이학박사 학위논문

Cryptographic Algorithms for Quantum Computations

(양자 컴퓨터에 대한 암호학적 알고리즘)

2022년 8월

서울대학교 대학원

수리과학부

한 민 기

Cryptographic Algorithms for Quantum Computations

(양자 컴퓨터에 대한 암호학적 알고리즘)

지도교수 이 훈 희

이 논문을 이학박사 학위논문으로 제출함

2022년 4월

서울대학교 대학원

수리과학부

한 민 기

한 민 기의 이학박사 학위논문을 인준함

2022년 6월

위 원 장 김 태 현 (인)

부 위 원 장 이 훈 희 (인)

위 원 이 수 준 (인)

위 원 김 재 완 (인)

위 원 윤 아 람 (인)

Cryptographic Algorithms for Quantum Computations

A dissertation
submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
to the faculty of the Graduate School of
Seoul National University

by

Hhan, Minki

Dissertation Director : Professor Hun Hee Lee

Department of Mathematical Sciences
Seoul National University

August 2022

Abstract

Cryptographic Algorithms for Quantum Computations

Hhan, Minki

Department of Mathematical Sciences
The Graduate School
Seoul National University

The advent of a quantum mechanical computer presents a clear threat to existing cryptography. On the other hand, the quantum computer also suggests the possibility of a new cryptographic protocol through the properties of quantum mechanics. These two perspectives, respectively, gave rise to a new field called post-quantum cryptography as a countermeasure against quantum attacks and quantum cryptography as a new cryptographic technology using quantum mechanics, which are the subject of this thesis.

In this thesis, we reconsider the security of the current post-quantum cryptography through a new quantum attack, model, and security proof. We present the fine-grained quantum security of hash functions as cryptographic primitives against preprocessing adversaries. We also bring recent quantum information theoretic research into cryptography, creating new quantum public key encryption and quantum commitment. Along the way, we resolve various open problems such as limitations of quantum algorithms with preprocessing computation, oracle separation problems in quantum complexity theory, and public key encryption using group action.

Key words: Quantum computer, Cryptography, Quantum algorithm, Random oracle model, Public-key encryption, Bit commitment

Student Number: 2016-20255

Contents

Chapter 1

Introduction

What would be the impact of quantum computers on cryptography? The most influential results for this question are due to Grover [Gro96] and Shor [Sho99]. The quantum search algorithm of Grover states that the estimated security, especially of symmetric-key cryptosystems, based on the exhaustive search will be reduced by half for quantum adversaries. Shor's algorithm says that quantum computers will break down some of the currently used public-key cryptography based on the integer factorization and discrete logarithm. These papers argue that we should move to the cryptographic schemes secure against quantum computations, called the *post-quantum cryptography*.

The response of cryptographic designers and governments to these attacks is rather straightforward, though based on careful analysis and discussions from the series of research. Roughly speaking, the decision is to double the length of symmetric-key cryptography [Ber09, PC09] and use the new public-key systems based on alternative assumptions; let us call this modification the "naive fix". The naive fix could seem reasonable as follows. The key-length doubling of symmetric-key cryptography makes the security against the naive application of the Grover algorithm be the same as the security of the original system against the exhaustive key search. On the other hand, the alternative cryptographic assump-

tions such as lattice or coding problems have some confidence in their security against quantum computations [Ajt98, Mic01], leading us to the post-quantum public-key cryptography [NIS]——the end of the story.

Unfortunately, our life is not that simple. The above perspective oversimplifies the efforts of cryptanalysts who try to find the vulnerability. Cryptography has been the history of the war between cryptographic designers and cryptanalysts, and the most recent battle seems to be won by the cryptanalysts. For symmetric-key cryptography, recent quantum attack papers suggested that the naive fix of constructions is much weaker than we expected—beyond quadratic speed-up [BSS22]—and some classically secure schemes turn out to be quantum insecure at all [ATTU16, KLLNP16]! For public-key cryptography, a line of quantum algorithms has found that the cryptographic assumptions believed to be post-quantum have some potential quantum weaknesses not found by the classical attacks [GKZ19, CLZ22]. Furthermore, a recent paper shows that a recently suggested that a candidate post-quantum cryptography called SIDH [JF11], with NIST PQC standardization candidate SIKE [ACC+17], is not even secure against classical polynomial time attack, questioning the confidence of new assumptions.

On the positive side, quantum computers could also introduce new abilities to cryptographic designers. Two papers [BB84, Wie83] have initiated this direction, by showing that some classically impossible tasks can be done through quantum channels. More recently, the quantum state's unique properties, represented by the no-cloning theorem, have provided new research directions called *quantum cryptography* with new exciting cryptographic schemes. For example, quantum money schemes [Aar09] suggest using the quantum state as the money because they are unclonable, and the blockchain-less cryptocurrency [Zha21] could be obtained by improving them. Unclonable-ciphertext encryptions [BL20] and quantum programs [ALL+21] (called unclonable encryption and quantum copy-protection, respectively) are also intriguing.

Quantum cryptography turns out to provide applications beyond cryptography.

The recent quantum supremacy experiments [AAB+19, ZWD+20] have shown the potential strength of quantum computers over classical computers. However, the hardness of classical simulation relies on newly suggested assumptions [AA11, AC17], which need more analysis, and the verification process of quantum advantage is currently the statistical test, sometimes invalidated by some tailor-made classical algorithms [BCG21, LLL+21]. The cryptographic tools resolve this state of affairs by showing the construction of the provably verifiable quantum supremacy based on the standard cryptographic assumptions [BKVV20, BCM+21, YZ21, YZ22]. Furthermore, much harder tasks including the *classical verification* of quantum computation [Mah18, CCY20] are resolved based on the cryptographic techniques. The ideas from cryptography have influenced on the depth-efficient protocols [LG21, HLG21] and a recent complex-theoretic breakthrough $\mathsf{MIP}^* = \mathsf{RE}$ [JNV+21].

This age of developing quantum computers leads to new opportunities for both cryptographic scheme designers and attackers. In particular, the following important problems are of interest to both, and should be addressed for our secure cyber life in the quantum era.

How to model the quantum adversary properly?
How to show the limitation of quantum adversary?
Can we find new quantum algorithms for cryptographic design or attacks?

1.1 Contributions

The topics in this thesis center on discovering the new powers and limitations of quantum computations in the context of cryptography. We mainly focus on two situations; the first case is the world where only adversaries have the quantum power, usually called the *post-quantum cryptography*. In the second world, the cryptographic designers also can use the quantum computation as well, which is we call the *quantum cryptography*.

1.1.1 Part I: Post-quantum Cryptography

In the first part of this thesis, we focus on the adversary's power and limit in post-quantum cryptography. We first argue that modeling quantum adversaries in post-quantum cryptography should be more carefully done by providing two new quantum attacks. In particular, our first attack shows the preprocessing quantum attack could be much better than the simple quantum attack based on Grover's algorithm. This result alarms the current security estimation of symmetric key cryptography makes reminiscent of the impact of Hellman's attack [Hel80] on DES [S+99].

We then present new models called *quantum random oracle model with (quantum) auxiliary input* by generalizing [Unr07, BDF+11a] for preprocessing quantum adversaries. We provide the basic security proofs in the new models showing that we can use the hash functions as basic cryptographic primitives in the random oracle model.

Along the way, we prove the first limitation of quantum algorithm with quantum advice for inversion problem, resolving an open question asked by Nayebi, Aaronson, Belovs, and Trevisan [NABT15]. This result also has a complexity-theoretic implication, answering an open problem posed by Aaronson [Aar05].

Attack 1: Function inversion problem. The function inversion problems for $f : [N] \to [N]$[1] ask to invert the image $y = f(x)$ of random input x. The cryptographic hash functions [ANWOW13, D+15, BDPA13] and symmetric key encryptions [DR02] are particular instances in which the function inversion problems are known to be (optimally) hard. The best classical and quantum function inversion algorithms for those functions are the exhaustive search and the Grover algorithm, respectively.

We show a new quantum function inversion algorithm using a classical advice.

[1]The domain and range of function could be different in practice. Here and below, we assume they are the same for a more straightforward exposition.

We extend and formally analyze a recent quantum time-space trade-off [DKRS21], which is a quantum version of classical function inversion algorithm [Hel80] based on the Grover search.

Our algorithm performs better than the Grover algorithm if the advice size is sufficiently large, showing the naive fix may fail in theory. Hopefully, the required advice size for a better-than-Grover attack is enormously large for the practical parameter setting.

The new quantum algorithm can be applied to an arbitrary function and is applicable to small success probability range thanks to the rigorous analysis following [FN00], and several technical tools from [DTT10]. Note that this algorithm has applications to the systematic substring search problem and 3-SUM indexing problems following [CGK18, GGH+20].

Attack 2: Quantum multiple discrete logarithm problem. The second problem we consider is the multiple discrete logarithm problem, where the m different discrete logarithm instances g^{x_i} for $i = 1, ..., m$ are given for the group G with generator g, and ask to find $x_1, ..., x_m$ simultaneously. The classical best algorithm takes $O(\sqrt{m|G|})$ group operations [KS01] and is known to be optimal [Yun15]. On the other hand, Shor's algorithm [Sho99] requires about $2 \log |G|$ group operations per instance so that the discrete logarithm problem is insecure in the quantum world.

We present a new quantum algorithm for the multiple discrete logarithm problem faster than applying the Shor algorithm multiple times, showing the large quantum computer would make a more substantial threat to cryptography. Based on the multi-exponentiation algorithms [Pip80, LL94], we show that the m instances of discrete logarithm problem can be solved by using $O(\log |G| / \log m)$ group operations per instance, showing an asymptotic better performance. For a practical parameter setting $m = \log |G| = 512$, the amortized group operation is reduced by 80% compared to the naive Shor's algorithm.

The downside of our algorithm is a large amount of quantum memory, which makes the algorithm requires much stronger devices. Still, considering the current estimation of optimized quantum discrete logarithm algorithm [Eke21, EH17] of few hours for 1024-bit groups [GE21], our faster multiple discrete logarithm algorithm alarms again on the impact of quantum computation over cryptography, especially when they become much larger.

Limit of quantum algorithms with classical advice. We complement the above attacks by showing the limitation of a quantum attack model for cryptographic hash functions, encompassing the above attacks. The researches on hash functions have shown the limit of adversaries for hash functions in various applications [BR93, BZ13, DGK17, CDGS18], but they only deal with the quantum adversaries or the preprocessing adversaries separately. To remedy this, we introduce a new model, the *quantum random oracle model with auxiliary input* (QROM-AI), combining the quantum random oracle model [BDF+11a] and the random oracle with auxiliary input model [Unr07].

In the QROM-AI, the cryptographic hash functions are modeled as a truly random function (following [BR93]). The adversaries have quantum oracle access the random function, and also take a preprocessing classical data string of the function with a bounded length.

We show there are the limitations of the adversary's advantages in the QROM-AI for various basic applications. More precisely, we show that the cryptographic hash functions, modeled as random functions $H : [N] \to [N]$, can be used for the following cryptographic primitives in the QROM-AI, that is, show the lower bounds of the success probability of the QROM-AI algorithms, provided the upper bound of the query number of adversary and the size of advice string.

- The one-way functions; given $y = H(x)$ for random x, no QROM-AI algorithm can efficiently find x' such that $H(x') = H(x)$.

- The pseudorandom generators; no QROM-AI algorithm can efficiently distinguish the hash evaluation $y = H(x)$ from the truly random value r.

Our lower bounds further include the applications for post-quantum pseudorandom functions and message authentication codes, where the adversary only has the classical oracle access to the cryptographic applications, while has the quantum oracle access to the base random functions. We extend the result to the salted random oracles [MT79], a practical countermeasure of the dictionary attack in the context of password hashing. Our result suggests that the salted random oracles enhance the security of basic applications of hash functions.

The compression lemma [GT00, GGKT05] lies at the center of our proof, following the classical preprocessing attack lower bound proofs [DTT10, DGK17]. That is, we construct the encoding-decoding scheme for random functions based on the adversaries for hash functions; the lower bound of encoding scheme implies the lower bound of the adversary. Our proofs are inspired by [DGK17] in many parts, but require a number of new idea such as the one-way-to-hiding lemma and its variants [Unr15, BHH+19], semi-classical oracles [AHU19] and some quantum tools such as the amplitude amplification [BHMT02].

Limit of quantum algorithms with quantum advice. Next, we further extend the lower bound of preprocessing algorithm to the *quantum advice* setting. While there are no known quantum advice algorithms for cryptographic applications, the complexity-theoretic studies [Wat00, NY04, Aar05] have shown the advantage of quantum advice over classical one, especially showing the oracle separation between $\mathsf{BQP/poly}$ and $\mathsf{BQP/qpoly}$ [AK07]

Our result focuses on the random permutations, proving the hardness of the function inversion problem against the quantum advice algorithms. This resolves an open problem posed by Nayebi et al [NABT15]. To extend the quantum advice setting, we employ the quantum compression lemma derived from [Nay99, NS06], and the gentle measurement lemma [Win99, Aar05, AR19].

The techniques used in our proof introduce some subtleties for the problems having many different answers, leading us to focus on the random permutations rather than random functions. Thus our results do not have the cryptographic applications; instead, we give the following complexity-theoretic implication.

Complexity-theoretic consequence. Using the hardness of permutation inversion, we obtain the first oracle separation between $\mathsf{NP} \cap \mathsf{coNP}$ and $\mathsf{BQP/qpoly}$, answering an open problem posed by Aaronson [Aar05]. We consider the decision problem for input (y, z) and a permutation P, asking if $P^{-1}(y) \leq z$ or not. The proof resembles the oracle separation proof of BQP and $\mathsf{NP} \cap \mathsf{coNP}$ [BBBV97].

1.1.2 Part II: Quantum Cryptography

In the second part of this thesis, we explore the potential applications of quantum mechanics in cryptography. Our main idea is to bring a recent theorem of Aaronson, Atia, and Susskind [AAS20] into cryptographic applications. Interestingly, their original motivation was the foundation of quantum mechanics and quantum gravity.

Based on the generalization of the equivalence theorem, we construct new quantum public key encryption (PKE) schemes with new structures and obtain a new conversion theorem of quantum bit commitments. In particular, we construct the first PKE scheme based on cryptographic group action, partly resolving an open question of [JQSY19]. On the other hand, our commitment conversion theorem is optimal in the number of queries to base scheme, outperforming the previous conversions [CLS01, Yan20].

Equivalence theorem. We first explain our main ingredient, the equivalence theorem, via the experiment of Schrödinger's cat. Roughly speaking, the equivalence theorem states that to distinguish between two states $|\text{Alive}\rangle \pm |\text{Dead}\rangle$ is as

hard as to revive the cat, i.e., to bring |Dead⟩ to |Alive⟩. In other words, to figure out the phase of quantum state of cat is a *necromancy-hard* problem.

This equivalence is in fact a folklore result in quantum mechanics, but the formalized analysis is recently done by [AAS20] through the notions of quantum circuit complexity. We extend their results to the advice algorithm setting, and observe some new properties of equivalence for our purpose. As we will see below, we interpret the equivalence theorem as a search-to-decision reduction that has an important role in cryptography.

Application 1: Quantum-ciphertext Public-key Encryptions. The first application of the equivalence theorem is public key encryption (PKE). A PKE scheme is a cryptographic object enabling any user to encrypt their message using a publicly announced key. Still, the encrypted message is hidden from all but one who holds the corresponding secret key.

We present a new abstraction of *swap-trapdoor function pairs*, and provide a construction of PKE based on swap-trapdoor function pairs and using the equivalence theorem above. Since this construction is based on the equivalence theorem for quantum states, the resulting PKE schemes have quantum ciphertexts.

We provide several relations between swap-trapdoor function pairs and trapdoor claw-free function pairs, and show that the (noisy) trapdoor claw-free function families presented in [BCM+21, BKVV20] based on (ring-)LWE are also swap-trapdoor functions. Further, we construct swap-trapdoor functions based on the cryptographic group actions, recently suggested in [JQSY19] as a new cryptographic assumption.

Our two main PKE constructions are 1) PKE based on (nonabelian) cryptographic group action, which resolves an open problem suggested in [JQSY19], and 2) lattice-based PKE that has an additive homomorphic property without any additional noise. To our knowledge, this is the first lattice-based additive homomorphic encryption without computation errors, as all previous lattice-based

constructions, e.g., [Reg09], suffers from the additive noise. The homomorphic computation is inspired by the dihedral hidden subgroup algorithm of Kuperberg [Kup05].

Application 2: Quantum bit commitments. Finally, we focus on quantum bit commitment schemes. Commitments are the interactive protocol between the committer and the receiver. It enable the committer to *commit* to a (classical) bit[2] in such a way that the committed bit is hidden from the receiver before the committer reveals it—the *hiding* property—and the committer cannot change the committed bit after sending the commitment—the *binding* property. The impossibility to achieve both hiding and binding properties against unbounded-time adversaries (called *statistical* hiding and binding, respectively) is shown even if we allow quantum communication [LC97, May97].

A common practice in cryptography, therefore, is to relax either of them to hold only against computationally bounded adversaries, and there are the two *flavors* of commitments: One is computationally hiding and statistically binding, and the other is computationally binding and statistically hiding, which is called in this thesis by the binding and hiding commitment, respectively.

A unique feature of quantum bit commitment, where the quantum channels are used in the interactions, is *the efficient conversion* between two flavors [Yan20, CLS01]. On the other hand, the *efficient* conversion between flavors for classical commitments is not known; both flavors of the commitments are equivalent to the existence of one-way functions [HILL99, HR07, Nao91], but the conversion through this equivalence is extremely expensive. Therefore, the study of hiding and binding commitments has been done separately; for example, the efficient binding commitment from one-way functions is possible [HILL99, Nao91] and the hiding commitment from the collision-resistant hash functions is practi-

[2]The commitment for a quantum bit can be obtained combining the classical bit commitment and quantum one-time pad [AMTDW00] as in [BJ15].

cal [HM96]. However, their counterparts have no efficient constructions; indeed, the hiding commitment from one-way functions may require the polynomial number of rounds [HHRS15].

Our main result for quantum bit commitment is the new compiler that converts the flavors; hiding and binding. This compiler is extremely simple and efficiency-preserving; it only calls the base scheme once and uses a constant number of quantum gates whereas the previous compilers requires at least $\Omega(\lambda^2)$ calls for the security parameter λ [Yan20, CLS01]. The resulting commitment can be considered as a dual commitment, meaning that if we apply the conversion twice, we have the original commitment.

We obtain a number of new quantum bit commitment schemes based on the new compiler and some new constructions. The following hiding commitment schemes are the *efficient* constructions

- using only a single call to the PRGs, pseudorandom state generators, or the collapsing hash functions, and
- with a shorter commitment length from the injective OWF.

Note that all of the results are not entirely new if we neglect the efficiency, as the previous conversions can be applied as well.

1.2 Related Works

There are a number of follow-up work for our quantum preprocessing algorithm lower bounds. Chung, Liao, Qian studied the function inversion problem with quantum advice, showing a similar bound for random functions to our random permutation bound [CLQ20] based on the similar idea to ours. Later, they together with Guo showed a much tighter bound for the function inversion problem [CGLQ20] using the multi-instance games inspired by [Aar05]. The same

bound is reproved by the presampling technique [GLLZ21], generalizing the classical preprocessing techniques [Unr07, CDGS18]. We remark that the incompressibility arguments also used to prove the black-box separation between cryptographic primitives [HY20, CX21] in the quantum world.

The quantum time-memory trade-offs for the function inversion problem is independently studied by [DKRS21], which is the starting point of our algorithm. However, their result is only applicable to random functions and based on some heuristic assumptions (as in [Hel80]). On the other hand, we rigorously analyze the use of independent hash functions following [FN00, DTT10] and obtain the algorithm for *any* functions.

A line of research has improved the efficiency of Shor algorithm, especially for the short discrete logarithm problems [Eke21, EH17, Eke20], resulting in the better estimated time complexity of 8 hours to factorize 2048-bit RSA integers [GE21][3]. A quantum memory-time trade-off was suggested in [BBM17], but they focus on the low-resource case, and our results take a different direction.

The name of quantum PKE was used in multiple times with different meanings. Note that our quantum PKE is the PKE scheme that only the encryption and ciphertexts are quantum. In [OTU00], the key generation *algorithm* is only the quantum part. On the other hand, in [KKNY05], the quantum PKE means the quantum ciphertexts and *quantum public keys,* thus their primitive is weaker than ours.

The quantum commitment schemes recently have shown several applications, especially with the statistical binding property. Yan et al. [YWLQ15] and Fang et al. [FUYZ20] showed the quantum binding commitments can be used for constructing the zero-knowledge proofs and quantum oblivious transfers, with the game-based security. Morimae and Yamakawa [MY21] show that the simulation-secure quantum oblivious transfer (and multi-party computation [BCKM21]) with

[3]Note that the initial experimental results [VSB+01] have oversimplified the Shor algorithm to make the practical implementation as shown in [SSV13].

the simulation-based security can be constructed from the binding commitment based on the observations of [Yan20, AQY21]. The frameworks of binding properties and canonical commitment developed in [FUYZ20, Yan20] supported the above applications, thus we expect the canonical binding commitments are as useful as the classical binding commitments in the future.

We note that the computational notion of binding property is less understood. In fact, there are many different notions of binding, especially for the computational setting. Unruh [Unr16] suggested the collapse-binding property, and Bitansky and Brakerski [BB21] recently introduced the classical binding of quantum commitment. The definitions of binding in [CDMS04, DFS04, AQY21] are all different and have different aspects.

1.3 Research Papers Contained in This Thesis

The results in this thesis are based on the following papers.

- [Hha22] *"A quantum time-memory trade-off for inverting any function (working title)"* (preprint). Section 3.2 is based on the result of this paper.
- [HY22] *"On Quantum Multiple Discrete Logarithm Problem (working title)"*, with Aaram Yun (preprint). The contents of Section 3.3 is taken from this paper.
- [HXY19] *"Quantum Random Oracle Model with Auxiliary Input"*, with Takashi Yamakawa and Keita Xagawa (in Asiacrypt 2019). Most of results in Chapter 4 and Chapter 5 are based on this paper.
- [HMY22] *"Efficiency-Preserving Conversion for the Flavor of Quantum Bit Commitments (working title)"*, with Tomoyuki Morimae and Takashi Yamakawa (preprint). The results in the second part of this thesis (Chapters 6 to 8) are based on this paper.

Chapter 2

Preliminaries

Notations and conventions. For a positive integer n, we denote a set $\{1, \ldots, n\}$ by $[n]$. We use the Landau notations $O(f(n))$ and $\Omega(f(n))$ and the tilde notations $\widetilde{O}(f(A, B, \cdots))$ or $\widetilde{\Omega}(f(A, B, \cdots))$, where we ignore non-negative degree polylogarithmic factors with respect to all capital variables which appear in the context. For example, we write $(T^2/N) \cdot \log M = \widetilde{O}(T^2/N)$. To denote the event that a (possibly probabilistic or quantum) algorithm A with input z outputs x, we write $A(z) \to x$.

The character λ denotes the security parameter. Most of other parameters, functions and schemes in this thesis are implicitly parameterized by λ. For example, when we say "a function $f : \{0,1\}^n \to \{0,1\}^m$ ...", we implicitly consider a family of functions $F = \{f_1, f_2, ...\}$ such that $f_\lambda : \{0,1\}^{n(\lambda)} \to \{0,1\}^{m(\lambda)}$. We say a function $\varepsilon(n)$ is negligible if $\varepsilon(n) < 1/|p(n)|$ for any polynomial p for sufficiently large n. When $n = \lambda$, we omit the parameter of polynomial and just write negl to denote a negligible function.

We write $U^\dagger$ to denote the conjugate transpose of matrix U. The ℓ_2-norm of vectors and quantum states is denoted by $\|\cdot\|$.

2.1 Quantum Computations

Quantum algorithms have intrinsic randomness when they perform measurements. The probability that a quantum algorithm A outputs x on an input z is denoted by $\Pr_A[A(z) \to x]$. To denote quantum objects such as quantum states or a quantum-accessible oracle, we use the ket notation $|\cdot\rangle$ or calligraphic letters. For example, $|\phi\rangle$ and $\mathcal{O}$ denote a quantum state and a quantum-accessible oracle, while x, z are classical strings. For basics of quantum computing, we refer readers to [NC00]. We say functions, algorithms, and unitary maps are polynomial-time computable, or simply efficient, if there is a polynomial time algorithm to compute the functions, algorithms, and unitaries given any input x. We fix an arbitrary set of universal gates, which is not relevant to the results of this thesis.

2.1.1 Quantum oracle algorithms

A quantum oracle algorithm is a quantum algorithm that can perform quantum computations and can access oracles. In this thesis, we consider three types of oracles: quantum-accessible oracle, classical-accessible oracle, and semi-classical oracle [AHU19], which is explained below.

Following [BBC+01], an oracle algorithm that accesses an oracle $\mathcal{O}$ at most T times are modeled by a sequence of unitary transforms

$$U_0, \mathcal{O}, U_1, \mathcal{O}, \cdots, \mathcal{O}, U_T$$

where the unitary transform $\mathcal{O} : \{0,1\}^n \to \{0,1\}^m$ is associated with a map from basis state $|x, y\rangle$ to $|x, y + \mathcal{O}(x)\rangle$. Here, we abuse the notation $\mathcal{O}$ by identifying the oracle and corresponding unitary transform. If the oracle $\mathcal{O}$ only admits classical queries, the first n bits should be classical bits before applying the unitary transform $\mathcal{O}$. This can be done by measuring the first register in the computational basis before making query.

If the oracle is quantum-accessible, there is no such restriction. For an algorithm with quantum-accessible oracle O that computes f, the query to oracle is of the form

$$\sum \alpha_{x,y,z}|x,y,z\rangle \to \sum \alpha_{x,y,z}|x,y+f(x),z\rangle.$$

We often use $A^{|f\rangle}$ to mean that A accesses a quantum-accessible oracle that computes f and A^f to mean that A accesses classical-accessible oracle that computes f. We allow a quantum oracle algorithm to make queries in parallel. Its query depth d is defined to be the number maximal sequential call of oracle queries.

2.1.2 One-way to hiding lemmas

Semi-classical oracles. We review *semi-classical oracles*, which is introduced in [AHU19]. Here, we only define a semi-classical oracle for the indicator function of a set S since we only need it in this thesis. An indicator semi-classical oracle O_S^{SC} for a set $S \subseteq X$ is queried with two registers, an input register Q with $\mathbb{C}^X$ and an output register R with space $\mathbb{C}^2$. When queried with a value $|x\rangle$ in Q, the oracle returns whether $x \in S$ in the output register R. More formally, it performs a measurement with projectors M_0 and M_1, where $M_0 := \sum_{x\in X\setminus S} |x\rangle\langle x|$ and $M_1 := \sum_{x\in S} |x\rangle\langle x|$, and initializes R to $|0\rangle$ or $|1\rangle$ corresponding to the measurement result.

In the execution of an algorithm $A^{O_S^{SC}}$, the flag Find denotes the event that O_S^{SC} returns $|1\rangle$ occurs. This event is a well-defined classical event since O_S^{SC} *measures* its outputs.

Punctured oracle. If H is an oracle with domain X and codomain Y, we define $|H\rangle \setminus S$ as an oracle for $S \subset X$ which, on input x, first queries $O_S^{SC}(x)$ and then queries $H(x)$. The following lemma states that the outcome of $A^{|H\rangle\setminus S}$ is independent of $\{H(x) : x \in S\}$ when Find does not occur.

Lemma 2.1.1 (Punctured Oracle [AHU19, Lemma 1])**.** *Let $S \subseteq X$ be a random subset and z be a random bit string. Let $G, H\colon X \to Y$ be random functions satisfying $G(x) = H(x)\ \forall x \notin S$. S, G, H, z may have an arbitrary joint distribution.*

Let A be a quantum oracle algorithm of query depth d (not necessarily unitary). Let E be an arbitrary (classical) event. Then we have

$$\Pr\left[E \wedge \neg\mathsf{Find} : x \leftarrow A^{|H\rangle\backslash S}(z)\right] = \Pr\left[E \wedge \neg\mathsf{Find} : x \leftarrow A^{|G\rangle\backslash S}(z)\right].$$

Semi-classical one-way to hiding lemma. The following lemma is called the semi-classical oneway-to-hiding lemma, the SC-O2H lemma in short.

Lemma 2.1.2 (The SC-O2H lemma [AHU19, Theorem 1])**.** *Let $S \subseteq X$ be a random subset and let z be a random bit string. Let $G, H\colon X \to Y$ be random functions satisfying $G(x) = H(x)$ for all $x \notin S$. S, G, H, z may have an arbitrary joint distribution.*

Let A be a (not-necessarily unitary) quantum oracle algorithm of query depth d. Let

$$\begin{aligned} P_{\text{left}} &:= \Pr\left[b = 1 : b \leftarrow A^{|H\rangle}(z)\right], \\ P_{\text{right}} &:= \Pr\left[b = 1 : b \leftarrow A^{|G\rangle}(z)\right], \\ P_{\text{find}} &:= \Pr\left[\mathsf{Find} : A^{|G\rangle\backslash S}(z)\right] = \Pr\left[\mathsf{Find} : A^{|H\rangle\backslash S}(z)\right]. \end{aligned}$$

Then we have

$$|P_{\text{left}} - P_{\text{right}}| \le 2\sqrt{(d+1)\cdot P_{\text{find}}} \text{ and } |\sqrt{P_{\text{left}}} - \sqrt{P_{\text{right}}}| \le 2\sqrt{(d+1)\cdot P_{\text{find}}}.$$

The lemma also holds with bound $\sqrt{(d+1)\cdot P_{\text{find}}}$ for the following alternative definition of P_{right}:

$$P_{\text{right}} := \Pr\left[b = 1 \wedge \neg\mathsf{Find} : b \leftarrow A^{|G\rangle\backslash S}(z)\right].$$

We often denote the above probability by $\Pr\left[\neg\mathsf{Find} : A^{|G\rangle\backslash S}(z) \to 1\right]$ for the notational simplicity.

Lemma 2.1.3 (Search in semi-classical oracle [AHU19, Theorem 2 and Corollary 1])**.** *Let A be any quantum oracle algorithm making at most q queries and depth d to a semi-classical oracle with domain X. Let $S \subseteq X$ and $z \in \{0,1\}^*$. S, z may have an arbitrary joint distribution.*

Let B be an algorithm that on input z chooses $i \leftarrow \{1, \dots, d\}$; runs $A^{O_\emptyset^{SC}}(z)$ until (just before) the i-th query; then measures all query input registers in the computational basis and outputs the set T of measurement outcomes.

Then we have

$$\Pr\left[\mathsf{Find} : A^{O_S^{SC}}(z)\right] \leq 4d \cdot \Pr[S \cap T \neq \emptyset : T \leftarrow B(z)].$$

In particular, if S and z are independent, A makes at most q queries, and we let $P_{\max} := \max_{x \in X} \Pr[x \in S]$, then we have

$$d \cdot \Pr[S \cap T \neq \emptyset : T \leftarrow B(z)] \leq q \cdot P_{\max}.$$

Double-sided one-way to hiding lemma. We will use the improved version of one-way to hiding lemma called the double-sided O2H [BHH+19] as the new algorithm should know both oracles.

Lemma 2.1.4 (Double-sided O2H [BHH+19, Lemma 5])**.** *Let $S \subset X$ be a random subset and let z be a random bit string. Let $G, H : X \to Y$ be random functions such that $G(x) = H(x)$ for all $x \notin S$. (G, H, S, z) may have an arbitrary joint distribution. Let $f : X \to \{0,1\}^n$ be an arbitrary function such that $f(S) = \{w^*\}$ is a single element. Let Ev be an arbitrary classical event.*

Let A be an (not-necessarily unitary) quantum oracle algorithm. Let

$$P_{\text{left}} := \Pr\left[\mathsf{Ev} : A^{|H\rangle}(z)\right], \qquad P_{\text{right}} := \Pr\left[\mathsf{Ev} : A^{|G\rangle}(z)\right].$$

Then there is another quantum oracle algorithm B that outputs only $\perp$ or w^*, which runs in about the same amount of time and space as A, but when A queries

to one of the oracle G, H, B queries both G and H and also runs f twice. Let $P_{\text{extract}} := \Pr\left[B^{|G,H\rangle}(z) \to w^*\right]$. *Then we have*

$$|P_{\text{left}} - P_{\text{right}}| \leq 2\sqrt{P_{\text{extract}}}, \text{ and } |\sqrt{P_{\text{left}}} - \sqrt{P_{\text{right}}}| \leq 2\sqrt{P_{\text{extract}}}.$$

Remark 1. In the original paper, the input z is assumed to be a classical string. However, we can obtain exactly the same bound even if z is a quantum state. This is because any quantum state can be described by a classical string with an exponential blowup of the size, and the above lemmas are only about query-complexities and the size of z does not matter.

Random oracle models. Cryptographic hash functions are usually modeled as follows: a random function H (of a certain domain and a codomain) is chosen uniformly at random first, and all algorithms including adversaries can access to the oracle that computes the function H, that is, returns $H(x)$ for a given input x. This model is called the random oracle model (ROM) [BR93] and widely used in cryptography.

The quantum random oracle model (QROM) [BDF+11b] was suggested as a quantum counterpart of the random oracle model, where all algorithms can access to a *quantum-accessible* oracle that computes the function H. More precisely, the oracle applies the following unitary transform.

$$\sum_{x,y,z} \alpha_{x,y,z} |x, y, z\rangle \mapsto \sum_{x,y,z} \alpha_{x,y,z} |x, y + H(x), z\rangle$$

Since the adversary may know the function H in advance as it is usually chosen to be the standard hash functions (e.g., SHA3 [D+15]), the auxiliary input random oracle model (ROM-AI or AI-ROM) is suggested in [Unr07, DGK17]. This model gives the oracle access to the adversary *before* the problem instance is chosen, and the adversary may do an arbitrary preprocessing and output a bounded-size advice string. Then the online adversary tries to solve the problem instance using the bounded number of oracle accesses and the preprocessing string.

2.2 Quantum Algorithms

In this section, we summarize the results of some well-known quantum algorithms that are relevant to the results of this thesis. More precisely, we use Grover's algorithm as a subroutine of Section 3.2 and and the algorithms for hidden subgroup problems in Section 3.3. The amplitude amplification algorithm is used as a technical tool in the proof of Chapter 4. More specified algorithms for each section are presented when they are used.

2.2.1 Quantum search and amplitude amplification

We use the following version [BBHT98] of Grover algorithm that does not require to know the number of solutions in advance.

Lemma 2.2.1 (Grover's algorithm)**.** *Let $f : X \to \{0,1\}$ be an arbitrary function. There is a quantum algorithm that makes $O(\sqrt{|X|})$ quantum queries to the oracle to compute f and outputs $x \in X$ such that $f(x) = 1$ with a constant probability.*

The amplitude amplification algorithm is described as follows.

Lemma 2.2.2 (Amplitude amplification [BHMT02])**.** *Let $f : X \to \{0,1\}$ be an arbitrary function. Let A be a unitary quantum algorithm (i.e., A is unitary except for the final measurement) that returns $x \in X$ such that $f(x) = 1$ with probability ε. Then there exists a quantum algorithm B that uses A, A^{-1}, and f as sub-routines $O(\varepsilon^{-1/2})$ times and returns $x \in X$ such that $f(x) = 1$ with probability $\Omega(1)$ where we abuse the notation to use A to mean the unitary corresponding to the algorithm A and A^{-1} to mean its inverse.*

2.2.2 Hidden subgroup problem

The hidden subgroup problem (HSP) is one of the most important problem of interest in quantum algorithms. For a group G, the hidden subgroup problem is

defined by a subgroup H of G and a function $f : G \to S$ that *hides* H meaning that

$$f(g) = f(gh) \Leftrightarrow h \in H$$

for an arbitrary $g \in G$. In other words, f is constant on each coset and takes distinct values for distinct cosets. The hidden subgroup problem asks to find (a basis of) the subgroup H given oracle access to f, as well as the knowledge of the group G along with efficient operations.

The standard algorithm for HSP relies on the Fourier sampling [BV97], which embraces the algorithms of Shor and Simon [Sho99, Sim97], and much more. We summarize the result of the algorithm for finite abelian groups.

Lemma 2.2.3 (HSP for abelian groups)**.** *Given a finite abelian group G and an oracle access to a function $f : G \to S$ that hides a subgroup H of G, there is an efficient quantum algorithm to find a basis of H with certainty.*

2.3 Cryptographic Primitives

This section summarizes the cryptographic primitives discussed in this thesis. Let n and m be positive integers. The negligible functions $\mathsf{negl} = \mathsf{negl}(n)$ is a function that is asymptotically smaller than an arbitrary inverse polynomial for every sufficiently large n.

2.3.1 Classical primitives

Definition 2.3.1 (One-way functions)**.** A classical efficient function $f : \{0,1\}^n \to \{0,1\}^m$ is *one-way* if it cannot be invertible by any efficient quantum algorithm. That is, for an arbitrary quantum polynomial time algorithm A, it holds that

$$\Pr[f(x') = f(x) : x \leftarrow \{0,1\}^n, x' \leftarrow A(f(x))] = \mathsf{negl}.$$

Definition 2.3.2 (PRGs)**.** A classical efficient function $G : \{0,1\}^n \to \{0,1\}^m$ is a *pseudorandom generator* (PRG) if it is length-increasing (i.e., $n < m$) and the outputs of G cannot be distinguished from random strings in $\{0,1\}^m$ by any efficient quantum algorithm. That is, for any quantum polynomial time algorithm A, it holds that

$$|\Pr[A(y) = 1 : y \leftarrow \{0,1\}^m] - \Pr[A(G(x)) = 1 : x \leftarrow \{0,1\}^n]| = \mathsf{negl}.$$

Definition 2.3.3 (PRFs)**.** A classical efficient function $F : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^m$ is a *pseudorandom function* (PRF) if for a randomly chosen key $k \leftarrow \mathcal{K}$, the function $F_k(\cdot) := F(k,\cdot)$ cannot be distinguished from random functions $f : \{0,1\}^n \to \{0,1\}^m$ as an oracle by any efficient quantum algorithm. That is, for any quantum polynomial time oracle algorithm A, it holds that

$$\left| \Pr_{k \leftarrow \mathcal{K}}[A()^{F_k} = 1] - \Pr_{f}[A()^{f} = 1] \right| = \mathsf{negl}.$$

2.3.2 Quantum-related primitives

The collapsing function is suggested as a quantum generalization of collision-resistant hash function.

