

Sorbonne Université

École Doctorale Informatique, Télécommunications et Electronique (ED130)

Laboratoire LIP6

Cryptographic Primitives in Quantum Idealized Models

Par Samuel Bouaziz--Ermann

Thèse de doctorat d'informatique

Dirigée par Damien Vergnaud et Alex Bredariol Grilo

Présentée et soutenue publiquement le 15 septembre 2025

Devant un jury composé de :

Gorjan Alagic	Associate Research Scientist, University of Maryland (rapporteur)
Alex Bredariol Grilo	Chercheur CNRS, Sorbonne Université, LIP6
Claude Crépeau	Professeur, École de technologie supérieure
Tomoyuki Morimae	Associate Professor, Kyoto University (rapporteur)
Adeline Roux-Langlois	Chercheuse CNRS, Université de Caen (présidente)
Damien Vergnaud	Professeur, Sorbonne Université, LIP6

À Sid et Versa.

Remerciements

Faire cette thèse n'aurait pas été possible sans un bon entourage, et il y a beaucoup de personnes que je tiens à remercier. Je vais essayer de faire court et de n'oublier personne !

Tout d'abord, Alex et Damien, merci pour m'avoir encadré pendant ces quatre années de thèse. Vous avez su me guider tout au long de ma thèse, et particulièrement au début qui était le plus difficile. Vous m'avez aidé sur le plan scientifique bien sûr, mais vous avez aussi su me conseiller sur bien d'autres aspects et pour ça je vous en remercie. J'ai maintenant le sentiment que je suis prêt à devenir un chercheur, et c'est en grande partie grâce à vous !

Thanks to Gorjan and Tomoyuki for agreeing to review my manuscript, and Claude and Adeline for being members of the jury. Un grand merci aussi à André et Sophie pour avoir accepté d'être dans mon comité de suivi !

I would also like to thanks all the person with whom I collaborated during my PhD, namely Garazi, Huy and Minki, but also all the others persons with whom the projects ended up not working out.

J'aimerais aussi remercier toutes les personnes que j'ai rencontrées dans le cadre de ma thèse à Almasty, Charles et son humour... questionnable, Ambroise, toujours chaud pour aller grimper, Florette, qui a su répondre à toutes mes questions administratives au début de ma thèse, Thomas, avec qui on a eu beaucoup trop de discussions esportives, Mickaël, avec qui j'ai partagé bieeen plus d'une bière, Ahmed, silencieusement tapis au fond du bureau 413 et Gaspard, le stagiaire. Il y aussi ceux qu'on voit rarement mais dont la présence est toujours appréciée : Thibauld le cryptoexpert, Andersson et Abdel dont je ne pourrais jamais égaler le pot de thèse, et Lucas qui profite du soleil à Montpellier. Jules à faire: trouver quelque chose de positif à dire. Julia tu es vraiment une collègue super collègue, enfin ça fait longtemps qu'on a dépassé ce stade maintenant. Merci pour les soirées passées à jouer à Stardew Valley, à Blue Prince et à regarder Brooklyn 99; je t'aime fort. Merci à tous les autres qui sont passés par le bureau 413, merci pour tous les *afterwork* aux rattrapages, au 2bis, sur les quais...

I would also like to thank all the persons that I met in the QI team and because of the amount of persons that this amount to, I prefer not to make a list for fear of missing someone. Despite technically not being a member of the team, I have always felt included and I want to thank you very much for that. Thank you also for the amazing retreats!

J'aimerais remercier les O.G. d'Edgar Poe avec qui j'ai gardé contact: Martin, Steven et Alexis d'abord mais aussi Paul et Ivan ! On se voit moins fréquemment maintenant mais c'est toujours un plaisir de vous capter !

Je veux aussi remercier les "copains de thèse", qui font une thèse en même temps que moi

mais pas dans le même labo. D'abord Paul, merci pour la motivation et les conseils en matière de sport, qui a eu un impact non-negligeable dans le déroulement de ma thèse. Julien, ta capacité à systématiquement tout improviser au dernier moment continue de toujours m'impressionner, mais après tout ça reste une question d'optimisation. Tom, on se connait depuis si longtemps maintenant et j'aimerais te remercier pour les conseils en tout genre, que ce soit en cuisine, en mode, en math, ... On a parcouru un long chemin depuis les épisodes de L'Attaque des Titans qu'on regardait en prépa, pendant la pause de midi. Vous trois, j'aimerais aussi vous remercier pour les parties de LOL et les soirées jeux vidéos en général, mais aussi toutes les vacances et les week-ends qu'on a passé ensemble. Maintenant qu'on a tous soutenu et que trois d'entre nous quittent la France, ça sera beaucoup plus compliqué de se voir mais j'espère qu'on est encore loin de la fin de notre amitié ! J'aimerais aussi remercier Nicolas et Rémi qui nous accompagnent la plupart du temps dans nos aventures ! J'accorde enfin une mention spéciale à Aaron pour les longues discussions qu'on a eu sur discord pendant nos thèses.

Léo, merci pour les problèmes d'échec et les discussions foot, et plus récemment les parties de CS2 même si notre niveau laisse à désirer. J'ai l'impression que tu m'accompagnes depuis la prépa alors qu'on a pas été dans la même salle de classe depuis le collège ! Je vous souhaite énormément de bonheur pour le futur avec Jeanne !

Enfin, Papa, Maman. Vous me faites confiance depuis longtemps et je vous en remercie. Vous m'accompagnez depuis le début et votre soutien inconditionnel m'a été d'une grande aide ! Je vous aime, merci pour tout. Elia, merci pour tes conseils sur à peu près tous les jeux vidéos auquel j'ai joué, tu es sans aucun doute le plus grand gamer que je connaisse. Ilan, parfois on ne discute pas pendant des mois, mais je sais que tu seras toujours là pour moi et c'est bien évidemment réciproque. Merci pour toutes les soirées Dofus, avec Yohann, que j'embrasse également ! Parfois je me dis qu'il faudrait vraiment qu'on trouve un autre jeu, parce que quand je le relance c'est pas pour jouer mon panda ou mon enu mais juste une excuse pour passer une soirée avec vous !

Finalement, merci Jaja et Nikita pour la présence chaleureuse pendant les deux dernières années de ma thèse.

Abstract

In this thesis, we study both classical and quantum cryptography within idealized quantum models. Previous work has shown that quantum resources can be used to construct cryptographic tasks that are proven or conjectured to be impossible in the classical setting. Here, we first prove lower bounds on the efficiency of any quantum algorithm that finds a subset-cover of a random function, a problem that has been conjectured to be hard for assessing the security of the post-quantum digital signature scheme SPHINCS+. Next, we extend existing impossibility results for constructing public-key encryption schemes in the quantum random oracle model by showing that a more general type of public-key encryption does not exist in this model. We then study quantum assumptions for cryptography that appear weaker than one-way functions, namely quantum pseudorandomness, and its relationship to quantum public key encryption and signature schemes, both clarifying and improving upon prior constructions and impossibility proofs. Finally, we establish the importance of the size of pseudorandomness by proving that quantum pseudorandomness cannot be shrunk, and we make progress toward showing that it cannot be amplified.

Contents

List of Abbreviations

1PRS one copy pseudorandom quantum states.

CC1QM-KA classical communication one quantum message key agreement.

CHFS common Haar function-like state.

EFI efficiently samplable, statistically far but computationally indistinguishable pairs of quantum states.

EV-OWP efficiently verifiable one-way puzzle.

EV-OWSP efficiently verifiable one-way state puzzles.

MPC multi-party computation.

NIST National Institute of Standards and Technology.

OT oblivious transfer.

OWF one-way function.

OWP one-way puzzle.

OWSG one-way state generator.

OWSP one-way state puzzles.

PCC polynomial compatibility conjecture.

PD-PRG pseudodeterministic pseudorandom generators.

PKE public key encryption.

PRF pseudorandom function.

PRFS pseudorandom function state generator.

PRG pseudorandom generator.

PRP pseudorandom permutation.

PRS pseudorandom quantum state generator.

PRU pseudorandom unitary.

Q-COM quantum commitment.

QCCC quantum computation classical communication.

QKD quantum key distribution.

qPKE quantum public key encryption.

QPRG quantum pseudorandom generator.

QPT quantum polynomial time.

QROM quantum random oracle model.

ROM random oracle model.

RSC restricted subset cover.

SC subset cover.

short-PRS short-output pseudorandom quantum states.

Chapter 1

Introduction

1.1 Introduction

In 1983, Wiesner [Wie83] showed a way to construct money that is provably impossible to counterfeit, which is not possible without the use of quantum resources. A year later, Bennett and Brassard [BB84] proposed an information-theoretically secure quantum key distribution protocol, which cannot exist in the classical setting. These results have shown that previously established results in classical cryptography do not hold in the quantum setting and that quantum resources can be used to achieve tasks that were previously impossible. Along these lines, in 1994, Shor [Sho97] described a quantum algorithm that can efficiently factorize numbers, a task for which no efficient classical algorithm is known. Unfortunately for classical cryptography, the security of the RSA cryptographic system [RSA78] relies on the hardness of this problem, which has resisted classical attacks for over 50 years. Although quantum computers are not yet powerful enough to execute Shor's algorithm in practice, this threat has been taken seriously by cryptographers: the field of post-quantum cryptography focuses on designing cryptographic systems with classical computers that are resistant to quantum attacks. This field is crucial for the near future, where powerful institutions such as Google [AAB+19], Microsoft [AARA+25] or Quandela [MFP+24], may possess quantum resources that are not available to the general public. With this goal in mind, the National Institute of Standards and Technology (NIST) has launched several competitions to identify post-quantum cryptographic schemes to replace the currently used ones that are vulnerable to quantum computers.

Parallel to these works, the field of quantum cryptography aims to study schemes where the parties and the communication can be quantum. Ji, Liu, and Song [JLS18] proposed the first two inherently quantum pseudorandom primitives, pseudorandom state generators (PRSs) and pseudorandom unitaries (PRUs). These primitives are collections of states

(respectively, unitaries) that can be efficiently generated, but are indistinguishable from truly random states (respectively, unitaries). Quantum pseudorandomness has been shown to be useful for constructing various quantum cryptographic primitives such as quantum commitments and oblivious transfers [MY22b, AQY22]. In the classical setting, one-way functions — functions that are easy to compute, but hard to invert — are considered the minimal assumption for cryptography, meaning that most cryptographic schemes require one-way functions, and there is nothing relevant that is weaker. In 2021, Kretschmer [Kre21] showed that PRSs and PRUs are potentially weaker primitives than one-way functions, which sparked dramatic interest in fundamentally quantum cryptographic primitives.

In this thesis, we study classical and quantum cryptography in idealized quantum models, where we establish bounds on the efficiency of quantum algorithms for solving certain tasks that are related to post-quantum and quantum cryptography. We also demonstrate the impossibility of specific constructions in these models, building on a series of recent papers in the literature that aim to identify the minimal assumptions for building quantum cryptography.

1.1.1 Sphincs+

In 2017, the NIST opened a call for proposals of post-quantum signatures and public key encryption schemes. In 2024, three signature schemes were selected after the final round of the competition to be standardized: CRYSTALS-Dilithium [LDK+22], Falcon [PFH+22] and SPHINCS+ [HBD+22]. Of particular interest to this thesis, SPHINCS+ is a hash-based signature scheme, i.e. a signature whose security relies solely on the existence of an idealized hash function. Hash functions are a central tool in cryptography. They are functions that compress their input and have the additional property that finding a *collision*, that is, two distinct inputs that map to the same output, should be computationally hard. Quantum computers have been shown to have a quadratic speedup in finding collisions [Gro97] compared to classical computers, which makes them more efficient than classical computers, but they still require exponential time. That is the main reason why such functions are interesting for constructing post-quantum cryptographic primitives.

There have been constructions of digital signatures from hash functions [Lam79, Rom90] proposed in the classical setting, and the fact that quantum computers are not significantly better at breaking the security of hash functions makes the study of such constructions very compelling. Unlike CRYSTALS-Dilithium and Falcon, the construction of SPHINCS+ does not fundamentally rely on computational assumptions, and thus could be proven to be secure under the assumption that the hash function is secure. Moreover, there exist constructions of signatures from hash functions that have been proven to be secure against quantum adversaries [BDF+11]. However, at the time of writing, there is no formal proof that SPHINCS+ is secure against quantum computers. In their submission, the authors provide intuitions about why their scheme should be secure, by using arguments similar to the ones above. Also, in order to make the schemes more efficient, cryptographers add additional structure to their constructions. By having more structure, the schemes

become less secure, and in the case of SPHINCS+, this means that finding a collision in the hash function is not necessary to break the security of the signature. It is hard to define informally what is required to break the security of SPHINCS+; in fact, it is even harder to define this formally, as the complexity of the construction of SPHINCS+ makes it difficult to analyze it, especially against quantum adversaries.

In this work, we make some progress towards proving the security of SPHINCS+ against quantum adversaries, by establishing lower and upper bounds for problems on hash functions that are related to the security of SPHINCS+, called the *subset cover problem* and some of its variants.

To show our results, we perform an analysis in an idealized model known as the Quantum Random Oracle Model (QROM). Before we explain this model, let us first discuss about oracles and black-box access, which are central notions in complexity theory.

Relativized worlds Oracles are used to answer the following question: let f be a function, what would happen if we had an efficient implementation of f in the real world? To answer this question, we must act as if we already had an efficient implementation of f, and then we can start deducing the consequences of such a statement. So, an efficient implementation of f exists in the *relativized world* where everyone is given a *black box oracle* access to f.

The notion of *black box* access models the fact that you can only use f as if you were given a black-box for it, i.e. an oracle whose underlying implementation is not accessible, but which you can query and it gives you answers. The world in which everybody has black-box access to f is called a relativized world because it is a world in which an efficient implementation of f is assumed to exist.

The motivation behind black-box access to f is that no matter what the implementation of f is, if there is an efficient one, then anything that holds in the relativized world with oracle access to f will hold in the real world. Hence the interest in black-box access comes from the study of relativized worlds. We note that the function f can be any function, in particular, it can be a one-way function — a function that is easy to compute, but hard to invert. Note that a hash function has to be a one-way function.

Impagliazzo's five worlds In his seminal work, Impagliazzo [Imp95] introduced five relativized worlds. Even today, we still do not know in which world we are, in fact, none of these worlds has been ruled out so far. The five worlds are described in Figure 1.1 where they are ranked from worst to best for cryptography.

1.1.2 One-Way Functions and Key Exchange

The existence of one-way functions is a central question in classical cryptography, as they are required to construct more advanced cryptographic primitives. One-way functions can be seen as a tool used by cryptographers to build cryptographic primitives. They are considered the minimal assumption for classical cryptography, which means that most

Algorithmica $\mathsf{P} = \mathsf{NP}$.
This is bad for cryptography, because there are no one-way functions in this world, hence there is no classical cryptography.

Heuristica $\mathsf{P} \neq \mathsf{NP}$ but NP problems are easy on average.
This world has a similar flavor to the previous one, because being able to solve NP problems efficiently on average is enough to show that one-way functions do not exist.

Pessiland $\mathsf{P} \neq \mathsf{NP}$ but one-way functions do not exist.
Since we do not know if one-way functions are equivalent to $\mathsf{P} \neq \mathsf{NP}$, this world is possible. For a cryptographer, this is as bad as the previous worlds, since no one-way functions means that classical cryptography is impossible.

Minicrypt One-way functions exist, but public key cryptography is impossible.
This is the first world in which some cryptography is possible. Minimalist cryptography, such as commitments [HR07] and signatures [Rom90] exist. But more sophisticated schemes do not necessarily exist, such as key exchange [IR89].

Cryptomania Public key cryptography is possible.
This is great for cryptography, because public key schemes are powerful primitives. Note that in this world, necessarily, one-way functions exist. There are still primitives that might not exist, such as indistinguishable obfuscation or fully homomorphic encryption.

Figure 1.1: Impagliazzo's five worlds.

schemes imply the existence of one-way functions, or equivalently, the non-existence of one-way functions implies the non-existence of most schemes. One can prove that the existence of a one-way function implies that the two complexity classes P and NP are different, where P is the class of problems solvable in deterministic polynomial time, and NP is the class of problems solvable in non-deterministic polynomial time. The question of whether $\mathsf{P} = \mathsf{NP}$ or $\mathsf{P} \neq \mathsf{NP}$ has been open for decades in complexity theory, and is one of the most important open problems in computer science. But we have just seen that showing that one-way functions exist would imply $\mathsf{P} \neq \mathsf{NP}$; hence, it is expected that this aforementioned question is extremely difficult to tackle. This shows how difficult it is to determine which world we live in among those proposed by Impagliazzo. That is why we are interested in a different, easier question: what would it mean if we lived in Minicrypt? What kind of cryptography would be possible? To answer this question, we need an oracle relative to which one-way functions exist.

Idealized one-way functions: the Random Oracle Model The Random Oracle Model (ROM) is an oracle model in which the function f is picked uniformly at random among all possible functions, and everyone is given oracle access to this function. It turns out that a random function is a perfect one-way function and hash function: since the output is uniformly random, it is almost impossible to predict the output of the function on any entry, hence to invert the function. This model is widely used in cryptography, for example for showing the security of cryptographic schemes: one can replace a candidate hash function with an idealized one, and thus prove the security assuming that the hash function is secure.

Key exchange An other important cryptographic primitive that we will consider is key exchange. A secure key exchange protocol is a protocol between two parties, Alice and Bob. At the beginning of the protocol, Alice and Bob share no common information. During the protocol, Alice and Bob communicate over a public channel, and at the end of the protocol, they both compute a key on their side. We say that the protocol is *correct* if with high probability, Alice's key and Bob's key are the same. The protocol is said *secure* if there exists no attacker Eve, who, given access to the messages sent between Alice and Bob during the protocol, can guess the key with high probability.

In 1989, Impagliazzo and Rudich [IR89] proved that there exists an oracle relative to which one-way functions exist, but key exchange does not. This is the opposite of showing a black-box reduction: this is a black-box impossibility result, also known as a black-box barrier. There is no construction of key exchange from one-way functions in a black-box way. This does not rule out all possible constructions, only the one that does not use the way the one-way function is implemented. However, black-box constructions are the most powerful and the most natural type of construction; hence, ruling out this type of construction remains interesting.

It was proven in 1984 by Bennett and Brassard [BB84] that quantum communication can be used to do quantum key distribution (QKD). Surprisingly enough, this is possible even without one-way functions. This example shows that quantum resources do not necessarily

give an advantage to the adversary, as it is the case in the post-quantum setting, but can instead advantage the honest parties. However, unlike the setting of post-quantum cryptography, the parties need a quantum channel to communicate.

Hence, an interesting question arises: what if we restrict the capabilities of the parties, and they are now only allowed classical communication? Is quantum key distribution still possible? It turns out that the answer is no. The follow-up question is then: what if there exist one-way functions on top of that? Does classical communication-based quantum key distribution exist in the *quantum* random oracle model? There exists a partial answer to that question, as it was shown in [ACC+22] a conditional impossibility result. Their result is partial because there are two restrictions. First, the protocol needs to be perfect, meaning that Alice and Bob must agree on the same key with probability one. Secondly, if Alice and Bob make quantum queries to the oracle, then the result holds only if a conjecture on the distribution of polynomials holds. While these restrictions make it seems like this result is weak, we believe that it reveals how difficult the question is in the general setting.

This work In this thesis, we extend their impossibility result to the setting where, after a first phase of classical communication, the last message sent in the protocol is a quantum state, but Alice cannot query the one-way function after receiving this last message from Bob. This is the opposite setting of the construction of quantum key distribution in [BB84], where the first message is quantum and the subsequent messages are classical. We also show that if Alice and Bob query the oracle classically, then imperfect protocol are also ruled out.

Our result implies the impossibility of building public key encryption with classical public key and quantum ciphertexts, in the quantum random oracle model, where the decryption algorithm makes no query to the oracle. This stands in spark contrast to a known construction of quantum public key encryption from one-way functions [BGH+23]. These results raise two important issues: the first one is the variety of different types of quantum public key encryption that exist and whether they can be constructed from standard assumptions, and the second concerns the appropriate definition for Cryptomania in the quantum setting.

1.1.3 New Worlds in Quantum Cryptography

These concerns suggest that Impagliazzo's five worlds need to be adapted for the quantum setting, and new worlds might need to be defined. We start by defining **MiniQcrypt**, the world where quantum computers exist, quantum-resistant one-way functions exist, but quantum-secure public key cryptography does not. We also define **Quantum Cryptomania**, the world where quantum-secure public key cryptography exists. Defining these worlds is quite natural, but even more recently, new worlds have emerged.

Pseudorandom Quantum States In their seminal work, Ji, Liu, and Song introduced Pseudorandom Quantum States (PRSs) [JLS18]. They are motivated as the quantum

equivalent of PRNGs: PRSs are quantum states that can be generated by a quantum polynomial-time algorithm, such that no quantum polynomial-time algorithm can distinguish them from Haar random states. Haar random states follow the Haar measure, which can be seen as the quantum equivalent of the uniform distribution[1]. Thus, similar to Pseudorandom Generators (PRGs), whose output looks like a random string, PRSs' output looks like random quantum states. In their paper, they show that PRSs can be constructed from one-way functions, and subsequent works have demonstrated that PRSs can be used to construct quantum cryptographic primitives, such as quantum commitments, quantum signatures and oblivious transfers [MY22b, AQY22].

Kretschmer's oracle The key result that sparked the interest of quantum cryptographers was a black-box impossibility result by Kretschmer in 2021 [Kre21]. He showed that there is no black-box construction of one-way functions from PRSs. More precisely, he exhibited an oracle relative to which PRSs exist, but $\mathsf{BQP} = \mathsf{QMA}$, which is the quantum equivalent of $\mathsf{P} = \mathsf{NP}$. Thus quantum cryptography is possible even if $\mathsf{BQP} = \mathsf{QMA}$, whereas, as mentioned earlier, in the classical setting, $\mathsf{P} = \mathsf{NP}$ rules out all cryptographic schemes.

Kretschmer's oracle separation has significant implications for quantum cryptography because it means that quantum secure one-way functions are not the weakest assumption required for quantum cryptography. In fact, PRSs are plausibly a *weaker* assumption for quantum cryptography.

Thus, it was natural that **Microcrypt**[2] was introduced in the literature as a world where Pseudorandom Quantum States exist, but one-way functions do not. In this world, classical cryptography is impossible, but quantum cryptography is possible.

Kretschmer's work left many open questions at the time:

1. What is the minimal assumption for quantum cryptography? It is not OWF, and PRSs are a good candidate, but could there be something weaker?
2. What can we construct from PRSs? What can we *not* construct from PRSs? Since OWFs imply PRSs, anything that cannot be constructed from OWF cannot be constructed from PRSs either. But what about other schemes?
3. How do quantum cryptographic primitives relate to each other? Or equivalently, what are the different worlds that we can define, and how do they relate to each other? At this point, many new primitives have been introduced in the literature. Can we quantify how powerful (or weak) they are, by comparing them to one another?

Many advances have been made since Kretschmer's work in 2021, but there is still no definitive answer to these questions. The best candidate for the minimal assumption for quantum cryptography is EFI pairs, a pair of efficiently generable distribution over quantum states that are statistically far, but for which there is no efficient algorithm that can distinguish

[1]The notion of *measure* comes from the fact that we are considering a continuous distribution, whereas in the classical setting, we are considering a discrete distribution.

[2]The name is due to Tomoyuki Morimae.

them. It has been shown that EFI pairs are weaker than PRSs [CCS24, AGL24], and can be used to construct cryptographic primitives, such as quantum commitments [MY22b] and thus MPC [BCQ23].

This work In this thesis, we study the relationship between different quantum assumptions.

We study the output length of pseudorandom quantum states and show that relative to Kretschmer's oracle where PRSs exist, short-PRSs — i.e. PRSs whose output length is logarithmic — do not exist. We also show that if an isoperimetric inequality-style conjecture is true, then there exists a quantum oracle relative to which short-PRSs exist but "pure"-PRSs do not. Classically, a pseudorandom number generator (PRG) with any fixed output length enables one to generate arbitrarily long pseudorandom sequences, either by recursively composing the PRG with itself to extend its output or by simply truncating its output when fewer bits are needed. Our results suggest that there is no such techniques for quantum pseudorandomness.

Finally, we study the relationship between PRSs and quantum public key encryption and signatures, clarifying the known constructions and impossibility results in the literature, and improving some of them.

1.1.4 Organization of This Thesis

In Chapter 3 we study the security of the subset cover problem and its variant. We show a lower bound for the k-restricted subset cover problem that matches the upper bound of [YTA22]. We show a lower bound for the $(1, k)$-subset cover problem, and present quantum algorithms for the (r,k)-subset cover problem. This chapter is based on joint work with Alex Bredariol Grilo and Damien Vergnaud and appears in the following papers.

Quantum security of subset cover problems [BGV23]

In Chapter 4 we study the construction of public key encryption in the quantum random oracle model. We show that public key encryption with quantum ciphertexts is impossible in the quantum random oracle model when the decryption algorithm makes no query to the oracle. This chapter is based on joint work with Alex Bredariol Grilo, Quoc-Huy Vu and Damien Vergnaud and appears in the following paper.

Towards the Impossibility of Quantum Public Key Encryption with Classical Keys from One-Way Functions [BGVV24]

In Chapter 5 we study the construction of cryptographic schemes from quantum pseudorandomness. We show separations between different quantum cryptographic primitives, and also study the possibility of constructing different kinds of public-key encryption and signatures from PRUs. This chapter is based on joint work with Minki Hhan, Garazi Muguruza and Quoc-Huy Vu and appears in the following papers.

Quantum Pseudorandomness Cannot Be Shrunk In a Black-Box Way [BM24]

On Limits on the Provable Consequences of Quantum Pseudorandomness [BHMV25]

1.2 Post-quantum Security of Sphincs(+)

1.2.1 The Signature Scheme Sphincs(+)

The SPHINCS+ signature scheme and its predecessor SPHINCS [BHH+15] make use of a Merkle-hash tree and of HORST, a variant of a hash-based scheme called HORS [RR02]. HORS (for "Hash to Obtain Random Subset") uses a hash function to select the subset of secret pre-images to reveal in a signature, and the knowledge of these secrets for several subsets may not be enough to produce a forgery—a property that makes HORS a *few-time signature* scheme. To prove that SPHINCS and SPHINCS+ are resistant to quantum attacks, we must first prove the quantum security of HORS and HORST.

Subset cover The security of HORS (and HORST) relies on the hardness of finding a subset cover (SC) for the underlying hash function. More formally, for any $r, k \in \mathbb{N}$, to define the (r,k)*–SC problem*, we consider the hash function as the concatenation of k hash functions $h_1, \ldots, h_k$ and the problem is to find $r+1$ elements $x_0, x_1, \ldots, x_r$ in the hash function domain such that $x_0 \notin \{x_1, \ldots, x_r\}$, and

$$\{h_i(x_0) | 1 \leq i \leq k\} \subseteq \bigcup_{j=1}^{r} \{h_i(x_j) | 1 \leq i \leq k\}\,.$$

The hardness of this problem for concrete popular hash functions has not been studied in depth, but Aumasson and Endignoux [AE17] proved in 2017 a lower bound on the number of classical queries to hash functions for the SC problem in the Random Oracle Model (ROM). However, the exact security of HORS (and more generally HORST, SPHINCS and SPHINCS+) with respect to quantum attacks is still not clear. With the recent selection of SPHINCS+ by the NIST standardization process, we believe it is important to analyze its security not just against classical adversaries, but also against quantum adversary. Since quantum computing provides speedups for many problems (e.g. Grover's search algorithm [Gro96] and Brassard, Høyer, and Tapp [BHT98] collision search algorithm), we expect this problem to be solvable more efficiently on a quantum computer.

Our results In this work, we explore the difficulty of finding a subset cover for idealized hash functions using quantum algorithms. We also consider a variant called the k-restricted subset cover (k–RSC) problem where, given k functions $h_1, \ldots, h_k : \mathcal{X} \to \mathcal{Y}$ such that $N = |\mathcal{Y}|$, one has to find $k+1$ elements $x_0, x_1, \ldots, x_k$ such that:

$$\forall 1 \leq i \leq k, h_i(x_0) = h_i(x_i)$$

and $x_0 \notin \{x_1, \ldots, x_k\}$.

This variant was defined recently by Yuan, Tibouchi and Abe [YTA22], who showed a quantum algorithm to solve it.

The main contributions of this work are:

1. **Lower bound on k–RSC**: we prove that $\Omega\left((k+1)^{-\frac{2^k}{2^{k+1}-1}} \cdot N^{\frac{2^k-1}{2^{k+1}-1}}\right)$ quantum queries to the idealized hash functions are needed to find a k–RSC with constant probability.
(Theorem 3.8)
2. **Lower bound on $(1,k)$–SC**: we prove that $\Omega\left((k!)^{-1/5} \cdot N^{k/5}\right)$ quantum queries to the idealized hash functions are needed to find a $(1,k)$–SC with constant probability.
(Theorem 3.13)
3. **Upper bound on (r,k)–SC:** we present a quantum algorithm that finds a (r,k)–SC with constant probability using $O\left(N^{k/(2+2r)}\right)$ queries to the hash functions when k is divisible by $r+1$, and $O\left(N^{k/(2+2r)+1/2}\right)$ otherwise.
(Theorem 3.21)

We now explain how we achieve these results, and start by discussing the random oracle model, an idealized model for hash functions.

1.2.2 The Random Oracle Model

In the Random Oracle Model (ROM) [BR93], every party has access to a random function. More precisely, a function f is chosen uniformly at random from the set of all possible functions, and then every party has access to this function f through an oracle $\mathcal{O}_f$.In this model, one can compute the probability that certain events happen with respect to the distribution of the function. For example, we know that the function f is a one-way function with overwhelming probability; hence the ROM is a model for idealized one-way functions. Moreover, a random function is, in fact, the perfect hash function: since the value of the function on any input is independent of the rest of the function, finding a collision is as difficult as it can be.

If there is a cryptographic construction that uses hash functions, then replacing the hash function with one drawn from the ROM is equivalent to assuming that your hash function is perfect. A proof of security in the ROM can be interpreted as the primitive being secure, unless one can break the security of the hash function. Here, we want to show a lower bound on the success probability of an algorithm in terms of the number of queries it makes to the random oracle. The benefit of proving such lower bounds, is that they relativize: the time and space complexity of the algorithm is irrelevant here—only the number of queries matters. Hence, if the algorithm is given access to another independent oracle, the lower bound will still hold.

Showing lower bounds in the ROM is usually done using combinatorial techniques. As a toy example, consider giving a lower bound on the success probability of finding a pre-image of 0, that is, an x such that $f(x) = 0$[3]. The formalism is as follows: the oracle $\mathcal{O}_f$ implements the function

$$f : \{0,1\}^n \to \{0,1\}^m ,$$

[3]Since the function is random, finding a pre-image of 0 is as hard as finding a pre-image of any y.

where $n, m \in \mathbb{N}$ are fixed. More precisely, $\mathcal{A}^{\mathcal{O}_f}$ is an algorithm that is given access to f via $\mathcal{O}_f$, and the function f is chosen uniformly at random before the execution of $\mathcal{A}^{\mathcal{O}_f}$. We are interested in bounding the probability:

$$\Pr_f \left[f(x) = 0 \,\middle|\, x = \mathcal{A}^{\mathcal{O}_f}(\cdot) \right].$$

For any $x \in \{0,1\}^n$, the probability that $f(x) = 0$ is equal to:

$$\Pr_f \left[f(x) = 0 \right] = \frac{1}{2^m}.$$

Thus, every time an adversary queries the function, the probability that he finds a pre-image of 0 is $\frac{1}{2^m}$. Since these events are all independent, the probability that an adversary finds a pre-image of 0 after q queries is at most $\frac{q}{2^m}$. Since $\mathcal{A}^{\mathcal{O}_f}$ is a polynomial-time algorithm, it makes only a polynomial-time number of queries to $\mathcal{O}_f$, hence its success probability is bounded by

$$\Pr_f \left[f(x) = 0 \,\middle|\, x = \mathcal{A}^{\mathcal{O}_f}(\cdot) \right] \leq \frac{\mathsf{poly}(\lambda)}{2^m},$$

where λ is the security parameter. It follows, that on average, an adversary needs to make $O(2^m)$ queries to the function to find a pre-image of 0 with constant probability. Note that we only need the number of query to $\mathcal{O}_f$ to be polynomially bounded; hence, increasing the time or space complexity of $\mathcal{A}$ will not make him more efficient at inverting f. More precisely, the proof relativizes: let $\mathcal{O}'$ be any oracle independent of the random oracle. Then, any polynomially-bounded algorithm $\mathcal{A}^{\mathcal{O},\mathcal{O}'}$ who is given access the random oracle $\mathcal{O}$ and the oracle $\mathcal{O}'$ succeeds in finding the pre-image of 0 with probability at most

$$\Pr_{\mathcal{O}} \left[f(x) = 0 \,\middle|\, x = \mathcal{A}^{\mathcal{O},\mathcal{O}'}(\cdot) \right] \leq \frac{\mathsf{poly}(\lambda)}{2^m}.$$

We emphasize that $\mathcal{O}'$ can be *any* oracle Finally, this shows that if $m = \omega(\lambda)$, any algorithm succeeds with probability exponentially small at finding a pre-image of 0.

1.2.3 The Quantum Random Oracle Model

The Quantum Random Oracle Model was first defined in [BDF+11], and in this model, every party has *quantum* access to a random function. More precisely, a function $f : \mathcal{X} \to \mathcal{Y}$ is chosen uniformly at random, and every party has access to a unitary oracle $\mathcal{O}_f$ that acts on two registers as follows:

$$|x\rangle_{\mathbf{X}} |y\rangle_{\mathbf{Y}} \to |x\rangle_{\mathbf{X}} |y + f(x)\rangle_{\mathbf{Y}}.$$

Two registers are needed to ensure that the operation is unitary: indeed, we have that $\mathcal{O}_f^2 = I$, where I is the identity matrix. If we instead chose to implement it with one register, as follows:

$$|x\rangle_{\mathbf{X}} \to |x + f(x)\rangle_{\mathbf{X}},$$

then this would not necessarily be unitary: for example, if f is such that for all x, $f(x) = -x$, then $\mathcal{O}_f$ would be equal to the null matrix. Not also that since $\mathcal{O}_f^\dagger = \mathcal{O}_f$, every party also has access to the inverse of $\mathcal{O}_f$.

Because of how we model quantum computers, and therefore because of quantum mechanics, it is hard to quantify the knowledge that an algorithm has about the random function after a query. In the classical setting, it is straightforward: if you query x, you learn $f(x)$; but in the quantum setting, you do not learn anything until a measurement is performed. For some time there was no generic way of proving lower bounds in the quantum setting and every proof was unique. It was not possible to "just" do combinatorics in the QROM as one can in the ROM, and thus new techniques were needed.

1.2.4 The Compressed Oracle Technique

In 2019, Zhandry [Zha19] introduced the Compressed Oracle Technique, which can be used to simulate a quantum random oracle on-the-fly. The idea is somewhat simple: instead of picking a function f at random at the beginning of the computation, we initialize a new register $\mathbf{F}$ that contains the uniform superposition over all possible function, i.e.,

$$\frac{1}{|F|}\sum_{f\in F}|f\rangle\,.$$

This is similar to postponing the measurement of the function f until the end of the computation. The register $\mathbf{F}$ contains the truth table of the function f. Zhandry showed that this perfectly simulates the quantum random oracle model. Note that this is not efficient and thus cannot be used as is for lazy sampling, which is the efficient simulation of sampling a distribution without sampling the whole distribution beforehand. For the random oracle model, this means simulating *queries* to the random oracle without generating the whole function in advance. In the classical setting, an adversary keeps a table of all the queries made before by the algorithm, and returns the output that corresponds to the input if it exists; otherwise it samples an output uniformly at random, records it, and returns it to the algorithm. No algorithm can distinguish between such lazy sampling and a random oracle. Moreover, the sampling is efficient: if the algorithm is polynomially bounded, so is the adversary. In the quantum setting, however, such lazy sampling does not work, because with one query over the uniform superposition, the adversary would need to sample and store the entire oracle, which would not be efficient.

Quantum lazy sampling To perform quantum lazy sampling, Zhandry introduces a new symbol $\perp$. The new register that contains the function f consists of a database, in which for each entry $x \in \mathcal{X}$ there is an associated register $\mathbf{X}$ where $f(x)$ is stored. Thus the register $\mathbf{F}$ that contains the function can be written as

$$|f\rangle_{\mathbf{F}} = \bigotimes_{x\in\mathcal{X}}|f(x)\rangle_{\mathbf{X}}\,.$$

Before the computation begins, the register is initialized with the uniform superposition

$$\bigotimes_{x\in\mathcal{X}} \frac{1}{|\mathcal{Y}|}\sum_{y\in\mathcal{Y}} |y\rangle_{\mathbf{X}}\,.$$

In the Fourier basis, this becomes

$$\bigotimes_{x\in\mathcal{X}} |\hat{0}\rangle_{\mathbf{X}}\,.$$

Then comes the compression part: every $\hat{0}$ is mapped to $\perp$. To decompress, we map every $\perp$ back to $\hat{0}$. The oracle register is always maintained in it compressed form, and a query decompresses the register, queries the oracle, then compresses it again. This time, the simulation can be done efficiently: Zhandry showed that the $\mathbf{F}$ register is supported on only q vectors after q queries. Moreover, no polynomial-time algorithm can distinguish between the compressed oracle and a real oracle (this time the simulation is not perfect: the compression and decompression operators introduce some losses).

Compressed oracle for lower bounds We can use the compressed oracle model to prove lower bounds on query complexity. The analysis somewhat resembles the classical analysis in the ROM. The register $\mathbf{F}$, where the function is stored, contains all the information that the adversary has learned about the function so far. Hence a recursive analysis is possible: we bound the probability that the adversary wins the game after q queries by:

i. the probability that he won with $q-1$ queries, plus

ii. the probability that he did not won with $q-1$ queries, but wins with the q^{th} query.

The main difference compared to the classical setting is that we work with amplitudes instead of probabilities, which introduces a square-root factor. Moreover, the analysis is more complicated: the probability that the adversary wins with the q^{th} query is not independent of the previous queries made to the oracle. This is not really an issue for "unstructured" problems such as finding a collision or a pre-image, but it becomes one when the problem has more structure. Unlike in the classical setting, where one query can reveal only limited information about the oracle, in the quantum setting, one can choose which information to extract.

Let us take the example of the collision problem: the goal is to find two distinct values x_0, x_1 such that $f(x_0) = f(x_1)$. The best quantum algorithm for finding a collision [BHT98] achieves a 1.5-exponent speedup over the best classical algorithm by first making t classical queries to learn a set of "targets" values, and then uses Grover's algorithm to find a pre-image among these values. The choice of t can in principle be arbitrary, but when analyzing in the compressed oracle model, one must account for *all* possible t. Indeed, the success probability of finding a pre-image with the q^{th} query depend on the number of target t, and since an algorithm can use any strategy, all values must be considered. Although most choices of t yield only negligible success, they nonetheless contribute terms to the analysis.

When considering *recursive* problems, such as multi-collision finding (see [LZ19]), the analysis becomes very involved, because one must bound many terms, most of which turn out to be negligible. We are still able to show lower bounds in the quantum settings, but the proofs are much more involved than in the classical setting for the same problems, and there are negligible terms that do not appear in the classical setting, which we believe are unavoidable. We now explain a high-level overview of how we obtain our bound for the problem of RSC.

1.2.5 Technical Overview

In Section 3.1, we prove a lower bound on the query complexity to solve the RSC problem. We consider an algorithm $\mathcal{A}$ after i quantum queries to the random oracle and denote its state at this moment by $|\psi_i\rangle$. Our goal is to compute an upper bound for the value $|P_k^{RSC}|\psi_i\rangle|^2$, where P_k^{RSC} is the projection onto the databases that contain a k–RSC. Computing such a bound leads to a lower bound on the number of queries needed to solve k–RSC with constant probability. To prove our bound, we proceed by induction: assuming we have proved a bound for the k'–RSC problem for all $k' < k$, we then prove a bound for the k–RSC problem. The analysis is naturally divided into two parts: whenever $\mathcal{A}$ finds a k–RSC after i quantum queries, it means that either:

1. $\mathcal{A}$ finds it after $i-1$ quantum queries;
2. or $\mathcal{A}$ finds it with the i^{th} quantum query.

The first is handled recursively. It remains to bound $|P_k^{RSC}|\psi_i\rangle|$ in the second case. In this second case, the database (after $i-1$ quantum queries) must contain a certain number of k'–RSC (for some $k' < k$) in order for $\mathcal{A}$ to find a k–RSC with the i^{th} query. Using this strategy, we obtain a recursive formula from which we can deduce the bound on $|P_k^{RSC}|\psi_i\rangle|$.

In Section 3.2.1, we prove a lower bound for the $(1,k)$–SC problem. The idea of the proof is similar to the proof for the lower bound of the k–RSC problem but we must compute a bound for another problem that we define: the *j–repetition* problem.

Finally in Sections 3.2.2 and 3.2.3, we design a family of quantum algorithms for finding a (r,k)–SC. These algorithms are inspired by the algorithm from [YTA22] for solving the k–RSC problem and [LZ19]'s algorithm for finding multi-collisions. These algorithms are recursive and take as input two parameters $t, k' \in \mathbb{N}$, performing the following steps:

1. Find t distinct $(r-1,k')$–SC;
2. Find the (r,k)–SC.

The parameters t and k' are chosen in order to optimize the complexity of the algorithm. The first step is carried out by applying the algorithm for the value k' $r-1$ times, and the second step uses Grover's algorithm.

1.2.6 Related Works and Discussions

Restricted Subset Cover There is currently only one quantum algorithm for finding RSC [YTA22]. Our lower bound for finding a RSC matches their upper bound when k, the number of functions, is constant. However when k is not a constant, their algorithm makes $O\left(k \cdot N^{\frac{2^k-1}{2^{k+1}-1}}\right)$ queries to $h_1, \ldots, h_k$, which roughly leaves a $k^{3/2}$ gap between the best known attack and our lower bound. To the best of our knowledge, this is the first lower bound on the RSC problem for a quantum algorithm, and there are no such results for classical algorithms. It would be interesting to see if this gap can be further closed.

Tighter bounds for $(1,k)$–SC When k is constant, the lower bound for $(1,k)$–SC is $\Omega\left(N^{k/5}\right)$, while our algorithm for this problem makes $O\left(N^{k/4}\right)$ queries to the oracle (when k is even). It would be interesting to tighten this gap, especially since the results for $(1,k)$–SC are likely necessary to prove the lower bounds for (r,k)–SC when $r \geq 2$.

For non-constant k, our lower bound for $(1,k)$–SC is

$$\Omega\left(C_k^{-1/5} \cdot N^{k/5}\right),$$

where $C_k = \sum_{j=2}^{k} \frac{k!}{(j-1)!} \leq k! \cdot e$. Notice that this term cannot be neglected for large values of k. For example with $k = \log(N)$, we have $C_k \geq N$. In comparison, our best algorithm for $(1,k)$–RSC has a factor in k given by

$$\binom{k}{(k+1)/2}^{-1/2} \leq \frac{2^{(k+1)/2}}{\left(\frac{k+1}{2} \cdot \pi\right)^{1/4}},$$

which is very far from our bound on C_k. It would also be interesting to see if this gap can be tightened further.

Bounds for (r,k)–SC Unfortunately, expanding our result for the (r,k)–SC problem is much more complicated than the case $r = 1$; in fact, even proving the case $r = 2$ is non-trivial. To prove such a result, one would need a bound for the problem of finding j distinct $(1,k)$–SC instances. While proving such a bound is challenging, it is also unclear how to define the problem of finding j distinct $(1,k)$–SC instances. Indeed, an important property for our technique in the first lower bound proofs is that, by making one query to the oracle, the adversary cannot find two or more k–RSC. The same property must hold for the problem of finding j distinct $(1,k)$–SC instances, and this definition—and the subsequent analysis—remains open. However, in a concurrent work, Yuan, Tibouchi and Abe [YTA23] showed a lower bound for the (r,k)–SC that appears to be tight. They invoke [YZ21, Theorem 4.12], which gives a generic, fully classical framework for computing lower bounds using only combinatorial arguments. However, the resulting bounds are usually not tight. Using this result, they show that the bound is $O\left(N^{\frac{k}{2(r+1)}}\right)$ which matches our algorithm when k is constant. Thus, surprisingly, the bound obtained is tight for the (r,k)–SC problem.

Security of SPHINCS and SPHINCS+ against quantum adversaries The signature scheme SPHINCS relies on the HORST scheme (for "HORS with trees"), which adds a Merkle tree to the HORS scheme to compress the public key. The security of HORST also relies on the (r, k)–SC problem, but the security of SPHINCS relies on different security notions for the underlying hash functions. In particular, it depends on a variation of the SC problem called the *target subset cover* (TSC) problem [RR02]. The main difference arises from the fact that the message signed using HORST is an unpredictable function of the actual message, preventing an attacker from constructing a subset cover beforehand.

Nevertheless, the authors of [BHH+15] stated an existential unforgeability result for SPHINCS [BHH+15, Theorem 1] under q_s-adaptive chosen message attacks. The success probability in such attacks is roughly upper-bounded by:

$$\sum_{r=1}^{\infty} \min\left(2^{r(\log q_s - h)+h}, 1\right) \cdot Succ_A((r,k) - SC),$$

where h is the height of the tree used in SPHINCS, and $Succ_A((r,k) - SC)$ denotes the success probability of an adversary $\mathcal{A}$ in finding a (r, k)–SC. The authors made the assumption that this term is negligible for any probabilistic adversary $\mathcal{A}$, and our quantum lower bound on the query number to find a $(1, k)$–SC can be seen as a first step towards proving this assumption (for idealized hash functions). With the bounds of Yuan, Tibouchi and Abe [YTA23] previously mentioned, one can now bound this term for specific parameters of the scheme.

SPHINCS+ is an enhancement of SPHINCS that improves efficiency, and its security relies on another variant of the SC problem, namely the interleaved target subset cover (ITSC) problem, which is not studied in this work.

1.3 Impossibility of Key Agreements from One-Way Functions

1.3.1 Key Agreements

In 1989, Impagliazzo and Rudich [IR89] published a paper that started a series of works on black-box constructions and separations of classical cryptographic primitives. In the quantum setting, after decades of focusing on the possibility of information-theoretically secure quantum protocols—initiated by the land-marking results on money schemes [Wie83] and key-agreement [BB84]—there has been recent progress in understanding how quantum resources can be used to implement cryptographic primitives under weaker computational assumptions.

More concretely, it has been shown in [GLSV21, BCKM21] that using quantum resources can be used to achieve Oblivious Transfer (OT) and Multi-party computation (MPC), two

central primitives in cryptography, from one-way functions (OWF), which are the weakest classical cryptographic assumption.

More recently, it has been asked if quantum protocols are possible for public key encryption from OWFs (or weaker assumptions). While the conditional impossibility result for key-agreement of [ACC+22] implies that public key encryption (PKE) from OWFs with classical communication is impossible—even if the honest parties are quantum[4]— it has been recently shown that PKE can be constructed from OWFs if we have a quantum public key [Col23, BGH+23, KMNY23]. However, having a quantum public key is not ideal, given the issues that arise with public key distribution, authentication, and reusability. These results leave open the question of whether quantum PKE from OWFs is possible with a classical public key and quantum ciphertext.

