, R. Lewis, M. Sacchi, N. S. Scrutton, S. R. P. Silva, and J. McFadden. Quantum biology: An update and perspective. *Quantum Reports*, 3(1): 80–126, 2021.
DOI: 10.3390/quantum3010006.

[Kil88] J. Kilian. Founding cryptography on oblivious transfer. In *STOC '88: Proceedings of the twentieth ACM symposium on Theory of computing*, pages 20–31, 1988.
DOI: 10.1145/62212.62215.

[KL15] J. Katz and Y. Lindell. *Introduction to Modern Cryptography*. Chapman & Hall/CRC, 2nd edition, 2015.

[KSS21] D. M. Kane, S. Sharif, and A. Silverberg. Quantum money from quaternion algebras. E-print arXiv:2109.12643 [quant-ph], 2021.
arXiv: 2109.12643.

[KST22] S. Kundu, J. Sikora, and E. Y.-Z. Tan. A device-independent protocol for XOR oblivious transfer. *Quantum*, 6: 725, 2022.
DOI: 10.22331/q-2022-05-30-725.

[Kum19] N. Kumar. Practically feasible robust quantum money with classical verification. *Cryptography*, 3(4): 26:1–26:24, 2019.
DOI: 10.3390/cryptography3040026.

[KW16] J. Kaniewski and S. Wehner. Device-independent two-party cryptography secure against sequential attacks. *New Journal of Physics*, 18(5): 55004, 2016.
DOI: 10.1088/1367-2630/18/5/055004.

[LAF+10] A. Lutomirski, S. Aaronson, E. Farhi, D. Gosset, J. A. Kelner, A. Hassidim, and P. W. Shor. Breaking and making quantum money: Toward

a new quantum cryptographic protocol. In *Innovations in Computer Science—ICS 2010*, pages 20–31, 2010.
Online: https://arxiv.org/abs/0912.3825.

[LC97] H.-K. Lo and H. F. Chau. Is quantum bit commitment really possible? *Physical Review Letters*, 78(17): 3410–3413, 1997.
DOI: 10.1103/PhysRevLett.78.3410.

[LCC+13] N. Lambert, Y.-N. Chen, C.-Y. Cheng, C.-M. Li, G.-Y. Chen, and F. Nori. Quantum biology. *Nature Physics*, 9: 10–18, 2013.
DOI: 10.1038/nphys2474.

[LG12] F. Le Gall. Quantum private information retrieval with sublinear communication complexity. *Theory of Computing*, 8(16): 369–374, 2012.
DOI: 10.4086/toc.2012.v008a016.

[LH19] M. Liu and Y. Hu. Universally composable oblivious transfer from ideal lattice. *Frontiers of Computer Science*, 13(4): 879–906, 2019.
DOI: 10.1007/s11704-018-6507-4.

[LMZ23] J. Liu, H. Montgomery, and M. Zhandry. Another round of breaking and making quantum money: How to not build it from lattices, and more. In *Advances in Cryptology — EUROCRYPT 2023*, pages 611–638, 2023.
DOI: 10.1007/978-3-031-30545-0_21.

[Lut10] A. Lutomirski. An online attack against Wiesner's quantum money. E-print arXiv:1010.0256 [quant-ph], 2010.
arXiv: 1010.0256.

[Mah18] U. Mahadev. Classical verification of quantum computations. In *2018 59th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2018)*, pages 259–267, 2018.
DOI: 10.1109/FOCS.2018.00033.

[May97] D. Mayers. Unconditionally secure quantum bit commitment is impossible. *Physical Review Letters*, 78(17): 3414–3417, 1997.
DOI: 10.1103/PhysRevLett.78.3414.

[MDCA21] T. Metger, Y. Dulek, A. Coladangelo, and R. Arnon-Friedman. Device-independent quantum key distribution from computational assumptions. *New Journal of Physics*, 23: 123021, 2021.
DOI: 10.1088/1367-2630/ac304b.

[Mit05] M. Mitzenmacher. *Probability and computing: randomized algorithms and probabilistic analysis*. Cambridge University Press, 2005.

[MV21] T. Metger and T. Vidick. Self-testing of a single quantum device under computational assumptions. *Quantum*, 5: 544, 2021.
DOI: 10.22331/q-2021-09-16-544.

[MVW13] A. Molina, T. Vidick, and J. Watrous. Optimal counterfeiting attacks and generalizations for Wiesner's quantum money. In *Theory of Quantum Computation, Communication, and Cryptography (TQC 2012)*, pages 45–64, 2013.
DOI: 10.1007/978-3-642-35656-8_4.

[NC10] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 10th anniversary edition, 2010.
DOI: 10.1017/CBO9780511976667.

[NSBU16] D. Nagaj, O. Sattath, A. Brodutch, and D. Unruh. An adaptive attack on Wiesner's quantum money. *Quantum Information & Computation*, 16(11&12): 1048–1070, 2016.
DOI: 10.26421/QIC16.11-12-7.

[Pei16] C. Peikert. A decade of lattice cryptography. *New Foundations and Trends in Theoretical Computer Science*, 10(4): 283–424, 2016.
DOI: 10.1561/0400000074.

[PFP15] M. C. Pena, J.-C. Faugère, and L. Perret. Algebraic cryptanalysis of a quantum money scheme: the noise-free case. In *Public-Key Cryptography – PKC 2015*, pages 194–213, 2015.
DOI: 10.1007/978-3-662-46447-2_9.

[PVW08] C. Peikert, V. Vaikuntanathan, and B. Waters. A framework for efficient and composable oblivious transfer. In *Advances in Cryptology — CRYPTO 2008*, pages 554–571, 2008.
DOI: 10.1007/978-3-540-85174-5_31.