Definition 2.3.4 (Collapsing function [Unr16])**.** A length-decreasing (i.e., $n > m$) function family $\mathcal{H} = \{H_k : \{0,1\}^n \to \{0,1\}^m\}_{k \in \mathcal{K}}$, the *collapsing experiment* $\mathsf{Exp}^{\mathsf{collapse}}(A)$ is defined as follows.

1. The challenger generates $k \leftarrow \mathcal{K}$.

2. A is given k as an input and generates a hash value $y \in \{0,1\}^m$ and a quantum state σ over registers $(\mathbf{X}, \mathbf{A})$ where X stores an element of $\{0,1\}^n$ and $\mathbf{A}$ is the A's internal register Then it sends y and register $\mathbf{X}$ to the challenger, and keeps $\mathbf{A}$ on its side. We say that A is *valid* if the measurement in the computational basis of $\mathbf{X}$ gives x such that $H_k(x) = y$ with probability 1 at this point.

3. The challenger picks $b \leftarrow \{0,1\}$. If $b = 0$, the challenger does nothing and if $b = 1$, the challenger measures the register $\mathbf{X}$ in the computational basis. The challenger returns the register $\mathbf{X}$ to A.

4. A outputs a bit b'. The experiment outputs 1 if $b' = b$ and 0 otherwise.

The function family $\mathcal{H}$ is called *collapsing* if for any valid quantum polynomial time algorithm A, it holds that

$$|\Pr\left[1 \leftarrow \mathsf{Exp}^{\mathsf{collapse}}(A)\right] - 1/2| = \mathsf{negl}.$$

The Haar measure on m-qubit states is denoted by μ_m. The pseudorandom quantum states [JLS18] are defined to be indistinguishable from the Haar random states even if the multiple copies are given as follows.

Definition 2.3.5 (PRSGs)**.** A *pseudorandom quantum states generator* (PRSG) is a QPT algorithm $\mathsf{StateGen}$ that, on input $k \in \{0,1\}^n$, outputs an m-qubit quantum state $|\phi_k\rangle$, which are indistinguishable from the Haar random quantum states even if the multiple copies are given. That is, for any QPT algorithm A and for any polynomial t, it holds that

$$\left| \Pr_{k\leftarrow\{0,1\}^n}\left[A\left(|\phi_k\rangle^{\otimes t(n)}\right) \rightarrow 1\right] - \Pr_{|\psi\rangle\leftarrow\mu_m}\left[A\left(|\mu\rangle^{\otimes t(n)}\right) \rightarrow 1\right]\right| \leq \mathsf{negl}.$$

The single-copy version of PRSG is defined as follows.

Definition 2.3.6 (Single-copy-secure PRSGs [MY21])**.** A *single-copy pseudorandom quantum states generator* (PRSG) is a QPT algorithm $\mathsf{StateGen}$ that, on input $k \in \{0,1\}^n$, outputs an m-qubit quantum state $|\phi_k\rangle$. As the security, we require the following: for any non-uniform QPT adversary A,

$$|\Pr_{k\leftarrow\{0,1\}^n}[A(|\phi_k\rangle) \rightarrow 1] - \Pr_{|\psi\rangle\leftarrow\mu_m}[A(|\psi\rangle) \rightarrow 1]| = \mathsf{negl}.$$

Note that if one-way functions exist, (poly-copy-secure) PRSGs exist [JLS18]. On the other hand, there is an evidence that (poly-copy-secure) PRSGs do not imply one-way functions [Kre21].

Part I

Post-Quantum Cryptography: Attacks, New Models, and Proofs

Chapter 3

Quantum Cryptanalysis

This chapter presents two new quantum algorithms for the function inversion problem and the multiple discrete logarithm problem. In Section 3.2, we describe a new quantum time-space trade-off on the function inversion problem for any function. A quantum algorithm for the discrete logarithm problem with multiple instances is introduced in Section 3.3.

3.1 Introduction

The new attacks in this chapter show a new ability of quantum computation, especially with a large amount of *preprocessing* or *quantum memory*. The new function inversion algorithm is much more efficient than Grover's algorithm, albeit with a large preprocessed data, thus invalidating the naïve fix for symmetric key cryptography. On the other hand, the multiple discrete logarithm algorithm has a better complexity than Shor's algorithm, albeit with a large quantum memory, thus showing the scalable quantum computer could threaten the individual users as well. In what follows, we describe a brief description of attacks.

For the function inversion problem, we formally analyze and extend the quantum version of Hellman's algorithm [Hel80] suggested in [DKRS21] with more

rigors. These analysis follows the rigorous classical analysis in [FN00, DTT10], partly complementing the result in Chapter 4. The new algorithm takes the same advice string as the classical preprocessing attack [Hel80], and Grover's algorithm is used to speed up the algorithm. The attack seems to be straightforward at first glance, we found and resolve some subtle problems, including the quantum memory consumption of online algorithm and the choice of the independent hash function family. Our inversion algorithm has applications to the data structure problem. For example, the systematic substring search problems [CGK18] and 3-SUM indexing problems [GGH+20] have quantum advantages using our algorithm.

The second algorithm solves the multiple discrete logarithm (MDL) problem: Let G be a group and g a generator. In the m-MDL problem, we are given multiple instances $g^{x_1}, ..., g^{x_m}$ of the DL problem, and asked to find $x_1, ..., x_m$ simultaneously. In the classical setting, m-MDL problem can be solved faster than solving the instances one-by-one manner as shown in [KS01]. Similarly, we prove that Shor's algorithm is not the best when the multiple instances are given. Our algorithm is based the multi-exponentiation algorithm, and has a better asymptotic complexity. Previous improvements and estimation [GE21] also show a better complexity than the original Shor's algorithm, but they do not improve the asymptotic complexity.

3.2 QROM-AI Algorithm for Function Inversion

In this section, we introduce a new quantum attack on the function inversion using the classical auxiliary input by generalizing the preprocessing algorithm [FN00] using the multiple ideas of [DTT10]. Let $f : D \to R$ for $D = R = [N]$ for simplicity. We are given $y \in R$ and asked to find x such that $f(x) = y$. If f is sampled uniformly at random, the algorithm with time T and space S has a trade-off of $T^2S^3 = \widetilde{O}(N^3)$. On the other hand, the trade-off of algorithm becomes

$TS^2 = \widetilde{O}(N^2)$ for an arbitrary function f.

Remark 2 (Comparison to the independent work)**.** A recent quantum algorithm suggested in [DKRS21] obtains a similar result by generalizing [Hel80]. Our algorithm is inspired by their work, but with more rigorous analysis based on [FN00, DTT10]. In particular, we use the concrete independent hash functions for analysis while they just assume that hash function works independently. In fact, the correctness of the algorithm for any function is not a straightforward generalization of [FN00], and requires some nontrivial ingredients from [DTT10]. They also consider the trade-off for time/space/data of $T^2S^3D^3 = \widetilde{O}(N^3)$. We did not include the trade-off for the number of data in this thesis, but the generalization is straightforward.

3.2.1 Offline Algorithm

The preprocessing data consists of 1) a set of precomputed inversion for inverting *bad* images and 2) a set of chains that is used for inverting other points. To describe the chains, we introduce the independent hash functions.

A family F of k-wise independent hash function is a set of functions such that for any points $x_1, ..., x_k$ in domain, the values $f(x_1), ..., f(x_k)$ behaves uniform at random over the range for the random choice of $f \in F$.

For $k < N^{0.99}$ and any N', a concrete family F of k-wise independent hash function $[N] \times [N'] \to [N]$ is constructed in [DTT10]. Furthermore, to sample r pairwise independent functions from F requires $\widetilde{O}(k)$ randomness, and the evaluation time of such functions is $N^{o(1)}$.

The offline algorithm constructs the data set as follows for parameters the size of list ℓ, the length of chain t, the number of independent hash functions r, and the number of chains m. Let $k = 2t(\log N)^2$.

1. A list L of size ℓ, which consists of $(x \in f^{-1}(y), y)$ for y with the highest values of $I(y) = |\{x : f(x) = y\}|$.

2. The randomness of size $\widetilde{O}(k)$ for sampling k-wise functions $g_1^*, ..., g_r^* : [N] \times [N'] \to [N]$ from F pairwise independently for $N' = (\log N)^2$. Using these functions, the functions $g_i : [N] \to [N]$ are defined as follows.

$$g_i(x) = \begin{cases} g_i^*(x,u) & \text{if } u \text{ is the least index such that } f(g_i^*(x,u)) \notin L \\ \perp & \text{if there is no such } u \end{cases}$$

3. Define $h_i(x) := g_i(f(x))$. For $i \in [r]$ and $j \in [m]$, by starting a random point x_{ij}, computing the chain $(x_{ij}, h_i(x_{ij}), ..., h_i^t(x_{ij}))$; we *discard* the chain and restart the procedure if

   - for some $h_i^{t_1}(x)$ is undefined for $t_1 \leq t$, or
   - the chain cycles, i.e. for $t_1, t_2 \leq t$, $h_i^{t_1}(x) = h_i^{t_2}(x)$.

   If the chain is not discarded, store $(x_{ij}, h_i^t(x_{ij}))$ as chain W_{ij}.

3.2.2 Classical Online Algorithm

The online phase of algorithm is given $y = f(x)$ and asked to find x' such that $f(x') = y$. The online algorithm proceeds in two steps.

1. If $(x', y) \in L$, then return x'.

2. Otherwise, do the following procedure for each $i \in [r]$:

   (a) Construct a chain $(g_i(y), h_i(g_i(y)), ..., h_i^{t-1}(g_i(y)))$.

   (b) If there is $j_0 \in [m]$ and $t_0 \leq t-1$ such that $h_i^{t_0}(g_i(y)) = h_i^t(x_{ij_0})$, then compute $h_i^{t-t_0-1}(x_{ij_0})$. If there are multiple choice of j_0, pick the smallest one.

   (c) If $f(h_i^{t-t_0-1}(x_{ij_0})) = y$, output $h_i^{t-t_0-1}(x_{ij_0})$ else output $\perp$.

The algorithm may meet the *false hit* for y if $(g_i(y), h_i(g_i(y)), ..., h_i^{t-1}(g_i(y)))$ contains the stored value but the stored chain fails to invert y.

As we will see, the online algorithm succeeds to invert y, without meet the false hit, with a constant probability given the following conditions hold. Here, $\lambda_\ell := \sum_{y \notin L} (I(y)/N)^2$ is the effective collision probability.

- $\ell, m, t \geq 1$, $t = O(\ell)$
- $mrt = \Omega(N)$.
- $mt^2\lambda_\ell = O(1)$.

With those parameters, the space complexity is $S = \widetilde{O}(mr + t + \ell)$ and $T = \widetilde{O}(rt)$. In particular, the trade-offs between time and space of our algorithm are $N^2 = S^2T$ and $N^3 = S^3T$ (ignoring logarithmic factors) for random functions and any function, respectively. The analysis for these conditions are briefly discussed in Section 3.2.4.

Remark 3. We note that the practical algorithms and its (heuristic) analysis do not consider the list L. The rigorous analysis requires this list.

3.2.3 Quantum Online Algorithm

The main idea of a quantum version is to apply Grover algorithm to find an appropriate $i \in [r]$. A naive calculation gives the time complexity of $T = \widetilde{O}(t\sqrt{r})$. Assume that this can be achieved for now. Note that $\lambda_\ell = O(1/N)$ for random function f, and either $1/\lambda_\ell = \Omega(\ell)$ or the first step of algorithm inverts most of inputs as shown in Lemma 3.2.1. We obtain the following two trade-offs. The conditions for both trade-offs can be easily checked.

- For random function f which has $1/\lambda_\ell \approx N$ with almost certainty, we obtain the trade-off curve $N^3 = S^3T^2$ (modulo logarithmic factor) by choosing

$r = t = N/S$ and $m = N/t^2$ and $\ell = \widetilde{O}(S)$. For any $S = \Omega(N^{2/3})$, the new algorithm provides a better performance than the naive application of Grover algorithm.

- For arbitrary function f, we obtain the trade-off curve $\lambda_\ell N^4 = S^3 T^2$ (modulo logarithmic factor) by choosing $t = N/S, m = N^2/ST^2, r = (ST/N)^2$ and $\ell = \widetilde{\Theta}(S)$. In particular, using $1/\lambda_\ell = \Omega(\ell) = \widetilde{\Omega}(S)$, we have the trade-off $N^2 = S^2 T$. For any $S = \Omega(N^{3/4})$ or $\lambda_\ell < (S/N)^3$, the new algorithm gives a better time complexity than Grover algorithm.

There are multiple issues for quantumizing the algorithm in Section 3.2.2, especially for Item 2 as follows. 1) We need to implement this procedure coherently, and 2) $\widetilde{\Theta}(t)$ quantum memory is used to compute the chain. We describe how to subvert the first problem with a careful implementation of algorithm below. Recall that the quantum oracle access to f is computed by

$$|x, y\rangle \mapsto |x, y + f(x)\rangle .$$

Implementation of online algorithm. A coherent implementation is to compute the whole chain starting from y; using a large amount of quantum memory and without removing the intermediate data. More precisely, for a given input y, we consider the following map $C_y : [r] \to \{0, 1\}$: First apply the map $U_{f,y}$

$$(0, 0, ..., 0) \mapsto (g_i(y), h_i(g_i(y)), ..., h_i^{t-1}(g_i(y)))$$

which can be computed by applying the quantum query to g_i and $h_i = g_i \circ f$ sequentially. Then we find the first $j_0 \in [m]$ and $t_0 \leq t - 1$ such that $h_i^{t_0}(g_i(y)) = h_i^t(x_{ij_0})$. Then compute $h_i^{t-t_0-1}(x_{ij_0})$ coherently, that is, computing $H_i^k(x_{ij_0})$ for $k = 0, ..., t - 1$ sequentially, where $H_i^k = h_i \circ H_i^{k-1}$ if $k < t - t_0 - 1$ and $H_i^k = H_i^{k-1}$ otherwise. Finally check if $f(h_i^{t-t_0-1}(x_{ij_0})) = y$; if true it returns 1 and 0 otherwise.

The overall query complexity is $O(t)$ to compute C_y (and uncompute the ancillary register), and the quantum space complexity is $O(t)$ times the unit memory.

Note that except the step for computing H_i^k, all steps can be coherently computed without any care. Applying Grover's algorithm returns the correct $i \in [r]$ (if any) with the query complexity $O(t\sqrt{r})$.

Note that the computation of H_i should be separately for each i; the original analysis of [FN00] relies on the careful amortized analysis thus cannot be applied here. We rely on the hash function of [DTT10] as shown below.

The other problematic step is to compute and record the whole chain, which requires a huge quantum memory. This can be reduced following Bennett's trick as in [DKRS21].

3.2.4 Sketch of Analysis

We provide a brief sketch of the analysis for correctness of the algorithm. In fact, the original analysis of [DTT10] can be applied straightforwardly, thus most parts of this section are borrowed from their analysis. Note that L stores ℓ number of $(x \in f^{-1}(y), y)$ with different y's, where y's have the largest $I(y) = |\{x : x \in f^{-1}(y)\}|$. Also note that the analysis of this section can be applied to both classical and quantum cases.

Inversion by the list

Let $N_0 := \sum_{y \notin L} I(y)$. We define the effective collision probability as follows.

$$\lambda_\ell = \sum_{y \notin \ell} \left(\frac{I(y)}{N_0} \right)^2$$

When the list L inverts relatively small inputs, the following lemma ensures some bounds on the parameters about the outside of lists.

Lemma 3.2.1 ([DTT10, Claim 4.2])**.** *If the list L does not invert f on ε_L-fraction of the inputs for $\varepsilon_L = 2/3$, then the following holds.*

1. $N_0 = \sum_{y \notin L} I(y) \geq (1 - \varepsilon_L)N$, *thus* $N_0 = \Theta(N)$.

2. *For all elements* $y \notin L$, $I(y) \leq \varepsilon_L N/\ell = O(N/\ell)$.

3. $1/N_0 \leq \lambda_\ell \leq \varepsilon_L N/\ell N_0 = O(1/\ell)$.

If L inverts most of inputs, we need to adjust some parameters dealing with the inputs in $[N] \setminus f^{-1}(L)$. We refer [FN00, Section 5] for more details.

Inversion by walks

Based on the estimations from Lemma 3.2.1, we can establish the success probability of inversion algorithm for each chain, a set of chains for a function g_i, and then a family of such sets. The results are essentially taken from [DTT10]. We summarize the results here for completeness.

The first statement on walk focuses on a single walk with a single function. Recall the walk is discarded if it meets undefined point, or it cycles. We say that the walk $W = (x, h(x), ..., h^t(x))$ inverts y if there is $j \leq t-1$ such that $f(h^j(x)) = y$. All probabilities appearing in the following lemmas are over the randomness of the data structure, unless specified otherwise.

Lemma 3.2.2 ([DTT10, Claim 4.3])**.** *Let h be a random function defined as in the preprocessing procedure. Let $W = (x, h(x), ..., h^t(x))$ be a walk constructed using randomly chosen x and h. For a given $y \notin L$, it holds that*

$$Pr[W \text{ is not discarded and inverts } y] \geq \frac{tI(y)}{2N_0} \cdot \left(1 - \frac{2tI(y)}{N_0} - t^2\lambda_\ell\right).$$

When we consider the probability that at least one of the walks inverts y, two inferences can appear. First, we need to account for the case that two walks both inverting y for showing a lower bound of the probability of at least one walk inverting y. The second interference is so-called the *false alarm* or false hit. Let us consider two walks W_1 and W_2 based on the same function $h = g \circ f$ with different starting points x_1, x_2. If $f(h^j(x_1)) = y$ and there is some $j_1 > j$ such that $f(h^{j_1}(x_1)) = f(h^{j_2}(x_2))$, then the online procedure to compute the walk from

$g(y)$ will meet both of two points $f(h^t(x_1))$ and $f(h^t(x_2))$. Then we need to recover both of the walks W_1 and W_2, which cause the running time slower. The following lemma gives upper bounds for those cases.

Lemma 3.2.3 ([DTT10, Claim 4.4])**.** *Let h and x_1, x_2 be randomly chosen function and points and $W_1 = (x_1, ..., h^t(x_1))$ and $W_2 = (x_2, h(x_2), ..., h^t(x_2))$ be two walks that are not discarded. The for any $y \notin L$, the following inequalities hold.*

1. $\Pr[W_1, W_2$ *are not discarded and both inverts* $y] \leq \left(\frac{tI(y)}{N_0}\right)^2$
2. $\Pr[W_1$ *inverts* $y \wedge W_2$ *generates a false hit*$] \leq \frac{t^3 I(y)\lambda_\ell}{N_0}$

We then analyze the case of m walks $W_1, ..., W_m$ for a single function g. We say that y is *inverted without false hit* if none of W_i produce a false hit and some W_j inverts y. The proof of this lemma is based on the inclusion-exclusion principle with

Lemma 3.2.4 ([DTT10, Claim 4.5])**.** *Let h and $x_1, ..., x_m$ be randomly chosen function and points. Suppose $mt^2\lambda_\ell \leq 1/8$ and $\varepsilon_L t/\ell \leq 1/4$ hold. For any $y \notin L$, the following inequality holds.*

$$\Pr[y \text{ is inverted without false hits by } W_1, ..., W_m] \geq \min\left(\frac{1}{32}, \frac{mtI(y)}{4N_0}\right)$$

We consider the general case, where r sets of m walks

$$\mathcal{W}_i = \{W_{i1}, ..., W_{im}\}$$

for $i = 1, ..., r$ are constructed according to pairwise independently and randomly chosen g_i^*. The lower bound of the probability that there is an index $i \in [r]$ such that $\mathcal{W}_i$ inverts given y without false hits. Note that we do not need to the case that W_{ij} inverts y and $W_{i'j'}$ produces a false hit for some $i \neq i'$.

Lemma 3.2.5 ([DTT10, Claim 4.6])**.** *Let $\mathcal{W}_1, ..., \mathcal{W}_r$ be r sets of m walks. Suppose the conditions of Lemma 3.2.4 all hold. Then, for any $y \notin L$, the following inequality holds.*

$$\Pr\left[y \text{ is inverted without false hits by one of } \mathcal{W}_i \text{ for } i \in [r]\right] \leq \min\left(\frac{1}{32}, \frac{rmtI(y)}{8N_0}\right)$$

Finally, the real algorithm repeats the above procedures constant time to amplify the probability in Lemma 3.2.5 by $\min(0.99, C \cdot rmtI(y)/N)$ for a sufficiently large constant C. By choosing $Crmt/N > 0.99$, we have the algorithm that can invert any input with probability 0.99.

3.3 Quantum Multiple Discrete Logarithm Problem

Now we present an algorithm for solving multiple instances of the discrete logarithm problem simultaneously. For a fixed cyclic group G, the multiple discrete logarithm problem $\mathsf{MDL}_{\mathsf{m}}(G)$ is defined as follows: We are given the unit element $1_G \in G$ and an generator g of G and m different instances of discrete logarithm problem of the form $y_i = g^{x_i}$, and asked to find $x_1, ..., x_m \in \mathbb{Z}$. The number of group operation is our complexity measure; we have the quantum oracle access to group operation $|b, g_1, g_2\rangle \mapsto \left|b, g_1, g_1 \cdot g_2^b\right\rangle$ for $b \in \{-1, +1\}$ and any $g_1, g_2 \in G$.

The main idea is to embed this problem into the $(m+1)$-dimensional abelian hidden subgroup problem with rank m; note that the original Shor's algorithm for discrete logarithm instance g^x embeds the instance into the 2-dimensional hidden subgroup problem with the rank 1 hidden group generated by $(1, -x)$. Then, we solve this hidden subgroup problem a bit faster, using the classic idea of multi-exponentiation.

Our algorithm requires $O(\log|G|/\log m)$ group operations per each instance, or $O(m \log|G|/\log m)$ group operations for solving $\mathsf{MDL}_{\mathsf{m}}(G)$, showing better performance than $O(m \log|G|)$ (or $\log|G|$ per instance) of Shor's algorithm. For practical parameters, our estimation predicts that the amortized number of group op-

erations for 512-MDL over group $|G|$ with $\log|G| = 512$ is about 200, giving $\times 5$ speed up for the naïve Shor's algorithm. Still, the attack is impractical as it requires a large quantum memory and the practical improvements are already done, e.g., in [GE21], we mostly focus on the asymptotic operation complexity in the remainder of this section,

Remark 4. Note that the classical $\mathsf{MDL}_{\mathsf{m}}(G)$ requires $\Theta(\sqrt{m|G|})$ group operations [KS01], which is shown to be optimal in the generic group model [Yun15]. We expect that our attack is optimal, but we are not able to show the corresponding lower bounds.

3.3.1 Multi-exponentiation problem

In the multi-exponentiation problem, we are given the group elements $1, x_1, ..., x_m$ and the nonnegative exponents $e_1, ..., e_m$, and asked to find $x_1^{e_1} \cdot ... \cdot x_m^{e_m}$ only using the multiplication. This problem was already well studied in several decades ago with the study of addition chain [Pip80, DLS81, Yao76, Oli81], and revived in the cryptographic context around the 2000s in the cryptographic context [LL94, Ber02, BGMW92]. The minimal number of multiplication for this problem is denoted by $\ell(x_1^{e_1} \cdot ... \cdot x_m^{e_m})$; unfortunately, to compute the exact value of ℓ itself is NP-complete [DLS81, Oli81].

Still, we can find an fairly short way to compute the multi-exponentiation. The following result due to Pippenger [Pip80] shows the (proven to be) almost-optimal complexity can be achieved if the number of group elements are mildly small. The modern exposition can be accessed in [Ber02, Hen10].

Proposition 3.3.1. *Let B be an integer, and $\lg m / \lg B = o(1)$. Suppose $e_i \le B$ for all i. Given $y_1, ..., y_m$ and $e_1, ..., e_m$, there is an efficient deterministic algorithm to compute $y_1^{e_1} \cdot ... \cdot y_m^{e_m}$ with $\lg B + (1 + o(1))(m \lg B / \lg(m \lg B))$ multiplications.*

We are interested in the case of $B = |G|$ and $m \gg 1$, and the asymptotic group

operation complexity becomes

$$\frac{(1+o(1))m\lg|G|}{\lg(m\lg|G|)},$$

which is much smaller than $m\lg|G|$ of the naïve computation.

As noted in [Hen10], this algorithm achieves the almost-optimal asymptotic complexity, but is less practical. We may use the practical alternatives in practice, e.g., [LL94, Boo02, BGMW92]. Following the rudimentary example of [Boo02], the required number of group operation is, for $m \leq \lg B$ and for $M = \sqrt{m\lg B}$,

$$\lg B + M + \frac{M^2}{\lg M - \lg\lg M} + \frac{M^2}{\lg M(\lg M - \lg\lg M)}.$$

Setting $\lg B = 512$ and $m = 512$ gives $M = 512$, which gives the above number of operations about 100,000. A naive square and multiply algorithm would requires about 500,000 group operations.

3.3.2 Multiple Discrete Logarithm Algorithm

Now we are describing the algorithm for MDL problem. We assume that $|G| = N$. The main observations for this algorithm are

- the map $f : Z_N^{m+1} \to G$ that $(k_0, ..., k_m) \mapsto g^{k_0}y_1^{k_1}...y_m^{k_m}$ hides $H \leq \mathbb{Z}_N^{m+1}$ that is rank m, namely generated by $\{x_i e_0 - e_i\}_{1\leq i\leq m}$ and
- to speed up this computation by Proposition 3.3.1.

Given the instances $g, y_1 = g^{x_1}, ..., y_k = g^{x_k}$, the algorithm proceeds as follows. We omit the amplitudes for normalization.

1. Compute $\otimes_{i=0}^{m}(\sum_{0\leq k<N}|k\rangle|0\rangle)$ using quantum Fourier transform. Here we store inputs in auxiliary registers.

2. Using Proposition 3.3.1, we rearrange the registers and compute

$$\sum_{0 \le k_0, \ldots, k_m < N} |k_0 \ldots k_m\rangle \otimes \left| g^{k_0} y_1^{k_1} \ldots y_m^{k_m} \right\rangle . \tag{3.1}$$

   This takes about $O(\lg B + m \lg B / \lg(m \lg B))$ oracle queries. Also note that we can safely uncompute the garbage state.

3. Apply the inverse quantum Fourier transform to the first register and measure it. This gives a vector included in rank 1 vector space $H^{\perp}$. In other words, except exponentially small probability (that the algorithm outputs the zero vector), we can solve m instances simultaneously.

Note that the algorithm is the standard algorithm for solving the hidden subgroup problem, except that we used the multi-exponentiation algorithm as the intermediate step. Thus the correctness of the algorithm is clear, i.e., the algorithm successfully finds $x_1, \ldots, x_k$ such that $y_i = g^{x_i}$ for all $i \in [k]$ with certainty. We include the analysis for completeness at the end of this section. The query complexity of algorithm is solely from the second step or the multi-exponentiation algorithm, thus is $(2 + o(1))m \lg |G| / \lg(m \lg |G|)$. The amortized complexity is then $O(\lg |G| / \lg(m \lg |G|)) = O(\log |G| / \log m)$ for any $m = \Omega(\log |G|)$.

3.3.3 Analysis

We conclude this chapter with the correctness analysis of the multiple discrete logarithm problem. We already calculated the overall state until the second step

in Equation (3.1). The inverse Fourier transform would give

$$\begin{aligned}
&\sum_{0\le k_0,\ldots,k_m<N} |k_0\ldots k_m\rangle \otimes \left|g^{k_0}y_1^{k_1}\ldots y_m^{k_m}\right\rangle \\
&\mapsto \sum_{0\le k_0,\ldots,k_m<N, 0\le j_0,\ldots,j_m<N} w_N^{\mathbf{k}\cdot\mathbf{j}} |j_0\ldots j_m\rangle \otimes \left|g^{k_0}y_1^{k_1}\ldots y_m^{k_m}\right\rangle \\
&= \sum_{0\le j_0,\ldots,j_m,v<N} \left(\sum_{0\le k_0,\ldots,k_m<N:\ g^{k_0}y_1^{k_1}\ldots y_m^{k_m}=g^v} w_N^{\mathbf{k}\cdot\mathbf{j}} \right) |j_0\ldots j_m\rangle \otimes |g^v\rangle\,.
\end{aligned}$$

Here, $\mathbf{j} = (j_0, \ldots, j_m), \mathbf{k} = (k_0, \ldots, k_m)$. The condition $g^{k_0}y_1^{k_1}\ldots y_m^{k_m} = g^v$ can be rephrased as $v = k_0 + x_1k_1 + \ldots + x_mk_m \bmod N$.

We argue that if $\mathbf{j}$ is not orthogonal to one of

$$(-1, 1/x_1, 0, \ldots, 0), (-1, 0, 1/x_2, 0, \ldots, 0), \ldots, (-1, 0, \ldots, 0, 1/x_m), \tag{3.2}$$

then it does not appear as the measurement result. Without loss of generality, assume that $\mathbf{j}$ is not orthogonal to $(-1, 1/x_1, 0, \ldots, 0)$. Then the amplitude is

$$\begin{aligned}
&\sum_{0\le k_0,\ldots,k_m<N:\ v=k_0+x_1k_1+\ldots+x_mk_m \bmod N} w_N^{\mathbf{k}\cdot\mathbf{j}} \\
&= \sum_{0\le k_2,\ldots,k_m<N} w_N^{k_2j_2+\ldots+k_mj_m} \left(\sum_{k_1,\ k_0=v-(x_1k_1+\ldots+x_mk_m) \bmod N} w_N^{k_0j_0+k_1j_1} \right) \\
&= \sum_{0\le k_2,\ldots,k_m<N} w_N^{k_2j_2+\ldots+k_mj_m} \left(\sum_{k_1} w_N^{k_1j_1+j_0(v-(x_1k_1+\ldots+x_mk_m))} \right)
\end{aligned}$$

where the exponent is equal to

$$(v - (x_1k_1 + \ldots + x_mk_m))j_0 + k_1j_1 = k_1(j_1 - x_1j_0) + j_0(v - (x_2k_2\ldots + x_mk_m))$$

so that the summation over k_2 becomes 0, using the fact $\sum_{0\le k<N} w_N^k = 0$. This proves the argument, thus the measurement result is always orthogonal to the vectors in Equation (3.2). In other words, we can recover all vectors in Equation (3.2), as well as all x_i. Therefore, the suggested algorithm solves the multiple discrete logarithm problem with certainty.

3.4 Discussion and Open problems

We conclude this chapter with an interpretation and discussion of the attacks, and some potential future works inspired by the attacks.

The two suggested attacks alarm for the naïve countermeasures, in theory and practice, for post-quantum cryptography. The non-uniform function inversion algorithm is faster than Grover's algorithm when the large amount of preprocessing data is given. On the other hand, the multiple discrete logarithm algorithm shows that the break-down of classical cryptosystem may be much efficient than the current expectation, given that a salable quantum memory.

The current security estimation of symmetric key cryptography is, however, based on the cost of Grover's algorithm. While the attack in Section 3.2 is not practical due to the requirement for a large amount of memory, the better-than-Grover cost of the attack contradicts with the simple estimation. A similar concern is suggested in [BL13]. This implies the definition of security should be more carefully defined, considering the non-uniform attacks as well.

The attack on the multiple discrete logarithm problem is a bit more subtle. The discrete logarithm problem and the integer factoring problem have already been known to be insecure against quantum attack due to Shor's algorithm. However, the current estimation [GE21] predicts that the quantum attack for a single RSA integer requires about 8 hours, even with careful optimizations.

The service providers, based on this prediction, delay the update of system to the post-quantum cryptography, because the individual user's instance may lie outside of the expensive quantum attack. However, the attack in Section 3.3 states that the cost to break multiple instances increases sub-linearly. Thus the multiple users could be the target of attack as well.

3.4.1 Variations

We can consider a scale-down version of the quantum non-uniform function inversion algorithm following [DTT10]. Some calculation gives the trade-off curve $S^3T^2 = \varepsilon^4 N^3$ for random functions, provided that $T > 1/\varepsilon$ and $S^2 > \varepsilon N$. For an arbitrary function, $S^2T = \varepsilon^3 N^2$ can be achieved for $T > 1/\varepsilon$. Note that these trade-offs do follow the lower bounds $\varepsilon = O(ST/N)$ from [CGLQ20].

We can slightly improve the multiple discrete logarithm algorithm (as well as the original algorithm due to Shor) by allowing preprocessing and the quantum-accessible classical memory. Note that this algorithm already requires a huge quantum memory for processing the multi-exponentiation algorithm.

In Section 3.3, we assumed the order $|G|$ is known in advance, and the generator g is fixed for $\mathsf{MDL}_{\mathsf{m}}(G)$ for the sake of simplicity. Both of the assumptions are not essential. The order N can be found using Shor's algorithm, and also can be factorized as well if needed. On the other hand, the problem with multiple generators that we are given $(g_i, y_i = g_i^{x_i})$ is almost equivalent to our setting: we can find $(z_1, ..., z_m)$ such that $g_i = g_1^{z_i}$ and $(w_1, ..., w_m)$ such that $y_i = g_1^{w_i}$ by invoking the $\mathsf{MDL}_{\mathsf{m}}(G)$ solver twice. Then it holds that $g_1^{z_i x_i} = g_i^{x_i} = y_i = g_1^{w_i}$, which implies $x_i = y_i/z_i \bmod N$.

3.4.2 Open problems

A first natural problem is to ask the role of quantum advice for the function inversion problems, which is currently unknown at all. The relevant question if the quantum advice can be used for solving multiple instance is also interesting. This question is discussed with more details in Section 5.5.

We may want to show the multiple instance discrete logarithm algorithm is asymptotically optimal. Unfortunately, we are unable to show any lower bound so far. A natural direction is to generalize the lower bound of Simon's problem [KNP07] to our context.

Another problem is to find a quantum algorithm for solving at least one of solution among multiple instances. This problem also has a practical implication, because the faster algorithm for solving one of multiple instances could be interesting for some practical scenarios.

Chapter 4

Quantum Random Oracle Model with Classical Advice

We present a new model that formalizes the quantum attacks with the preprocessing classical data for hash functions in this chapter, complementing the preprocessing attack in Chapter 3 and [DKRS21]. In this model, the hash functions is assumed to be a truly random function O in the initial stage following the previous models [BR93, BDF+11a], and the adversary only can access this function via making a *superposition* query of the form

$$|x, y\rangle \mapsto |x, y \oplus O(x)\rangle$$

to the oracle to compute the random function O.

The behavior of adversary has two phases: First, it runs an arbitrary process with the oracle for random functions, and outputs the preprocessing *classical* data with a bounded size. Then, the adversary takes the problem and preprocessing data as inputs, and tries to solve the problem by making a small number of superposition queries to the oracle. We call this model by the quantum random oracle model with auxiliary input (QROM-AI), as the online adversary is given the preprocessing data as an auxiliary input.

The main result of this section states that the hash functions in the QROM-AI can be used as basic cryptographic primitives such as one-way functions, pseudorandom generators, (post-quantum) pseudorandom functions, and (post-quantum) message authentication codes.

The *incompressibility argument* is the starting point of the proof, which is developed by Genarro, Gertner, Katz, and Trevisan [GT00, GGKT05]. The basic idea behind the technique is summarized as follows: Any correct encoding-decoding scheme cannot have a much shorter encoding length than the message.

The lower bound proof based on the incompressibility argument roughly proceeds as follows. Let A be an adversary for our interest problem, say the function inversion. We construct the encoding and decoding scheme for the set of all possible functions $f : [N] \to [N]$ using A. The encoding of a function f consists of a function table of f for $D \subset [N]$, i.e., $(x, f(x))$ for $x \in D$. To recover the remainder points $[N] \setminus D$, we run the algorithm $A(y)$ for each $y \in [N] \setminus f(D)$. The better algorithms make the better encoding schemes. At this point, the incompressibility argument comes into play, giving the limitation of the encoding scheme. This in turn implies the limitation of the inversion algorithm, which completes the proof.

The actual proof is much involved. The algorithm $A(y)$ may fail to find inversion, or in the decoding process it may halt due to the imperfect data of f. Furthermore, we need to deal with the (inherently randomized) quantum algorithm and the superposition query to f, which potentially includes all information of f. With various techniques such as the one-way to hiding lemma [AHU19] and the measurement simulation we developed, we can obtain meaningful security bounds of the hash functions.

In Section 4.1, we formalize the above model and show some basic observations. Then we state and prove the limitation of adversaries in QROM-AI for one-way functions and pseudorandom generators in Section 4.2 and Section 4.3, respectively. Finally, we discuss some further results in QROM-AI in Section 4.4. We refer the original paper [HXY19] for more detailed proofs.

Remark 5 (Improvements)**.** The theorems in this thesis have better bounds compared to the corresponding theorems in [HXY19], especially for the exponents. One of the main reason of improvements is use of *the bucketing argument*, which gives a slightly better results than the standard averaging argument. We include this idea in the proof of Theorem 4.2.1. On the other hand, we use an improved bound in [CGLQ20] for Yao's box problem and the double-sided O2H [BHH+19].

Remark 6 (Follow-up and independent works)**.** As noted in the introduction, several related works have been conducted after the publication of our work. [CLQ20] proves a similar lower bound for function inversion problem based on a similar technique, and the follow-up work [CGLQ20] gives a tighter bound using a seemingly different method. This bound is reproved in [GLLZ21]. We discuss some more related works and possible future directions in Section 4.5.

Notations. In this chapter, we consider a random oracle with the domain $[K] \times [N]$ (or $[K] \times [N] \times [L]$ for some cases) and the codomain $[M]$, which is denoted by $\mathsf{Func}([K] \times [N], [M])$. If $K = 1$, we omit $[K]$ for simplicity. We also consider the set of permutations, denoted by $\mathsf{Perm}([N])$. We omit to state a distribution of a random oracle O if that is uniformly chosen from the set of functions with the corresponding domain and codomain. We use a and x to represent elements of $[K]$ and $[N]$ respectively throughout the section, and often omit to state distributions when they are uniform. For example, we write $\Pr_{a,x}[f(a,x) = y]$ instead of $\Pr_{a \leftarrow [K], x \leftarrow [N]}[f(a,x) = y]$.

4.1 Quantum ROM with Auxiliary Input

We describe our new model called the quantum random oracle model with (classical) auxiliary input (QROM-AI). This model assumes that the hash functions are sampled from the uniform random function as an initialization of the cryptographic problem.

The algorithm $A = (A_0, A_1)$, or the adversary, in QROM-AI proceeds by two phases. In the first phase, the *preprocessing* or *offline* algorithm A_0 takes no inputs, accesses the random oracle O in an arbitrary way, and outputs a classical string st_O. The preprocessing algorithm has no bound on the resource such as time and space, but the output string st_O has a bounded bit-length S.