In this work, we extend the result of [ACC+22] by showing that key agreement is impossible when Alice and Bob exchange classical messages and in the very last round Bob sends a quantum message to Alice. Our result holds under the same conjecture as [ACC+22], but is limited to the setting where Alice does not query the random oracle in the last round of the protocol. More concretely, we achieve the following result.

Theorem 1.1 (Informal)**.** *Let* Π *be a key agreement protocol between Alice and Bob, where they first exchange classical messages and, in the last round, Bob sends a quantum message, and Alice and Bob agree on a key* k*. Let* n *be the number of queries that Alice and Bob make to a random oracle* $\mathcal{O}$*. Then, assuming Alice does not query the oracle after receiving the quantum message from Bob, Eve can recover* k *with* $\mathcal{O}(\mathsf{poly}(n))$ *classical queries to* $\mathcal{O}$ *with probability at least* $\frac{1}{\mathsf{poly}(n)}$*.*

With this result in hand, we show that quantum PKE (qPKE) is impossible with a classical public key in the Quantum Random Oracle Model (QROM), when the decryption algorithm does not query the random oracle.

Corollary 1.2 (Informal)**.** *Assume* $(\mathcal{Gen}, \mathcal{Enc}, \mathcal{Dec})$ *is a Public Key Encryption scheme in which the public key is classical and the ciphertext is a quantum state. Assuming the algorithms* $\mathcal{Gen}$ *and* $\mathcal{Enc}$ *make at most* n *quantum queries to a random oracle* $\mathcal{O}$*, then there exists an algorithm* Eve *that can decrypt by making* $\mathcal{O}(\mathsf{poly}(n))$ *classical queries to* $\mathcal{O}$*.*

Using known techniques from black-box separation, our results can easily be translated to yield separations of quantum PKE from black-box OWFs. We also note that our result (Corollary 1.2) marks an initial step towards proving the conjecture of [MY22a] regarding the possibility of black-box constructions of qPKE with classical public keys from quantum symmetric key encryption.

Moreover, we show the impossibility of imperfect key agreements for OWFs in the restricted setting where Alice and Bob make classical query to the OWF. This extends the result of [ACC+22], which only held for perfect protocols.

[4]Such a result is actually conditioned on a conjecture that we state in Conjecture 2.1.

Before we explain how we achieve our results, we begin by reviewing the proof of [IR89] in the classical setting.

1.3.2 In the Classical Setting

In the plain model To break the security of any key agreement in the plain model, it suffices to simulate Alice, postselected on the messages she sends during the protocol. An attacker Eve can achieve this in PSPACE, which is enough to establish the impossibility result.

In the Random Oracle Model One might assume that the same strategy applies in the Random Oracle Model. Simulating Alice is indeed possible: assuming that the oracles' output is random, Eve can simulate Alice by choosing oracle responses at random, while postselecting on the message sent to Bob, all within PSPACE. Unfortunately this is not enough to break the scheme's security. The key issue is that Alice's internal state is not independent of the actual oracle used during the protocol. By querying the oracle, Alice may learn valuable information essential for computing the key.

However, since Alice and Bob must agree on the same key, any information Alice uses must also be accessible to Bob. In [IR89], this dependence is captured through *intersection queries*— queries to the oracle made by both Alice and Bob. If the attacker can recover these intersection queries, they can reconstruct the secret key. The method they use to achieve this consists of incrementally guessing the oracle calls made by Alice and Bob. To do so, at each round of the protocol, Eve simulates Alice (or Bob) up to this round, and makes the same oracle queries as the simulated party. By repeating this simulation enough times, the attacker can, with high probability, determine all the intersection queries—ultimately breaking the scheme.

Barak and Mahmoody's method In [BM09], a similar approach is used to simulate Alice and Bob, but in a more refined way. Instead of repeatedly simulating their behavior, Eve predict the queries that they will make with the highest probability. More precisely, at each round, Eve computes the probability for each potential query u to the oracle for both Alice and Bob. She only considers ε-heavy queries, i.e. those made with probability at least ε/n, where n is the total number of queries by Alice and Bob. Eve then queries the oracle on these values, updating her knowledge and adjusting Alice and Bob's heavy queries. She repeats this operation until she queried all heavy queries. More precisely the attack is as follows

Eve's algorithm At every round of the protocol, Eve does the following:

1. Consider all values u not already queried by Eve such that:

$$P(\text{Alice calls the oracle } \mathcal{O} \text{ on } u|\text{Eve's knowledge}) \geq \frac{\varepsilon}{n} \text{ or}$$
$$P(\text{Bob calls the oracle } \mathcal{O} \text{ on } u|\text{Eve's knowledge}) \geq \frac{\varepsilon}{n}$$

2. Query $\mathcal{O}$ on the first such u in lexicographical order.
3. Repeat until there is no such u.

The analysis To prove that the success of the attack, two things must be shown: (1) that Eve learns the intersection queries with high probability and (2) that Eve is efficient. They prove that Eve makes at most $O(n^2)$ queries, which is optimal (see [GRM78]).

1.3.3 In the Quantum Setting

Challenges in the Quantum Setting Adapting this approach to the quantum setting introduces a problem: how do we define intersection queries? Since Alice and Bob can query the oracle in superposition, there is no straightforward way to characterize which queries they "share". Two natural definitions of intersection queries fail:

1. **Queries as quantum states:** One could define an intersection query as a quantum state that is queried to the random oracle. However, it is easy to construct a protocol where Alice and Bob learn information from the oracle without having any identical queries. With this definition, we fail at making intersection queries a quantitative description of the information that Alice and Bob both know about the oracle.
2. **Queries with high probability:** An alternative is to define intersection queries as those made with high probability. While similar to the classical case, this raises another question: how do we formally define such queries in a quantum setting?

We now explain how they resolve these issues in [ACC+22].

The attack of Austrin et al. The quantum attack of [ACC+22], introduces the concept of *quantum* heavy queries, which serves as the quantum analogue of the classical heavy queries from [BM09] These are defined as queries with high amplitude (see Definition 2.23 for a formal definition). However, since the notion of intersection queries does not extend naturally to the quantum setting, they propose the Polynomial Compatibility Conjecture (PCC) as a replacement. The PCC states that if two quantum states satisfy certain conditions, then there exists an oracle that is consistent with both states. This conjecture is key to their attack strategy. Their attack proceeds as in the classical setting, by learning all of the heavy queries. Then, Eve simply outputs the key that is the most likely. The correctness of Eve's attack follows from the PCC as follows:

1. If Eve does not output the correct key with high probability, then they construct an internal state of Alice that outputs key 1, and an internal state of Bob that outputs key 0.
2. By the PCC, there must exist an oracle h that is consistent with both states
3. Consequently, there must exist an execution of the protocol where Alice outputs key 1 while Bob outputs key 0, contradicting perfect correctness.

Crucially, the proof only works for protocols with perfect correctness—the proof does not extend to cases where correctness error is negligible. In this work, we extend their result to the setting where the last message of the protocol is quantum.

1.3.4 Technical Overview

To prove Theorem 1.1, we start with a key-agreement protocol with perfect correctness where Alice and Bob have quantum access to a random oracle and exchange polynomially many rounds of classical messages, and Bob sends a final quantum message $|\psi\rangle$ to Alice.

We show an attack where with inverse polynomial probability:

1. Given the classical transcript and $|\psi\rangle$, Eve guesses the key k of Alice and Bob.
2. Eve sends a forged quantum state ψ^E to Alice, such that Alice agrees on the key k at the end of the protocol.

While the first item is sufficient to break the security, the second step item allows a much stronger attack: Eve is an active adversary that not only retrieves the key, but is undetectable to Alice and Bob, since they both agree on the same shared key.

Finding the key Following [ACC+22, Construction 4.10], we define a quantum-heavy query learner algorithm (formally defined in Construction 2.1), which queries all ε-heavy queries.

In this overview, for simplicity assume that Bob's final message is a pure state $|\psi\rangle$. Using her knowledge, Eve simulates Alice's internal state $\left|\phi_A^E\right\rangle$. Then, she runs Alice's last step of the protocol $\mathsf{A}_{\mathsf{fin}}$ (which is public) on the simulated internal state of Alice and the quantum message from Bob. We then show that Eve retrieves the correct key with high probability, i.e., for some noticeable parameter ν:

$$\mathrm{Tr}\left(\Pi_{\mathsf{k}}\mathsf{A}_{\mathsf{fin}}\left(\left|\phi_A^E\right\rangle\!\left\langle\phi_A^E\right|_{\mathbf{W'_A}}\otimes|h\rangle\!\langle h|_{\mathbf{H}}\otimes|\psi\rangle\!\langle\psi|_{\mathbf{M}}\right)(\mathsf{A}_{\mathsf{fin}})^\dagger\right)\geq 1-\nu. \tag{1.1}$$

Here,$\mathbf{W'_A}$ contains Eve's simulated state of Alice $\left|\phi_A^E\right\rangle$, $\mathbf{H}$ contains the superposition of all possible oracles that are consistent with Eve's knowledge, $\mathbf{M}$ contains Bob's message $|\psi\rangle$, $\mathsf{A}_{\mathsf{fin}}$ corresponds to Alice's final operation and Π_{k} is the projector that measures the key.

This inequality means that, given Bob's *real* message $|\psi\rangle$, Eve can find the correct key by applying Alice's operation on the simulation state of Alice $\left|\phi_A^E\right\rangle$ that she obtained using the quantum-heavy queries learner.

The proof follows from the fact that since Alice does not query the oracle during $\mathsf{A}_{\mathsf{fin}}$, so the register $\mathbf{H}$ is unchanged and thus the resulting state keeps the properties necessary to apply the PCC. The full proof is in Section 4.2.2.

Forging the final message Next, Eve forged a quantum message ψ^E that she sends to Alice. The idea is the following: Eve will pick the post-measurement state from Inequality 1.1, and apply $\mathsf{A}_{\mathsf{fin}}^\dagger$ to it. Then, Eve traces out the registers $\mathbf{W}'_{\mathbf{A}}$ and $\mathbf{H}$, leaving only ψ^E as the remaining state in register $\mathbf{M}$.

To show that Alice computes the same key as Bob and Eve with high probability, we show that ψ^E is close to the real message $|\psi\rangle$:

$$\langle\psi|\,\psi^E\,|\psi\rangle \geq 1-\nu. \tag{1.2}$$

Using Inequality 1.2 and the perfect correctness of the protocol, we show that Alice and Bob will agree on the same key with high probability. This corresponds to proving the following inequality:

$$\mathrm{Tr}\left(\Pi_{\mathsf{k}}\mathsf{A}_{\mathsf{fin}}(|\phi_{\mathsf{A}}\rangle\langle\phi_{\mathsf{A}}|_{\mathbf{W}_{\mathbf{A}}}\otimes|h\rangle\langle h|_{\mathbf{H}}\otimes\psi^E)\,(\mathsf{A}_{\mathsf{fin}})^\dagger\right)\geq 1-\nu, \tag{1.3}$$

where $|\phi_{\mathsf{A}}\rangle$ is Alice's real internal state and ψ^E is the forged message that Eve sends to Alice. This ensures that given the message of Eve, Alice will find the same key as Eve with high probability when she does her final computation. The proof appears in section 4.2.2.

Finally, Corollary 1.2 follows from the fact that, if public key encryption with quantum ciphertexts is possible, then we can construct a key agreement protocol: Alice sends the public key and Bob answers with the encryption of a random key k. Since our attack breaks key agreement, it also breaks public key encryption with quantum ciphertexts that follows the same restrictions.

1.3.5 Related Works and Discussions

The Polynomial Compatibility Conjecture. First introduced in [ACC+22], the Polynomial Compatibility Conjecture (PCC) is already known to imply separation results for key agreement [ACC+22] and non-interactive commitments [CLM23]. The conjecture has an alternative expression that uses polynomials and is equivalent to the statement in Conjecture 2.1. The PCC is known to be true with exponential parameters [ACC+22], but it is still open with polynomial parameters. Proving it would be interesting as it would now also establish the separation result for quantum PKE, along with potentially more results as it is a strong statement.

Quantum Public Key Encryption. Classically, public key encryption (PKE) cannot be constructed from black-box one-way functions [IR89]. In the quantum context, various definitions of quantum PKE exist, leading to different feasibility outcomes. With quantum

public keys and classical ciphertexts, quantum PKE can be constructed from one-way functions [Col23, BGH+23, KMNY24, MW24]. However, it remains unclear how the distribution of such public keys could be effectively distributed in practice among different parties. Our result focuses on quantum PKE with classical public keys and quantum ciphertexts. In this setting, the distribution of public keys could be implemented using currently available public key infrastructure. Moreover, compared to having a quantum public key and a classical ciphertext, having a classical public key and a quantum ciphertext is less problematic for implementations, as the message is supposed to be received by only one party and thus the potential destruction of the message after the decryption is inconsequential. With this definition of quantum PKE, we achieve a step towards proving a similar result as the classical case.

Classical Communication One Quantum Message Key Agreement Protocols. In our work, we introduce a scheme that we call Classical Communication One Quantum Message Key Agreement (CC1QM-KA) protocols. In these types of protocol, Alice and Bob communicate classically, except for the last message that is quantum. We show that key agreement is impossible with this type of protocol in the QROM if Alice does not query the random oracle after receiving the last message.

One natural question is what happens if we allow the *first* message to be quantum, while the rest of the communication is classical. Interestingly enough, [BB84] falls into this category of protocol, thus key agreement is possible *unconditionally* in this setting. This asymmetry in terms of feasibility results is quite surprising. A possible explanation is that we cannot postselect on quantum messages, i.e. prepare a global state conditioned on measuring a subregister and obtaining a specific outcome. Indeed, the classical Eve attacks imply a simulation of the internal state of the parties that is consistent with the message, which corresponds to computing an internal state postselected on the classical messages that are communicated. With a quantum message, this would be possible with a classical description of the quantum message since Eve is unbounded, but it is non-trivial with only the quantum state as Eve must learn what the quantum message is in the first place somehow. However, in the CC1QM setting, we do not need to do this postselection, as a simulation of the last part of the protocol is enough to find the right key.

Allowing oracle queries in the decryption algorithm To prove the stronger result that qPKE is impossible even when the decryption algorithms query the oracle, one needs to show an attack on CC1QM-KA protocols where Alice queries the oracle in the last part of the protocol. At first glance, one may think that Inequality 1.1 should be true even if Alice makes queries to the oracle in $\mathsf{A}_{\mathsf{fin}}$, because every new information about the oracle that she learns at this stage of the protocol will not be transmitted to Bob since there is no communication afterward. However, some issues that do not appear in [ACC+22] arise when trying to prove such an inequality.

The first (natural) problem is that since the last message is quantum, Eve cannot compute the heavy queries (which would be sufficient for the attack). Therefore, we need to find

another way of simulating Alice's last oracle calls without learning the heavy queries.

A first attempt is to use the operator $\mathsf{A}_{\mathsf{fin}}^{\mathcal{O}}$, that corresponds to Alice's computation in the last step of the protocol with the *real* oracle $\mathcal{O}$. Because this corresponds to the operation that the real Alice would have done and the real outcome is deterministic (since the protocol has perfect correctness), it could allow Eve to find the real key. However, the problem in this approach is that Eve has her simulated state that was constructed using a *simulated* oracle (with correct values for heavy-queries) and Alice's algorithm could use some consistency check that would fail when we decide to change the oracle.

On the other hand, if we want to use the simulated oracle instead of the real oracle, then there is a trivial protocol for which the attack does not work. In this protocol, Bob just picks a random value $x \in \mathcal{X}$, queries it, and sends $|x\rangle$ to Alice. Alice and Bob agree then on the key $H(x)$. By using the simulation oracle, Eve would not be able to find the key with non-negligible probability.

While these two complications are artificial since they do not lead to a secure protocol, they put a barrier to finding a common attack that would make Eve find the keys from Alice and Bob.

Comparison with other work and discussion Recently, a series of papers by Li, Li, Li and Liu [LLLL24, LLLL25b, LLLL25a] showed new impossibility results for building quantum public key encryption in the quantum random oracle model. Their proof techniques are very different from the ones of [ACC+22] and this thesis, as they use entropy. They do not rely on the conjecture to achieve their results, and they show the impossibility of the existence of perfect quantum public key encryption in the quantum random oracle model when the secret key is classical. Hence their result is strictly better than Corollary 1.2 if the secret key is a classical string, but they do not say anything about qPKE with quantum secret keys. It is unknown if their result can be extended to a more general setting with a quantum secret key or imperfect correctness.

1.4 Quantum Pseudorandomness

We have seen that quantum cryptographic primitives differ from classical cryptographic primitives. Quantum primitives can take many forms, for example, key exchange can have either classical or quantum communication. But a more fine-grained analysis is possible, as quantum communication can occur at the beginning or the end of the protocol. The same applies to other quantum cryptographic primitives: in classical definitions, every string can become a quantum state. This is the case, for example, in public key encryption, where the secret key, public key or ciphertext can be either classical or quantum. A classical string is stronger in this context, because as quantum states are more general. Thus if a cryptographic primitive exists with a classical string, it also exists with a quantum state.

In Section 5.3, we examine the different types of public key encryption and the known construction and separation results in the literature. We conduct a similar study in Section 5.4

for one-time signatures and digital signatures.

Pseudorandom Quantum States We can also define the quantum equivalent of classical tools for cryptography, such as OWFs and PRGs. This is what Ji, Liu and Song [JLS18] did in 2018 when they introduced a quantum analog of PRGs, called *Pseudorandom Quantum States* (PRSs). They consist of a family of polynomial size keyed-states $\{|\phi_k\rangle\}_{k\in\mathcal{K}}$ that can be efficiently generated, and such that no quantum polynomial-time algorithm can distinguish between a randomly sampled element from the PRS family or a Haar-random state (see Definition 2.4 for a formal definition). Haar-random states are states that follow the Haar measure, which is a measure over quantum states. We will not delve into the Haar measure in this work, but it can be interpreted as the quantum analog of the uniform distribution. It is a continuous distribution, unlike the classical uniform distribution over strings which is discrete. An introduction to the Haar measure can be found in [Mel24] for interested readers. The difference with PRGs is that instead of generating classical pseudorandomness, we generate quantum pseudorandomness. Also note that in the security definition, we allow the adversary to have a polynomial number of copies of the states.[5]

PRSs are relevant to quantum cryptography, as we already know how to construct several cryptographic primitives from (variants of) PRSs: public key encryption with quantum keys [BGH+23], quantum one-time signatures [MY22b], pseudo one-time pad encryption schemes [AQY22], statistically binding and computationally hiding commitments [AQY22, MY22a, KT24a] and quantum computational zero knowledge proofs [BCQ23]. Such rapid interest probably derives from the fact that PRSs can be constructed from OWFs [JLS18] (and thus PRGs), but there exists oracle separations between OWFs and PRSs [Kre21, KQST23, KQT24], which makes them a potentially weaker building block for quantum cryptography, with a purely quantum description.

1.4.1 Minimal Assumptions for Quantum Cryptography

These recent results raise the following question: what is the minimal assumption for quantum cryptography? In 2021, Kretschmer [Kre21] exhibited an oracle relative to which one-way functions do not exist, while pseudorandom quantum states exists. This separation proves that unlike in the classical setting, the minimal assumption for quantum cryptography is not one-way functions. On the other hand, pseudorandom quantum states are a good candidate.

In this section, we start by discussing the quantum primitives that were introduced in recent literature. We note that most of these primitives are inspired by classical primitives that were introduced before, sometimes more than 20 years ago. In the classical setting however, most of these primitives have been shown to be equivalent to PRGs. In the following, we

[5]For readers familiar with t-designs, there are two key differences between t-designs and PRSs: the type of indistinguishability and the number of copies. While a PRS must be computationally indistinguishable from a Haar-random state given any polynomial number of copies to the adversary, a t-design must be statistically close to t copies of a Haar-random state.

say that primitive A implies primitive B if there is a black-box construction of primitive B from primitive A. On the other hand, we say that primitive A cannot be constructed from primitive B, or that primitive A is separated from primitive B, if there is no black-box construction of primitive A from primitive B, or equivalently, if there is an oracle relative to which primitive B exists, but primitive A does not.

We discuss the different kinds of black-box separations in Section 1.4.4. In Figure 1.2, we present a graph that summarizes the relationship between these primitives.

Pseudorandom Function State Generator Pseudorandom Function State Generators (PRFSs) [AQY22, AGQY22], are the quantum analog of pseudorandom functions and a natural generalization of PRSs. Whereas a PRS is a single generator G that, given a key k, outputs a pseudorandom state $|\phi_k\rangle$, a PRFS is a family of generators $\{G_k\}$ that each take an extra classical input x. On input (k, x), $G_k(x)$ outputs a distinct pseudorandom state $|\phi_{k,x}\rangle$. Moreover, these generators admit quantum queries in superposition over the input x. It was shown in [AQY22] that there are non-trivial constructions of PRFSs from PRSs, but with a limited output length of the PRFSs. The general question of a black-box construction of PRFSs from PRSs remains open.

Pseudorandom Unitary Defined first in [JLS18], pseudorandom unitaries (PRUs) are unitaries that are indistinguishable from the Haar measure over unitaries, called Haar-random unitaries. It is straightforward to show that PRUs imply PRFSs and PRSs, because if U is a Haar-random unitary, then $U|\phi\rangle$ is a Haar-random state, for any input state $|\phi\rangle$. It was proven very recently that PRUs can be constructed from OWF [MH25]. However, Kretschmer's separation [Kre21] shows that there exists a quantum oracle relative to which PRUs exist, but OWFs do not, making them a weaker assumption than OWFs. The important property about Haar random unitaries, which is also how they are defined, is that they are left and right invariant. This means that if U is drawn from the Haar measure, then for any fixed unitary V, the products UV and VU are also Haar-distributed.

The interest of PRUs for cryptographic applications is somewhat limited, because there are no constructions based on PRUs that cannot already be achieved using PRFSs. All constructions of cryptographic schemes from PRUs are, in fact, done using PRSs. An interesting open question is the following: are there any cryptographic primitives that can be constructed from PRUs but not from PRFSs?

One-Way State Generators One-way state generators (OWSGs), first defined in [MY22b], are the quantum equivalent of one-way functions. Given an input k, the generator outputs a quantum state $|\phi_k\rangle$, and given (polynomially many copies of) the quantum state, no adversary should be able to find the key k. The difference with one-way functions is that the output is a quantum state instead of a classical string. It was shown that PRSs imply OWSGs, and even that PRSs are themselves OWSGs [CGG+23]. The question of whether OWSGs imply PRSs or not is still open. In most constructions from PRSs, the important

property is their one-wayness, and not the fact that the state appears Haar-random. We note that the output state of OWSGs is pure, but there is a variant called IV-OWSGs (for Inefficiently Verifiable One-Way State Generators) introduced in [MY22a], where the state can be a mixed state. IV-OWSGs are separated from OWSGs [BJ24].

EFI pairs Efficiently samplable, statistically far but computationally indistinguishable pairs of quantum states (EFI pairs) [Yan22, BCQ23] are states whose distribution is indistinguishable from a different distribution over quantum states, but are still (statistically) far to this distribution. In the classical setting, the equivalent of EFI have been show to be equivalent to PRG [Gol90]. OWSGs imply EFI pairs [KT24a], but not the other way around [BMM+24, BCN24].

Quantum commitments Quantum schemes, just like public key encryption or signatures, can have quantum commitments (with quantum messages) and quantum openings. Quantum commitments with a classical openings are equivalent to EFI pairs [BCQ23]. Quantum commitments and EFI pairs are considered weak assumptions; they are among the lowest candidates for minimal assumption for quantum cryptography. They imply Multi-Party Computation (MPC) with quantum communications [BCQ23, GLSV21]. In fact, they are even equivalent to MPC. In the classical setting, MPC is unlikely to be constructed from stronger assumptions such as one-way functions, because they are separated from oblivious transfer [IR89], which is believed to be necessary to construct MPC.

1-Pseudorandom Quantum States 1-Pseudorandom Quantum States (1PRSs) are PRSs in which the adversary is given only one copy of the state in the security game. Naturally, PRSs imply 1PRSs, because if they are secure against an adversary that can access a polynomial number of copies of the state, then they are also secure against an adversary that has only one copy of the state. However, it was shown that 1PRSs do not imply PRSs [CCS24, AGL24]. 1PRSs imply EFIs and quantum commitments, but whether the reverse implication holds remains an open question.

One-way Puzzles Unlike the other primitives defined so far, one-way puzzles (OWPs) have a classical input and output. OWPs consists of a generation algorithm, which is a quantum polynomial-time algorithm that outputs a classical puzzle and a classical secret. There is an inefficient verifier, that, given a puzzle and a secret, decides if they were generated by the generation algorithm. The security requirement ensures that no efficient adversary can find the key given the puzzle. One-way puzzles were first defined in [KT24a].

In the classical setting, a one-way puzzle is equivalent to a one-way function, because we can use the randomness of the generation algorithm as an input to construct a one-way function. OWPs are a central primitive in the Quantum Computation Classical Communication (QCCC) model [ACC+22], in which protocols have classical communication, but parties can perform quantum computation. This means that most QCCC primitives imply the existence of OWPs, as shown in [CGG24]. A variant exists, called Efficiently-Verifiable

One-Way Puzzles (EV-OWPs), where the verification algorithm is efficient (i.e. it is a quantum polynomial-time algorithm). This primitive is quite powerful and has been proven to be separated from OWPs [CGG24].

Two recent works [CGGH25, HM24] shows a meta-complexity characterization of OWPs, by proving that their existence is equivalent to the existence of a quantum distribution that can be efficiently sampled from, but for which it is hard to estimate the Kolmogorov complexity.

One-way State Puzzles One-Way State Puzzles (OWSPs) are one-way puzzles in which the secret is a quantum state [KT24b]. There is no verification algorithm, and the security requirement ensures that no adversary can generate a state that is close to the secret quantum state. They have been shown to be equivalent to OWPs [KT24b]. In this work, we define a natural variant, Efficiently-Verifiable One-Way States Puzzles (EV-OWSPs), in which an efficient verification algorithm exists. It was indirectly proven in [GMMY24] that EV-OWSPs are a weaker assumption than EV-OWPs, and we explicitly state this result in this work.

Short Pseudorandom Quantum States Classically, short-output pseudorandom generators ("short-PRGs") and PRGs are equivalent [6], but in the quantum setting, it is unclear how quantum pseudorandomness behaves with different output sizes. Thus, we consider the shortest possible cryptographic PRS, called a short-PRS, which, on input $k \in \{0,1\}^\lambda$ outputs a quantum state of size $n(\lambda) = O(\log \lambda)$. Brakerski and Shmueli [BS20] proved that $c \cdot \log \lambda$-output PRSs exist for any $c \geq 1$, assuming the existence of post-quantum OWFs, whereas there exists a constant $c_0 < 1$ such that for any $c \leq c_0$, $c \cdot \log \lambda$-output PRSs can be constructed unconditionally [7].

One could think that PRSs with a short size would be weaker than PRSs with a long size, but the reality is more nuanced. It was shown in [ALY24] that short-PRSs imply Pseudodeterministic Pseudorandom generators (PD-PRGs), which are almost deterministic PRGs. The security property of PD-PRGs is the same as for PRGs, but unlike PRGs, PD-PRGs are not functions but rather algorithms whose output is identical with probability exponentially close to one, except on a non-negligible fraction of inputs. We can think of PD-PRGs as "almost PRGs", and in fact, they are a powerful primitive, as they are sufficient to construct commitment schemes with classical communication and private-key cryptography [ALY24], as well as digital signatures [BBO+24]. Furthermore, there exists a lifting theorem for constructions from PRGs to constructions from PD-PRGs [BBO+24]: if there is a construction of a scheme from PRGs that makes uniformly random queries to the PRG, then there is a construction of the scheme from PD-PRGs.

In this thesis, we show that PRSs do not imply short-PRSs. Our result also demonstrates

[6] As long as the PRG's output is at least one bit longer than its input, one can compose the PRG with itself, allowing to stretch the output length.

[7] Note that the constant c_0 is not explicitly computed in [BS20].

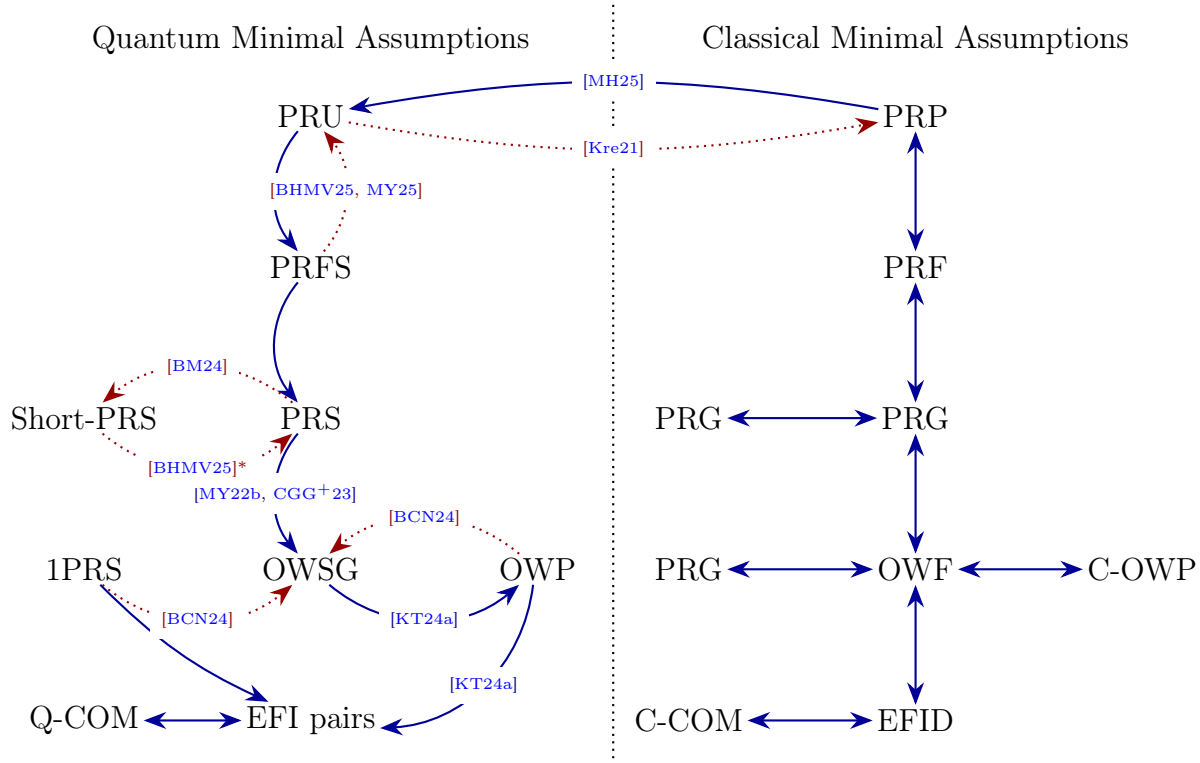


Figure 1.2: Minimal assumptions for classical and quantum cryptography. Here, C-COM refers to classical commitments, and C-OWP refers to classical one-way puzzles. Quantum assumptions are mirrored with their classical equivalent assumption, which makes PRG appear multiple time on the figure. Note that there is no standard definition of one-way puzzles in the classical setting because it is easy to see that they are equivalent to one-way functions.

that PRSs whose output is shorter than that of short-PRSs do not imply short-PRSs, by combining it with a result of [BS20], which showed that such PRSs exist unconditionally. Thus, short-PRSs cannot be constructed from either shorter or longer PRSs. Even more surprisingly, we establish a limited separation of PRSs from short-PRSs, suggesting that they are likely separated and thus incomparable. Hence, in stark contrast to classical pseudorandomness, the output size of quantum pseudorandomness plays a crucial role.

1.4.2 Other Notable Quantum Primitives

Pseudorandom States with Proof of Destruction Pseudorandom states with proof of destruction (PRSPDs) were first defined in [BBSS23]. They are pseudorandom quantum states that possess an additional property: they allow the generation of a string that certifies its destruction. That is, there exists a verifier that will accept this string (along with the key) with high probability. It was shown that PRSPDs can be constructed from OWFs, and can be used to construct various QCCC primitives, including commitment schemes,

one-time signatures and message authentification.

Pseudorandom isometries Pseudorandom isometries (PRIs) [AGKL24] are a generalized notion of pseudorandomness. They are defined as isometries that are computationally indistinguishable from Haar-random isometries. PRIs generalize quantum pseudorandom states (PRSs), as certain pseudorandom isometries can serve as generators for PRSs. It has been shown that PRIs can be constructed from quantum-accessible one-way functions, and can be used to construct all cryptographic primitives that can be built from PRSs, while also possessing additional properties that PRSs lack. For example, PRIs have a stretching property, which allows them to expand the dimension of quantum states in a structured way—something that PRSs do not have.

PRIs have received less attention than weaker assumptions such as PRSs. However, we believe that PRIs could be more practical than PRSs, as their additional structure may make them easier to utilize for constructing cryptographic schemes. One of the key open questions is whether PRIs are separated from OWFs, and for PRIs to be considered a fundamental cryptographic primitive, it would be crucial to show that they are strictly weaker than OWFs.

Unpredictable state generators The concept of unpredictable functions was introduced by Naor and Reingold [NR98]. An unpredictable function ensures that an adversary cannot guess the value $f(x)$ unless it has previously queried f on x. It was proven in [NR98] that unpredictable functions are equivalent to pseudorandom functions (PRFs) in the classical setting. The quantum analogue is Unpredictable State Generators (UPSGs) first defined in [MYY24]. The security definition follows the classical one: an adversary cannot approximate $|\phi_x\rangle$ unless it has previously queried the generation algorithm on x.

It was shown that PRFSs imply UPSGs meaning that PRFSs are at least as strong unpredictable state generators. Moreover, UPSGs imply One-Way State Generators (OWSGs). However, it remains an open question whether UPSGs imply PRFSs, or whether there exists a strict separation between the two. Interestingly, most cryptographic constructions that rely on PRFSs can also be realized using UPSGs instead.

1.4.3 Quantum Worlds

We now introduce quantum worlds, hypothetical models from the literature that capture different assumptions about cryptography in a quantum setting. These mirror Impagliazzo's five classical worlds, but are tailored to quantum cryptography. Moreover, rather than giving formal definitions, we give an intuitive overview, as we believe more perspective is needed before these worlds can be rigorously defined.

Microcrypt Microcrypt is a fundamental world in quantum cryptography. We define Microcrypt as the world where quantum cryptography is possible, but classical cryptography

is not. Classical cryptography failing to exist means that one-way functions do not exist, as they are the minimal assumption for classical cryptography. However, a question remains: what kind of quantum cryptography is possible in Microcrypt? It could be that pseudorandom unitaries (PRUs), pseudorandom function states (PRFSs) or pseudorandom states (PRSs) exist, but it could also be the case that only that EFI pairs exist. Since EFIs are a weaker assumption than these other quantum primitives, one could imagine a world where only the weakest form of quantum cryptography is possible. However, this is not the defining feature of Microcrypt: the crucial property is that quantum cryptography exists, while classical cryptography does not.

Thus, we do not consider the world where only EFIs exist, but neither PRSs nor OWFs exist as Microcrypt, since it aims to categorize quantum versus classical cryptographic feasibility, rather than provide a fine-grained classification of quantum primitives. In essence, Microcrypt should make it clear that *some form* of quantum cryptography is possible, while all classical cryptography is impossible, but without specifying *which* quantum cryptographic primitives exist.

To date, three known relativized worlds in the literature can be considered instances of Microcrypt. We now discuss them in chronological order.

1. **Kretschmer's Oracle World ([Kre21])** The first known instance of Microcrypt was constructed by Kretschmer (2021), where pseudorandom unitaries (PRUs) exist, but $\mathsf{BQP} = \mathsf{QMA}$. The oracle $\mathcal{O}$ consists of two component:

   (a) A quantum oracle $\mathcal{O}_1$, which is a quantum analog of a random function, since it consists of random unitaries. This component ensures that PRUs exist.

   (b) A classical oracle $\mathcal{O}_2$ that ensures that $\mathsf{BQP} = \mathsf{QMA}$.

   Originally, the second oracle was designed to solve a PSPACE-complete problem, but in a revised version, it was replaced with a recursive oracle, similar to those used in diagonalization proofs. The key property, however, remains: this oracle collapses QMA to BQP, ensuring that OWFs do not exist. However, this type of quantum oracle separation is not unanimously accepted in complexity, and a separation with respect to a classical oracle would be a strictly stronger result. Moreover, oracle $\mathcal{O}_1$ can only be queried in a forward direction, meaning that its inverse $\mathcal{O}_1^\dagger$ is inaccessible, which is another restriction. We note that this oracle is the one that is used in proving the impossibility of shrinking PRSs output lengths [BM24, CGG24].

2. **A Classical Oracle World ([KQST23])** The second known Microcrypt world was proposed by Kretschmer, Qian, Sinha and Tal (2023) [KQST23], improving on Kretschmer's oracle. In this world, 1-PRSs exist, but $\mathsf{P} = \mathsf{NP}$, hence one-way functions do not exist. Similar to the previous case, this oracle consists of two components, and their result is based on the hardness of the $\mathsf{OR} \circ \mathsf{FORRELATION}$ problem [AIK22]. We omit a discussion of this problem, as it is beyond the scope of this thesis. The key advantage of this result is that the separation is relative to a classical oracle, making it a stronger separation than Kretschmer's quantum oracle world. However,

the cryptographic primitives that exist in this world are weaker, since only 1-PRSs are known to exist, whereas it is unknown if PRSs exist relative to this oracle.

3. **Quantum vs Classical OWF Separation ([KQT24])** A more recent result by Kretschmer, Qian and Tal (2024) [KQT24] separates classical and quantum-computable one-way functions. However, it is unclear if this world belongs to Microcrypt or Minicrypt. Indeed, the definition of Minicrypt state one-way functions exist in some form. In this world, quantum-computable OWFs exist, which can be interpreted as an indication that "morally", OWFs exist. Furthermore, their oracle is powerful enough to allow for public key encryption. For these reasons, this result is surprising, but its implications are somewhat different from those of previous Microcrypt constructions.

In Microcrypt, many quantum cryptographic primitives can exist that are impossible in Minicrypt, including public key encryption (PKE) with quantum public key and multiparty computation (MPC) with quantum communication.

On the other hand, certain quantum primitives are unlikely to exist in Microcrypt, such as quantum digital (multi-time) signatures, which have been shown to be separated from PRUs [CM24], but also some types of public key encryptions, as demonstrated in [KT24a] and in this thesis. An interesting limitation is the case of short-PRSs, which are linked to the computational assumption that $\mathsf{BQP} \neq \mathsf{QMA}$. Since this assumption is of a similar complexity-theoretic nature than to the existence of OWFs, the feasibility of short-PRSs in Microcrypt is unclear.

Finally, proving that we live in Microcrypt would require proving that OWFs do not exist, which is expected to be difficult to establish.

Nanocrypt The term Nanocrypt was first introduced in [GMMY24], though we present a different definition here. We define Nanocrypt as a world where *weak* quantum cryptography is possible, but *strong* quantum cryptography is impossible. Unlike Microcrypt, which focuses on the separation between quantum and classical cryptography, Nanocrypt exclusively concerns separations between purely quantum primitives. In Nanocrypt, classical cryptography does not exist. We deliberately do not provide an explicit definition of Nanocrypt, as its exact boundaries remain an open question. However, we propose that *strong* quantum primitives include pseudorandom unitaries (PRUs) and pseudorandom states (PRSs). On the other hand,*weak* quantum primitives include one-copy pseudorandom states (1PRSs) and EFI pairs. It is unclear how to classify one-way state generators (OWSGs) as they could be in either of the two categories.

We believe that Nanocrypt will become more precisely defined over time, as our understanding of quantum primitives improves.

Several relativized worlds have been proposed in the literature that exhibit Nanocrypt-like separations. Below, we list some key examples:

- In [BCN24], they exhibit an oracle relative to which 1PRSs exist (and thus EFIs) and OWPs exist, but OWSGs do not (and thus PRSs do not either).
- In [CCS24], they exhibit an oracle relative to which 1PRSs exist, but PRSs do not.
- In [BMM+24], they exhibit an oracle relative to EFI pairs exist[8], but OWSGs do not. They also show a black-box separation between quantum money and EFI pairs.
- In this work, we exhibit an oracle relative to which pseudorandom function states (PRFSs) exist, but PRUs do not.

Proving that we live in Nanocrypt seems at least as hard as proving that $\mathsf{BQP} = \mathsf{PP}$[9]. However, an alternative approach could be proving statements about the existence of quantum primitives under computational assumptions. For example in [KT24b], it was shown that any assumption that imply quantum advantage (such as sampling-base advantage) also imply the existence of quantum cryptography, under the assumption that $\mathsf{P}^{\#\mathsf{P}} \notin (\mathsf{io})\mathsf{BQP}/\mathsf{qpoly}$. This suggests that Nanocrypt might be provable via different complexity-theoretic assumption than $\mathsf{BQP} = \mathsf{PP}$.

Other worlds The definition of Nanocrypt in [GMMY24] is different, where it is the class of primitives that can exist if $\mathsf{BQP} = \mathsf{PP}$. Additionally, [GMMY24] introduces Countcrypt, the class of primitives that are broken if $\mathsf{BQP} = \mathsf{PP}$, but can exist if $\mathsf{BQP} = \mathsf{QCMA}$. They also introduce Quantumania, the class of primitives that are broken if $\mathsf{BQP} = \mathsf{QCMA}$. Another proposal, from [BCN24] is the world Entanglementia, which is the world where the minimum of quantum cryptography is possible, that is quantum commitments.

1.4.4 Idealized Quantum Models

Let us go back a little and discuss why we want a unitary oracle in the quantum random oracle model. We have seen that an oracle implements a function, but we can also define an oracle that implements algorithms. In the classical setting, the difference is that the algorithm can be randomized. However, since randomized (classical) algorithms are equivalent to deterministic algorithms with a random input, this model is equivalent to the deterministic model. Instead of just requiring an input x, the oracle also requires some randomness r.

This is not true in the quantum setting: a quantum algorithm is inherently probabilistic by construction, and there is no way of defining a quantum algorithm as a function. Instead, a quantum algorithm is defined as a quantum channel or completely positive trace-preserving (CPTP) map. This is the most general definition for an algorithm, although an algorithm can also be unitary.

[8] In fact, they show that QEFID pairs, a stronger variant, exist.
[9] It was shown in [Kre21] that PRSs do not exist if $\mathsf{BQP} = \mathsf{PP}$

On oracle separations A discussion on quantum oracle separations and black-box constructions can be found in [CCS24, Section 5.3]. In their discussion, they do not define quantum oracle separations with respect to a CPTP map, but these have been considered in recent literature [GMMY24]. The different kinds of black-box separations can be with respect to a:

1. classical oracle,
2. quantum unitary oracle with access to its inverse,
3. quantum unitary oracle,
4. isometry oracle,
5. CPTP map oracle.

They are ordered from strongest to weakest: for example, a black-box separation with respect to a quantum unitary oracle for primitive A from primitive B rules out any black-box construction of primitive A from any quantum unitary implementation of B, as well as from any isometry or CPTP map implementation of B.

Common Haar function-like state model. All of our separations are based on variants of the Common Haar Function-Like State (CHFS) oracles, where for each input $x \in \{0,1\}^*$ the oracle outputs a Haar random state $|\phi_x\rangle$ of length $\ell(|x|)$, where $|x|$ is the bit-length of x. This is the quantum analog of the Common String Reference (CRS) model, in which every party has access to the same random string. In the quantum setting, the string becomes a quantum state, and it follows the Haar measure instead of the uniform distribution. The CHFS oracle is an isometry, but we can also consider the unitary variants that instantiate this oracle. In a recent work, Goldin and Zhandry [GZ25] prove that some separations can be lifted from the isometry model to the unitary model.

Other separations The isometry version of the CHFS oracle provides a world in which pseudorandom function states (PRFSs) exist but Quantum Computation Classical Communication (QCCC) primitives do not [AGL24]. On the other hand, an oracle world with QCCC key exchange but in which $\mathsf{BQP} = \mathsf{QCMA}$ holds was introduced in [GMMY24], along with additional separations.

1.4.5 Our Results

The landscape of quantum cryptographic pseudorandomness seems quite different from its classical counterparts. We are left with an unsatisfactory state of affairs; unlike in classical pseudorandomness, there is no single assumption that unifies quantum pseudorandomness. In this work, we take steps toward understanding how quantum primitives relate to each other by showing multiple oracle separations.

Impossibility of Shrinking PRSs We show that Kretschmer's oracle [Kre21] not only implies that OWFs do not exist, but also none of the pseudodeterministic variants do either. Since PD-PRGs and PD-OWFs can be constructed from short-PRSs [ALY24, BBO+24], our work provides a separation between PRSs and short-PRSs[10] and can be stated as follows.

Theorem 1.3 (Theorem 5.1, informal)**.** *There exists a quantum oracle $\mathcal{O}$ relative to which PRSs exist but short-PRSs do not.*

This result might sound counterintuitive, as it shows that we cannot shrink a pseudorandom quantum state to a smaller one. An explanation of this result could be that requiring a polynomial number of copies of a logarithmic quantum state to be indistinguishable from a polynomial number of copies of a random state is a strong assumption, akin to that of OWFs, as demonstrated by previous works.

Pseudorandom Quantum States and Quantum Public Key Encryption We show that Kretschmer's oracle can also be used to separate PRUs and qPKE. More formally, we have the following result:

Theorem 1.4. *There exists a quantum oracle $\mathcal{O}$ relative to which PRUs exist but*

1. *quantum public key encryption with classical public key, quantum secret key and classical ciphertext does not exist,*
2. *quantum public key encryption with quantum public key, classical secret key and classical ciphertext does not exist.*

This result complements the other separations and constructions of quantum public key encryption from PRUs in the literature, which we summarize in Section 5.3. We provide a similar analysis on the feasibility of one-time signature and digital signature from PRUs in Section 5.4.

On constructing QPRGs from short PRFSs. We tackle the second question of comparing classical and quantum pseudorandomness. We suggest a candidate oracle, the CHFS oracle with log-length outputs, relative to which short PRFSs exist but QPRGs do not, that is pseudorandom number generators with a quantum algorithm and a negligible correctness error. We prove the separation under a measure-theoretical conjecture with a flavor of isoperimetric inequality; we refer the technical overview section and section 5.6 for an informal and formal descriptions, respectively.

Theorem 1.5. *Assuming conjecture 5.1 is true, there is no black-box way to construct QPRGs with negligible correctness error using short quantum-accessible PRFSs, unless* $\mathsf{BQP} \neq \mathsf{QMA}$.

This result suggests that the black-box construction of QPRGs from short PRFSs is at least as hard as proving $\mathsf{BQP} \neq \mathsf{QMA}$, and rephrases an open problem posed in [ALY24] for

[10] Note that relative to Kretschmer's oracle, not only do we have PRSs, but we also have pseudorandom unitaries (PRUs).

reducing the correctness error, in terms of a measure-theoretical conjecture. Recall that the typical way to construct classical (quantum-computable) primitives from short PRSs uses tomography that incurs inverse polynomial correctness error. For some applications, this error can be dealt with by repetition to construct commitments and encryption [ALY24], or using a recognizable abort to construct signatures [BBO+24]. However, our result indicates that such an inverse polynomial error (e.g. from tomography) is unavoidable.