[PYJ+12] F. Pastawski, N. Y. Yao, L. Jiang, M. D. Lukin, and J. I. Cirac. Unforgeable noise-tolerant quantum tokens. *Proceedings of the National Academy of Sciences of the United States of America*, 109(40): 16079–16082, 2012.
DOI: 10.1073/pnas.1203552109.

[Reg05] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. In *STOC '05: Proceedings of the thirty-seventh ACM symposium on Theory of computing*, pages 84–93, 2005.
DOI: 10.1145/1060590.1060603.

[Rén61] A. Rényi. On measures of entropy and information. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 547–561, 1961.

[Ren05] R. Renner. Security of quantum key distribution. *International Journal of Quantum Information*, 06(01): 1–127, 2005.
DOI: 10.1142/S0219749908003256.

[Rob21] B. Roberts. Security analysis of quantum lightning. In *Advances in Cryptology — EUROCRYPT 2021*, pages 562–567, 2021.
DOI: 10.1007/978-3-030-77886-6_19.

[Rof19] J. Roffe. Quantum error correction: an introductory guide. *Contemporary Physics*, 60(3): 226–245, 2019.
DOI: 10.1080/00107514.2019.1667078.

[RTK+18] J. Ribeiro, L. P. Thinh, J. Kaniewski, J. Helsen, and S. Wehner. Device-independence for two-party cryptography and position verification with memoryless devices. *Physical Review A*, 97(6): 062307, 2018.
DOI: 10.1103/PhysRevA.97.062307.

[RW04] R. Renner and S. Wolf. Smooth Rényi entropy and applications. In *2004 IEEE International Symposium on Information Theory (ISIT 2004)*, page 233, 2004.
DOI: 10.1109/ISIT.2004.1365269.

[RW05] R. Renner and S. Wolf. Simple and tight bounds for information reconciliation and privacy amplification. In *Advances in Cryptology — ASIACRYPT 2005*, pages 199–216, 2005.
DOI: 10.1007/11593447_11.

[RW20] J. Ribeiro and S. Wehner. On bit commitment and oblivious transfer in measurement-device independent settings. E-print arXiv:2004.10515 [quant-ph], 2020.
arXiv: 2004.10515.

[SCA+11] J. Silman, A. Chailloux, N. Aharon, I. Kerenidis, S. Pironio, and S. Massar. Fully distrustful quantum bit commitment and coin flipping. *Physical Review Letters*, 106(22): 220501, 2011.
DOI: 10.1103/PhysRevLett.106.220501.

[Sch07] C. Schaffner. *Cryptography in the Bounded-Quantum-Storage Model*. PhD thesis, University of Aarhus, 2007.

[SSS09] L. Salvail, C. Schaffner, and M. Sotáková. On the power of two-party quantum cryptography. In *Advances in Cryptology — ASIACRYPT 2009*, pages 70–87, 2009.
DOI: 10.1007/978-3-642-10366-7_5.

[SW14] A. Sahai and B. Waters. How to use indistinguishability obfuscation: deniable encryption, and more. In *STOC '14: Proceedings of the fourty-sixth ACM symposium on Theory of computing*, pages 475–484, 2014.
DOI: 10.1145/2591796.2591825.

[TL17] M. Tomamichel and A. Leverrier. A largely self-contained and complete security proof for quantum key distribution. *Quantum*, 1: 14, 2017.
DOI: 10.22331/q-2017-07-14-14.

[Unr13] D. Unruh. Everlasting multi-party computation. In *Advances in Cryptology — CRYPTO 2013*, pages 380–397, 2013.
DOI: 10.1007/978-3-642-40084-1_22.

[Wat08] J. Watrous. Quantum computational complexity. E-print arXiv:0804.3401 [quant-ph], 2008.
DOI: 10.48550/arXiv.0804.3401.

[Wat18] J. Watrous. *The Theory of Quantum Information*. Cambridge University Press, 2018.
DOI: 10.1017/9781316848142.

[Wie83] S. Wiesner. Conjugate coding. *ACM SIGACT News*, 15(1): 78–88, 1983.
DOI: 10.1145/1008908.1008920.

[Wil13] M. M. Wilde. *Quantum Information Theory*. Cambridge University Press, 2nd edition, 2013.

[WLQ+21] P. Wang, C.-Y. Luan, M. Qiao, M. Um, J. Zhang, Y. Wang, X. Yuan, M. Gu, J. Zhang, and K. Kim. Single ion qubit with estimated coherence time exceeding one hour. *Nature Communications*, 12(1): 233:1–233:8, 2021.
DOI: 10.1038/s41467-020-20330-w.

[WW21] H. Wee and D. Wichs. Candidate obfuscation via oblivious LWE sampling. In *Advances in Cryptology — EUROCRYPT 2021*, pages 127–156, 2021.
DOI: 10.1007/978-3-030-77883-5_5.

[WZ82] W. K. Wootters and W. H. Zurek. A single quantum cannot be cloned. *Nature*, 299: 802–803, 1982.
DOI: 10.1038/299802a0.

[Yue25] P. Yuen. Noise-tolerant public-key quantum money from a classical oracle. *Quantum*, 9: 1691, 2025.
DOI: 10.22331/q-2025-04-07-1691.

[Zha19] M. Zhandry. Quantum lightning never strikes the same state twice. In *Advances in Cryptology — EUROCRYPT 2019*, volume 3, pages 408–438, 2019.
DOI: 10.1007/978-3-030-17659-4_14.

[Zha21] M. Zhandry. Quantum lightning never strikes the same state twice. Or: Quantum money from cryptographic assumptions. *Journal of Cryptology*, 34(1): 6:1–6:56, 2021.
DOI: 10.1007/s00145-020-09372-x.