In the second phase, the *online* algorithm A_1 takes the preprocessing data st_O. Then, the algorithm is given a problem instance and asked to solve the problem with quantum oracle access to O. The resource of the online algorithm is bounded; in this thesis, we usually count the number of oracle query T of online algorithm. We focus on the QROM-AI algorithm with a small amount of space S and query T, and do not care about the running time of algorithms.

The unbounded running time of algorithms gives some useful tricks. First, the preprocessing algorithm A_0 can explore all possible string st_O and choose the best one that maximizes the advantage of online algorithm. Therefore, we assume that the QROM-AI adversary is the (single) algorithm A taking an advice string st_O. We use this observation throughout in this thesis. The second trick is based on a more involved observation, described as follows.

4.1.1 Simulating Measurement

The measurements introduce an inherent randomness of the output of quantum algorithms, thus the prediction of output of algorithm is also imperfect in general. However, the proof based on the incompressibility argument [GT00, GGKT05] requires the deterministic algorithms, makes some troubles for the quantum algorithm's lower bounds. We resolve this problem by considering the simulation of quantum algorithm including the intrinsic randomness of measurements, as the decoder does not have the limit of running time. We describe the detail of this simulation below for the completeness of this thesis.

Let us assume that the quantum algorithm A applies the measurement only at

the final stage, which can be done without loss of generality due to the principle of deferred measurement; we also cover the case without this assumption at the end of this section. To simulate the quantum algorithm $A(z)$, we compute *all* of the amplitude of the state right before measurement described by $\sum_i \alpha_i |i\rangle$. Then the simulation determines the output by sampling a randomness $r \in [0,1]$. Precisely, the output is the largest j such that $\sum_{i=1}^{j-1} |\alpha_{x_i}|^2 \leq r$.

We denote this procedure by $\mathsf{Sim}_r(A(z))$. If we consider many inputs $z \in Z$ and a corresponding random coin $R = \{r_z\} \in [0,1]^{|Z|}$, we just denote $\mathsf{Sim}_{r_z}(A(z))$ by $\mathsf{Sim}_R(A(z))$ for the simplicity of notation.

We note that exactly the same procedure is possible for an oracle-aided quantum algorithm $A^{|f\rangle}$ that accesses a quantum oracle $|f\rangle$ that computes a function f *if the simulator knows the whole data of* f since we can think of the combination of A and $|f\rangle$ as a single quantum algorithm.

When the quantum algorithm accesses the classical algorithm, it may measure some registers as an intermediate step to compute the classical oracle. The simulation for this case is done by augmenting the amount of randomness used by the simulator so that fresh randomness is available in the simulation of each measurement.

4.2 Function Inversion

Now we discuss the lower bound of the advantage of the QROM-AI adversary. We first show that the hardness to find the preimage of evaluation of random functions. The basic result is summarized by the following theorem. This theorem roughly speaks that any quantum polynomial time algorithm cannot find the preimage of input for random permutations.

Theorem 4.2.1. *Let* $O \in \mathsf{Perm}([N])$ *be a random permutation. Suppose that* A *is an oracle-aided quantum algorithm that takes an* S*-bit classical advice* st_O *(that*

may depend on O) as input, makes at most T oracle queries. Then it holds that

$$\Pr_{A,O,x}\left[A^{|O\rangle}(\mathsf{st}_O, O(x)) \to x\right] = \widetilde{O}\left(\frac{ST^2}{N}\right).$$

The proof is based the following so-called compression lemma. In particular, we set $\mathcal{M} = \mathsf{Perm}([N])$ and will construct the encoding and decoding scheme using the algorithm A.

Lemma 4.2.2 (Compression lemma, [DTT10, Fact 8.1])**.** *Let $\mathcal{M}, C, \mathcal{R}$ be sets. Let $E : \mathcal{M} \times \mathcal{R} \to C$ and $D : C \times \mathcal{R} \to \mathcal{M}$ be deterministic algorithms. For $\delta \in [0, 1]$, if we have*

$$\Pr_{r \leftarrow R}[D(E(m, r), r) = m] \geq \delta$$

for all $m \in M$, then we have $|C| \geq \delta|\mathcal{M}|$, which can be rephrased as $\log|C| \geq \log|\mathcal{M}| - \log 1/\delta$.

Precisely, we will compress most parts of the function table of O, i.e., $(x, O(x))$, along with the classical advice string st_O. Then we recover the remainder parts of O by exploiting the adversary A with the partial table of O; the correctness of algorithm A with a partial table of O can be ensured by the one-way to hiding lemma in Lemma 2.1.2. We include some additional information as a part of encoding for specifying the way to recover the remainder parts of O.

The following results are about the random functions, and can be proven with the similar arguments. Theorem 4.2.3 shows the function inversion problem for salted random functions. Lemma 4.2.4 is technical tool for proving the other lower bounds. In this lemma, we give an upper bound for the probability that the event Find occurs when an adversary is given a punctured oracle on the correct answer. (See Section 2.1.1 for the definitions of Find and the punctured oracle.) This corresponds to [DGK17, Corollary 1], which gives a bound for the probability that an adversary *ever queries* the correct answer to the oracle in the classical case.

Theorem 4.2.3. *Let $O \in \mathsf{Func}([K] \times [N], [M])$ be a random oracle. Suppose that A is an oracle-aided quantum algorithm that takes an S-bit classical advice st_O (that may depend on O) as input, makes at most T oracle queries, and satisfies*

$$\Pr_{A,O,a,x}\left[O(a,x) = O(a,x') : A^{|O\rangle}(\mathsf{st}_O, a, O(a,x)) \to x'\right] = \varepsilon.$$

Then it holds that

$$\varepsilon = \widetilde{O}\left(\frac{ST^2}{K\min(M,N)} + \frac{T^2N}{\min(M,N)^2}\right).$$

Lemma 4.2.4. *Let $O \in \mathsf{Func}([K] \times [N], [M])$ be a random oracle. Suppose that A is an oracle-aided quantum algorithm that takes an S-bit classical advice st_O (that may depend on O) as input, and makes at most T oracle queries. Then it holds that*

$$\Pr_{A,O,a,x}\left[A^{|O\rangle,|O\rangle\setminus\{(a,x)\}}(\mathsf{st}_O, a, O(a,x)) \to x\right] = O\left(\frac{ST^2}{KN} + \frac{T^2\log N}{N}\right).$$

A careful reader may notice that the problem in Theorem 4.2.1 is about the random *permutation*, not about the random functions, or the random oracle. We opted to state the theorem for random permutations for showing the main idea of the proof.

The main idea of the proof of these theorems is to compress the function table of the random function into a smaller encoding by using an algorithm that inverts the function. Then by invoking Lemma 4.2.2, we obtain a bound for the advantage to invert the function. Specifically, we encode a function into an encoding that consists of a partial function table and information to recover the remaining information of the function similarly to [DGK17] with several new ideas. We conclude this section with a (slight sketch) proof of Theorem 4.2.1. The function cases are much involved, especially because even the always-winning adversary may output the random preimage. This makes the encoding and decoding complex.

We also remark that Lemma 4.2.4 is different from the statement in the original paper. This is because the lemma is for using the double-sided one-way to hiding

lemma (Lemma 2.1.4) in the proof of Theorem 4.3.1 that improves some factors of the lower bounds.

4.2.1 Proof of Theorem 4.2.1

Transform the advantage of adversary. The adversary in Theorem 4.2.1 has an ε-advantage over *random* instances, meaning that the advantage is the expectation over all oracles and instances. This is the standard notion of adversary in cryptographic context. In contrast, the lower bound proofs become easier and tight for the advantage that counts the portion of easy instances. We describe the transform bridging two cases based on the bucketing argument.

Remark 7. The original paper [HXY19] used the standard averaging argument for obtaining the latter adversary (which is called the *biased* adversary in that paper), and the other previous papers [NABT15, CLQ20] do use the easier form of adversary without any transform. As mentioned previously, the transform based on the bucketing argument improves the result by a factor in exponent, while the averaging argument makes a loss of advantage, as explicitly noted in the original paper.

Suppose that an adversary A takes S-bit classical advice, and makes T_0 oracle queries, then outputs the correct answer with probability ε_0 on expectation over the choice of random instances. Let $L := \lceil \log(2/\varepsilon) \rceil$. We divide the problem instance parameters (O, x) by

$$S_j := \{(O, x) : 2^{-j} \leq \Pr_A[A^{|O\rangle}(\mathsf{st}_O, O(x)) \to x] < 2^{-j+1}\}$$

for $j \in [L]$ and set $S_0 := \{(O, x) : \Pr_A[A^{|O\rangle}(\mathsf{st}_O, O(x)) \to x] < \varepsilon_0/2\}$. Let $|S_j|/|\{O, x \in \mathsf{Perm}([N]) \times [N]\}| = p_j$ for $j \geq 0$. Then the given lower bound of advantage of A ensures that

$$\varepsilon_0 \leq \frac{\varepsilon_0}{2} + \sum_{j=1}^{L} 2^{-j+1} \cdot p_j$$

where we used $p_0 \leq 1$. By the averaging argument, there exists a $j \in [L]$ such that $p_j \geq 2^j\varepsilon_0/4L$. In other words, there is $j = O(\log(1/\varepsilon_0))$ such that the following inequality holds.

$$\Pr_{O,x}\left[\Pr_{A}[A^{|O\rangle}(\mathsf{st}_O, O(x)) \to x] \geq 2^{-j}]\right] \geq \frac{2^j\varepsilon_0}{4L}$$

Now we apply the amplitude amplification [BHMT02] (see Lemma 2.2.2), obtaining another algorithm A' that uses A, A^{-1} and O (for checking the correctness) as sub-routines $O(2^{j/2})$ times and satisfies

$$\Pr_{O,x}\left[\Pr_{A}[A^{|O\rangle}(\mathsf{st}_O, O(x)) \to x] \geq 2/3]\right] \geq \frac{2^j\varepsilon_0}{4L} =: \varepsilon. \tag{4.1}$$

Note that the algorithm A' has the query number $T = 2^{j/2}T_0$ and the (biased) advantage $\varepsilon := 2^j\varepsilon_0/4L$, thus satisfies $T^2/\varepsilon = \widetilde{\Theta}(T_0^2/\varepsilon)$. Since the algorithm A' uses the same advice[1] st_O, the trade-off bound $\varepsilon = \widetilde{O}(ST^2/N)$ of the algorithm A' would imply the equivalent trade-off bound $\varepsilon_0 = \widetilde{O}(ST_0^2/N)$ of the original algorithm A. Therefore, we consider the algorithm A such that Equation (4.1) holds below instead of the original adversary.

Algorithm with partial data. We now describe some observations of algorithm A that satisfies Equation (4.1) for the later encoding and decoding schemes. In more details, we will remove a random portion R from the domain $[N]$, and include the function table of O for $[N] \setminus R$. Then we will show that the algorithm A can find the correct solution for a sufficiently large portion of $x \in R$, only using this partial data of O. This is formalized by the language of the semi-classical oracles, or punctured oracles.

Let A be an adversary using T query and S-bit classical advice, and satisfying Equation (4.1). We consider another adversary B works as follows, for invoking the one-way to hiding lemma Lemma 2.1.2.

[1] Technically, we may add j as the advice, but we omit this as it is $O(\log\log 1/\varepsilon)$-bit.

$B^{|O\rangle}(\mathsf{st}_O, y)$: It runs $A^{|O\rangle}(\mathsf{st}_O, y)$. Then B outputs 1 if the answer z of the algorithm A satisfies $O(z) = y$, and outputs 0 otherwise.

Note that the advantage of the adversary A can be rephrased as

$$\Pr_{O,x}\left[\Pr_B[B^{|O\rangle}(\mathsf{st}_O, O(x)) \to 1] \geq 2/3]\right] \geq \varepsilon.$$

By applying the standard averaging argument for the random functions, we have that for at least $\varepsilon/2$-fraction of $f \in \mathsf{Perm}([N])$ satisfies

$$\Pr_x\left[\Pr_B[B^{|f\rangle}(\mathsf{st}_f, f(x)) \to 1] \geq 2/3\right] \geq \varepsilon/2.$$

We say that such a function f is nice; we only consider the nice f below. Let I be a set of $x \in [N]$ that B outputs 1 with probability at least $2/3$ as follows. From the above inequality, $|I| \geq \varepsilon N/2$ is obvious.

$$\Pr_B[B^{|f\rangle}(\mathsf{st}_f, f(x)) \to 1] \geq 2/3 \tag{4.2}$$

Now we choose a random set $R \subset [N]$. This random set will be fed as a public random coin for encoding and decoding algorithms, and the function table of f for $[N] \setminus R$ will be included as the encoding of f. We will show that A works well for $R \cap I$, even if we replace f by a certain function reconstructed from the partial data of f.

The specific choice of R is as follows: For any $x \in [N]$, we include x in R with probability $b/T(T+1)$ for a constant b to be specified later. We fix R in the below discussion. We say that $x \in I$ is good if the following two conditions simultaneously hold, where a constant $c < 1/36$ to be specified later.

$$(A)\ x \in R \qquad\qquad (B)\ \Pr_B[\mathsf{Find} : B^{|f\rangle\setminus(R\setminus\{x\})}(\mathsf{st}_f, f(x))] \leq \frac{c}{T+1}$$

We denote the set of all good elements by $G = G(R)$. Looking ahead, we will show that A works well for G with the partial data. The following claim ensures that G has the expected size with a high probability. Note that I and R contain about ε-fraction and $(1/T^2)$-fraction of $[N]$, we cannot hope a better bound.

Claim 1. $\Pr_R[|G(R)| \geq \delta\varepsilon N/T^2] \geq 0.8$ *for some constant* $\delta > 0$.

The proof can be found in the end of this section.

For fixed R and $x \in G$, let $y = f(x)$ and we consider the following function g_y that can be computed with only the knowledge of partial information of f plus y, and only differ from f at R.

$$g_y(z) = \begin{cases} f(z), & \text{if } z \in [N] \setminus R, \\ y, & \text{if } z \in R \end{cases}$$

The semi-classical one-way to hiding lemma (Lemma 2.1.2) gives

$$\begin{aligned} &\left| \Pr_B \left[B^{|g_{f(x)}\rangle}(\mathsf{st}_f, f(x)) \to 1 \right] - \Pr_B \left[B^{|f\rangle}(\mathsf{st}_f, f(x)) \to 1 \right] \right| \\ &\leq \sqrt{(T+1) \Pr_B[\mathsf{Find} : B^{|f\rangle \setminus R}(\mathsf{st}_f, f(x))]} \leq \sqrt{c}. \end{aligned}$$

where we used the condition (B) in the last inequality. Equation (4.2) and $c < 1/36$ implies that

$$\Pr_B \left[B^{|g_{f(x)}\rangle}(\mathsf{st}_f, f(x)) \to 1 \right] > 1/2$$

for any $x \in G$, which is equivalent to

$$\Pr_A \left[A^{|g_{f(x)}\rangle}(\mathsf{st}_f, f(x)) \to x \right] > 1/2.$$

In other words, the algorithm A can successfully find the preimage of $f(x)$ *without* the knowledge of $f(R)$ for $x \in G$ given the appropriate choice of randomness, which is what we desired.

Simulation of the algorithm. Now we choose another randomness R' as described in Section 4.1.1 for simulating the quantum algorithms A with the quantum oracle $\left|g_{f(x)}\right\rangle$. Again by the standard averaging argument, 1/4-fraction of R' satisfies that

$$\mathsf{Sim}_{R_2} \left(A^{|g_{f(x)}\rangle}(\mathsf{st}_f, f(x)) \right) \to x$$

for $1/4$-fraction of G.

In summary, we have the following result on the simulation of the algorithm A given the advantage of A given in Equation (4.1).

($*$) For $\varepsilon/2$-fraction of $f \in \mathsf{Perm}([N])$, $\Omega(1)$-fraction of R, and $\Omega(1)$-fraction of R', there is a subset V of $[N]$ with the size at least $\Omega(\varepsilon N/T^2)$ such that for all $x \in V$ it holds that

$$\mathsf{Sim}_{R_2}\left(A^{|g_{f(x)}\rangle}(\mathsf{st}_f, f(x))\right) \to x.$$

Encoding procedure. We now describe the encoding scheme for a nice permutation f. It first picks random R, R' described as above. The encoding of f for the randomness R, R' consists of the following data.

1. The advice string st_f.
2. The values of f on $[N] \setminus R$.
3. The description of $f(V)$ as a subset of $f(R)$ along with its size.
4. The values of f on $R \setminus V$.

Decoding procedure. Given the encoding data, the permutation f can be recovered as follows.

1. Parse the advice st_f.
2. Fill the function table for $f : [N]\setminus R \to [N]\setminus f(R)$ using the data from Item 2. At this point, the decoder automatically knows the set $f(R)$.
3. Recover $f(V)$ as the subset of $f(R)$.
4. For each $y \in V$, recover $f^{-1}(y)$ as the output of $\mathsf{Sim}_{R_2}\left(A^{|g_y\rangle}(\mathsf{st}_f, y)\right)$. This procedure recovers the partial table $f : V \to f(V)$.

5. Finally fill the function table $f : R\backslash V \to f(R\backslash V)$ using the data from Item 4.

If the conditions in (*) all holds, the decoding procedure fully recovers f.

The advantage bound from Lemma 4.2.2. We finally derive the lower bound of advantage ε in terms of S, T, N using the incompressibility argument. We first compute the size of encoding. Let us assume some convenient conditions, e.g., N is a power-of-two integer. We can prove the general case by augmenting constant factors with a bit involved argument. The size of encoding is computed as follows, where we use the base-2 logarithm as $\log$.

The advice is S-bit by definition. The function table of f over $[N] \setminus R$ has $N\cdot(N-1)\cdot\ldots\cdot(|R|+1) = N!/|R|!$ possibility, thus can be described by $\log(N!/|R|!)$-bit. The size of V is described by $\log N$ bits. Since the number of subset of R with size $|V|$ is $\binom{|R|}{|V|}$, V can be described by $\log\binom{|R|}{|V|}$. Finally the remainder data is described by $\log(|R| - |V|)!$-bit string. Thus, the overall encoding size is bounded by

$$S + \log\left(\frac{|R|!}{|V|!(|R| - |V|)!}\right) + \log\left(\frac{N!}{|R|!}\right) + \log(|R| - |V|)! = S + \log\left(\frac{N!}{|V|!}\right)$$

Here the size of $|V|$ is at least $\Omega(\varepsilon N/T^2)$. Lemma 4.2.2 states that this quantity has a lower bound $\log(\varepsilon N!/2) - \Omega(1)$ where the term $\Omega(1)$ comes from the randomness R, R'. Finally, we have $S + O(1) \geq \log(\varepsilon|V|!/2)$. Since $\log n! = \Theta(n \log n)$, we have

$$\varepsilon = \widetilde{O}\left(\frac{ST^2}{N}\right),$$

which concludes the proof.

Proof of Claim 1. Recall that the conditions that $x \in R$ with probability $p = b/T(T+1)$ for any $x \in [N]$ and $G(R)$ is defined by the set of x satisfying the following conditions.

$$(A)\ x \in R \qquad (B)\ \Pr_B[\mathsf{Find} : B^{|f\rangle\backslash(R\backslash\{x\})}(\mathsf{st}_f, f(x))] \leq \frac{c}{T+1}$$

Also note that $|I| \geq \varepsilon N/2$. We will prove that $\Pr_R[|G(R)| \geq \delta\varepsilon N/T^2] \geq 0.8$ for some constant $\delta > 0$.

Let H be a subset of I that consists of all elements satisfying (A). Since the event $x \in R$ is independent for each x, we have $\mathbb{E}[|H|] = p|I|$. The Chernoff bound implies that the following inequality holds[2].

$$\Pr_R\left[|H| \geq p|I|/2\right] \geq 1 - \exp(-p|I|/8) \geq 0.9 \tag{4.3}$$

Let $p_{\mathsf{Find}} := \Pr_{R,B}[\mathsf{Find} : B^{|f\rangle\setminus R}(\mathsf{st}_f, f(x))]$. Lemma 2.1.3 states that the following inequality holds.

$$p_{\mathsf{Find}} \leq 4T \cdot \max_{x\in[N]} \Pr[x \in R] = 4T \cdot p = \frac{4b}{T+1}$$

The Markov inequality implies the following inequality. Note that the second inequality is obvious since the punctured set is decreased.

$$\frac{4b}{c} \geq \Pr_R\left[p_{\mathsf{Find}} \geq \frac{c}{T+1}\right] \geq \Pr_R\left[\mathsf{Find} : B^{|f\rangle\setminus(R\setminus\{x\})}(\mathsf{st}_f, f(x))] \geq \frac{c}{T+1}\right]$$

Let J be a subset of I that consists of all elements satisfying (A) but not (B). Note that two events (A) and (B) are independent since (A) only depends on if $x \in R$ and (B) only depends on if the other points (i.e. that are in $[N] \setminus \{x\}$) are in R. Therefore, for any $x \in I$, we have

$$\Pr_R[x \in J] \leq 4b/c \cdot p = 4b^2/cT(T+1).$$

This gives $\mathbb{E}[|J|] \leq 4b^2|I|/cT(T+1)$, and the Markov inequality gives

$$\Pr_R\left[|J| \leq \frac{40b^2|I|}{cT(T+1)}\right] \geq 0.9.$$

This inequality combined with Equation (4.3) gives that the lower bound of $|G|$ holds as follows with probability at least 0.8.

$$|G| = |H| - |J| \geq \frac{b|I|}{2T(T+1)} - \frac{40b^2|I|}{cT(T+1)} = \Omega\left(\frac{\varepsilon N}{T^2}\right)$$

if $40b/c < 1/2$. Overall, Claim 1 holds by setting $80b < c < 1/36$.

[2] In fact, we need an assumption $\varepsilon N/T(T+1) > C$ for a sufficiently large C; here $\varepsilon N/T(T+1) = \Theta(p|I|)$. If this does not hold, the conclusion is obvious. We omit the detailed discussion for this assumption.

4.3 Pseudorandom Generators

The second main result of this chapter is the (pseudo-)randomness of the outputs of random oracles. The following theorem asserts that any QROM-AI adversary cannot distinguish the outputs of random oracle from the genuine random values.

Theorem 4.3.1. *Let $O \in \mathsf{Func}([K] \times [N], [M])$ be a random oracle. Suppose that A is an oracle-aided quantum algorithm that takes an S-bit classical advice st_O (that may depend on O) as input, and makes at most T oracle queries. Then it holds that*

$$\left| \Pr_{A,O,a,x}\left[A^{|O\rangle}(\mathsf{st}_O, a, O(a,x)) \to 1\right]\right| - \left| \Pr_{A,O,a,y}\left[A^{|O\rangle}(\mathsf{st}_O, a, y) \to 1\right]\right| = \widetilde{O}\left(\sqrt[3]{\frac{ST^2}{KN} + \frac{T^2}{N}}\right),$$

where y is uniform in $[M]$.

We need the following auxiliary lemma for proving Theorem 4.3.1, which can be seen as a security bound for a quantum average case version of Yao's box problem [Yao90], thus could be of an independent interest. We note that the classical average case version was proven in [DTT10, Lemma 8.4] and quantum worst-case version was proven in [NABT15, Theorem 1], neither of which suffices for our purpose.

Lemma 4.3.2. *Let $F = \mathsf{Func}([N], \{0,1\})$ be a set of functions. Let $f_x : [N] \to \{0,1\}$ such that $f_x(x') = f(x')$ for $x' \neq x$ and $f_x(x) = 1$. Suppose that A is an oracle-aided quantum algorithm that takes an S-bit classical advice st_f (that may depend on $f \in F$) as input, makes at most T oracle queries, and satisfies*

$$\Pr_{A,x}[A^{|f_x\rangle}(\mathsf{st}_f, x) \to f(x)] \geq \frac{1}{2} + \varepsilon$$

for all $f \in F$. Then it holds that $\varepsilon^6 = \widetilde{O}(ST^2/N)$. Or more strongly it holds that $\varepsilon^3 = \widetilde{O}(ST/N)$.

This lemma can be proven similarly to its classical counterpart in [DTT10, Lemma 8.4] while we need to deal with some issues as described in [HXY19]. The later part of theorem is from [CGLQ20]. we will use this version below. Theorem 4.3.1 can be proven as follows. Note that we use the double-sided O2H instead of the SC-O2H as in the original proof, which improves the lower bounds.

Proof of Theorem 4.3.1. Let f be a function $f_{(a,x)}$ We first sketch the outline of the proof by the following diagram:

$$
\begin{aligned}
& p_0 := \Pr_{A,f,a,x}[A^{|f\rangle}(\mathsf{st}_f, a, f(a,x)) \to 1] \\
\overset{\text{O2H+Lemma 4.2.4}}{\approx}\ & p_1 := \Pr_{A,f,a,x}[A^{|f_{(a,x)}\rangle}(\mathsf{st}_f, a, f(a,x)) \to 1] \\
\overset{\text{Lemma 4.3.2}}{\approx}\ & p_2 := \Pr_{A,f,a,y}[A^{|f_{(a,x)}\rangle}(\mathsf{st}_f, a, y) \to 1] \\
\overset{\text{O2H+Lemma 4.2.4}}{\approx}\ & p_3 := \Pr_{A,f,a,y}[A^{|f\rangle}(\mathsf{st}_f, a, y) \to 1].
\end{aligned}
$$

Step 1. $|p_0 - p_1| = \widetilde{O}\left(\sqrt[2]{\frac{ST^2}{KN} + \frac{T^2}{N}}\right)$

This is simply proven by using the double-sided O2H lemma (Lemma 2.1.4). This lemma states that there is a quantum oracle algorithm B with almost the same query complexity to A such that

$$
|p_0 - p_1| \leq \sqrt{\Pr_{B,O,a,x}\left[B^{|f\rangle,|f_{(a,x)}\rangle}(\mathsf{st}_f, a, f(a,x)) \to x\right]}
$$

holds. The advantage of B is $O(ST^2/KN + T^2/N)$ as shown in Lemma 4.2.4, which proves the result.

Step 2. $|p_2 - p_3| = \widetilde{O}\left(\sqrt[2]{\frac{ST^2}{KN} + \frac{T^2}{N}}\right)$

This is almost the same as Step 1; consider another algorithm A' that additionally takes but ignores $f(a,x)$ as input, and samples y uniformly at random at the initial stage. This can simulate $A(\mathsf{st}_f, a, y)$, and we can apply a similar argument for A'.

Step 3. $|p_1 - p_2| = \widetilde{O}\left(\sqrt[3]{\frac{ST}{KN}}\right)$

First, we consider an oracle-aided quantum algorithm B that uses A as a subroutine as follows.

$B^{|f\rangle}(\mathsf{st}_f, a, x, y)$**:** It runs $A^{|f\rangle}(\mathsf{st}_f, a, y)$ and outputs the output of A.

We assume that M is a power of 2 for simplicity. Yao's equivalence of pseudorandomness and unpredictability [Yao82] states that there is i such that there is an algorithm C that satisfies[3]

$$\Pr_{C,f,a,x}[C^{|f_{(a,x)}\rangle}(\mathsf{st}_f, a, x) \to F(i,a,x)] \geq \frac{1}{2} + \frac{\varepsilon}{\log M}$$

where $F(i,a,x)$ is the i-th bit of $f(a,x)$. Lemma 4.3.2 implies that it holds that $\varepsilon^3 = \widetilde{O}(ST/KN)$, which gives the result.

Combining three steps, we obtain $|p_0 - p_3| = \widetilde{O}\left(\sqrt[3]{\frac{ST^2}{KN} + \frac{T^2}{N}}\right)$. □

4.4 Post-quantum Primitives

We summarize some further results for post-quantum cryptographic primitives, where the adversary have quantum access to the random oracle but the specific part of oracle that is used for problem is only classically accessible. The following theorem shows that random oracles are secure post-quantum pseudorandom functions in the QROM-AI.

Theorem 4.4.1. *Let $O \in \mathsf{Func}([K]\times[N]\times[L],\{0,1\})$ be a random oracle. Suppose that A is an oracle-aided quantum algorithm that takes an S-bit classical advice st_O (that may depend on O) as input, and makes at most T (quantum) queries to the oracle O and at most Q classical queries to the other oracle. Then it holds that*

$$\left|\Pr_{A,O,a,k}\left[A^{|O\rangle,O(a,k,\cdot)}(\mathsf{st}_O, a) \to 1\right] - \Pr_{A,O,a,F}\left[A^{|O\rangle,F}(\mathsf{st}_O, a) \to 1\right]\right| = \widetilde{O}\left(\sqrt[2]{\frac{ST^2}{KN} + \frac{T^2}{N}} + Q\sqrt[6]{\frac{ST^2}{KN}}\right),$$

[3] Technically, we may add few bits as an advice.

where F is uniform in $\mathsf{Func}([L],\{0,1\})$.

The proof of Theorem 4.4.1 is very similar to the proof of Theorem 4.3.1, albeit requiring the following function variant of Lemma 4.3.2. The classical counterpart of this lemma is implicitly proven in [DGK17, Theorem 7] for a similar purpose.

Lemma 4.4.2. *Let $O \in \mathsf{Func}([K]\times[N]\times[L],\{0,1\})$ be a random oracle. For any oracle-aided quantum algorithm A with a set of S-bit classical advice $\{\mathsf{st}_O\}_O$ that makes at most T oracle queries to the oracle O satisfying*

$$\Pr_{A,O,a,k}[A^{|O\rangle,O(a,k,\cdot)}(\mathsf{st}_O,a,k)\to(m,t)\wedge t=O(a,k,m)]\geq\frac{1}{2}+\varepsilon,$$

where A has the query magnitude 0 for $\{(a,k,\cdot)\}$ to its first oracle and never queries m to its second oracle, we have

$$\varepsilon^6=O(ST^2/KN).$$

Finally, the following theorem shows that random oracles are secure post-quantum message authentication codes in the QROM-AI.

Theorem 4.4.3. *Let $O \in \mathsf{Func}([K]\times[N]\times[L],[M])$ be a random oracle. Suppose that A is an oracle-aided quantum algorithm that takes an S-bit classical advice st_O (that may depend on O) as input, and makes at most T oracle queries to the oracle O. Then it holds that*

$$\Pr_{A,O,a,k}\left[O(a,k,m)=t:A^{|O\rangle,O(a,k,\cdot)}(\mathsf{st}_O,a)\to(m,t)\right]\quad=\quad\widetilde{O}\left(\sqrt[2]{\frac{ST^2}{KN}+\frac{T^2}{N}+\frac{1}{M}}\right)$$

where A never queries m to its second oracle.

4.5 Discussion and Open Problems

We explore the various limitation of quantum preprocessing attack with the classical advice, or more formally in the QROM-AI. Most parts of this chapter is based

on [HXY19]. We improve several bounds using new techniques and some independent works. We summarize the differences and comparisons below. For a simpler discussion, we exclude the factor K. The *original* work means that [HXY19] below, and other references are explicitly noted.

- Section 4.2 describes the bounds for one-wayness style problems. Most results in the original paper (and [CLQ20]) has a similar bound for ε^2, but we remove the square factor using the bucketing argument, obtaining $\varepsilon = O(ST^2/N)$. The follow-up work [CGLQ20, GLLZ21] improves this bound to $\varepsilon = O(ST/N + T^2/N)$ using a different technique. We also note that Lemma 4.2.4 is different from the original result; the original paper considers the bound for semi-classical oracle algorithm, but this thesis considers the double-sided one-way to hiding lemma [BHH+19] instead of the semi-classical one-way to hiding lemma [AHU19].

- Section 4.3 provides the security bound of pseudorandom generators and Yao's box problem. The original bound is $\varepsilon^6 = O(ST^4/N)$, but we improve the bound to $\varepsilon^3 = O(ST^2/N)$ using the double-sided one-way to hiding lemma. Note that improving Lemma 4.2.4 to $O(ST/N + T^2/N)$ gives the bound $\varepsilon^3 = O(ST/N + T^2/N)$, which coincides the current best bound of [CGLQ20].

- Similarly, the bounds in Section 4.4 are also improved as well. To our knowledge, those results have not yet improved except the improvements in this thesis.

We also note that the other improvements have been made. In [CGLQ20], the authors prove that the preprocessing quantum adversary cannot find the collision of salted hash functions. On the other hand, a tighter classical bound for PRGs is proven in [GGKL21]. The following questions are natural open problems.

1. The QROM-AI lower bounds for pseudorandom functions and MACs are still open.

2. The random permutation lower bounds are usually much easier to prove, at least regarding our technique. Ironically, the tight inversion lower bound in [CGLQ20] cannot be extended to the permutation case, because they heavily rely on the compressed oracle technique of Zhandry [Zha19], whose permutation version is a big open problem.

3. Extending the tight classical bound of PRGs in [GGKL21] to the quantum setting should be interesting.

4. For the function inversion problem, the best attack is still variants of Hellman attack [Hel80], thus there is a gap between the attack trade-offs of $S^2T \approx N^2$ (or $S^3T^2 = N^3$ for quantum, as shown in Section 3.2) and the lower bounds of $ST + T^2 \approx N$. In fact, filling this gap should give a significant impacts on the other areas such as the circuit lower bounds as shown in [CGK18].

5. For the other problems, the exponent factors still have a significant gap between the attacks and the lower bounds.

The other natural question is to extend the attacks or lower bounds to the quantum advice setting. The main topic of the next chapter deals with this direction.

Chapter 5

Quantum Random Permutations with Quantum Advice

This chapter continues the study on the limitation of quantum preprocessing algorithms. The main focus of this chapter is the algorithm that takes the *quantum* advice.

The main result is the lower bound of the success probability of quantum inversion algorithm for random permutations with quantum auxiliary input. This answers the open problem raised by Nayebi et al. [NABT15], who proved the similar lower bound only for the classical auxiliary input.

This lower bound has an implication in complexity theory: An oracle separation of two complexity classes $\mathsf{NP} \cap \mathsf{coNP}$ and $\mathsf{BQP/qpoly}$. The latter is the class of problems solvable by a polynomial-time quantum algorithm with a polynomial-size quantum advice [NY04, Aar05]. Precisely, we prove that $\mathsf{NP} \cap \mathsf{coNP} \not\subseteq \mathsf{BQP/qpoly}$ relative to a random permutation oracle. This affirmatively answers the open problem left by Aaronson [Aar05], who showed the existence of an oracle relative to which $\mathsf{NP} \not\subseteq \mathsf{BQP/qpoly}$ and left it open to show the existence of an oracle relative to which $\mathsf{NP} \cap \mathsf{coNP} \not\subseteq \mathsf{BQP/qpoly}$.

Our proof technique is based on the incompressibility argument similar to the

previous chapter. At first glance, to extend the results in Chapter 4 seems to be straightforward. This turns out to be not the case; the quantum advice is in nature different from the classical advice string.

The first problem is the reusability of advice string; the quantum advice may be broken down and the algorithm with this broken advice may fail to solve the other problems. This situations collide to our proof that relies on the encoding-decoding paradigm, using the algorithm multiple times. In fact, the compression lemma (Lemma 4.2.2) is already problematic as it does not allow to include the quantum state as a part of encoding.

To prove the lower bounds of the algorithms with the quantum advice, we resolve such technical problems. We introduce the new compression lemma based on the quantitative versions of Holevo theorem [Hol73] for quantum encoding of classical data [Nay99, NS06]. On the other hand, we exploit the gentle measurement [Win99, Aar05, AR19] to avoid the destruction of the quantum state.

The model of adversary is defined similar to Section 4.1, but the offline algorithm may output the quantum state as the advice. We consider the random permutation as a main object in this chapter, which is due to another subtle issue that the hardness of simulation of the quantum advice algorithm.

Remark 8 (Follow-up work)**.** The follow-up work [CLQ20] shows that the function inversion problem for the random functions also has the similar bound. Later then, the recent work [CGLQ20] proves that a tighter bound of the function inversion problem, showing that either Grover's algorithm and the ideal preprocessing attack is the best. Note that any improvement of the lower bound or the attack would give a breakthrough in the other fields [CGK18].

Remark 9 (Improvements)**.** The theorems in this chapter are slightly improved upon the previous results in [HXY19]. Theorem 5.1.1 has much better exponents, even compared to the corresponding bounds for random functions [CGLQ20]. This improvement is based on the bucketing arguments, and the same bound could be achieved using a similar argument for [CLQ20].

5.1 Bound for Inverting Random Permutations

We describe the main theorem in this chapter in detail. The main object here is the set of keyed permutations $\mathsf{KeyPerm}([K],[N])$ defined by

$$\{f : [K] \times [N] \to [N] \text{ such that } f(K,\cdot) : [N] \to [N] \text{ is a permutation}\}.$$

In particular, we prove that the inversion problem for a random keyed permutation is intractable for any polynomial time algorithm with a polynomial size advice. More quantitatively, we prove the following theorem. Here we assume that the advice is a pure state, but this does not lose any generality due to the purification.

Theorem 5.1.1. *Let $O \in \mathsf{KeyPerm}([K],[N])$ be a random keyed permutation. Suppose that A is an oracle-aided quantum algorithm that takes an S-bit quantum advice $|\mathsf{st}_O\rangle$ (that may depend on O) as input, makes at most T oracle queries, and satisfies*

$$\Pr_{A,O,a,x}\left[A^{|O\rangle}(|\mathsf{st}_O\rangle, a, O(a,x)) \to x\right] = \varepsilon.$$

Then it holds that $\varepsilon = \widetilde{O}\left(\sqrt{\frac{ST^2}{KN}} + \frac{T^2}{N}\right)$.

The strategy is to compress the set $\mathsf{KeyPerm}([K],[N])$ as in Chapter 4. As described above, we need new tools for dealing with the quantum advice, which we summarize in the next section.

5.2 Preparation

5.2.1 Compression Lemmas

We start with the quantum compression lemma that asserts the limitation of compression procedure of classical messages to the quantum state. Intuitively, the power of such quantum compression is not that stronger than the classical compression; the theorem of Holevo [Hol73] roughly states that we can only retrieve

the n-bit information from the n-qubit quantum states, using the language of quantum information theory.

The following theorem [Nay99, NS06] of Nayak and Salzman gives a quantitative version concerning the success probability of retrieval.

Theorem 5.2.1. [Nay99, NS06, adapted] *Suppose that Alice holds a string $x \in \mathcal{M}$ and wants to convey it to Bob via a (noiseless) quantum channel. For any (possibly two-way interactive) protocol, for any $x \in \mathcal{M}$, if the probability that Bob successfully recovers x is at least $p \in (0, 1]$, then the number of qubits m transmitted by Alice is at least $\log |\mathcal{M}| - \log 1/p$.*

Note that this theorem can be applied to the compression scenario, by choosing the encoder as Alice and the decoder as Bob, and send the encoded quantum state from Alice to Bob. This interpretation gives us to obtain the following compression lemma.