Length extension of PRSs. We finally turn to the problem of extending the output length of PRSs.[11] We consider a natural restriction on the generation algorithms; that they do not make any partial trace, which we refer to as *pure* algorithms. We prove the following result for the CHFS oracle with $\ell(|x|) = \log |x|$ under the aforementioned conjecture.

Theorem 1.6. *Assuming conjecture 5.1 is true, there exists an isometry oracle relative to which short PRFSs exist but long PRSs with pure generation algorithms do not.*

This result complements our impossibility of shrinking the output length of PRSs, and suggests that both primitives are in fact incomparable. Moreover, given the construction of one-way state generators (OWSGs) from short PRSs [CGG+23], it provides evidence for the hardness of constructing PRSs from OWSGs.

The proof requires new observations on the purity test, i.e., the swap test on two copies, for the state generated by pure quantum algorithms.

1.4.6 Technical Overview

Impossibility of shrinking PRSs Kretschmer's oracle consists of two oracles: the first one is a quantum oracle, which can be interpreted as a quantum version of a random function, since it consists of random unitaries. It is this part of the oracle that ensures the existence of poly-size PRSs. The second one is a classical oracle that ensures that $\mathsf{PromiseBQP} = \mathsf{PromiseQMA}$. In this work, we show that not only does the existence of OWFs imply $\mathsf{PromiseBQP} \neq \mathsf{PromiseQMA}$, but also the existence of polynomial-error pseudodeterministic OWFs (a possibly weaker assumption, but implied by short-PRSs) implies $\mathsf{PromiseBQP} \neq \mathsf{PromiseQMA}$. For the proof to work, we rely on the promise version of the complexity classes. Promise problems are such that there are yes instances and no instances, but also other instances where the output of an algorithm does not matter. In our proof, we define a language with the yes instances as the values for which there exists a *high* probability pre-image, and the no instances being the values for which there is no *low* probability pre-image. Thus, there is a gap between the possible success probabilities, and this gap is needed to distinguish between the yes and no instances in polynomial time.

Separating PRSs and Quantum PKE To show that qPKE with classical public key, quantum secret key and classical ciphertext does not exist relative to Kretschmer's oracle,

[11] We remark that a recent work [LV24] discusses the possibility of the length extension of the PRSs, but only for very specific forms.

we show that their existence implies PromiseBQP $\neq$ PromiseQMA. We note that their existence already implies the existence of OWPs [KT24a], but that is not sufficient to show the separation, as OWPs exist relative to Kretschmer's oracle. We perform a similar analysis for qPKE with quantum public key, classical secret key and classical ciphertext, inspired by the work of [CM24]. In this paper, they show the impossibility of digital signatures with quantum public key from PRUs, and require multiple signatures for the proof to go through; similarly, for our proof to work, we require multiple ciphertexts.

(Unitarized) Common Haar function-like state oracles and PRFSs. All of our other results are in relativized world with the common Haar function-like state (CHFS) oracles. The CHFS oracles with length ℓ is defined as follows: it is a family of unitaries $\{S_x\}_{x\in\{0,1\}^*}$ defined the following way:

$$S_x : \begin{cases} |0\rangle \to |\phi_x\rangle \\ |\phi_x\rangle \to |0\rangle \\ |\psi\rangle \to |\psi\rangle \quad \text{if } |\psi\rangle \notin \mathsf{span}(|0\rangle, |\phi_x\rangle), \end{cases}$$

where $|\phi_x\rangle$ is a predetermined Haar random state of length $\ell(|x|)$, with $|x|$ denoting the bit-length of x. This oracle is inspired by the reflection/swap oracles in [CCS24, BCN24, BMM+24].

In this overview, we assume that the algorithm accesses the unitaries S_x one by one, and also assume that $\langle 0|\phi_x\rangle = 0$ for simplicity, so that S_x can be understood as a reflection

$$S_x = I - 2\,|\phi_x-\rangle\langle\phi_x-|\,,$$

where $|\phi-\rangle = \frac{|0\rangle-|\phi_x\rangle}{\sqrt{2}}$.[12]

The construction of PRFSs with the CHFS oracles is rather straightforward: the generation algorithm on input (k, x) for key k and input x outputs $\left|\phi_{k||x}\right\rangle$ by querying $S_{k||x}$, where $k||x$ is the concatenation of k and x. The security can be shown by a standard hybrid argument. Note that the output length of the PRFSs is $\ell(k||x)$.

Candidate separation of QPRGs from short PRFSs We conjecture that relative to the CHFS oracles with length $\ell(n) = \log n$, short PRFSs exist but QPRGs with negligible correctness error do not; we will simply denote QPRGs with negligible correctness by QPRGs from now on. Once again, given the CHFS oracle the existence of short PRFSs is immediate, thus we need to argue about the non-existence of QPRGs.

Concentration inequality fails. The concentration inequality of Haar measures (see theorem 2.11) is the most standard tool currently used for separation arguments. However, this concentration inequality is not strong enough to deal with small dimensional qubits, thereby it is hard to use to rule out QPRGs.

[12]We have a slightly different definition in the main body.

We instead observe an extreme concentration case that must happen in QPRGs: consider the single-bit-output QPRG G^O, relative to CHFS oracles O, with negligibly small correctness error. For a fixed input x, $G^O(x)$ must be either 0 or 1 for almost all oracles O; these are the two extreme points in the concentration inequality. A natural question is thus whether these two extreme points can be simultaneously concentrated.

Impossibility of QPRGs from new conjecture. We start from this point: if a function $f(O) = \Pr_G[G^O(x) \to 1]$ from quantum states to $[0, 1]$ has two highly concentrated points near 0 and 1, how do the regions $f^{-1}([0, \varepsilon])$ and $f^{-1}([1 - \varepsilon, 1])$ of the state space look like? If $G^O(x)$ for a fixed x can output both 0 and 1 with non-zero probability, both pre-image regions are large. We also expect the distance between the two pre-image regions to be large, as close oracles would likely induce close outputs. Our conjecture asserts that under such conditions, the intermediate region $f^{-1}((\varepsilon, 1 - \varepsilon))$ is large:

Conjecture *(Informal version of conjecture 5.1).* Let X be the product space of pure quantum states with the corresponding product Haar measure σ. If S_0, S_1 are two measurable subsets of X such that $\sigma(S_0), \sigma(S_1) \geq A$, and if $d(S_0, S_1) \geq B$ for some distance d on X, then $\sigma(X \setminus (S_0 \cup S_1)) \geq \mathsf{poly}(A, B)$.

We can cast this conjecture in a purely geometric way, with the flavor of an isoperimetric inequality. For example, in the extreme case of one of the regions being small and the other one large (as in the isoperimetric inequality), the conjecture states that the Δ-gap region between the surfaces is still large. We inspected some cases, which indeed follow this intuition. We refer to appendix C for some more details.

Now we turn back to the QPRGs G^O with negligible correctness error. Again, for convenience, we assume that G^O outputs a single bit and let $f(O) = \Pr_G[G^O(x) \to 1]$. It rules out the case where $f^{-1}([0, \varepsilon])$ and $f^{-1}([1 - \varepsilon, 1])$ are both large. That is, $G^O(x)$ must be either 0 or 1, regardless of O! This means that from G^O we can derive QPRGs without querying O, without any assumption. This is impossible to construct unless $\mathsf{BQP} \neq \mathsf{QMA}$, and theorem 1.5 follows.

Length extension of PRSs, without partial trace We again consider the CHFS oracle with $\ell(|x|) = \lfloor \log |x| \rfloor$ together with the $\mathsf{QPSPACE}$ oracle. Here, we consider the classical-accessible isometry version: a family of unitaries $\{O_\lambda\}_\lambda$ where O_λ takes a λ-bit classical string $|x\rangle$ as input and outputs $|x\rangle |\phi_x\rangle$, where $|\phi_x\rangle$ is a Haar random state stored in a new register. Alternatively, we write

$$O_\lambda = \sum_{x \in \{0,1\}^\lambda} |x\rangle\langle x|_{\mathbf{X}} \otimes |\phi_x\rangle\langle 0|_{\mathbf{Y}} \,,$$

and assume that the register $\mathbf{Y}$ is initialized to $|0\rangle$ and never touched before the query. The existence of short PRS(F)Gs is shown using the same argument as the long case setting for $\ell(n) = n$.

We remark that the separation in the CHS model [CCS24] assumes non-adaptive queries to the oracle without loss of generality. This can be done because there is only a linear number of oracles. As we have exponentially many oracles, we cannot make queries to all of them. We must consider the adaptive queries, which introduce numerous technical difficulties.

Pure generation algorithm. We consider the following form of PRSs generation algorithm G, without any partial trace:

$$\rho = U_t \circ O_{\lambda_t} \circ \mathcal{M}_t \circ \cdots \circ U_1 \circ O_{\lambda_1} \circ \mathcal{M}_1 \circ U_0(|0\rangle\langle 0|_{\mathbf{X}_1\mathbf{Y}_1\ldots\mathbf{X}_t\mathbf{Y}_t\mathbf{Z}}),$$

where we omit the key k for simplicity. Here, the U_i are unitary operators for $i = 0, \ldots, t$, O_{λ_i} are the oracle queries acting on the registers $\mathbf{X}_i\mathbf{Y}_i$ for $i = 1, \ldots, t$, and $\mathcal{M}_i$ are intermediate measurements on $\mathbf{X}_i$ used to decide which state is generated by the oracle. We use $\circ$ to denote the composition of operators.[13] We assume that this algorithm acts on n-qubit for $n(\lambda) = \omega(\log \lambda)$. This form of generation algorithm reflects adaptive queries, making it hard to analyze.

Step 1: To non-adaptive form using purity test. The output of the PRSs generation algorithm must be indistinguishable from Haar random state, hence it must be close to pure. This intuition can be formalized by the swap test on two copies that estimating the purity $\mathrm{Tr}(\rho^2)$.

Our main technical tool here is that if a state ρ generated by a pure algorithm passes this test with high probability, then all the intermediate measurements must be almost deterministic, i.e., there exist $x_1, \ldots, x_t$ such that

$$\rho \approx U_t \circ O_{\lambda_t} \circ |x_t\rangle\langle x_t|_{\mathbf{X}_t} \circ \cdots \circ U_1 \circ O_{\lambda_1} \circ |x_1\rangle\langle x_1|_{\mathbf{X}_1} \circ U_0(|0\rangle\langle 0|_{\mathbf{X}_1\mathbf{Y}_1\ldots\mathbf{X}_t\mathbf{Y}_t\mathbf{Z}}).$$

Then, using the fact that the registers $\mathbf{Y}$'s are not touched before queries, we can argue that

$$\rho \approx \tilde{U}\left(|0\rangle_{\mathbf{X}_1\ldots\mathbf{X}_t\mathbf{Z}} |\phi_{x_1}\rangle_{\mathbf{Y}_1} \cdots |\phi_{x_t}\rangle_{\mathbf{Y}_t}\right),$$

for some unitary $\tilde{U}$. The actual argument is more complicated to derive *unitary* $\tilde{U}$, which we omit in this overview.

We remark that *both* parts become problematic when we consider the algorithms with partial traces. We are only able to show that the last few measurements are almost deterministic, and we do not have any clue to use a similar argument for turning algorithms to unitaries.

Step 2: Product test with quantum OR lemma. We previously omitted the key k, but we consider them here again. We can assume that the generation algorithm on key k outputs

$$\rho_k \approx \tilde{U}_k\left(|0\rangle_{\mathbf{X}_1\ldots\mathbf{X}_t\mathbf{Z}} \left|\phi_{x_1^{(k)}}\right\rangle_{\mathbf{Y}_1} \cdots \left|\phi_{x_t^{(k)}}\right\rangle_{\mathbf{Y}_t}\right),$$

[13] Because of the measurements, we need to consider mixed states so that the operator U on ρ is acted by $U\rho U^\dagger$.

where we assume that t and the lengths of $x_i^{(k)}$'s are all the same for different keys for exposition. Alternatively, we have

$$\tilde{U}_k^\dagger \rho_k \approx |0\rangle_{\mathbf{X}_1 \dots \mathbf{X}_t \mathbf{Z}} \left|\phi_{x_1^{(k)}}\right\rangle_{\mathbf{Y}_1} \dots \left|\phi_{x_t^{(k)}}\right\rangle_{\mathbf{Y}_t},$$

is a product of many pure states. On the other hand, for a Haar random state $|\psi\rangle$, $\tilde{U}_k^\dagger |\psi\rangle$ definitely does not have such a product structure. Given the efficient product test algorithm [HM10], we can run the quantum OR tester with the QPSPACE oracle as in the separation between PRUs and PRFSs.

Missing step: Learning $\lambda_1, \dots, \lambda_t$ based on the conjecture. In the above algorithm, we assume that $\lambda_1, \dots, \lambda_t$ are decided a priori. However, these may be also determined by the intermediate measurements.

At this point, we recall the implication of the conjecture: if a quantum algorithm with access to the short CHFS oracle O outputs a fixed bit with high probability, then this bit is likely independent from O. Therefore, we can apply the same strategy to learn the results of the intermediate measurements. This allows the algorithm to fix $\lambda_1, \dots, \lambda_t$ a priori, filling the missing step.

1.4.7 Related Works and Dicussions

Output size of quantum primitives There is much evidence that the output length of quantum pseudodeterministic primitives is fundamental and must be chosen carefully. For example, in [HMY24] they show that $O(\log \lambda)$ EFIs and OWSGs do not exist, while they show how to construct $\omega(\log \lambda)$ EFIs from a classical oracle. In [CGG+23] they show that $\omega(\log \lambda)$ OWSGs exist unconditionally. This is in spark contrast with classical primitives such as OWFs and PRGs, whose output length can be extended or shrunk.

Concurrent work. Our result on the impossibility of shrinking PRSs was proven independently in [CGG24]. In their paper, they study One Way Puzzles (OWPs), and show a black-box separation between general OWPs and efficiently verifiable OWPs. Their result also builds on Kretschmer's oracle, and implies a separation between short-PRSs and PRSs, because PRSs imply OWPs and short-PRSs imply efficiently verifiable OWPs. A concurrent work by Coladangelo and Mutreja [CM24] separating quantum digital signatures from PRSs also provides a separation between short-PRSs and PRSs, since short-PRSs imply quantum digital signatures as proven by Barhoush et al. [BBO+24].

A concurrent and independent work [MY25] shows the oracle separation between PRFSs and PRUs using similar oracles and techniques. They use the maximally entangled state $\sum_x |x, x\rangle$ to simplify the analysis. The results concerning the log-length CHFS oracles are unique to this work.

Chapter 2

Preliminaries

2.1 Notations

The following notations will be used throughout the manuscript,

- By λ we denote the security parameter.
- For any $m \in \mathbb{N}$, we use the notation $[m]$ to refer to the set $\{1, \ldots, m\}$.
- For a bit string $x \in \{0,1\}^*$, we denote its bit-length by $|x|$.
- We use $\mathsf{poly}(\cdot)$ to denote a polynomially-bounded function. We use $\mathsf{negl}(\cdot)$ to denote a negligible function. A negligible function is a function who is asymptotically smaller than any inverse of polynomial, i.e. for every polynomial $\mathsf{poly}(\cdot)$, there exists an $N \in \mathbb{N}$ such that for any $n > N$, $\mathsf{negl}(n) \leq \frac{1}{\mathsf{poly}(n)}$.
- We use calligraphic letters (e.g., $\mathcal{X}$) to denote sets. We use $\mathcal{Y}^{\mathcal{X}}$ to denote the set of all functions from $\mathcal{X}$ to $\mathcal{Y}$.
- We use bold letters (e.g., $\mathbf{m}$) to denote random variables and distributions. We write $m \leftarrow\!\!\$\, \mathbf{m}$ to denote that m is sampled from the distribution $\mathbf{m}$. We write $m \leftarrow\!\!\$\, \mathcal{M}$ to denote that m is sampled uniformly from the set $\mathcal{M}$.
- We use the Dirac notation for pure states, e.g., $|\psi\rangle$, while mixed states will be denoted by lowercase Greek letters, e.g., ρ.
- We use ε to denote the empty string.
- We use $||$ to denote the concatenation operator.
- We use $x \prec y$ to denote the fact that x is a prefix of y, i.e. there exists x' such that $y = x||x'$. We use $x \not\prec y$ to denote the fact that x is not a prefix of y.

- For $n \in \mathbb{N}$ and $N = 2^n$, we write $\mathbb{S}(N)$ and $\mathbb{U}(N)$ to denote the set of N-dimensional pure quantum states and the group of $N \times N$ unitary matrices. We denote by σ_n and μ_n the Haar distribution over n-qubit states and n-qubit unitaries, i.e., over $\mathbb{S}(2^n)$ and $\mathbb{U}(2^n)$, respectively. When the dimension is clear from the context, we drop the parameter and use σ or μ.

For the basics of quantum computation, we refer readers to [NC10], and for completeness, we recall Grover's algorithm and Quantum Fourier Transform (QFT) in Appendix A.

We assume that all functions used to represent the lengths of the cryptographic primitives are computable in Quantum Polynomial Time (QPT). We will also use standard notations from quantum information and cryptography.

2.2 Cryptography

We include here the relevant cryptographic notions.

Definition 2.1 (One-way functions)**.** *Let $\{0,1\}^{n(\lambda)}$ be the output length, a function $f : \{0,1\}^{\lambda} \to \{0,1\}^{n(\lambda)}$ is a one-way function (OWF) if the following holds.*

1. ***Efficient generation**. There exists an polynomial time algorithm that given $x \in \{0,1\}^{\lambda}$, computes $y = f(x)$.*
2. ***One-wayness**. For any polynomial time algorithm $\mathcal{A}$, we have that:*

$$\Pr_{x \in \{0,1\}^{\lambda}}[A(f(x)) = x] \leq \mathsf{negl}(\lambda).$$

We also include pseudo deterministic quantum one-way function (PD-QOWF), whose definition is adapted from [BBO+24, Definition 9] and are algorithms whose output is one-way, and is always the same with probability negligible close to one for a fraction of the inputs.

Definition 2.2 (Pseudo Deterministic Quantum One-Way Functions)**.** *Let $n(\lambda)$ be the output length, a QPT algorithm $F : \{0,1\}^{\lambda} \to \{0,1\}^{n(\lambda)}$ is a* pseudo deterministic quantum one-way function *(PD-QOWF) if the following conditions hold:*

- ***Pseudodeterminism**. There exists a constant $c > 0$ and a function $\mu(\lambda) = O(\lambda^{-c})$ such that for all $\lambda \in \mathbb{N}$, there exists a set $\mathcal{K}_\lambda \subset \{0,1\}^{\lambda}$ such that:*
  1. $\Pr\left[x \in \mathcal{K}_\lambda \,\middle|\, x \leftarrow \{0,1\}^{\lambda}\right] \geq 1 - \mu(\lambda)$.
  2. *For any $x \in \mathcal{K}_\lambda$, it holds that*

$$\max_{y \in \{0,1\}^{n(\lambda)}} \Pr[y = F(x)] \geq 1 - \mathsf{negl}(\lambda), \tag{2.1}$$

  where the probability is over the randomness of F.

- ***Security***. *For every QPT inverter $\mathcal{A}$:*

$$\Pr_{x\leftarrow\{0,1\}^\lambda}\left[F\left(\mathcal{A}(F(x))\right)=F(x)\right]\leq \mathsf{negl}(\lambda), \tag{2.2}$$

where the probability is over the randomness of F and $\mathcal{A}$.

Note that the pseudodeterminism factor in the above definition comes from the size of the *good* key space $\mathcal{K}_\lambda$, which is the whole space minus an inverse-polynomial fraction of the space. This means that for a non-negligible number of elements in the key space (the elements in $\{0,1\}^\lambda\setminus\mathcal{K}_\lambda$), the OWF could behave arbitrarily. Also note that the lower bound in the success probability in Equation (2.3) could be replace by $\frac{2}{3}$, as it can be amplified with repetition (a standard completeness amplification technique).

We define quantum one-way functions (QOWF) where the size of the good key space $\mathcal{K}_\lambda$ is the whole space minus a negligible fraction of the space, i.e. $\mu(\lambda)=\mathsf{negl}(\lambda)$.

We also define pseudo deterministic quantum pseudorandom generators (PD-QPRGs), who are algorithms whose output is indistinguishable from random, and is always the same with probability negligibly close to one for a fraction of the inputs.

Definition 2.3 (Pseudo Deterministic Quantum Pseudorandom Generators)**.** *Let $n(\lambda)$ be the output length, a QPT algorithm F is a* pseudo deterministic quantum pseudorandom generator *(PD-QPRG) if the following conditions hold:*

- ***Pseudodeterminism***. *There exists a constant $c>0$ and a function $\mu(\lambda)=O(\lambda^{-c})$ such that for all $\lambda\in\mathbb{N}$, there exists a set $\mathcal{K}_\lambda\subset\{0,1\}^\lambda$ such that:*

  1. $\Pr\left[x\in\mathcal{K}_\lambda\;\middle|\;x\leftarrow\{0,1\}^\lambda\right]\geq 1-\mu(\lambda)$.

  2. *For any $x\in\mathcal{K}_\lambda$, it holds that*

$$\max_{y\in\{0,1\}^{n(\lambda)}}\Pr\left[y=F(x)\right]\geq 1-\mathsf{negl}(\lambda), \tag{2.3}$$

where the probability is over the randomness of F.

- ***Security***. *For any oracle QPT algorithm $\mathcal{A}=\{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$, there exists a negligible function ε such that*

$$\left|\Pr_{y\leftarrow\{0,1\}^{n(\lambda)}}\left[1\leftarrow\mathcal{A}_\lambda(y)\right]-\Pr_{x\leftarrow\{0,1\}^\lambda}\left[1\leftarrow\mathcal{A}_\lambda(F(x))\right]\right|\leq\varepsilon(\lambda),$$

where the probability is over the randomness of F and $\mathcal{A}_\lambda$.

- ***Length extension.*** *$n(\lambda)>\lambda$ holds for all $\lambda\in\mathbb{N}$.*

Similarly, we define pseudorandom generators (QPRG) where the size of the good key space $\mathcal{K}_\lambda$ is the whole space minus a negligible fraction of the space, i.e. $\mu(\lambda)=\mathsf{negl}(\lambda)$.

Definition 2.4 (Pseudorandom quantum states [JLS18])**.** *Let $n(\lambda)$ be the number of qubits in the quantum system. A keyed family of n-qubit quantum states $\{|\varphi_k\rangle\}_{k\in\{0,1\}^\lambda}$ is* pseudorandom *if the following two conditions hold:*

1. ***Efficient generation.*** *There is a quantum polynomial time algorithm G that on input $k \in \{0,1\}^\lambda$ generates*
$$G_\lambda(k) = |\varphi_k\rangle\langle\varphi_k| .$$
2. ***Pseudorandomness.*** *For any quantum polynomial time adversary $\mathcal{A}$ and all polynomials $t(\cdot)$, we have*
$$\left| \Pr_{k\leftarrow\{0,1\}^\lambda}\left[\mathcal{A}\left(1^\lambda, |\varphi_k\rangle^{\otimes t(\lambda)}\right)=1\right] - \Pr_{|\nu\rangle\leftarrow\sigma_n}\left[\mathcal{A}\left(1^\lambda, |\nu\rangle^{\otimes t(\lambda)}\right)=1\right]\right| \leq \mathsf{negl}(\lambda).$$

We say that Gen is a $n(\lambda)$-PRS to indicate that its output length is $n(\lambda)$. We further say that a PRS is a *short PRS* when its output length is $\Theta(\log\lambda)$, and a *(long) PRS* when its output length is $\omega(\log\lambda)$.

We by default consider the adaptively-secure PRFSs defined as follows.

Definition 2.5 (Pseudorandom Function-like State Generators)**.** *We say that a QPT algorithm Gen is a secure pseudorandom function-like state generator (PRFS) if the following holds for some functions $\kappa, m, n : \mathbb{N} \to \mathbb{N}$ such that $\kappa, m = \omega(\log\lambda)$:*

- ***State Generation:*** *For any $\lambda \in \mathbb{N}$ and $k \in \{0,1\}^{\kappa(\lambda)}$, the algorithm Gen_k takes as input $x \in \{0,1\}^{m(\lambda)}$ and outputs $n(\lambda)$-qubit (possibly mixed) state $\mathsf{Gen}_k(x)$ stored in a new register.*
- ***Pseudorandomness:*** *For any QPT adversary $\mathcal{A} = \{\mathcal{A}_\lambda\}_{\lambda\in\mathbb{N}}$, there exists a negligible function $\varepsilon(\cdot)$ such that for all $\lambda \in \mathbb{N}$:*
$$\left| \Pr_{k\leftarrow\{0,1\}^\lambda}\left[1 \leftarrow \mathcal{A}_\lambda^{\mathsf{Gen}(k,\cdot)}\right] - \Pr_{G_{\mathsf{Haar}}}\left[1 \leftarrow \mathcal{A}_\lambda^{G_{\mathsf{Haar}}(\cdot)}\right]\right| \leq \varepsilon(\lambda),$$
where $G_{\mathsf{Haar}}(\cdot)$ on input $x \in \{0,1\}^{m(\lambda)}$, output $|\psi_x\rangle$ stored in a new register, where, for every $y \in \{0,1\}^{m(\lambda)}$, $|\psi_y\rangle \leftarrow \mathcal{H}_{n(\lambda)}$.

We say that Gen is a $(\kappa(\lambda), m(\lambda), n(\lambda))$-PRFS to indicate that its key length is $\kappa(\lambda)$, it input length is $m(\lambda)$, and its output length is $n(\lambda)$. We say that a PRFS is a short PRFS *when $n = \Theta(\log\lambda)$, and a* (long) PRFS *when $n = \omega(\log\lambda)$.*

Definition 2.6 (Pseudorandom unitaries [JLS18])**.** *Let $n(\lambda)$ be the dimension of the quantum system. A keyed family of n-qubit unitary operator $\{U_k\}_{k\in\{0,1\}^\lambda}$ is* pseudorandom *if the following two conditions hold:*

1. ***Efficient generation.*** *There is a quantum polynomial time algorithm G that on input $k \in \{0,1\}^\lambda$ and $|\phi\rangle$, generates*
$$G_\lambda(k, |\phi\rangle) = U_k |\phi\rangle .$$

2. ***Pseudorandomness***. *For any quantum polynomial time adversary $\mathcal{A}$ and all polynomials $t(\cdot)$, we have*

$$\left| \Pr_{k \leftarrow \{0,1\}^\lambda} \left[\mathcal{A}^{U_k} \left(1^\lambda\right) = 1 \right] - \Pr_{U \leftarrow \mu_n} \left[\mathcal{A}^{U} \left(1^\lambda\right) = 1 \right] \right| \leq \mathsf{negl}(\lambda).$$

We now define public key encryption.

Definition 2.7 (Public key encryption (PKE))**.** *A quantum public key encryption scheme (PKE) consists of three algorithms:*

- $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$: *a quantum algorithm, which takes as input the security parameter and output a couple of keys* $(\mathsf{pk}, \mathsf{sk})$.
- $\mathsf{c} \leftarrow \mathsf{Enc}(\mathsf{pk}, m)$*: a quantum algorithm, which takes as input a public key* pk*, a classical message* m*, and outputs a ciphertext* c.
- $m/\perp \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{c})$*: a quantum algorithm, which takes as input a secret key* sk*, a ciphertext* c*, and outputs a classical plaintext* m *or a distinguished symbol* $\perp$ *indicating decryption failure.*

We say that a PKE scheme is *correct* if for every message $m \in \{0,1\}^\lambda$ and any security parameter $\lambda \in \mathbb{N}$, the following holds:

$$\Pr\left[\mathsf{Dec}(\mathsf{sk}, \mathsf{c}) = m \;\middle|\; \begin{array}{c} (\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda) \\ \mathsf{c} \leftarrow \mathsf{Enc}(\mathsf{pk}, m) \end{array} \right] \geq 1 - \mathsf{negl}(\lambda),$$

where the probability is taken over the randomness of KGen, Enc and Dec.

We emphasize that the public key pk, the secret key sk and the ciphertext c can be either a classical string or a quantum state. Thus there is in total eight different types of quantum public key encryption, the strongest being with classical public key, secret key and ciphertext, and the weakest being with quantum public key, secret key and ciphertext. Looking ahead, we use c to denote a classical string, and q to denote a quantum state. We let $(\mathsf{pk}, \mathsf{sk}, \mathsf{c}) \in \{c, q\}^3$ and define $(\mathsf{pk}, \mathsf{sk}, \mathsf{c})$-PKE where $\mathsf{pk} = c$, $\mathsf{sk} = c$ and $\mathsf{c} = c$ indicate that the public key, secret key and ciphertext are classical strings, respectively. Similarly, $\mathsf{pk} = q$, $\mathsf{sk} = q$ and $\mathsf{c} = q$ indicate that the public key, secret key and ciphertext are quantum states, respectively. For example, a (c, c, q)-PKE scheme is a public key encryption where the public key and secret key are classical strings, while the ciphertext is a quantum state.

In the case where the public key is a quantum state, we assume that there also exists an algorithm $\mathsf{QPKGen}(\cdot)$ that given the (classical or quantum) secret key, output a public key that satisfies correctness, i.e.

$$\Pr\left[\mathsf{Dec}(\mathsf{sk}, \mathsf{c}) = m \;\middle|\; \begin{array}{c} (\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda) \\ \mathsf{pk}' \leftarrow \mathsf{QPKGen}(\mathsf{sk}) \\ \mathsf{c} \leftarrow \mathsf{Enc}(\mathsf{pk}', m) \end{array} \right] \geq 1 - \mathsf{negl}(\lambda).$$

We now give a very weak security notion for public-key encryption. Because our goal is to prove impossibility result for constructing PKE under this notion from PRUs, we will also establish separation for stronger security notions. In the security definition, the adversary receives polynomially many distinct message-ciphertext pairs and succeeds if, given the encryption of a new, distinct message, it can recover that message.

Definition 2.8. *A public-key encryption scheme is secure if for all adversary $\mathcal{A}$, there exist a negligible function* $\mathsf{negl}(\cdot)$ *such that for all polynomials* $\mu(\cdot)$*,*

$$\Pr\left[m^* = m_0 \;\middle|\; \begin{array}{c} (\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda) \\ \forall 0 \le i \le \mu(\lambda), m_i \leftarrow\!\!\$\ \{0,1\}^\lambda \setminus \{m_j | 0 \le j \le i-1\} \\ \forall 0 \le i \le \mu(\lambda), \mathsf{c}_i \leftarrow \mathsf{Enc}(\mathsf{pk}, m_i) \\ m^* \leftarrow \mathcal{A}(\mathsf{c}_0, m_1, \mathsf{c}_1, \ldots, m_{\mu(\lambda)}, \mathsf{c}_{\mu(\lambda)}) \end{array}\right] \le \mathsf{negl}(\lambda),$$

We now define signature schemes.

Definition 2.9 (Signature scheme)**.** *A signature scheme consists of three algorithms:*

- $(\mathsf{vk},\mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$: *a quantum algorithm, which takes as input the security parameter and output a couple of keys* $(\mathsf{vk},\mathsf{sk})$*.*
- $\mathsf{s} \leftarrow \mathsf{Sign}(\mathsf{sk}, m)$*: a quantum algorithm, which takes as input a signing key* sk*, a message* m*, and outputs a signature* s*.*
- $\top/\bot \leftarrow \mathsf{Ver}(\mathsf{vk},\mathsf{s},m)$*: a quantum algorithm, which takes as input a verification key* vk*, a signature* s*, a message* m *and outputs a distinguished symbol* $\bot$ *indicating failure or a distinguished symbol* $\top$ *indicating success.*

We say that a signature scheme is *correct* if for every message $m \in \{0,1\}^\lambda$ and any security parameter $\lambda \in \mathbb{N}$, the following holds:

$$\Pr\left[\mathsf{Ver}(\mathsf{vk},\mathsf{s},m) = \bot \;\middle|\; \begin{array}{c} (\mathsf{vk},\mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda) \\ \mathsf{s} \leftarrow \mathsf{Sign}(\mathsf{sk}, m) \end{array}\right] \ge 1 - \mathsf{negl}(\lambda),$$

where the probability is taken over the randomness of KGen, Sign and Ver.

As for quantum public key encryption, we emphasize that the verification key vk, the secret key sk and the signature s can be either a classical string or a quantum state. Thus there is in total eight different types of signatures, the strongest being with classical verification key, secret key and signature, and the weakest being with quantum verification key, secret key and signature. We let $(\mathsf{vk},\mathsf{sk},\mathsf{s}) \in \{c,q\}^3$ and define $(\mathsf{vk},\mathsf{sk},\mathsf{s})$ signatures where $\mathsf{vk} = c$, $\mathsf{sk} = c$ and $\mathsf{s} = c$ indicate that the verification key, the secret key and the signature is a classical string respectively. Similarly, $\mathsf{vk} = q$, $\mathsf{sk} = q$ and $\mathsf{s} = q$ indicate that the verification key, the secret key and the signature is a quantum state, respectively. For example, a c,c,q signature scheme is a signature where the verification key and the secret key are classical strings, while the signature is a quantum state.

We now define the security definition of one-time signature, unforgeability.

Definition 2.10. *A signature scheme is one-time secure if for every quantum polynomial time adversary $\mathcal{A}$, there is a negligible function* $\mathsf{negl}(\cdot)$ *such that the following holds:*

$$\Pr\left[\begin{array}{c|c} \mathsf{Ver}(\mathsf{vk},\mathsf{s}^*,m^*)=\bot & (\mathsf{vk},\mathsf{sk})\leftarrow\mathsf{Gen}(1^\lambda) \\ \wedge \quad m\neq m^* & m\leftarrow\!\!\$\ \{0,1\}^* \\ & \mathsf{s}\leftarrow\mathsf{Sign}(\mathsf{sk},m) \\ & (\mathsf{s}^*,m^*)\leftarrow\mathcal{A}(\mathsf{vk},\mathsf{s},m) \end{array}\right]\leq\mathsf{negl}(\lambda),$$

where the probability is taken over the randomness of KGen, Sign, Ver, $\mathcal{A}$ *and the choice of* $m\leftarrow\!\!\$\ \{0,1\}^*$.

Similarly, we define the security notion for digital signature, in which the adversary receives polynomially many message-signatures pairs and must forge a valid signature on a new, distinct message.

We give two definitions of efficiently verifiable one-way puzzles, the first one is the usual one where the puzzle and the secret are classical, and the second have been recently introduced in [KT24b], where the secret is a quantum state. The latter are called state puzzles, and here we define efficiently verifiable one, where there is an efficient algorithm that tell you if the secret is valid for the puzzle or not.

Definition 2.11 (Efficiently Verifiable One-way Puzzles [KT24a])**.** *An efficiently verifiable one-way puzzle (EV-OWP) is a pair of sampling and verification algorithms* $(\mathsf{Gen},\mathsf{Ver})$ *with the following syntax.*

- $\mathsf{Gen}(1^\lambda)\to(\mathsf{k},\mathsf{s})$*, is a QPT algorithm that outputs a pair of classical strings* (k,s)*. We refer to* k *as the puzzle and* s *as its key.*
- $\mathsf{Ver}(\mathsf{k},\mathsf{s})\to\top/\bot$*, is a Boolean function that maps every pair of classical strings* (k,s) *to either* $\top$ *or* $\bot$*.*

These satisfy the following properties.

- **Correctness.** *Outputs of the sampler pass verification with overwhelming probability, i.e.,*
$$\Pr_{(\mathsf{k},\mathsf{s})\leftarrow\mathsf{Gen}(1^\lambda)}[\mathsf{Ver}(\mathsf{k},\mathsf{s})=\top]=1-\mathsf{negl}(\lambda).$$
- **Security.** *Given* k*, it is (quantum) computationally infeasible to find* s *satisfying* $\mathsf{Ver}(\mathsf{k},\mathsf{s})=\top$*, i.e., for every quantum polynomial-sized adversary* $\mathcal{A}$*,*
$$\Pr_{(\mathsf{k},\mathsf{s})\leftarrow\mathsf{Gen}(1^\lambda)}[\mathsf{Ver}(\mathsf{k},\mathcal{A}(\mathsf{k}))=\top]=\mathsf{negl}(\lambda).$$

The following is adapted from the definition of one-way state puzzle in [KT24b].

Definition 2.12 (Efficiently Verifiable One-way State Puzzles)**.** *An efficiently verifiable one-way state puzzle (EV-OWSP) is a pair of sampling and verification algorithms* $(\mathsf{Gen},\mathsf{Ver})$ *with the following syntax.*

- $\mathsf{Gen}(1^\lambda) \to (\mathsf{k}, |\mathsf{s}\rangle)$, *is a QPT algorithm that outputs a classical string* k *and a quantum state* $|\mathsf{s}\rangle$. *We refer to* k *as the puzzle and* s *as its key.*
- $\mathsf{Ver}(\mathsf{k}, |\mathsf{s}\rangle) \to \top/\bot$, *is an algorithm that takes as input a classical string* k *and a quantum state* $|\mathsf{s}\rangle$, *and output either* $\top$ *or* $\bot$.

These satisfy the following properties.

- **Correctness.** *Outputs of the sampler pass verification with overwhelming probability, i.e.,*
$$\Pr_{(\mathsf{k},|\mathsf{s}\rangle)\leftarrow\mathsf{Gen}(1^\lambda)}[\mathsf{Ver}(\mathsf{k}, |\mathsf{s}\rangle) = \top] = 1 - \mathsf{negl}(\lambda).$$
- **Security.** *Given* k, *it is (quantum) computationally infeasible to find* $|\mathsf{s}\rangle$ *satisfying* $\mathsf{Ver}(\mathsf{k}, |\mathsf{s}\rangle) = \top$, *i.e., for every quantum polynomial-sized adversary* $\mathcal{A}$,
$$\Pr_{(\mathsf{k},|\mathsf{s}\rangle)\leftarrow\mathsf{Gen}(1^\lambda)}[\mathsf{Ver}(\mathsf{k}, \mathcal{A}(\mathsf{k})) = \top] = \mathsf{negl}(\lambda).$$

Remark 2.1. *All of these primitives can be defined relative to an oracle* $\mathcal{O}$, *and when that is the case, all the algorithm in the definition of the primitive can query the oracle, including the adversary of the security definition. In the security definition of quantum pseudorandomness primitives, when the adversary makes non-adaptive queries to* G *or* $\mathcal{U}$, *we say that* G *is non-adaptively secure. When the adversary always measure the input register before making queries to* G *or* $\mathcal{U}$, *we say that* G *is classical-accessible. Otherwise, we say that its quantum-accessible.*

2.3 Compressed Oracle Technique

We now present the key ingredients of Zhandry's compressed oracle technique, first defined in [Zha19] and refined in [CFHL21]. As mentioned in the introduction, the technique uses a register to keep a record of a so-called *database* of the random oracle and this register is updated whenever an adversary A makes a query to the random oracle. This new register that contains the database is at the gist of our lower bounds.

We consider the *Quantum Random Oracle Model*, first defined in [BDF+11]. In this model, we are given black-box access to a *random* function $H : \mathcal{X} \to \mathcal{Y}$. For our model, the adversary will work on three different registers $|x, y, z\rangle$. The first register is the query register, the second register is the answer register and the third register is the work register. The first two registers are used for queries and answers to the oracle, while the last register is for the adversary's other computations. We first define the unitary StO that represents the *Standard Oracle* and that computes as follows:

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

This unitary corresponds to a query to H.

Now, we define Zhandry's compressed oracle. In this model, instead of starting with a random function H, we start with the uniform superposition of all random functions $|H\rangle$, where $|H\rangle$ encodes the truth table of the function H. In this model, there is a register for each $x \in \mathcal{X}$, and the value of this register in the state $|H\rangle$ corresponds to $H(x)$. That is, we have that $|H\rangle = \bigotimes_{x\in\mathcal{X}} |H(x)\rangle_x$ Let $\mathcal{H} = \{H : \mathcal{X} \to \mathcal{Y}\}$ be the set of all possible functions H. We define a new register, the database register $|H\rangle$, that starts in the uniform superposition $\frac{1}{|\mathcal{H}|}\sum_{H\in\mathcal{H}} |H\rangle$. This register starts in product state with the other registers, and Zhandry's idea is that instead of modifying the adversary's register when querying the oracle, we will modify the database register instead. To do so, we simply consider the *Fourier basis* for the y and the H register before querying the Standard Oracle.

We write this unitary O and it works as follows:

$$\mathsf{O}\sum_{x,\hat{y},z} \alpha_{x,\hat{y},z} |x,\hat{y},z\rangle \otimes \sum_{\hat{H}\in\mathcal{H}} \alpha_{\hat{H}} \left|\hat{H}\right\rangle \to \sum_{x,\hat{y},z} \alpha_{x,\hat{y},z} |x,\hat{y},z\rangle \otimes \sum_{\hat{H}\in\mathcal{H}} \alpha_{\hat{H}} \left|\hat{H} \ominus (x,\hat{y})\right\rangle,$$

where, for any fixed $x \in \mathcal{X}$ and $z \in \mathcal{Y}$, $H \ominus (x,z) : \mathcal{X} \to \mathcal{Y}$ is defined as:

$$H \ominus (x,z)(x') = \begin{cases} H(x') & \text{if } x' \neq x \\ H(x) - z & \text{if } x' = x. \end{cases}$$

In other words, $H \ominus (x,z)$ is obtained by replacing the value of $H(x)$ by $H(x) - z$ in H.

This unitary can be implemented by applying the QFT to the registers $|y\rangle$ and $|H\rangle$, applying the Standard Oracle, then applying the $QFT^\dagger$ again on the $|y\rangle$ and $|H\rangle$ registers.

Finally, we define the compression part. The idea behind the compression is that for every x in the database mapped to $\left|\hat{0}\right\rangle$, we remap it to $|\perp\rangle$, where $\perp$ is a new value outside of $\mathcal{Y}$. More formally, the compression part is done by applying:

$$\mathsf{Comp} = \bigotimes_x \left(|\perp\rangle \left\langle\hat{0}\right| + \sum_{\hat{y}:\hat{y}\neq\hat{0}} |\hat{y}\rangle \langle\hat{y}| \right)$$

in the Fourier basis.

Since at the start of the computation, the database will be initiated with the uniform superposition over all $\mathcal{H}$ possible, then after q queries the state of the database can be described with q vectors. In order to apply the compression as a unitary, we declare that $\mathsf{Comp}\,|\perp\rangle = |0\rangle$.

Now, we can define the *Compressed Oracle*:

$$\mathsf{cO} = \mathsf{Comp} \circ \mathsf{O} \circ \mathsf{Comp}^\dagger.$$

Of course the compression part inevitably creates some losses, compared to only using the Standard Oracle. The precise characterization of these losses is given in one of Zhandry's lemma, and can be stated as follows:

Lemma 2.1 (Lemma 5 from [Zha19]). *Let A be an algorithm that makes queries to a random oracle $H : \mathcal{X} \to \mathcal{Y}$, and output $(x_1, \ldots, x_k, y_1, \ldots, y_k) \in \mathcal{X}^k \times \mathcal{Y}^k$. Let p be the probability that $\forall 1 \leq i \leq k$, $H(x_i) = y_i$. Similarly, consider the algorithm A running with the Compressed Oracle cO, and output $(x'_1, \ldots, x'_k, y'_1, \ldots, y'_k) \in \mathcal{X}^k \times \mathcal{Y}^k$. Let p' be the probability that $\forall 1 \leq i \leq k$, $H'(x'_i) = y'_i$, where H' is obtained by measuring the H register at the end of the execution of the algorithm A. Then:*

$$\sqrt{p} \leq \sqrt{p'} + \sqrt{\frac{k}{|\mathcal{Y}|}}$$

In Chapter 3, we will have that $\sqrt{\frac{k}{|\mathcal{Y}|}}$ is negligible, and thus we will neglect this term.

We also have the following lemma from [CFHL21] that describes the operator $\mathsf{cO}_{(x,\hat{y})} : \mathcal{H} \to \mathcal{H}$, which is defined as the operator applied on $|H\rangle$ when applying cO to $|x\rangle |\hat{y}\rangle \otimes |H\rangle$. More formally, we have that:

$$\mathsf{cO} |x\rangle |\hat{y}\rangle \otimes |H\rangle = |x\rangle |\hat{y}\rangle \otimes \mathsf{cO}_{(x,\hat{y})} |H\rangle$$

Lemma 2.2 (Lemma 4.3 from [CFHL21]). *For any $\hat{y} \neq \hat{0}$, the operator $\mathsf{cO}_{(x,\hat{y})}$ is represented by the following matrix:*

	$\perp$	r
$\perp$	0	$\frac{\omega_N^{-ry}}{\sqrt{\lvert\mathcal{Y}\rvert}}$
y'	$\frac{\omega_N^{yy'}}{\sqrt{\lvert\mathcal{Y}\rvert}}$	$\begin{cases} \left(1 - \frac{2}{\lvert\mathcal{Y}\rvert}\right)\omega_N^{yy'} + \frac{1}{\lvert\mathcal{Y}\rvert} & \text{if } y' = r \\ \frac{1-\omega_N^{yy'}-\omega_N^{ry}}{\lvert\mathcal{Y}\rvert} & \text{if } y' \neq r \end{cases}$

For $\hat{y} = \hat{0}$, we have that $\mathsf{cO}_{(x,\hat{0})}$ is the identity.

We also define, for any compressed $H : \mathcal{X} \to \mathcal{Y} \cup \{\perp\}$, for any fixed $x \in \mathcal{X}$ and $z \in \mathcal{Y}$, $H \cup (x, z) : \mathcal{X} \to \mathcal{Y}$ as:

$$H \cup (x,z)(x') = \begin{cases} H(x') & \text{if } x' \neq x \\ z & \text{if } x' = x. \end{cases}$$

In other words, $H \cup (x, z)$ is obtained by replacing the value of $H(x)$ by z in H.

In the following, we will model the adversary (A) as a series of computation alternating between unitaries and oracle calls. The adversary's quantum state will first be initialized to $|0\rangle^{\otimes N}$. Then, his computation will be decomposed as:

$$A = U_k \mathsf{cO} U_{k-1} \mathsf{cO} \ldots \mathsf{cO} U_2 \mathsf{cO} U_1 \tag{2.4}$$

So that, if $|\psi_i\rangle = \sum_{x,y,z,D} \alpha_{x,y,z,D} |x,y,z,D\rangle$ is the state of the adversary after i quantum queries to cO, then U_{i+1} operates on the registers x, y and z only. We also define *database properties*:

Definition 2.13 (Database property)**.** *A database property is a subset of $\mathcal{H}$. Any database property D can be seen as a projector on $\mathcal{H}$, as follows:*

$$\sum_{d \in D} |d\rangle \langle d|$$

We write $\mathcal{D} = \{I | I \subseteq \mathcal{H}\}$ the set of all subspaces of $\mathcal{H}$, that also corresponds to the set of all database properties.

We now state and prove two lemmas adapted from [LZ19] that we will use thoroughly in Chapter 3. The first lemma will allow us to ignore the unitaries that the adversary A applies on the first registers of the state.

Lemma 2.3 (adapted from Lemma 8 from [LZ19])**.** *For any unitary U, any projector P, and any state $|\phi\rangle$,*

$$|(I \otimes P) \cdot (U \otimes I) |\phi\rangle| = |(I \otimes P) |\phi\rangle|$$

The second lemma bounds the amplitude of measuring a database that satisfies a property P at the i^{th} step of the algorithm, i.e. just after the i^{th} query to the oracle. In this bound, the first term captures the case where we succeed to find a database that satisfies P before the i^{th} query. The second term captures the case where we did not have it before the i^{th} query, but found it with the i^{th} one.