Lemma 5.2.2 (Quantum compression lemma)**.** *Let $\mathcal{M}, \mathcal{R}$ be a set. Let E be a procedure that takes $(x, r) \in \mathcal{M} \times \mathcal{R}$ and outputs a m-qubit quantum state and D a procedure that takes a quantum state along with string $r \in \mathcal{R}$. If we have*

$$\Pr_r[D(E(x, r), r) = x] \geq p$$

for all $x \in \mathcal{M}$, then it holds that $m \geq \log |\mathcal{M}| - 2 \log 1/p$.

Proof. By the standard averaging argument, there exist an $r_0 \in \mathcal{R}$ and a set $\mathcal{M}' \subset \mathcal{M}$ with $|\mathcal{M}'| \geq p|\mathcal{M}|$ such that $\Pr[D(E(x, r_0), r_0) = x] \geq p$ for all $x \in \mathcal{M}'$. We then apply Theorem 5.2.1 on $D'(\cdot) = D(\cdot, r_0)$ and $E'(\cdot) = E(\cdot, r_0)$ and the set $\mathcal{M}'$, we obtain the desired result as follows:

$$m \geq \log |\mathcal{M}'| - \log 1/p \geq \log(p \cdot |\mathcal{M}|) - \log 1/p = \log |\mathcal{M}| - 2 \log 1/p$$

□

5.2.2 Gentle Measurement

The next tool we use is the so-called almost-as-good-as-new lemma [Aar05] that is closely related to the gentle measurement lemma [Win99]. This lemma states that if the measurement basis is extremely close to the original state, then the measurement gently destructs the state so that we can construct another quantum state that is almost as good as new original state.

Lemma 5.2.3 ([Aar05, Lemma 2.2])**.** *Let ρ be a (mixed) quantum state and a 2-outcome measurement of ρ yields the outcome 1 with probability at least $1-\varepsilon$. Then we can recover another state ρ' such that $\mathrm{tr}(\rho,\rho') \leq \sqrt{\varepsilon}$ after the measurement where $\mathrm{tr}(\rho,\rho')$ denotes the trace distance between ρ and ρ'.*

For a sequence of gentle measurements, we can apply the procedure in the above lemma sequentially. This composition of the procedure also gives an almost as good as new state as follows.

Lemma 5.2.4 ([AR19, Corollary 16])**.** *Let ρ be a mixed state and let $S_1,\ldots,S_m$ be quantum operations. Suppose that for all i, we have*

$$\mathrm{tr}(S_i(\rho),\rho) \leq \varepsilon_i.$$

Then

$$\mathrm{tr}(S_m(S_{m-1}(\cdots(S_1(\rho))),\rho) \leq \varepsilon_1+\cdots+\varepsilon_m.$$

Finally, we derive the following lemma using the above two theorems. This lemma states that if a quantum algorithm A with a quantum advice ρ can solve each problem instance with almost certainty, then this algorithm along with a single quantum state ρ can be used to solve all of the problem instances simultaneously. In fact, a similar fact is implicitly used in [Aar05].

Lemma 5.2.5. *Let ρ be any (mixed) quantum state, and n be any positive integer. Let (y_i,x_i) be the problem instance and the corresponding solution for each $i\in[n]$.*

If a unitary quantum algorithm A satisfies

$$\Pr\left[A(\rho, y_i) \to x_i\right] \geq 1 - \frac{1}{9n^4},$$

then there exists a quantum algorithm B such that

$$\Pr\left[B(\rho, y_1, ..., y_n) = \{x_1, ..., x_n\}\right] > 2/3.$$

Proof. We assume that $n \geq 2$, since the case of $n = 1$ is obvious. The algorithm B is defined as follows:

1. Set $\rho_0 = \rho$.
2. For each $i = 1, 2, \cdots, n$, do:
   (a) Run $A(\rho_{i-1}, y_i)$ and measure the output register to obtain x_i'.
   (b) Construct the state ρ_i using Lemma 5.2.3.

It suffices to show that the algorithm B successes to find $\{x_1, ..., x_n\}$ with probability at least $2/3$. Let S_i be the quantum operator that corresponds to the i-th loop in the execution of B. By Lemma 5.2.3, where we consider a projective measurement $(M_0 = I - |x_i\rangle\langle x_i|, M_1 = |x_i\rangle\langle x_i|)$, we have $\mathrm{tr}(S_i(\rho), \rho) \leq \frac{1}{3n^2}$. Lemma 5.2.4 implies that

$$\mathrm{tr}(\rho_i, \rho) = \mathrm{tr}(S_i(S_{i-1}(...S_1(\rho))), \rho) \leq \frac{i}{3n^2}.$$

Therefore we have

$$\Pr[x_i' \neq x_i] \leq \mathrm{tr}(\rho_i, \rho) + \frac{1}{9n^4} \leq \frac{i}{3n^2} + \frac{1}{9n^4}.$$

By union bound, we obtain

$$\Pr[x_i' = x_i \text{ for all } i \in [n]] \geq \frac{2}{3}.$$

□

5.3 Proof of Theorem

We now prove Theorem 5.1.1. The overall flow is similar to the proof of Theorem 4.2.1. The differences are from the several issues and their resolutions described in the previous section, and the fact that we consider the keyed permutation here.

Transform the advantage of adversary. The first step is to transform the advantage to the easier form for compression. However, as we will see, we need to amplify the success probability only using the parallel repetition instead of the amplitude amplification because of the quantum advice. Instead, we take into account the query depth in the middle of proof.

We assume that the adversary takes S_0-qubit quantum advice and makes T_0 oracle queries such that

$$\Pr_{A,O,a,x}\left[A^{|O\rangle}\left(|\mathsf{st}_O\rangle, a, O(a,x)\right) \to x\right] = \varepsilon_0.$$

The bucketing argument ensures that there is $j = O(\log(1/\varepsilon_0))$ such that the following inequality holds.

$$\Pr_{O,a,x}\left[\Pr_{A}[A^{|O\rangle}\left(|\mathsf{st}_O\rangle, a, O(a,x)\right) \to x] \geq 2^{-j}]\right] \geq \frac{2^j\varepsilon_0}{4L}$$

We consider another algorithm B works as follows.

1. Run $\Theta(2^j)$ copies of A in parallel except the final measurements.
2. Check the correctness of outputs of each A by querying them to O.
3. Output x if there is a correct answer x, and $\perp$ otherwise.

We again stress that the algorithm B outputs either x or $\perp$. The number and depth of queries of B are $T = \Theta(2^j T_0)$ and $D = T_0 + 1$, respectively, and the new quantum advice $\left|\widetilde{\mathsf{st}}_O\right\rangle$ takes $\Theta(2^j S_0)$-qubit. This new algorithm satisfies

$$\Pr_{O,a,x}\left[\Pr_{B}[B^{|O\rangle}\left(\left|\widetilde{\mathsf{st}}_O\right\rangle, a, O(a,x)\right) \to x] \geq 3/4\right] \geq \frac{2^j\varepsilon_0}{4L} =: \varepsilon. \tag{5.1}$$

Note that $ST/\varepsilon^2 = \widetilde{\Theta}(S_0T_0/\varepsilon_0^2)$ and $T/\varepsilon = \widetilde{\Theta}(T_0/\varepsilon_0)$. In what follow, we will show that any adversary that satisfies Equation (5.1) should follows the trade-offs $\varepsilon = \widetilde{O}(STD/KN + TD/N)$. This implies that

$$\varepsilon_0 = \widetilde{O}\left(\sqrt{\frac{S_0T_0^2}{KN}} + \frac{T^2}{N}\right)$$

which is what we desired. Therefore, we consider the algorithm A such that Equation (5.1) holds below instead of the original adversary.

Algorithm with partial data. Now we describe how the algorithm A with T queries, D depths and S-qubit quantum advice satisfying Equation (5.1) successfully finds the inversion only using a partial data. By the standard averaging argument, there exists a set of functions F that is an $\varepsilon/2$-fraction of functions $f \in \mathsf{Perm}([K],[N])$ such that

$$\Pr_{a,x}\left[\Pr_{A}[A^{|f\rangle}\left(\left|\mathsf{st}_f\right\rangle, a, f(a,x)\right) \to x] \geq 3/4\right] \geq \varepsilon/2.$$

We say such a function f is nice. We fix a nice function f and consider it as the target of encoding. Let I be a set of $(a,x) \in [K]\times[N]$ such that the following inequality holds. The above inequality gives the lower bound of $|I| \geq \varepsilon KN/2$.

$$\Pr_{A}\left[A^{|f\rangle}\left(\left|\mathsf{st}_f\right\rangle, a, f(a,x)\right) \to x\right] \geq 3/4$$

We consider the following algorithm B that outputs a binary value for invoking the one-way to hiding lemma.

$B^{|f\rangle}\left(\left|\mathsf{st}_f\right\rangle, a, x, y\right)$: It runs $A^{|f\rangle}\left(\left|\mathsf{st}_f\right\rangle, a, y\right)$. Then B outputs 1 if the answer z of the algorithm A satisfies $z = x$, and outputs 0 otherwise.

For a nice function f and $(a,x) \in I$, the inequality for A can be rephrased by

$$\Pr_{B}\left[B^{|f\rangle}\left(\left|\mathsf{st}_f\right\rangle, a, x, f(a,x)\right) \to 1\right] \geq 3/4. \tag{5.2}$$

We now choose a random set $R \subset [K] \times [N]$ so that each $(a, x) \in [K] \times [N]$ is included in R with probability $p = b/T(D+1)$ for a constant b, independently for each (a, x). We say that $(a, x) \in I$ is good if both of the following conditions hold.

$$(A)\ (a,x) \in R, \qquad (B)\ \Pr_B[\mathsf{Find} : B^{|f\rangle\backslash(R\backslash\{(a,x)\})}(\left|\mathsf{st}_f\right\rangle, a, x, f(a,x))] \leq \frac{c}{D+1}$$

Here $c \leq 1/576$ is a constant to be specified later. We denote a set of good elements by G. The following claim ensures the minimum size of $|G|$ with a high probability similar to Claim 1. We postpone to prove this claim at the end of this section.

Claim 2. $\Pr_R[|G| \geq \delta\varepsilon KN/TD] \geq 0.8$ *for some constant* $\delta > 0$.

We fix a *good* randomness R such that $|G| \geq \delta\varepsilon^2 KN/T^2$. For $y \in [N]$, we define a function $g_y : [K] \times [N] \to [N]$ by

$$g_y(a,z) = \begin{cases} f(a,z) & \text{if } (a,z) \notin R, \\ y & \text{otherwise.} \end{cases}$$

We note that g_y agrees with f except $R \setminus \{(a,x)\}$ for a preimage (a,x) of y (i.e., $f(a,x) = y$). By Lemma 2.1.2 and Remark 1, for any $(a,x) \in G$, we have

$$\left|\Pr_B\left[B^{|g_{f(a,x)}\rangle}\left(|\mathsf{st}_f\rangle, a, x, f(a,x)\right) \to 1\right] - \Pr_B\left[B^{|f\rangle}\left(|\mathsf{st}_f\rangle, a, x, f(a,x)\right) \to 1\right]\right|$$
$$\leq 2\sqrt{(D+1)\Pr_B\left[\mathsf{Find} : B^{|f\rangle\backslash(R\backslash\{(a,x)\})}\left(\left|\mathsf{st}_f\right\rangle, a, x, f(a,x)\right)\right]} \leq 2\sqrt{c} \leq \frac{1}{12}.$$

Thus we have

$$\Pr_B[B^{|g_{f(a,x)}\rangle}(|\mathsf{st}_f\rangle, a, x, f(a,x)) \to 1] \geq \frac{3}{4} - \frac{1}{12} = \frac{2}{3},$$

which is equivalent to

$$\Pr_A[A^{|g_{f(a,x)}\rangle}(|\mathsf{st}_f\rangle, a, f(a,x)) \to x] \geq \frac{2}{3},$$

Since A outputs either x or $\perp$, we can amplify the success probability by running $r = O(\log(KN))$ copies of A in parallel and taking any output that is not $\perp$ as

the final output, if any. We call this algorithm by C, and for brevity we write the $O(\log(KN))$ copies of $\left|\mathsf{st}_f\right\rangle$ by $\left|\overline{\mathsf{st}}_f\right\rangle$. The algorithm C satisfies

$$\Pr_C[C^{|g_{f(a,x)}\rangle}(|\overline{\mathsf{st}}_f\rangle, a, f(a,x)) \to x] \geq 1 - \frac{1}{9(KN)^4}.$$

Encoding. Now we are ready to encode a nice function f for good R. Let

$$R_a := R \cap (\{a\} \times [N]), \text{ and } G_a = G \cap (\{a\} \times [N]).$$

The encoding of f includes the following information:

- The advice string $|\overline{\mathsf{st}}_f\rangle$: $O(S \log(KN))$ qubits.
- The set $f(R_a)$ for each $a \in [K]$: $\sum_a \log \binom{N}{|R_a|}$ bits.
- The values of f on $(\{a\} \times [N]) \setminus R_a$ for each $a \in [K]$: $\sum_a \log(N - |R_a|)!$ bits.
- The cardinality of G_a for each $a \in [K]$: $K \log N$ bits.
- The set $f(G_a)$ for each $a \in [K]$: $\sum_a \log \binom{|R_a|}{|G_a|}$ bits.
- The values of f on $R_a \setminus G_a$: $\sum_a \log(|R_a| - |G_a|)!$ bits.

Decoding. The decoding procedure initializes an empty table to store the values of f and then fills the table as follows:

1. Recover $\left|\overline{\mathsf{st}_f}\right\rangle$, G_a, and G.
2. Fill the values of f on inputs in $([K] \times [N]) \setminus R$. This can be done since the decoder knows R as a shared random string.
3. Fill the table of f for G by the following procedures. For each $(a, y) \in f(G_a)$, let $x \in [N]$ be the inversion of y at a, i.e., $y = f(a, x)$ (which is unknown to the decoder so far). Note that the function g_y can be evaluated by the

decoder since it only needs values of f on $([K] \times [N]) \setminus R$ which is already recovered. As discussed above, we have

$$\Pr_{\widetilde{B}}[\widetilde{B}^{|g_{f(a,x)}\rangle}(|\overline{\mathsf{st}}_f\rangle, a, f(a,x)) \to x] \geq 1 - \frac{1}{9(KN)^4}.$$

Then the decoder uses the procedure in Lemma 5.2.5 to recover x for all $(a,y) \in f(G)$. Noting that $|f(G)| \leq KN$, by Lemma 5.2.5, the decoder succeeds in correctly recovering x for all $(a,y) \in f(G)$ with probability at least $2/3$. We note that the set G is also recovered at this point.

4. The decoder fills the values of f on inputs in $R \setminus G$ by using the partial truth table and the description of G that is recovered in the previous step.

The decoding procedure succeeds with a constant probability (over the choice of R and the randomness of measurements) since a constant fraction of R is good and the decoding succeeds with a constant probability for good R.

The size of encoding. Note that $|G| = \Omega(\varepsilon KN/TD)$ and $\sum_a |G_a| = |G|$. The overall encoding size except the size of advice string and the size of G_a is

$$\begin{aligned}
&\sum_{a\in[K]} \left(\log \binom{N}{|R_a|} + \log(N - |R_a|)! + \log \binom{|R_a|}{|G_a|} + \log(|R_a| - |G_a|)! \right) \\
&= \sum_{a\in[K]} \log \left(\frac{N!}{(N-|R_a|)!|R_a|!} \cdot (N-|R_a|)! \cdot \frac{|R_a|!}{(|R_a|-|G_a|)!|G_a|!} \cdot (|R_a|-|G_a|)! \right) \\
&= K \log N! - \sum_{a\in[K]} \log |G_a|! \\
&\leq K \log N! - \sum_{a\in[K]} |G_a| \log(|G_a|/e) \leq K \log N! - |G| \log \left(\frac{|G|}{eK} \right),
\end{aligned}$$

where we used the fact that $n! \geq (n/e)^n$ and $x \log x$ is convex in the last two inequalities. Then by Lemma 5.2.2, we obtain the inequality

$$O\left(S \log(KN) + K \log N\right) \geq |G| \log \left(\frac{|G|}{eK} \right) + \Theta(1).$$

Then we have either $|G|/eK < 2$, which implies $\varepsilon = O(TD/N)$, or

$$O\left(S\log(KN) + K\log N\right) \geq |G| \geq \delta\varepsilon KN/TD.$$

Combining them, we obtain $\varepsilon = \widetilde{O}\left(\frac{STD}{KN} + \frac{TD}{N}\right)$. As discussed before, this completes the proof.

Proof of Claim 2. Recall that the conditions that $(a,x) \in R$ with probability $p = b/T(D+1)$ for any $(a,x) \in [K]\times[N]$ and $G = G(R)$ is defined by the set of x satisfying the following conditions.

$$(A)\ (a,x) \in R, \qquad (B)\ \Pr_B[\mathsf{Find} : B^{|f\rangle\backslash(R\backslash\{(a,x)\})}\left(\left|\mathsf{st}_f\right\rangle, a, x, f(a,x)\right)] \leq \frac{c}{D+1}$$

Also note that $|I| \geq \varepsilon KN/2$. We need to prove that $\Pr_R[|G| \geq \delta\varepsilon KN/TD] \geq 0.8$ for some constant $\delta > 0$.

Let H be a subset of I that consists of all elements satisfying (A). Since the event $(a,x) \in R$ is independent for each (a,x), we have $\mathbb{E}[|H|] = p|I|$. The Chernoff bound implies that the following inequality holds.

$$\Pr_R\left[|H| \geq p|I|/2\right] \geq 1 - \exp(-p|I|/8) \geq 0.9 \tag{5.3}$$

Let $p_{\mathsf{Find}} := \Pr_{R,B}[\mathsf{Find} : B^{|f\rangle\backslash R}\left(\left|\mathsf{st}_f\right\rangle, a, x, f(a,x)\right)]$. Lemma 2.1.3 states that the following inequality holds.

$$p_{\mathsf{Find}} \leq 4T \cdot \max_{x\in[N]} \Pr[x \in R] = 4T \cdot p = \frac{4b}{D+1}$$

The Markov inequality implies the following inequality. Note that the second inequality is obvious since the punctured set is decreased.

$$\frac{4b}{c} \geq \Pr_R\left[p_{\mathsf{Find}} \geq \frac{c}{D+1}\right] \geq \Pr_R\left[\mathsf{Find} : B^{|f\rangle\backslash(R\backslash\{(a,x)\})}\left(\left|\mathsf{st}_f\right\rangle, a, x, f(a,x)\right)] \geq \frac{c}{D+1}\right]$$

Let J be a subset of I that consists of all elements satisfying (A) but not (B). Note that two events (A) and (B) are independent since (A) only depends on if $x \in R$

and (B) only depends on if the other points (i.e. that are in $[K] \times [N] \setminus \{(a, x)\}$) are in R. Therefore, for any $(a, x) \in I$, we have

$$\Pr_R [(a, x) \in J] \leq 4b/c \cdot p = 4b^2/cT(D+1).$$

This gives $\mathbb{E}[|J|] \leq 4b^2|I|/cT(D+1)$, and the Markov inequality gives

$$\Pr_R \left[|J| \leq \frac{40b^2|I|}{cT(D+1)} \right] \geq 0.9.$$

This inequality combined with Equation (5.3) gives that the lower bound of $|G|$ holds as follows with probability at least 0.8.

$$|G| = |H| - |J| \geq \frac{b|I|}{2T(D+1)} - \frac{40b^2|I|}{cT(D+1)} = \Omega\left(\frac{\varepsilon N}{TD}\right)$$

if $40b/c < 1/2$. Overall, Claim 2 holds by setting $80b < c < 1/36$.

5.4 Implication in Complexity Theory

In this section, we extend the implication of the result beyond cryptography. Namely, we have an oracle separation between two complexity classes. We denote by $\mathsf{BQP/qpoly}$ the class of languages that can be decided in quantum polynomial time with a polynomial-size quantum advice.[1]

Theorem 5.4.1. $\mathsf{NP} \cap \mathsf{coNP} \not\subseteq \mathsf{BQP/qpoly}$ *relative to a random permutation oracle with probability* 1.

This theorem resolves an open question posed by Aaronson [Aar05], who proved a slightly weaker separation of $\mathsf{NP} \not\subseteq \mathsf{BQP/qpoly}$. On the other hand, this theorem strengthens the previous separation of $\mathsf{NP} \cap \mathsf{coNP} \not\subseteq \mathsf{BQP}$ relative to a random permutation oracle with probability 1 by Bennett, Bernstein, Brassard, and Vazirani [BBBV97]. In fact, our proof is inspired by their proof.

[1] This class was originally introduced by Nishimura and Yamakami [NY04] with the name $\mathsf{BQP/{}^*Qpoly}$, and renamed to $\mathsf{BQP/qpoly}$ by Aaronson [Aar05]. See these papers for the detailed definition.

Proof of Theorem 5.4.1. Let $O = \{O_n : \{0,1\}^n \to \{0,1\}^n\}_{n\in\mathbb{N}}$ be a function such that O_n is a permutation for any n, and $O(x) = O_n(x)$ for $x \in \{0,1\}^n$. We denote the set of such functions by Perm and let $O^{-n} := \{O_{n'}\}_{n'\in\mathbb{N}\setminus\{n\}}$. We denote the subset of Perm such that consists of functions whose values on $\{0,1\}^{n'}$ match $O_{n'}$ for all $n' \in \mathbb{N} \setminus \{n\}$ by $\mathsf{Perm}[O^{-n}]$.

Relative to O, we consider a language

$$\mathcal{L}^O = \{(y,z) : \exists x \text{ s.t. } O(x) = y \wedge x \leq z\}$$

where $\leq$ means the inequality in the lexicographical order. We denote the restriction of $\mathcal{L}$ on $(\{0,1\}^n)^2$ by $\mathcal{L}^O_n := \mathcal{L}^O \cap (\{0,1\}^n)^2$. For any (y,z), there is a unique x such that $O(x) = y$, and this x can be used as a witness to verify if $O(x) = y$ and $x \leq z$. In other words, there always exists a witness for both YES and NO instances, so that $\mathcal{L}^O$ is included both of NP^O and coNP^O. What is left is to prove $\mathcal{L}^O \notin \mathsf{BQP}^O/\mathsf{qpoly}$ with probability 1 over the choice of $O \leftarrow \mathsf{Perm}$.

Here we say that $M^{|O\rangle}(|\mathsf{st}\rangle, \cdot)$ decides $\mathcal{L}^O_n$ if

$$\Pr_M[M^{|O\rangle}(|\mathsf{st}\rangle, (y,z)) = \mathcal{L}^O(y,z)] > 2/3 \tag{5.4}$$

for all $(y,z) \in (\{0,1\}^n)^2$, where we define

$$\mathcal{L}^O(y,z) = \begin{cases} 1 & \text{if } (y,z) \in \mathcal{L}^O, \\ 0 & \text{otherwise.} \end{cases}$$

Now we show that an arbitrary algorithm possibly taking advice that decides $\mathcal{L}^O$ can be used to construct an algorithm to solve the inversion problem for permutations, given that the advice size and the number of queries are sufficiently large. Then we will call Theorem 5.1.1 to complete the proof.

Let M be an oracle-aided quantum Turing machine which makes at most $T(n)$ queries to the oracle and takes at most $S(n)$-qubit quantum advice when its input length is $2n$ bits. We first show that for all sufficiently large n and any fixed $O^{-n} =$

$\{O_{n'}\}_{n' \in \mathbb{N} \setminus \{n\}}$, we have

$$(*) \quad \Pr_{O \leftarrow \mathsf{Perm}[O^{-n}]}[\exists \left|\mathsf{st}\right\rangle \in \mathcal{H}^{\otimes S(n)},\ M^{\left|O\right\rangle}(\left|\mathsf{st}\right\rangle, \cdot) \text{ decides } \mathcal{L}_n^O] < 1/2$$

where $\mathcal{H}^{\otimes S(n)}$ denotes the set of all $S(n)$-qubit quantum states.

For the sake of contradiction, suppose that the above claim $(*)$ is false. Without loss of generality, we can assume that there exists a fixed choice of O^{-n} and $S(n)$-qubit quantum state $\left|\mathsf{st}_O\right\rangle$ that depend on O such that $M^O(\left|\mathsf{st}_O\right\rangle, \cdot)$ decides $\mathcal{L}_n^O$ for at least $1/2$-fraction of $O \in \mathsf{Perm}[O^{-n}]$.

Since Equation (5.4) holds for any (y, z), we can amplify the success probability of M by repeating $r = O(n)$ times using the r copies of $\left|\mathsf{st}_O\right\rangle$, so that we have the following algorithm B.

$$\Pr_B[B^{\left|O\right\rangle}(\left|\mathsf{st}\right\rangle^{\otimes r}, (y, z)) = \mathcal{L}^O(y, z)] > 1 - \exp(-n)$$

Then, given any $y = O(x)$, we can find x by using the binary search that invokes the algorithm B $O(n)$ times. In other words, we can construct an algorithm A that makes $T(n) \cdot r \cdot O(n) = T(n) \cdot \mathsf{poly}(n)$ queries to O and takes $S(n) \cdot \mathsf{poly}(n)$-qubit advice such that

$$\Pr_{x,A}[A^O\left(\left|\overline{\mathsf{st}}_O\right\rangle, O(x)\right) = x] = 1 - \mathrm{poly}(n) \cdot \exp(-n)$$

for at least $(1/2)$-fraction of $O \in \mathsf{Perm}[O^{-n}]$. However, this advantage of A clearly contradicts Theorem 5.1.1, given that $S(n) \cdot T(n)^2 = 2^{n - \omega(\log n)} = N$. Therefore, we conclude that $(*)$ holds.

Note that for O sampled from Perm, O_n and $O_{n'}$ are independent for $n \neq n'$. Thus using $(*)$ for each sufficiently large $n \in \mathbb{N}$, we conclude that for any quantum machine M that makes $T(n)$ queries and takes $S(n)$-qubit advice, we have

$$\Pr_{O \leftarrow \mathsf{Perm}}[\forall n \in \mathbb{N}, \exists \left|\mathsf{st}_n\right\rangle \in \mathcal{H}^{\otimes \mathrm{poly}(n)},\ M^{\left|O\right\rangle}(\left|\mathsf{st}_n\right\rangle, \cdot) \text{ decides } \mathcal{L}_n^O] = 0.$$

Since the number of such Turing machine is countable and the union of countable number of probability 0 events has probability 0, we have

$$\Pr_{O \leftarrow \mathsf{Perm}}[\exists M, \forall n \in \mathbb{N}, \exists \left|\mathsf{st}_n\right\rangle \in \mathcal{H}^{\otimes \mathrm{poly}(n)},\ M^{\left|O\right\rangle}(\left|\mathsf{st}_n\right\rangle, \cdot) \text{ decides } \mathcal{L}_n^O] = 0.$$

This means that $\mathcal{L}^O \notin \mathsf{BQP}^O/\mathsf{qpoly}$ with probability 1. □

5.5 Discussion and Open Problems

In this chapter, we show that no efficient preprocessing algorithm, even with a quantum advice, cannot solve the inversion problem for random permutations.

The follow-up work [CLQ20] resolves the same problem for the random function case, and later the better bound of $ST \approx N$ for $\varepsilon = \Theta(1)$ is proven for random functions in [CGLQ20]. We present some questions regarding the current state of affairs.

- Can we prove the tighter lower bound for random *permutations* as $ST \approx N$? This is indeed an intriguing question, as it gives the exact complexity of permutation inversion problem, regarding the Hellman attack [Hel80] gives such a trade-off $ST \approx N$ for random permutations. On the other hand, the function inversion problem still has a gap between the lower bound and the best attack, which is believed to be hard to close [CGK18].

- What is the exact exponent of the advantage? Our lower bound gives $\varepsilon^2 = O(ST^2/N)$ and the bound in [CGLQ20] gives $\varepsilon^3 = O(ST/N)$. We believe that the bound in [CGLQ20] can be improved to ε^2, but we are not aware of the corresponding attack that gives $\varepsilon^2 = (\mathsf{poly}(S,T))/N$ or any way to reduce the exponent of the lower bound. More generally, is such a gap between the average advantage over the instances and for the number of easy instances the general property of quantum advice? In fact, this problem is related to the *reuse* of quantum advice, which is perhaps a most basic but nontrivial property of quantum states. This problem was also introduced in [CGLQ20].

We give a corollary: A oracle separation of $\mathsf{BQP}/\mathsf{qpoly}$ and $\mathsf{NP} \cap \mathsf{coNP}$. In fact, the proof shows a lot more stronger result: For any $T(n)$ and $S(n)$ such that

$T(n)^2 \cdot S(n) = 2^{n-\omega(\log n)}$, we have that $\mathsf{BQPTIME}(T(n))/\mathsf{q}(S(n))$, which is the class of problems solvable by a $T(n)$-time machine that takes a $S(n)$-qubit quantum advice. In the complexity-theoretic view, we leave the following questions, which might be of independent interests.

- Is there an oracle separation between $\mathsf{BQP/qpoly}$ and $\mathsf{BQP/poly}$? In other words, does the quantum advice gives any strong power over the classical advice? This problem was also suggested in [Aar05] and the quantum oracle separation is given in [AK07]. An interesting direction is to prove the inseparability of them relative to random oracles.

- Similarly, can we separate QMA from QCMA using the classical oracle? The current separations are based on the quantum oracles [AK07] or the in-place oracles [FK18].

- What about the space-bounded complexity instead of advice? The incompressibility arguments may not be applicable. The major open problem is the time-space trade-offs for collision problems, also suggested in [Aar21]. Note that the space-bounded complexity of learning of quantum systems and dynamics have recently been proven [CCHL22].

Part II

Quantum Cryptography: Public-key Encryptions and Bit Commitments

Chapter 6

Equivalence Theorem

This chapter is devoted to describe the equivalence theorem, a folklore result in quantum mechanics and formally proved in [AAS20] with its tightness. Roughly speaking, for two orthogonal states $|x\rangle$ and $|y\rangle$, this theorem states that if we can distinguish two orthogonal states $|\psi\rangle = \frac{|x\rangle+|y\rangle}{\sqrt{2}}$ and $|\phi\rangle = \frac{|x\rangle-|y\rangle}{\sqrt{2}}$, then we can also swap $|x\rangle$ and $|y\rangle$ with a slightly larger circuit, meaning that to bring $|x\rangle$ to $|y\rangle$ and vice versa. The converse direction also holds with some additional conditions.

We explain the meaning of this equivalence via the infamous experiment of Schrödinger's cat. Let x = Alive and y = Dead be the possible state of cat. The above description suggests that to distinguish between $|\text{Alive}\rangle \pm |\text{Dead}\rangle$ is as hard as to revive the cat, i.e., to bring $|\text{Dead}\rangle$ to $|\text{Alive}\rangle$. In particular, this implies that we cannot realize that the cat is quantum or classical (as it is measured), unless we can resurrect the dead cat by the convexity argument, i.e., to distinguish the classical mixture from the quantum state is a *necromancy-hard* problem.

While the original motivation of this theorem is from the fundamental theory of quantum physics and the quantum gravity, we find that this theorem can be interpreted as an interesting tool from cryptography, namely a search-to-decision reduction. Search-to-decision reductions have an important role in cryptography dating back to (at least) Goldreich-Levin theorem [GL89], and the oneway-to-

hiding lemmas [Unr15, AHU19] used in this thesis.

Roughly speaking, the task to swap can be seen as a search problem for $|\text{Alive}\rangle$ given $|\text{Dead}\rangle$ (and vice versa), and the corresponding distinguishing problem is apparently a decision problem. Therefore this theorem introduces a new search-to-decision reduction with almost no loss in the efficiency.

In the following sections, we will formalize and prove the equivalence theorem with some new notions that may be of independent interests. Then, we use this theorem to obtain quantum cryptographic results in the next two chapters.

6.1 Equivalence Theorem

We formalize the equivalence theorem in this section. Recall that $|x\rangle$ and $|y\rangle$ are two orthogonal quantum states. We define the *dual states* of $(|x\rangle, |y\rangle)$ by $(|\psi\rangle, |\phi\rangle)$ for $|\psi\rangle = \frac{|x\rangle+|y\rangle}{\sqrt{2}}$ and $|\phi\rangle = \frac{|x\rangle-|y\rangle}{\sqrt{2}}$. The theorem of [AAS20] states that the unitary map distinguishing $|x\rangle, |y\rangle$ is equivalent to the swapping unitary of their dual states. More generally, we consider the imperfect unitaries.

Definition 6.1.1. Let $|x\rangle$ and $|y\rangle$ be orthogonal quantum states.

- A Δ-*swapping unitary* U for $|x\rangle, |y\rangle$ such that
$$\frac{|\langle y| U |x\rangle + \langle x| U |y\rangle|}{2} \geq \Delta.$$
- A Δ-*distinguishing unitary* A for $|x\rangle, |y\rangle$ such that
$$|\Pr[A(|x\rangle) \to 1] - \Pr[A(|y\rangle) \to 1]| \geq \Delta$$

If there is an efficient Δ-swapping (or Δ-distinguishing) unitary for $|x\rangle$ and $|y\rangle$, then we say that $(|x\rangle, |y\rangle)$ is Δ-*swappable* (or Δ-*distinguishable*). If there is no such unitary, we say that $(|x\rangle, |y\rangle)$ is Δ-*swapping-hard* (or Δ-*indistinguishable*, respectively). If Δ is negligible, we omit the factor Δ.

Here, the notion of $A(\cdot) \to 1$ for unitary algorithm means that the first qubit in the computational basis is 1. For convenience, we sometimes write V_A to denote the unitary map and A to denote the algorithm that outputs the measurement result of the first qubit in the computational basis.

In the above definition, the ancilla register is allowed for the above unitaries only when they should be returned to all 0 qubits in the end. Looking ahead, this restriction prohibits to deal with the non-uniform algorithms.

Now we can introduce the formal statement of the imperfect equivalence theorem as follows.

Theorem 6.1.2 ([AAS20, Theorem 2])**.** *Let $|x\rangle, |y\rangle$ be two orthogonal states and $|\psi\rangle = \frac{|x\rangle+|y\rangle}{\sqrt{2}}$ and $|\phi\rangle = \frac{|x\rangle-|y\rangle}{\sqrt{2}}$ be their dual states. Then the Δ-swappability of $|x\rangle$ and $|y\rangle$ is equivalent to the Δ-distinguishability of $|\psi\rangle$ and $|\phi\rangle$. Equivalently, $|x\rangle$ and $|y\rangle$ are (Δ-)swapping-hard if and only if $|\psi\rangle$ and $|\phi\rangle$ are ($\Delta-$)indistinguishable. Moreover, the following two maps can be used to construct each others.*

- *Δ-swapping unitary U for $|x\rangle, |y\rangle$*
- *Δ-distinguishing unitary A for $|\psi\rangle, |\phi\rangle$*

Indeed, the Δ-swapping unitary can be built only using A and $A^\dagger$ once plus a single additional gate, and the Δ-distinguishing unitary can be built only using the controlled-U once plus $O(1)$ additional gates. If A (or U) does not act on some qubits, then the constructed U (A, respectively) also does not act on those qubits.

The final observation is not explicitly stated in the original paper. However, this property is clear from the constructions. We describe the constructions of U and A from the other below. The detailed proof of theorem is deferred to Section 6.3. Note that the proof presented here slightly differs from the original paper. We believe our proof is much intuitive than the original proof.

We first start from the swapping unitary. We consider the following unitary map $U' := e^{i\theta}U$ for θ such that

$$\mathrm{Re}(\langle y|\, U'\, |x\rangle + \langle x|\, U'\, |y\rangle) = |\langle y|\, U\, |x\rangle + \langle x|\, U\, |y\rangle|\,.$$

Then, A is constructed as in Figure 6.1, which is essentially the Hadamard test.

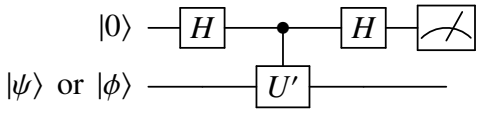

Figure 6.1: Quantum circuit for A

For the other direction, let V_A be the unitary part of A. The unitary map U from A is constructed as in Figure 6.2.

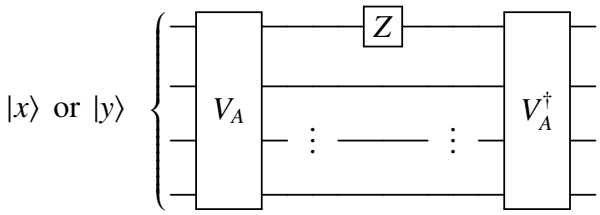

Figure 6.2: Quantum circuit for U

6.2 Non-uniform Equivalence Theorem

We extend the equivalence theorem to the non-uniform setting for cryptographic purpose. The non-uniform setting means that the algorithm may do some pre-processing so that it takes an ancillary state as an additional input, and uses it for solving the problem. As discussed before, the non-uniform adversary is more reasonable and stronger adversaries in cryptography.

The following lemma states that using an arbitrary swapping algorithm with an advice state, we can construct a new swapping algorithm that takes an advice state and then return the advice state without destroying it. Looking ahead, this lemma is used to relax the condition on ancillary registers in the equivalence theorem.

Lemma 6.2.1. *Let $U_{\mathbf{A},\mathbf{Z}}$ be a unitary map and $|\tau\rangle$ a quantum state such that*

$$\frac{\left\|\langle y|_{\mathbf{A}} U_{\mathbf{A},\mathbf{Z}} |x\rangle_{\mathbf{A}} |\tau\rangle_{\mathbf{Z}} + \langle x|_{\mathbf{A}} U_{\mathbf{A},\mathbf{Z}} |y\rangle_{\mathbf{A}} |\tau\rangle_{\mathbf{Z}}\right\|}{2} = \Delta.$$

Then there is another unitary map $\widetilde{U}_{\mathbf{A},\mathbf{Z}'}$ and $|\tau'\rangle$ such that

$$\frac{\left|\langle y|_{\mathbf{A}} \langle \tau'|_{\mathbf{Z}'} \widetilde{U}_{\mathbf{A},\mathbf{Z}'} |x\rangle_{\mathbf{A}} |\tau'\rangle_{\mathbf{Z}'} + \langle x|_{\mathbf{A}} \langle \tau'|_{\mathbf{Z}'} U_{\mathbf{A},\mathbf{Z}'} |y\rangle_{\mathbf{A}} |\tau'\rangle_{\mathbf{Z}'}\right|}{2} = \Delta^2.$$

Proof sketch. and $\mathbf{A}'$ be n-qubit registers, $\mathbf{Z}$ be an m-qubit register, and $\mathbf{B}$ be a single qubit register. Let $\mathbf{Z}' = (\mathbf{Z}, \mathbf{A}', \mathbf{B})$. We define a unitary $\widetilde{U}$ over $(\mathbf{A}, \mathbf{Z}, \mathbf{A}', \mathbf{B})$ as follows.

$$\widetilde{U} := X_{\mathbf{B}} (U_{\mathbf{A}',\mathbf{Z}})^{\dagger} U_{\mathbf{A},\mathbf{Z}} \tag{6.1}$$

Here $(U_{\mathbf{A}',\mathbf{Z}})^{\dagger}$ means the inverse of $U_{\mathbf{A}',\mathbf{Z}}$, which works similarly to $U_{\mathbf{A},\mathbf{Z}}$ except that it acts on $\mathbf{A}'$ instead of on $\mathbf{A}$. Let the new ancilla state

$$|\tau'\rangle_{\mathbf{Z}'} = \frac{|\tau\rangle_{\mathbf{Z}} |x\rangle_{\mathbf{A}'} |0\rangle_{\mathbf{B}} + |\tau\rangle_{\mathbf{Z}} |y\rangle_{\mathbf{A}'} |1\rangle_{\mathbf{B}}}{\sqrt{2}}. \tag{6.2}$$

The proof follows by a straightforward computation. □

Before presenting the non-uniform equivalence theorem, we should extend the notions for two orthogonal states. More precisely, we say that two orthogonal states are *non-uniform* (Δ-)swappable if there exists an efficient unitary map U that takes a polynomial size advice state $|\tau\rangle$ such that

$$\left\| \frac{\langle y| U |x\rangle |\tau\rangle + \langle x| U |y\rangle |\tau\rangle}{2} \right\| \geq \Delta.$$

The non-uniform Δ-distinguishable, swapping-hard, and indistinguishable are all defined in a similar way. We stress that the ancillary state, or advice, does not need ot be returned to the original state at the end of computation, which is clearly different from the original notions and the equivalence theorem.

In the remainder of this thesis, we omit *non-uniform* if we only work on the non-uniform setting for the convenience of notions and readability. When we work with the uniform algorithms (i.e., without taking advice), we explicitly note the uniformity.

The non-uniform equivalence theorem is as follows. All unitary maps and notions in this theorem are in the non-uniform setting.

Theorem 6.2.2 (Generalization of [AAS20, Theorem 2] with auxiliary states)**.** *Let $|x\rangle, |y\rangle$ be two orthogonal states and $|\psi\rangle = \frac{|x\rangle+|y\rangle}{\sqrt{2}}$ and $|\phi\rangle = \frac{|x\rangle-|y\rangle}{\sqrt{2}}$ be their dual states. Then the Δ-swappability of $|x\rangle$ and $|y\rangle$ implies the Δ^2-distinguishability of $|\psi\rangle$ and $|\phi\rangle$, and the Δ-distinguishability of $|\psi\rangle$ and $|\phi\rangle$ implies the Δ-swappability of $|x\rangle$ and $|y\rangle$. In particular, the swapping-hardness of $(|x\rangle, |y\rangle)$ and the indistinguishability of $(|\psi\rangle, |\phi\rangle)$ are equivalent.*

More precisely, the following statements hold.

- *A swapping unitary U with an auxiliary state $|\tau\rangle$ for $|x\rangle, |y\rangle$ such that*

$$\frac{\left\| \langle y|_{\mathbf{A}} U_{\mathbf{A},\mathbf{Z}} |x\rangle_{\mathbf{A}} |\tau\rangle_{\mathbf{Z}} + \langle x|_{\mathbf{A}} U_{\mathbf{A},\mathbf{Z}} |y\rangle_{\mathbf{A}} |\tau\rangle_{\mathbf{Z}} \right\|}{2} \geq \Delta.$$

  implies a distinguishing unitary A for $|\psi\rangle, |\phi\rangle$ with another auxiliary state $|\tau'\rangle$ such that

$$|\Pr[A(|\psi\rangle, |\tau'\rangle) \to 1] - \Pr[A(|\phi\rangle, |\tau'\rangle) \to 1]| \geq \Delta^2$$

- *A distinguishing unitary A with another auxiliary state $|\tau\rangle$ for $|\psi\rangle, |\phi\rangle$ such that*

$$|\Pr[A(|\psi\rangle, |\tau\rangle) \to 1] - \Pr[A(|\phi\rangle, |\tau\rangle) \to 1]| \geq \Delta$$

implies a swapping unitary U with an auxiliary state $|\tau\rangle$ such that

$$\frac{\left\| \langle y|_{\mathbf{A}} U_{\mathbf{A},\mathbf{Z}} |x\rangle_{\mathbf{A}} |\tau\rangle_{\mathbf{Z}} + \langle x|_{\mathbf{A}} U_{\mathbf{A},\mathbf{Z}} |y\rangle_{\mathbf{A}} |\tau\rangle_{\mathbf{Z}} \right\|}{2} \geq \Delta.$$

Proof. The second item is directly obtained from Theorem 6.1.2 by choosing $|x'\rangle = |x, \tau\rangle$ and $|y'\rangle = |y, \tau\rangle$. For the first item, we first apply Lemma 6.2.1 to obtain the swapping unitary map that preserves the auxiliary state. Then the result follows from the corresponding part of Theorem 6.1.2. □

6.3 Proof of Equivalence Theorem

We give a proof of the equivalence theorem (Theorem 6.1.2) in this section. The proof in this section differs from the original paper [AAS20]. Our proof is based on the *standard distinguishing algorithm* that may be of independent interest. While the proof presented in this section seems a bit lengthy, most parts are devoted to formalize the notion of standard distinguishing algorithms. With this notion, the proof is highly intuitive and does not involved with some garbage states.

A property of dual states. We begin with the following simple property of the dual state pairs $(|x\rangle, |y\rangle)$ and $(|\psi\rangle, |\phi\rangle)$. The straightforward computation proves this lemma, thus we omit the detailed proof.

Lemma 6.3.1. *Let $|x\rangle, |y\rangle$ be two orthogonal states and $|\psi\rangle = \frac{|x\rangle+|y\rangle}{\sqrt{2}}$ and $|\phi\rangle = \frac{|x\rangle-|y\rangle}{\sqrt{2}}$ be their dual states. For any unitary U, it holds that*

$$\langle x| U |y\rangle + \langle y| U |x\rangle = \langle \psi| U |\psi\rangle - \langle \phi| U |\phi\rangle .$$

The standard distinguishing algorithm. The standard form of a distinguishing algorithm A for two states $|x\rangle, |y\rangle$ follows the following procedures. We may assume that the input z of algorithm is one of x or y.

1. Given an input z, prepare $|+, z\rangle$,
2. apply the controlled unitary CU for a unitary U, then
3. measure the first register in the Hadamard basis.

We may omit the last measurement for A being unitary. We write A_U to denote the above algorithm with the unitary map U.

An arbitrary distinguishing algorithm can be transformed into the standard form with the same success probability. Without loss of generality, we assume that A is a unitary map and the output is written in the first qubit. The standard form A^S works as follows.

1. Given an input z, prepare $|+, z\rangle$,
2. apply A on the second register,
3. apply CZ on the second qubit (i.e. output of A), then
4. apply $A^\dagger$ on the second register.

Note that the last operation $A^\dagger$ is needed for ensuring the overall operation is controlled by the first qubit. Furthermore, this construction is almost the same with Figure 6.2, except that we add a single register for the controlled operations. In fact, the overall construction can be thought as a combination of Figures 6.1 and 6.2.

We may write the result of A on $|z\rangle$ as $|0, z_0\rangle + |1, z_1\rangle$, and the probability that A outputs 0 is $|z_0|^2$. The probability that A^S outputs $|+\rangle$ is $|z_0|^2$ as follows

$$CZ \cdot A\,|+, z\rangle = \frac{|0\rangle\,(|0, z_0\rangle + |1, z_1\rangle) + |1\rangle\,(|0, z_0\rangle - |1, z_1\rangle)}{\sqrt{2}} = |+\rangle\,|0, z_0\rangle + |-\rangle\,|1, z_1\rangle\,,$$

and the final state after applying $A^\dagger$ is

$$\frac{|0\rangle\,|z\rangle + |1\rangle\,A^\dagger Z A\,|z\rangle}{\sqrt{2}}.$$

Proof of the equivalence theorem. We prove the following variant of Theorem 6.1.2 for the standard distinguishing algorithm.

Lemma 6.3.2. *The standard distinguishing algorithm for $|\phi\rangle$ and $|\psi\rangle$ can be obtained by using the swapping algorithm for $|x\rangle$ and $|y\rangle$ only once (plus $O(1)$ additional gates), and vice versa.*

Proof. We first prove the statement from distinguishing to swapping. Let $A = A_U$ be a standard distinguishing algorithm for $|\phi\rangle$ and $|\psi\rangle$ with an advantage Δ. The following observation is a folklore: For the controlled map CU

$$CU\,|+,x\rangle = \frac{|0,x\rangle + |1\rangle\,U\,|x\rangle}{\sqrt{2}} = |+\rangle\left(\frac{|x\rangle + U\,|x\rangle}{2}\right) + |-\rangle\left(\frac{|x\rangle - U\,|x\rangle}{2}\right),$$

thus the probabilities that observing $+$ or $-$ by measuring the first register are

$$\frac{1}{2} + \mathsf{Re}\left(\frac{\langle x|\,U\,|x\rangle}{2}\right), \quad \text{or,} \quad \frac{1}{2} - \mathsf{Re}\left(\frac{\langle x|\,U\,|x\rangle}{2}\right),$$

respectively.

In particular, the standard algorithm A_U has probability $1/2 + \mathsf{Re}(\langle z|\,|U\,|z\rangle)$ for outputting $+$. Therefore the advantage of A is

$$\left|\Pr[A(|\phi\rangle) \to 1] - \Pr[A(|\psi\rangle) \to 1]\right| = \left|\mathsf{Re}\left(\frac{\langle\psi|\,U\,|\psi\rangle - \langle\phi|\,U\,|\phi\rangle}{2}\right)\right|$$

By Lemma 6.3.1, this advantage is equal to

$$\left|\mathsf{Re}\left(\frac{\langle x|\,U\,|y\rangle + \langle y|\,U\,|x\rangle}{2}\right)\right|.$$

Finally, by correcting the phase of U as in Figure 6.1, we have the Δ-swapping algorithm.

We also can obtain the reverse direction, from swapping to distinguishing, by reversing the above proof. The only difference is the phase correcting step, which can be omitted for this case. □

We can prove the original statement Theorem 6.1.2 using the transform from an arbitrary distinguishing algorithm to its standard form, which requires to run the original algorithm twice.

Chapter 7

Quantum Public Key Encryption

This chapter focuses on the new constructions of public key encryptions (PKE). The public key encryption allows anyone to encrypt its own message using a publicly known key and the resulting ciphertexts can be decrypted using the corresponding ciphertext.

We construct the PKE schemes through the new abstraction called the swap-trapdoor function pairs and the equivalence theorem of swapping and distinguishing. We show that the swap-trapdoor function pairs can be constructed from the cryptographic (non-abelian) group actions and lattices. Our group-action based scheme is the first PKE based on the non-abelian group action, which resolves an open problem posed in [JQSY19], albeit with the quantum ciphertexts. On the other hand, our lattice-based construction is the first additive homomorphic PKE from lattice without any error, though the homomorphic addition destructs the based ciphertexts.

In Section 7.1, we introduce the swap-trapdoor function pairs and explore its properties. Then, in Section 7.2, we present our quantum-ciphertext PKE construction based on the swap-trapdoor function pairs. We explore concrete instantiations and properties in the subsequent sections.

7.1 Swap-trapdoor Function Pairs

We introduce a new notion of swap-trapdoor function pairs (STFs), which can be seen as a variant of claw-free function pairs. Intuitively, a STF consists of two functions $f_0, f_1 : D \to R$ such that there is a trapdoor which enables us to *swap* preimages under two functions f_0 and f_1, that is, given x_b, we can find $x_{b\oplus 1}$ such that $f_{b\oplus 1}(x_{b\oplus 1}) = f_b(x_b)$ using the trapdoor. The formal definition of STFs is described below.

Definition 7.1.1. A *swap-trapdoor function pair (STF)* consists of three algorithms $(\mathsf{Setup}, \mathsf{Eval}, \mathsf{Swap})$ as follows.

$\mathsf{Setup}(1^\lambda) \to (\mathsf{pp}, \mathsf{td})$**:** This is a PPT algorithm that takes the security parameter 1^λ as an input, and outputs a public parameter pp and a trapdoor td. The public parameter pp specifies two functions $f_0^{(\mathsf{pp})}, f_1^{(\mathsf{pp})} : D_\lambda \to R_\lambda$.

$\mathsf{Eval}(1^\lambda, \mathsf{pp}, b, x) \to y$**:** This is a deterministic (classical) polynomial time algorithm that takes the security parameter 1^λ, a public parameter pp, a bit $b \in \{0, 1\}$, and an element $x \in D_\lambda$ as inputs, and outputs $y \in R_\lambda$.

$\mathsf{Swap}(1^\lambda, \mathsf{td}, b, x) \to x'$: This is a deterministic (classical) polynomial time algorithm that takes the security parameter 1^λ, a trapdoor td, and an element $x \in D_\lambda$ as inputs, and outputs $x' \in D_\lambda$.

The correctness of STFs is defined as follows.

Evaluation correctness. For any λ, and any $(\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda)$, $b \in \{0, 1\}$, and $x \in D_\lambda$, we have $\mathsf{Eval}(1^\lambda, \mathsf{pp}, b, x) = f_b^{(\mathsf{pp})}(x)$.

Swapping correctness. For any λ, and any $(\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda)$, $b \in \{0, 1\}$, and $x \in D_\lambda$, if we let $x' \leftarrow \mathsf{Swap}(1^\lambda, \mathsf{td}, b, x)$, then we have $f_{b\oplus 1}(x') = f_b(x)$ and $\mathsf{Swap}(1^\lambda, \mathsf{td}, b \oplus 1, x') \to x$. In particular, $\mathsf{Swap}(1^\lambda, \mathsf{td}, b, \cdot)$ induces an efficient invertible one-to-one mapping between $(f_0^{(\mathsf{pp})})^{-1}(y)$ and $(f_1^{(\mathsf{pp})})^{-1}(y)$ for any $y \in R_\lambda$.

We occasionally omit the dependence on λ and pp if there is no confusion. We additionally requires the efficiently samplable domain of functions later. We will occasionally use *the superposition state* $|X\rangle := \sum_{x\in X} |x\rangle / \sqrt{|X|}$ for a set X.

Definition 7.1.2. A set D is said to be *efficiently samplable* if there is a PPT algorithm that samples a almost uniform random element from D (i.e., the distribution of the sample is statistically close to the unifrom distribution). Similarly, D is *efficiently superposition samplable* if there is a QPT algorithm that produces a state whose trade distance from the superposition state

$$|D\rangle = \frac{\sum_{x\in D} |x\rangle}{\sqrt{|D|}}$$

is negligible.

Remark 10. In the rest of the thesis, we just assume that we can *exactly* sample the uniform random elements and $|D\rangle$. This assumption simplifies the overall presentation of results, and all of the results hold for the imperfect case as well up to only an additive negligible loss for the security or correctness.

We define two security notion for the security of STFs which we call *claw-freeness* and *conversion hardness*. Looking ahead, our construction only requires STFs to be conversion hard, but we present both definition due to the relations between them as we show later.

Definition 7.1.3 (Claw-freeness)**.** We say that a STF $(\mathsf{Setup}, \mathsf{Eval}, \mathsf{Swap})$ is claw-free STF if for any (non-uniform) QPT algorithm A, we have

$$\Pr\left[f_0(x_0) = f_1(x_1) : (\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda), (x_0, x_1) \leftarrow A(\mathsf{pp})\right] = \mathsf{negl}(\lambda).$$

Definition 7.1.4 (Conversion hardness)**.** We say that a STF $(\mathsf{Setup}, \mathsf{Eval}, \mathsf{Swap})$ is claw-free STF if for any (non-uniform) QPT algorithm A, we have

$$\Pr\left[f_1(x_1) = y : (\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda), x_0 \leftarrow D, y := f_0(x_0), x_1 \leftarrow A\left(\mathsf{pp}, \left|f_0^{-1}(y)\right\rangle\right)\right]$$

is negligible, where $\left|f_0^{-1}(y)\right\rangle$ is the superposition state for $\{x : f_0(x) = y\}$.

Remark 11. Conversion hardness is asymmetrically defined for f_0 and f_1. We can define the other notion be swapping the role of f_0 and f_1, but these two notions do not seem to be equivalent.

7.1.1 From claw-free to conversion hard

As the notion of conversion hardness is newly introduced here, we show some conditions and evidences that conversion hardness holds for claw-free function pairs.

The first lemma states that claw-freeness implies conversion hardness if the first function f_0 is collapsing (Definition 2.3.4). Here the collapsingness of f_0 is defined natural way by simply ignoring f_1 and considering pp as an index for f_0.

Lemma 7.1.5 (Claw-free + collapsing ⇒ Conversion hard)**.** *If f_0 is collapsing, then claw-freeness of STF implies conversion hardness.*

Proof. Suppose that $(\mathsf{Setup}, \mathsf{Eval}, \mathsf{Swap})$ does not satisfy conversion hardness. In other words, there is a non-uniform QPT adversary A such that

$$\Pr\left[f_1(x_1) = y : (\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda), x_0 \leftarrow D, y := f_0(x_0), x_1 \leftarrow A\left(\mathsf{pp}, \left|f_0^{-1}(y)\right\rangle\right)\right]$$

is non-negligible. Since f_0 is collapsing, the outputs of A with input $\left|f_0^{-1}(y)\right\rangle$ and with the measurement results of $\left|f_0^{-1}(y)\right\rangle$ in the computational basis only negligibly differs. This implies that

$$\varepsilon = \Pr\left[f_1(x_1) = y : (\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda), x_0 \leftarrow D, y := f_0(x_0), x_1 \leftarrow A\left(\mathsf{pp}, |x_0\rangle\right)\right]$$

is also non-negligible. We define a non-uniform QPT algorithm B as follows.

$B(\mathsf{pp})$: Pick $x_0 \leftarrow D$, run $x_1 \leftarrow A(\mathsf{pp}, |x_0\rangle)$, and output (x_0, x_1).

The advantage of algorithm B for claw-freeness is the same as the probability ε, which is non-negligible. □

As a special case, Lemma 7.1.5 asserts that an injective STF, i.e., f_0 is injective, satisfies conversion hardness, because an injective function is trivially collapsing.

Remark 12. More generally, if f_0 is collision-resistant, recent results [CX22, Zha22, DS22] provide many conditions that f_0 being collapsing, and some strong evidences that f_0 also satisfies collapsing.

The second lemma is a "win-win" result from claw-freeness to conversion hardness, inspired from [Zha21]. We show that a claw-free but not conversion hard STF can be used to construct one-shot signatures [AGKZ20], which is a notoriously hard to construct object. The formal definition of one-shot signatures are deferred to the end of this chapter.

Before stating the lemma, we note that the win-win results usually suffer a so-called *infinitely-often* subtleties, meaning that it only requires the security hold for infinitely many security parameters instead of all but finite parameters. See [Zha21, Section 4.1] for more explanations about infinitely-often security.

Our win-win result is given below. We sketch the proof here, and the full proof is placed in Section 7.6.

Lemma 7.1.6 (Claw-free + non-conversion hard ⇒ One-shot signature)**.** *For any STF that satisfies claw-freeness, the following statements hold:*

1. *If the STF is not uniform conversion hard, then we can use it to construct infinitely-often one-shot signatures.*

2. *If the STF is not infinitely-often uniform conversion hard, then we can use it to construct one-shot signatures.*

Proof sketch. We give a sketch of proof when the conversion hardness is totally broken, meaning that there is an efficient algorithm to find x_1 such that $f_1(x_1) = y$ given $(\mathsf{pp}, \left|f_0^{-1}(y)\right\rangle)$ with certainty.

The one-shot signature has pp as the public parameters, and $\left|f_0^{-1}(y)\right\rangle$ as the secret key and y as the corresponding verification key. For signing 0, the signer

uses the measurement result x_0 of the secret key in the computational basis as the signature so that $f_0(x_0) = y$. For signing 1, the signer runs the conversion adversary to get x_1 such that $f_1(x_1) = y$ and uses x_1 as the signature. The security of this scheme follows from the claw-freeness of STF. □

7.1.2 Counterexample of the other direction

We show that a conversion hard STF does not necessarily claw-free for completeness. Briefly speaking, our counterexample is constructed by adding an easy claw in the conversion hard STF. Precisely, our counterexample lemma is as follows.

Lemma 7.1.7. *If there is a conversion hard STF, then there is a conversion hard but not claw-free STF.*

Proof. Let $(\mathsf{Setup}, \mathsf{Eval}, \mathsf{Swap})$ be a conversion hard STF. Let ∞ be an element that is not included in D or R. Note that $|D|$ and $|R|$ should be super-polynomially large; if not, the random guess breaks conversion hardness.

We define a new STF' by appending ∞ to the original STF. Precisely, let $D' := D \cup \{\infty\}, R' := R \cup \{\infty\}$ and $f'_b := D' \to R'$ by $f'_b(x) = x$ for all $x \in D$ and $f'_b(\infty) = \infty$.

The conversion hardness of STF implies the conversion hardness of STF'. On the other hand, we can find the claw (∞, ∞) such that $f'_0(\infty) = f'_1(\infty) = \infty$ with certainty, which implies that STF' is not claw-free, which concludes the proof. □

7.2 Quantum-Ciphertext Public Key Encryption

We now turn to the construction of public key encryption (PKE) with quantum ciphertexts from STFs. We first present the formal definition of quantum-ciphertext PKE as follows.

Definition 7.2.1 (Quantum-ciphertext Public key encryption)**.** A public key encryption (PKE) scheme (with single-bit messages) consists of three algorithms $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$:

$\mathsf{KeyGen}(1^\lambda) \to (\mathsf{pk}, \mathsf{sk})$**:** This is a PPT algorithm that takes the security parameter 1^λ as input, and outputs a classical public key pk and classical secret key sk.

$\mathsf{Enc}(\mathsf{pk}, b) \to ct$**:** This is a QPT algorithm that takes a classical public key pk and a message $b \in \{0,1\}$ as input, and outputs a quantum ciphertext ct.

$\mathsf{Dec}(\mathsf{sk}, ct) \to b'/\perp$**:** This is a QPT algorithm that takes a secret key sk and a ciphertext ct as input, and outputs a message $b' \in \{0,1\}$ or $\perp$.

The PKE scheme must satisfy the correctness defined below:

Correctness. For any $m \in \{0,1\}$, the following probability

$$\Pr\left[m' = m : (\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^\lambda), m' \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, m))\right]$$

is negligibly close to 1.

The IND-CPA security of quantum-ciphertext PKE is defined analogously to the classical PKE as follows.

Definition 7.2.2 (IND-CPA security)**.** We say that a quantum-ciphertext PKE scheme $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ is IND-CPA secure if for any non-uniform QPT adversary A, we have

$$|\Pr[A(\mathsf{pk}, ct_0) \to 1] - \Pr[A(\mathsf{pk}, ct_1) \to 1]| = \mathsf{negl}(\lambda)$$

where $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$, $ct_0 \leftarrow \mathsf{Enc}(\mathsf{pk}, 0)$, and $ct_1 \leftarrow \mathsf{Enc}(\mathsf{pk}, 1)$.

Remark 13 (PKE for single bit suffices)**.** While we only consider a single bit encryption in this thesis, a simple parallel repetition works to expand the message length. Moreover, we can further extend the message space to quantum states by a hybrid encryption with quantum one-time pad as in [BJ15], i.e., we encrypt a quantum message by a quantum one-time pad, and then encrypt the key of the quantum one-time pad by quantum PKE for classical messages.

Construction from STF. Let (Setup, Eval, Swap) be a STF. Our quantum-ciphertext PKE scheme (KeyGen, Enc, Dec) is as follows.

$\mathsf{KeyGen}(1^\lambda)$**:** Run $\mathsf{Setup}(1^\lambda) \to (\mathsf{pp}, \mathsf{td})$ and output $\mathsf{pk} := \mathsf{pp}$ and $\mathsf{sk} := \mathsf{td}$.

$\mathsf{Enc}(\mathsf{pk}, b \in \{0,1\})$**:** Parse $\mathsf{pk} = \mathsf{pp}$. Prepare two registers $\mathbf{D}$ and $\mathbf{X}$ and generate the state

$$\frac{(|0\rangle + (-1)^b |1\rangle)_{\mathbf{D}} |D\rangle_{\mathbf{X}}}{\sqrt{2}} = \frac{(|0\rangle + (-1)^b |1\rangle)_{\mathbf{D}} \sum_{x\in D} |x\rangle_{\mathbf{X}}}{\sqrt{2|D|}}.$$

Prepare another register $\mathbf{Y}$ and coherently compute f_0 or f_1 into $\mathbf{Y}$ controlled by $\mathbf{D}$ to get

$$\frac{\sum_{x\in D} \left(|0\rangle_{\mathbf{D}} |x\rangle_{\mathbf{X}} |f_0(x)\rangle_{\mathbf{Y}} + (-1)^b |1\rangle_{\mathbf{D}} |x\rangle_{\mathbf{X}} |f_1(x)\rangle_{\mathbf{Y}}\right)}{\sqrt{2|D|}}$$

and measure $\mathbf{Y}$ to get $y \in R$. At this point, $\mathbf{D}$ and $\mathbf{X}$ collapse to the following state

$$\frac{1}{\sqrt{2}} \left(|0\rangle_{\mathbf{D}} \left|f_0^{-1}(y)\right\rangle_{\mathbf{Y}} + (-1)^b |1\rangle_{\mathbf{D}} \left|f_1^{-1}(y)\right\rangle_{\mathbf{Y}}\right),$$

where we have $|f_0^{-1}(y)| = |f_1^{-1}(y)|$ due to the swapping correctness. The above state is set to be ct. Note that y does not need to be included in the ciphertext.

$\mathsf{Dec}(\mathsf{sk}, ct)$**:** Parse $\mathsf{sk} = \mathsf{td}$. Let U_{td} be a unitary over $\mathbf{D}$ and $\mathbf{X}$ such that

$$U_{\mathsf{td}} |0\rangle_{\mathbf{D}} |x\rangle_{\mathbf{X}} = |0\rangle_{\mathbf{D}} |x\rangle_{\mathbf{X}},$$
$$U_{\mathsf{td}} |1\rangle_{\mathbf{D}} |x\rangle_{\mathbf{X}} = |1\rangle_{\mathbf{D}} |\mathsf{Swap}(\mathsf{td}, 1, x)\rangle_{\mathbf{X}}.$$

In particular, $U_{\mathsf{td}} |1\rangle_{\mathbf{D}} \left|f_1^{-1}(y)\right\rangle_{\mathbf{X}} = |1\rangle_{\mathbf{D}} \left|f_0^{-1}(y)\right\rangle$. Then measure $\mathbf{D}$ in the Hadamard basis and output the measurement outcome $b' \in \{0,1\}$.

We describe the above PKE scheme is correct and secure in the following lemmas.

Lemma 7.2.3 (Correctness)**.** *For any STF* (Setup, Eval, Swap)*, PKE* (KeyGen, Enc, Dec) *satisfies correctness.*

Proof. An honestly generated ciphertext ct is of the form

$$\frac{1}{\sqrt{2}}\left(|0\rangle_{\mathbf{D}}\left|f_0^{-1}(y)\right\rangle_{\mathbf{Y}}+(-1)^b|1\rangle_{\mathbf{D}}\left|f_1^{-1}(y)\right\rangle_{\mathbf{Y}}\right),$$

and the unitary U_{td} maps this state to

$$\frac{|0\rangle_{\mathbf{D}}+(-1)^b|1\rangle_{\mathbf{D}}}{\sqrt{2}}\left|f_0^{-1}(y)\right\rangle_{\mathbf{X}}.$$

The measurement of $\mathbf{D}$ in Hadamard basis gives b. □

Lemma 7.2.4 (Security)**.** *If STF* (Setup, Eval, Swap) *satisfies conversion hardness, then PKE* (KeyGen, Enc, Dec) *is IND-CPA secure.*

Proof. We first note that the computational indistinguishability of $|\psi_0\rangle$ and $|\psi_1\rangle$ defined below against any non-uniform QPT adversary that does not act on $\mathbf{Y}$ is equivalent to the IND-CPA security of the scheme:

$$|\psi_b\rangle := \mathrm{Tr}_{\mathbf{P}'}\left(\sum_{\mathsf{pp}}|\mathsf{pp}\rangle_{\mathbf{P}}|\mathsf{pp}\rangle_{\mathbf{P}'}\frac{\sum_{x\in D}\left(|0\rangle_{\mathbf{D}}|x\rangle_{\mathbf{X}}|f_0(x)\rangle_{\mathbf{Y}}+(-1)^b|1\rangle_{\mathbf{D}}|x\rangle_{\mathbf{X}}|f_1(x)\rangle_{\mathbf{Y}}\right)}{\sqrt{2|D|}}\right)$$

where we omit the normalization factor.

Suppose that there is a QPF distinguisher A with an advice $|\tau\rangle_{\mathbf{Z}}$ that does not act on $\mathbf{Y}$ and distinguishes $|\psi_0\rangle$ and $|\psi_1\rangle$ with non-negligible advantage. Then, since $|\psi_0\rangle$ and $|\psi_1\rangle$ are orthogonal, Theorem 6.1.2 states that there exists a QPT unitary U such that swaps $|\phi_0\rangle$ and $|\phi_0\rangle$ defined by

$$|\phi_b\rangle := \frac{|\psi_0\rangle|\tau\rangle+(-1)^b|\psi_1\rangle|\tau\rangle}{\sqrt{2}}$$

with the same non-negligible advantage.[1] In other words, we have a unitary U that does not act on $\mathbf{Y}$ and such that

$$|\langle\phi_0|U|\phi_1\rangle+\langle\phi_1|U|\phi_0\rangle|$$

[1] We only need the original equivalence theorem at this point, as we include the advice state as an states to be swapped.

is non-negligible.

By triangular inequality, we have one of $|\langle\phi_0| U |\phi_1\rangle|$ or $|\langle\phi_1| U |\phi_0\rangle|$ is non-negligible. We can assume that the latter inequality holds without loss of the generality, since otherwise the unitary $U^\dagger$ does satisfy the latter inequality.

We construct a non-uniform QPT adversary B that breaks the conversion hardness of STF as follows.

$B\left(\mathsf{pp}, \left|f_0^{-1}(y)\right\rangle_{\mathbf{X}}; |\tau\rangle_{\mathbf{Z}}\right)$: On input $\mathsf{pp}, \left|f_0^{-1}(y)\right\rangle_{\mathbf{X}}$ and a quantum advice $|\tau\rangle_{\mathbf{Z}}$, prepare a single qubit register $\mathbf{D}$ that is initialized to be $|0\rangle_{\mathbf{D}}$, apply U on $|\mathsf{pp}\rangle_{\mathbf{P}} |0\rangle_{\mathbf{D}} \left|f_0^{-1}(y)\right\rangle_{\mathbf{X}} |\tau\rangle_{\mathbf{Z}}$, measure $\mathbf{X}$ to obtain an outcome x', and output x'.

For any pp, we have

$$\begin{aligned}
&\Pr\left[f_1(x') = y : x \leftarrow D, y := f_0(x), x' \leftarrow B\left(\mathsf{pp}, \left|f_0^{-1}(y)\right\rangle_{\mathbf{X}}; |\tau\rangle_{\mathbf{Z}}\right)\right] \\
&= \sum_{\substack{y\in R \\ x'\in f_1^{-1}(y)}} \frac{|f_0^{-1}(y)|}{|D|} \left\| \langle x'|_{\mathbf{X}} U |\mathsf{pp}\rangle_{\mathbf{P}} |0\rangle_{\mathbf{D}} \left|f_0^{-1}(y)\right\rangle_{\mathbf{X}} |\tau\rangle_{\mathbf{Z}} \right\|^2 && (7.1) \\
&\geq \frac{1}{|D|} \left(\sum_{\substack{y\in R \\ x'\in f_1^{-1}(y)}} \sqrt{\frac{|f_0^{-1}(y)|}{|D|}} \left\| \langle x'|_{\mathbf{X}} U |\mathsf{pp}\rangle_{\mathbf{P}} |0\rangle_{\mathbf{D}} \left|f_0^{-1}(y)\right\rangle_{\mathbf{X}} |\tau\rangle_{\mathbf{Z}} \right\| \right)^2 && (7.2) \\
&\geq \frac{1}{|D|^2} \left\| \sum_{\substack{y\in R \\ x'\in f_1^{-1}(y)}} \sqrt{|f_0^{-1}(y)|} \langle x'|_{\mathbf{X}} U |\mathsf{pp}\rangle_{\mathbf{P}} |0\rangle_{\mathbf{D}} \left|f_0^{-1}(y)\right\rangle_{\mathbf{X}} |\tau\rangle_{\mathbf{Z}} \right\|^2 && (7.3) \\
&\geq \frac{1}{|D|^2} \left| \sum_{\substack{y\in R \\ x\in f_0^{-1}(y) \\ x'\in f_1^{-1}(y)}} \langle \mathsf{pp}|_{\mathbf{P}} \langle 1|_{\mathbf{D}} \langle x'|_{\mathbf{X}} \langle \tau|_{\mathbf{Z}} U |\mathsf{pp}\rangle_{\mathbf{P}} |0\rangle_{\mathbf{D}} |x\rangle_{\mathbf{X}} |\tau\rangle_{\mathbf{Z}} \right|^2 && (7.4) \\
&= \frac{1}{|D|^2} \left| \begin{array}{l} \left(\sum_{x'\in D} \langle \mathsf{pp}|_{\mathbf{P}} \langle \mathsf{pp}|_{\mathbf{P}'} \langle 1|_{\mathbf{D}} \langle x'|_{\mathbf{X}} \langle f_1(x')|_{\mathbf{Y}} \langle \tau|_{\mathbf{Z}}\right) \\ (U \otimes I_{\mathbf{P}',\mathbf{Y}}) \left(\sum_{x\in D} |\mathsf{pp}\rangle_{\mathbf{P}} |\mathsf{pp}\rangle_{\mathbf{P}'} |0\rangle_{\mathbf{D}} |x\rangle_{\mathbf{X}} |f_0(x)\rangle_{\mathbf{Y}} |\tau\rangle_{\mathbf{Z}}\right) \end{array} \right|^2 . && (7.5)
\end{aligned}$$

where Equation (7.1) follows from the definition of B, Equation (7.2) follows from Cauchy–Schwarz inequality and $\sum_{y\in R}|f_1^{-1}(y)| = |D|$, Equation (7.3) follows from the triangle inequality, and Equation (7.4) follows from the definition $\left|f_0^{-1}(y)\right\rangle = \frac{1}{|f_0^{-1}(y)|^{1/2}}\sum_{x\in f_0^{-1}(y)}|x\rangle$ and the fact that inserting $\langle \mathsf{pp}|_{\mathbf{P}}\langle 1|_{\mathbf{D}}\langle\tau|_{\mathbf{Z}}$ can only decrease the norm.

Therefore, for the probability of public parameters $\Pr(\mathsf{pp})$, we have

$$
\begin{aligned}
&\Pr\left[f_1\left(B\left(\mathsf{pp},\left|f_0^{-1}(y)\right\rangle_{\mathbf{X}};|\tau\rangle_{\mathbf{Z}}\right)\right)=y:(\mathsf{pp},\mathsf{td})\leftarrow\mathsf{Setup}(1^\lambda),x\leftarrow D,y:=f_0(x)\right]\\
&=\sum_{\mathsf{pp}}\Pr(\mathsf{pp})\left[\Pr\left[f_1(x')=y:x\leftarrow D,y:=f_0(x),x'\leftarrow B\left(\mathsf{pp},\left|f_0^{-1}(y)\right\rangle_{\mathbf{X}};|\tau\rangle_{\mathbf{Z}}\right)\right]\right]\\
&\geq\sum_{\mathsf{pp}}\frac{\Pr(\mathsf{pp})}{|D|^2}\left|\begin{array}{l}\left(\sum_{x'\in D}\langle\mathsf{pp}|_{\mathbf{P}}\langle\mathsf{pp}|_{\mathbf{P}'}\langle 1|_{\mathbf{D}}\langle x'|_{\mathbf{X}}\langle f_1(x')|_{\mathbf{Y}}\langle\tau|_{\mathbf{Z}}\right)\\(U\otimes I_{\mathbf{P}',\mathbf{Y}})\left(\sum_{x\in D}|\mathsf{pp}\rangle_{\mathbf{P}}|\mathsf{pp}\rangle_{\mathbf{P}'}|0\rangle_{\mathbf{D}}|x\rangle_{\mathbf{X}}|f_0(x)\rangle_{\mathbf{Y}}|\tau\rangle_{\mathbf{Z}}\right)\end{array}\right|^2\\
&\geq\left|\sum_{\mathsf{pp}}\frac{\Pr(\mathsf{pp})}{|D|}\left(\begin{array}{l}\left(\sum_{x'\in D}\langle\mathsf{pp}|_{\mathbf{P}}\langle\mathsf{pp}|_{\mathbf{P}'}\langle 1|_{\mathbf{D}}\langle x'|_{\mathbf{X}}\langle f_1(x')|_{\mathbf{Y}}\langle\tau|_{\mathbf{Z}}\right)\\(U\otimes I_{\mathbf{P}',\mathbf{Y}})\left(\sum_{x\in D}|\mathsf{pp}\rangle_{\mathbf{P}}|\mathsf{pp}\rangle_{\mathbf{P}'}|0\rangle_{\mathbf{D}}|x\rangle_{\mathbf{X}}|f_0(x)\rangle_{\mathbf{Y}}|\tau\rangle_{\mathbf{Z}}\right)\end{array}\right)\right|^2\\
&=\left|\langle\phi_1|_{\mathbf{P},\mathbf{P}',\mathbf{D},\mathbf{X},\mathbf{Y}}\langle\tau|_{\mathbf{Z}}(U\otimes I_{\mathbf{P}',\mathbf{Y}})|\phi_0\rangle_{\mathbf{P},\mathbf{P}',\mathbf{D},\mathbf{X},\mathbf{Y}}|\tau\rangle_{\mathbf{Z}}\right|^2,
\end{aligned}
$$

where the first inequality follows from Equation (7.5), the second inequality follows from Jensen's inequality. and the final equality follows from the definition of $|\phi_b\rangle$.