Lemma 2.4 (adapted from Lemma 9 from [LZ19])**.** *Let $|\phi_i\rangle$ be the state of an algorithm A just before the i^{th} quantum query to cO, and $|\psi_i\rangle$ the state of the same algorithm right after the i^{th} quantum query to cO. Let P be any projector on D. We have that:*

$$|P |\psi_i\rangle| \leq |P |\phi_i\rangle| + |P\mathsf{cO}(I - P) |\phi_i\rangle|$$

Proof.

$$\begin{aligned} |P |\psi_i\rangle| = |P\mathsf{cO} |\phi_i\rangle| &= |P\mathsf{cO}(P |\phi_i\rangle + (I - P) |\phi_i\rangle)| \\ &\leq |P |\phi_i\rangle| + |P\mathsf{cO}(I - P) |\phi_i\rangle)| , \end{aligned}$$

where the inequality comes from the triangle inequality and the fact that $P\mathsf{cO}P \leq P$. □

Remark 2.2. *In the next section and in Chapter 3, we will consider multiple functions $h_1, \ldots, h_k : \mathcal{X} \to \mathcal{Y}$ for some fixed k. Note that this is equivalent to considering one function $H : \mathcal{X} \to \mathcal{Y}^k$, such that we interpret, for any $x \in \mathcal{X}$, the output $H(x)$ as the concatenation of values of the functions applied to x, i.e. $H(x) = h_1(x)||h_2(x)||\cdots||h_k(x)$. Hence, in this setting, the compressed oracle is used on the function H, and a query to any of the h_i is a query to all of the h_i's. Thus, in our results, we count the number of queries to the function H and thus the number of queries to all of the h_i's. It may seem that we lose some accuracy in this setting, however this is with the same method that multiple random functions are implemented in the literature.*

2.4 The Problem of Subset Cover and Its Variants

We define the problem of subset cover.

Definition 2.14 ((r,k)–SC)**.** *Let $k, r \in \mathbb{N}^*$. Let $h_1, \cdots, h_k : \mathcal{X} \to \mathcal{Y}$. A (r,k)–SC for $(h_1, \cdots, h_k)$ is a set of $r+1$ elements $x_0, x_1, x_2, \cdots, x_r$ in $\mathcal{X}$ such that:*

$$\{h_i(x_0) | 1 \leq i \leq k\} \subseteq \bigcup_{j=1}^{r} \{h_i(x_j) | 1 \leq i \leq k\}$$

In other words, for each $1 \leq i \leq k$, there exists a $1 \leq j \leq r$ and a $1 \leq \ell \leq k$ such that $h_i(x_0) = h_\ell(x_j)$.

We notice two facts regarding the parameters of (r,k)–SC. First, we have that the problem becomes easier when r increases. Secondly, we have that when $r > k$, a (r,k)–SC contains a (k,k)–SC. Thus finding a (r,k)–SC when $r > k$ is the same as when $r = k$. For simplicity, we use k–SC as a shorthand of (k,k)–SC.

We also define the database properties $P^{SC}_{(r,k)}$ of containing a (r,k)–SC, that is the set of databases that contains a (r,k)–SC. More formally, we have that:

$$P^{SC}_{(r,k)} = \left\{ D \in \mathcal{D} \middle| \exists x_0, x_1, \ldots, x_r, \forall i \neq 0, x_0 \neq x_i, H(x_0) \subseteq \bigcup_{i=1}^{r} H(x_i) \right\},$$

where for $x \in \mathcal{X}, H(x) = \{h_1(x), \ldots, h_k(x)\}$.

We follow now with the definition of a harder variation of the k–subset cover called the *k–restricted subset cover* (k–RSC).

Definition 2.15 (k–RSC)**.** *Let $k \in \mathbb{N}^*$. Let $h_1, \ldots, h_k : \mathcal{X} \to \mathcal{Y}$. A k–*restricted subset cover *(k–RSC) for $(h_1, \ldots, h_k)$ is a set of $k+1$ elements $x_0, x_1, x_2, \ldots, x_k$ in $\mathcal{X}$ such that:*

$$\forall i \in \{1, \ldots, k\}, h_i(x_0) = h_i(x_i) \text{ and } x_0 \neq x_i.$$

We also define the database properties $P^{RSC}_{k,\ell}$ of k distinct ℓ–RSC, that is the set of databases that contains k distinct ℓ–RSC. More formally, we have that:

$$P^{RSC}_{k,\ell} = \left\{ D \in \mathcal{D} \middle| \begin{array}{l} \exists x_{0,1}, \ldots, x_{\ell,1}, \forall i \neq 0, x_{0,1} \neq x_{i,1}, \forall i, h_i(x_{0,1}) = h_i(x_{i,1}) \\ \exists x_{0,2}, \ldots, x_{\ell,2}, \forall i \neq 0, x_{0,2} \neq x_{i,2}, \forall i, h_i(x_{0,2}) = h_i(x_{i,2}) \\ \vdots \\ \exists x_{0,\ell}, \ldots, x_{\ell,k}, \forall i \neq 0, x_{0,k} \neq x_{i,k}, \forall i, h_i(x_{0,k}) = h_i(x_{i,k}) \\ \forall i \neq j, (h_1(x_{0,i}), \ldots, h_\ell(x_{0,i})) \neq (h_1(x_{0,j}), \ldots, h_\ell(x_{0,j})) \end{array} \right\} \tag{2.5}$$

The problem of finding a k–RSC was introduced in [YTA22], in which the authors describe an algorithm that finds a k–RSC in $O\left(kN^{\frac{1}{2}\left(1-\frac{1}{2^{k+1}-1}\right)}\right)$ quantum queries to $h_1, \ldots, h_k$ when the h_i's are such that $|\mathcal{X}| \geq (k+1)|\mathcal{Y}|$.

We discuss now the last condition in Equation (2.5). We remark that while such condition was not explicitly imposed in [LZ19] for their lower bound for finding multi-collisions, this property is implicitly and extensively used in their proof. Such a property is needed because when they count k–collisions (that is, k distinct $x_1, \dots, x_k$ such that $H(x_1) = \cdots = H(x_k)$), they are actually interested in the number of possible *images* that would be helpful to reach a $(k+1)$–collision. In particular, this is helpful since one query can only transform *one* k–collision (with such a property) into a $(k+1)$–collision.

In our case, the last line of (2.5) ensures that the "supporting set" of the k–RSC (i.e. the set of images of the $x_{0,i}$ by the different random functions $h_1, \dots, h_k$) is unique. As in the multi-collision case, this condition will be crucial to extend a k–RSC to a $(k+1)$–RSC, and for this reason we define it explicitly in $P^{RSC}_{k,\ell}$.

Finally, we state a result from [LZ19], regarding the amplitude of finding j distinct 2–collisions:

Lemma 2.5 (adapted from [LZ19], Corollary 11)**.** *Given a random function $h : \mathcal{X} \to \mathcal{Y}$ where $|N| = \mathcal{Y}$, let $f^{col}_{i,j}$ be the amplitude of the D containing at least j distinct 2–collisions after i quantum queries. Then:*

$$f^{col}_{i,j} \leq \left(\frac{4e \cdot i^{3/2}}{j\sqrt{N}} \right)^j .$$

For completeness, the proof of Lemma 2.5 is given in Appendix B.1. The proof closely follows the proof of Corollary 11 in [LZ19] but we need to consider some negligible factors that incur an extra constant factor in the statement.

2.5 Quantum Computation

Definition 2.16 (Oracle-aided quantum algorithms)**.** *A quantum algorithm $\mathcal{A}$ is a family of quantum circuits $\mathcal{A} \coloneqq \{A_\lambda\}_{\lambda \in \mathbb{N}}$ that act on three sets of registers: input registers $\mathbf{X}$, output registers $\mathbf{Y}$, and work registers $\mathbf{Z}$. For convenience, we let $\mathbf{W} \coloneqq (\mathbf{X}, \mathbf{Y}, \mathbf{Z})$ denote the* internal registers *of $\mathcal{A}$. For each input $x \in \{0,1\}^\lambda$, the output is computed by running the algorithm A_λ on $|x\rangle_{\mathbf{X}} |0\rangle_{\mathbf{Y}} |0\rangle_{\mathbf{W}}$ and at the end the output registers are measured in the computational basis to obtain the output.*

A d-query quantum oracle algorithm $\mathcal{A}^h$ that has access to an oracle h, defined by the unitary $\mathcal{O}_h$ can be specified by a sequence of unitary matrices $(U_d, U_{d-1}, \dots, U_0)$. The final state of the algorithm is defined as $U_d \mathcal{O}_h U_{d-1} \mathcal{O}_h \dots \mathcal{O}_h U_0 |x\rangle_{\mathbf{X}} |0\rangle_{\mathbf{Y}} |0\rangle_{\mathbf{Z}}$. When the oracle h implements some classical function $h : \mathcal{X} \to \mathcal{Y}$, the corresponding query operator $\mathcal{O}_h$ is defined as $|x\rangle_{\mathbf{X}} |y\rangle_{\mathbf{Y}} \mapsto |x\rangle_{\mathbf{X}} |y \oplus h(x)\rangle_{\mathbf{Y}}$.

When $\mathcal{A}^h$ is clear from the context, we omit the superscript h and write $\mathcal{A}$.

The following preliminary is borrowed from the formalization of [ACC+22].

Definition 2.17 (The computational and the Fourier basis)**.** *Let $\mathcal{Y}$ be a finite Abelian group with cardinality $|\mathcal{Y}|$. Let $\{|y\rangle\}_{y\in\mathcal{Y}}$ be an orthonormal basis of $\mathbb{C}^{|\mathcal{Y}|}$. We refer to this basis as the computational basis. Let $\hat{\mathcal{Y}}$ be the dual group which is known to be isomorphic to $\mathcal{Y}$. Recall that a member $\hat{y}\in\hat{\mathcal{Y}}$ is a character function (i.e., a function from $\mathcal{Y}$ to the multiplicative group of non-zero complex numbers). The Fourier basis $\{|\hat{y}\rangle\}_{\hat{y}\in\hat{\mathcal{Y}}}$ of $\mathbb{C}^{|\mathcal{Y}|}$ is defined as*

$$|\hat{y}\rangle = \frac{1}{\sqrt{|\mathcal{Y}|}}\sum_{y\in\mathcal{Y}}\hat{y}(y)^\dagger\,|y\rangle \quad \textit{and} \quad |y\rangle = \frac{1}{\sqrt{|\mathcal{Y}|}}\sum_{\hat{y}\in\hat{\mathcal{Y}}}\hat{y}(y)\,|\hat{y}\rangle\,.$$

Definition 2.18 (Functions and their (quantum) representations)**.** *For any function $h\in\mathcal{Y}^{\mathcal{X}}$, we define its quantum representation to be $|h\rangle_H := \bigotimes_{x\in\mathcal{X}}|h(x)\rangle_{H_x}$ in the computational basis, where the register H_x is associated with $\mathbb{C}^{\mathcal{Y}}$ for all $x\in\mathcal{X}$, and the register H is compounded of all H_x. Similarly, for any $\hat{h}\in\hat{\mathcal{Y}}^{\mathcal{X}}$ we define $\left|\hat{h}\right\rangle_H := \bigotimes_{x\in\mathcal{X}}\left|\hat{h}(x)\right\rangle_{H_x}$ in the Fourier basis.*

Zhandry [Zha19] shows that the purified random oracle is perfectly indistinguishable from the (standard) quantum random oracle, and thus instead of considering the query operator $\mathcal{O}_h$, we can consider another equivalent query oracle $\mathcal{O}$ acting on three registers $\mathbf{X},\mathbf{Y},\mathbf{H}$ as follows.

$$|x\rangle_{\mathbf{X}}\,|y\rangle_{\mathbf{Y}}\,|h\rangle_{\mathbf{H}} \mapsto |x\rangle_{\mathbf{X}}\,|y\oplus h(x)\rangle_{\mathbf{Y}}\,|h\rangle_{\mathbf{H}}\,,$$

where the oracle register H is initialized as $|\Phi_0\rangle_H = \sum_{h\in\mathcal{H}}\frac{1}{\sqrt{|\mathcal{H}|}}|h\rangle_H$.

Note that in the Fourier basis, the unitary $\mathcal{O}$ acts as follows:

$$|x\rangle_X\,|\hat{y}\rangle_Y\left|\hat{h}\right\rangle_H \mapsto |x\rangle_X\,|\hat{y}\rangle_Y\bigotimes_{x'\in\mathcal{X}}\left|\hat{h}(x')-\delta_{x,x'}\cdot\hat{y}\right\rangle_H,$$

where $\delta_{x,x'}$ is equal to 1 if $x=x'$, and 0 otherwise, and the oracle register H is initialized as $|\Phi_0\rangle_H = \bigotimes_{x\in\mathcal{X}}\left|\hat{0}\right\rangle_{H_x}$.

Definition 2.19 (Purified view of two-party protocols in the QROM)**.** *A two-party protocol in the Quantum-Computation Classical-Computation (QCCC) model is a protocol in which two quantum algorithms, Alice and Bob, can query the oracle, apply quantum operation on their internal registers, and send classical strings over the public (authenticated) channel to the other party. The sequence of the strings sent during the protocol is called the* transcript *of the protocol. Let $\mathbf{W}_A$ and $\mathbf{W}_B$ be Alice's and Bob's internal registers, respectively. Let $\mathcal{H}:=\mathcal{Y}^{\mathcal{X}}$. For any two-party protocol, we define its purified version as follows.*

- *If the protocol is inputless, start with $|0\rangle_{\mathbf{W}_A}|0\rangle_{\mathbf{W}_B}\sum_{h\in\mathcal{H}}\frac{1}{\sqrt{|\mathcal{H}|}}|h\rangle_{\mathbf{H}}$. Otherwise, if Alice takes as input a classical string $a\in\mathcal{X}$ and Bob takes as input a classical string $b\in\mathcal{X}$, start with $|a\rangle_{\mathbf{W}_A}|b\rangle_{\mathbf{W}_B}\sum_{h\in\mathcal{H}}\frac{1}{\sqrt{|\mathcal{H}|}}|h\rangle_{\mathbf{H}}$.*
- *Alice and Bob run the protocol in superposition, that is, all the measurements (including those used for generating the transcript) are delayed and the query operator $\mathcal{O}_h$ is replaced by $\mathcal{O}$.*

- *Let* $|\Psi\rangle_{\mathbf{W}_A\mathbf{W}_B\mathbf{H}}$ *denote the state at the end of the protocol, and let* $|\Psi_t\rangle_{\mathbf{W}_A\mathbf{W}_B\mathbf{H}}$ *denote the post-measurement state of* $|\Psi\rangle_{\mathbf{W}_A\mathbf{W}_B\mathbf{H}}$ *which is consistent with the transcript* t*.*

We now define some properties related to this new register for the database $|h\rangle_H$.

Definition 2.20 (Non-zero queries in Fourier basis)**.** *Let* $\mathcal{Y}$ *be a finite Abelian group and* $\hat{\mathcal{Y}}$ *be the dual group. For any* $\hat{y} \in \hat{\mathcal{Y}}^{\mathcal{X}}$*, we define the size of* $\hat{h}$ *to be*

$$\left|\hat{h}\right| \coloneqq \left|\{x : x \in \mathcal{X}, \hat{h}(x) \neq \hat{0}\}\right|.$$

Definition 2.21 (Oracle support)**.** *Let* $\hat{\mathcal{H}} \coloneqq \hat{\mathcal{Y}}^{\mathcal{X}}$*. For any vector* $|\phi\rangle_{\mathbf{WH}} = \sum_{w,\hat{h}\in\hat{\mathcal{H}}} \alpha_{w,\hat{h}} |w\rangle_{\mathbf{W}} \left|\hat{h}\right\rangle_{\mathbf{H}}$*, we define the* oracle support in the Fourier basis *of* $|\phi\rangle$ *as*

$$\widehat{\mathrm{supp}}^H(|\phi\rangle) \coloneqq \left\{\hat{h} : \exists w \text{ s.t. } \alpha_{w,\hat{h}} \neq 0\right\}.$$

We denote $\hat{h}^H_{max}(|\phi\rangle)$ *the function* $\hat{h} \in \widehat{\mathrm{supp}}^H(|\phi\rangle)$ *that has the largest size* $\left|\hat{h}\right|$ *(if such function is not unique, by default we pick the lexicographically first one). The definition extends naturally when the register* $\mathbf{W}$ *does not exist.*

Similarly, if we write the oracle part in the computational basis $|\phi\rangle_{\mathbf{WH}} = \sum_{w,h\in\mathcal{H}} \beta_{w,h} |w\rangle_{\mathbf{W}} |h\rangle_{\mathbf{H}}$*, then we define the oracle support in the computational basis of* $|\phi\rangle$ *as*

$$\mathrm{supp}^H(|\phi\rangle) \coloneqq \{h : \exists w \text{ s.t. } \beta_{w,h} \neq 0\}.$$

Definition 2.22. *A partial oracle* L *is a partial function from* $\mathcal{X}$ *to* $\mathcal{Y}$*. The domain of* L *is denoted by* $Q_L \coloneqq dom(L)$*. Equivalently, we view* L *as a finite set of pairs* $(x, y_x) \in \mathcal{X} \times \mathcal{Y}$ *such that for all* $(x, y_x), (x', y'_x) \in L, x \neq x'$*. We say a partial oracle* L *is consistent with* $h : \mathcal{X} \to \mathcal{Y}$ *if and only if* $h(x) = y_x$ *holds for all* $x \in Q_L$*.*

For any partial oracle L*, we define the associated projector* Π_L *by*

$$\Pi_L \coloneqq \bigotimes_{x\in Q_L} |y_x\rangle\langle y_x|_{H_x} \bigotimes_{x\notin Q_L} \mathcal{I}_{H_x},$$

where $\mathcal{I}_{H_x}$ *is the identity operator acting on* H_x*. It holds that* $\Pi_L |h\rangle_H = |h\rangle_H$ *if* h *is consistent with* L*, and* $\Pi_L |h\rangle_H = 0$ *otherwise.*

Lemma 2.6. *If* A *asks at most* d *queries to the superposition oracle, then for all possible outcomes of* A*'s intermediate measurements, the joint state* $|\psi\rangle_{WH}$ *conditioned on the outcome satisfies* $\left|\hat{h}^H_{max}(|\psi\rangle)\right| \leq d$*.*

Lemma 2.7. *Given a state* $|\psi\rangle_{WH}$ *and a partial oracle* L*, the state* $\Pi_L |\psi\rangle_{WH}$ *can be written as*

$$\Pi_L |\psi\rangle_{WH} \coloneqq \sum_{w\in\mathcal{W},\hat{h}\in\hat{\mathcal{H}}'} \alpha'_{w,\hat{h}} |w\rangle_W \bigotimes_{x\notin Q_L} \left|\hat{h}(x)\right\rangle_{H_x} \bigotimes_{x\in Q_L} |y_x\rangle_{H_x},$$

where $\hat{\mathcal{H}}'$ *is the set of functions from* $\mathcal{X} \setminus Q_L$ *to* $\hat{\mathcal{Y}}$*. Furthermore, if* $\left|\hat{h}^H_{max}(|\psi\rangle)\right| \leq d$*, then* $\left|\hat{h}^{H'}_{max}(\Pi_L |\psi\rangle)\right| \leq d$*, where* H' *is the set of registers corresponding to* $\mathcal{X} \setminus Q_L$*.*

2.6 Quantum-Heavy Queries Learner

We now define the *quantum-heavy queries learner* algorithm. It was first defined in [ACC+22, Construction 4.10], which can be seen as the quantum counterpart of the classical *independence learner* of [BM09], where Eve learns all the ε-*heavy queries* of both Alice and Bob.

Definition 2.23 (Quantum ε-heavy queries [ACC+22, Definition 4.9])**.** *For $x \in \mathcal{X}$, define the projector*

$$\Pi_x \coloneqq \sum_{\hat{y} \in \hat{\mathcal{Y}} \setminus \{\hat{0}\}} |\hat{y}\rangle\langle\hat{y}|_{\mathbf{H}_x} .$$

Given a quantum state $|\psi\rangle_{\mathbf{W}_A \mathbf{W}_B \mathbf{H}}$, the weight of any $x \in \mathcal{X}$ is defined as

$$w(x) \coloneqq \|\Pi_x |\psi\rangle\|^2,$$

that is, the quantum heaviness of x is the probability of obtaining a non-$\hat{0}$ outcome while measuring $\mathbf{H}_x$ in the Fourier basis. We call $x \in \mathcal{X}$ a quantum ε-heavy *query if $w(x) \geq \varepsilon$.*

Construction 2.1 (Quantum-heavy queries learner [ACC+22])**.** *Let (A, B) be an inputless two-party QCCC protocol relative to a random oracle h, in which Alice and Bob make at most d quantum queries to the oracle. Given the transcript t, (computationally-unbounded) attacking algorithm* Eve *is parameterized by ε and works as follows.*

1. *Let L denote the set of oracle query-answer pairs obtained by* Eve *from the oracle, and $\mathcal{Q}_L$ is defined similarly while only containing the queries. Initially prepare $L = \emptyset$ and the classical description of the state*

$$|\psi\rangle_{W'_{\mathsf{A}} W'_{\mathsf{B}} H'} = |0\rangle_{W'_{\mathsf{A}}} |0\rangle_{W'_{\mathsf{B}}} |\Phi_0\rangle_{H'} ,$$

   where $|\Phi_0\rangle$ is a uniform superposition over all $h \in \mathcal{H}$, W'_{A} , W'_{B} and H' are the simulated registers for Alice, Bob, and the oracle prepared by Eve.

2. *Simulate the state evolution during the protocol. Concretely,* Eve *calculates the state in $W'_{\mathsf{A}} W'_{\mathsf{B}} H'$ after each round in the protocol. Whenever* Eve *encounters the moments in which Alice (Bob) sends their message,* Eve *calculates the post-measurement state that is consistent with t.*

3. *While there is any query $x \notin \mathcal{Q}_L$ that is quantum ε-heavy conditioned on (t, L), do the following:*

   (a) *Ask the lexicographically first quantum ε-heavy query x from the real oracle h.*

   (b) *Update the state in $W'_{\mathsf{A}} W'_{\mathsf{B}} H'$ to the post-measurement state that is consistent with $(x, h(x))$.*

   (c) *Update L by adding $(x, h(x))$ to L.*

4. *When there is no quantum ε-heavy query left to ask,* Eve *outputs the simulated quantum state $|\psi_t\rangle_{W'_{\mathsf{A}} W'_{\mathsf{B}} H'}$ and her list L, conditioned on the transcript t.*

Remark 2.3. *We note that construction 2.1 described above is almost identical to [ACC+22, Construction 4.10]. The only difference is that* Eve *outputs the simulated state, which can be constructed from the classical description that* Eve *has computed, along with the list of queries she made to the oracle.*

The technical properties of the quantum-heavy queries learner in construction 2.1 are stated in the following lemma.

Lemma 2.8 ([ACC+22])**.** *For any $0 < \varepsilon < 1$, the quantum-heavy queries learner described in construction 2.1 satisfies the following properties:*

- ***Efficiency:*** *$\mathbb{E}[|L|] \leq \frac{d}{\varepsilon}$, where the expectation is over the randomness of the oracle and the algorithm* Eve.
- ***Security:*** *When the learner stops and learns a list L, there is no $x \in \mathcal{Q}_L$ that is ε-quantum heavy in the purified view of* Eve *conditioned on knowing L and the transcript t.*

2.7 Polynomial Compatibility Conjecture

In this section, we recall the Polynomial Compatibility Conjecture (PCC) of [ACC+22]. The formulation we use here is based on quantum states.

Definition 2.24 ($(\mathcal{Y}, \delta, d, N)$-state [ACC+22, Definition 4.1])**.** *Let H be a register over the Hilbert space $\mathcal{Y}^N$. A quantum state $|\psi\rangle$ over registers $\mathbf{W}$ and $\mathbf{H}$ is a* $(\mathcal{Y}, \delta, d, N)$-state *if it satisfies the following two conditions:*

- *d-**sparsity**: $\left|\hat{h}^H_{max}(|\psi\rangle)\right| \leq d$. This means that for any measurement of registers H in the Fourier basis, and W in any basis, the oracle support in the Fourier basis is at most d.*
- *δ-**lightness**: For every $x \in \mathcal{X}$, if we measure the $\mathbf{H}_x$ register of $|\psi\rangle$ in the Fourier basis, the probability of getting $\hat{0}$ is at least $1 - \delta$. This mean that $|\psi\rangle$ has no δ-heavy queries.*

Definition 2.25 (Compatible states [ACC+22, Definition 4.2])**.** *Two quantum states $|\phi\rangle$ and $|\psi\rangle$ over registers $\mathbf{W}$ and $\mathbf{H}$ are* compatible *if their oracle supports in the computational basis (as defined in definition 2.21) have non-empty intersection, i.e., if $\mathrm{supp}^H(|\phi\rangle) \cap \mathrm{supp}^H(|\psi\rangle) \neq \emptyset$.*

We now state the conjecture.

Conjecture 2.1 (Polynomial compatibility conjecture [ACC+22, Conjecture 4.3])**.** *There exists a finite Abelian group $\mathcal{Y}$ and $\delta = 1/\mathsf{poly}(d)$ such that for any $d, N \in \mathbb{N}$, it holds that any two $(\mathcal{Y}, \delta(d), d, N)$-states $|\phi\rangle$ and $|\psi\rangle$ are compatible.*

2.8 Quantum States, Channels, and Trace

A d-dimensional quantum state is a positive semi-definite Hermitian density matrix $\rho = \sum_{x\in[d]} p_x |\phi_x\rangle\langle\phi_x|$, where the pure states $|\phi_x\rangle\langle\phi_x|$ have trace one, and $p_1, \ldots, p_d$ is a probability distribution, i.e., $p_1, \ldots, p_d \geq 0$ and $p_1 + \cdots + p_d = 1$. Pure states are the rank-1 quantum state that can be written as $|\phi\rangle\langle\phi|$. We sometimes write $|\phi\rangle$ or just ϕ to denote the pure state $|\phi\rangle\langle\phi|$ for simplicity. We can consider any positive semi-definite Hermitian matrix (with any unit trace) as an unnormalized quantum state, e.g., $\Pi\rho\Pi$ for some projection Π and quantum state ρ, and call them unnormalized states.

A quantum channel Φ is a completely positive and trace-preserving operator, that can be represented by matrices $B_1, \ldots, B_k$ satisfying

$$I - \sum_{i=1}^{k} B_i^\dagger B_i \geq 0.$$

The matrices $B_1, \ldots, B_k$ are the Kraus operators of the channel, and with this notation, Φ maps a quantum state ρ to $\Phi(\rho) = \sum_{i=1}^{k} B_i \rho B_i^\dagger$. Quantum channels can represent unitary operations, projective measurements, or applying a projection Π. We write the composition of two quantum channels Φ, Ψ by $\Phi \circ \Psi$.

The trace norm of a Hermitian matrix A is defined by $\|A\|_1 := \sum_{i=1}^{d} |\lambda_i|$, where $\lambda_1, \ldots, \lambda_d$ are the eigenvalues of A. If A is positive semi-definite, we can write $\|A\|_1 = \mathrm{Tr}(A)$. This induces the *trace distance* $\|\rho - \sigma\|_{tr} = \frac{1}{2}\|\rho - \sigma\|_1$ between two (possibly unnormalized) mixed states, which forms a distance over (unnormalized) mixed states. A quantum channel Φ does not increase the trace norm. That is, for any Hermitian matrix A, it holds that $\|\Phi(A)\|_1 \leq \|A\|_1$. In particular, we have $\mathrm{Tr}(\Phi(A)) \leq \mathrm{Tr}(A)$ for any positive semi-definite matrix A. For any two (possibly unnormalized) states ρ, σ,

$$\|\Phi(\rho) - \Phi(\sigma)\|_{tr} = \frac{1}{2}\|\Phi(\rho - \sigma)\|_1 \leq \frac{1}{2}\|\rho - \sigma\|_1 = \|\rho - \sigma\|_{tr}. \tag{2.6}$$

For a positive semi-definite matrix A, it holds that

$$\mathrm{Tr}\big(A^2\big) \leq \mathrm{Tr}(A)^2. \tag{2.7}$$

Lemma 2.9 (Almost as good as new lemma [Aar04, Aar16])**.** *Let $\mathcal{M} = (\Pi_0, \Pi_1)$ be a binary measurement that acts as $\mathcal{M}(\rho) = \Pi_0\rho\Pi_0 + \Pi_1\rho\Pi_1$. If $\mathrm{Tr}[\Pi_0\rho] \geq 1 - \varepsilon$ for $\varepsilon > 0$, then it holds that $\|\rho - \mathcal{M}(\rho)\|_{tr} \leq \sqrt{\varepsilon}$.*

Corollary 2.10. *In the same setting, $\|\rho - \Pi_0\rho\Pi_0\|_{tr} \leq \varepsilon + \sqrt{\varepsilon} \leq 2\sqrt{\varepsilon}$.*

Proof. We have $\|\mathcal{M}(\rho) - \Pi_0\rho\Pi_0\|_{tr} = \|\Pi_1\rho\Pi_1\|_{tr} \leq \varepsilon$, which gives the result. □

We stress that most of the facts on the trace norm and distance also holds for unnormalized states, i.e., positive semi-definite Hermitian matrices.

2.9 The QPSPACE Oracle

We recall the definition of the QPSPACE oracle that implements the arbitrary unitary operation described by polynomial size input [CCS24, BMM+24].

Definition 2.26 (QPSPACE Oracle)**.** *The* unitary *QPSPACE machine oracle, denoted by* QPSPACE*, is defined as follows: it takes a pair (ρ, M, t) of an ℓ-qubit quantum state ρ, a classical Turing machine M, and an integer $t \in \mathbb{N}$. The oracle runs M for t steps to obtain the description of a unitary quantum circuit C that operates on ℓ qubits; if M does not terminate after t steps or the output is not described as above, the oracle halts and returns $\perp$. Otherwise, the oracle applies C on ρ and returns the output quantum state without measurement.*

The quantum access to the QPSPACE oracle is done by allowing coherent (M, t). For any unitary quantum circuit C that is output by a machine M after t step, there is a QPT algorithm with QPSPACE oracle that implements $C^{-1}(\rho)$ on input ρ [BMM+24, Proposition 3.5].

2.10 Haar Random States and Unitaries

The Frobenius norm $\|A\|_F$ of a matrix A is defined by $\sqrt{\mathrm{Tr}(A^\dagger A)}$.

Theorem 2.11 ([Mec19, Theorem 5.17])**.** *Let $n_1, \ldots, n_k \in \mathbb{N}$ and $\mu = \mu_{n_1} \times \cdots \times \mu_{n_k}$ be the product of Haar unitary measures over $X = \mathbb{U}(2^{n_1}) \times \cdots \times \mathbb{U}(2^{n_k})$. Suppose that $f : X \to \mathbb{R}$ is L-Lipschitz in the Frobenius norm. Let $N = \min(2^{n_1}, \ldots, 2^{n_k})$. For every $t > 0$, it holds that*

$$\Pr_{U \leftarrow \mu}\left[f(U) \geq \mathbb{E}_{V \leftarrow \mu}[f(V)] + t\right] \leq \exp\left(-\frac{(N-2)t^2}{24L^2}\right).$$

Corollary 2.12. *Let C^U be an m-query quantum oracle algorithm for the product of Haar random unitaries U chosen from X according to μ defined above. Let $g(U) := \Pr\left[1 \leftarrow C^U\right]$. Then it holds that*

$$\Pr_{U \leftarrow \mu}\left[g(U) \geq \mathbb{E}_{V \leftarrow \mu}[g(V)] + t\right] \leq \exp\left(-\frac{t^2(N-2)}{24m^2}\right).$$

Proof. In [Kre21], it is shown the following statement.

Lemma 2.13 ([Kre21])**.** *Let A^U be a quantum algorithm that makes T queries to the unitary oracle U. Define $f(U) := \Pr\left[1 \leftarrow A^U\right]$. Then f is T-Lipschitz in the Frobenius norm, i.e., $|f(U) - f(V)| \leq T \cdot \|U - V\|_F$.*

This lemma ensure that C is m-Lipschitz, thus applying theorem 2.11, we obtain the desired result. □

Lemma 2.14. *For any rank-D projection Π on m qubits for $m \geq n$,*

$$\mathbb{E}_{|\phi\rangle\leftarrow\sigma_n} \left\langle \phi, 0^{m-n} \right| \Pi \left| \phi, 0^{m-n} \right\rangle \leq \frac{D}{2^n}.$$

If $m = n$, the equality holds. In particular, for any n-qubit mixed state ρ, $\mathbb{E}_{|\phi\rangle\leftarrow\sigma_n} \langle\phi| \rho |\phi\rangle = \frac{1}{2^n}$.

Proof. We simply write 0 to denote 0^{m-n}. We can write $\mathbb{E}_{|\phi\rangle\leftarrow\sigma_n} \langle\phi, 0| \Pi |\phi, 0\rangle$ by

$$\mathbb{E}_{|\phi\rangle\leftarrow\sigma_n} \operatorname{Tr}(\Pi \cdot |\phi, 0\rangle\langle\phi, 0|) = \operatorname{Tr}\left(\Pi \cdot \frac{I \otimes |0\rangle\langle 0|}{2^n}\right) \leq \frac{1}{2^n} \operatorname{Tr}(\Pi) = \frac{D}{2^n},$$

where the last equality follows from the fact that $\operatorname{Tr}(\Pi) = \operatorname{rank}(\Pi)$. If $m = n$, the inequality is saturated. The last statement can be shown by writing $\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|$ for $\sum_i p_i = 1$. □

2.11 State Property Tests

2.11.1 Swap Test

We review the basic results of the swap test, which can be used to test the purity of a state. We provide some lemmas about the swap test on a state that is close to pure states, which are essential to obtain our results.

For two quantum states σ, ρ stored in two different registers $\mathbf{A}, \mathbf{B}$, the swap test is executed on the registers $\mathbf{A}, \mathbf{B}$ and a control register $\mathbf{C}$ initialized to $|1\rangle\langle 1|$. It applies Hadamard on $\mathbf{C}$, swaps $\mathbf{A}$ and $\mathbf{B}$ conditioned on $\mathbf{C}$, and measures $\mathbf{C}$ on the Hadamard basis.

Lemma 2.15 (Swap test)**.** *The swap test on input (σ, ρ) outputs 1 with probability*

$$\frac{1 + \operatorname{Tr}(\rho\sigma)}{2},$$

in which case we say that it passes the swap test. For pure states $|\sigma\rangle, |\rho\rangle$, it equals to $\frac{1+|\langle\rho|\sigma\rangle|^2}{2}$.

When $\sigma = \rho$, we sometimes call it a purity test on ρ, which outputs 1 with certainty if and only if ρ is a pure state.

Lemma 2.16. *Suppose that $\operatorname{Tr}(\rho^2) \leq 1 - 1/T$ for some state ρ and $T \in \mathbb{N}$. Let $\lambda \in \mathbb{N}$. If we run the purity test $16T\lambda$ times on ρ, then the probability that at least 8λ tests fail among $16T\lambda$ is at least $1 - 2^{-\lambda}$.*

Proof. Note that each test succeeds with probability $(1 + \operatorname{Tr}(\rho^2))/2 \leq 1 - 1/2T$, and is independent to each other. Applying Chernoff's inequality (lemma 2.21) for $\delta = 1/2$, we obtain the desired result. □

2.11.2 Product Test

We first recall the product test to determine whether an n-partite state $|\phi\rangle$ is a product state or far from any product state from [HM10], then give a bound on the success of the product test on Haar-random states.

Lemma 2.17 ([HM10, Lemma 3], Product test for mixed states)**.** *Let $m \in \mathbb{N}$ and $d_1, \ldots, d_m$ be the local dimensions of a n-qubit system, i.e. $\prod_{i\in[m]} d_i = 2^n$. Let ρ be a mixed state of n-qubits and for every $S \subseteq [m]$, denote by ρ_S the state after tracing out the subsystem $\overline{S} := [m] \setminus S$. Let $\mathcal{A}_{\text{PTEST}}$ denote the algorithm that given two copies of ρ performs the swap test on each of the m pairs of corresponding subsystems of the two copies of ρ, and that outputs 1 if all the tests succeeds, and 0 otherwise. Then, the probability that the algorithm $\mathcal{A}_{\text{PTEST}}$ outputs 1 when applied to two copies of ρ is equal to*

$$\Pr\left(1 \leftarrow \mathcal{A}_{\text{PTEST}}(|\phi\rangle^{\otimes 2})\right) = \frac{1}{2^m} \sum_{S\subseteq[m]} \operatorname{Tr}\left[\rho_S^2\right].$$

For Haar-random states, the above formula is explicitly calculated for any partition $S \cup \overline{S}$ of $[m]$ by [Lub78]:

$$\mathop{\mathbb{E}}_{|\psi\rangle\leftarrow\sigma} \operatorname{Tr}\left[\rho_S^2\right] = \frac{d_S + d_{\overline{S}}}{d_S \cdot d_{\overline{S}} + 1}.$$

As a consequence, we have the following bound for the success of the product test on Haar-random states.

Lemma 2.18 (Product test for Haar-random states)**.** *Let $m \in \mathbb{N}$ and $\{d_i\}_{i\in[m]}$ be the local dimensions of a n-qubit system, i.e. $\prod_{i\in[m]} d_i = 2^n$. Then, the probability that the algorithm $\mathcal{A}_{\text{PTEST}}$ outputs 1 when applied to two copies of a n-qubit Haar-random state $|\psi\rangle$ satisfies:*

$$\mathop{\mathbb{E}}_{|\psi\rangle\leftarrow\sigma} \Pr\left(1 \leftarrow \mathcal{A}_{\text{PTEST}}(|\psi\rangle^{\otimes 2})\right) \leq 2\left(\frac{3}{4}\right)^m.$$

Proof. For every partition $S \cup \overline{S}$ of $[m]$, the local dimension of each partition is given by $d_S = \prod_{i\in S} d_i$.

$$\begin{aligned}
\mathop{\mathbb{E}}_{|\psi\rangle\leftarrow\sigma} \Pr\left(1 \leftarrow \mathcal{A}_{\text{PTEST}}(|\psi\rangle^{\otimes 2})\right) &= \mathop{\mathbb{E}}_{|\psi\rangle\leftarrow\sigma} \left[\frac{1}{2^m} \sum_{S\subseteq[m]} \operatorname{Tr}\left[\rho_S^2\right]\right] \\
&= \frac{1}{2^m} \sum_{S\subseteq[m]} \frac{d_S + d_{\overline{S}}}{d_S \cdot d_{\overline{S}} + 1} \leq \frac{1}{2^m} \sum_{S\subseteq[m]} \frac{d_S + d_{\overline{S}}}{d_S \cdot d_{\overline{S}}} \\
&= \frac{1}{2^m} \left(\sum_{S\subseteq[m]} \frac{1}{d_S} + \frac{1}{d_{\overline{S}}}\right) = \frac{2}{2^m} \left(\sum_{S\subseteq[m]} \frac{1}{d_S}\right) \\
&= \frac{2}{2^m} \prod_{i\in[m]} \left(1 + \frac{1}{d_i}\right) \leq \frac{2}{2^m} \prod_{i=1}^{m} \left(\frac{3}{2}\right) = 2\left(\frac{3}{4}\right)^m,
\end{aligned}$$

where we use the fact that each $d_i \geq 2$ to obtain the last inequality. □

2.12 The Quantum OR lemma

Lemma 2.19 ([HLM17, Corollary 3.1], Quantum OR lemma)**.** *Let $\{\Pi_i\}_{i\in[N]}$ be binary-valued POVMs. Let $0 < \varepsilon < 1/2$ and $\delta > 0$. Let Ψ be a quantum state such that either*

1. *there exists $i \in [N]$ such that $\mathrm{Tr}[\Pi_i\Psi] \geq 1-\varepsilon$, or*
2. *for all $i \in [N]$, $\mathrm{Tr}[\Pi_i\Psi] \leq \delta$.*

Then, there is a quantum circuit C, called "OR tester", such that measuring the first qubit in case i) yields

$$\Pr(1 \leftarrow C(\Psi)) \geq \frac{(1-\varepsilon)^2}{7},$$

and in case ii),

$$\Pr(1 \leftarrow C(\Psi)) \leq 4N\delta.$$

Moreover, the circuit C can be implemented by a unitary quantum poly-space machine as long as each POVM Π_i can be implemented by a quantum poly-space machine and the set of measurements has a concise polynomial description. In other words, the quantum OR tester can be executed by a QPSPACE*-aided BQP algorithm, where the oracle* QPSPACE *is defined in definition 2.26.*

Remark 2.4. *"Moreover" part of the above theorem for the projective measurements is shown in [CCS24, Appendix A], and the extension to the POVMs is observed in [BMM+24, Lemma 5.2].*

2.13 Complexity Classes

We include the definitions of PromiseBQP and PromiseQMA and we refer the reader to [Kre21] for a lengthier explanation.

Definition 2.27. *A promise problem $\mathcal{L} = \mathcal{L}_{\text{yes}} \cup \mathcal{L}_{\text{no}} \cup \mathcal{L}_{\perp}$ with $\mathcal{L} \subseteq \{0,1\}^*$ is in* PromiseQMA *(Quantum Merlin–Arthur) if there exists a polynomial-time quantum algorithm $\mathsf{V}(x, |\psi\rangle)$ called a* verifier *and a polynomial p such that:*

1. *(Completeness) If $x \in \mathcal{L}_{\text{yes}}$, then there exists a quantum state $|\psi\rangle$ on $p(|x|)$ qubits (called a* witness *or* proof*) such that $\Pr[\mathsf{V}(x, |\psi\rangle) = 1] \geq \frac{2}{3}$.*
2. *(Soundness) If $x \in \mathcal{L}_{\text{no}}$, then for every quantum state $|\psi\rangle$ on $p(|x|)$ qubits, $\Pr[\mathsf{V}(x, |\psi\rangle) = 1] \leq \frac{1}{3}$.*

Definition 2.28. *A promise problem $\mathcal{L} = \mathcal{L}_{\text{yes}} \cup \mathcal{L}_{\text{no}} \cup \mathcal{L}_{\perp}$ with $\mathcal{L} \subseteq \{0,1\}^*$ is in* PromiseBQP *(Bounded-error Quantum Polynomial time) if there exists a randomized polynomial-time quantum algorithm $\mathcal{A}(x)$ such that:*

1. *If* $x \in \mathcal{L}_{\text{yes}}$, *then* $\Pr[\mathcal{A}(x) = 1] \geq \frac{2}{3}$.
2. *If* $x \in \mathcal{L}_{\text{no}}$, *then* $\Pr[\mathcal{A}(x) = 1] \leq \frac{1}{3}$.

2.14 Process Tomography

The diamond norm of an operator A, denoted by $\|A\|_\diamond$, is defined by:

$$\|A\|_\diamond := \sup_{\mathrm{Tr}(\rho)=1, \rho \geq 0} \|(A \otimes I)(\rho)\|_1,$$

where I denotes the identity acting with the same dimension as A. We only use the following fact about the diamond norm: for quantum channels A, B and a density matrix ρ, it holds that

$$\|A \otimes I(\rho) - B \otimes I(\rho)\|_{tr} \leq \frac{1}{2}\|A - B\|_\diamond.$$

Theorem 2.20 ([HKOT23])**.** *There exists a quantum algorithm* Tom *that, given black-box access to a unitary Z acting on the d-dimensional space, satisfies the following for any input $\varepsilon, \delta \in (0, 1)$:*

Accuracy: *It outputs a classical description of a unitary Z such that*

$$\Pr_{Z' \leftarrow \mathtt{Tom}}\left[\|Z(\cdot)Z^\dagger - Z'(\cdot)Z'^\dagger\|_\diamond \leq \varepsilon\right] \geq 1 - \delta.$$

Efficiency: *It makes $O\left(\frac{d^2}{\varepsilon}\log\frac{1}{\delta}\right)$ queries to Z, and takes $\mathsf{poly}(d, \frac{1}{\varepsilon}, \log\frac{1}{\delta})$ time.*

2.15 Chernoff Bounds

We use the following concentration inequalities.

Lemma 2.21 (Multiplicative Chernoff bound)**.** *Let $X_1, \ldots, X_n$ be some independent random variables over $\{0, 1\}$. Let $X = \sum_{i=1}^n X_i$ and $\mu = \mathbb{E}[X]$. It holds that*

- $\Pr[X \geq (1+\delta)\mu] \leq \exp\left(-\frac{\mu\delta^2}{2+\delta}\right)$ *for $\delta \geq 0$, and*
- $\Pr[X \leq (1-\delta)\mu] \leq \exp\left(-\frac{\mu\delta^2}{2}\right)$ *for $0 < \delta < 1$.*

2.16 Lemmas on Quantum Primitives

We can build PD-OWFs from short-PRSs.

Theorem 2.22 (Adapted from [BBO+24, Theorem 6][1]). *Assuming the existence of $(c \log \lambda)$-PRSs with $c > 12$, there exists a $O(\lambda^{-c/12+1})$-PD-OWF $F : \{0,1\}^{\ell(\lambda)} \to \{0,1\}^{\ell(\lambda)}$ with input/output length $\ell(\lambda) = \lambda^{c/6}$.*

We will need Kretschmer's (quantum) oracle $\mathcal{O}$ relative to which OWFs do not exist, but PRSs do. Note that the former is because PromiseBQP and PromiseQMA are equal relative to this oracle.

Theorem 2.23 ([Kre21]). *There exists a quantum oracle $\mathcal{O}$, such that:*

1. $\mathsf{PromiseBQP}^{\mathcal{O}} = \mathsf{PromiseQMA}^{\mathcal{O}}$.
2. *λ-PRUs exist relative to $\mathcal{O}$. Moreover this implies that λ-PRSs exist [JLS18].*

We will also need a result from [CGG24].

Lemma 2.24. *The existence of EV-OWPs imply that* $\mathsf{BQP} \neq \mathsf{QCMA}$. *Moreover, the proof relativizes.*

We will use the following lemma in Section 4.2.

Lemma 2.25 (Independence [ACC+22, Lemma 3.2]). *Suppose two quantum algorithms* A *and* B *interact classically in the quantum random oracle model. Let $\mathbf{W}_{\mathsf{A}}$ and $\mathbf{W}_{\mathsf{B}}$ denote their internal registers respectively. Then, at any time during the protocol, conditioned on the (classical) transcript t and the fixed oracle $h \in \mathcal{H}$, the joint state of the registers $\mathbf{W}_{\mathsf{A}}$ and $\mathbf{W}_{\mathsf{B}}$ conditioned on t and h is a product state.*

[1] Here we also use the PD-OWF variant of their theorem originally for PD-OWHF. This choice affects the parameters of the domain and range in the theorem statement because constructing a PD-OWHF requires more steps than constructing a PD-OWF (we only need the first step of their proof). However, note that changing the domain/range of the function to some different polynomials in λ would still make the proofs in Section 5.1 go through by changing some parameters in the proof.

Chapter 3

Quantum Security of Subset Cover Problems

In this chapter, we show upper and lower bounds on quantum algorithms for finding a subset cover.

3.1 The k–Restricted Subset Cover Problem

In this section, we prove a lower bound for the k–RSC problem defined in Definition 2.15. This section follows closely [LZ19]'s proof of their lower bound on finding multi-collisions. We will first prove a lower bound for the problem when $k = 2$. Then, we will prove a lower bound for finding k distinct 2–RSC, which will be necessary in our induction step. Finally, we will prove the induction step in the last subsection and obtain a lower bound on finding s distinct k–RSC.