This is non-negligible by our assumption. In other words, B breaks the conversion hardness of the STF $(\mathsf{Setup},\mathsf{Eval},\mathsf{Swap})$, which is a contradiction. Thus, $(\mathsf{KeyGen},\mathsf{Enc},\mathsf{Dec})$ is IND-CPA secure. □

Finally, we have the main theorem combining Lemma 7.2.3 and Lemma 7.2.4.

Theorem 7.2.5. *There is a quantum-ciphertext PKE, assuming the existence of conversion hard swap trapdoor function pairs.*

7.3 Group Action based Construction

Our main theorem in this section is stated as follows.

Theorem 7.3.1. *There is a quantum-ciphertext PKE that is IND-CPA secure, assuming the existence of pseudorandom group actions without a dominant orbit.*

In this section, we review the basic definitions for cryptographic group actions following [JQSY19] and the cryptographic assumptions used in the above theorem. Then we construct a STF based on the cryptographic assumptions of group actions with the proof of this theorem in Section 7.3.4.

7.3.1 Definitions

We first recall the group action. Note that the group is not necessarily abelian.

Definition 7.3.2 (Group actions)**.** Let G be a group, S be a set, and $\star : G \times S \to S$ be a function where we write $g \star s$ to mean $\star(g, s)$. We say that $(G, S, \star)$ is a group action if it satisfies the following:

1. For the identity element $e \in G$ and any $s \in S$, we have $e \star s = s$.
2. For any $g, h \in G$ and any $s \in S$, we have $(gh) \star s = g \star (h \star s)$.

For the cryptographic use, we only consider the group where the following operations are efficient, which is formalized as follows. These requirements are identical to those in [JQSY19] except for the *superposition over G* property. We remark that all candidate constructions proposed in [JQSY19] satisfy this property as explained later, thus we safely assume this operation being efficient as well.

Definition 7.3.3 (Group actions with efficient algorithms)**.** We say that a group action $(G, S, \star)$ has efficient algorithms if it satisfies the following:[2]

[2]Strictly speaking, we have to consider a family $\{(G_\lambda, S_\lambda, \star_\lambda)\}_{\lambda \in \mathbb{N}}$ of group actions parameterized by the security parameter to meaningfully define the efficiency requirements. We omit the dependence on λ for notational simplicity throughout the thesis.

Unique representations: Each element of G and S can be represented as a bit string of length $\mathsf{poly}(\lambda)$ in a unique manner. Thus, we identify these elements and their representations.

Efficient recognizability: There are classical polynomial-time algorithms that decide if a given bit string represents an element of G or S, respectively.

Group operations: There are deterministic polynomial-time algorithms that compute gh from $g \in G$ and $h \in G$ and g^{-1} from $g \in G$.

Group action: There is a classical polynomial-time algorithm that computes $g \star s$ from $g \in G$ and $s \in S$.

Random sampling: There are PPT algorithms that sample almost uniform elements of G or S (i.e., the distribution of the sample is statistically close to the uniform distribution), respectively.

Superposition over G: There is a QPT algorithm that generates a state whose trace distance from $|G\rangle$ is $\mathsf{negl}(\lambda)$.

Remark 14 (A convention on Random sampling and Superposition over G properties)**.** In the rest of this thesis, we assume that we can sample elements from *exactly* uniform distributions of G and S. Similarly, we assume that we can *exactly* generate $|G\rangle$ in QPT. They are just for simplifying the presentations of our results, and all the results hold with the above imperfect version with additive negligible loss for security or correctness.

Remark 15 (Group operations + generators $\Rightarrow$ Random sampling)**.** We note that Babai [Bab91] studied the random sampling procedure in the black-box group model. In particular, if we have the generator of group G and have the efficient group operations, then we can sample an almost uniform distribution in time $\mathsf{poly}(\lambda)$. In our instantiation below, we know the generator of group and have the efficient group operations.

Remark 16 (Superposition over G)**.** Contrary to the efficient random sampling, the efficient superposition over G may not be obtained from the other assumptions due to the entanglements. In particular, the hardness of sampling superposition over G is implicitly used in the oracle separation of QMA and MA in [Wat00].

7.3.2 Cryptographic Assumptions

Now we define cryptographic assumptions for group actions following [JQSY19]: one-wayness and pseudorandomness.

Definition 7.3.4 (One-wayness)**.** We say that a group action $(G, S, \star)$ is one-way if, given $g \star s$ for random $g \in G$ and $s \in S$, no efficient algorithm can find g' such that $g' \star s = g \star s$. That is, for an arbitrary non-uniform QPT adversary A, we have

$$\Pr\left[g' \star s = g \star s : s \leftarrow S, g \leftarrow G, g' \leftarrow A(s, g \star s)\right] = \mathsf{negl}(\lambda).$$

Definition 7.3.5 (Pseudorandomness)**.** We say that a group action $(G, S, \star)$ is pseudorandom if, for random $s, t \in S$ and $g \in G$, no efficient algorithm can distinguish between (s, t) and $(s, t' = g \star s)$. That is, for an arbitrary non-uniform QPT adversary A, we have

$$\left|\Pr\left[1 \leftarrow A(s,t) : s \leftarrow S, g \leftarrow G, t := g \star s\right] - \Pr\left[1 \leftarrow A(s,t) : s, t \leftarrow S\right]\right| = \mathsf{negl}(\lambda).$$

In particular, we say that a group action $(G, S, \star)$ does not have *a dominant orbit* if

$$\Pr[\exists g \in G \text{ s.t. } g \star s = t : s, t \leftarrow S] = \mathsf{negl}(\lambda).$$

We note that the group action *with* a dominant orbit is almost transitive, and in that case the pseudorandomness of group action obviously holds. For more discussions, we refer [JQSY19, Section 4]. We also note that the pseudorandomness immediately implies the one-wayness as noted in the original paper.

7.3.3 Instantiation

For completeness, we briefly describe one of the candidate cryptographic group actions called *the general linear group action on tensors (GLAT)* presented in [JQSY19].

Let $\mathbb{F}$ be a finite field, $k, d_1, ..., d_k$ be positive integers. We may assume $k = 3$ and $d_1 = d_2 = d_3$. We set $G := \prod_{j=1}^{k} GL_{d_j}(\mathbb{F})$ and $S := \bigotimes_{j=1}^{k} \mathbb{F}^{d_j}$ where $GL_n(\mathbb{F})$ denotes the set of general linear maps (i.e. matrices with nonzero determinants).

For $(M_j)_{j\in[k]} \in \prod_{j=1}^{k} GL_{d_j}(\mathbb{F})$ and $T \in \bigotimes_{j=1}^{k} \mathbb{F}^{d_j}$, the group action $\star$ is defined by the matrix-vector multiplication

$$(M_j)_{j\in[k]} \star T := \left(\bigotimes_{j=1}^{k} M_j \right) T.$$

The original paper [JQSY19] presents several attempts of cryptanalysis and justifications of the one-wayness and pseudorandomness of this action.

We remark that the additional requirement of the efficient *superposition over* G holds for this candidate, as well as others suggested in the original paper. We briefly describe the procedure for GLAT, which suffices to construct a state that is negligibly close to the uniform superposition over the *invertible* matrices. The quantum Fourier transform exactly does this, since for large $|\mathbb{F}|$ and d_1, d_2, d_3, the probability that a uniform random matrix is invertible is overwhelming.

7.3.4 STF and PKE from Group Actions

We present the construction of STF based on group actions. Let $\alpha = (G, S, \star)$ be a group action with efficient algorithms (as defined in Definition 7.3.3). In other words, we prove the following theorem.

Theorem 7.3.6. *There is a conversion hard STF assuming the pseudorandom group action without a dominant orbit.*

To prove this theorem, we construct a STF from A, denoted by $STF(\alpha)$.

$\mathsf{Setup}(1^\lambda)$**:** Generate $s_0 \leftarrow S$ and $g \leftarrow G$, set $s_1 := g \star s_0$, and output $\mathsf{pp} := (s_0, s_1)$ and $\mathsf{td} := g$. For $b \in \{0,1\}$, we define $f_b : G \to S$ by $f_b(h) := h \star s_b$.

$\mathsf{Eval}(\mathsf{pp} = (s_0, s_1), b, h)$**:** Output $f_b(h) = h \star s_b$.

$\mathsf{Swap}(\mathsf{td} = g, b, h)$**:** If $b = 0$, output hg^{-1}. If $b = 1$, output hg.

Correctness and efficient operations. The evaluation correctness is obvious. The swapping correctness can be seen as follows: For any $h \in G$,

$$f_1(\mathsf{Swap}(\mathsf{td}, 0, h)) = f_1(hg^{-1}) = (hg^{-1}) \star s_1 = (hg^{-1}) \star (g \star s_0) = h \star s_0 = f_0(h).$$

Similarly, $f_0(\mathsf{Swap}(\mathsf{td}, 1, h)) = f_1(h)$ holds for any $h \in G$. For any $h \in G$, $\mathsf{Swap}(\mathsf{td}, 1, \mathsf{Swap}(\mathsf{td}, 0, h)) = \mathsf{Swap}(\mathsf{td}, 1, hg^{-1}) = (hg^{-1})g = h$ also holds. The efficient sampling and efficient superposition properties directly follow from the corresponding properties of the group action.

Security. We prove the following lemma.

Lemma 7.3.7. *For an arbitrary group action $\alpha = (G, S, \star)$, the following statements hold:*

1. *if α is one-way, then $STF(\alpha)$ is claw-free.*
2. *If α is pseudorandom without a dominant orbit, then $STF(\alpha)$ is conversion hard.*

Proof. Let us consider Item 1. Suppose that $STF(\alpha) = (\mathsf{Setup}, \mathsf{Eval}, \mathsf{Swap})$ is not claw-free. Then there is a non-uniform QPT adversary A such that

$$\Pr\left[f_0(h_0) = f_1(h_1) : (\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda), (h_0, h_1) \leftarrow A(\mathsf{pp})\right] \tag{7.6}$$

is non-negligible. We use A to construct a non-uniform QPT adversary B that breaks one-wayness of $(G, S, \star)$ as follows:

$B(s_0, s_1)$**:** Set $\mathsf{pp} := (s_0, s_1)$, run $(h_0, h_1) \leftarrow A(\mathsf{pp})$, and outputs $h_1^{-1}h_0$.

Due to Equation (7.6), we have $f_0(h_0) = f_1(h_1)$ with a non-negligible probability. This equality implies

$$h_0 \star s_0 = f_0(h_0) = f_1(h_1) = h_1 \star s_1,$$

which means $h_1^{-1}h_0 \star s_0 = s_1$. Since this event occurs with a non-negligible probability, B breaks one-wayness of $(G, S, \star)$, which is a contradiction. Thus, $STF(\alpha) = (\mathsf{Setup}, \mathsf{Eval}, \mathsf{Swap})$ is claw-free.

Next, we prove Item 2. Suppose that $STF(\alpha) = (\mathsf{Setup}, \mathsf{Eval}, \mathsf{Swap})$ is not conversion hard. Then there is a non-uniform QPT algorithm A such that

$$\Pr\left[f_1(x_1) = y : (\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda), x_0 \leftarrow D, y := f_0(x_0), x_1 \leftarrow A(\mathsf{pp}, \left|f_0^{-1}(y)\right\rangle)\right]$$

is non-negligible. As in the above proof, this is equivalent to that the probability

$$\Pr\left[h_0 \star s_0 = h_1 \star s_1 : \begin{array}{l} s_0 \leftarrow S, g, h_0 \leftarrow G, \\ s_1 := g \star s_0, y := h_0 \star s_0, \\ h_1 \leftarrow A(s_0, s_1, \left|f_0^{-1}(y)\right\rangle) \end{array}\right]$$

is non-negligible. On the other hand, since there is no dominant orbit, we have

$$\Pr\left[h_0 \star s_0 = h_1 \star s_1 : \begin{array}{l} s_0, s_1 \leftarrow S, h_0 \leftarrow G, \\ y := h_0 \star s_0, \\ h_1 \leftarrow A(s_0, s_1, \left|f_0^{-1}(y)\right\rangle) \end{array}\right] = \mathsf{negl}(\lambda).$$

Therefore, the difference of two probabilities

$$\left| \Pr\left[h_0 \star s_0 = h_1 \star s_1 : \begin{array}{l} s_0 \leftarrow S, g, h_0 \leftarrow G, \\ s_1 := g \star s_0, y := h_0 \star s_0, \\ h_1 \leftarrow A(s_0, s_1, \left|f_0^{-1}(y)\right\rangle) \end{array}\right] - \Pr\left[h_0 \star s_0 = h_1 \star s_1 : \begin{array}{l} s_0, s_1 \leftarrow S, h_0 \leftarrow G, \\ y := h_0 \star s_0, \\ h_1 \leftarrow A(s_0, s_1, \left|f_0^{-1}(y)\right\rangle) \end{array}\right] \right| \tag{7.7}$$

is non-negligible.

We construct the following non-uniform QPT adversary B that breaks pseudorandomness of $(G, S, \star)$ based on the above observation:

$B(s_0, s_1)$**:** Generate a state $\frac{1}{\sqrt{|G|}} \sum_{h_0 \in G} |h_0\rangle |h_0 \star s_0\rangle$ and measure the second register to get $y \in S$. Then, the first register collapses to $\left|f_0^{-1}(y)\right\rangle$. Run $h_1 \leftarrow A(s_0, s_1, \left|f_0^{-1}(y)\right\rangle)$. Output 1 if $h_1 \star s_1 = y$ and otherwise 0.

The advantage of B to distinguish $(s_0, s_1 \leftarrow S)$ or $(s_0 \leftarrow S, g \star s_0$ for $g \leftarrow G)$ is exactly Equation (7.7), which is non-negligible, contradicting to pseudorandomness of $(G, S, \star)$. Thus, $STF(\alpha) = (\mathsf{Setup}, \mathsf{Eval}, \mathsf{Swap})$ is conversion hard. □

Quantum-ciphertext PKE from group actions. Recall that in Theorem 7.2.5 we showed the conversion hard STFs suffice for constructing IND-CPA secure quantum-ciphertext PKE. Combining Theorem 7.3.6 and this fact, Theorem 7.3.1 is obviously obtained.

Remark 17 (Lossy encryption)**.** We can show that the quantum-ciphertext PKE constructed from a pseudorandom group action is lossy encryption [BHY09], which is stronger than IND-CPA secure one. We omit the detail since our focus is on constructing IND-CPA secure scheme.

Furthermore, we have the following corollaries.

Corollary 7.3.8. *If there exists a one-way group action with efficient algorithms such that f_0 is collapsing*[3]*, there exists a uniform IND-CPA secure quantum-ciphertext PKE scheme.*

Corollary 7.3.9. *If there exists a one-way group action with efficient algorithms, then there exists a uniform IND-CPA secure quantum-ciphertext PKE scheme or infinitely-often one-shot signatures.*[4]

[3]We currently have no candidate of such a one-way group action.

[4]The uniform IND-CPA security is defined similarly to the IND-CPA security in Definition 7.2.2 except that the adversary is restricted to be *uniform* QPT.

7.4 Lattice based Construction

Now we provide an alternative construction of STF from lattices. Since it is already known how to construct classical PKE schemes based on LWE [Reg09, GPV08], this construction does not give a new feasibility result unlike the group action-based one as PKE. However, we will see that this construction has an interesting property of additive homomorphic.

Intriguingly, the (bit-wise) additive homomorphic computation destructs the base ciphertexts, and has no additional error, meaning that we can do an arbitrary number of homomorphic computation, unlike the previous construction whose noises are accumulated while doing homomorphic addition. We call this property by *destructive* and *noiseless* additive homomorphic.

The main theorem is stated as follows.

Theorem 7.4.1. *There is a quantum-ciphertext PKE that is IND-CPA secure and with a destructive and noiseless homomorphic addition, assuming the hardness of learning with error problems.*

We first see the lattice based STF and PKE in Section 7.4.1 and Section 7.4.2, then provide an homomorphic addition in Section 7.4.3 with some more properties.

7.4.1 Lattice based STF

The construction of STF presented here is based on the noisy trapdoor claw-free function family (NTCF) based on LWE, which is constructed in [BCM+21]. For ease of presentation, we assume non-noisy ideal trapdoor claw-free permutations defined below. We describe slight more details in Remark 18. We omit several system parameters pp or 1^λ in the definition for simplicity.

Definition 7.4.2 (Trapdoor claw-free permutation pair (TCP))**.** A trapdoor claw-free permutation pair consists of three algorithms $(\mathsf{Setup}, \mathsf{Eval}, \mathsf{TdInv})$ as follows.

$\mathsf{Setup}(1^\lambda)$**:** A setup algorithm Setup, given the security parameter 1^λ, outputs a public parameter pp and corresponding trapdoor td. Given pp, two permutations f_0 and f_1 over a set D are determined.

$\mathsf{Eval}(b, x) \to f_b(x)$**:** This is a deterministic classical polynomial time algorithm that outputs $f_b(x)$.

$\mathsf{TdInv}(\mathsf{td}, b, y) \to x'$: This is a deterministic classical polynomial time algorithm that outputs x' such that $f_b(x') = y$.

As claw-freeness, we require that it is hard to find x_0 and x_1 such that $f_0(x_0) = f_1(x_1)$ even for (non-uniform) QPT adversaries.

In what follows, we construct STF from TCP. A similar construction works with NTCFs by similar techniques as in [BCM+21]. Also note that this is in fact rephrase of Lemma 7.1.5.

STF from trapdoor claw-free permutation pairs. We assume that a TCP pair $(\mathsf{Setup}_{TCP}, \mathsf{Eval}_{TCP}, \mathsf{TdInv}_{TCP})$ is given. We construct a STF from this pair denoted by $STF(TCP)$ as follows.

$\mathsf{Setup}(1^\lambda)$**:** Run $\mathsf{Setup}_{TCP}(1^\lambda)$ to generate $(\mathsf{pp}, \mathsf{td})$ and corresponding trapdoor claw-free permutation pair (f_0, f_1), and output $(\mathsf{pp}, \mathsf{td})$.

$\mathsf{Eval}(b, x)$**:** Run $\mathsf{Eval}_{TCP}(b, x)$ to output $f_b(x)$.

$\mathsf{Swap}(\mathsf{td}, b, x)$**:** Compute $\mathsf{Eval}(b, x) = f_b(x) =: y$, run $\mathsf{TdInv}_{TCP}(\mathsf{td}, b \oplus 1, y)$ to get x' and output x'.

Correctness and efficiency. The correctness of evaluation is obvious due to the efficiency of Eval_{TCP}. The swapping correctness is also straightforward since the output x' must satisfy $f_{b\oplus 1}(x') = y = f_b(x)$ due to the correctness of TdInv_{TCP}.

Security. Conversion hardness of $STF(TCP)$ is already proven in Lemma 7.1.5, as the permutation is always collapsing.

Additional property. The lattice based NTCF has an additional property that for secret key s, $x_1 = x_0 - s \bmod q$ for any claw (x_0, x_1). This will be used in the homomorphic computation.

Remark 18. As noted above, the only known trapdoor claw-free functions are *noisy*, meaning that the function evaluation outputs a distribution rather than a single point. Still, our arguments are readily applicable to the known construction from lattice [BCM+21]. This is especially because the swapping is done by adding/subtracting secret key s.

Remark 19 (On the trapdoor.)**.** It is worth mentioning that we actually do not need the full power of td. For the decryption, we do not need to recover (x_0, x_1) from y. We only need a trapdoor that enables us to compute x_0 from x_1 and x_1 from x_0. In the LWE-based construction (with noises) [BCM+21], this is very easy because we always have $x_1 = x_0 - s \mod q$ for some secret vector s which corresponds to the LWE secret. Thus, the secret key of the above quantum-ciphertext PKE scheme can be set to be the LWE secret rather than so called lattice trapdoors[GPV08].

Remark 20 (On Ring-LWE)**.** We do not use the adaptive hardcore property introduced in [BCM+21] and only showed for the LWE-based constructions. Thus, we can also use the Ring-LWE based construction of NTCFs given in [BKVV20], which gives more efficient construction than the LWE-based one (though our focus in this thesis is not on the actual efficiency).

7.4.2 Quantum-ciphertext PKE from lattice

Based on the lattice based STF, we construct quantum-ciphertext PKE scheme following Section 7.2. We give some details for later use. We assume that the underlying LWE problem has a secret key s.

$\mathsf{KeyGen}(1^\lambda)$**:** Sample pp, td to specify (f_0, f_1). Output $\mathsf{pk} := \mathsf{pp}$ and $\mathsf{sk} := \mathsf{td} = s$ for secret key s.

$\mathsf{Enc}(\mathsf{pk}, b \in \{0,1\})$**:** Output a ciphertext

$$ct := \frac{1}{\sqrt{2}}(|0\rangle |x_0\rangle + (-1)^b |1\rangle |x_1\rangle)$$

where $f(x_0) = f(x_1) = y$ for uniformly random y. The pair (x_0, x_1) has an additional property $x_1 = x_0 - s$. This state is generated by constructing

$$\frac{1}{\sqrt{2}}(|0\rangle + (-1)^b |1\rangle)\frac{\sum_{x\in X} |x\rangle}{\sqrt{|D|}}$$

and coherently compute f_0 or f_1 controlled by the first qubit in an additional register to get

$$\sum_{x\in X} \frac{|0\rangle |x\rangle |f_0(x)\rangle + (-1)^b |1\rangle |x\rangle |f_1(x)\rangle}{\sqrt{2|D|}},$$

and measure the rightmost register to get y. At this point, the first two registers collapse to the desired state

$$\frac{1}{\sqrt{2}}(|0\rangle |x_0\rangle + (-1)^b |1\rangle |x_1\rangle).$$

$\mathsf{Dec}(\mathsf{sk}, ct)$**:** Let $\mathbf{D}$ and $\mathbf{X}$ be the first and second registers of ct, respectively. Prepare $|+\rangle_{\mathbf{B}} = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)_{\mathbf{B}}$ in an additional one-qubit register $\mathbf{B}$. Let U_{td} be a unitary over $\mathbf{D}$ and $\mathbf{X}$ such that

$$U_{\mathsf{td}} |0\rangle_{\mathbf{D}} |x\rangle_{\mathbf{X}} = |1\rangle_{\mathbf{D}} |x - s\rangle_{\mathbf{X}},$$
$$U_{\mathsf{td}} |1\rangle_{\mathbf{D}} |x\rangle_{\mathbf{X}} = |0\rangle_{\mathbf{D}} |x + s\rangle_{\mathbf{X}}$$

for any x. Apply the controlled-U_{td} controlled by $\mathbf{B}$. Finally, measure $\mathbf{B}$ in the Hadamard basis and output the measurement outcome b'.

Correctness and security both are from Theorem 7.2.5.

7.4.3 Homomorphic computation and compression

Now we show that the quantum-ciphertext PKE in Section 7.4.2 has some interesting properties such as the additive homomorphic computation, and quantum part compression. Most of them are inspired by the algorithm for dihedral hidden subgroup problem introduced by Kuperberg [Kup05].

We first prepare two encryptions of b_1, b_2

$$ct_1 = \frac{|0\rangle_{\mathbf{D}_1}|x_0\rangle_{\mathbf{X}_1} + (-1)^{b_1}|1\rangle_{\mathbf{D}_1}|x_1\rangle_{\mathbf{X}_1}}{\sqrt{2}},\ ct_2 = \frac{|0\rangle_{\mathbf{D}_2}|z_0\rangle_{\mathbf{X}_2} + (-1)^{b_2}|1\rangle_{\mathbf{D}_2}|z_1\rangle_{\mathbf{X}_2}}{\sqrt{2}}$$

where $x_1 = x_0 - s, z_1 = z_0 - s$. and observe that their concatenation $|ct_1, ct_2\rangle$ is

$$\begin{aligned}&\frac{\left(|0\rangle_{\mathbf{D}_1}|x_0\rangle_{\mathbf{X}_1} + (-1)^{b_1}|1\rangle_{\mathbf{D}_1}|x_1\rangle_{\mathbf{X}_1}\right)\otimes\left(|0\rangle_{\mathbf{D}_2}|z_0\rangle_{\mathbf{X}_2} + (-1)^{b_2}|1\rangle_{\mathbf{D}_2}|z_1\rangle_{\mathbf{X}_2}\right)}{2}\\ &= \frac{|00\rangle|x_0z_0\rangle + (-1)^{b_1+b_2}|11\rangle|x_1z_1\rangle}{2} + \frac{(-1)^{b_1}|10\rangle|x_1z_0\rangle + (-1)^{b_2}|01\rangle|x_0z_1\rangle}{2}\end{aligned}$$

where we rearrange the registers. Measuring the xor of the register $\mathbf{D} = (\mathbf{D}_1, \mathbf{D}_2)$ then results in 0 and 1 with probability $1/2$ and the overall state collapses to

$$\frac{|00\rangle|x_0z_0\rangle + (-1)^{b_1+b_2}|11\rangle|x_1z_1\rangle}{\sqrt{2}} = \frac{|00\rangle|x_0, z_0\rangle + (-1)^{b_1+b_2}|11\rangle|x_0 - s, z_0 - s\rangle}{\sqrt{2}}$$

for measurement outcome 0, and

$$\frac{|10\rangle|x_1z_0\rangle + (-1)^{b_1+b_2}|01\rangle|x_0z_1\rangle}{\sqrt{2}} = \frac{|10\rangle|x_0 - s, z_0\rangle + (-1)^{b_1+b_2}|01\rangle|x_0, z_0 - s\rangle}{\sqrt{2}}$$

for 1, where we multiply the global phase $(-1)^{b_1}$.

For the first case, applying the map $|a, b\rangle_{\mathbf{D}}|c, d\rangle_{\mathbf{X}} \mapsto |a \oplus b, b\rangle_{\mathbf{D}}|c - d, d\rangle_{\mathbf{X}}$ results in

$$\begin{aligned}&\frac{|00\rangle|x_0 - z_0, z_0\rangle + (-1)^{b_1+b_2}|01\rangle|x_0 - z_0, z_0 - s\rangle}{\sqrt{2}}\\ &= |0, x_0 - z_0\rangle_{\mathbf{D}_1,\mathbf{X}_1} \otimes \frac{|0\rangle_{\mathbf{D}_2}|z_0\rangle_{\mathbf{X}_2} + (-1)^{b_1+b_2}|1\rangle_{\mathbf{D}_2}|z_0 - s\rangle_{\mathbf{X}_2}}{\sqrt{2}}\end{aligned}$$

where the second term is an encryption of $b_1 + b_2$. On the other hand, for the second case, applying the map $|a, b\rangle_{\mathbf{D}} |c, d\rangle_{\mathbf{X}} \mapsto |a \oplus b, b\rangle_{\mathbf{D}} |c + d, d\rangle_{\mathbf{X}}$ gives

$$\frac{|10\rangle |x_0 + z_0 - s, z_0\rangle + (-1)^{b_1+b_2} |11\rangle |x_0 + z_0 - s, z_0 - s\rangle}{\sqrt{2}}$$
$$= |1, x_0 + z_0 - s, z_0\rangle_{\mathbf{D}_1,\mathbf{X}_1} \otimes \frac{|0\rangle_{\mathbf{D}_2} |z_0\rangle_{\mathbf{X}_2} + (-1)^{b_1+b_2} |1\rangle_{\mathbf{D}_2} |z_0 - s\rangle_{\mathbf{X}_2}}{\sqrt{2}}$$

which also has an encryption of $b_1 + b_2$ in the second term. In any case, discarding the registers $\mathbf{D}_1, \mathbf{X}_1$ and the measurement outcome obtained at the first step gives the encryption of $b_1 + b_2$

$$\frac{|0\rangle_{\mathbf{D}_2} |z_0\rangle_{\mathbf{X}_2} + (-1)^{b_1+b_2} |1\rangle_{\mathbf{D}_2} |z_0 - s\rangle_{\mathbf{X}_2}}{\sqrt{2}}.$$

Note that this step destructs the encryption in the register $(\mathbf{D}_1, \mathbf{X}_1)$ and does not have any additional errors. This implies that we can add an arbitrary number of ciphertexts by repeating this procedure.

Overall, we show that the construction in Section 7.4.2 is destructive and noiseless additive homomorphic quantum-ciphertext PKE, proving Theorem 7.4.1.

Compressing quantum parts. We prove an additional property of our PKE from lattice. Recall that a ciphertext is of the form

$$ct = \frac{|0\rangle |x_0\rangle + (-1)^b |1\rangle |x_1\rangle}{\sqrt{2}}.$$

Let $x_b = (z_b, w_b)$ for a single bit w_b for $b \in \{0, 1\}$. The ciphertext ct can be rephrased as

$$\frac{|0\rangle |z_0\rangle (|+\rangle + (-1)^{w_0} |-\rangle) + (-1)^b |1\rangle |z_1\rangle (|+\rangle + (-1)^{w_1} |-\rangle)}{\sqrt{2}}$$

thus if a Hadamard measurement on the last qubit results in the outcome d, then the remaining state is

$$\frac{|0\rangle |z_0\rangle + (-1)^{b \oplus c} |1\rangle |z_1\rangle}{\sqrt{2}}$$

for $c = d \cdot (w_0 \oplus w_1)$. Extending this to a Hadamard measurement on all but the first register yielding a string d, the remaining state becomes

$$\frac{|0\rangle + (-1)^{b \oplus c} |1\rangle}{\sqrt{2}}$$

for $c = d \cdot (x_0 \oplus x_1)$.

We can use this state along with y as the quantum-part compressed ciphertext, namely

$$ct_{comp} := \left(d, y, \frac{|0\rangle + (-1)^{b \oplus c} |1\rangle}{\sqrt{2}}\right).$$

The decryption can be done by recovering x_0, x_1 from y with td, uncompute c in the exponent, then measure the state in a Hadamard basis. We note that this compression is applicable to any quantum-ciphertext PKE construction based on Lemma 7.1.5 plus Lemma 7.2.4.

7.5 Discussion and Open Problems

We present new constructions of quantum-ciphertext public key encryptions using the equivalence theorem from [AAS20]. We employ the equivalence theorem as a search-to-decision reduction to obtain the security of PKE schemes. This construction shows that the equivalence theorem is useful in cryptography. In particular, we construct the first following PKE schemes.

- PKE based on the cryptographic group action, resolving an open problem posed in [JQSY19].
- PKE based on lattice assumption equipped with a noiseless additive homomorphic operation.

We conclude this chapter by presenting some open problems of the questions on (quantum-ciphertext) PKE schemes and swap-trapdoor function pairs inspired by our constructions below.

1. A classical-ciphertext PKE from group action is still open.

2. Is there any other construction of swap-trapdoor function pairs? Are hardness of conversion or swap, or other similar assumptions useful for classical cryptography? For quantum cryptographic applications, we expect more fundamental primitive related to the swap or conversion problem. For example, we can think the unclonability as a special kind of swap hard problem.

3. Can we extend our additive homomorphic scheme for more larger circuits? Or, the destruction of one ciphertext is necessary? In particular, we hope that the quantum homomorphic encryption for classical circuits could be obtained, and it may have applications beyond cryptography, for example, what is the relation between the quantum homomorphic encryption for classical circuits and black hole?

4. Can we improve the efficiency of the schemes? For example, how can we construct a PKE that encrypt multiple bits using the equivalence theorem without concatenation? We guess that a generalization of equivalence theorem is required for this question.

7.6 Deferred Proof

We give a proof of Lemma 7.1.6. Before giving the proof, we clarify definitions of terms that appear in the statement of the lemma. First, we define (infinitely-often) uniform conversion hardness for group actions.

Definition 7.6.1 ((Infinitely-often) uniform conversion hardness)**.** We say that an STF $(\mathsf{Setup}, \mathsf{Eval}, \mathsf{Swap})$ is uniform conversion hard if for any uniform QPT adversary A, we have the following quantity is negligible

$$\Pr\Big[f_1(x_1) = y : (\mathsf{pp}, \mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda), x_0 \leftarrow D, y := f_0(x_0), x_1 \leftarrow A(\mathsf{pp}, \left|f_0^{-1}(y)\right\rangle)\Big].$$

We say that it is infinitely-often uniform conversion hard if the above holds for infinitely many security parameters $\lambda \in \mathbb{N}$.

Next, we define (infinitely-often) one-shot signatures. We focus on the case of single-bit messages for simplicity. The message space can be extended to multiple bits by a simple parallel repetition as shown in [AGKZ20].

Definition 7.6.2 (One-shot signatures)**.** A one-shot signature scheme consists of algorithms $(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verify})$.

$\mathsf{Setup}(1^\lambda) \to \mathsf{pp}$**:** This is a PPT algorithm that takes the security parameter 1^λ as input, and outputs a classical public parameter pp.

$\mathsf{KeyGen}(\mathsf{pp}) \to (\mathsf{vk}, \mathit{sk})$**:** This is a QPT algorithm that takes a public parameter pp as input, and outputs a classical verification key vk and a quantum signing key sk.

$\mathsf{Sign}(\mathsf{pp}, \mathit{sk}, b) \to \sigma$**:** This is a QPT algorithm that takes a public parameter pp, a signing key sk and a message $b \in \{0,1\}$ as input, and outputs a classical signature σ.

$\mathsf{Verify}(\mathsf{pp}, \mathsf{vk}, b, \sigma) \to \top/\bot$**:** This is a PPT algorithm that takes a public parameter pp, a verification key vk, a message b, and a signature σ as input, and outputs the decision $\top$ or $\bot$.

We require a one-shot signature scheme to satisfy the following properties.

Correctness. For any $b \in \{0,1\}$, and for randomized procedure

$$\mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda), (\mathsf{pk}, \mathit{sk}) \leftarrow \mathsf{KeyGen}(\mathsf{pp}), \sigma \leftarrow \mathsf{Sign}(\mathsf{pp}, \mathit{sk}, b),$$

we have

$$\Pr\left[\mathsf{Verify}(\mathsf{pp}, \mathsf{vk}, b, \sigma) \to \top\right] = 1 - \mathsf{negl}(\lambda).$$

(Infinitely-often) Security. We say that a one-shot signature scheme is secure if for any non-uniform QPT adversary A, we have the negligible for all $b \in \{0,1\}$

$$\Pr\left[\mathsf{Verify}(\mathsf{pp},\mathsf{vk},b,\sigma_b) = \top : \mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda), (\mathsf{vk},\sigma_0,\sigma_1) \leftarrow A(\mathsf{pp})\right].$$

We say that it is infinitely-often secure if the above holds for infinitely many security parameters $\lambda \in \mathbb{N}$.

Then, we give a proof of Lemma 7.1.6.

Proof of Lemma 7.1.6. Since the proof is almost identical for both cases, we first prove the first item and then explain the second one.

Proof of ¬STF⇒infinite-often one-shot signature. Let $(\mathsf{Setup},\mathsf{Eval},\mathsf{Swap})$ be an STF that is claw-free but not infinitely-often uniform conversion hard. Then, there is a uniform QPT algorithm A and a polynomial poly such that

$$\Pr\left[f_1(x_1) = y : (\mathsf{pp},\mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda), x_0 \leftarrow D, y := f_0(x_0), x_1 \leftarrow A(\mathsf{pp}, \left|f_0^{-1}(y)\right\rangle)\right] \tag{7.8}$$

is not negligible for all λ. Then, we construct a one-shot signature scheme as follows. Let $N := \mathsf{poly}(\lambda)\cdot\lambda$.

$\mathsf{Setup}(1^\lambda)$**:** For $i \in [N]$, generate $(\mathsf{pp}_i,\mathsf{td}_i) \leftarrow \mathsf{Setup}(1^\lambda)$, and output $\mathsf{pp} := \{\mathsf{pp}_i\}_{i\in[N]}$. We write $f_{i,0}$ and $f_{i,1}$ to mean $f_0^{(\mathsf{pp}_i)}$ and $f_1^{(\mathsf{pp}_i)}$, respectively.

$\mathsf{KeyGen}(\mathsf{pp})$**:** Given $\mathsf{pp} = \{\mathsf{pp}_i\}_{i\in[N]}$, for $i \in [N]$, generate

$$|D\rangle = \frac{1}{|D|^{1/2}}\sum_{x\in D}|x\rangle,$$

coherently compute $f_{i,0}$ in another register to get

$$|D\rangle = \frac{1}{|D|^{1/2}}\sum_{x\in D}|x\rangle\left|f_{i,0}(x)\right\rangle,$$

measure the second register to get y_i. At this point, the first register collapses to $\left|f_{i,0}^{-1}(y_i)\right\rangle$. Output $\mathsf{vk} := \{y_i\}_{i\in[N]}$ and $sk := \{y_i, \left|f_{i,0}^{-1}(y_i)\right\rangle\}_{i\in[N]}$.

$\mathsf{Sign}(\mathsf{pp}, s\mathcal{k}, b) \to \sigma$**:** Given $\mathsf{pp} = \{\mathsf{pp}_i\}_{i\in[N]}$, $s\mathcal{k} = \{y_i, \left|f_{i,0}^{-1}(y_i)\right\rangle\}_{i\in[N]}$, and $b \in \{0,1\}$, do the following.

- If $b = 0$, for $i \in [N]$, measure $\left|f_{i,0}^{-1}(y_i)\right\rangle$ to get $x_i \in f_{i,0}^{-1}(y_i)$ and output $\sigma := \{x_i\}_{i\in[N]}$.
- If $b = 1$, for $i \in [N]$, run $A(\mathsf{pp}_i, \left|f_{i,0}^{-1}(y_i)\right\rangle)$ to get x'_i. If $f_{i,1}(x'_i) \neq y_i$ for all $i \in [N]$, it aborts. Otherwise, it outputs $\sigma := (i^*, x'_{i^*})$ where i^* is the smallest index such that $f_{i^*,1}(x'_{i^*}) = y_{i^*}$.

$\mathsf{Verify}(\mathsf{pp}, \mathsf{vk}, b, \sigma) \to \top/\bot$**:** Given $\mathsf{pp} = \{\mathsf{pp}_i\}_{i\in[N]}$, $\mathsf{vk} = \{y_i\}_{i\in[N]}$, $b \in \{0,1\}$, and a signature σ, do the following.

- If $b = 0$, parse $\sigma = \{x_i\}_{i\in[N]}$, and output $\top$ if $f_{i,0}(x_i) = y_i$ for all $i \in [N]$ and $\bot$ otherwise.
- If $b = 1$, parse $\sigma = (i, x'_i)$, and output $\top$ if $f_{i,1}(x'_i) = y_i$ and $\bot$ otherwise.

Correctness. It is easy to see that the signing algorithm outputs a valid signature whenever it does not abort. By Equation (7.8), the probability that the signing algorithm abort (when $b = 1$) is

$$(1 - 1/\mathsf{poly})^N = \mathsf{negl}(\lambda)$$

by $N = \mathsf{poly}(\lambda) \cdot \lambda$.