3.1.1 Lower Bound on Finding a 2–Restricted Subset Cover

In this section, we will prove that the number of queries necessary to find a 2–RSC is $\Omega(N^{3/7})$, matching the query complexity of the quantum algorithm proposed in [YTA22], up to a constant factor.

As presented in Definition 2.15, in the 2–RSC problem, we are given 2 random functions h_1, h_2 such that for $i \in \{1, 2\}$, $h_i : \mathcal{X} \to \mathcal{Y}$. The main theorem of this subsection can be stated as follows:

Theorem 3.1. *Given two random functions $h_1, h_2 : \mathcal{X} \to \mathcal{Y}$ where $|N| = \mathcal{Y}$, a quantum algorithm needs to make $\Omega(N^{3/7})$ queries to h_1 and h_2 to find a 2–RSC with a constant probability.*

In order to prove this theorem, we first introduce some database properties:

- $P'_{\ell-col-h_1}$ corresponds to the set of databases that contain *at least* ℓ *distinct* collisions on h_1.[1] As explained in the previous section, here we will use the fact that we cannot reach a database containing $\ell+2$ or more collisions from a database containing ℓ collisions by making a single query:

$$P'_{\ell-col-h_1} = \left\{ D \in \mathcal{D} \middle| \begin{array}{l} \exists x_1, \ldots, x_\ell, y_1, \ldots, y_\ell, \forall i, h_1(x_i) = h_1(y_i) \neq \perp \\ \forall i, x_i \neq y_i \\ \forall i \neq j, h_1(x_i) \neq h_1(x_j) \end{array} \right\}$$

- $P_{\ell-col-h_1}$ corresponds to the set of databases that contain *exactly* ℓ *distinct* collisions on h_1:

$$P_{\ell-col-h_1} = P'_{\ell-col-h_1} \cap \neg P'_{(\ell+1)-col-h_1}$$

- $P_{preimage-h_1}$ corresponds to the set of databases that contain a preimage of 0:[2]

$$P_{preimage-h_1} = \{D \in \mathcal{D} | \exists x, h_1(x) = 0\}.$$

Finally, for $i, \ell \in \mathbb{N}$, we write:

$$\widetilde{f}^{col}_{i,\ell} = \left| P_{\ell-col-h_1} |\psi_i\rangle \right|, f^{col}_{i,\ell} = \left| P'_{\ell-col-h_1} |\psi_i\rangle \right|, g_i = \left| P^{RSC}_{1,2} |\psi_i\rangle \right|, \tag{3.1}$$

where $|\psi_i\rangle$ is the state just after the i^{th} query to $H = (h_1, h_2)$ and $P^{RSC}_{1,2}$ was defined in Equation (2.5). For convenience, we write $P_2 = P^{RSC}_{1,2}$ in this section.

The goal here is to bound the term g_i, and to achieve this we first prove a recursive formula that involves $\widetilde{f}^{col}_{i,\ell}$ as well:

Lemma 3.2. *For every $i \in \mathbb{N}$, we have that:*

$$g_i \leq g_{i-1} + \sqrt{2 \sum_{\ell \geq 0} \frac{\ell}{N} \widetilde{f}^{col}_{i-1,\ell}{}^2} + 4\frac{i-1}{N}. \tag{3.2}$$

Proof. Let $i \in \mathbb{N}$. Let $|\phi_i\rangle$ be the state just before the i^{th} query to $H = (h_1, h_2)$, namely

$$|\phi_i\rangle = \sum_{x,\hat{y},z,D} \alpha_{x,\hat{y},z,D} |x, \hat{y}, z\rangle \otimes |D\rangle,$$

[1] We do not define the equivalent property for h_2. Since both h_1 and h_2 are random functions, we can swap them when considering database property by symmetry, thus we do not need to define more unnecessary properties.

[2] Note that the amplitude of finding *any* preimage is the same as the amplitude of finding the preimage of 0.

where x is the query register, y is the answer register, z is the work register and D is the database register. Let $|\psi_i\rangle$ be the state right after the i^{th} query to H, namely

$$|\psi_i\rangle = \sum_{\substack{x,\hat{y},z,D \\ D(x)=\perp}} \frac{1}{\sqrt{N^2}} \sum_{y'} \omega_N^{yy'} \alpha_{x,\hat{y},z,D} |x,\hat{y},z\rangle \otimes |D \cup (x,y')\rangle + \mathsf{cO} \sum_{\substack{x,\hat{y},z,D \\ D(x)\neq\perp}} \alpha_{x,\hat{y},z,D} |x,\hat{y},z\rangle \otimes |D\rangle .$$

From Lemma 2.4, we have that:

$$|P_2 |\psi_i\rangle| \leq |P_2 |\phi_i\rangle| + |P_2 \mathsf{cO}(I - P_2) |\phi_i\rangle| . \tag{3.3}$$

We focus now on bounding the second term:

$$\begin{aligned} |P_2 \mathsf{cO}(I - P_2) |\phi_i\rangle| &= \left| P_2 \mathsf{cO} \sum_{\substack{x,\hat{y},z \\ D\text{: no 2–RSC}}} \alpha_{x,\hat{y},z,D} |x,\hat{y},z,D\rangle \right| \\ &\leq \left| P_2 \sum_{\substack{x,\hat{y},z \\ D\text{: no 2–RSC} \\ D(x)=\perp}} \frac{1}{\sqrt{N^2}} \sum_{y'} \omega_N^{yy'} \alpha_{x,\hat{y},z,D} |x,\hat{y},z,D \cup (x,y')\rangle \right| \\ &\quad + \left| P_2 \mathsf{cO} \sum_{\substack{x,\hat{y},z \\ D\text{: no 2–RSC} \\ D(x)\neq\perp}} \alpha_{x,\hat{y},z,D} |x,\hat{y},z\rangle \otimes |D\rangle \right| \end{aligned}$$

The second term can be bounded by

$$
\begin{aligned}
&\left| P_2 \mathsf{cO} \sum_{\substack{x,\hat{y},z \\ D\text{: no 2–RSC} \\ D(x)\neq\perp}} \alpha_{x,\hat{y},z,D} \left|x,\hat{y},z\right\rangle \otimes \left|D\right\rangle \right| \\
&= \left| P_2 \sum_{\substack{x,\hat{y},z \\ D\text{: no 2–RSC} \\ D(x)\neq\perp}} \frac{1}{N^2} \sum_{y'} \left(1 - \omega_N^{yy'} - \omega_N^{D(x)y}\right) \alpha_{x,\hat{y},z,D} \left|x,\hat{y},z\right\rangle \otimes \left|D \cup (x,y')\right\rangle \right| \\
&\leq 3 \left| \frac{1}{N^2} \sum_{y'} P_2 \sum_{\substack{x,\hat{y},z \\ D\text{: no 2–RSC} \\ D(x)\neq\perp}} \alpha_{x,\hat{y},z,D} \left|x,\hat{y},z\right\rangle \otimes \left|D \cup (x,y')\right\rangle \right| \\
&\leq \frac{3(i-1)}{N}, \qquad (3.4)
\end{aligned}
$$

where the first inequality comes from Lemma 2.2 and the fact that if the new value in the x register is $|\perp\rangle$ or stays the same, then there is still no 2-RSC in D. The second inequality comes from the triangular inequality and the last inequality comes from using the triangular inequality and the fact that there is at most $(i-1)$ values in D such that $D(x) \neq \perp$.

For bounding the first term, we analyze now the possibilities for achieving a 2–RSC, considering the different cases of the inner sum. We have four possible ways to get from D that does not have a 2–RSC to $D_{y'} := D \cup (x, y')$ that has a 2–RSC.

- $(x = x_2)$ Here, we consider the case where there exists an x_0 and x_1 such that $h_1(x_0) = h_1(x_1)$ and we query x such that $h_2(x_0) = h_2(x)$. If we have found ℓ collisions of h_1 in D, then ℓ values of y' can make $D_{y'}$ contain a 2–RSC, out of the N possible values for the outcome of h_2 (notice that the value of $h_1(x)$ is not relevant for this case).
- $(x = x_1)$ Similar to the previous case, but swapping the roles of h_1 and h_2.
- $(x = x_0)$ Otherwise, we consider the case where we query x such that we have x_1 and x_2 (which might be equal), such that $h_1(x) = h_1(x_1)$ and $h_2(x) = h_2(x_2)$. Only $i-1$ values of y' will make $D_{y'}$ contain a collision on h_1. Similarly, only $i-1$ values of y' will make $D_{y'}$ contain a collision on h_2.

Thus, we have

$$
\left|P_2 \mathsf{cO} (I - P_2) \left|\phi_i\right\rangle\right| \leq \left(2 \cdot \sum_{\ell \geq 0} \frac{\ell}{N} \left|P_{\ell - col - h_1} \left|\phi_i\right\rangle\right|^2\right)^{1/2} + 4\frac{(i-1)}{N}, \qquad (3.5)
$$

and we give the details on Equation (3.5) in Appendix B.2.

Let $|\psi_{i-1}\rangle$ be the state just after the $(i-1)^{th}$ query, and let U_i be the unitary such that $|\phi_i\rangle = (U_i \otimes I)\,|\psi_{i-1}\rangle$ (see Equation (2.4)). Note that we also have $|\psi_i\rangle = \mathsf{cO}\cdot(U_i \otimes I)\,|\psi_{i-1}\rangle$. Using Lemma 2.3, we get that:

$$\begin{aligned}|P_2\mathsf{cO}(I-P_2)\,|\phi_i\rangle| &\leq \sqrt{2\sum_{\ell\geq 0}\frac{\ell}{N}\,|P_{\ell-col-h_1}(U_i\otimes I)\,|\psi_{i-1}\rangle|^2} + 4\frac{i-1}{N}\\ &\leq \sqrt{2\sum_{\ell\geq 0}\frac{\ell}{N}\,|P_{\ell-col-h_1}\,|\psi_{i-1}\rangle|^2} + 4\frac{i-1}{N}. \end{aligned} \tag{3.6}$$

Similarly, using Lemma 2.3:

$$|P_2\,|\phi_i\rangle| = |P_2\,(U_i\otimes I)\,|\psi_{i-1}\rangle| = |P_2\,|\psi_{i-1}\rangle|\,. \tag{3.7}$$

Then, using Equation (3.3), Equation (3.6) and Equation (3.7), and the notation from Equation (3.1), we have:

$$g_i \leq g_{i-1} + \sqrt{2\sum_{\ell\geq 0}\frac{\ell}{N}\widetilde{f}^{col}_{i-1,\ell}{}^2} + 4\frac{i-1}{N}.$$

□

We will now expand this recursive formula to obtain a bound on g_i.

Lemma 3.3. *For every $i \in \mathbb{N}$, we have that:*

$$g_i \leq \sqrt{2}\sum_{j=1}^{i-1}\sqrt{\frac{\mu_3(j)}{N}} + \sqrt{2}\cdot 2^{-9.5N^{1/8}} + 4\frac{i^2}{N},$$

where

$$\mu_3(j) = \max\left\{8e\frac{j^{3/2}}{\sqrt{N}}, 10N^{1/8}\right\}.$$

Proof. From Lemma 3.2, we expand recursively Equation (3.2), and obtain (using that $g_0 = 0$):

$$g_i \leq \sum_{j=1}^{i-1}\sqrt{2\sum_{\ell\geq 0}\frac{\ell}{N}\widetilde{f}^{col}_{j,\ell}{}^2} + 4\sum_{j=1}^{i-1}\frac{j}{N}. \tag{3.8}$$

The second term of Equation (3.8) can be bounded by

$$4\sum_{j=1}^{i-1}\frac{j}{N} \leq 4\sum_{j=1}^{i-1}\frac{i}{N} \leq 4\frac{i^2}{N}. \tag{3.9}$$

As for the first term of Equation (3.8), we have:

$$\begin{aligned}
\sum_{j=1}^{i-1}\sqrt{2\sum_{\ell\geq 0}\frac{\ell}{N}\widetilde{f}_{j,\ell}^{col}{}^2} &= \sqrt{2}\sum_{j=1}^{i-1}\sqrt{\sum_{\ell=0}^{\mu_3(j)}\frac{\ell}{N}\widetilde{f}_{j,\ell}^{col}{}^2 + \sum_{\ell>\mu_3(j)}\frac{\ell}{N}\widetilde{f}_{j,\ell}^{col}{}^2} \\
&\leq \sqrt{2}\sum_{j=1}^{i-1}\left(\sqrt{\sum_{\ell=0}^{\mu_3(j)}\frac{\ell}{N}\widetilde{f}_{j,\ell}^{col}{}^2} + \sqrt{\sum_{\ell>\mu_3(j)}1\cdot\widetilde{f}_{j,\ell}^{col}{}^2}\right) \\
&\leq \sqrt{2}\sum_{j=1}^{i-1}\left(\sqrt{\frac{\mu_3(j)}{N}\cdot f_{j,1}^{col}} + f_{j,\mu_3(j)}^{col}\right) \\
&\leq \sqrt{2}\left(\sum_{j=1}^{i-1}\sqrt{\frac{\mu_3(j)}{N}} + \sum_{j=1}^{i-1}f_{j,\mu_3(j)}^{col}\right)
\end{aligned} \tag{3.10}$$

where in the second inequality, we used the fact that the term $\sum_{\ell>\mu_3(j)}\widetilde{f}_{j,\ell}^{col}{}^2$ is equal to the amplitude of finding *at least* $\mu_3(j)$ distinct ℓ–collisions on h_1, thus is exactly equal to $f_{j,\mu_3(j)}^{col}{}^2$ (defined in Equation (3.1)), and similarly for $f_{j,1}^{col}$.

It follows that

$$\sum_{j=1}^{i-1}f_{j,\mu_3(j)}^{col} \leq \sum_{j=1}^{i-1}\left(\frac{4e\cdot j^{3/2}}{\mu_3(j)\cdot\sqrt{N}}\right)^{\mu_3(j)} \leq \sum_{j=1}^{i-1}\left(\frac{1}{2}\right)^{10N^{1/8}} \leq 2^{-9.5N^{1/8}}, \tag{3.11}$$

where the first inequality comes from Lemma 2.5, the second inequality comes from the definition of $\mu_3(j)$ and in the last inequality we assume that $i \leq N^{1/2}$. Indeed, otherwise A can execute [YTA22]'s algorithm whose query complexity for finding a k–RSC is upper-bounded by $O\left(N^{1/2}\right)$.

Putting together Equation (3.8), Equation (3.9), Equation (3.10) and Equation (3.11) gives the result. □

We can now use Lemma 3.3 to prove Theorem 3.1

Proof of Theorem 3.1. Using Lemma 3.3, we have for $i \in \mathbb{N}$:

$$g_i \leq \sqrt{2}\sum_{j=1}^{i-1}\sqrt{\frac{\mu_3(j)}{N}} + \sqrt{2}\cdot 2^{-9.5N^{1/8}} + 4\frac{i^2}{N}.$$

We can bound the first term by:

$$\begin{aligned}
\sqrt{2}\sum_{j=1}^{i-1}\sqrt{\frac{\mu_3(j)}{N}} &= \sqrt{2}\left(\sum_{j:\mu_3(j)=8e\cdot\frac{j^{3/2}}{\sqrt{N}}}\frac{\sqrt{8ej^{3/2}}}{N^{3/4}}+\sum_{j:\mu_3(j)=10N^{1/8}}\frac{\sqrt{10N^{1/8}}}{N^{1/2}}\right)\\
&\leq \sqrt{2}\left(\sum_{j=1}^{i-1}\frac{\sqrt{8ej^{3/2}}}{N^{3/4}}+\sum_{j:\mu_3(j)=10N^{1/8}}\frac{\sqrt{10N^{1/8}}}{N^{1/2}}\right)\\
&\leq 4\sqrt{e}\frac{i^{7/4}}{N^{3/4}}+\left(\frac{10}{8e}\right)^{2/3}\cdot N^{5/12}\cdot\frac{\sqrt{10N^{1/8}}}{N^{1/2}}\\
&\leq 4\sqrt{e}\frac{i^{7/4}}{N^{3/4}}+O(N^{-1/48}),
\end{aligned}$$

where the second inequality comes from counting the number of j such that $\mu_3(j) = 10N^{1/8}$, which is equal to the number of j such that $8e\frac{j^{3/2}}{\sqrt{N}} \leq 10N^{1/8}$.

Thus, we have the following bound on g_i:

$$g_i \leq 4\sqrt{e}\frac{i^{7/4}}{N^{3/4}}+4\frac{i^2}{N}+O(N^{-1/48}).$$

This bound is in the compressed oracle model, and using Lemma 2.1 we obtain the same bound in the random oracle model by putting the negligible term in the $O\left(N^{-1/48}\right)$.

So when $i = o(N^{3/7})$, we have $g_i = o(1)$. Hence if we want g_i to be constant, i.e. not $o(1)$, we must have $i = \Omega\left(N^{3/7}\right)$. □

3.1.2 Lower Bound on Finding k Distinct 2–Restricted Subset Cover

We are now interested in bounding the number of queries needed to find k distinct triplets that satisfy a 2–RSC. We have the following result:

Theorem 3.4. *Given two random functions $h_1, h_2 : \mathcal{X} \to \mathcal{Y}$ where $N = |\mathcal{Y}|$, a quantum algorithm needs to make $\Omega(k^{4/7} \cdot N^{3/7})$ queries to h_1 and h_2 to find k distinct 2–RSC with constant probability, for any $k \leq N^{1/8}$.*

To prove this theorem, we first introduce some notation. We denote $P_{2,k,\ell}$ the set of databases that satisfies k distinct 2–RSC, and that contain exactly ℓ collisions on h_1. Using the notation from the Section 3.1.1 and Equation (2.5), we have that $P_{2,k,\ell} = P_{k,2}^{RSC} \cap P_{\ell-col-h_1}$. We denote $g_{i,k} = \left|P_{k,2}^{RSC}\left|\psi_i\right\rangle\right|$ and $\widehat{g}_{i,k,\ell} = \left|P_{2,k,\ell}\left|\psi_i\right\rangle\right|$, where $|\psi_i\rangle$ is the state just after the i^{th} query to $H = (h_1, h_2)$.

Our goal is to bound $g_{i,k}$, and as in the previous subsection, we will first prove a recursive formula stated in the next lemma.

Lemma 3.5. *For every $i \in \mathbb{N}$, and every $k \in \mathbb{N}$, we have that:*

$$g_{i,k} \leq g_{i-1,k} + \sqrt{2\sum_{\ell \geq 0} \frac{\ell}{N} \widehat{g}^2_{i-1,k-1,\ell}} + \frac{(i-1)}{N} g_{i-1,k-1}.$$

Proof. From Lemma 2.4, we have the following inequality:

$$\left|P^{RSC}_{k,2} |\psi_i\rangle\right| \leq \left|P^{RSC}_{k,2} |\phi_i\rangle\right| + \left|P^{RSC}_{k,2} \mathsf{cO}(I - P^{RSC}_{k,2}) |\phi_i\rangle\right|.$$

And we have that:

$$\begin{aligned}
&\left|P^{RSC}_{k,2} \mathsf{cO}(I - P^{RSC}_{k,2}) |\phi_i\rangle\right| \\
&\leq \left| P^{RSC}_{k,2} \sum_{\substack{x,\hat{y},z \\ D\text{:k-1 2-RSC} \\ D(x)=\perp}} \frac{1}{\sqrt{N^2}} \sum_{y'} \omega_N^{yy'} \alpha_{x,\hat{y},z,D} |x, \hat{y}, z, D \cup (x, y')\rangle \right| + 3\frac{i-1}{N} \left|P^{RSC}_{k-1,2} |\phi_i\rangle\right|^2 \\
&\leq \left(2\sum_{\ell \geq 0} \frac{\ell}{N} \sum_{\substack{x,\hat{y},z \\ D\text{:k-1 2-RSC} \\ \ell \text{ collisions} \\ \text{on } h_1}} |\alpha_{x,\hat{y},z,D}|^2 \right)^{1/2} + \left(\frac{(i-1)^2}{N^2} \sum_{\substack{x,\hat{y},z \\ D\text{:k-1 2-RSC}}} |\alpha_{x,\hat{y},z,D}|^2 \right)^{1/2} + 3\frac{i-1}{N} \left|P^{RSC}_{k-1,2} |\phi_i\rangle\right|^2 \\
&\leq \left(2\sum_{\ell \geq 0} \frac{\ell}{N} |P_{2,k-1,\ell} |\phi_i\rangle|^2 \right)^{1/2} + 4\frac{i-1}{N} \left|P^{RSC}_{k-1,2} |\phi_i\rangle\right|^2,
\end{aligned}$$

where the first inequality comes from the same calculations done to obtain Equation (3.4), and the second equality uses the same cases as for the case $k = 1$ in Lemma 3.2.

Using Lemma 2.3 and previous notation (as in Lemma 3.2), we obtain that:

$$g_{i,k} \leq g_{i-1,k} + \left(2\sum_{\ell \geq 0} \frac{\ell}{N} \widehat{g}^2_{i-1,k-1,\ell} \right)^{1/2} + 4\frac{(i-1)}{N} g_{i-1,k-1}.$$

□

Following the proof from the case $k = 1$, we will split the sum in two using $\mu_3(j)$ as a threshold. We also define a new notation that will simplify expressions:

Definition 3.1.

$$A_i = \sum_{\ell=0}^{i-1} \sqrt{2} \left(\sqrt{\frac{\mu_3(\ell-1)}{N}} + \sqrt{8}\frac{\ell-1}{N} \right),$$

where

$$\mu_3(\ell) = \max\left\{8e\frac{\ell^{3/2}}{\sqrt{N}}, 10N^{1/8}\right\}.$$

Before bounding $g_{i,k}$, we first prove a bound on A_i.

Lemma 3.6. *For every $i \in \mathbb{N}$, we have that:*

$$A_i \leq 8\sqrt{e}\frac{i^{7/4}}{N^{3/4}} + 4\frac{i^2}{N} + O\left(N^{-1/48}\right).$$

It follows that $A_i < 2eN^{1/8}$ for $i \leq N^{1/2}$.

We leave the proof of Lemma 3.6 to Appendix B.3. We can now state the lemma that bounds $g_{i,k}$.

Lemma 3.7. *For every $i \in \mathbb{N}$ and $k \in \mathbb{N}$, we have that:*

$$g_{i,k} < \frac{A_i^k}{k!} + \sqrt{2}\cdot 2^{-N^{1/8}}.$$

Proof. We write $f_{i,j}^{col} = \left|P'_{j-col-h_1}|\phi_i\rangle\right|$. From Lemma 3.5, we have that:

$$\begin{aligned} g_{i,k} &\leq g_{i-1,k} + \sqrt{2\sum_{\ell\geq 0}\frac{\ell}{N}\cdot\widehat{g}^2_{i-1,k-1,\ell}} + 4\frac{i-1}{N}\cdot g_{i-1,k-1} \\ &\leq g_{i-1,k} + \sqrt{2}\left(\sqrt{\frac{\mu_3(i-1)}{N}}\cdot g_{i-1,k-1} + f^{col}_{i-1,\mu_3(i-1)}\right) + 4\frac{i-1}{N}\cdot g_{i-1,k-1} \\ &= g_{i-1,k} + \sqrt{2}\left(\sqrt{\frac{\mu_3(i-1)}{N}} + \sqrt{8}\frac{i-1}{N}\right) g_{i-1,k-1} + \sqrt{2}\cdot f^{col}_{i-1,\mu_3(i-1)}, \end{aligned} \tag{3.12}$$

where the second inequality comes from separating the sum in two, similar to the proof of Lemma 3.3.

Following [LZ19]'s proof for Lemma 14, by expanding the recursion we get:

$$g_{i,k} \leq \frac{A_i^k}{k!} + \sqrt{2}\cdot e^{A_i}2^{9.5N^{1/8}}. \tag{3.13}$$

For completeness, the proof of Equation (3.13) is given in Appendix B.4. Using Lemma 3.6, we can bound the second term, and:

$$g_{i,k} < \frac{A_i^k}{k!} + \sqrt{2}\cdot 2^{-N^{1/8}}.$$

□

We can now prove the main theorem of this subsection.

Proof of Theorem 3.4. Following from Lemma 3.7, we have that:

$$g_{i,k} \leq \frac{A_i^k}{k!} + \sqrt{2} \cdot 2^{-N^{1/8}} \leq \left(\frac{A_i \cdot e}{k}\right)^k + \sqrt{2} \cdot 2^{-N^{1/8}}.$$

We now use the bound on A_i of Lemma 3.6:

$$g_{i,k} \leq \left(\frac{8e^{3/2}}{k} \cdot \frac{i^{7/4}}{N^{3/4}} + \frac{4e}{k} \cdot \frac{i^2}{N} + \frac{e}{k} \cdot O\left(N^{-1/48}\right)\right)^k + \sqrt{2} \cdot 2^{-N^{1/8}}.$$

So if $i = o(k^{4/7} \cdot N^{3/7})$, then $g_{i,k} = o(1)$. Hence if we want $g_{i,k}$ to be a constant, i.e. not $o(1)$, we must have $i = \Omega\left(k^{4/7} \cdot N^{3/7}\right)$. □

3.1.3 Lower Bound on Finding k Distinct s–Restricted Subset Cover

In this section, we generalize the result to the problem of finding k distinct s–RSC, for any $s \geq 3$ and any $k \geq 1$. We are given s random functions $h_1, \ldots, h_s$ such that for any $i \in [1, s]$, $h_i : \mathcal{X} \to \mathcal{Y}$. We will prove the following theorem.

Theorem 3.8. *Given s random functions $h_1, \ldots, h_s : \mathcal{X} \to \mathcal{Y}$ where $N = |\mathcal{Y}|$, a quantum algorithm needs to make $\Omega\left((s+1)^{-\frac{2^s}{2^{s+1}-1}} \cdot k^{\frac{2^s}{2^{s+1}-1}} \cdot N^{\frac{2^s-1}{2^{s+1}-1}}\right)$ queries to $h_1, \ldots, h_s$ to find k distinct s–RSC with constant probability, for any $s \leq \log(\log(N))$ and any $k \geq N^{1/2^{s+1}}$.*

And naturally we have the following corollary for $k = 1$:

Corollary 3.9. *Given s random functions $h_1, \ldots, h_s : \mathcal{X} \to \mathcal{Y}$ where $N = |\mathcal{Y}|$, a quantum algorithm needs to make $\Omega\left((s+1)^{-\frac{2^s}{2^{s+1}-1}} \cdot N^{\frac{2^s-1}{2^{s+1}-1}}\right)$ queries to $h_1, \ldots, h_s$ to find one s–RSC with constant probability, for any $s \leq \log(\log(N))$.*

In order to prove Theorem 3.8, we first define some notations, starting with the notations for the amplitudes. We define:

1. $f_{i,j}$ as the amplitude of the databases D containing at least j distinct $(s-1)$–RSC after i quantum queries.
2. $\widehat{g}_{i,j,k}$ as the amplitude of the databases D containing at least j distinct $(s-1)$–RSC and exactly k distinct s–RSC after i quantum queries.
3. $g_{i,k}$ as the amplitude of the databases D containing exactly k distinct s–RSC after i quantum queries.

More formally, let $|\phi_i\rangle$ (resp. $|\psi_i\rangle$) be the state of the algorithm just before (resp. after) the i^{th} query to the oracle. We have:

$$\begin{aligned}
f_{i,j} &= \left\| P^{RSC}_{j,(s-1)} |\psi_i\rangle \right\|, \\
\widehat{g}_{i,j,k} &= \left\| P^{RSC}_{j,(s-1)} P^{RSC}_{k,s} \neg P^{RSC}_{k+1,s} |\psi_i\rangle \right\|, \\
g_{i,k} &= \left\| P^{RSC}_{k,s} \neg P^{RSC}_{k+1,s} |\psi_i\rangle \right\|.
\end{aligned}$$

We want to bound $g_{i,k}$, and to do so, we define some convenient notation. We start by defining Π_s, a term that appears in the bound of $g_{i,k}$.

Definition 3.2. *Let Π_s be defined as follows:*

$$\begin{cases} \Pi_1 = 1 \\ \Pi_2 = 1 \\ \forall s \geq 2, \quad \Pi_{s+1} = 2 \cdot \sqrt{s} \cdot \sqrt{\Pi_s} \end{cases}$$

We define $A_{i,s}$ and $\mu_s(\ell)$ as follows:

Definition 3.3.

$$A_{i,s} = \sum_{\ell=0}^{i-1} B_{\ell,s-1},$$

where

$$B_{\ell,s} = \sqrt{s \cdot \frac{\mu_{s+1}(\ell)}{N}} + 4\left(\frac{\ell}{N}\right)^{s/2} + \left(\sum_{r=2}^{s} \frac{\ell}{N^r}\right)^{1/2},$$

and

$$\mu_s(\ell) = \max\left\{ \Pi_{s-1} \cdot (8e)^{\frac{2^{s-2}-1}{2^{s-3}}} \frac{\ell^{(2^{s-1}-1)/2^{s-2}}}{N^{(2^{s-2}-1)/2^{s-2}}}, 40 \cdot s^2 \cdot \Pi_{s-1} \cdot N^{1/2^s} \right\}.$$

We can now state the bound on $g_{i,k}$ that we will need to prove Theorem 3.8:

Lemma 3.10. *For every $i \in \mathbb{N}$ and every $k \in \mathbb{N}$, we have that:*

$$g_{i,k} \leq \frac{A_{i,s+1}^k}{k!} + O\left(2^{-(s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}}\right).$$

In order to prove Lemma 3.10, we first prove a bound on $A_{i,s}$.

Lemma 3.11. $A_{i,s} \leq (8e)^{\frac{2^{s-2}-1}{2^{s-2}}} \frac{i^{(2^s-1)/2^{s-1}}}{N^{(2^{s-1}-1)/2^{s-1}}} \cdot \Pi_s + O\left(s^4 \cdot \Pi_s \cdot N^{-1/(2^s(2^s-2))}\right)$

In the interest of space, we leave the proof of Lemma 3.11 to Appendix B.5, and we now prove Lemma 3.10.

Proof of Lemma 3.10. We prove this theorem by induction. The case $s = 3$ corresponds to Section 3.1.2. Fix $s \geq 3$. We assume that $f_{i,j} \leq \frac{A_{i,s}^j}{j!} + O\left(2^{-s^2 \cdot \Pi_{s-1} \cdot N^{1/2^s}}\right)$ for every $i \in \mathbb{N}$ and $j \in \mathbb{N}$. We will show that $g_{i,k} \leq \frac{A_{i,s+1}^k}{k!} + O\left(2^{-(s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}}\right)$.

Similarly to the previous subsection, we will bound $g_{i,k}$ recursively. Using Lemma 2.4, we have that:

$$\left|P_{k,s}^{RSC} |\psi_i\rangle\right| \leq \left|P_{k,s}^{RSC} |\phi_i\rangle\right| + \left|P_{k,s}^{RSC} \mathsf{cO}\left(I - P_{k,s}^{RSC}\right) |\phi_i\rangle\right|,$$

where the second term can be written as:

$$\left| P_{k,s}^{RSC} \sum_{\substack{x,\hat{y},z \\ D:(k-1) \text{ distinct } s-RSC \\ D(x)=\perp}} \frac{1}{\sqrt{N^s}} \sum_{y'} \omega_n^{yy'} \alpha_{x,\hat{y},z,D} \left| x, \hat{y}, z, D \cup (x, y') \right\rangle \right| \tag{3.14}$$

$$+ \left| P_{k,s}^{RSC} \mathsf{cO} \sum_{\substack{x,\hat{y},z \\ D:(k-1) \text{ distinct } s-RSC \\ D(x)\neq\perp}} \alpha_{x,\hat{y},z,D} \left| x, \hat{y}, z, D \right\rangle \right|. \tag{3.15}$$

To bound the term of Equation (3.14), we analyze now the possibilities for achieving a s–RSC, considering the different cases of the inner sum. We have different possible ways to get from D that does not have a s-RSC to $D_{y'} := D \cup (x, y')$ that has a s-RSC.

- $(x = x_0)$ As for the case $s = 2$, we consider the cases where we query x such that we have $x_1, \ldots, x_s$, such that $\forall 1 \leq j \leq s$, $h_s(x) = h_s(x_s)$. For every $1 \leq j \leq s$, only $i - 1$ values of y' will make $D_{y'}$ contain a collision on h_s. Thus there are at most $\frac{(i-1)^s}{N^s}$ values of y' such that $D_{y'}$ contain a new s–RSC in this case.

- $(x = x_s)$ Similarly to the case $s = 2$, we consider the case where there exists $x_0, \ldots, x_{s-1}$ such that $x_0, \ldots, x_{s-1}$ is a $(s-1)$–RSC, and we query x such that $h_j(x) = h_j(x_0)$ for some $1 \leq j \leq s$. If we have found ℓ distinct $(s-1)$–RSC in D previously, then ℓ values of y' can make $D_{y'}$ contain a s–RSC, out of the N possible values for the outcome of h_j (notice that the values of $h_i(x)$ for $i \neq j$ are not relevant for this case), and there are s different values for j.

- However, some new terms do not appear in the case of 2–RSC. That would be the case where the query x is equal to $x_{i_1} = x_{i_2} = \cdots = x_{i_r}$ for some $r \in \{2, \ldots, s\}$ in the new s–RSC. We bound these terms as follows: for each r, there is at most $(i-1)$ distinct $(s-r)$–RSC. For each of these $(s-r)$–RSC, there are r collisions missing on some $h_{i_1}, \ldots, h_{i_r}$. And exactly one value of y' will make $D_{y'}$ contain a collision for h_{i_j}. The values of the other hash functions are irrelevant here. Hence using Lemma 2.3 we can bound the probability of this event by:

$$\sum_{r=2}^{s} \frac{i-1}{N^r} {g_{i-1,k-1}}^2, \tag{3.16}$$

  where we bound the amplitude of the database containing at least one $(s-r)$–RSC and $k-1$ distinct s–RSC after $i-1$ quantum queries by $g_{i-1,k-1}$, the amplitude of the databases containing only $k-1$ distinct s–RSC after $i-1$ quantum queries.

Using Lemma 2.3, and as for the previous cases, by bounding the term of Equation (3.15) by $3\left(\frac{(i-1)}{N}\right)^{s/2} g_{i-1,k-1}$, we can upper bound $g_{i,k}$ by

$$g_{i-1,k} + \sqrt{s\sum_{\ell\geq 0}\frac{\ell}{N}\widehat{g}^2_{i-1,\ell,k-1}} + 4\sqrt{\frac{(i-1)^s}{N^s}{g_{i-1,k-1}}^2} + \sqrt{\sum_{r=2}^{s}\frac{i-1}{N^r}{g_{i-1,k-1}}^2}$$
$$\leq g_{i-1,k} + \sqrt{s\sum_{\ell\geq 0}\frac{\ell}{N}\widehat{g}^2_{i-1,\ell,k-1}} + \left(4\left(\frac{i-1}{N}\right)^{s/2} + \left(\sum_{r=2}^{s}\frac{i-1}{N^r}\right)^{1/2}\right) g_{i-1,k-1}, \qquad (3.17)$$

where the second term can be split in two, similar to the proof of Lemma 3.3:

$$\sqrt{s\sum_{\ell\geq 0}\frac{\ell}{N}\widehat{g}^2_{i-1,\ell,k-1}} \leq \sqrt{s\cdot\frac{\mu_{s+1}(i-1)}{N}} g_{i-1,k-1} + \sqrt{s}\cdot f_{i-1,\mu_{s+1}(i-1)}v$$

The term $f_{i-1,\mu_{s+1}(i-1)}$ can be bounded by induction hypothesis by:

$$f_{i-1,\mu_{s+1}(i-1)} \leq \frac{A^{\mu_{s+1}(i-1)}_{i-1,s}}{\mu_{s+1}(i-1)!} + O\left(2^{-s^2\cdot\Pi_{s-1}\cdot N^{1/2^s}}\right),$$

and the first term can be bounded by using Lemma 3.11 and the definition of $\mu_{s+1}(i-1)$ by:

$$\left(\frac{e(4e)^{\frac{2^{s-2}-1}{2^{s-2}}}\frac{i^{(2^s-1)/2^{s-1}}}{N^{(2^{s-1}-1)/2^{s-1}}}\Pi_s + O\left(s^4\Pi_s N^{-1/(2^s(2^s-2))}\right)}{\max\left\{(8e)^{\frac{2^{s-1}-1}{2^{s-2}}}\frac{i^{(2^s-1)/2^{s-1}}}{N^{(2^{s-1}-1)/2^{s-1}}}\Pi_s, 40(s+1)^2\Pi_s\cdot N^{1/2^s}\right\}}\right)^{40(s+1)^2\Pi_s N^{1/2^{s+1}}},$$

which is smaller than

$$\left(\frac{1}{2}+o(1)\right)^{40(s+1)^2\cdot\Pi_s\cdot N^{1/2^{s+1}}},$$

which leads to:

$$f_{i-1,\mu_{s+1}(i-1)} < 2^{-9.8\cdot 4\cdot(s+1)^2\cdot\Pi_s\cdot N^{1/2^{s+1}}}.$$

Using Definition 3.3, we rewrite Equation (3.17) as:

$$g_{i,k} \leq g_{i-1,k} + B_{\ell,s}\cdot g_{i-1,k-1} + \sqrt{s}\cdot 2^{-9.8\cdot 4\cdot(s+1)^2\cdot\Pi_s\cdot N^{1/2^{s+1}}}.$$

Then, by expanding the inequality and using the fact that $g_{0,k-1} = 0$, we get:

$$\begin{aligned}
g_{i,k} \leq & g_{i-1,k} + B_{\ell,s} \cdot g_{i-1,k-1} + \sqrt{s} \cdot 2^{-9.8 \cdot 4 \cdot (s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}} \\
& \vdots \\
\leq & \sum_{\ell=0}^{i-1} \left(B_{\ell,s} \cdot g_{\ell,k-1} + \sqrt{s} \cdot 2^{-9.8 \cdot 4 \cdot (s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}} \right) \\
\leq & \left(\sum_{\ell=0}^{i-1} B_{\ell,s} \cdot g_{\ell,k-1} \right) + s \cdot N^{1/2} \cdot \sqrt{s} \cdot 2^{-9.8 \cdot 4 \cdot (s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}} \\
\leq & \left(\sum_{\ell=0}^{i-1} B_{\ell,s} \cdot g_{\ell,k-1} \right) + s^{3/2} \cdot 2^{-9.5 \cdot 4 \cdot (s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}},
\end{aligned}$$

where we use the fact that $i \leq s \cdot \sqrt{N}$ for the third inequality.

Expanding this inequality, we obtain

$$g_{i,k} \leq \frac{A_{i,s+1}^k}{k!} + s^{3/2} \cdot e^{A_{i,s+1}} \cdot 2^{-9.5 \cdot 4 \cdot (s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}}. \tag{3.18}$$

For details on Equation (3.18), see Appendix B.6.

And because $i \leq s \cdot \sqrt{N}$, we have $A_{i,s+1} \leq 8e \cdot (s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}$. Using this and the fact that $s^{3/2} \leq 2^{\Pi_s \cdot (s+1)^2 \cdot N^{1/2^{s+1}}}$, we conclude:

$$g_{i,k} \leq \frac{A_{i,s+1}^k}{k!} + 2^{-(s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}}.$$

□

At last we bound Π_s to conclude the analysis.

Proposition 3.12. *We have for any $s \in \mathbb{N}$ that:*

$$\Pi_s \leq 4s$$

Proof. The statement is true for $s = 1, 2$. Assume it is true for $s \geq 2$. Then,

$$\Pi_{s+1} = 2\sqrt{s} \cdot \sqrt{\Pi_s} \leq 2\sqrt{s} \cdot \sqrt{4s} \leq 4(s+1).$$

□

Finally, we can prove Theorem 3.8:

Proof of Theorem 3.8. From Lemma 3.11, we have:

$$A_{i,s} \leq (8e)^{\frac{2^{s-2}-1}{2^{s-2}}} \frac{i^{(2^s-1)/2^{s-1}}}{N^{(2^{s-1}-1)/2^{s-1}}} \cdot \Pi_s + O\left(s^4 \cdot \Pi_s \cdot N^{-1/(2^s(2^s-2))}\right).$$

Hence we can bound $g_{i,k}$ for any i, k, by:

$$\begin{aligned}
g_{i,k} &\leq \frac{A_{i,s+1}^k}{k!} + O\left(2^{-(s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}}\right) \\
&\leq \left(\frac{e \cdot A_{i,s+1}}{k}\right)^k + O\left(2^{-(s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}}\right) \\
&\leq \left(\frac{e}{k}(8e)^{\frac{2^{s-1}-1}{2^{s-1}}} \frac{i^{(2^{s+1}-1)/2^s}}{N^{(2^s-1)/2^s}} \cdot \Pi_{s+1} + \frac{e}{k} \cdot O\left((s+1)^4 \Pi_{s+1} \cdot N^{-1/(2^{s+1}(2^{s+1}-2))}\right)\right)^k \\
&\quad + O\left(2^{-(s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}}\right) \\
&\leq \left(\frac{e}{k} \cdot (8e)^{\frac{2^{s-1}-1}{2^{s-1}}} \frac{i^{(2^{s+1}-1)/2^s}}{N^{(2^s-1)/2^s}} \cdot 4(s+1) + \frac{e}{k} \cdot O\left(4(s+1)^5 \cdot N^{-1/(2^{s+1}(2^{s+1}-2))}\right)\right)^k \\
&\quad + O\left(2^{-4s(s+1)^2 \cdot N^{1/2^{s+1}}}\right),
\end{aligned}$$

where the first inequality comes from Lemma 3.10, the third inequality comes from Lemma 3.11 and the last inequality comes from Proposition 3.12.

If $i = o\left((s+1)^{-\frac{2^s}{2^{s+1}-1}} \cdot k^{\frac{2^s}{2^{s+1}-1}} \cdot N^{\frac{2^s-1}{2^{s+1}-1}}\right)$, then $g_{i,k} = o(1)$. Hence if we want $g_{i,k}$ to be constant, i.e. not $o(1)$, we must have $i = \Omega\left(s^{-\frac{2^s}{2^{s+1}-1}} \cdot k^{\frac{2^s}{2^{s+1}-1}} \cdot N^{\frac{2^s-1}{2^{s+1}-1}}\right)$. □

3.2 The (r, k)–Subset Cover Problem

In this section, we prove some upper and lower bounds on the (r, k)–SC problem. As far as we know, there is no quantum algorithm to find a (r, k)–SC problem, except for [YTA22]'s algorithm when $k = r$, and for the harder problem of finding a k–RSC. We first prove a lower bound on the $(1, k)$–SC problem, then design new algorithms for finding a (r, k)–SC.

3.2.1 Lower Bound on Finding a $(1, k)$–Subset Cover

In this subsection, we will prove a lower bound on the $(1, k)$–SC problem. We are given k random functions $h_1, \ldots, h_k$ such that for $i \in [1, k]$, $h_i : \mathcal{X} \to \mathcal{Y}$. We write $N = |\mathcal{Y}|$ and for $x \in \mathcal{X}$, we write $H(x) = \{h_i(x) | i \in [1, k]\}$. The goal of this subsection is to prove the following theorem.

Theorem 3.13. *Given k random functions $h_1, \ldots, h_k : \mathcal{X} \to \mathcal{Y}$ where $N = |\mathcal{Y}|$, a quantum algorithm needs to make $\Omega\left(C_k^{-1/5} \cdot N^{k/5}\right)$ queries to $h_1, \ldots, h_k$ to find one (1,k)–SC with constant probability, where $C_k = \sum_{j=2}^k \frac{k!}{(j-1)!}$.*

To prove Theorem 3.13, we introduce the problem of finding a *j–repetition* on $h_{i_1}, \ldots, h_{i_j}$, that consists in finding an $x \in \mathcal{X}$ such that $h_{i_1}(x) = \cdots = h_{i_j}(x)$. More formally, we define the following database property:

Definition 3.4.

$$\forall \ell, j, P^{rep}_{\ell,j} = \left\{ D \in \mathcal{D} \;\middle|\; \begin{array}{l} \exists x_1, x_2, \ldots, x_\ell, \forall i, \forall 1 \leq \ell \leq j, h_1(x_i) = h_\ell(x_i) \\ \forall i \neq p, x_i \neq x_p \end{array} \right\}.$$

Note that we define the property only for ℓ distinct j–repetition on $h_1, \ldots, h_j$, because by symmetry, the probability of finding a j–repetition on $h_1, \ldots, h_j$ is the same as finding a j–repetition on $h_{i_1}, \ldots, h_{i_\ell}$.

We also define:

1. $\widetilde{f}^{rep}_{i,\ell,j}$ as the amplitude of the databases D containing *at least* ℓ distinct j–repetitions on $h_1, \ldots, h_j$ after i quantum queries.

2. $f^{rep}_{i,\ell,j}$ as the amplitude of the databases D containing *exactly* ℓ distinct j–repetitions on $h_1, \ldots, h_j$ after i quantum queries.

3. $g_{i,k}$ as the amplitude of the databases D containing at least one $(1,k)$–SC after i quantum queries.

More formally, let $|\psi_i\rangle$ be the state just after the i^{th} query to the oracle, then $\widetilde{f}^{rep}_{i,\ell,j} = \left\| P^{rep}_{\ell,j} |\psi_i\rangle \right\|$, $f^{rep}_{i,\ell,j} = \left\| P^{rep}_{\ell,j} \neg P^{rep}_{\ell+1,j} |\psi_i\rangle \right\|$, and $g_{i,k} = \left\| P^{SC}_{(1,k)} |\psi_i\rangle \right\|$.

Our goal is to bound $g_{i,k}$ and for that we will bound $\widetilde{f}^{rep}_{i,\ell,j}$.

Lemma 3.14. *For all $i, \ell, j \in \mathbb{N}$, we have that:*

$$\widetilde{f}^{rep}_{i,\ell,j} \leq \left(\frac{4e \cdot i}{\ell \cdot N^{\frac{j-1}{2}}} \right)^{\ell}.$$

Proof. Following the proof of Lemma 3.5, we have that:

$$\begin{aligned}
\widetilde{f}^{rep}_{i,\ell,j} &\leq \widetilde{f}^{rep}_{i-1,\ell,j} + \sqrt{\frac{1}{N^{j-1}}\widetilde{f}^{rep}_{i-1,\ell-1,k}{}^2} + \frac{3(i-1)}{N^j}\widetilde{f}^{rep}_{i-1,\ell-1,k} \\
&\leq \widetilde{f}^{rep}_{i-1,\ell,j} + 4\sqrt{\frac{1}{N^{j-1}}\widetilde{f}^{rep}_{i-1,\ell-1,k}{}^2} \\
&\leq \sum_{m=0}^{i-1} 4\sqrt{\frac{1}{N^{j-1}}}\widetilde{f}^{rep}_{m,\ell-1,k} \\
&\leq \sum_{m_1=0}^{i-1}\sum_{m_2=0}^{m_1} 4\sqrt{\frac{1}{N^{j-1}}}4\sqrt{\frac{1}{N^{j-1}}}\widetilde{f}^{rep}_{m_2,\ell-2,k} \\
&\vdots \\
&\leq \sum_{0\leq m_\ell<m_{\ell-1}<\cdots<m_1<i} \left(\frac{16}{N^{j-1}}\right)^{\ell/2} \\
&\leq \frac{i^\ell}{\ell!}\left(\frac{16}{N^{j-1}}\right)^{\ell/2} \\
&\leq \left(\frac{4e\cdot i}{\ell\cdot N^{(j-1)/2}}\right)^\ell,
\end{aligned}$$

where the second inequality comes from the fact that we can assume $i \leq N^{j/2}$. □

We now bound the amplitude $g_{i,k}$ with an inductive formula, as for the RSC problem.