Security. Suppose that there is a non-uniform QPT adversary that breaks the above one-shot signature scheme. The adversary is given $\mathsf{pp} = \{\mathsf{pp}_i\}_{i\in[N]}$ and finds $\mathsf{vk} = \{y_i\}_{i\in[N]}$, $\sigma_0 = \{x_i\}_{i\in[N]}$, and $\sigma_1 = (i^*, x'_{i^*})$ such that $f_{i,0}(x_i) = y_i$ for all $i \in [N]$ and $f_{i^*,1}(x'_{i^*}) = y_{i^*}$ with a non-negligible probability. In particular, when the above happens, (x_{i^*}, x'_{i^*}) forms a claw, i.e., we have $f_{i^*,0}(x_{i^*}) = f_{i^*,1}(x'_{i^*})$. Thus, by randomly guessing i^* and embedding a problem instance of the claw-freeness into the i^*-th coordinate, we can break the claw-freeness of the STF $(\mathsf{Setup}, \mathsf{Eval}, \mathsf{Swap})$, which is a contradiction. Thus, the above one-shot signature scheme is secure.

This completes the proof of the first item.

Proof of ¬infinite-often STF⇒one-shot signature The proof is similar to the above. The difference is that since we only assume the STF is not uniform conversion hard, we can only assume that Equation (7.8) holds for infinitely many λ rather than all λ. In this case, the correctness of the above one-shot signature scheme only holds for infinitely many λ. To deal with this, we modify the verification algorithm so that it approximates A's success probability up to additive error $1/(4\mathsf{poly}(\lambda))$ (except for a negligible probability) and simply accepts if the approximated success probability is smaller than $1/(2\mathsf{poly}(\lambda))$. Then, the correctness holds on all $\lambda \in \mathbb{N}$ because

- if the real success probability is smaller than $1/(4\mathsf{poly}(\lambda))$, the estimated success probability is smaller than $1/(2\mathsf{poly}(\lambda))$ with overwhelming probability, and thus the verification algorithm accepts with overwhelming probability on these security parameters, and
- if the real success probability is larger than $1/(4\mathsf{poly}(\lambda))$, the signing algorithm should succeed in generating a valid proof with overwhelming probability and thus the verification algorithm accepts with overwhelming probability on these security parameters.

For the security, we observe that the estimated success probability is smaller than $1/(2\mathsf{poly}(\lambda))$ with a negligible probability when the real success probability is larger than $1/\mathsf{poly}(\lambda)$. Thus, for those security parameters, the adversary should find valid signatures in the original scheme. Since there are infinitely many such λ, this is not possible by the claw-freeness of the STF. □

Chapter 8

Quantum Bit Commitment

Commitments are one of the most fundamental primitives in cryptography. The commitment scheme for bit[1] allows one to commit a chosen bit while keeping it hidden to the receiver (hiding property), and enabling the sender can reveal the committed value later. But it is prohibited from changing the committed value after sending the commitment (binding property).

Formally defining the security of commitments is subtle. We may hope the hiding and binding holds for an arbitrary adversary, or statistically secure, meaning that the security holds against unbounded time adversaries. However, it is impossible to achieve both hiding and binding properties against the unbounded-time adversaries, even for quantum commitments [May97, LC97]. In practice, the one of the security is relaxed to hold only for the computationally bounded adversaries.

In cryptography, it is a common practice to relax either of security notions to hold only against computationally bounded adversaries. We say that a commitment scheme is computationally (resp. statistically) binding/hiding, if it holds against (classical or quantum depending on the context) polynomial-time (resp.

[1] We can consider commitments for multi-bit strings, but we focus on bit-commitment in this thesis.

unbounded-time) adversaries.

The impossibility introduces two *flavors* of commitments: One is computationally hiding and statistically binding and the other is computationally binding and statistically hiding, which we call the *binding* commitment and *hiding* commitment to emphasize the stronger security.

The quantum bit commitment is the commitment scheme using quantum communication. Quantum commitments have interesting features, such as the conversions between hiding and binding commitments. In particular, we can convert any binding commitment scheme into a statistically hiding interactive commitment scheme using quantum communication [CLS01]. Later, Yan [Yan20] revisits the conversion and removes the interaction as well as proving that it works for the other direction (from statistical hiding to statistical binding). This establishes *equivalence* between the *non-interactive* hiding and binding quantum commitments, which does not known in the classical setting. In fact, the classical non-interactive hiding commitment is not known and highly unlikely, as some conditional impossibility results are known [HHRS15, Fis02].

The main contribution in this chapter is an extremely efficient conversion between the flavors of quantum bit commitments. Our compiler calls the base scheme only once in superposition, whereas known compilers [CLS01, Yan20] call it $\Omega(\lambda^2)$ times for the security parameter λ.

Our new compiler, with some new and recent constructions, implies the first *efficient* constructions of the non-interactive hiding commitment from various primitives. Note that the feasibility itself is not new due to the previous known (inefficient) conversions.

8.1 Quantum Commitments

We describe the quantum bit commitment in this section. We first define a simple version of quantum bit commitment, called *canonical* quantum bit commitment,

defined in [Yan20]. Then we discuss some alternative notions of commitments and security, and provide comparisons between them.

The basic definition of quantum commitment is more involved than the canonical one as it takes multiple rounds of interactions along with multiple unitary operators. Yan showed a (round-)collapsing theorem that states any quantum bit commitment has its corresponding canonical quantum bit commitment, thus we only consider the canonical one in this thesis. See Remark 21 and the original paper for more detailed explanation.

Definition 8.1.1 (Canonical quantum bit commitments)**.** A canonical quantum bit commitment scheme is represented by a family $\{Q_0(\lambda), Q_1(\lambda)\}_{\lambda\in\mathbb{N}}$ of polynomial-time computable unitaries over two registers $\mathbf{C}$ (called the commitment register) and $\mathbf{R}$ (called the reveal register). We often omit λ and simply write Q_0 and Q_1 to mean $Q_0(\lambda)$ and $Q_1(\lambda)$. The canonical quantum bit commitment is supposed to proceed as follows.

1. In the *commitment* phase, to commit a bit $b \in \{0,1\}$, the sender computes $Q_b |0\rangle_{\mathbf{C},\mathbf{R}}$ and sends $\mathbf{C}$ to the receiver.

2. In the *reveal* phase, the sender sends the register $\mathbf{R}$ along with b to the receiver. Then the receiver projects the state on $(\mathbf{C},\mathbf{R})$ onto $Q_b |0\rangle_{\mathbf{C},\mathbf{R}}$ and accepts if it succeeds, and rejects otherwise.

The hiding and binding are defined as follows.

Definition 8.1.2 (Hiding)**.** We say that a canonical quantum bit commitment scheme $\{Q_0, Q_1\}$ is computationally (rep. statistically) hiding if $\mathrm{Tr}_{\mathbf{R}}(Q_0(|0\rangle\langle 0|)_{\mathbf{C},\mathbf{R}}Q_0^\dagger)$ is computationally (resp. statistically) indistinguishable from $\mathrm{Tr}_{\mathbf{R}}(Q_1(|0\rangle\langle 0|)_{\mathbf{C},\mathbf{R}}Q_1^\dagger)$. We say that it is perfectly hiding if they are identical states.

Definition 8.1.3 (Binding)**.** We say that a canonical quantum bit commitment scheme $\{Q_0, Q_1\}$ is computationally (rep. statistically) binding if for any polynomial-time (resp. unbounded-time) unitary U over $\mathbf{R}$ and an additional register $\mathbf{Z}$ and any

polynomial-size state $|\tau\rangle_{\mathbf{Z}}$, it holds that

$$\left\|(Q_1 |0\rangle \langle 0| Q_1^\dagger)_{\mathbf{C},\mathbf{R}}(I_{\mathbf{C}} \otimes U_{\mathbf{R},\mathbf{Z}})((Q_0 |0\rangle)_{\mathbf{C},\mathbf{R}} |\tau\rangle_{\mathbf{Z}})\right\| = \mathsf{negl}(\lambda).$$

We say that it is perfectly hiding if the left hand side is 0 for all unbounded-time unitary U.

Remark 21 (Canonical form suffices)**.** We note that one of the main contribution of [Yan20] is the round-collapsing theorem of quantum bit commitment, meaning that any quantum bit commitment has its corresponding non-interactive canonical form. This result significantly simplifies the overall analysis and allows new conversion theorems by Yan and ours.

Remark 22 (Definition of binding)**.** We use so-called *honest*-binding as a default definition of binding. There are several other definitions of binding for quantum commitments. We review the other notations and give comparisons with honest-binding.

The notion of *classical-binding* was recently introduced by Bitanski and Brakerski [BB21], which roughly requires that the committed message is uniquely determined by the commitment. Though this is impossible to achieve for canonical quantum bit commitments, they avoid the impossibility by having the receiver *measure* the commitment in a certain way. The advantage of the classical binding property is that it is conceptually similar to the binding of classical commitments, and thus it is easy to give security proofs when plugging it into some protocol as a substitute for classical commitments. On the other hand, existing works [YWLQ15, FUYZ20, MY21] show that the statistical honest-binding quantum commitments are already useful for many applications. Indeed, there seems no known application for which classical-binding suffices but honest-binding does not.

Ananth, Qian, and Yuen [AQY21] introduced a new definition of a statistical binding property for quantum commitments, which we call AQY-binding. The

motivation of this definition is for the application to quantum oblivious transfers and multi-party computation [BCKM21]. However, [MY21, Appendix B] observed that the statistical honest-binding property implies the AQY-binding property based on the technique of [FUYZ20]. A full proof is given in [Yan20, Appendix B].

Yan [Yan21] proved that the *computational* honest-binding property implies what is called the computational *predicate-binding* property, which is sufficient for implementing Blum's Hamiltonicity protocol.

There are several other definitions of *computational* binding for quantum (string) commitments [CDMS04, DFS04] that are shown to be more useful in applications than computational honest binding ones. However, there is no known construction that satisfies the definition of [CDMS04], and the only known construction that satisfies [DFS04] is in the CRS model and based on a special assumption that is tailored to their construction. (See [Unr16, Yan21] for more details of these definitions.)

Remark 23 (Physical assumptions)**.** Some works [Sal98, DFSS08] suggested to use physical assumptions such as the bounded quantum memory for constructing the quantum commitments or other cryptographic primitives. Interestingly, their constructions detour the impossibility results of [May97, LC97]. Still, we do not use such physical assumptions in this thesis, and focus on the standard model, e.g., without quantum memory limits.

8.2 Efficient Conversion

The main theorem of this section is the efficiency-preserving conversion theorem. Since we are considering various security notions, we set the following throughout in this section.

$$\mathtt{X}, \mathtt{Y} \in \{\text{computationally,statistically,perfectly}\}$$

Let $Q = \{Q_0, Q_1\}$ be a canonical quantum bit commitment scheme and $\mathbf{C}$ and $\mathbf{R}_0$ be the commitment and reveal registers of Q. The following unitary map represents the canonical quantum bit commitment

$$V_Q = Q_0 \otimes |0\rangle\langle 0|_{\mathbf{B}} + Q_1 \otimes |1\rangle\langle 1|_{\mathbf{B}}$$

which we call the representing unitary of Q, or unitary commitment. Let $\mathbf{R} = (\mathbf{R}_0, \mathbf{B})$ to include the augmented register as the output of commitment so that $\mathbf{C}$ and $\mathbf{R}$ are the commitment and reveal registers of V_Q. Note that the augmented register $\mathbf{B}$ does not affect the security of commitment scheme because the hiding adversary cannot see the reveal register and the binding adversary already knows the committed bit b.

The dual[2] unitary commitment Q^* employs the same representing unitary V_Q to commit $|\pm\rangle_{\mathbf{B}}$, but reverses the role of $\mathbf{C}$ and $\mathbf{R}$, that is, $\mathbf{C}^* = \mathbf{R}$ and $\mathbf{R}^* = \mathbf{C}$ are the commitment and reveal registers, respectively. We may consider Q^* commits the bit b by applying V_Q to $H|b\rangle$, i.e., the representing unitary of Q^* is $V_Q \cdot H_{\mathbf{B}}$.

Let us write the overall states of commitment Q for bit b by $Q(b)$. The concrete commitment computations are as follows.

$$Q(0) = Q_0 |0\rangle |0\rangle_{\mathbf{B}}, \qquad Q(1) = Q_1 |0\rangle |1\rangle_{\mathbf{B}},$$
$$Q^*(0) = \frac{Q_0 |0\rangle |0\rangle_{\mathbf{B}} + Q_1 |0\rangle |1\rangle_{\mathbf{B}}}{\sqrt{2}}, \qquad Q^*(1) = \frac{Q_0 |0\rangle |0\rangle_{\mathbf{B}} - Q_1 |0\rangle |1\rangle_{\mathbf{B}}}{\sqrt{2}}$$

This makes the consistency between the duality of quantum states in the equivalence theorem and the duality of quantum bit commitment. We show the following theorem.

Theorem 8.2.1 (Converting Flavors)**.** *Let $Q = \{Q_0, Q_1\}$ be a canonical quantum bit commitment scheme with the commitment and reveal registers $\mathbf{C}$ and $\mathbf{R} = (\mathbf{R}_0, \mathbf{B})$, and V_Q be the representing unitary $V_Q = Q_0 \otimes |0\rangle\langle 0|_{\mathbf{B}} + Q_1 \otimes |1\rangle\langle 1|_{\mathbf{B}}$. Let Q^* be the dual commitment of Q defined by the representing unitary $V_Q^* = V_Q \cdot H_{\mathbf{B}}$*

[2]The conjugate transpose is written by $U^\dagger$ in this thesis.

with the reversed registers $\mathbf{C}' = \mathbf{R}$ *and* $\mathbf{R}' = \mathbf{C}$. *The following statements hold for any* $\mathtt{X}, \mathtt{Y} \in \{$*computationally,statistically,perfectly*$\}$.

1. *Q is* $\mathtt{Y}$ *hiding if and only if* Q^* *is* $\mathtt{Y}$ *binding.*

2. *Q is* $\mathtt{X}$ *binding if and only if* Q^* *is* $\mathtt{X}$ *hiding.*

Proof sketch. We focus on $\mathtt{X}, \mathtt{Y}$="computational", and the other cases are almost identical.

For Item 1, let U be the binding adversary for Q^*, which only acts on the registers $\mathbf{R}' = \mathbf{C}$ and $\mathbf{Z}$. The advantage Δ of binding adversary U is the maximum of

$$\|(Q^*(1)^\dagger)(I \otimes U)(Q^*(0)\,|\tau\rangle_{\mathbf{Z}})\|,\ \|(Q^*(0)^\dagger)(I \otimes U)(Q^*(1)\,|\tau\rangle_{\mathbf{Z}})\|.$$

Using the fact that U does not touch the register $\mathbf{B}$, it follows from the straightforward calculation that

$$(Q^*(1)^\dagger)(I \otimes U)(Q^*(0)\,|\tau\rangle_{\mathbf{Z}}) = (Q^*(0)^\dagger)(I \otimes U)(Q^*(1)\,|\tau\rangle_{\mathbf{Z}})$$

which implies that the advantage Δ of U is

$$\frac{\|(Q^*(1)^\dagger)(I \otimes U)(Q^*(0)\,|\tau\rangle_{\mathbf{Z}}) + (Q^*(0)^\dagger)(I \otimes U)(Q^*(1)\,|\tau\rangle_{\mathbf{Z}})\|}{2}.$$

Theorem 6.2.2 states that this U is equivalent to the distinguishing algorithm A for dual states $Q(0)$ and $Q(1)$, which takes a quantum advice and acts only on $\mathbf{C}$ and $\mathbf{Z}$. In other words, the binding adversary U for Q^* implies the hiding adversary for Q, and vice versa.

The proof of Item 2 is also a direct application of Theorem 6.2.2, after correcting the phase and sign of binding adversary for Q, which is possible because the binding adversary accesses the register $\mathbf{B}$. □

8.3 Applications of Conversion

In this section, we show applications of our conversion (Theorem 8.2.1) to the quantum bit commitments.

When we describe a canonical quantum bit commitment scheme $\{Q_0, Q_1\}$, we only describe how Q_0 and Q_1 act on $|0\rangle$ for simplicity. Quantum circuits that implement Q_0 and Q_1 can be defined in a natural way.

8.3.1 Construction from PRG

Naor [Nao91] constructed a *classical* commitment scheme that is computationally hiding and statistically binding based on PRGs (See Definition 2.3.2). Yan et al. [YWLQ15] constructed a quantum *non-interactive* version of Naor's commitment.[3] Let $G : \{0,1\}^n \to \{0,1\}^{3n}$ be a PRG. Then Yan et al.'s commitment scheme $\{Q_{\mathsf{YWLQ},0}, Q_{\mathsf{YWLQ},1}\}$ is described as follows:

$$Q_{\mathsf{YWLQ},0}\,|0\rangle_{\mathbf{C},\mathbf{R}} := \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |G(x)\rangle_{\mathbf{C}} \left|x, 0^{2n}\right\rangle_{\mathbf{R}}$$
$$Q_{\mathsf{YWLQ},1}\,|0\rangle_{\mathbf{C},\mathbf{R}} := \frac{1}{\sqrt{2^{3n}}} \sum_{y\in\{0,1\}^{3n}} |y\rangle_{\mathbf{C}}\,|y\rangle_{\mathbf{R}}\,.$$

Yan et al. [YWLQ15] proved the following theorem.

Theorem 8.3.1 ([YWLQ15])**.** *If G is a PRG, then $\{Q_{\mathsf{YWLQ},0}, Q_{\mathsf{YWLQ},1}\}$ is computationally hiding and statistically binding.*

To apply our conversion theorem, we consider the following dual scheme $\{Q'_{\mathsf{YWLQ},0}, Q'_{\mathsf{YWLQ},1}\}$ with the state $Q'_{\mathsf{YWLQ},b}\,|0\rangle_{\mathbf{C},\mathbf{R}}$ of the form

$$\frac{1}{\sqrt{2^{n+1}}} \sum_{x\in\{0,1\}^n} \left|0, x, 0^{2n}\right\rangle_{\mathbf{C}'} |G(x)\rangle_{\mathbf{R}'} + (-1)^b \frac{1}{\sqrt{2^{3n+1}}} \sum_{y\in\{0,1\}^{3n}} |1, y\rangle_{\mathbf{C}'}\,|y\rangle_{\mathbf{R}'}\,.$$

[3] Yan [Yan20, Appendix C] shows an alternative more direct translation of Naor's commitment to the quantum setting. We could also apply our conversion to that scheme, but we focus on the scheme of [YWLQ15] since that is simpler.

By Theorems 8.2.1 and 8.3.1, we obtain the following theorem.

Theorem 8.3.2. *If G is a PRG, then $\{Q'_{\mathsf{YWLQ},0}, Q'_{\mathsf{YWLQ},1}\}$ is statistically hiding and computationally binding. In particular, if there exists a quantum-secure pseudorandom generator, then there is a statistical hiding and computational binding quantum bit commitment that makes only a single call to the PRG.*

We note that if we apply existing conversions [CLS01, Yan20] to the commitment $\{Q_{\mathsf{YWLQ},0}, Q_{\mathsf{YWLQ},1}\}$ (or other PRG-based schemes), they result in schemes that make $\Omega(\lambda^2)$ calls to the PRG.

Remark 24 (On PRGs based on OWFs)**.** It is known that PRG exists assuming the existence of one-way functions [HILL99].[4] In the current state of the art, a construction of PRG makes at least $\Omega(\lambda^3)$ calls to the base one-way function [HRV13, VZ12]. Thus, if we construct a PRG from a one-way function and count the number of calls to the one-way function, $\{Q'_{\mathsf{YWLQ},0}, Q'_{\mathsf{YWLQ},1}\}$ makes $\Omega(\lambda^3)$ calls to the one-way function.

We observe that this is asymptotically the same number as that of Koshiba and Odaira [KO11]. However, it does not seem possible to instantiate the scheme of [KO11] with a single call to a PRG instead of $\Omega(\lambda^3)$ calls to a one-way function. Also, our security analysis is much simpler than theirs once we establish Theorem 8.2.1.

8.3.2 Construction from Pseudorandom State Generators

Ananth, Qian, Yuen [AQY21], and Morimae and Yamakawa [MY21] concurrently showed that a primitive called pseudorandom state generators (PRSGs) [JLS18] can be used to construct computationally hiding and statistically binding quantum

[4]Though the original security proof in [HILL99] only considers classical adversaries, it also works against quantum adversaries as well assuming quantum-secure one-way functions.

bit commitments. Especially, Morimae and Yamakawa [MY21, footnote 12] mentioned that replacing PRGs with single-copy secure PRSGs in $\{Q_{\mathsf{YWLQ},0}, Q_{\mathsf{YWLQ},1}\}$ yields a computationally hiding and statistically binding scheme.

Let StateGen be a single-copy-secure PRSG that, on input $k \in \{0,1\}^n$, outputs an m-qubit state $|\phi_k\rangle$ where $m = 3n$. Then, Morimae and Yamakawa's commitment scheme $\{Q_{\mathsf{MY},0}, Q_{\mathsf{MY},1}\}$ is described as follows:

$$Q_{\mathsf{MY},0}\,|0\rangle_{\mathbf{C},\mathbf{R}} := \frac{1}{\sqrt{2^n}} \sum_{k\in\{0,1\}^n} |\phi_k\rangle_{\mathbf{C}} \left|k, 0^{2n}\right\rangle_{\mathbf{R}}$$
$$Q_{\mathsf{MY},1}\,|0\rangle_{\mathbf{C},\mathbf{R}} := \frac{1}{\sqrt{2^{3n}}} \sum_{r\in\{0,1\}^{3n}} |r\rangle_{\mathbf{C}} |r\rangle_{\mathbf{R}} .$$

Theorem 8.3.3. *If* StateGen *is single-copy-secure, then* $\{Q_{\mathsf{MY},0}, Q_{\mathsf{MY},1}\}$ *is computationally hiding and statistically binding.*

The proof of the above theorem is not included in [MY21] as it was not the main construction. We give a security proof of this commitment for completeness.

Proof of Theorem 8.3.3. We let $|\psi_b\rangle_{\mathbf{C},\mathbf{R}} := Q_{\mathsf{MY},b}\,|0\rangle_{\mathbf{C},\mathbf{R}}$.

Computational hiding. Note that $\mathrm{Tr}_{\mathbf{R}}\left(|\psi_1\rangle\langle\psi_1|_{\mathbf{C},\mathbf{R}}\right)$ is a maximally mixed state, which is a Haar random state when given a single copy. On the other hand, we have $\mathrm{Tr}_{\mathbf{R}}\left(|\psi_0\rangle\langle\psi_0|_{\mathbf{C},\mathbf{R}}\right) = \frac{1}{2^n}\sum_{k\in\{0,1\}^n} |\phi_k\rangle\langle\phi_k|$. Thus, the computational hiding property immediately follows from the single-copy security of StateGen.

Statistical binding. The proof is similar to the proof of binding in [MY21]. Let $F(\rho,\sigma)$ be the fidelity between ρ and σ. Then, we have

$$
\begin{aligned}
&F\Big(\mathrm{Tr}_{\mathbf{R}}(|\psi_0\rangle\langle\psi_0|_{\mathbf{C},\mathbf{R}}), \mathrm{Tr}_{\mathbf{R}}(|\psi_1\rangle\langle\psi_1|_{\mathbf{C},\mathbf{R}})\Big)\\
&= F\Big(\frac{1}{2^n}\sum_k |\phi_k\rangle\langle\phi_k|, \frac{I^{\otimes m}}{2^m}\Big)\\
&= \Big\| \sum_{i=1}^{\xi} \sqrt{\lambda_i}\frac{1}{\sqrt{2^m}}|\lambda_i\rangle\langle\lambda_i| \Big\|_1^2\\
&= \left(\sum_{i=1}^{\xi} \sqrt{\lambda_i}\frac{1}{\sqrt{2^m}}\right)^2\\
&\leq \left(\sum_{i=1}^{\xi} \lambda_i\right)\left(\sum_{i=1}^{\xi} \frac{1}{2^m}\right)\\
&\leq 2^{-2n}.
\end{aligned}
$$

where in the second equality, $\sum_{i=1}^{\xi} \lambda_i|\lambda_i\rangle\langle\lambda_i|$ is the diagonalization of $\frac{1}{2^n}\sum_k |\phi_k\rangle\langle\phi_k|$, in the first inequality, we have used Cauchy–Schwarz inequality, and in the final inequality, we have used $\xi \leq 2^n$ and $m = 3n$. This means that $\{Q_{\mathsf{MY},0}, Q_{\mathsf{MY},1}\}$ is statistically binding. □

By applying our conversion to the commitment scheme $\{Q_{\mathsf{MY},0}, Q_{\mathsf{MY},1}\}$, we obtain the following dual scheme $\{Q'_{\mathsf{MY},0}, Q'_{\mathsf{MY},1}\}$.[5]

$$
Q'_{\mathsf{MY},b}|0\rangle_{\mathbf{C},\mathbf{R}} := \frac{1}{\sqrt{2^{n+1}}}\sum_{k\in\{0,1\}^n} \left|0,k,0^{2n}\right\rangle_{\mathbf{C}'} |\phi_k\rangle_{\mathbf{R}'} + (-1)^b \frac{1}{\sqrt{2^{3n+1}}}\sum_{r\in\{0,1\}^{3n}} |1,r\rangle_{\mathbf{C}'}|r\rangle_{\mathbf{R}'}\,.
$$

Combining Theorems 8.2.1 and 8.3.3, we obtain the following theorem.

Theorem 8.3.4. *If* StateGen *is single-copy-secure, then* $\{Q'_{\mathsf{MY},0}, Q'_{\mathsf{MY},1}\}$ *is statistically hiding and computationally binding. In particular, if there exists a single-copy-secure pseudorandom quantum state generator, then there is a statistical hiding and computational binding quantum bit commitment that makes only a single call to the PRSG.*

[5] We could apply our conversion to the main construction of [MY21] to obtain a similar scheme.

This is the first statistically hiding and computationally binding quantum bit commitment scheme from PRSGs that makes only a single call to the PRSG. If we apply existing conversions [CLS01, Yan20] to $\{Q_{\mathsf{MY},0}, Q_{\mathsf{MY},1}\}$ (or other PRSG-based schemes [AQY21]), they result in a schemes that make $\Omega(\lambda^2)$ calls to the PRSG.

8.3.3 Construction from Injective One-Way Functions

In this section, we show simple constructions of commitments based on any injective one-way functions.

Perfectly hiding and computationally binding commitment. We first construct a perfectly hiding and computationally binding quantum bit commitment scheme from injective one-way function. We note that such a commitment is already known from any one-way *permutations* in [DMS00]. Our construction is more general since every permutation is also injective but the converse is not true.

Let $f : \{0,1\}^n \to \{0,1\}^m$ be an injective one-way function. Then, we define a canonical quantum bit commitment scheme $\{Q_{\mathsf{inj},0}, Q_{\mathsf{inj},1}\}$ as follows:

$$Q_{\mathsf{inj},0} |0\rangle_{\mathbf{C},\mathbf{R}} := \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle_{\mathbf{C}} |f(x)\rangle_{\mathbf{R}}$$
$$Q_{\mathsf{inj},1} |0\rangle_{\mathbf{C},\mathbf{R}} := \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle_{\mathbf{C}} \left|x, 0^{m-n}\right\rangle_{\mathbf{R}}.$$

Theorem 8.3.5. *If f is an injective one-way function, $\{Q_{\mathsf{inj},0}, Q_{\mathsf{inj},1}\}$ is perfectly hiding and computationally binding.*

Proof. Due to the injectivity of f, if we trace out $\mathbf{R}$, then the reduced state in $\mathbf{C}$ is $\sum_{x\in\{0,1\}^n} |x\rangle\langle x|$ for both $b = 0, 1$. This implies perfect hiding. We focus on the computational binding below.

Suppose that the $\{Q_{\mathsf{inj},0}, Q_{\mathsf{inj},1}\}$ is not computationally binding. Then there exists a polynomial-time computable unitary U over $(\mathbf{R}, \mathbf{Z})$ and an auxiliary state $|\tau\rangle_{\mathbf{Z}}$

such that

$$\left\|(Q_{\mathsf{inj},1}\left|0\right\rangle\left\langle 0\right|Q_{\mathsf{inj},1}^{\dagger})_{\mathbf{C},\mathbf{R}}(I_{\mathbf{C}}\otimes U_{\mathbf{R},\mathbf{Z}})((Q_{\mathsf{inj},0}\left|0\right\rangle)_{\mathbf{C},\mathbf{R}}\left|\tau\right\rangle_{\mathbf{Z}})\right\|$$

is non-negligible. In particular, its square is also non-negligible. It holds that

$$\begin{aligned}
&\left\|(Q_{\mathsf{inj},1}\left|0\right\rangle\left\langle 0\right|Q_{\mathsf{inj},1}^{\dagger})_{\mathbf{C},\mathbf{R}}(I_{\mathbf{C}}\otimes U_{\mathbf{R},\mathbf{Z}})((Q_{\mathsf{inj},0}\left|0\right\rangle)_{\mathbf{C},\mathbf{R}}\left|\tau\right\rangle_{\mathbf{Z}})\right\|^2 \\
&= \frac{1}{2^{2n}}\left\|\sum_{x\in\{0,1\}^n}\left\langle x,0^{m-n}\right|_{\mathbf{R}}U_{\mathbf{R},\mathbf{Z}}\left|f(x)\right\rangle_{\mathbf{R}}\left|\tau\right\rangle_{\mathbf{Z}}\right\|^2 \\
&\leq \frac{1}{2^{2n}}\left(\sum_{x\in\{0,1\}^n}\left\|\left\langle x,0^{m-n}\right|_{\mathbf{R}}U_{\mathbf{R},\mathbf{Z}}\left|f(x)\right\rangle_{\mathbf{R}}\left|\tau\right\rangle_{\mathbf{Z}}\right\|\right)^2 \\
&\leq \frac{1}{2^{n}}\sum_{x\in\{0,1\}^n}\left\|\left\langle x,0^{m-n}\right|_{\mathbf{R}}U_{\mathbf{R},\mathbf{Z}}\left|f(x)\right\rangle_{\mathbf{R}}\left|\tau\right\rangle_{\mathbf{Z}}\right\|^2, \qquad (8.1)
\end{aligned}$$

where the first equality follows from the definition of $\{Q_{\mathsf{inj},0}, Q_{\mathsf{inj},1}\}$, the first inequality follows from the triangle inequality, and the second inequality follows from the Cauchy–Schwarz inequality. Thus, the value of Equation (8.1) is non-negligible.

Then, we can construct an adversary A that breaks the one-wayness of f with advice $|\tau\rangle$ as follows:

$A(y;|\tau\rangle)$**:** Given an instance y and advice $|\tau\rangle$, it generates a state $U\left|y\right\rangle_{\mathbf{R}}\left|\tau\right\rangle_{\mathbf{Z}}$ and measures $\mathbf{R}$. If the measurement outcome is $(x,0^{m-n})$ such that $f(x)=y$, it outputs x and otherwise $\perp$.

We can see that the probability that A outputs the correct preimage x is the value of Equation (8.1), which is non-negligible. This contradicts the one-wayness of f. Thus, $\{Q_{\mathsf{inj},0}, Q_{\mathsf{inj},1}\}$ is computationally binding. □

This is the first *perfectly* hiding quantum bit commitment scheme from injective one-way functions that makes only a single quantum call to the base function.

Prior to our work, such a commitment scheme was only known to exist from one-way *permutations* [DMS00]. We remark that Koshiba and Odaira [KO09, KO11] generalized [DMS00] to make the assumption weaker than the existence of injective one-way functions, but those constructions only achieve *statistical* hiding.

Alternatively, we can also construct such a commitment scheme by applying our conversion to the (purified version of) construction of computationally hiding and perfectly binding commitment scheme based on Goldreich-Levin theorem [GL89].

Computationally hiding and perfectly binding commitment. Next, we apply our conversion to $\{Q_{\mathsf{inj},0}, Q_{\mathsf{inj},1}\}$ to obtain the following scheme $\{Q'_{\mathsf{inj},0}, Q'_{\mathsf{inj},1}\}$:

$$Q'_{\mathsf{inj},b}|0\rangle_{\mathbf{C}',\mathbf{R}'} := \frac{1}{\sqrt{2^{n+1}}}\sum_{x\in\{0,1\}^n}\left(\left(|0\rangle|f(x)\rangle + (-1)^b|1\rangle\left|x,0^{m-n}\right\rangle\right)_{\mathbf{C}'}|x\rangle_{\mathbf{R}'}\right).$$

By Theorems 8.2.1 and 8.3.5, we obtain the following theorem.

Theorem 8.3.6. *If f is an injective one-way function, $\{Q'_{\mathsf{inj},0}, Q'_{\mathsf{inj},1}\}$ is computationally hiding and perfectly binding.*

Comparison with classical construction. It is well-known that we can *classically* construct a computationally hiding and perfectly binding non-interactive commitment scheme from injective one-way functions by using Goldreich-Levin theorem [GL89]. The construction also only makes a single call to the base function. Then, one may wonder if it is meaningful to give a *quantum* construction for that. We argue this by remarking the following two points.

1. A minor parameter improvement. Our construction has a shorter commitment size than the classical construction (albeit with the apparent disadvantage of the usage of quantum communication). Specifically, commitment length of our construction is $m + 1$ whereas it is $n + m + 1$ in the classical construction. The additional n-bit is needed to send the seed for the hardcore bit function in the classical construction.

We remark that the decommitment length is the same, n for both constructions. Though the improvement is somewhat minor, we believe that it is still worthwhile to show that the quantum communication can reduce the communication complexity of such an important construction of commitments from injective one-way functions.

2. The second is rather conceptual. We remark that our construction does not make use of any sort of classical hardcore predicates. On the other hand, to our knowledge, the only known way to classically construct a commitment scheme from injective one-way functions (or even one-way permutations) is to rely on some hardcore predicates [GL89, GRS00, HMS04]. Thus, the source of the pseudorandomness of our construction seems conceptually very different from that for classical constructions. In a nutshell, we interpret the theorem shown by [AAS20] in a completely irrelevant context as a kind of search-to-decision reduction. We believe that this new search-to-decision reduction technique is interesting and will be useful in the future work.

Construction from keyed injective one-way functions. Unfortunately, there is no known candidate of post-quantum injective one-way functions based on standard assumptions.[6] On the other hand, there are many candidates of *keyed* injective one-way functions. We remark that our construction can be easily extended to one based on keyed injective one-way functions. Let $\{f_k : \{0,1\}^n \to \{0,1\}^m\}_{k\in\mathcal{K}}$ be a keyed injective one-way function. Then, we construct a modified scheme

[6] Candidate constructions of post-quantum injective one-way functions based on hash functions or block ciphers can be found in [Unr12, Section 5].

$\{Q_{\mathsf{keyed-inj},0}, Q_{\mathsf{keyed-inj},1}\}$ as follows:

$$Q_{\mathsf{keyed-inj},0}\,|0\rangle_{\mathbf{C},\mathbf{R}} := \frac{1}{\sqrt{2^n|\mathcal{K}|}} \sum_{x\in\{0,1\}^n, k\in\mathcal{K}} |x,k\rangle_{\mathbf{C}}\,|f_k(x),k\rangle_{\mathbf{R}}$$
$$Q_{\mathsf{keyed-inj},1}\,|0\rangle_{\mathbf{C},\mathbf{R}} := \frac{1}{\sqrt{2^n|\mathcal{K}|}} \sum_{x\in\{0,1\}^n} |x,k\rangle_{\mathbf{C}}\,\left|x,0^{m-n},k\right\rangle_{\mathbf{R}}$$

We can show that $\{Q_{\mathsf{keyed-inj},0}, Q_{\mathsf{keyed-inj},1}\}$ is perfectly hiding and computationally binding similarly to the proof of Theorem 8.3.5. Then, by applying our conversion, we obtain the following scheme $\{Q'_{\mathsf{keyed-inj},0}, Q'_{\mathsf{keyed-inj},1}\}$ with the state $Q'_{\mathsf{keyed-inj},b}\,|0\rangle_{\mathbf{C}',\mathbf{R}'}$ equal to

$$\frac{1}{\sqrt{2^{n+1}|\mathcal{K}|}} \sum_{x\in\{0,1\}^n, k\in\mathcal{K}} \left(\left(|0\rangle\,|f_k(x),f_k\rangle + (-1)^b\,|1\rangle\left|x,0^{m-n},f_k\right\rangle\right)_{\mathbf{C}'} |x,k\rangle_{\mathbf{R}'}\right).$$

By Theorem 8.2.1, $\{Q'_{\mathsf{keyed-inj},0}, Q'_{\mathsf{keyed-inj},1}\}$ is computationally binding and statistically hiding.

We remark that we can also view it as a quantum-ciphertext PKE (Definition 7.2.1) if we assume that f_k is a trapdoor function. That is, we can use

$$\mathrm{Tr}_{\mathbf{R}'}\left(Q'_{\mathsf{keyed-inj},b}\,|0\rangle_{\mathbf{C}',\mathbf{R}'}\,\langle 0|_{\mathbf{C}',\mathbf{R}'}\,Q'^{\dagger}_{\mathsf{keyed-inj},b}\right)$$

as an encryption of b. We can decrypt it with a trapdoor for f_k by applying a unitary $|x,0^{m-n},f_k\rangle \mapsto |f_k(x),f_k\rangle$ on the second register of $\mathbf{C}'$ controlled on the first register of $\mathbf{C}'$ (which is efficiently computable with the trapdoor) and then measuring the first register of $\mathbf{C}'$ in the Hamadard basis. The IND-CPA security directly follows from the computational hiding property of $\{Q'_{\mathsf{keyed-inj},0}, Q'_{\mathsf{keyed-inj},1}\}$. This gives a conceptually different way to construct (quantum-ciphertext) PKE from trapdoor functions than that based on hardcore predicates.

8.3.4 Construction from Collapsing Functions

In this section, we show simple constructions of commitments based on collapsing functions. Interestingly, the constructions are almost identical to those based on

injective one-way functions given in Section 8.3.3, but they achieve the different flavors of security than those based on injective one-way functions.

Computationally hiding and statistically binding commitment. We first construct a computationally hiding and statistically binding quantum bit commitment scheme from collapsing functions (Definition 2.3.4).

Let $\{H_k : \{0,1\}^n \to \{0,1\}^m\}_{k\in\mathcal{K}}$ be a family of collapsing functions such that $n \geq m + \lambda$. Then, we define a canonical quantum bit commitment scheme $\{Q_{\mathsf{col},0}, Q_{\mathsf{col},1}\}$ as follows:

$$Q_{\mathsf{col},0}\left|0\right\rangle_{\mathbf{C},\mathbf{R}} := \frac{1}{\sqrt{2^n|\mathcal{K}|}}\sum_{x\in\{0,1\}^n,k\in\mathcal{K}}\left|x,k\right\rangle_{\mathbf{C}}\left|H_k(x),0^{n-m},k\right\rangle_{\mathbf{R}}$$
$$Q_{\mathsf{col},1}\left|0\right\rangle_{\mathbf{C},\mathbf{R}} := \frac{1}{\sqrt{2^n|\mathcal{K}|}}\sum_{x\in\{0,1\}^n,k\in\mathcal{K}}\left|x,k\right\rangle_{\mathbf{C}}\left|x,k\right\rangle_{\mathbf{R}}.$$

Theorem 8.3.7. *If $\{H_k : \{0,1\}^n \to \{0,1\}^m\}_{k\in\mathcal{K}}$ is a family of collapsing functions such that $n \geq m + \lambda$, $\{Q_{\mathsf{inj},0}, Q_{\mathsf{inj},1}\}$ is computationally hiding and statistically binding.*

Proof. We first prove the computational hiding.

Computational hiding. We have

$$\mathrm{Tr}_{\mathbf{R}}(Q_{\mathsf{col},0}\left|0\right\rangle_{\mathbf{C},\mathbf{R}}) = \frac{1}{|\mathcal{K}|}\sum_{y\in\{0,1\}^m,k\in\mathcal{K}}\frac{|S_{k,y}|}{2^n}\left(\frac{1}{\sqrt{|S_{k,y}|}}\sum_{x\in S_{k,y}}\left|x,k\right\rangle\right)\left(\frac{1}{\sqrt{|S_{k,y}|}}\sum_{x'\in S_{k,y}}\left\langle x',k\right|\right)$$

where

$$S_{k,y} := \{x \in \{0,1\}^n : H_k(x) = y\}.$$

Then, by the collapsing property of $\{H_k\}_{k\in\mathcal{K}}$, we can show that $\mathrm{Tr}_{\mathbf{R}}(Q_{\mathsf{col},0}\left|0\right\rangle_{\mathbf{C},\mathbf{R}})$ is computationally indistinguishable from

$$\frac{1}{2^n|\mathcal{K}|}\sum_{x\in\{0,1\}^n,k\in\mathcal{K}}\left|x,k\right\rangle\left\langle x,k\right|.$$

The above state is exactly the same as $\mathrm{Tr}_{\mathbf{R}}(Q_{\mathsf{col},1}\left|0\right\rangle_{\mathbf{C},\mathbf{R}})$. Thus, the computational hiding property is proven.

Statistical binding. Suppose that the $\{Q_{\mathsf{col},0}, Q_{\mathsf{col},1}\}$ is not statistically binding. Then by a similar argument to that for the proof of computational binding of $\{Q_{\mathsf{inj},0}, Q_{\mathsf{inj},1}\}$ in Section 8.3.3, we can construct an unbounded-time adversary A such that

$$\Pr[A(k, H_k(x)) = x : k \leftarrow \mathcal{K}, x \leftarrow \{0,1\}^n]$$

is non-negligible. However, this is information-theoretically impossible since $n \geq m + \lambda$. Thus, $\{Q_{\mathsf{col},0}, Q_{\mathsf{col},1}\}$ is statistically binding. □

This is the first *statistically* binding quantum bit commitment scheme from collapsing functions that makes only a single quantum call to the base function. To our knowledge, the only known way to construct statistically binding (classical or quantum) commitments from collapsing functions (or collision-resistant functions in the classical case) is to first construct PRGs regarding collapsing (or collision-resistant) functions as one-way functions and then convert it to commitments by [Nao91]. This requires super-constant number of calls to the base function since known constructions of PRGs from one-way functions require super-constant number of calls [HILL99, HRV13, VZ12].

Note that post-quantum statistically *hiding* commitments from collapsing functions are known [HM96, Unr16]. Thus, by applying our conversion to the purified version of the scheme, we can obtain an alternative construction of statistically binding commitments from collapsing functions.

Statistically hiding and computationally binding commitment. Next, we apply our conversion to $\{Q_{\mathsf{col},0}, Q_{\mathsf{col},1}\}$ to obtain the following scheme $\{Q'_{\mathsf{col},0}, Q'_{\mathsf{col},1}\}$:

$$Q'_{\mathsf{col},b}|0\rangle_{\mathbf{C}',\mathbf{R}'} := \frac{1}{\sqrt{2^{n+1}|\mathcal{K}|}} \sum_{x\in\{0,1\}^n} \left(\left(|0\rangle \left|H_k(x), 0^{n-m}, k\right\rangle + (-1)^b |1\rangle |x,k\rangle\right)_{\mathbf{C}'} |x,k\rangle_{\mathbf{R}'}\right).$$

By Theorems 8.2.1 and 8.3.7, we obtain the following theorem.

Theorem 8.3.8. *If $\{H_k\}_{k\in\mathcal{K}}$ is a family of collapsing functions, $\{Q'_{\mathsf{col},0}, Q'_{\mathsf{col},1}\}$ is statistically hiding and computationally binding.*

As mentioned earlier, the statistically hiding and computationally binding commitments from collapsing functions are known even without using quantum communications [HM96, Unr16]. The above theorem gives an alternative construction for such commitments albeit with quantum communications.

Remark 25 (More constructions)**.** We note that in the original paper discusses more constructions of quantum bit commitments, e.g., from one-way permutations [DMS00], approximable-preimage-size OWFs [KO09], Goldreich-Levin theorem [GL89], and collapsing hash following [HM96]. We exclude them in this thesis for simpler exposition.

8.4 Discussion and Open Problems

We present an efficient preserving conversion theorem for quantum bit commitments based on the equivalence theorem from [AAS20]. More precisely, our result only requires a single call to the base schemes, outperforming the previous conversions that requires at least a polynomial number of calls to the base schemes. Using this compiler and new constructions, we have the following first quantum bit commitments:

- based on PRGs with a single call to the PRGs,
- based on PRSGs with a single call to the PRSGs,
- based on collapsing hashes with a single call to the collapsing hash functions, and
- based on injective OWFs with a shorter commitment lengths.

The following questions are related to our compiler and the constructions of commitment schemes, as well as relations between cryptographic primitives.

1. Is there any compiler for the other notion of bindings, such as collapse-binding or classical binding?

2. Can we construct an efficient quantum bit commitment from any one-way functions? We note that there are some works [KO11, Nao91, HILL99, HR07] that are less efficient, and some conditional lower bounds on the number of call for functions are known in [HHRS15].

3. Constructing quantum string commitments that is much efficient than a simple concatenation of bit commitments is also an intriguing question.

4. Relations between quantum cryptographic primitives such as one-way quantum states, pseudorandom quantum states and unitaries are important questions. We note that some of the prior works [JLS18, AQY21, Kre21, MY21] had explored this direction.

Bibliography

[AA11] Scott Aaronson and Alex Arkhipov. The computational complexity of linear optics. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 333–342, 2011. 3

[AAB+19] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019. 3

[Aar05] Scott Aaronson. Limitations of quantum advice and one-way communication. *Theory of Computing*, 1(1):1–28, 2005. 4, 7, 8, 11, 62, 63, 66, 74, 78

[Aar09] Scott Aaronson. Quantum copy-protection and quantum money. In *2009 24th Annual IEEE Conference on Computational Complexity*, pages 229–242. IEEE, 2009. 2

[Aar21] Scott Aaronson. Open problems related to quantum query complexity. *ACM Transactions on Quantum Computing*, 2(4):1–9, 2021. 78

BIBLIOGRAPHY

[AAS20] Scott Aaronson, Yosi Atia, and Leonard Susskind. On the hardness of detecting macroscopic superpositions. *arXiv preprint arXiv:2009.07450*, 2020. 8, 9, 80, 81, 82, 85, 86, 113, 133, 137

[AC17] Scott Aaronson and Lijie Chen. Complexity-theoretic foundations of quantum supremacy experiments. In *Proceedings of the 32nd Computational Complexity Conference*, pages 1–67, 2017. 3

[ACC+17] Reza Azarderakhsh, Matthew Campagna, Craig Costello, LD Feo, Basil Hess, Amir Jalali, David Jao, Brian Koziel, Brian LaMacchia, Patrick Longa, et al. Supersingular isogeny key encapsulation. *Submission to the NIST Post-Quantum Standardization project*, 152:154–155, 2017. 2

[AGKZ20] Ryan Amos, Marios Georgiou, Aggelos Kiayias, and Mark Zhandry. One-shot signatures and applications to hybrid quantum/classical authentication. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pages 255–268, 2020. 93, 115

[AHU19] Andris Ambainis, Mike Hamburg, and Dominique Unruh. Quantum security proofs using semi-classical oracles. In *Annual International Cryptology Conference*, pages 269–295. Springer, 2019. 7, 15, 16, 17, 18, 43, 60, 81

[Ajt98] Miklós Ajtai. The shortest vector problem in l2 is np-hard for randomized reductions. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 10–19, 1998. 2

[AK07] Scott Aaronson and Greg Kuperberg. Quantum versus classical proofs and advice. In *Twenty-Second Annual IEEE Conference*

on Computational Complexity (CCC'07), pages 115–128. IEEE, 2007. 7, 78

[ALL+21] Scott Aaronson, Jiahui Liu, Qipeng Liu, Mark Zhandry, and Ruizhe Zhang. New approaches for quantum copy-protection. In *Annual International Cryptology Conference*, pages 526–555. Springer, 2021. 2

[AMTDW00] Andris Ambainis, Michele Mosca, Alain Tapp, and Ronald De Wolf. Private quantum channels. In *Proceedings 41st Annual Symposium on Foundations of Computer Science*, pages 547–553. IEEE, 2000. 10

[ANWOW13] Jean-Philippe Aumasson, Samuel Neves, Zooko Wilcox-O'Hearn, and Christian Winnerlein. Blake2: simpler, smaller, fast as md5. In *International Conference on Applied Cryptography and Network Security*, pages 119–135. Springer, 2013. 4

[AQY21] Prabhanjan Ananth, Luowen Qian, and Henry Yuen. Cryptography from pseudorandom quantum states. *Cryptology ePrint Archive*, 2021. 13, 122, 127, 130, 138

[AR19] Scott Aaronson and Guy N Rothblum. Gentle measurement of quantum states and differential privacy. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 322–333, 2019. 7, 63, 66

[ATTU16] Mayuresh Vivekanand Anand, Ehsan Ebrahimi Targhi, Gelo Noel Tabia, and Dominique Unruh. Post-quantum security of the cbc, cfb, ofb, ctr, and xts modes of operation. In *Post-Quantum Cryptography*, pages 44–63. Springer, 2016. 2

[Bab91] László Babai. Local expansion of vertex-transitive graphs and random generation in finite groups. In *Proceedings of the twenty-third annual ACM symposium on Theory of computing*, pages 164–174, 1991. 101

[BB84] Charles H Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and coin tossing. In *Proc. IEEE Int. Conf. Computers, Systems, and Signal Processing, Bangalore, India, 1984*, pages 175–179, 1984. 2

[BB21] Nir Bitansky and Zvika Brakerski. Classical binding for quantum commitments. In *Theory of Cryptography Conference*, pages 273–298. Springer, 2021. 13, 122

[BBBV97] Charles H Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM journal on Computing*, 26(5):1510–1523, 1997. 8, 74

[BBC+01] Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and Ronald De Wolf. Quantum lower bounds by polynomials. *Journal of the ACM (JACM)*, 48(4):778–797, 2001. 15

[BBHT98] Michel Boyer, Gilles Brassard, Peter Høyer, and Alain Tapp. Tight bounds on quantum searching. *Fortschritte der Physik: Progress of Physics*, 46(4-5):493–505, 1998. 20

[BBM17] Daniel J Bernstein, Jean-François Biasse, and Michele Mosca. A low-resource quantum factoring algorithm. In *International Workshop on Post-Quantum Cryptography*, pages 330–346. Springer, 2017. 12

[BCG21] Boaz Barak, Chi-Ning Chou, and Xun Gao. Spoofing linear cross-entropy benchmarking in shallow quantum circuits. In *12th Inno-*

vations in Theoretical Computer Science Conference (ITCS 2021). Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2021. 3

[BCKM21] James Bartusek, Andrea Coladangelo, Dakshita Khurana, and Fermi Ma. One-way functions imply secure computation in a quantum world. In *Annual International Cryptology Conference*, pages 467–496. Springer, 2021. 12, 123

[BCM+21] Zvika Brakerski, Paul Christiano, Urmila Mahadev, Umesh Vazirani, and Thomas Vidick. A cryptographic test of quantumness and certifiable randomness from a single quantum device. *Journal of the ACM (JACM)*, 68(5):1–47, 2021. 3, 9, 107, 108, 109

[BDF+11a] Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random oracles in a quantum world. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 41–69. Springer, 2011. 4, 6, 42

[BDF+11b] Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random oracles in a quantum world. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 41–69. Springer, 2011. 19

[BDPA13] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Keccak. In *Annual international conference on the theory and applications of cryptographic techniques*, pages 313–314. Springer, 2013. 4

[Ber02] Daniel J Bernstein. Pippenger's exponentiation algorithm. 2002. 35

BIBLIOGRAPHY

[Ber09] Daniel J Bernstein. Introduction to post-quantum cryptography. In *Post-quantum cryptography*, pages 1–14. Springer, 2009. 1

[BGMW92] Ernest F Brickell, Daniel M Gordon, Kevin S McCurley, and David B Wilson. Fast exponentiation with precomputation. In *Workshop on the Theory and Application of of Cryptographic Techniques*, pages 200–207. Springer, 1992. 35, 36

[BHH+19] Nina Bindel, Mike Hamburg, Kathrin Hövelmanns, Andreas Hülsing, and Edoardo Persichetti. Tighter proofs of cca security in the quantum random oracle model. In *Theory of Cryptography Conference*, pages 61–90. Springer, 2019. 7, 18, 44, 60

[BHMT02] Gilles Brassard, Peter Hoyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. *Quantum Computation and Quantum Information*, 305:53–74, 2002. 7, 20, 50

[BHY09] Mihir Bellare, Dennis Hofheinz, and Scott Yilek. Possibility and impossibility results for encryption and commitment secure under selective opening. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 1–35. Springer, 2009. 106

[BJ15] Anne Broadbent and Stacey Jeffery. Quantum homomorphic encryption for circuits of low t-gate complexity. In *Annual Cryptology Conference*, pages 609–629. Springer, 2015. 10, 95

[BKVV20] Zvika Brakerski, Venkata Koppula, Umesh Vazirani, and Thomas Vidick. Simpler proofs of quantumness. In *15th Conference on the Theory of Quantum Computation, Communication and Cryptography*, 2020. 3, 9, 109

BIBLIOGRAPHY

[BL13] Daniel J Bernstein and Tanja Lange. Non-uniform cracks in the concrete: the power of free precomputation. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 321–340. Springer, 2013. 39

[BL20] Anne Broadbent and Sébastien Lord. Uncloneable quantum encryption via oracles. In *15th Conference on the Theory of Quantum Computation, Communication and Cryptography*, 2020. 2

[Boo02] Jonathan Bootle. Efficient multi-exponentiation. 2002. 36

[BR93] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *Proceedings of the 1st ACM Conference on Computer and Communications Security*, pages 62–73, 1993. 6, 19, 42

[BSS22] Xavier Bonnetain, André Schrottenloher, and Ferdinand Sibleyras. Beyond quadratic speedups in quantum attacks on symmetric schemes. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 315–344. Springer, 2022. 2

[BV97] Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. *SIAM Journal on computing*, 26(5):1411–1473, 1997. 21

[BZ13] Dan Boneh and Mark Zhandry. Secure signatures and chosen ciphertext security in a quantum computing world. In *Annual cryptology conference*, pages 361–379. Springer, 2013. 6

[CCHL22] Sitan Chen, Jordan Cotler, Hsin-Yuan Huang, and Jerry Li. Exponential separations between learning with and without quantum memory. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 574–585. IEEE, 2022. 78

[CCY20] Nai-Hui Chia, Kai-Min Chung, and Takashi Yamakawa. Classical verification of quantum computations with efficient verifier. In *Theory of Cryptography Conference*, pages 181–206. Springer, 2020. 3

[CDGS18] Sandro Coretti, Yevgeniy Dodis, Siyao Guo, and John Steinberger. Random oracles and non-uniformity. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 227–258. Springer, 2018. 6, 12

[CDMS04] Claude Crépeau, Paul Dumais, Dominic Mayers, and Louis Salvail. Computational collapse of quantum state with application to oblivious transfer. In *Theory of Cryptography Conference*, pages 374–393. Springer, 2004. 13, 123

[CGK18] Henry Corrigan-Gibbs and Dmitry Kogan. The discrete-logarithm problem with preprocessing. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 415–447. Springer, 2018. 5, 26, 61, 63, 77

[CGLQ20] Kai-Min Chung, Siyao Guo, Qipeng Liu, and Luowen Qian. Tight quantum time-space tradeoffs for function inversion. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 673–684. IEEE, 2020. 11, 40, 44, 57, 60, 61, 63, 77

[CLQ20] Kai-Min Chung, Tai-Ning Liao, and Luowen Qian. Lower bounds for function inversion with quantum advice. In *1st Conference on Information-Theoretic Cryptography (ITC 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020. 11, 44, 49, 60, 63, 77

BIBLIOGRAPHY

[CLS01] Claude Crépeau, Frédéric Légaré, and Louis Salvail. How to convert the flavor of a quantum bit commitment. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 60–77. Springer, 2001. 8, 10, 11, 120, 127, 130

[CLZ22] Yilei Chen, Qipeng Liu, and Mark Zhandry. Quantum algorithms for variants of average-case lattice problems via filtering. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 372–401. Springer, 2022. 2

[CX21] Shujiao Cao and Rui Xue. Being a permutation is also orthogonal to one-wayness in quantum world: Impossibilities of quantum one-way permutations from one-wayness primitives. *Theoretical Computer Science*, 855:16–42, 2021. 12

[CX22] Shujiao Cao and Rui Xue. The gap is sensitive to size of preimages: Collapsing property doesn't go beyond quantum collision-resistance for preimages bounded hash functions. *Cryptology ePrint Archive (to appear in Crypto'2022)*, 2022. 93

[D+15] Morris J Dworkin et al. Sha-3 standard: Permutation-based hash and extendable-output functions. 2015. 4, 19

[DFS04] Ivan Damgård, Serge Fehr, and Louis Salvail. Zero-knowledge proofs and string commitments withstanding quantum attacks. In *Annual International Cryptology Conference*, pages 254–272. Springer, 2004. 13, 123

[DFSS08] Ivan B Damgård, Serge Fehr, Louis Salvail, and Christian Schaffner. Cryptography in the bounded-quantum-storage model. *SIAM Journal on Computing*, 37(6):1865–1890, 2008. 123

[DGK17] Yevgeniy Dodis, Siyao Guo, and Jonathan Katz. Fixing cracks in the concrete: Random oracles with auxiliary input, revisited. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 473–495. Springer, 2017. 6, 7, 19, 47, 48, 59

[DKRS21] Orr Dunkelman, Nathan Keller, Eyal Ronen, and Adi Shamir. Quantum time/memory/data tradeoff attacks. *Cryptology ePrint Archive*, 2021. 5, 12, 25, 27, 31, 42

[DLS81] Peter Downey, Benton Leong, and Ravi Sethi. Computing sequences with addition chains. *SIAM Journal on Computing*, 10(3):638–646, 1981. 35

[DMS00] Paul Dumais, Dominic Mayers, and Louis Salvail. Perfectly concealing quantum bit commitment from any quantum one-way permutation. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 300–315. Springer, 2000. 130, 132, 137

[DR02] Joan Daemen and Vincent Rijmen. *The design of Rijndael*, volume 2. Springer, 2002. 4

[DS22] Marcel Dall'Agnol and Nicholas Spooner. On the necessity of collapsing. *Cryptology ePrint Archive*, 2022. 93

[DTT10] Anindya De, Luca Trevisan, and Madhur Tulsiani. Time space tradeoffs for attacks against one-way functions and prgs. In *Annual Cryptology Conference*, pages 649–665. Springer, 2010. 5, 7, 12, 26, 27, 31, 32, 33, 34, 40, 47, 56, 57

[EH17] Martin Ekerå and Johan Håstad. Quantum algorithms for computing short discrete logarithms and factoring rsa integers. In *Interna-*

tional Workshop on Post-Quantum Cryptography, pages 347–363. Springer, 2017. 6, 12

[Eke20] Martin Ekerå. On post-processing in the quantum algorithm for computing short discrete logarithms. *Designs, Codes and Cryptography*, 88(11):2313–2335, 2020. 12

[Eke21] Martin Ekerå. Quantum algorithms for computing general discrete logarithms and orders with tradeoffs. *Journal of Mathematical Cryptology*, 15(1):359–407, 2021. 6, 12

[Fis02] Marc Fischlin. On the impossibility of constructing non-interactive statistically-secret protocols from any trapdoor one-way function. In *Cryptographers' Track at the RSA Conference*, pages 79–95. Springer, 2002. 120

[FK18] Bill Fefferman and Shelby Kimmel. Quantum vs. classical proofs and subset verification. In *43rd International Symposium on Mathematical Foundations of Computer Science*, page 1, 2018. 78

[FN00] Amos Fiat and Moni Naor. Rigorous time/space trade-offs for inverting functions. *SIAM Journal on Computing*, 29(3):790–803, 2000. 5, 12, 26, 27, 31, 32

[FUYZ20] Junbin Fang, Dominique Unruh, Jun Yan, and Dehua Zhou. How to base security on the perfect/statistical binding property of quantum bit commitment? *Cryptology ePrint Archive*, 2020. 12, 13, 122, 123

[GE21] Craig Gidney and Martin Ekerå. How to factor 2048 bit rsa integers in 8 hours using 20 million noisy qubits. *Quantum*, 5:433, 2021. 6, 12, 26, 35, 39

[GGH+20] Alexander Golovnev, Siyao Guo, Thibaut Horel, Sunoo Park, and Vinod Vaikuntanathan. Data structures meet cryptography: 3sum with preprocessing. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pages 294–307, 2020. 5, 26

[GGKL21] Nick Gravin, Siyao Guo, Tsz Chiu Kwok, and Pinyan Lu. Concentration bounds for almost k-wise independence with applications to non-uniform security. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2404–2423. SIAM, 2021. 60, 61

[GGKT05] Rosario Gennaro, Yael Gertner, Jonathan Katz, and Luca Trevisan. Bounds on the efficiency of generic cryptographic constructions. *SIAM journal on Computing*, 35(1):217–246, 2005. 7, 43, 45

[GKZ19] Alex B Grilo, Iordanis Kerenidis, and Timo Zijlstra. Learning-with-errors problem is easy with quantum samples. *Physical Review A*, 99(3):032314, 2019. 2

[GL89] Oded Goldreich and Leonid A Levin. A hard-core predicate for all one-way functions. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 25–32, 1989. 80, 132, 133, 137

[GLLZ21] Siyao Guo, Qian Li, Qipeng Liu, and Jiapeng Zhang. Unifying presampling via concentration bounds. In *Theory of Cryptography Conference*, pages 177–208. Springer, 2021. 12, 44, 60

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceed-*

ings of the fortieth annual ACM symposium on Theory of computing, pages 197–206, 2008. 107, 109

[Gro96] Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 212–219. ACM, 1996. 1

[GRS00] Oded Goldreich, Ronitt Rubinfeld, and Madhu Sudan. Learning polynomials with queries: The highly noisy case. *SIAM Journal on Discrete Mathematics*, 13(4):535–570, 2000. 133

[GT00] Rosario Gennaro and Luca Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. In *Proceedings 41st Annual Symposium on Foundations of Computer Science*, pages 305–313. IEEE, 2000. 7, 43, 45

[Hel80] Martin E. Hellman. A cryptanalytic time-memory trade-off. *IEEE Trans. Information Theory*, 26(4):401–406, 1980. 4, 5, 12, 25, 26, 27, 61, 77

[Hen10] Ryan Henry. Pippenger's multiproduct and multiexponentiation algorithms. *Extended Ver*, 2010. 35, 36

[Hha22] Minki Hhan. A quantum time-memory trade-off for inverting any function (working title), 2022. 13

[HHRS15] Iftach Haitner, Jonathan J Hoch, Omer Reingold, and Gil Segev. Finding collisions in interactive protocols—tight lower bounds on the round and communication complexities of statistically hiding commitments. *SIAM Journal on Computing*, 44(1):193–242, 2015. 11, 120, 138

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999. 10, 127, 136, 138

[HLG21] Shuichi Hirahara and François Le Gall. Test of quantumness with small-depth quantum circuits. In *46th International Symposium on Mathematical Foundations of Computer Science (MFCS 2021)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2021. 3

[HM96] Shai Halevi and Silvio Micali. Practical and provably-secure commitment schemes from collision-free hashing. In *Annual International Cryptology Conference*, pages 201–215. Springer, 1996. 11, 136, 137

[HMS04] Thomas Holenstein, Ueli Maurer, and Johan Sjödin. Complete classification of bilinear hard-core functions. In *Annual International Cryptology Conference*, pages 73–91. Springer, 2004. 133

[HMY22] Minki Hhan, Tomoyuki Morimae, and Takashi Yamakawa. From the Hardness of Detecting Superpositions to Cryptography: Quantum Public Key Encryption and Commitments (working title), 2022. 13

[Hol73] Alexander Semenovich Holevo. Bounds for the quantity of information transmitted by a quantum communication channel. *Problemy Peredachi Informatsii*, 9(3):3–11, 1973. 63, 64

[HR07] Iftach Haitner and Omer Reingold. Statistically-hiding commitment from any one-way function. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 1–10, 2007. 10, 138

[HRV13] Iftach Haitner, Omer Reingold, and Salil Vadhan. Efficiency improvements in constructing pseudorandom generators from one-way functions. *SIAM Journal on Computing*, 42(3):1405–1430, 2013. 127, 136

[HXY19] Minki Hhan, Keita Xagawa, and Takashi Yamakawa. Quantum random oracle model with auxiliary input. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 584–614. Springer, 2019. 13, 43, 44, 49, 57, 60, 63

[HY20] Akinori Hosoyamada and Takashi Yamakawa. Finding collisions in a quantum world: quantum black-box separation of collision-resistance and one-wayness. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 3–32. Springer, 2020. 12

[HY22] Minki Hhan and Aaram Yun. On Quantum Multiple Discrete Logarithm Problem (working title), 2022. 13

[JF11] David Jao and Luca De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In *International Workshop on Post-Quantum Cryptography*, pages 19–34. Springer, 2011. 2

[JLS18] Zhengfeng Ji, Yi-Kai Liu, and Fang Song. Pseudorandom quantum states. In *Annual International Cryptology Conference*, pages 126–152. Springer, 2018. 23, 127, 138

[JNV+21] Zhengfeng Ji, Anand Natarajan, Thomas Vidick, John Wright, and Henry Yuen. MIP*= RE. *Communications of the ACM*, 64(11):131–138, 2021. 3

[JQSY19] Zhengfeng Ji, Youming Qiao, Fang Song, and Aaram Yun. General linear group action on tensors: A candidate for post-quantum cryptography. In *Theory of Cryptography Conference*, pages 251–281. Springer, 2019. 8, 9, 89, 100, 102, 103, 113

[KKNY05] Akinori Kawachi, Takeshi Koshiba, Harumichi Nishimura, and Tomoyuki Yamakami. Computational indistinguishability between quantum states and its cryptographic application. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 268–284. Springer, 2005. 12

[KLLNP16] Marc Kaplan, Gaëtan Leurent, Anthony Leverrier, and María Naya-Plasencia. Breaking symmetric cryptosystems using quantum period finding. In *Annual international cryptology conference*, pages 207–237. Springer, 2016. 2

[KNP07] Pascal Koiran, Vincent Nesme, and Natacha Portier. The quantum query complexity of the abelian hidden subgroup problem. *Theoretical computer science*, 380(1-2):115–126, 2007. 40

[KO09] Takeshi Koshiba and Takanori Odaira. Statistically-hiding quantum bit commitment from approximable-preimage-size quantum one-way function. In *Workshop on Quantum Computation, Communication, and Cryptography*, pages 33–46. Springer, 2009. 132, 137

[KO11] Takeshi Koshiba and Takanori Odaira. Non-interactive statistically-hiding quantum bit commitment from any quantum one-way function. *arXiv preprint arXiv:1102.3441*, 2011. 127, 132, 138

[Kre21] William Kretschmer. Quantum pseudorandomness and classical complexity. In *16th Conference on the Theory of Quantum Computation, Communication and Cryptography*, 2021. 23, 138

[KS01] Fabian Kuhn and René Struik. Random walks revisited: Extensions of pollard's rho algorithm for computing multiple discrete logarithms. In *International Workshop on Selected Areas in Cryptography*, pages 212–229. Springer, 2001. 5, 26, 35

[Kup05] Greg Kuperberg. A subexponential-time quantum algorithm for the dihedral hidden subgroup problem. *SIAM Journal on Computing*, 35(1):170–188, 2005. 10, 111

[LC97] Hoi-Kwong Lo and Hoi Fung Chau. Is quantum bit commitment really possible? *Physical Review Letters*, 78(17):3410, 1997. 10, 119, 123

[LG21] Zhenning Liu and Alexandru Gheorghiu. Depth-efficient proofs of quantumness. *arXiv preprint arXiv:2107.02163*, 2021. 3

[LL94] Chae Hoon Lim and Pil Joong Lee. More flexible exponentiation with precomputation. In *Annual International Cryptology Conference*, pages 95–107. Springer, 1994. 5, 35, 36

[LLL+21] Yong Liu, Xin Liu, Fang Li, Haohuan Fu, Yuling Yang, Jiawei Song, Pengpeng Zhao, Zhen Wang, Dajia Peng, Huarong Chen, et al. Closing the" quantum supremacy" gap: achieving real-time simulation of a random quantum circuit using a new sunway supercomputer. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–12, 2021. 3

[Mah18] Urmila Mahadev. Classical verification of quantum computations. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 259–267. IEEE, 2018. 3

[May97] Dominic Mayers. Unconditionally secure quantum bit commitment is impossible. *Physical review letters*, 78(17):3414, 1997. 10, 119, 123

[Mic01] Daniele Micciancio. The shortest vector in a lattice is hard to approximate to within some constant. *SIAM journal on Computing*, 30(6):2008–2035, 2001. 2

[MT79] Robert Morris and Ken Thompson. Password security: A case history. *Communications of the ACM*, 22(11):594–597, 1979. 7

[MY21] Tomoyuki Morimae and Takashi Yamakawa. Quantum commitments and signatures without one-way functions. *Cryptology ePrint Archive (to appear in Crypto'2022)*, 2021. 12, 23, 122, 123, 127, 128, 129, 138

[NABT15] Aran Nayebi, Scott Aaronson, Aleksandrs Belovs, and Luca Trevisan. Quantum lower bound for inverting a permutation with advice. *Quantum Information & Computation*, 15(11-12):901–913, 2015. 4, 7, 49, 56, 62

[Nao91] Moni Naor. Bit commitment using pseudorandomness. *Journal of cryptology*, 4(2):151–158, 1991. 10, 126, 136, 138

[Nay99] Ashwin Nayak. Optimal lower bounds for quantum automata and random access codes. In *40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039)*, pages 369–376. IEEE, 1999. 7, 63, 65

[NC00] Michael A Nielsen and Isaac L Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000. 15

[NIS] NIST. Post-Quantum Cryptography Standardization. `https://bit.ly/3lfzIub`. Accessed: 2022-06-17. 2

[NS06] Ashwin Nayak and Julia Salzman. Limits on the ability of quantum states to convey classical messages. *Journal of the ACM (JACM)*, 53(1):184–206, 2006. 7, 63, 65

[NY04] Harumichi Nishimura and Tomoyuki Yamakami. Polynomial time quantum computation with advice. *Information Processing Letters*, 90(4):195–204, 2004. 7, 62, 74

[Oli81] Jorge Olivos. On vectorial addition chains. *Journal of Algorithms*, 2(1):13–21, 1981. 35

[OTU00] Tatsuaki Okamoto, Keisuke Tanaka, and Shigenori Uchiyama. Quantum public-key cryptosystems. In *Annual international cryptology conference*, pages 147–165. Springer, 2000. 12

[PC09] Ray A Perlner and David A Cooper. Quantum resistant public key cryptography: a survey. In *Proceedings of the 8th Symposium on Identity and Trust on the Internet*, pages 85–93, 2009. 1

[Pip80] Nicholas Pippenger. On the evaluation of powers and monomials. *SIAM Journal on Computing*, 9(2):230–250, 1980. 5, 35

[Reg09] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):1–40, 2009. 10, 107

[S+99] Data Encryption Standard et al. Data encryption standard. *Federal Information Processing Standards Publication*, 112, 1999. 4

[Sal98] Louis Salvail. Quantum bit commitment from a physical assumption. In *Annual International Cryptology Conference*, pages 338–353. Springer, 1998. 123

[Sho99] Peter W Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review*, 41(2):303–332, 1999. 1, 5, 21

[Sim97] Daniel R Simon. On the power of quantum computation. *SIAM journal on computing*, 26(5):1474–1483, 1997. 21

[SSV13] John A Smolin, Graeme Smith, and Alexander Vargo. Oversimplifying quantum factoring. *Nature*, 499(7457):163–165, 2013. 12

[Unr07] Dominique Unruh. Random oracles and auxiliary input. In *Annual International Cryptology Conference*, pages 205–223. Springer, 2007. 4, 6, 12, 19

[Unr12] Dominique Unruh. Quantum proofs of knowledge. In *Annual international conference on the theory and applications of cryptographic techniques*, pages 135–152. Springer, 2012. 133

[Unr15] Dominique Unruh. Revocable quantum timed-release encryption. *Journal of the ACM (JACM)*, 62(6):1–76, 2015. 7, 81

[Unr16] Dominique Unruh. Computationally binding quantum commitments. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 497–527. Springer, 2016. 13, 22, 123, 136, 137

[VSB+01] Lieven MK Vandersypen, Matthias Steffen, Gregory Breyta, Costantino S Yannoni, Mark H Sherwood, and Isaac L Chuang. Experimental realization of shor's quantum factoring algorithm

using nuclear magnetic resonance. *Nature*, 414(6866):883–887, 2001. 12

[VZ12] Salil Vadhan and Colin Jia Zheng. Characterizing pseudoentropy and simplifying pseudorandom generator constructions. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 817–836, 2012. 127, 136

[Wat00] John Watrous. Succinct quantum proofs for properties of finite groups. In *Proceedings 41st Annual Symposium on Foundations of Computer Science*, pages 537–546. IEEE, 2000. 7, 102

[Wie83] Stephen Wiesner. Conjugate coding. *ACM Sigact News*, 15(1):78–88, 1983. 2

[Win99] Andreas Winter. Coding theorem and strong converse for quantum channels. *IEEE Transactions on Information theory*, 45(7):2481–2485, 1999. 7, 63, 66

[Yan20] Jun Yan. General properties of quantum bit commitments. Cryptology ePrint Archive, Report 2020/1488, 2020. `https://ia.cr/2020/1488`. 8, 10, 11, 13, 120, 121, 122, 123, 126, 127, 130

[Yan21] Jun Yan. Quantum computationally predicate-binding commitments with application in quantum zero-knowledge arguments for np. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 575–605. Springer, 2021. 123

[Yao76] Andrew Chi-Chih Yao. On the evaluation of powers. *SIAM Journal on computing*, 5(1):100–103, 1976. 35

[Yao82] Andrew C Yao. Theory and application of trapdoor functions. In *23rd Annual Symposium on Foundations of Computer Science (SFCS 1982)*, pages 80–91. IEEE, 1982. 58

[Yao90] AC-C Yao. Coherent functions and program checkers. In *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pages 84–94, 1990. 56

[Yun15] Aaram Yun. Generic hardness of the multiple discrete logarithm problem. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 817–836. Springer, 2015. 5, 35

[YWLQ15] Jun Yan, Jian Weng, Dongdai Lin, and Yujuan Quan. Quantum bit commitment with application in quantum zero-knowledge proof. In *International Symposium on Algorithms and Computation*, pages 555–565. Springer, 2015. 12, 122, 126

[YZ21] Takashi Yamakawa and Mark Zhandry. Classical vs quantum random oracles. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 568–597. Springer, 2021. 3

[YZ22] Takashi Yamakawa and Mark Zhandry. Verifiable quantum advantage without structure. *arXiv preprint arXiv:2204.02063 (to appear in FOCS'2022)*, 2022. 3

[Zha19] Mark Zhandry. How to record quantum queries, and applications to quantum indifferentiability. In *Annual International Cryptology Conference*, pages 239–268. Springer, 2019. 61

[Zha21] Mark Zhandry. Quantum lightning never strikes the same state twice. or: quantum money from cryptographic assumptions. *Journal of Cryptology*, 34(1):1–56, 2021. 2, 93

[Zha22] Mark Zhandry. New constructions of collapsing hashes. *Cryptology ePrint Archive (to appear in Crypto'2022)*, 2022. 93

[ZWD+20] Han-Sen Zhong, Hui Wang, Yu-Hao Deng, Ming-Cheng Chen, Li-Chao Peng, Yi-Han Luo, Jian Qin, Dian Wu, Xing Ding, Yi Hu, et al. Quantum computational advantage using photons. *Science*, 370(6523):1460–1463, 2020. 3

국문초록

양자역학을 이용한 컴퓨터의 등장은 쇼어의 알고리즘 등을 통해 기존 암호학에 명백한 위협을 제시하며, 양자역학의 성질을 통한 새로운 암호프로토콜의 가능성 또한 제시한다. 이러한 두 가지 관점은 각각 이 학위 논문의 주제가 되는 양자공격에 대한 대응책으로써의 대양자암호와 양자역학을 이용한 암호기술인 양자암호라고 불리는 새로운 분야를 발생시켰다.

이 학위 논문에서는 현재 대양자암호의 안전성을 새로운 양자암호 공격 알고리즘과 모델, 안전성 증명을 통해 재고한다. 특히 일방향함수, 암호학적 난수생성기 등의 프로토콜의 대양자 암호 안전성의 구체적인 평가를 제시한다. 또한 최근 양자역학의 연구를 양자암호에 도입함으로써 새로운 양자 공개키암호와 양자 커밋먼트 등의 새로운 발견을 제시한다. 이 과정에서 전처리 계산을 포함한 양자알고리즘의 한계, 양자 복잡계들의 오라클분리 문제, 군의 작용을 이용한 공개키 암호 등의 여러 열린문제들의 해결을 제시한다.

주요어휘: 양자컴퓨터, 암호학, 양자알고리즘, 난수오라클모델, 공개키암호, 비트커밋먼트

학번: 2016-20255