Lemma 3.15. *For all $i \in \mathbb{N}$ and $k \in \mathbb{N}$, we have that:*

$$g_{i,k} \leq g_{i-1,k} + 4\left(k^k\frac{i-1}{N^k}\right)^{1/2} + \left(\sum_{j=2}^{k}\sum_{\ell\geq 0}\frac{\ell}{N^{k+1-j}}\cdot\frac{k!}{(j-1)!}f^{rep}_{i-1,\ell,j}{}^2\right)^{1/2}.$$

Proof. For convenience, we denote $P_k = P^{SC}_{(1,k)}$ the projector on the databases D that contain at least a $(1,k)$–SC. We write $|\phi_i\rangle$ the state just before the i^{th} quantum query, and $|\psi_i\rangle$ the state just after the i^{th} quantum query.

Using Lemma 2.4, and writing $D_{y'} := D \cup (x, y')$ we have that:

$$|P_k|\psi_i\rangle| \leq |P_k|\phi_i\rangle| + \left|P_k \sum_{\substack{x,\hat{y},z \\ D\text{:no (1,k)–SC}}} \frac{1}{\sqrt{N^k}}\sum_{y'}\omega_N^{yy'}\alpha_{x,\hat{y},z,D}|x,\hat{y},z,D_{y'}\rangle\right| + 3\frac{i-1}{N^k} \quad (3.19)$$

We analyze now the possibilities for achieving a $(1,k)$–SC, considering the different cases of the inner sum. We have multiple possible ways to get from D that does not have a $(1,k)$–SC to $D_{y'}$ that has a $(1,k)$–SC.

- $(x = x_0)$ Here, we consider the case where we query x such that $\{h_i(x)\} \subseteq \{h_i(x_1)\}$, where x_1 was queried before. Notice that there are $(i-1)$ possible values of x_1, and for each fixed value of x_1, we have k^k possible values of $H(x)$ that would lead to this value. This leads to $k^k(i-1)$ possible values of y' that would lead to an $(1,k)$-SC.
- $(x = x_1)$ Here, we consider the case where we query x such that $\{h_i(x_0)\} \subseteq \{h_i(x)\}$, where x_0 was queried before.

  Let us suppose that x_0 has a j-repetition on $h_{i_1}, \ldots, h_{i_j}$, for some distinct $i_1, \ldots, i_j$. Notice that in this case, $S := \{h_i(x_0)\}$ has $k-j+1$ elements and we will count the number of possible $H(x)$ that contains all of these elements. Out of the k functions $h_1, \ldots, h_k$, we have $\binom{k}{k-j+1}$ possible ways of choosing the functions that will be filled with the values in S. When we fix such functions, there are $|S|! = (k-j+1)!$ ways of filling them with the elements of S, and N^{j-1} ways of filling the other functions. Therefore, there are $\binom{k}{k-j+1}(k-j+1)!N^{j-1}$ values of $H(x)$ such that $\{h_i(x_0)\} \subseteq \{h_i(x)\}$.

This gives, bounding the last term of Equation (3.19) by $3\left(k^k \frac{i-1}{N^k}\right)^{1/2}$:

$$|P_k|\psi_i\rangle| \leq |P_k|\phi_i\rangle| + 4\left(k^k \frac{i-1}{N^k} \sum_{\substack{x,\hat{y},z \\ D:\text{no k–SC}}} |\alpha_{x,\hat{y},z,D}|^2\right)^{1/2} + \left(\sum_{j=2}^{k} \sum_{\ell \geq 0} \frac{\ell}{N^{k+1-j}} \cdot \frac{k!}{(j-1)!} \sum_{\substack{x,\hat{y},z \\ D:\text{no k–SC} \\ \ell \text{ distinct } j-repetitions}} |\alpha_{x,\hat{y},z,D}|^2\right)^{1/2}.$$

Using Lemma 2.3 and our notations, we conclude:

$$g_{i,k} \leq g_{i-1,k} + 4\left(k^k \frac{i-1}{N^k}\right)^{1/2} + \left(\sum_{j=2}^{k} \sum_{\ell \geq 0} \frac{\ell}{N^{k+1-j}} \cdot \frac{k!}{(j-1)!} {f^{rep}_{i-1,\ell,j}}^2\right)^{1/2}.$$

□

We now bound $g_{i,k}$ in the following lemma.

Lemma 3.16. *For every $i \in \mathbb{N}$ and $k \in \mathbb{N}$, we have that:*

$$g_{i,k} \leq 4k^{k/2} \cdot \frac{i^{3/2}}{N^{k/2}} + \sqrt{\sum_{j=2}^{k} \frac{k!}{(j-1)!}} \cdot \frac{4e \cdot i^{5/2}}{N^{k/2}}.$$

Proof. From Lemma 3.15, we have that:

$$g_{i,k} \leq g_{i-1,k} + 4\left(k^k \frac{i-1}{N^k}\right)^{1/2} + \left(\sum_{j=2}^{k}\sum_{\ell\geq 0} \frac{\ell}{N^{k+1-j}} \cdot \frac{k!}{(j-1)!} {f_{i-1,\ell,j}^{rep}}^2\right)^{1/2}.$$

We want to bound each term in the sum indexed by j. Fix $j \in \{2,\ldots,k\}$. We have that:

$$\sum_{\ell\geq 0} \frac{\ell}{N^{k+1-j}} \cdot \frac{k!}{(j-1)!} {f_{i-1,\ell,j}^{rep}}^2 = \frac{k!}{(j-1)!} \cdot \sum_{\ell\geq 0} \frac{\ell}{N^{k+1-j}} {f_{i-1,\ell,j}^{rep}}^2.$$

Next, we have that:

$$\begin{aligned}
\sum_{\ell\geq 0} \frac{\ell}{N^{k+1-j}} {f_{i-1,\ell,j}^{rep}}^2 &\leq \frac{i-1}{N^{k+1-j}} \cdot \sum_{\ell\geq 1} {f_{i-1,\ell,j}^{rep}}^2 \\
&= \frac{i-1}{N^{k+1-j}} \cdot {\widetilde{f}_{i-1,1,j}^{rep}}^2 \\
&\leq \frac{i-1}{N^{k+1-j}} \cdot \left(\frac{4e\cdot(i-1)}{N^{\frac{j-1}{2}}}\right)^2 \\
&= \frac{(4e)^2(i-1)^3}{N^k},
\end{aligned}$$

where $\widetilde{f}_{i-1,1,j}^{rep}$ is the amplitude of the databases D containing *at least* one j–repetition on $h_1,\ldots,h_j$ after $i-1$ quantum queries. The first inequality follows since there cannot be more than $i-1$ distinct j–repetitions on $h_1,\ldots,h_j$ after $i-1$ quantum queries. The second inequality comes from the bound on $\widetilde{f}_{i-1,1,j}^{rep}$ in Lemma 3.14.

This gives:

$$\left(\sum_{j=2}^{k} \frac{k!}{(j-1)!} \sum_{\ell\geq 0} \frac{\ell}{N^{k+1-j}} {f_{i-1,\ell,j}^{rep}}^2\right)^{1/2} \leq \sqrt{\sum_{j=2}^{k} \frac{k!}{(j-1)!}} \cdot \frac{4e\cdot(i-1)^{3/2}}{N^{k/2}}.$$

Finally, by developing the recursive terms (and using that $g_{0,k}=0$), we get that:

$$\begin{aligned}
g_{i,k} &\leq g_{i-1,k} + 4\sqrt{k^k \frac{i-1}{N^k}} + \sqrt{\sum_{j=2}^{k} \frac{k!}{(j-1)!}} \cdot \frac{4e\cdot(i-1)^{3/2}}{N^{k/2}} \\
&\vdots \\
&\leq \sum_{\ell=0}^{i-1} \left(4\sqrt{k^k \frac{\ell}{N^k}} + \sqrt{\sum_{j=2}^{k} \frac{k!}{(j-1)!}} \cdot \frac{4e\cdot\ell^{3/2}}{N^{k/2}}\right) \\
&\leq 4k^{k/2}\frac{i^{3/2}}{N^{k/2}} + \sqrt{\sum_{j=2}^{k} \frac{k!}{(j-1)!}} \cdot \frac{4e\cdot i^{5/2}}{N^{k/2}}.
\end{aligned}$$

□

We can now prove Theorem 3.13.

Proof of Theorem 3.13. From Lemma 3.16, we have that:

$$g_{i,k} \leq 4k^{k/2} \cdot \frac{i^{3/2}}{N^{k/2}} + \sqrt{\sum_{j=2}^{k} \frac{k!}{(j-1)!}} \cdot \frac{4e \cdot i^{5/2}}{N^{k/2}}.$$

Writing $C_k = \sum_{j=2}^{k} \frac{k!}{(j-1)!}$, this rewrites as:

$$g_{i,k} \leq 4k^{k/2} \cdot \frac{i^{3/2}}{N^{k/2}} + \sqrt{C_k} \cdot \frac{4e \cdot i^{5/2}}{N^{k/2}}.$$

If $i = o\left(C_k^{-1/5} \cdot N^{k/5}\right)$, then $g_{i,k} = o(1)$. Hence if we want $g_{i,k}$ to be constant, i.e. not $o(1)$, we must have $i = \Omega\left(C_k^{-1/5} \cdot N^{k/5}\right)$. □

3.2.2 Algorithm for Finding a $(1,k)$–Subset Cover

We now describe an algorithm that finds a $(1,k)$–SC, assuming $|\mathcal{X}| = |\mathcal{Y}|^k = N^k$. We first notice that an algorithm that finds a collision on H also finds a $(1,k)$–SC in an expected $O(N^{k/3})$ number of queries. We show now that there is a more efficient algorithm, as stated in the following theorem:

Theorem 3.17. *There exists a quantum algorithm that finds a $(1,k)$–SC in expected $O\left(N^{k/4}\right)$ quantum queries if k is even, and $O(N^{k/4+1/12})$ if k is odd.*

To prove this theorem, we describe the following algorithm (which takes as parameters j and t, whose values will be chosen later):

Algorithm 3.1. *Input: $j \in \{2, \ldots, k\}$ and $t \in \mathbb{N}$.*

1. *Define $F_1 : \mathcal{X} \to \{0,1\}$ as follows:*

$$F_1(x) = \begin{cases} 1, & \text{if } h_1(x) = h_2(x) = \cdots = h_j(x) \\ 0, & \text{otherwise.} \end{cases}$$

   *(Note that an element $x \in \mathcal{X}$ such that $F_1(x) = 1$ is a j–*repetition*.)*

2. *Execute Grover's algorithm t times on F_1 to find t distinct j–repetitions in H. Let $T = \{x_1, \ldots, x_t\}$ be the set of these j–repetitions.*

3. *Define $F_2 : \mathcal{X} \to \{0,1\}$ as follows:*

$$F_2(x) = \begin{cases} 1, & \text{if there exists } x_0 \in T \text{ such that } h_1(x) = h_1(x_0) \\ & \text{and for } 1 \leq m \leq k-j, h_{m+1}(x) = h_{j+m}(x_0) \\ 0, & \text{otherwise.} \end{cases}$$

4. *Execute Grover's algorithm to find an x such that $F_2(x) = 1$*
5. *Find x_0 in T corresponding to x, and output (x, x_0).*

Lemma 3.18. *Algorithm 3.1 makes an expected number of $O\left(N^{(2k-j+1)/6}\right)$ queries to the oracle when $j \leq \frac{k+2}{2}$ for $t = N^{(k-2j+2)/3}$.*

Proof. Notice that if we consider a uniformly random function, we have that $Pr[h_1(x) = \cdots = h_j(x)] = N^{-j+1}$. Therefore, the expected number of elements in $\mathcal{X}$ such that $F_1(x) = 1$ is $N^k \cdot N^{-j+1} = N^{k-j+1}$. We write $X_1, \ldots, X_{N^k}$ the random variables corresponding to F_1's output on each $x \in \mathcal{X}$, X the sum of these variables, $\mu = N^{k-j+1}$ their mean. Chernoff bound tells us that for any $0 \leq \delta \leq 1$,

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

With $\delta = 1/2$, we have:

$$Pr\left(|X - \mu| \geq \frac{\mu}{2}\right) \leq e^{-\mu/12}.$$

Thus, unless with probability $e^{-(N^{k-j+1})/12}$, the number of elements $x \in \mathcal{X}$ such that $F_1(x) = 1$ is greater than $N^{k-j+1}/2$.

Hence using Theorem A.1, the second step of the algorithm is expected to make $O\left(t \cdot \sqrt{\frac{N^k}{N^{k-j+1}}}\right) = O\left(t \cdot N^{(j-1)/2}\right)$ quantum queries to the oracle.

Notice that for a fixed value x_0, if we consider a uniformly random function, we have that

$$Pr[h_1(x) = h_1(x_0) \wedge h_2(x) = h_{m+1}(x_0) \wedge \cdots \wedge h_{k-j+1}(x) = h_k(x)] = N^{j-k-1}.$$

Therefore, the expected number of elements such that $F_2(x) = 1$ is $t \cdot N^k \cdot N^{j-k-1} = t \cdot N^{j-1}$. Similarly, using Chernoff bound, unless with probability $e^{-(t \cdot N^{j-1})/12}$, the number of elements such that $F_2(x) = 1$ is greater than $t \cdot N^{j-1}/2$. Hence, using Theorem A.1, the fourth step of the algorithm is expected to make $O\left(\sqrt{\frac{N^k}{t \cdot N^{j-1}}}\right) = O\left(\frac{N^{(k-j+1)/2}}{\sqrt{t}}\right)$ quantum queries to the oracle.

By picking $t = N^{(k-2j+2)/3}$ with $j \leq \frac{k+2}{2}$ (otherwise $t < 1$), the complexity of the algorithm is $O(N^{(k-2j+2)/3} \cdot N^{(j-1)/2}) = O(N^{(2k-j+1)/6})$. □

We now prove Theorem 3.17

Proof of Theorem 3.17. From Lemma 3.18, the complexity of Algorithm 3.1 is $O(N^{(2k-j+1)/6})$ when $j \leq \frac{k+2}{2}$.

- If k is even, then we pick $j = \frac{k+2}{2}$ to reach a complexity of $O(N^{k/4})$.
- If k is odd, then we pick $j = \frac{k+1}{2}$ to reach a complexity of $O(N^{k/4+1/12})$.

Note that if $j > \frac{k+1}{2}$, then the second step of the algorithm is expected to make at least $O\left(N^{\frac{k+1}{4}}\right)$ quantum queries, which is worse than $O(N^{k/4+1/12})$. □

Remark 3.1. *Note that we do not reach the lower bound of Theorem 3.13, and it would be interesting to see if the gap can be further reduced by either improving our lower bounds or designing a more efficient algorithm.*

A slightly better algorithm We describe a more efficient algorithm when k is not constant. The idea is to take into account the fact that we do not necessarily need the j–repetitions from the previous algorithm to occur on the first j functions $h_1, \ldots, h_j$, but they could rather be on any $h_{i_1}, \ldots, h_{i_j}$ instead. We also consider permutations of the $h_1, \ldots, h_k$ in the fourth step of Algorithm 3.1.

Theorem 3.19. *There exists a quantum algorithm that finds a $(1,k)$–SC in:*

- $O\left(\binom{k}{(k+2)/2}^{-1/2} \cdot N^{k/4}\right)$ *quantum queries if k is even,*
- $O\left(\binom{k}{(k+1)/2}^{-1/2} \cdot N^{k/4+1/12}\right)$ *quantum queries if k is odd.*

The gain that we obtain is a function of k and is therefore not significant if k is constant. However, as we have shown in Theorem 3.13, the dependence in k can be quite large for the $(1,k)$–SC problem.

To prove this theorem, we describe the algorithm as follows (which takes again as input two integers j and t playing the role of parameters whose optimal values will be determined later):

Algorithm 3.2. *Input: $j \in \{2, \ldots, k\}$ and $t \in \mathbb{N}$.*

1. *Define $F_1 : \mathcal{X} \to \{0,1\}$ as follows:*

$$F_1(x) = \begin{cases} 1, & \text{if there exists distinct } i_1, \ldots, i_j \in [1,k] \text{ such that} \\ & h_{i_1}(x) = h_{i_2}(x) = \cdots = h_{i_j}(x) \\ 0, & \text{otherwise.} \end{cases}$$

   *(Note that an element $x \in \mathcal{X}$ such that $F_1(x) = 1$ is a j–*repetition*.)*

2. *Execute Grover's algorithm t times on F_1 to find t distinct j–repetitions in H. Let $T = \{x_1, \ldots, x_t\}$ be the set of these j–repetitions. We write, for $\ell \in [1,t]$ $I_\ell = \{i_1^\ell, \ldots, i_j^\ell\}$ the set of indices such that $h_{i_1^\ell}(x_t) = \cdots = h_{i_k^\ell}(x_t)$, and $I'_\ell = [1,k] \backslash I_\ell = \{i_{j+1}^\ell, \ldots, i_k^\ell\}$.*

3. *Define $F_2 : \mathcal{X} \to \{0,1\}$ as follows:*

$$F_2(x) = \begin{cases} 1, & \text{if there exists distinct } j_0, j_1, \ldots, j_{k-j+1} \in [1,k], \\ & \quad \text{and } \ell \in [1,t] \text{ s.t. } h_{i_1^\ell}(x_\ell) = h_{j_0}(x) \\ & \quad \text{and for all } 1 \le m \le k-j, h_{j_m}(x) = h_{i_{j+m}^\ell}(x_\ell) \\ 0, & \text{otherwise.} \end{cases}$$

4. *Execute Grover's algorithm to find an x such that $F_2(x) = 1$*
5. *Find x_0 in T, and output (x, x_0).*

Remark 3.2. *F_1 (resp. F_2) can be constructed with $O\left(\binom{k}{j}\right)$ (resp. $O\left(\frac{k!}{(j-1)!}\right)$) quantum gates and one query to H.*

Lemma 3.20. *Algorithm 3.2 makes an expected number of $O\left(\binom{k}{j}^{-1/2} N^{(2k-j+1)/6}\right)$ queries to the oracle when $j \leq \frac{k+2}{2}$ for $t = N^{(k-2j+2)/3}$.*

The proof of Lemma 3.20 is given in Appendix B.7. We now prove Theorem 3.19.

Proof of Theorem 3.19. From Lemma 3.20, the complexity of Algorithm 3.2 is $O\left(\binom{k}{j}^{-1/2} \cdot N^{(2k-j+1)/6}\right)$ when $j \leq \frac{k+2}{2}$.

- If k is even, for $j = \frac{k+2}{2}$, we get a complexity of $O\left(\binom{k}{(k+2)/2}^{-1/2} \cdot N^{k/4}\right)$.
- If k is odd, for $j = \frac{k+1}{2}$, we get a complexity of $O\left(\binom{k}{(k+1)/2}^{-1/2} \cdot N^{k/4+1/12}\right)$.

Note that if $j > \frac{k+1}{2}$, then the second step of the algorithm is expected to make at least $O\left(\binom{k}{(k+1)/2}^{-1/2} \cdot N^{\frac{k+1}{4}}\right)$ quantum queries. □

3.2.3 Algorithm for Finding a (r, k)–Subset Cover

In this section, we describe an algorithm for solving the (r, k)–SC problem. We consider the case where $|\mathcal{X}| = |r \cdot \mathcal{Y}|^k = r^k \cdot N^k$. The result is stated as follows:

Theorem 3.21. *There exists a quantum algorithm that finds a (r, k)–SC in $O\left(N^{k/(2+2r)}\right)$ quantum queries to H, if k is divisible by $r + 1$, and $O\left(N^{k/(2+2r)+1/2}\right)$ otherwise.*

The idea of the algorithm is essentially the same as Algorithm 3.1 of Section 3.2.2:

1. we first find t distinct $(r-1, k')$–SC for some integers t and k';
2. we then find the (r, k)–SC.

The first step is done recursively, using the algorithm defined for lower values of k' and $r-1$. The second step uses Grover's algorithm. The algorithm can be defined for any value of k' and t, and we pick them to optimize the complexity.

More formally, we define the algorithm recursively. Assume that we have an algorithm that can output a $(r-1, k')$–SC in $O\left(N^{k'/2r}\right)$ queries, for any $k' < k$ such that k' is divisible by r. Then, we can find a (r, k)–SC as follows:

Algorithm 3.3. *Input: $t \in \mathbb{N}$, $k' \in \mathbb{N}$.*

1. *Execute the $(r-1, k')$–SC algorithm t times to find t distinct $(r-1, k')$–SC in H. Let $T = \{(x_{1,0}, x_{1,1}, \ldots, x_{1,r-1}), \ldots, (x_{t,0}, x_{t,1}, \ldots, x_{t,r-1})\}$ be the set of these $(r-1, k')$–SC.*

2. *Define* $F : \mathcal{X} \to \{0,1\}$ *as follows:*

$$F(x) = \begin{cases} 1, & \text{if there exists } (x_{i,0}, x_{i,1}, \ldots, x_{i,r-1}) \in T \text{ such that} \\ & \forall 1 \leq m \leq k - k', h_m(x) = h_{k'+m}(x_{i,0}), \\ 0, & \text{otherwise.} \end{cases}$$

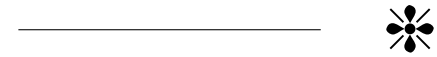
3. *Execute Grover's algorithm to find an* x *such that* $F(x) = 1$

4. *Find* $(x_{i,0}, x_{i,1}, \ldots, x_{i,r-1})$ *in* T *and output* $(x_{i,0}, x_{i,1}, \ldots, x_{i,r-1}, x)$.

Lemma 3.22. *Algorithm 3.3 makes an expected number of* $O\left(N^{k/(2+2r)}\right)$ *queries to the oracle, when* k *is divisible by* r*, and* $O\left(N^{k/(2+2r)+1/2}\right)$ *otherwise.*

We defer the proof of Lemma 3.22 to Appendix B.8.

Chapter 4



Key Agreements and Public Key Encryption from One-Way Functions

In this chapter, we study the security of the subset cover problem and its variant. We show a lower bound for the k-restricted subset cover problem that matches the upper bound of [YTA22]. We show a lower bound for the $(1, k)$-subset cover problem, and present quantum algorithms for the (r,k)-subset cover problem.

4.1 Impossibility of Key Agreement with Classical Queries

In this section, we are interested in key agreement protocols with classical communication, which we define as follow.

Definition 4.1 (Key agreement protocols with classical communication)**.** *We say that* (A, B) *is a key agreement protocol between two parties Alice and Bob with classical communication (CC-KA) if the following holds:*

1. ***Setup.*** *At the beginning of the protocol, Alice and Bob share no common information. Their corresponding algorithms,* A *and* B*, are stateful oracle-aided quantum algorithms which make at most d oracle queries.*

2. ***Classical Communication.*** *All of the messages are classical messages. The transcript of the protocol is denoted as* $T \coloneqq (m_1, \cdots, m_\ell)$*.*

3. ***Completeness.*** *At the end of the protocol, Alice and Bob agree on a key* $\mathsf{k} \in \{0, 1\}$ *with probability p when the protocol succeeds (i.e. when neither Alice or Bob outputs* $k = \perp$*).*

4. ***Security.*** *Let* $(T, \mathsf{k}_\mathsf{A}, \mathsf{k}_\mathsf{B}) \leftarrow \langle \mathsf{A} \models \mathsf{B} \rangle$*, be the output of an execution of the protocol* $T := (m_1, \cdots, m_\ell)$ *is the transcript of the execution,* k_A *and* k_B *are the keys* A *and* B*, respectively.* (A, B) *is secure if for any polynomially-bounded query adversary* $\mathcal{E}$*:*

$$\Pr\left[\begin{array}{c|c} \mathsf{k} = \mathsf{k}_\mathsf{A} = \mathsf{k}_\mathsf{B} & (T, \mathsf{k}_\mathsf{A}, \mathsf{k}_\mathsf{B}) \leftarrow \langle \mathsf{A} \models \mathsf{B} \rangle \\ \mathsf{k}_\mathsf{A}, \mathsf{k}_\mathsf{B} \neq \perp & \mathsf{k} \leftarrow \mathcal{E}(1^\lambda, T) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

We say that a CC-KA protocol (A, B) *is* (ε, s)*-broken if there exists an attacker Eve that finds the key of* (A, B) *with probability at least* ε*,* (A, B) *succeeds with probability at least* $\mathsf{poly}(\varepsilon)$*, and Eve makes an expected number of queries at most* s*.*

We show that we are able to use the Theorem 6.4 of [IR89] to prove that there exists an attacker that can find the key of any CC-KA protocol with classical queries to the random oracle. The proof consists of showing that a CC-KA protocol with classical queries to a random oracle can be simulated by a classical algorithm that query a PSPACE oracle and the random oracle.

Theorem 4.1. *Let* $\Pi = (\mathsf{A}, \mathsf{B})$ *be CC-KA protocol relative to a random oracle* $\mathcal{O}$*, where* A *and* B *makes classical queries to* $\mathcal{O}$*. If the protocol has completeness* $1 - p$*, then there exists a* PSPACE *attacker Eve that finds the key with probability* $1 - p$*.*

Proof. From $\Pi = (\mathsf{A}, \mathsf{B})$, we can construct A' such that A' is a classical algorithm that queries a PSPACE oracle, and the distribution of the algorithms A and A' are the same. First of all, because the queries to $\mathcal{O}$ are classical, then before every oracle query there must be a measurement. Since $\mathsf{BQP} \subseteq \mathsf{PSPACE}$, A' simulates A by querying the PSPACE oracle to generate every query to $\mathcal{O}$. It follows that A' can simulate the first round of the protocol. There is a technicality in subsequent rounds, because it is possible that A has an internal quantum states (that cannot be represented in polynomial space). Fortunately, by a result of [Aar05], $\mathsf{PostBQP} = \mathsf{PP} \subseteq \mathsf{PSPACE}$, hence A' can simulate the distribution of A postselected on the transcript by querying the PSPACE oracle. Similarly, B can be simulated classically with a PSPACE oracle, and we write B' this simulation.

With $\Pi' = (\mathsf{A}', \mathsf{B}')$, the proofs of [IR89] work, because they relativize. So, there exists an attacker Eve for the protocol $\Pi' = (\mathsf{A}', \mathsf{B}')$, and the distribution of the protocol is the same a $\Pi = (\mathsf{A}, \mathsf{B})$, hence Eve breaks the security of Π. □

We note that the attacker makes $O(n^6 \log n)$ queries to $\mathcal{O}$, where n is the number of queries that Alice and Bob makes to $\mathcal{O}$ during the protocol.

At last, we have the following corollary.

Corollary 4.2. *There exists an oracle relative to which one-way functions exists, but CC-KA with classical queries to the one-way functions do not exist.*

Remark 4.1. *The attack requires* $O(n^6 \log n)$ *queries to* $\mathcal{O}$*, which is not tight. An interesting open question is whether the attack of [BM09] can be adapted to this setting, which would reduce the query complexity to* $O(n^2)$*. Note that when Alice and Bob are fully classical, the*

best known attack is the classical one of [BM09], which uses $O(n^2)$ queries to the oracle. On the other hand, the classical Merkle-puzzle protocol [GRM78] can be broken with $O(n)$ quantum queries. In [BHK+19], the authors propose a new protocol parameterized by any $\varepsilon > 0$, for which the best quantum attacker requires $O(n^{1.5-\varepsilon})$ queries. However, it is unknown whether there exists a classical protocol such that any quantum attack requires $O(n^2)$ queries to the oracle, or whether there is an exponent $s < 2$ so that every classical key-exchange can be broken in $O(n^s)$ quantum queries. More generally, it remains open how quantum queries can be exploited to improve attacks on arbitrary key-exchange protocols.

4.2 Towards the Impossibility of Quantum Public Key Encryption with Classical Keys from One-Way Functions

In this section, we consider key agreement protocols in an extended setting where both parties are quantum algorithms but they can only send classical strings over the public authenticated channel to the other party, except that the last message in the protocol can be a quantum state (in this case, the last message is not authenticated). We call this the Classical Communication One Quantum Message (CC1QM) model. In this extended setting, we show a conditional result based on the polynomial compatibility conjecture, that any protocol in the CC1QM model with perfect completeness where Alice does not query the oracle after receiving the last message can be broken with an expected polynomial number of queries. We present the formal definition of key agreement protocols in the CC1QM model in section 4.2.1. In section 4.2.2, we state the main result of the section and its proof.

4.2.1 Preparation

We start by defining the model of Classical Communication One Quantum Message, where two quantum parties (Alice and Bob) communicate using the public authenticated classical channel, except for the last message that can be quantum. We assume the first message is from Alice to Bob, while the last message is from Bob to Alice, and the last quantum message is non-authenticated. This can be assumed without loss of generality since if the first message is from Bob to Alice, we can always transform it into the other case, by letting Alice sends a dummy message to Bob for the first message. Furthermore, we consider the case where the key that Alice and Bob agree on is one bit and the protocol succeeds with probability 1 (i.e., perfect correctness). Also, as for Quantum Key Distribution (QKD), we allow the parties to abort the protocol at any time, if they detect suspicious activity in the quantum communication. Formally, this is done by making Alice output the character $\perp$ instead of a key when the protocol is aborted. More formally, we define:

Definition 4.2 (Key agreement protocols in the CC1QM model)**.** *We say that* (A, B) *is a key agreement protocol between two parties Alice and Bob in the CC1QM model (CC1QM-KA)*

if the following holds:

1. *At the beginning of the protocol, Alice and Bob share no common information. Their corresponding algorithms,* A *and* B*, are stateful oracle-aided quantum algorithms which make at most* d *oracle queries.*
2. ***CC1QM.*** *All of the messages are classical messages, except for the last message (from Bob to Alice) that can be a (mixed) quantum state, denoted as* ψ*. The transcript of the protocol is denoted as* $T := (m_1, \cdots, m_\ell, \psi)$*.*
3. ***Perfect completeness.*** *At the end of the protocol, Alice and Bob agree on a key* $\mathsf{k} \in \{0,1\}$ *with probability* 1 *when the protocol succeeds (i.e. when neither Alice or Bob outputs* $k = \bot$*).*
4. ***Security.*** *Let* $\mathsf{A}'_{\mathsf{fin}}$ *be Alice's last computation in the protocol after she receives the final message from Bob. By deferred measurement principle, we can modify* $\mathsf{A}'_{\mathsf{fin}}$ *so that it applies a unitary transformation* $\mathsf{A}_{\mathsf{fin}}$ *followed by a measurement in the computational basis* $\{\Pi_{\mathsf{k}}\}_{\mathsf{k}\in\{0,1\}}$ *and outputting a key* k*, and we write* $\mathsf{A}'_{\mathsf{fin}} := \Pi_{\mathsf{k}}\mathsf{A}_{\mathsf{fin}}$*. Similarly, let* $\mathsf{B}'_{\mathsf{fin}} := \Pi_{\mathsf{k}}\mathsf{B}_{\mathsf{fin}}$ *be Bob's last computation in the protocol after he sends the final message to Alice. We note that* $\mathsf{A}_{\mathsf{fin}}$ *and* $\mathsf{B}_{\mathsf{fin}}$ *can make quantum queries to the oracle, and the output of* $\mathsf{A}_{\mathsf{fin}}$ *(resp.* $\mathsf{B}'_{\mathsf{fin}}$*) is the output key of Alice (resp. Bob) at the end of the protocol execution. Let* $(T, \phi_{\mathsf{A}}, \phi_{\mathsf{B}}) \leftarrow \langle \mathsf{A} \models \mathsf{B} \rangle$ *be the output of an execution of the protocol right before Alice receives the last quantum message from Bob, where* $T := (m_1, \cdots, m_\ell, \psi)$ *is the transcript of the execution,* ϕ_{A} *and* ϕ_{B} *are the internal state of* A *and* B*, respectively.* (A, B) *is secure if for any polynomially-bounded query adversary* $\mathcal{E}$*:*

$$\Pr\left[\begin{array}{c|c} \mathsf{k} = \mathsf{k}_{\mathsf{A}} = \mathsf{k}_{\mathsf{B}} & (T, \phi_{\mathsf{A}}, \phi_{\mathsf{B}}) \leftarrow \langle \mathsf{A} \models \mathsf{B} \rangle \\ \mathsf{k}_{\mathsf{A}} \neq \bot & (\mathsf{k}, \psi') \leftarrow \mathcal{E}(1^\lambda, T) \\ \mathsf{k}_{\mathsf{B}} \neq \bot & \mathsf{k}_{\mathsf{A}} \leftarrow \mathsf{A}'_{\mathsf{fin}}(\phi_{\mathsf{A}}, \psi') \\ & \mathsf{k}_{\mathsf{B}} \leftarrow \mathsf{B}'_{\mathsf{fin}}(\phi_{\mathsf{B}}) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

We say that a CC1QM-KA protocol (A, B) *is* (ε, s)*-broken if there exists an attacker Eve that finds the key of* (A, B) *with probability at least* ε*,* (A, B) *succeeds with probability at least* $\mathsf{poly}(\varepsilon)$*, and Eve makes an expected number of queries at most* s*.*

4.2.2 The Attack on Key Agreements Protocols

The goal of the section is to prove the following theorem that states that are no CC1QM-KA protocol in the QROM.

Theorem 4.3. *Let* (A, B) *be a CC1QM-KA protocol, where Alice and Bob make at most* d *queries to a random oracle* $h : \mathcal{X} \to \mathcal{Y}$*, and Alice does not query the oracle in the last part of the protocol (after receiving the quantum message from Bob). Assuming Conjecture 2.1 is true, then there exists an attacker Eve that makes at most* $\mathsf{poly}(d, |\mathcal{Y}|)$ *many* classical *queries to* h *and breaks the security (according to definition 4.2) with probability at least* 0.8*.*

The proof consists of two parts, the first one shows that Eve manages to find the same key as the one computed as Bob, and this is proven in Section 4.2.2. The second part consists of showing that Alice agrees on the same key as Eve and Bob and this corresponds to Section 4.2.2. First, in the next section, we prove that the attack does not depend on the group of the domain of the function.

Group Equivalence of the Attack

We first show that if there is an attack for an Abelian group $\mathcal{Y}$, then there is an attack for any other Abelian group $\mathcal{Y}'$, up to some error terms. This allows us to relax the conjecture to be true for *any* Abelian group, as in [ACC+22]. The proof follows closely [ACC+22]'s proof as they are almost identical, and we include it here for completeness.

Lemma 4.4. *Suppose there exists a finite Abelian group $\mathcal{Y}$, a constant $\tau > 0$ and a function $s(\cdot)$ such that for all $d \in \mathbb{N}$ and any CC1QM-KA protocol $\left(\mathsf{A}_1^h, \mathsf{B}_1^h\right)$ where Alice and Bob asks d queries to a random oracle h whose range is $\mathcal{Y}$, and Alice does not query the oracle after receiving the last message, it holds that $\left(\mathsf{A}_1^h, \mathsf{B}_1^h\right)$ is $(\tau, s(d))$-broken. Then, for any finite Abelian group $\mathcal{Y}'$, any $d' \in \mathbb{N}$, $\delta > 0$ and any CC1QM-KA protocol $\left(\mathsf{A}_2^{h'}, \mathsf{B}_2^{h'}\right)$ where Alice and Bob asks d' queries to another random oracle h' whose range is $\mathcal{Y}'$, $\left(\mathsf{A}_2^{h'}, \mathsf{B}_2^{h'}\right)$, and Alice does not query the oracle after receiving the last message, can be $(\tau - \delta, 4s(md'))$-broken, where*

$$m = \left\lceil \log_{|\mathcal{Y}|}(d'^3 \left|\mathcal{Y}'\right| / 4\delta^2) \right\rceil.$$

Proof. The proof follows from the proof of Lemma 4.8 from [ACC+22]. The only difference is that we must also show that with probability at least $\tau - \delta$, Alice and Bob agree on the same key as Eve. However, their proof relies on the fact that we can simulate a random oracle with another random oracle, even when their ranges are different, up to some errors. Thus, their proof follows through in our setting as well, and with the same parameters. □

Lemma 4.5 (Attacking CC1QM-KA protocols)**.** *Assume Conjecture 2.1 is true for some Abelian group $\mathcal{Y}$ and parameters d and $\delta = \nu/\varepsilon$. Let (A, B) be a CC1QM-KA protocol where Alice and Bob make at most d queries to a random oracle $h : \mathcal{X} \to \mathcal{Y}$, and Alice does not query the oracle after receiving the last message. Then, there exists an active attacker Eve who finds the secret key k with probability $1 - \nu$ according to definition 4.2. Moreover, Eve is expected to make at most d/ε queries to h.*

The proof of Lemma 4.5 is given in subsequent Section 4.2.2.

We can now prove Theorem 4.3:

Proof of Theorem 4.3. The proof follows immediately from Lemma 4.5, Lemma 4.4 and the proof of [ACC+22, Theorem 4.5]. □

Part 1: Finding Bob's Key

In this subsection, we show that the attack algorithm described in Construction 2.1 can efficiently find Bob's key with high probability, assuming that Conjecture 2.1 is true. We first state and show a useful lemma that allows us to assume that when Bob sends the last message, he has already computed the key k on his side.

Lemma 4.6. *Let* (A,B) *be a CC1QM-KA protocol. Let* ϕ_{B} *be the internal state of* B *after he computed the message* ψ*. Then, we can assume w.l.o.g. that Bob has computed the key* k_{B} *from* ϕ_{B} *before he sends the last message* ψ *to Alice.*

Proof. Since the last message of the protocol is sent to Alice by Bob, by the no-signaling principle, Alice's computation after receiving ψ must commute with Bob's computation after sending ψ. Thus, Bob can compute the key on his side before sending the last message ψ. □

Lemma 4.7 (Simulation)**.** *Let* (A,B) *be a CC1QM-KA protocol where Alice and Bob make at most* d *queries to an oracle* $h : \mathcal{X} \to \mathcal{Y}$*, and Alice does not query the oracle after receiving the last message. Assuming conjecture 2.1 is true, for* $0 < \nu < 1$*, there exists an active attacker Eve that finds Bob's key* k_{B} *with probability at least* $1-\nu$ *and Eve is expected to make at most* $\mathsf{poly}\big(d, \frac{1}{\nu}\big)$ *queries to* h*.*

Proof of lemma 4.7. Let Bob's last message be $\psi_M = \sum_i q_i \left|\psi_i\right\rangle\!\left\langle\psi_i\right|_M$, and let $\mathsf{A}'_{\mathsf{fin}} \coloneqq \Pi_{\mathsf{k}}\mathsf{A}_{\mathsf{fin}}$ be Alice's computation in the last step of the protocol. Let k_{B} be the key computed by Bob at the end of the protocol. By lemma 4.6, we can assume that Bob already computes his key k_{B} before sending the last message to Alice.

Our attacking algorithm Eve_1 is described below.

Construction 4.1. Eve_1 *runs the quantum-heavy queries learner* Eve *in construction 2.1 with parameter* $\varepsilon \coloneqq \frac{1}{\mathsf{poly}\left(d,\frac{1}{\nu}\right)}$ *conditioned on the classical transcript* t *until before Bob sends his last message, except that it aborts if* Eve *asks more than* $\frac{d}{\varepsilon}$ *queries. In the case* Eve_1 *does not abort, let* $\left|\Psi_t^{\mathsf{Eve}}\right\rangle_{W'_{\mathsf{A}}W'_{\mathsf{B}}H'}$ *be the state that* Eve *outputs, conditioned on the classical transcript* t*.* Eve_1 *then outputs the measurement outcome of*

$$\mathsf{A}'_{\mathsf{fin}}\left(\left|\Psi_t^{\mathsf{Eve}}\right\rangle_{W'_{\mathsf{A}}W'_{\mathsf{B}}H'} \otimes \psi_M\right),$$

where $\mathsf{A}'_{\mathsf{fin}}$ *makes no oracle query to* h *and acts on two registers* W'_{A} *and* M *only.*

By lemma 2.8, the number of queries asked by Eve satisfies $\mathbb{E}\left[|L|\right] \leq \frac{d}{\varepsilon}$. By Markov's inequality, we have

$$\Pr\left[|L| \geq \frac{d}{\nu\cdot\varepsilon}\right] \leq \nu.$$

Thus, we can conclude that with probability at least $1-\nu$, all of the following events hold:

- Eve_1 is efficient: Eve_1 does not abort and asks at most $\frac{d}{\nu\cdot\varepsilon} = \mathsf{poly}(d, 1/\nu)$ queries.
- Up until before Bob sends his last message, no quantum ε-heavy query is left: for all $x \notin \mathcal{Q}_L, w(x) < \varepsilon$, where $w(\cdot)$ is defined in definition 2.23.

Suppose that all the above events occur for the rest of the proof ($\star$). For simplicity, denote $\left|\Psi_t^{\mathsf{Eve}}\right\rangle_{W'_{\mathsf{A}}W'_{\mathsf{B}}H'}$ as $\left|\Psi_t^{\mathsf{Eve}}\right\rangle_{W'_{\mathsf{A}}E}$.

We will consider the purified version of the protocol. Let $|\phi_t\rangle_{WH}$ be the joint state of the real protocol before Bob sends his last message to Alice, conditioned on the classical transcript t. After Eve_1 learns the heavy queries, the resulting state becomes $|\phi_{t,L}\rangle$ conditioned on t and Eve's list of query-answer L. Since the oracle registers corresponding to Q_L are now measured, we can consider the "truncated" version of $|\phi_{t,L}\rangle_{WH}$ by discarding those registers. Let $\widetilde{H} \coloneqq \{H_x\}_{x\in\mathcal{X}\setminus Q_L}$ be the set of remaining registers. By $|\phi_{t,L}\rangle_{W\widetilde{H}}$ we denote the truncated $|\phi_{t,L}\rangle_{WH}$.

Let $\left|\widehat{\Psi}_{t,L}\right\rangle_{W'_{\mathsf{A}}EW\widetilde{H}} \coloneqq \left|\Psi_t^{\mathsf{Eve}}\right\rangle_{W'_{\mathsf{A}}E} |\phi_{t,L}\rangle_{W\widetilde{H}}$ be the joint state of Eve_1 and the real protocol right before Bob sends his last message to Alice. By lemma 2.6, it holds that $\left|\hat{h}^H_{max}\left(\left|\widehat{\Psi}_{t,L}\right\rangle\right)\right| \leq \mathsf{poly}(d, 1/\nu)$, and by lemma 2.7, it holds that $\left|\hat{h}^{\widetilde{H}}_{max}\left(\left|\widehat{\Psi}_{t,L}\right\rangle\right)\right| \leq \mathsf{poly}(d, 1/\nu)$.

By the assumption ($\star$) above, we have that $\left|\widehat{\Psi}_{t,L}\right\rangle$ is a $(\mathcal{Y}, \varepsilon, \mathsf{poly}(d, \nu), |\mathcal{X}|)$-state (with the register H in definition 2.24 being $\widehat{H}$). Next, let $\psi_M = \sum_i q_i |\psi_i\rangle\langle\psi_i|_M$, we need to show that

$$\forall i, \left\|\Pi_{\mathsf{k}}\mathsf{A}_{\mathsf{fin}}\left|\widehat{\Psi}_{t,L}\right\rangle_{W'_{\mathsf{A}}EW\widetilde{H}} |\psi_i\rangle_M\right\|^2 \geq 1 - \frac{1}{\nu},$$

where $\mathsf{A}_{\mathsf{fin}}$ cannot make queries to h and only acts on W'_{A} and M.

Fix i and write $\left|\widehat{\Psi}^{(\mathrm{i})}_{t,L}\right\rangle_{W'_{\mathsf{A}}EW\widetilde{H}M} \coloneqq \mathsf{A}_{\mathsf{fin}}\left(\left|\widehat{\Psi}_{t,L}\right\rangle_{W'_{\mathsf{A}}EW\widetilde{H}} \otimes |\psi_i\rangle_M\right)$.

Claim 4.8. *If $\left|\widehat{\Psi}_{t,L}\right\rangle$ is a $(\mathcal{Y}, \varepsilon, \mathsf{poly}(d, \nu), |\mathcal{X}|)$-state, it follows that $\left|\widehat{\Psi}^{(\mathrm{i})}_{t,L}\right\rangle$ is a $(\mathcal{Y}, \varepsilon, \mathsf{poly}(d, \nu), |\mathcal{X}|)$-state as well.*

Proof. Assume that $\left|\widehat{\Psi}_{t,L}\right\rangle_{RH}$ is a $(\mathcal{Y}, \varepsilon, \mathsf{poly}(d, \nu), |\mathcal{X}|)$-state. Then, $\left|\widehat{\Psi}_{t,L}\right\rangle_{RH} \otimes |\psi_i\rangle_M$ is also a $(\mathcal{Y}, \varepsilon, \mathsf{poly}(d, \nu), |\mathcal{X}|)$-state, because this property only depends on the H register, who is unchanged there. Then, since $\mathsf{A}_{\mathsf{fin}}$ makes no query to the random oracle, the oracle register H is not modified and thus $\left|\widehat{\Psi}^{(\mathrm{i})}_{t,L}\right\rangle$ is a $(\mathcal{Y}, \varepsilon, \mathsf{poly}(d, \nu), |\mathcal{X}|)$-state. □

We are going to show that there exists a key $\mathsf{k}' = b \in \{0, 1\}$ such that the probability of the key b in the key distribution of $\left|\widehat{\Psi}^{(\mathrm{i})}_{t,L}\right\rangle$ is larger than $1 - \nu$. By contradiction, assume that

for both $b = 0$ and $b = 1$, we have that the probability of this key is smaller than $1 - \nu$. By considering the complementary events, we have that:

$$\left\| \Pi_0 \left| \widehat{\Psi}^{(\mathrm{i})}_{t,L} \right\rangle \right\|^2 \geq \nu, \text{ and}$$
$$\left\| \Pi_1 \left| \widehat{\Psi}^{(\mathrm{i})}_{t,L} \right\rangle \right\|^2 \geq \nu.$$

Let $\left| \widehat{\Psi}^{(\mathrm{i})}_{t,L,\mathsf{k}'=b} \right\rangle$ be the residual state conditioned on the key equal to b. Then, it follows that $\left| \widehat{\Psi}^{(\mathrm{i})}_{t,L,\mathsf{k}'=b} \right\rangle$ is a $(\mathcal{Y}, \varepsilon/\nu, \mathsf{poly}(d, 1/\nu), |\mathcal{X}|)$-state for both $b = 0$ and $b = 1$ because

1. $\left| \widehat{\Psi}^{(\mathrm{i})}_{t,L,\mathsf{k}'=b} \right\rangle$ is $\mathsf{poly}\big(d, \frac{1}{\nu}\big)$-sparse since $\left| \widehat{\Psi}^{(\mathrm{i})}_{t,L} \right\rangle$ is $\mathsf{poly}\big(d, \frac{1}{\nu}\big)$-sparse and $\mathrm{supp}^{\widetilde{H}}(\left| \widehat{\Psi}^{(\mathrm{i})}_{t,L,\mathsf{k}'=b} \right\rangle) \subseteq \mathrm{supp}^{\widetilde{H}}(\left| \widehat{\Psi}^{(\mathrm{i})}_{t,L} \right\rangle)$.

2. $\left| \widehat{\Psi}^{(\mathrm{i})}_{t,L,\mathsf{k}'=b} \right\rangle$ is ε/ν-light because:

$$\begin{aligned}
\Pr\left[\text{Not measuring } \hat{0} \text{ in } \left| \widehat{\Psi}^{(\mathrm{i})}_{t,L,\mathsf{k}'=b} \right\rangle\right] &= \Pr\left[\text{Not measuring } \hat{0} \text{ in } \left| \widehat{\Psi}^{(\mathrm{i})}_{t,L,\mathsf{k}'=b} \right\rangle \,\middle|\, \mathsf{k}' = b\right] \\
&= \frac{\Pr\left[\text{Not measuring } \hat{0} \text{ in } \left| \widehat{\Psi}^{(\mathrm{i})}_{t,L} \right\rangle \text{ and } \mathsf{k}' = b\right]}{\Pr\left[\mathsf{k}' = b\right]} \\
&\leq \frac{\Pr\left[\text{Not measuring } \hat{0} \text{ in } \left| \widehat{\Psi}^{(\mathrm{i})}_{t,L} \right\rangle\right]}{\Pr\left[\mathsf{k}' = b\right]} \\
&\leq \varepsilon/\nu,
\end{aligned}$$

   where the last inequality comes from the fact that $\left| \widehat{\Psi}^{(\mathrm{i})}_{t,L} \right\rangle$ is ε-light and $\Pr[\mathsf{k}' = b] = \left\| \Pi_b \left| \widehat{\Psi}^{(\mathrm{i})}_{t,L} \right\rangle \right\|^2 \geq \nu$.

Then Conjecture 2.1 implies that the states $\left| \widehat{\Psi}^{(\mathrm{i})}_{t,L,\mathsf{k}'=0} \right\rangle$ and $\left| \widehat{\Psi}^{(\mathrm{i})}_{t,L,\mathsf{k}'=1} \right\rangle$ are compatible, which means that there exists two different states $w^0, w^1 \in W'_{\mathsf{A}} E W$ and an oracle $\hat{h}$ such that $\hat{h}$ is consistent with w^0 and w^1. And for this specific oracle, w^0 outputs the key 0 and w^1 outputs the key 1, both with non-zero probability. However, Bob's key has already been computed by Lemma 4.6, and is fixed to some $\mathsf{k}_{\mathsf{B}} \in \{0, 1\}$. Thus, there is an oracle such that Bob outputs k_{B}. Plus, for this specific oracle, Alice outputs key 0 with non-zero probability, and outputs key 1 with non-zero probability as well. Hence there is an execution of the protocol such that Bob outputs k_{B} and Alice outputs $\mathsf{k}_{\mathsf{A}} = 1 - \mathsf{k}_{\mathsf{B}}$, which breaks the perfect completeness of the protocol.

We now show that the key computed by Eve is the same as hypothetical Alice's key, defined by $\mathsf{k}_{\mathsf{A}'} = \mathsf{A}'_{\mathsf{fin}}(|\phi_{\mathsf{A}}\rangle_{AH} \otimes |\psi_i\rangle_M)$, that is the key that Alice would have computed if the protocol had continued normally. Since the protocol is perfect, we have that $\mathsf{k}_{\mathsf{A}'} = \mathsf{k}_{\mathsf{B}}$.

Recall that Eve's key is computed from the state $\left|\widehat{\Psi}^{(\mathrm{i})}_{t,L}\right\rangle = \mathsf{A}_{\mathsf{fin}}\left(\left|\widehat{\Psi}_{t,L}\right\rangle_{W'_{\mathsf{A}}EW\widetilde{H}} \otimes |\psi_i\rangle_M\right)$, and the state $\left|\widehat{\Psi}_{t,L}\right\rangle_{W'_{\mathsf{A}}EW\widetilde{H}}$ is a superposition of all of Alice's internal states that are consistent with Eve's view so far. The state $|\psi_i\rangle_M$ corresponds to the real message that Bob sent to Alice. First note that Alice will never output $\mathsf{k}_{\mathsf{A}} = \bot$, because the state of Eve consists of a superposition of Alice's states that are consistent with the transcript. Indeed, if the message $|\psi_i\rangle_M$ from Bob is inconsistent with the oracle, Alice is not able to detect it as she does not query the oracle in $\mathsf{A}_{\mathsf{fin}}$. Note that the real oracle used in the protocol is one of the oracles in the superposition of oracles that are consistent with Eve's view. Also, for a fixed oracle, the key is computed deterministically by the perfectness of the protocol, and thus Eve's key is equal to Bob's hypothetical key with probability $\left\|\Pi_{\mathsf{k}}\left|\widehat{\Psi}^{(\mathrm{i})}_{t,L}\right\rangle\right\|^2$ over the random oracles. This shows that Eve succeeds with probability at least $1-\nu$. □

Part 2: Making Alice Agrees on the Same Key as Bob

Using Lemma 4.7, we can now prove Lemma 4.5.

Proof of Lemma 4.5. Let (A,B) be a CC1QM-KA protocol where Alice and Bob make at most d queries to an oracle $h : \mathcal{X} \to \mathcal{Y}$, and Alice does not query the oracle after receiving the last message. Consider the following construction for Eve:

Construction 4.2. *Input:* ε, ν

1. Eve *applies the quantum ε-heavy query learner of Construction 2.1 to compute a state $\left|\phi^E_A\right\rangle_{W'_A}\left|\phi^E_B\right\rangle_{W'_B}|h\rangle_H$ which corresponds to a simulation of the internal state of Alice and Bob after the classical communication part of the protocol.*
2. *Let* $\mathsf{A}_{\mathsf{fin}}$ *be the operations that Alice applies at the end of the protocol after receiving the message ψ from Bob. Then,* Eve *outputs the resulting key k_E of*

$$\Pi_{\mathsf{k}}\mathsf{A}_{\mathsf{fin}}\left|\phi^E_A\right\rangle\!\!\left\langle\phi^E_A\right|\left|h\right\rangle\!\!\left\langle h\right|\psi(\mathsf{A}_{\mathsf{fin}})^\dagger,$$

   where ψ is the quantum message Bob sends to Alice.
3. *Writing* $\tau_{EM} = \frac{\widetilde{\tau}}{\|\widetilde{\tau}\|}$, *where* $\widetilde{\tau} = (\mathsf{A}_{\mathsf{fin}})^\dagger\Pi_{\mathsf{k}}\mathsf{A}_{\mathsf{fin}}\left|\phi^E_A\right\rangle\!\!\left\langle\phi^E_A\right|\left|h\right\rangle\!\!\left\langle h\right|\psi$, Eve *sends the resulting state* $\mathrm{Tr}_E(\tau_{EM})$ *to Alice, where she traces out everything but the register that contains the message.*

In the last part of the construction, Eve applies the operator $(\mathsf{A}_{\mathsf{fin}})^\dagger$ to uncompute Alice's operation before sending her state to Alice. Note that in step 2, we use the fact that Alice and Bob's states are unentangled, as shown by Lemma 2.25

Now, we prove that Construction 4.2 succeeds with probability at least $1-\nu$. Using Lemma 4.7, we have that Eve finds the right key k in Step 2 with probability at least $1-\nu$.

Writing $\psi = \sum_i q_i |\psi_i\rangle$, this means that

$$\forall i, \left\| \Pi_{\mathsf{k}} \mathsf{A}_{\mathsf{fin}} \left|\phi_A^E\right\rangle |h\rangle |\psi_i\rangle \right\|^2 \geq 1 - \lambda. \tag{4.1}$$

We write $\psi^E = \mathrm{Tr}_E(\tau_{EM})$ the message that Eve sends to Alice. The first thing that we want to show is that the message ψ^E from Eve is "close" to the real message ψ from Bob. More precisely, we will show that:

$$\forall i, \langle\psi_i| \psi^E |\psi_i\rangle \geq 1 - \lambda. \tag{4.2}$$

For every i, we have that:

$$\begin{aligned}
\langle\psi_i| \psi^E |\psi_i\rangle &= \langle\psi_i| \mathrm{Tr}_E(\tau_{EM}) |\psi_i\rangle \\
&= \langle\psi_i| \mathrm{Tr}_E\left(\frac{\widetilde{\tau}}{\|\widetilde{\tau}\|}\right) |\psi_i\rangle \\
&\geq \langle\psi_i| \mathrm{Tr}_E(\widetilde{\tau}) |\psi_i\rangle \\
&= \mathrm{Tr}\left(|\psi_i\rangle\langle\psi_i| \mathrm{Tr}_E(\widetilde{\tau})\right) \\
&= \mathrm{Tr}\left(I_E \otimes \langle\psi_i|_M \widetilde{\tau} \cdot I_E \otimes |\psi_i\rangle_M\right) \\
&\geq \mathrm{Tr}\left(\left\langle\phi_A^E\right| \langle h| \otimes \langle\psi_i|_M \widetilde{\tau} \cdot \left|\phi_A^E\right\rangle |h\rangle \otimes |\psi_i\rangle_M\right),
\end{aligned}$$

where we used elementary properties of the trace operator. Next, we have that

$$\mathrm{Tr}\left(\left\langle\phi_A^E\right| \langle h| \otimes \langle\psi_i|_M \widetilde{\tau} \cdot \left|\phi_A^E\right\rangle \otimes |\psi_i\rangle_M\right) = \left\langle\phi_A^E\right| \otimes \langle\psi_i|_M \widetilde{\tau} \cdot \left|\phi_A^E\right\rangle |h\rangle \otimes |\psi_i\rangle_M,$$

since the right term is a pure state. Replacing $\widetilde{\tau}$ with its value, we have:

$$\begin{aligned}
\langle\psi_i| \psi^E |\psi_i\rangle &\geq \left\langle\phi_A^E\right| \langle h| \langle\psi_i| \left((\mathsf{A}_{\mathsf{fin}})^\dagger \Pi_{\mathsf{k}} \mathsf{A}_{\mathsf{fin}} \left|\phi_A^E\right\rangle\!\!\left\langle\phi_A^E\right| |h\rangle\langle h| \psi\right) \left|\phi_A^E\right\rangle |h\rangle |\psi_i\rangle \\
&= \sum_j q_j \left\langle\phi_A^E\right| \langle h| \langle\psi_i| \left((\mathsf{A}_{\mathsf{fin}})^\dagger \Pi_{\mathsf{k}} \mathsf{A}_{\mathsf{fin}} \left|\phi_A^E\right\rangle\!\!\left\langle\phi_A^E\right| |H\rangle\langle H| |\psi_j\rangle\langle\psi_j|\right) \left|\phi_A^E\right\rangle |h\rangle |\psi_i\rangle \\
&= \left\langle\phi_A^E\right| \langle h| \langle\psi_i| \left((\mathsf{A}_{\mathsf{fin}})^\dagger \Pi_{\mathsf{k}} \mathsf{A}_{\mathsf{fin}} \left|\phi_A^E\right\rangle |h\rangle |\psi_i\rangle\right) \\
&= \left\|(\mathsf{A}_{\mathsf{fin}})^\dagger \Pi_{\mathsf{k}} \mathsf{A}_{\mathsf{fin}} \left|\phi_A^E\right\rangle |h\rangle |\psi_i\rangle\right\|^2 \\
&= \left\|\Pi_{\mathsf{k}} \mathsf{A}_{\mathsf{fin}} \left|\phi_A^E\right\rangle |h\rangle |\psi_i\rangle\right\|^2 \\
&\geq 1 - \lambda,
\end{aligned}$$

where the last inequality comes from Equation (4.1).

Now fix i. We write $|\Phi_A\rangle = \mathsf{A}_{\mathsf{fin}} |\phi_{\mathsf{A}}\rangle \otimes |h\rangle \otimes |\psi_i\rangle$ where $|\phi_{\mathsf{A}}\rangle$ corresponds to Alice's *real* register before receiving the message ψ. Since Π_{k} is a projector and $\Pi_{\mathsf{k}} |\Phi_A\rangle = |\Phi_A\rangle$ from perfect correctness, we can write it:

$$\Pi_{\mathsf{k}} = |\Phi_A\rangle\langle\Phi_A| + \sum_i |\sigma_i\rangle\langle\sigma_i|,$$

where the σ_i are such that $\langle\sigma_i|\Phi_A|\sigma_i|\Phi_A\rangle = 0$.

We write:

$$\psi^E = \alpha\,|\psi_i\rangle\langle\psi_i| + \beta\rho,$$

where $\rho = \sum_j p_j\,|\Psi_j\rangle\langle\Psi_j|$ is a mixed state such that $\langle\psi_i|\,\rho\,|\psi_i\rangle = 0$.

For every $|\psi_i\rangle$, we have that:

$$\begin{aligned}
&\mathrm{Tr}\left(\Pi_{\mathsf{k}}\mathsf{A}_{\mathsf{fin}}(|\phi_{\mathsf{A}}\rangle\langle\phi_{\mathsf{A}}|\otimes|h\rangle\langle h|\otimes\psi^E)\,(\mathsf{A}_{\mathsf{fin}})^\dagger\right)\\
&= \mathrm{Tr}\left(\left(|\Phi_A\rangle\langle\Phi_A| + \sum_i |\sigma_i\rangle\,\langle\sigma_i|\right)\left(\mathsf{A}_{\mathsf{fin}}\,|\phi_{\mathsf{A}}\rangle\langle\phi_{\mathsf{A}}|\otimes|h\rangle\langle h|\otimes(\alpha\,|\psi_i\rangle\langle\psi_i| + \beta\rho)\,(\mathsf{A}_{\mathsf{fin}})^\dagger\right)\right)\\
&= \mathrm{Tr}\left(\left(|\Phi_A\rangle\langle\Phi_A| + \sum_i |\sigma_i\rangle\,\langle\sigma_i|\right)\left(\alpha\,|\Phi_A\rangle\langle\Phi_A| + \beta\mathsf{A}_{\mathsf{fin}}\,|\phi_{\mathsf{A}}\rangle\langle\phi_{\mathsf{A}}|\otimes|h\rangle\langle h|\otimes\rho\,(\mathsf{A}_{\mathsf{fin}})^\dagger\right)\right)\\
&= \alpha\,\langle\Phi_A|\Phi_A|\Phi_A|\Phi_A\rangle^2 + \beta\,\langle\Phi_A|\,\mathsf{A}_{\mathsf{fin}}\,|\phi_A\rangle\langle\phi_A|\otimes|h\rangle\langle h|\otimes\rho(\mathsf{A}_{\mathsf{fin}})^\dagger\,|\Phi_A\rangle\\
&\quad + \alpha\sum_i |\,\langle\Phi_A|\sigma_i|\Phi_A|\sigma_i\rangle\,|^2 + \beta\sum_i \langle\sigma_i|\,\mathsf{A}_{\mathsf{fin}}\,|\phi_{\mathsf{A}}\rangle\langle\phi_{\mathsf{A}}|\otimes|h\rangle\langle h|\otimes\rho(\mathsf{A}_{\mathsf{fin}})^\dagger\,|\sigma_i\rangle\\
&= \alpha + \beta\sum_i \langle\sigma_i|\,\mathsf{A}_{\mathsf{fin}}\,|\phi_{\mathsf{A}}\rangle\langle\phi_{\mathsf{A}}|\otimes|h\rangle\langle h|\otimes\rho(\mathsf{A}_{\mathsf{fin}})^\dagger\,|\sigma_i\rangle\\
&\geq \alpha\\
&\geq 1-\lambda,
\end{aligned}$$

where the fourth equality comes from the fact that

$$\begin{aligned}
\langle\Phi_A|\,\mathsf{A}_{\mathsf{fin}}\,|\phi_{\mathsf{A}}\rangle\langle\phi_{\mathsf{A}}|\otimes\rho(\mathsf{A}_{\mathsf{fin}})^\dagger\,|\Phi_A\rangle &= \langle\phi_{\mathsf{A}}|\,\langle\psi_i|\,(\mathsf{A}_{\mathsf{fin}})^\dagger\mathsf{A}_{\mathsf{fin}}\,|\phi_{\mathsf{A}}\rangle\langle\phi_{\mathsf{A}}|\otimes\rho(\mathsf{A}_{\mathsf{fin}})^\dagger\mathsf{A}_{\mathsf{fin}}\,|\phi_{\mathsf{A}}\rangle\,|\psi_i\rangle\\
&= \langle\psi_i|\,\rho\,|\psi_i\rangle = 0,
\end{aligned}$$

and that $\langle\sigma_i|\Phi_A|\sigma_i|\Phi_A\rangle = 0$. The first inequality comes from the fact that the terms in the sum are positive, because they correspond to the probability of measuring the state $|\phi_{\mathsf{A}}\rangle\langle\phi_{\mathsf{A}}|\otimes|h\rangle\langle h|\otimes\rho$ using the projection $\mathsf{A}_{\mathsf{fin}}\,\langle\sigma_i|\mathsf{A}_{\mathsf{fin}}\rangle^\dagger$, and the last inequality comes from Equation (4.2).

This means that Alice measures the key k with probability at least $1-\lambda$ when receiving the message ψ^E from Eve and for pure message $|\psi_i\rangle$, and if this is the case the meet-in-the-middle attack is a success. Since this is true for all of the $|\psi_i\rangle$, it also follows for ψ by convexity. This concludes the proof. □

4.2.3 Impossibility of Quantum Public Key Encryption with Classical Keys

In this section, we show that the (conditional) impossibility of CC1QM-KA protocols proven above also implies a (conditional) impossibility for quantum public key encryption (qPKE)

with classical public keys, but secret keys and ciphertexts can be quantum states. More precisely, we are interested in perfectly correct (c, q, q)-PKE, as defined in Definition 2.7.

Next, in Definition 4.3, we define the indistinguishability security of one-bit qPKE. This is a slight modification of the definition given in Definition 2.8, which we restate here for clarity. When considering one-bit encryption this notion coincides with the one-way security notion, which is considered the weakest security notion of encryption. Thus, using this notion makes our negative result stronger.

Definition 4.3. *A one-bit qPKE scheme with classical public keys is IND-CPA secure if for every* QPT *adversary* $\mathcal{A}$*, for any* $\lambda \in \mathbb{N}$*, there exists a negligible function* $\mathsf{negl}(\lambda)$ *such that*

$$\Pr\left[\mathtt{IND}-\mathtt{CPA}(\lambda, \mathcal{A}) = 1\right] \leq \frac{1}{2} + \mathsf{negl}(\lambda),$$

where $\mathtt{IND}-\mathtt{CPA}(\lambda, \mathcal{A})$ *is the following experiment:*

1. *The challenger chooses a random key pair* $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$*, and sends* pk *to the adversary* $\mathcal{A}$*.*
2. $\mathcal{A}$*, upon receiving the public key* pk*, sends two bits* $m_0, m_1 \in \{0,1\}$ *to the challenger.*
3. *The challenger samples a random bit* $b \leftarrow\!\!\$\ \{0,1\}$*, and sends* $\mathsf{c} \leftarrow \mathsf{Enc}(\mathsf{pk}, m_b)$ *to* $\mathcal{A}$*.*
4. $\mathcal{A}$ *responds with a guess* b' *for* b*.*
5. *The challenger outputs* 1 *if* $b' = b$*, and* 0 *otherwise.*

Since the existence of an IND-CPA secure qPKE scheme with classical public keys in the QROM implies the existence of a CC1QM-KA protocol in the QROM, we also obtain the following result.

Corollary 4.9. *Assuming Conjecture 2.1 is true, there is no IND-CPA secure qPKE scheme with classical public keys in the QROM, where the decryption algorithm does not query the random oracle.*

Proof. By contradiction, let $\Pi = (\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ be a qPKE scheme with classical public keys and assume it is IND-CPA secure. We construct a two-message one-bit CC1QM-KA protocol $\tilde{\Pi}$, where the first message from Alice to Bob is classical and the second message from Bob to Alice is quantum, as follows.

1. Alice generates $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$, and sends pk to Bob.
2. Bob generates uniformly at random a secret key $\mathsf{k} \in \{0,1\}$ and computes $\mathsf{c} \leftarrow \mathsf{Enc}(\mathsf{pk}, \mathsf{k})$, and sends c to Alice.
3. Alice recovers the common key by computing $\mathsf{k} \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{c})$.

It is easy to see that $\tilde{\Pi}$ is a secure CC1QC-KM protocol in the QROM if Π is IND-CPA secure. Furthermore, if Π is perfectly correct, $\tilde{\Pi}$ is also perfectly correct. Finally, if $\mathsf{Dec}(\cdot,\cdot)$ does not query the oracle, then Alice in the last step of $\tilde{\Pi}$ does not query the oracle as well. This contradicts theorem 4.3 and concludes our proof. □

Remark 4.2. *We note that our impossibility of CC1QM-KA is the strongest possible impossibility (conditioned on the assumption that conjecture 2.1 is true), in the sense that the adversary can find the shared key and maintain the correctness of the protocol (that is, Alice and Bob can still find the shared key), while the usual security definition only asks the adversary to find the key of one of two parties. This strong impossibility allows us to rule out the possibility of qPKE in the QROM with stronger requirements, for example, qPKE with decryption error detectability as defined in [KMNY23].*

Chapter 5

On Limits on The Provable Consequences of Quantum Pseudorandomness

In this chatper, we study the construction of cryptographic schemes from quantum pseudorandomness. We show separations between different quantum cryptographic primitives, and also study the possibility of constructing different kinds of public-key encryption and signatures from PRUs.

5.1 Separating PRSs from Short PRSs

In this section, we prove that we can separate poly-size PRSs and log-size PRSs. Formally, we will be proving the following theorem.

Theorem 5.1. *There exists a quantum oracle $\mathcal{O}$ such that relative to $\mathcal{O}$, λ-PRSs exist, but $(c\log\lambda)$-PRSs with $c > 12$ do not exist.*

The oracle necessary for the separation is actually the same oracle that Kretschmer used to separate PRSs and OWFs (Theorem 2.23). It turns out that this oracle is also separating PD-OWFs from PRSs, we prove here that if $\mathsf{PromiseBQP} = \mathsf{PromiseQMA}$, then we do not have PD-OWFs. This in addition with Barhoush and Salvail's result that short-PRSs are enough to build PD-OWFs (Theorem 2.22) will give us the theorem.

According to the above considerations, the main theorem follows directly from the following proposition.

Proposition 5.2. *If PD-OWFs exist relative to a quantum oracle $\mathcal{O}$, then* $\mathsf{PromiseBQP}^{\mathcal{O}} \neq \mathsf{PromiseQMA}^{\mathcal{O}}$.

Proof. Let $\lambda \in \mathbb{N}$ and $F : \{0,1\}^{\lambda} \to \{0,1\}^{\ell(\lambda)}$ be a PD-OWF relative to the oracle $\mathcal{O}$. Let us define a promise language $\mathcal{L} = \mathcal{L}_{\text{yes}} \cup \mathcal{L}_{\text{no}} \cup \mathcal{L}_{\perp}$ with $\mathcal{L} \subseteq \{0,1\}^*$ where *yes* instances have a pre-image with respect to F_λ but *no* instances do not. Formally,

$$\mathcal{L}_{\text{yes}} = \Big\{ (1^{\ell(\lambda)}, x', y) \in \{1^{\ell(\lambda)}\} \times \{0,1\}^{\lambda} \times \{0,1\}^{\ell(\lambda)} \Big| \exists x \in \{0,1\}^{\lambda}, x' \prec x \text{ and } \forall c > 0, \exists \lambda_0, \forall \lambda \geq \lambda_0, \Pr[y = F_\lambda(x)] \geq 1 - \frac{1}{\lambda^c} \Big\}, \quad (5.1)$$

$$\mathcal{L}_{\text{no}} = \Big\{ (1^{\ell(\lambda)}, x', y) \in \{1^{\ell(\lambda)}\} \times \{0,1\}^{\lambda} \times \{0,1\}^{\ell(\lambda)} \Big| \forall x \in \{0,1\}^{\lambda}, x' \not\prec x \text{ or } \Pr[y = F_\lambda(x)] \leq 1 - \frac{1}{\lambda} \Big\}. \quad (5.2)$$

Note that in the definition of $\mathcal{L}_{\text{yes}}$, we have that $\Pr[y = F_\lambda(x)]$ is a negligible function, and we will use this property in the rest of the section. We claim that $\mathcal{L} \in \mathsf{PromiseQMA}^{\mathcal{O}}$ but $\mathcal{L} \notin \mathsf{PromiseBQP}^{\mathcal{O}}$, thus there must be a separation between both complexity classes. These claims are proven in Lemma 5.3 and Lemma 5.4 respectively. □

We now prove the two claims from the proposition. We start by showing that the language defined in Proposition 5.2 is in $\mathsf{PromiseQMA}^{\mathcal{O}}$, this is, we will construct an algorithm (verifier) that given an element of the domain and a (quantum) proof can distinguish if the element is a yes or no instance of the language.

Lemma 5.3. *Let F be a PD-OWF and let $\mathcal{L}$ be the language defined in Proposition 5.2, then $\mathcal{L} \in \mathsf{PromiseQMA}^{\mathcal{O}}$.*

Proof. We define a quantum polynomial-time algorithm $\mathcal{A}^{\mathcal{O}}$ that given an element of the domain $(1^{\ell(\lambda)}, x', y) \in \{1^{\ell(\lambda)}\} \times \{0,1\}^{\lambda} \times \{0,1\}^{\ell(\lambda)}$ and a classical proof $x \in \{0,1\}^{\lambda}$, will check if the proof x is indeed a pre-image of the PD-OWF by checking if it coincides with the output y.

Algorithm 5.1. *Input:* $(1^{\ell(\lambda)}, x', y) \in \{1^{\ell(\lambda)}\} \times \{0,1\}^{\lambda} \times \{0,1\}^{\ell(\lambda)}$ *and* $x \in \{0,1\}^{\lambda}$.

1. *If $x' \not\prec x$, return* 0.
2. *For $1 \leq i \leq 2\lambda$, compute $y_i = F_\lambda(x)$. If $y_i \neq y$, return* 0.
3. *Return* 1.

Note that Algorithm 5.1 runs in polynomial-time trivially because computing F_λ is done efficiently relative to $\mathcal{O}$ by definition and we make 2λ calls to it. We now prove that Algorithm 5.1 distinguishes between the yes/no instances.

(i) Let $(1^{\ell(\lambda)}, x', y) \in \mathcal{L}_{\text{yes}}$. Then by definition there exists a proof $x \in \{0,1\}^{\lambda}$ such that

$$x' \prec x \text{ and } \Pr[y = F_\lambda(x)] \geq 1 - \mathsf{negl}(\lambda).$$

Then the proof x will be an element of the input $((1^{\ell(\lambda)}, x', y), x)$ for which the algorithm $\mathcal{A}$ will output 1 with high probability because

$$\begin{aligned}\Pr\left[\mathcal{A}^{\mathcal{O}}((1^{\ell(\lambda)}, x', y), x) = 1\right] &= \Pr\left[\forall 1 \leq i \leq 2\lambda, \text{ the execution of } F_\lambda(x) \text{ outputs y}\right] \\ &\geq (1 - \mathsf{negl}(\lambda))^{2\lambda} \geq 2/3,\end{aligned}$$

which holds whenever λ is big enough. Indeed, recall that $\mathsf{negl}(\lambda) \leq 1/\lambda^c$ for all $c > 1$ and λ big enough, thus in particular $\mathsf{negl}(\lambda) \leq 1/\lambda^2$ for λ big enough, hence

$$(1 - \mathsf{negl}(\lambda))^{2\lambda} \geq \left(1 - \frac{1}{\lambda^2}\right)^{2\lambda} \geq \frac{2}{3},$$

whenever λ is big enough and for the last inequality we need at least $\lambda \geq 6$, where we used that we have an increasing function in λ.

(ii) Let $(1^{\ell(\lambda)}, x', y) \in \mathcal{L}_{\text{no}}$. Then by definition for every potential proof $x \in \{0,1\}^\lambda$ we have that either

$$x' \not\prec x \quad \text{or} \quad \Pr[y = F_\lambda(x)] \leq 1 - \frac{1}{\lambda}.$$

Then for every possible input $((1^{\ell(\lambda)}, x', y), x)$ the algorithm will output 0 with high probability because

$$\begin{aligned}\Pr\left[\mathcal{A}((1^{\ell(\lambda)}, x', y), x) = 1\right] &= \Pr\left[x' \prec x \wedge \forall 1 \leq i \leq 2\lambda, \text{ the execution of } F_\lambda(x) \text{ outputs y}\right] \\ &\leq \Pr\left[\forall 1 \leq i \leq 2\lambda, \text{ the execution of } F_\lambda(x) \text{ outputs y}\right] \\ &\leq \left(1 - \frac{1}{\lambda}\right)^{2\lambda} \leq e^{-2} \leq 1/3.\end{aligned}$$

□

Lemma 5.4. *Let F be a PD-OWF and let $\mathcal{L}$ be the language defined in Proposition 5.2, then $\mathcal{L} \notin \mathsf{PromiseBQP}^{\mathcal{O}}$.*

Proof. We will prove this by contradiction. Let us assume that instead $\mathcal{L} \in \mathsf{PromiseBQP}^{\mathcal{O}}$, this is, there exists a BQP algorithm $\mathcal{A}^{\mathcal{O}}$ such that:

1. If $(1^{\ell(\lambda)}, x', y) \in \mathcal{L}_{\text{yes}}$, then $\Pr\left[\mathcal{A}^{\mathcal{O}}(1^{\ell(\lambda)}, x', y) = 1\right] \geq 2/3$.
2. If $(1^{\ell(\lambda)}, x', y) \in \mathcal{L}_{\text{no}}$, then $\Pr\left[\mathcal{A}^{\mathcal{O}}(1^{\ell(\lambda)}, x', y) = 1\right] \leq 1/3$.

Without loss of generality, we can assume that the algorithm $\mathcal{A}^{\mathcal{O}}$ has completeness $1 - \frac{1}{\ell(\lambda)}$ and soundness $\frac{1}{\ell(\lambda)}$. We will now show how we can construct a QPT algorithm $\mathcal{A}'^{\mathcal{O}}$ that finds a pre-image of every F_λ with high probability when it exists, by querying the original BQP algorithm $\mathcal{A}^{\mathcal{O}}$ at most $\ell(\lambda) + 1$ times.

Algorithm 5.2. *Input:* $(1^{\ell(\lambda)}, y) \in \{1\}^{\ell(\lambda)} \times \{0,1\}^{\ell(\lambda)}$.

1. $b \leftarrow \mathcal{A}^{\mathcal{O}}(1^{\ell(\lambda)}, \varepsilon, y)$.
2. *If* $b = 0$, *return* $\perp$.
3. $x_0 \leftarrow \varepsilon$.
4. *For* $1 \leq i \leq \ell(\lambda)$:
    1. $b \leftarrow \mathcal{A}^{\mathcal{O}}(1^{\ell(\lambda)}, x_0||0, y)$.
    2. *If* $b = 1$, $x_0 = x_0||0$. *Else,* $x_0 = x_0||1$.
5. *Return* x_0.

Indeed if $(1^{\ell(\lambda)}, \varepsilon, y) \in \mathcal{L}_{\text{yes}}$, then the probability that Algorithm 5.2 outputs a correct pre-image is very high

$$\Pr\left[y = \underset{y \in \{0,1\}^{\ell(\lambda)}}{\arg\max} \Pr\left[y = F_\lambda(x)\right] \,\middle|\, x \leftarrow \mathcal{A}'^{\mathcal{O}}\left(1^{\ell(\lambda)}, y\right)\right] \geq \left(1 - \frac{1}{\ell(\lambda)}\right)^{\ell(\lambda)+1}. \tag{5.3}$$

However, this raises a contradiction with the security of the PD-OWF from the assumption Definition 2.2,

$$\begin{aligned}
&\Pr_{x \leftarrow \{0,1\}^{\ell(\lambda)}}\left[F_\lambda(\mathcal{A}'^{\mathcal{O}}(1^{\ell(\lambda)}, F_\lambda(x))) = F_\lambda(x)\right] \\
&\quad= \Pr_{x \leftarrow \{0,1\}^{\ell(\lambda)}}\left[F_\lambda(\mathcal{A}'^{\mathcal{O}}(1^{\ell(\lambda)}, F_\lambda(x))) = F_\lambda(x) \,\middle|\, x \in \mathcal{K}_\lambda\right] \Pr_{x \leftarrow \{0,1\}^{\ell(\lambda)}}\left[x \in \mathcal{K}_\lambda\right] \\
&\qquad+ \Pr_{x \leftarrow \{0,1\}^{\ell(\lambda)}}\left[F_\lambda(\mathcal{A}'^{\mathcal{O}}(1^{\ell(\lambda)}, F_\lambda(x))) = F_\lambda(x) \,\middle|\, x \notin \mathcal{K}_\lambda\right] \Pr_{x \leftarrow \{0,1\}^{\ell(\lambda)}}\left[x \notin \mathcal{K}_\lambda\right] \\
&\quad\geq \Pr_{x \leftarrow \{0,1\}^{\ell(\lambda)}}\left[F_\lambda(\mathcal{A}'^{\mathcal{O}}(1^{\ell(\lambda)}, F_\lambda(x))) = F_\lambda(x) \,\middle|\, x \in \mathcal{K}_\lambda\right] \Pr_{x \leftarrow \{0,1\}^{\ell(\lambda)}}\left[x \in \mathcal{K}_\lambda\right] \\
&\quad\geq \Pr_{x \leftarrow \{0,1\}^{\ell(\lambda)}}\left[F_\lambda(\mathcal{A}'^{\mathcal{O}}(1^{\ell(\lambda)}, F_\lambda(x))) = F_\lambda(x) \,\middle|\, x \in \mathcal{K}_\lambda\right](1 - \mu(\lambda)),
\end{aligned}$$

where the first equality comes from the law of total probability and the second inequality

comes from the property of $\mathcal{K}_\lambda$. We can rewrite the last element as:

$$\begin{aligned}
&\Pr_{x\leftarrow\{0,1\}^{\ell(\lambda)}}\left[y_3 = y_1 \;\middle|\; y_1, y_2 \leftarrow F_\lambda(x), x_1 \leftarrow \mathcal{A}'^{\mathcal{O}}(1^{\ell(\lambda)}, y_2), y_3 \leftarrow F_\lambda(x_1), x \in \mathcal{K}_\lambda\right] \\
&\quad\geq \Pr_{x\leftarrow\{0,1\}^{\ell(\lambda)}}\left[\left(y_3 = y_2 = y_1 = \operatorname*{arg\,max}_{y\in\{0,1\}^{\ell(\lambda)}} \Pr[y = F_\lambda(x)]\right) \wedge (x_1 = x) \;\middle|\; \begin{array}{c} y_1, y_2, y_3 \leftarrow F_\lambda(x) \\ x_1 \leftarrow \mathcal{A}'^{\mathcal{O}}(1^{\ell(\lambda)}, y_2), x \in \mathcal{K}_\lambda \end{array}\right] \\
&\quad= \Pr_{x\leftarrow\{0,1\}^{\ell(\lambda)}}\left[F_\lambda(x) = \operatorname*{arg\,max}_{y\in\{0,1\}^{\ell(\lambda)}} \Pr[y = F_\lambda(x)] \;\middle|\; x \in \mathcal{K}_\lambda\right]^3 \\
&\qquad\cdot \Pr_{x\leftarrow\{0,1\}^{\ell(\lambda)}}\left[\mathcal{A}'^{\mathcal{O}}\left(1^{\ell(\lambda)}, \operatorname*{arg\,max}_{y\in\{0,1\}^{\ell(\lambda)}} \Pr[y = F_\lambda(x)]\right) = x \;\middle|\; x \in \mathcal{K}_\lambda\right] \\
&\quad\geq (1 - \mathsf{negl}(\lambda))^3 \left(1 - \frac{1}{\ell(\lambda)}\right)^{\ell(\lambda)+1},
\end{aligned}$$

where the first inequality comes from the law of total probability, and the last inequality from the definition of a PD-OWF and Equation (5.3). This gives that:

$$\begin{aligned}
\Pr_{x\leftarrow\{0,1\}^{\ell(\lambda)}}\left[F_\lambda(\mathcal{A}'^{\mathcal{O}}(1^{\ell(\lambda)}, F_\lambda(x))) = F_\lambda(x)\right] &\geq (1 - \mathsf{negl}(\lambda))^3 \left(1 - \frac{1}{\ell(\lambda)}\right)^{\ell(\lambda)+1} (1 - \mu(\lambda)) \\
&\geq (1 - O(\mu(\lambda))) \left(1 - \frac{1}{\ell(\lambda)}\right)^{\ell(\lambda)+1}.
\end{aligned}$$

Note that this bound is not negligible since

$$\left(1 - \frac{1}{\ell(\lambda)}\right)^{\ell(\lambda)+1} \geq \frac{1}{10},$$

whenever $\lambda \geq 2$, which contradicts Equation (2.2). □

5.2 Separating Short PRSs from Shorter PRSs

Theorem 5.1 shows that short-PRSs and PD-PRGs require a computational assumption to exist. Combining our result with a prior result of Brakerski and Shmueli [BS20] that show that statistically secure PRSs[1] with shorter than $\log(\lambda)$ length exist unconditionally, we can separate them from short-PRSs and PD-OWFs. More formally, we have the following result.

Corollary 5.5. *Let $\mathcal{L}$ be a* PSPACE*-complete language, and let $\mathcal{O}$ be an oracle that solves $\mathcal{L}$. There exist a constant $0 < c_0 < 1$ such that relative to $\mathcal{O}$:*

- *For any $n < c_0 \log \lambda$, n-PRSs exist.*

[1] We note that their construction relativizes because the security is statistical.

- *For any $c > 12$, $c \log \lambda$-PRSs do not exist.*

In particular, we have that $c \log \lambda$-PRSs for $c < c_0$ are separated from $c \log \lambda$-PRSs for $c > 12$ in a black-box way. Moreover, $c \log \lambda$-PRS for $c < c_0$ are separated from PD-OWFs, while $c \log \lambda$-PRS for $c > 12$ imply PD-OWFs.

5.3 On Constructing Public Key Encryption

The goal of this section is to study the construction of public key encryption schemes (PKEs) from PRUs. Remember that we let $(\mathsf{pk}, \mathsf{sk}, \mathsf{c}) \in \{c, q\}^3$ and discuss $(\mathsf{pk}, \mathsf{sk}, \mathsf{c})$-PKE where a c indicate a classical string, while a q indicate a quantum state.

5.3.1 (c,c,q) and (c,c,c) Encryption

In this section, we show that public key encryption with classical keys cannot be build from PRU. To do so, we simply show that EV-OWPs can be built from PKE with classical keys, and the proof follows by the oracle of Theorem 2.23 and Lemma 2.24. Note that this result was already proven in previous works [KT24a, CGG24], and we add it here for completeness. The result is the following theorem:

Theorem 5.6. *The existence of PKE with classical keys imply the existence of EV-OWP.*

Corollary 5.7. *There is no black-box construction of PKE with classical keys from PRUs.*

Proof. Let $(\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ be a public key encryption scheme with classical keys.

We define the following one-way puzzle:

- $\mathsf{Gen}(\lambda) = \mathsf{KGen}(\lambda) = (\mathsf{pk}, \mathsf{sk}) = (\mathsf{s}, \mathsf{k})$.
- $\mathsf{Ver}(\mathsf{s}, \mathsf{k})$: Sample $m \leftarrow\!\!\$\ \{0,1\}^\lambda$. Return $\top$ if $\mathsf{Dec}(\mathsf{Enc}(m, \mathsf{s}), \mathsf{k}) = m$, otherwise return $\bot$.

It is straightforward that $(\mathsf{Gen}, \mathsf{Ver})$ is an efficient verifiable one-way puzzle. Indeed, the correctness from the correctness the public key encryption scheme. Security also follows from the security of the public key encryption scheme, from an adversary that breaks the above construction, one can construct an adversary that breaks the security of the public key encryption scheme. At last, the verification is clearly efficient, because the public key encryption scheme is efficient. □

We note that we only require the keys to be classical and the ciphertext can be a quantum state.

5.3.2 (c,q,c) Encryption

In this section, we show that public key encryption with classical public key and classical ciphertext cannot be build from PRUs in a black-box way. To show this result, we show that

the existence of (c,q,c)-PKE imply that $\mathsf{BQP} \neq \mathsf{QMA}$, and the proof relativizes. Remark that unlike the previous section, we do not show that the existence of (c,q,c)-PKE imply the existence of EV-OWPs. It was proven that the existence of (c,q,c)-PKE imply the existence of (inefficient) OWPs [KT24a], but that is not enough to show the separation, as OWPs can be built from PRUs and thus exist relative to Kretschmer's oracle. However, we are still able to show that, just like EV-OWPs, (c,q,c)-PKE do not exist relative to Kretschmer's oracle.

Theorem 5.8. *Let $\mathcal{O}$ be an oracle relative to which PKE with classical public keys and classical ciphertexts exist. Then $\mathsf{BQP}^{\mathcal{O}} \neq \mathsf{QMA}^{\mathcal{O}}$.*

Proof. Let $\mathcal{O}$ be an oracle relative to which PKE with classical public key and classical ciphertext exists, and let $(\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ be such a PKE. We write $M_\lambda = \{1^{\ell(\lambda)}\} \times \{0,1\}^\lambda \times \{0,1\}^{\ell(\lambda)} \times \{0,1\}^{\ell(\lambda)}$ and define the promise problem $\mathcal{L} = (\mathcal{L}_{\text{yes}}, \mathcal{L}_{\text{no}})$ as follows:

$$\mathcal{L}_{\text{yes}} = \left\{ (1^{\ell(\lambda)},\, m',\, \mathsf{pk},\, \mathsf{c}) \in M_\lambda \;\middle|\; \begin{array}{l} \exists\, \mathsf{sk},\ \exists\, m_0 \in \{0,1\}^\lambda,\ \text{such that } m' \prec m_0, \\ \Pr\left[m = \mathsf{Dec}\big(\mathsf{Enc}(m, \mathsf{pk}),\, \mathsf{sk}\big)\right] \geq 1 - \mathsf{negl}(\lambda), \\ \Pr\left[\mathsf{Dec}(\mathsf{c}, \mathsf{sk}) = m_0\right] \geq 1 - \mathsf{negl}(\lambda). \end{array} \right\}$$

$$\mathcal{L}_{\text{no}} = \left\{ (1^{\ell(\lambda)},\, m',\, \mathsf{pk},\, \mathsf{c}) \in M_\lambda \;\middle|\; \begin{array}{l} \forall\, \mathsf{sk}, \forall\, m_0 \in \{0,1\}^\lambda,\ \text{either } m' \not\prec m_0 \text{ or} \\ \Pr\left[m = \mathsf{Dec}(\mathsf{Enc}(m, \mathsf{pk}), \mathsf{sk})\right] \leq 1 - \frac{1}{\lambda} \text{ or} \\ \Pr\left[\mathsf{Dec}(\mathsf{c}, \mathsf{sk}) = m_0\right] \leq 1 - \frac{1}{\lambda}. \end{array} \right\} \tag{5.4}$$

1. $\mathcal{L} \in \mathsf{QMA}$.

   We define a quantum polynomial time algorithm that can decide $\mathcal{L}$ as follows, given $m_0 \in \{0,1\}^\lambda$ and $\mathsf{sk} \in \{0,1\}^{\ell(\lambda)} v$.

   Algorithm 5.3. *Input:* $(1^{\ell(\lambda)}, m', \mathsf{pk}, \mathsf{c}) \in M_\lambda$ *and* $(m_0, \mathsf{sk}) \in \{0,1\}^\lambda \times \{0,1\}^{\ell(\lambda)}$.

   (a) If $m' \not\prec m_0$, return 1.

   (b) For $1 \leq i \leq 2\lambda$, compute $m_i = \mathsf{Dec}(\mathsf{c}, \mathsf{sk})$. If $m_i \neq m_0$, return 0.

   (c) For $1 \leq i \leq 2\lambda$, sample $m_i \leftarrow\!\!\$\ \{0,1\}^\lambda$. If $m_i \neq \mathsf{Dec}(\mathsf{Enc}(m_i, \mathsf{pk}), \mathsf{sk})$, return 0.

   (d) Return 1.

   It is easy to see that if $(1^{\ell(\lambda)}, m', \mathsf{pk}, \mathsf{c}) \in \mathcal{L}_{\text{yes}}$, then there exist $m_0 \in \{0,1\}^\lambda$ and sk such that Algorithm 5.3 output 1 on input $((1^{\ell(\lambda)}, m', \mathsf{pk}, \mathsf{c}), (m_0, \mathsf{sk}))$ with probability $1 - \mathsf{negl}(\lambda)$.

   Note that since the decryption algorithm succeeds with probability exponentially close to 1, by the gentle lemma, the secret key is not disturbed much, hence a polynomial amount of encryption is possible.

Now suppose that $(1^{\ell(\lambda)}, m', \mathsf{pk}, \mathsf{c}) \in \mathcal{L}_{\text{no}}$, and let $m_0 \in \{0,1\}^\lambda$ and sk be a secret key. If $m' \neq m_0$, then Algorithm 5.3 output 0 with probability 1. Otherwise, assume that we have that $\Pr[m = \mathsf{Dec}(\mathsf{Enc}(m, pk), sk)] \leq 1 - \frac{1}{\lambda}$. Hence we have that Algorithm 5.3 does not output 0 on the first loop with probability at most

$$\left(1 - \frac{1}{\lambda}\right)^{2\lambda} \leq \exp(-2) \leq \frac{1}{3}.$$

Similarly, if $\Pr[\mathsf{Dec}(ct, sk) = m_0] \leq 1 - \frac{1}{\lambda}$, we can conclude that algorithm 5.3 output 0 with probability at least $\frac{2}{3}$.

Also note that even if the secret key is partially destroyed when applying the decryption algorithm, this is not a problem. Indeed Equation (5.4) holds for any secret key sk, thus is also holds for the partially destroyed secret key.

2. $\mathcal{L} \notin \mathsf{BQP}$.

   By contradiction, assume that $\mathcal{L} \in \mathsf{BQP}$. Let $(\mathsf{pk}, \mathsf{sk}) = \mathsf{KGen}(1^\lambda)$, and $\mathsf{c}_0 = \mathsf{Enc}(m_0, \mathsf{sk})$ for some $m_0 \leftarrow\!\!\$\ \{0,1\}^\lambda$. There exists a quantum polynomial time algorithm that, on input $(\mathsf{pk}, \mathsf{c}_0)$, can find an m_1 such that with non-negligible probability, there exists sk' such that[2]

$$\Pr[m = \mathsf{Dec}(\mathsf{Enc}(m, \mathsf{pk}), \mathsf{sk}')] \geq 1 - \frac{1}{\lambda}, \text{ and} \tag{5.5}$$

$$\Pr\left[m_1 = \mathsf{Dec}(\mathsf{c}_0, \mathsf{sk}')\right] \geq 1 - \frac{1}{\lambda}. \tag{5.6}$$

   Let

$$M = \left\{ m_0 \middle| \Pr\left[\mathsf{Dec}(\mathsf{Enc}(m_0, \mathsf{pk}), \mathsf{sk}') = m_0\right] > \frac{2}{3} \right\}.$$

   By contradiction, assume that $\Pr\left[m_0 \in M \;\middle|\; m_0 \leftarrow\!\!\$\ \{0,1\}^\lambda\right] < \frac{1}{6}$. Then, we have that:

$$\begin{aligned}\Pr\left[m_0 = \mathsf{Dec}(\mathsf{Enc}(m_0, \mathsf{pk}), \mathsf{sk}')\right] &= \Pr\left[m_0 = \mathsf{Dec}(\mathsf{Enc}(m_0, \mathsf{pk}), \mathsf{sk}') \mid m_0 \in M\right] \Pr\left[m_0 \in M\right] \\ &\quad + \Pr\left[m_0 = \mathsf{Dec}(\mathsf{Enc}(m_0, \mathsf{pk}), \mathsf{sk}') \mid m_0 \notin M\right] \Pr\left[m_0 \notin M\right] \\ &\leq \frac{1}{6} + \frac{2}{3} = \frac{5}{6},\end{aligned}$$

   which contradicts Equation (5.5) for large enough λ. Thus, with probability at least $\frac{1}{6}$ over $m_0 \leftarrow\!\!\$\ \{0,1\}^\lambda$, we have that

$$\Pr\left[\mathsf{Dec}(\mathsf{Enc}(m_0, \mathsf{pk}), \mathsf{sk}') = m_0\right] > \frac{2}{3} \tag{5.7}$$

[2] A search-to-decision reduction is needed but omitted here.

Let us now assume that we are in the case where Equation (5.7) holds. Then, by a similar averaging argument, we have that with probability at least $\frac{1}{24}$ over $\mathsf{c}_0 \leftarrow \mathsf{Enc}(m_0, \mathsf{sk})$, we have that[3]:

$$\Pr\left[m_0 = \mathsf{Dec}(c_0, \mathsf{sk}')\right] \geq \frac{7}{12}. \tag{5.8}$$

If we are in the case where Equation (5.8) holds, then by Equation (5.6) we have that $m_0 = m_1$. And in that case, the output of the algorithm breaks the security of the scheme.

Thus, with probability at least $\frac{1}{24} \cdot \frac{1}{6}$, the algorithm breaks the security of the scheme, which is impossible. Thus $\mathcal{L} \notin \mathsf{BQP}$ which concludes the proof.

□

5.3.3 (q,c,c) Encryption

In this section, we show that public key encryption with classical ciphertexts and classical secret keys cannot be build from PRUs in a black-box way. Similarly to the previous section, we show that the existence of such a scheme imply that $\mathsf{BQP} \neq \mathsf{QMA}$, and the proof relativizes. We note that a previous work [AGL24] shows that (q,c,c)-PKE cannot be build from PRSs, which is a weaker primitive than PRUs. Thus we improve their result, by showing that PRUs cannot be used either to build (q,c,c)-PKE in a black-box way.

The proof follows the idea of the beginning of the proof of [CM24] that shows that digital signatures with classical signatures, classical secret key and quantum public key are separated from PRUs. We note that the results of this subsection are somewhat folklore, as the proof of [CM24] can easily be adapted to this setting, with a shorter proof. However, we include the proof here for completeness, and also because the oracle that we use (Kretschmer's oracle) is slightly weaker than the one of [CM24].

Theorem 5.9. *Let $\mathcal{O}$ be an oracle relative to which PKE with classical ciphertexts and classical secret keys exist. Then $\mathsf{BQP}^{\mathcal{O}} \neq \mathsf{QMA}^{\mathcal{O}}$.*

Proof. Let $\mathcal{O}$ be an oracle relative to which PKE with classical ciphertexts and classical secret keys exist, and let $(\mathsf{Gen}, \mathsf{QPKGen}, \mathsf{Enc}, \mathsf{Dec})$ be such a scheme.

Let $s = 200\lambda$ and $M_\lambda = \{1^{\ell(\lambda)}\} \times \{0,1\}^{\lambda\cdot(2s+1)}$. We define the following promise problem $\mathcal{L} = (\mathcal{L}_{\text{yes}}, \mathcal{L}_{\text{no}})$

$$\mathcal{L}_{\text{yes}} = \left\{ \left(1^{\ell(\lambda)}, (\mathsf{c}_0, \ldots, \mathsf{c}_s), (m', m_1, \ldots, m_s)\right) \in M_\lambda \middle| \begin{array}{l} \exists\, \mathsf{sk}, \text{ such that } \mathsf{QPKGen}(\mathsf{sk}) = \mathsf{pk}, \\ \exists\, m_0 \in \{0,1\}^\lambda, \text{ such that } m' \prec m_0, \\ \forall\, i, \Pr\left[\mathsf{Dec}(\mathsf{c}_i, \mathsf{sk}) = m_i\right] \geq 1 - \mathsf{negl}(\lambda). \end{array} \right\}$$

[3]That is because $\frac{1}{24} + \frac{7}{12} \leq \frac{2}{3}$

$$\mathcal{L}_{\text{no}} = \left\{ \begin{pmatrix} 1^{\ell(\lambda)}, (\mathsf{c}_0, \ldots, \mathsf{c}_s), \\ (m', m_1, \ldots, m_s) \end{pmatrix} \in M_\lambda \middle| \begin{array}{l} \forall\, \mathsf{sk} \text{ such that } \mathsf{QPKGen}(\mathsf{sk}) = \mathsf{pk}, \\ \forall\, m_0 \in \{0,1\}^\lambda, \text{ either } m' \not\approx m_0 \text{ or} \\ \exists\, i, \Pr\left[\mathsf{Dec}(\mathsf{c}_i, \mathsf{sk}) = m_i\right] \leq \frac{1}{\lambda}. \end{array} \right\}$$

1. $\mathcal{L} \in \mathsf{QMA}$.

   We define a quantum polynomial time algorithm that can decide $\mathcal{L}$ as follows, given $m_0 \in \{0,1\}^\lambda$ and $\mathsf{sk} \in \{0,1\}^{\ell(\lambda)}$.

   Algorithm 5.4. *$(1^{\ell(\lambda)}, (\mathsf{c}_0, \ldots, \mathsf{c}_s), (m', m_1, \ldots, m_s)) \in M_\lambda$ and $(m_0, \mathsf{sk}) \in \{0,1\}^\lambda \times \{0,1\}^{\ell(\lambda)}$:*

   (a) If $m' \not\approx m_0$, return 1.

   (b) For $1 \leq i \leq s$, and $1 \leq j \leq 2\lambda$, compute $m_{i,j} = \mathsf{Dec}(\mathsf{c}_i, \mathsf{sk})$. If $m_{i,j} \neq m_i$, return 0.

   (c) For $1 \leq i \leq 2\lambda$, sample $m_i' \leftarrow\!\!\$\ \{0,1\}^\lambda$. If $m_i' \neq \mathsf{Dec}(\mathsf{Enc}(m_i', \mathsf{pk}), \mathsf{sk})$, return 0.

   (d) Return 1.

   It is easy to see that if $(1^{\ell(\lambda)}, m', \mathsf{pk}, \mathsf{c}) \in \mathcal{L}_{\text{yes}}$, then there exist $m_0 \in \{0,1\}^\lambda$ and sk such that Algorithm 5.4 output 1 on input $((1^{\ell(\lambda)}, (m', m_1, \ldots, m_s), (\mathsf{c}_0, \ldots, \mathsf{c}_s)), (m_0, \mathsf{sk}))$ with probability $1 - \mathsf{negl}(\lambda)$.

   Now suppose that $(1^{\ell(\lambda)}, (m', m_1, \ldots, m_s), (\mathsf{c}_0, \ldots, \mathsf{c}_s)) \in \mathcal{L}_{\text{no}}$, and let $m_0 \in \{0,1\}^\lambda$ and sk be a secret key. If $m' \not\approx m_0$, then Algorithm 5.3 output 0 with probability 1. Otherwise, assume that we have that $\Pr[m = \mathsf{Dec}(\mathsf{Enc}(m, pk), sk)] \leq 1 - \frac{1}{\lambda}$. Hence we have that Algorithm 5.4 does not output 0 on the first loop with probability at most

   $$\left(1 - \frac{1}{\lambda}\right)^{2\lambda} \leq \exp(-2) \leq \frac{1}{3}.$$

   Similarly, if $\Pr[\mathsf{Dec}(\mathsf{c}_i, sk) = m_i] \leq 1 - \frac{1}{\lambda}$ for some $1 \leq i \leq s$, we can conclude that algorithm 5.3 output 0 with probability at least $\frac{2}{3}$.

2. $\mathcal{L} \notin \mathsf{BQP}$.

   By contradiction, assume that $\mathcal{L} \in \mathsf{BQP}$. Let $\mathsf{pk} = \mathsf{KGen}(1^\lambda)$, and for $1 \leq i \leq s$, $\mathsf{c}_i = \mathsf{Enc}(m_i, \mathsf{QPKGen}(\mathsf{pk}))$ for some $m_i \leftarrow\!\!\$\ \{0,1\}^\lambda$. Finally, we pick the challenge $m_0 \leftarrow\!\!\$\ \{0,1\}^\lambda$, and compute the ciphertext $\mathsf{c}_0 = \mathsf{Enc}(m_0, \mathsf{QPKGen}(\mathsf{pk}))$. By assumption, there exists a QPT algorithm that, on input $(m_1, \ldots, m_s), (\mathsf{c}_0, \ldots, \mathsf{c}_s)$, can find an $\widetilde{m}_0$ such that there exists sk',

   $$\Pr\left[\mathsf{Dec}(\mathsf{c}_0, \mathsf{sk}') = \widetilde{m}_0\right] \geq 1 - \frac{1}{\lambda} \text{ and } \forall 1 \leq i \leq s, \Pr\left[\mathsf{Dec}(\mathsf{c}_i, \mathsf{sk}') = m_i\right] \geq 1 - \frac{1}{\lambda}.$$

   We will need the following claim to finish the proof.

Claim 5.10. *Unless with probability $2^{-\lambda}$, we have that:*

$$\Pr\left[m = \mathsf{Dec}(\mathsf{Enc}(m, \mathsf{QPKGen}(\mathsf{sk})), \mathsf{sk}')\right] \geq \frac{2}{3},$$

Proof. We write, for any sk^*,

$$p(\mathsf{sk}^*) = \Pr\left[m = \mathsf{Dec}(\mathsf{Enc}(m, \mathsf{QPKGen}(\mathsf{sk})), \mathsf{sk}^*)\right].$$

Fix sk^*, and assume that $p(\mathsf{sk}^*) < \frac{2}{3}$. Then, by an averaging argument, there is a fraction at least $\frac{1}{9}$ of the m such that:

$$\Pr\left[m = \mathsf{Dec}(\mathsf{Enc}(m, \mathsf{QPKGen}(\mathsf{sk})), \mathsf{sk}^*)\right] \leq \frac{3}{4}.$$

By an other averaging argument, for those m, with probability at least $\frac{1}{16}$ over the choice of $\mathsf{c} = \mathsf{Enc}(m, \mathsf{QPKGen}(\mathsf{sk}))$, we have that:

$$\Pr\left[m = \mathsf{Dec}(\mathsf{c}, \mathsf{sk}^*)\right] < \frac{4}{5}.$$

Thus if $p(\mathsf{sk}^*) < \frac{2}{3}$, with probability $\frac{1}{16} \times \frac{1}{9} = \frac{1}{144}$ over the choice of m and c, we have that:

$$\Pr\left[m = \mathsf{Dec}(\mathsf{c}, \mathsf{sk}^*)\right] < \frac{4}{5}.$$

Now recall that for $(1^{\ell(\lambda)}, (m', m_1, \ldots, m_s), (\mathsf{c}_1, \ldots, \mathsf{c}_s)) \notin \mathcal{L}_{\text{no}}$, we have that:

$$\forall i, \Pr\left[m = \mathsf{Dec}(\mathsf{c}_i, \mathsf{sk}^*)\right] \geq 1 - \frac{1}{\lambda}.$$

This mean that we can bound, for any sk^*,

$$\Pr\left[\forall i, \Pr\left[m = \mathsf{Dec}(\mathsf{Enc}(m_i, \mathsf{sk}), \mathsf{sk}^*)\right] \geq 1 - \frac{1}{\lambda} \,\middle|\, p(sk^*) < \frac{2}{3}\right] \leq \left(\frac{143}{144}\right)^s \leq 2^{-2\lambda},$$

where the probability is over the choice of the m_i and the randomness of the encryption algorithm. And finally, by a union bound

$$\Pr\left[\exists \mathsf{sk}^*, \forall i, \Pr\left[m = \mathsf{Dec}(\mathsf{Enc}(m_i, \mathsf{sk}), \mathsf{sk}^*)\right] \geq 1 - \frac{1}{\lambda} \text{ and } p(\mathsf{sk}^*) < \frac{2}{3}\right] \leq 2^{-\lambda},$$

for sufficiently large λ. This proves the claim. □

From Claim 5.10 and using an averaging argument, we have that with probability at least $\frac{1}{24}$ over the choice of m_0,

$$\Pr\left[m_0 = \mathsf{Dec}(\mathsf{Enc}(m_0, \mathsf{QPKGen}(\mathsf{sk})), \mathsf{sk}')\right] \geq \frac{7}{12}.$$

In that case, using an other averaging argument, we have that with probability at least $\frac{1}{24}$ over the choice of c_0,

$$\Pr\left[m_0 = \mathsf{Dec}(c_0, \mathsf{sk}')\right] \geq \frac{3}{7}.$$

In that case, we have that $m_0 = m_0^*$ for large enough λ. Thus, with probability at least $\frac{1}{24} \times \frac{1}{24}$, $m_0 = \widetilde{m}_0$.

This breaks the security property of the public key encryption scheme, which is impossible by assumption. Thus $\mathcal{L} \notin \mathsf{BQP}$ which concludes the proof.

□

5.3.4 Overview

In Figure 5.1 we summarize the state of the art of the feasibility of black-box constructions of public key encryption from pseudorandom unitaries. The figure, which is depicted as a cube, helps illustrate the relationship between the different kind of public key encryption schemes, highlighting that having a classical string is stronger than having a quantum state. In terms of feasibility result, the strongest possible construction is a (c,c,c)-PKE scheme, hence the closer the construction is to this point, the stronger it is. Conversely, the weakest possible construction is a (q,q,q)-PKE scheme, so the nearer a separation is to this point, the stronger it is. Two open questions remain regarding whether (c,q,q)-PKE and (q,q,c)-PKE can be constructed from PRUs in a black-box way. We note that in Figure 6.1 we compare this figure with a similar one that presents feasibility result from one-way functions in a black-box way.

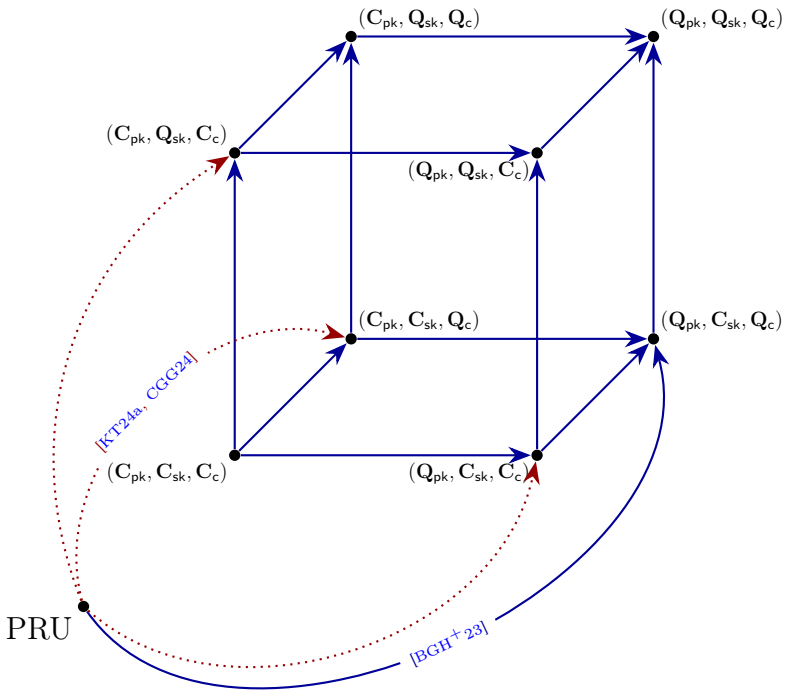


Figure 5.1: Summary of feasibility and impossibility result of public key encryption (PKE) from pseudorandom unitaries (PRUs). Each triplet corresponds to a different kind of PKE, where the order is $(\mathsf{pk}, \mathsf{sk}, \mathsf{c})$, and a $\mathbf{C}$ indicates a classical string, and a $\mathbf{Q}$ indicates a quantum state. A blue arrow indicates a black-box construction, and a dotted red arrow indicates a black-box separation.

5.4 On Constructing Signatures

The goal of this section is to study the relationship between (one-time and digital) signatures and PRUs in a black-box way. Remember that we let $(\mathsf{vk}, \mathsf{sk}, \mathsf{s}) \in \{c, q\}^3$ and discuss $(\mathsf{vk}, \mathsf{sk}, \mathsf{s})$ signatures where a c indicate a classical string, while a q indicate a quantum state.

5.4.1 (c,c,c) and (c,c,q) Signatures

In this subsection, we discuss signatures with classical verification key, classical secret key, and quantum signature. It has been proven that (c,c,q) signatures are separated from PRUs in previous work [KT24a, CGG24]. Similar to the case of public key encryption with classical keys, one can prove that the existence of (c,c,q) signature implies the existence of EV-OWPs.

5.4.2 (c,q,c) Signatures

In this subsection, we discuss signatures with classical verification key, quantum secret key, and classical signature. This case was also tackled in previous work [KT24a]. Similar to the previous case, the existence of (c,q,c) signatures implies the existence of EV-OWPs. Very briefly, the puzzle is the verification key, and the secret is a signature of 0.

5.4.3 (q,c,c) Signatures

In this subsection, we discuss signatures with quantum verification key, classical secret key, and classical signature. One-time (q,c,c) signatures can be constructed from PRSs [MY22b].

Multi-time (q,c,c) signatures schemes are known to be separated from PRUs in a black-box way from previous work [CM24].

5.4.4 (c,q,q) Signatures

In this subsection, we discuss signatures with classical verification key, quantum secret key, and quantum signature. We will show that (c,q,q) signatures imply the existence of EV-OWSPs. In fact, we even have that such signatures schemes are equivalent to EV-OWSPs.

Theorem 5.11. *One-time signatures with quantum secret keys, classical verification keys and quantum signatures are equivalent to EV-OWSPs in a black-box way.*

Proof. The proof is similar to the proof of [CGG24]. We include a sketch for completeness.

1. Signature imply EV-OWSP.

   Let $\mathsf{KGen}(\cdot), \mathsf{Sign}(\cdot), \mathsf{Ver}(\cdot)$ be a signature scheme with quantum secret keys, classical verification keys and quantum signatures. We define the following EV-OWSP:

   - $\mathsf{Gen}(1^\lambda) = \mathsf{KGen}(1^\lambda) = (\mathsf{sk}, \mathsf{vk})$.
   - $\mathsf{Ver}(\mathsf{s}, \mathsf{k})$: Sample $\psi \leftarrow\!\!\$\ \{0,1\}^*$. Return $\mathsf{Ver}(\mathsf{Sign}(\psi, \mathsf{s}), \mathsf{k})$.

   Correctness of the EV-OWSP follows from the correctness of the signature scheme. Security follows from the security of the signature scheme: if an attacker can break the puzzle, it can forge a signature with high probability, which is impossible.

2. EV-OWSP imply signature.

   Let $\mathsf{Gen}(\cdot), \mathsf{Ver}(\cdot)$ be a EV-OWSP. We define the following signature:

   - $\mathsf{KGen}(1^\lambda)$: Let $(\mathsf{s}_0, \mathsf{k}_0) = \mathsf{Gen}(1^\lambda)$ and $(\mathsf{s}_1, \mathsf{k}_1) = \mathsf{Gen}(1^\lambda)$. The algorithm output $(\mathsf{sk} = (\mathsf{s}_0, \mathsf{s}_1), \mathsf{pk} = (\mathsf{k}_0, \mathsf{k}_1))$.
   - $\mathsf{Sign}(\mathsf{sk}, m)$: If $m = 0$, output s_0, otherwise, output s_1.
   - $\mathsf{Ver}(\mathsf{vk}, \mathsf{s}, m)$: Let $\mathsf{vk} = (\mathsf{k}_0, \mathsf{k}_1)$. Output $\mathsf{Ver}(\mathsf{s}, \mathsf{k}_m)$.

Correctness follows from the correctness of the EV-OWSP. Security follows from the security of the EV-OWSP: if an attacker can forge a signature, it can find a state that passes the verification of the EV-OWSP with high probability.

□

Discussion on EV-OWSPs It is proven in [GMMY24] that there exists an oracle $\mathcal{O}$ relative to which OWSPs exist but $\mathsf{BQP} = \mathsf{QCMA}$. In fact, one can easily see that EV-OWSPs exist relative to $\mathcal{O}$. We note however that their oracle is a CPTP map, making it a non standard oracle. This still indicate that EV-OWSPs seems weaker than EV-OWPs.

We also note that (c,q,c)-PKE imply EV-OWSPs with a similar proof. Moreover, we have that QCCC-NIKE imply (c,q,c)-PKE with the standard construction of PKE from NIKE, and noticing that if the NIKE has classical communication, then the PKE scheme has classical public keys and classical ciphertexts. Since QCCC-NIKE exist relative to $\mathcal{O}$, this imply that (c,q,c)-PKE also exist relative to $\mathcal{O}$. We proved in Section 5.3.2 that the existence of (c,q,c)-PKE scheme imply that $\mathsf{BQP} \neq \mathsf{QMA}$ necessarily, thus is also hold relative to $\mathcal{O}$. Thus $\mathsf{BQP}^{\mathcal{O}} = \mathsf{QCMA}^{\mathcal{O}} \neq \mathsf{QMA}^{\mathcal{O}}$, but EV-OWSPs exist relative to $\mathcal{O}$.

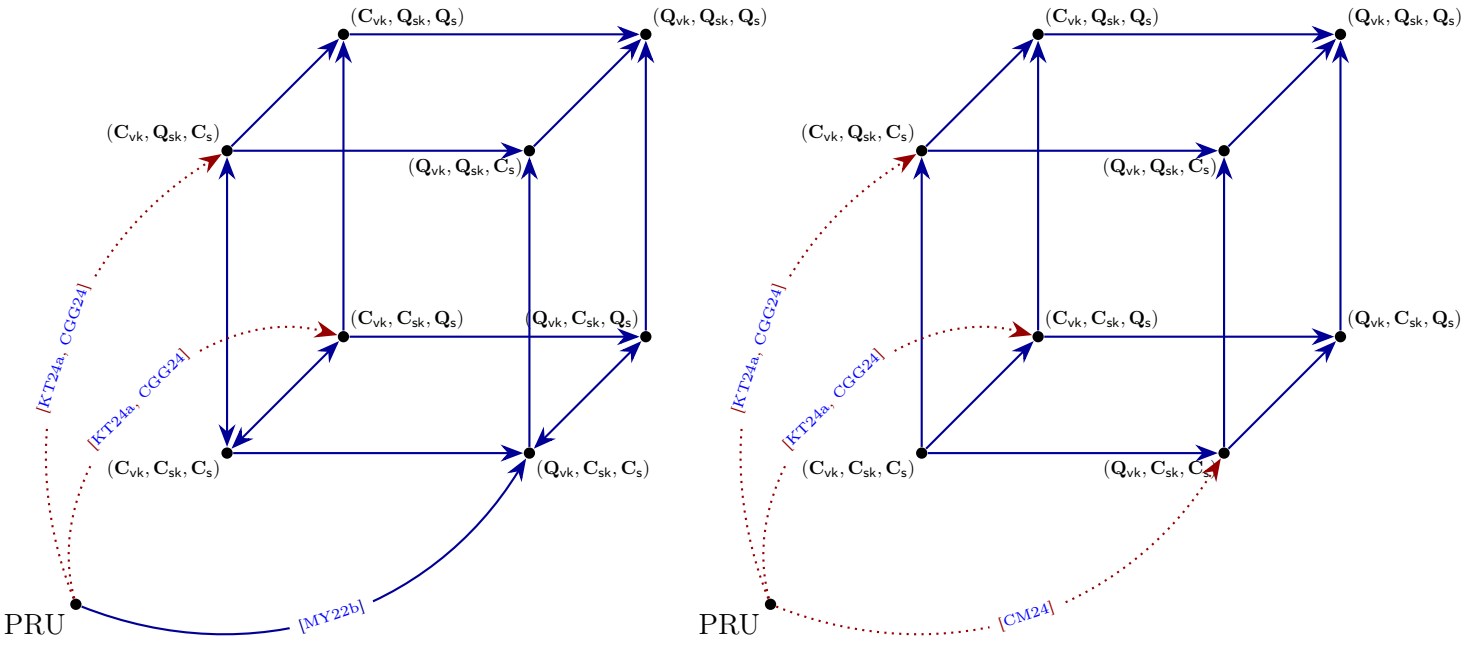
An interesting open question is: can EV-OWSPs exist if $\mathsf{BQP} = \mathsf{QMA}$? More generally, what is the computational complexity characterization of the existence of EV-OWSPs? The last paragraph suggests that $\mathsf{BQP} = \mathsf{QCMA}$ is not enough to break EV-OWSPs, what about $\mathsf{BQP} = \mathsf{QMA}$? Do EV-OWSPs exist relative to Kretschmer's oracle?

5.4.5 (q,c,q) Signatures

In this subsection, we discuss signatures with quantum verification key, classical secret key, and quantum signature. One-time (q,c,q) signatures can be constructed from PRSs [MY22b]. In fact, one can easily see that they also imply OWSGs using similar techniques as in the previous subsection, and thus are equivalent to one-time (q,c,c) signatures.

5.4.6 Overview

In Figure 5.2 we summarize the state of the art regarding the feasibility of black-box constructions of one-time and digital signatures from pseudorandom unitaries. Similar to the public key encryption figure, this figure is represented as a cube, and the same observations apply. Notably some one-time signature schemes are equivalent; in fact, (c,c,c)-one-time signatures, (c,q,c)-one-time signatures and (c,c,q)-one-time signatures are all equivalent. Likewise, (q,c,c)-one-time signatures and (q,c,c)-one-time signatures are equivalent. Furthermore, if EV-OWPs imply digital signatures, then it would follow that (c,c,c)-digital signature, (c,q,c)-digital signature and (c,c,q)-digital signatures are all equivalent.



(a) One-time signature.

(b) Digital signature.

Figure 5.2: Summary of feasibility and impossibility result of one-time signatures and digital signatures from pseudorandom unitaries (PRUs). Each triplet corresponds to a different kind of signature, where the order is $(\mathsf{vk}, \mathsf{sk}, \mathsf{s})$, and a $\mathbf{C}$ indicates a classical string, and a $\mathbf{Q}$ indicates a quantum state. A blue arrow indicates a black-box construction, and a dotted red arrow indicates a black-box separation.

5.5 Common Haar Function-like State Oracles

The rest of this chapter is dedicated to some separations relative to a variant of the Common Haar State Oracle, that we call the Common Haar Function-like State Oracle. We start by defining this model.

5.5.1 CHFS Oracles and Unitarization

We first recall the definition of *swap (or reflection) oracles* [BCN24, CCS24].

Definition 5.1. *For a n-qubit pure quantum state $|\phi\rangle$, the swap (or reflection) unitary is defined by*

$$S_{|\phi\rangle} := |0^n\rangle\langle\phi| + |\phi\rangle\langle 0^n| + I_\perp = I - 2\,|\phi-\rangle\langle\phi-|\,,$$

where we assume w.l.o.g. that $|\phi\rangle$ is orthogonal to $|0^n\rangle$, since if not, we can always append a single $|1\rangle$ to it in order to make it orthogonal. Here, $I_\perp$ is the identity on the subspace orthogonal to $\mathsf{span}\{|0^n\rangle\,, |\phi\rangle\}$ and $|\phi-\rangle = \frac{|0^n\rangle - |\phi\rangle}{\sqrt{2}}$.

The last equality implies that $S_{|\phi\rangle}$ is actually the reflection unitary with respect to $|\phi-\rangle$.

We proceed to define the length-ℓ common Haar-random function-like state (CHFS) oracle and its "unitarized" oracle. We fix a (QPT-computable) function $\ell : \mathbb{N} \to \mathbb{N}$ representing the output length for each oracle, where we typically consider $\ell(\lambda) = \Theta(\log \lambda)$ or $\ell(\lambda) = \lambda$. We define two versions of the CHFS oracles as follows.

Definition 5.2 (The isometry CHFS oracle)**.** *We denote by $\mathcal{O}_\ell$ the distribution over the family of isometry oracles where*

- **Randomness:** *Choose a $\ell(|x|)$-qubit Haar random quantum state $|\phi_x\rangle$ for each $x \in \{0,1\}^*$ and define $\Phi = \{|\phi_x\rangle\}_{x\in\{0,1\}^*}$.*
- **Setup:** *A family of oracles $O^\Phi = (O^\Phi_x)_{x\in\{0,1\}^*} \leftarrow \mathcal{O}_\ell$ is chosen by randomly sampling Φ, where $O^\Phi_x := |\phi_x\rangle\langle 0|$ denotes the isometry operator. Here $|0\rangle$ denotes the trivial quantum state of dimension 1.*
- **Query:** *It takes a quantum state $\rho_{\mathbf{XZ}}$ as input and applies the isometry*

$$O^\Phi := \sum_{x\in\{0,1\}^{|\mathbf{X}|}} |x\rangle\langle x|_{\mathbf{X}} \otimes O^\Phi_x = \sum_{x\in\{0,1\}^{|\mathbf{X}|}} |x\rangle\langle x|_{\mathbf{X}} \otimes |\phi_x\rangle_{\mathbf{Y}}\,\langle 0|\,,$$

on $\rho_{\mathbf{XZ}}$, where $\mathbf{Y}$ denotes a new $\ell(|\mathbf{X}|)$-qubit register, i.e., appending a new register $\mathbf{Y}$.

We say the CHFS oracle is classical-accessible *if the register $\mathbf{X}$ must always be measured in the computational basis before applying the query. Otherwise, we call the oracle* quantum-accessible*.*

Definition 5.3 (The unitarized CHFS oracle)**.** *We denote by $\mathcal{S}_\ell$ the distribution over the family of unitary oracles where*

- **Randomness:** *Choose a $\ell(|x|)$-qubit Haar random quantum state $|\phi_x\rangle$ for each $x \in \{0,1\}^*$ and define $\Phi = \{|\phi_x\rangle |1\rangle\}_{x\in\{0,1\}^*}$.*[4]
- **Setup:** *A family of oracles $S^\Phi = (S^\Phi_x)_{x\in\{0,1\}^*} \leftarrow \mathcal{S}_\ell$ is chosen by randomly sampling Φ, where $S^\Phi_x := S_{|\phi_x\rangle}$ denotes the reflection operator as defined in definition 5.1.*
- **Query:** *It takes a quantum state $\rho_{\mathbf{XYZ}}$ as input such that $|\mathbf{Y}| = \ell(|\mathbf{X}|) + 1$ and applies the unitary*

$$S^\Phi := \sum_{x\in\{0,1\}^{|\mathbf{X}|}} |x\rangle\langle x|_{\mathbf{X}} \otimes S^\Phi_x = \sum_{x\in\{0,1\}^{|\mathbf{X}|}} |x\rangle\langle x|_{\mathbf{X}} \otimes S_{|\phi_x\rangle},$$

on $\rho_{\mathbf{XYZ}}$, where S_x is applied on the register $\mathbf{Y}$.

The classical-accessible and quantum-accessible unitarized CHFS oracles are defined analogously.

The *(length-ℓ) CHFS model* is defined as follows. The randomness Φ is chosen as an initialization. We note that the sets of randomness Φ used to define the isometry and unitarized CHFS oracles are the same. We omit the superscript Φ if the context makes it clear. Then, all parties have oracle access to the CHFS oracle $O = O^\Phi$ or $S = S^\Phi$. We say the log-length CHFS model for $\ell(\lambda) = \mathcal{O}(\log\lambda)$, and the standard CHFS model for $\ell(\lambda) = \omega\log\lambda$.[5]

We call it the *state* (or *isometry*) CHFS model when the oracle is O^Φ, and the *unitary* (or *swap/reflection*) CHFS model when the oracle is S^Φ.

5.5.2 Construction of PRFSs in the CHFS Model

We show that PRFSs with output length ℓ exist in the length-ℓ CHFS model. Again, we stress that the PRFSs are adaptively-secure by default.

Theorem 5.12. *Quantum-accessible (resp. classical-accessible) (κ, m, ℓ)-PRFSs exist in the length-ℓ quantum-accessible (resp. classical-accessible) CHFS model for any key size $\kappa = \omega(\log\lambda)$ and input size $m = \mathsf{poly}(\lambda)$, regardless of the choice of unitary or isometry models. The same statement even holds relative to the* QPSPACE *oracle.*

Proof. We define the following (κ, m, ℓ)-PRFSs. We explain the construction in the isometry CHFS model, but modifying it to the unitary CHFS model is obvious.

$\mathsf{Gen}^O(k,\cdot)$**:** On the m-qubit input register $\mathbf{X}$, it applies the map

$$|x\rangle_{\mathbf{X}} \to |x\rangle_{\mathbf{X}} \otimes |\phi_{k,x}\rangle\,.$$

This is done by, on input $\rho_{\mathbf{XZ}}$, appending the κ-qubit register $|k\rangle_{\mathbf{K}}$ and makes a query to the oracle O^Φ on the register $\mathbf{KX}$ and discards the registers $\mathbf{K}$.

[4] Here we explicitly append $|1\rangle$ to make the unitary CHFS oracle well-defined w.r.t. definition 5.1.
[5] We usually consider the standard CHFS model with $\ell(\lambda) = \lambda$ for simplicity.

We have that $|k|+|x| = m+\kappa = \mathsf{poly}(\lambda)$ thus Gen can be implemented by a BQP algorithm with a single query to the CHFS oracle with $m+\kappa$ length input.

We claim that this construction is a secure PRFS. More precisely, we prove the following statement: For any algorithm A that makes q queries, it holds that

$$\left|\Pr\left[A^{\mathsf{Gen}(k,\cdot),O} \to 1\right] - \Pr\left[A^{G_{\mathsf{Haar}}(\cdot),O} \to 1\right]\right| = \mathcal{O}\left(\frac{q^2}{2^\kappa}\right),$$

where $G_{\mathsf{Haar}}(x)$ outputs an ℓ-qubit Haar random state $|\psi_x\rangle$. When we consider the classical-accessible model, the upper bound becomes $\mathcal{O}(q/2^\kappa)$.

This is done by reducing it to an unstructured search (cf. [Kre21, Section 5]). Formally, we consider a quantum oracle algorithm B^s for $s \in \{0,1\}^{2^\kappa}$ as follows. Let $\lambda' := \kappa + m$. B samples independent ℓ-qubit Haar random quantum states $\left|\tilde{\phi}_z\right\rangle$ for each $z \in \{0,1\}^{\lambda'}$ and $G_{\mathsf{Haar}}(\cdot)$ as defined in definition 2.5. After the initialization, B runs A, but the queries to the first oracle is answered by $G_{\mathsf{Haar}}(\cdot)$, and the query $z = (k', x) \in \{0,1\}^{\lambda'}$ for any x to the second oracle is answered by $G_{\mathsf{Haar}}(x)$ if $s_{k'} = 1$ and $\left|\tilde{\phi}_z\right\rangle$ if $s_{k'} = 0$.

Let e_k be the all-0 string except for the k-th entry 1, then it holds that

$$\left|\Pr_O\left[A^{\mathsf{Gen}(k,\cdot),O} \to 1\right] - \Pr_{O,G_{\mathsf{Haar}}}\left[A^{G_{\mathsf{Haar}}(\cdot),O} \to 1\right]\right| = \left|\Pr_k\left[B^{e_k} \to 1\right] - \Pr\left[B^{0^\kappa} \to 1\right]\right| = \mathcal{O}\left(\frac{q^2}{2^\kappa}\right),$$

where the last inequality holds because of the BBBV theorem [BBBV97]. The security proof for the unitary CHFS model works by replacing O into S. This concludes the proof. □

5.6 On Separating QPRGs from Short PRFSs

This section presents a candidate separation between QPRGs and log-length PRFSs, which can be rigorously proven under some geometric conjecture about the product Haar measure on states. We first present the conjecture, and then the formal statement together with the proof follows.

5.6.1 The conjecture and candidate separation

Let $X = \mathbb{S}(2^{n_1}) \times \cdots \times \mathbb{S}(2^{n_k})$ be the product space of quantum states equipped with the product Haar measure $\sigma := \sigma_{n_1} \times \cdots \times \sigma_{n_k}$. For two elements $\Phi = (|\phi_1\rangle, \ldots, |\phi_k\rangle), \Psi = (|\psi_1\rangle, \ldots, |\psi_k\rangle)$ in X, we define the max-trace distance $d_{tr}(\Phi, \Psi) := \max_{i\in[k]} \|\phi_i - \psi_i\|_{tr}$. For two subsets S, T of X, we define their distance as $d_{tr}(S,T) := \inf_{\Phi\in S, \Psi\in T} d_{tr}(\Phi, \Psi)$.

We consider the following mathematical conjecture.

Conjecture 5.1. *Let $X = \mathbb{S}(2^{n_1}) \times \cdots \times \mathbb{S}(2^{n_k})$ with the corresponding product Haar measure $\sigma = \sigma_{n_1} \times \cdots \times \sigma_{n_k}$, and let S_0, S_1 be two measurable subsets of X. If $d_{tr}(S_0, S_1) \geq \Delta$ and $\sigma(S_0), \sigma(S_1) \geq \Gamma$, then $\sigma(X \setminus (S_0 \cup S_1)) = \Omega(\Delta^a \Gamma^b)$ for some constants $a, b > 0$.*

Intuitively, the conjecture is stating that regardless of their shape, if two sets have a gap between them, then there must be a non-negligible section of the whole space that they are not covering. For a detailed geometric intuition, we refer the reader to appendix C.

Assuming the conjecture to be true, the candidate separation is with respect to log-length CHFS oracles, relative to which we showed in theorem 5.12 that log-length PRFSs exist. The result is stated in the following theorem

Theorem 5.13. *Relative to the quantum-accessible CHFS oracle $\mathcal{S}_\ell$ with $\ell(\lambda) = \lfloor \log \lambda \rfloor$, there exist adaptively-secure quantum-accessible short PRFSs but QPRGs do not exist unless* $\mathsf{BQP} \neq \mathsf{QCMA}$.

It remains to show the impossibility of QPRGs, which we prove in the next subsection.

5.6.2 Impossibility of QPRGs

In this section, we drop ℓ in $\mathcal{S}$ for simplicity.

Lemma 5.14. *Let $\mathcal{S}$ be the (unitarized) quantum-accessible CHFS oracle with $\ell(\lambda) = \lfloor \log \lambda \rfloor$ and let $A^{\mathcal{S}}$ be a polynomial-query oracle algorithm. Let $p = \mathsf{poly}(\lambda)$ be the maximal length of the CHFS oracles that A accesses. Suppose that there exist $b_\Phi \in \{0,1\}$ such that*

$$\Pr_{\Phi \leftarrow \sigma} \left[\Pr\left(A^{S^\Phi}(1^\lambda) \to b_\Phi \right) = 1 - \mathsf{negl}(\lambda) \right] = 1 - \mathsf{negl}(\lambda). \tag{5.9}$$

Assuming conjecture 5.1 is true, then there exists $b \in \{0,1\}$ such that

$$\Pr_{\Phi \leftarrow \sigma} (b = b_\Phi) = 1 - \mathsf{negl}(\lambda).$$

Proof. Given the upper bound of the maximum query length p, the algorithm accesses a finite number of reflection oracles. Let $X = \mathbb{S}(2^{n_1}) \times \cdots \times \mathbb{S}(2^{n_k})$ be the states[6] to define the CHFS oracle up to the length p, with the corresponding product Haar measure $\sigma = \sigma_{n_1} \times \cdots \times \sigma_{n_k}$. Let $S_0, S_1 \subseteq X$ be defined as

$$S_0 := \left\{ \Phi \in X : \ \Pr\left(A^{S^\Phi}(1^\lambda) \to 0 \right) \geq 2/3 \right\}, \quad S_1 := \left\{ \Phi \in X : \ \Pr\left(A^{S^\Phi}(1^\lambda) \to 1 \right) \geq 2/3 \right\}.$$

By the hypothesis in eq. (5.9), with overwhelming probability over σ, either

$$\Pr\left(A^{S^\Phi} \to 1 \right) \geq 2/3 \quad \text{or} \quad \Pr\left(A^{S^\Phi} \to 0 \right) \geq 2/3,$$

thus $\sigma(X \setminus (S_0 \cup S_1)) = \mathsf{negl}(\lambda)$.

[6] This is implicitly parameterized by λ.

We will prove the theorem by contradiction. Assume that for both $b \in \{0,1\}$, we have $\Pr_{\Phi \leftarrow \sigma}(b = b_\Phi) \leq 1 - 1/\mathsf{poly}(\lambda)$, thus $\sigma(S_b) \leq 1 - 1/\mathsf{poly}(\lambda)$. However, we just proved that $\sigma(S_0 \cup S_1) = 1 - \mathsf{negl}(\lambda)$, hence $\sigma(S_0), \sigma(S_1) \geq 1/\mathsf{poly}(\lambda)\Gamma$ necessarily. Given a pair of elements $\Phi \in S_0$ and $\Psi \in S_1$, we will show that the difference between the applications of classically accessible $\mathcal{S}^\Phi$ and $\mathcal{S}^\Psi$ cannot be too large. Indeed, for every input state $\gamma = \sum_x p_x |x\rangle\langle x| \otimes \gamma_x$, we have

$$\begin{aligned}
\|S^\Phi(\gamma) - S^\Psi(\gamma)\|_{tr} &\leq \sum_x p_x \|S_{|\phi_x\rangle}(\gamma_x) - S_{|\psi_x\rangle}(\gamma_x)\|_{tr} \\
&\leq \sum_x p_x \|S_{|\phi_x\rangle} - S_{|\psi_x\rangle}\|_{op} \|\gamma_x\|_{tr} \\
&\leq \sum_x p_x \sqrt{1 - |\langle\phi_x|\psi_x\rangle|^2} = \sum_x p_x \| |\phi_x\rangle - |\psi_x\rangle \|_{tr} \\
&\leq \sum_x p_x d_{tr}(\Phi, \Psi) \\
&\leq d_{tr}(\Phi, \Psi),
\end{aligned}$$

where we used the structure of the unitarized oracles $S^\Phi = \sum_x |x\rangle\langle x| \otimes S_{|\phi_x\rangle}$, and that the difference of reflection oracles is $\|S_{|\phi\rangle} - S_{|\psi\rangle}\|_{op} = 2\sqrt{1 - |\langle\phi|\psi\rangle|^2}$. This implies that the diamond distance of the unitary oracles S^Φ and S^Ψ must also be at most $2d_{tr}(\Phi, \Psi)$. On the one hand, if the algorithm A makes T queries to the oracles, the subadditivity of the diamond norm under composition implies that

$$\|A^{S^\Phi} - A^{S^\Psi}\|_\diamond \leq 2T d_{tr}(\Phi, \Psi).$$

On the other hand, by definition the diamond norm is the maximum distinguishability of two systems, therefore we can lower bound this quantity by

$$\|A^{S^\Phi} - A^{S^\Psi}\|_\diamond \geq \left|\Pr\left(A^{S^\Phi} \to 1\right) - \Pr\left(A^{S^\Psi} \to 1\right)\right| \geq \frac{1}{3},$$

where the last inequality is obtain from the definition of $\Phi \in S_0$ and $\Psi \in S_1$. Finally, since the lower bound is independent of Φ and Ψ, in particular it also holds for the infimum over the sets S_0 and S_1, this is

$$\Delta := d_{tr}(S_0, S_1) = \inf_{\substack{\Phi \in S_0 \\ \Psi \in S_1}} d_{tr}(\Phi, \Psi) \geq 1/\mathsf{poly}(\lambda).$$

We find therefore ourselves in the hypothesis of conjecture 5.1, thus $\sigma(X \setminus (S_0 \cup S_1)) = \mathsf{poly}(\Delta, \Gamma)$, however this is in contradiction with what we proved earlier, that $\sigma(X \setminus (S_0 \cup S_1)) = \mathsf{negl}(\lambda)$, concluding the proof. □

Remark 5.1. *The proof of lemma 5.14 above can be easily extended to the case of isometry CHFS oracles.*

Lemma 5.15. *Assuming conjecture 5.1 is true, there are no QPRGs relative to the quantum-accessible CHFS oracle $\mathcal{S}_\ell$ with $\ell(\lambda) = \lfloor \log \lambda \rfloor$, unless* $\mathsf{BQP} \neq \mathsf{QMA}$.

Proof. By the above lemma, the classical-output function relative to the short CHFS oracle must output a value independent of the oracle with overwhelming probability. That is, the existence of the QPRGs in this model implies the existence of the QPRGs without any oracle, which is impossible unless $\mathsf{BQP} \neq \mathsf{QMA}$. □

5.7 Toward Separating PRSs from Short-PRSs

In this section we show that, under conjecture 5.1, the output size of a pseudorandom state may be relevant, i.e. there exist short-PRSs but PRSs in a certain form do not exist.

5.7.1 Preparation

Universal oracle. For a quantum oracle algorithm with access to the oracle $O = \{O_\lambda\}_{\lambda\in\mathbb{N}}$, we consider a *universal* oracle $\tilde{O}$ that takes as input a state over two registers $\mathbf{\Lambda X}$, measures the register $\mathbf{\Lambda}$ to obtain λ, then apply O_λ on (the first parts of) $\mathbf{X}$. The (qu)bit-length n of $\mathbf{\Lambda}$ may be specified by $\tilde{O}_n$ if needed, in which case $\tilde{O}_n$ can make queries up to O_{2^n}.

We give the definition here because we explicitly discuss the measurement regarding λ here; the results in the previous section may use the universal oracles implicitly but are not changed.

Pure quantum algorithm, with the isometry CHFS oracles. In this section, we consider quantum oracle algorithms *without* trace-out operators, which we refer by *pure* algorithms, written as

$$A(\cdot) = U_t \circ \tilde{O} \circ \mathcal{N}_t \circ \cdots \circ U_1 \circ \tilde{O} \circ \mathcal{N}_1 \circ U_0(\cdot), \tag{5.10}$$

where each measurement $\mathcal{N}_i$ decides which oracle to query (the parameter λ) on what input x.

Recall that the isometry CHFS oracle with input x outputs $|\phi_x\rangle_{\mathbf{Y}}$ in a new register $\mathbf{Y}$. For the pure algorithm A with the isometry CHFS oracles, we assume that the register $\mathbf{Y}$ was included in the input register of A initialized by $|0\rangle_{\mathbf{Y}}$, but it is never changed until the oracle query is applied. After the query, it becomes $|\phi_x\rangle_{\mathbf{Y}}$ and arbitrary operation may be applied on $\mathbf{Y}$.

When the universal oracle is considered, we assume that some register is initialized by $|0^n\rangle$ for some n and the oracle query uses some qubits of them as Λ, which is measured when the query to the universal oracle is made. Arbitrary operations may be applied to these qubits at any point.

5.7.2 Purity test on the output of pure algorithms

Recall that the *purity* of a quantum state ρ is defined by $\mathrm{Tr}(\rho^2)$ and can be estimated by the swap test as shown in lemma 2.15 on the two copies of ρ. If the outcome of an algorithm is pure, then it can be shown that the initial or intermediate states must have also been pure and the intermediate measurements are deterministic (which is in fact nontrivial). This is the idea behind the following lemma, which states that if the output of a pure quantum algorithm is *nearly* pure, then the intermediate binary measurements are *almost deterministic*, and can be removed at the cost of a negligible difference in the output state.

Note that the measurements in the following lemmas are *binary*; when we apply this lemma, we may implicitly decompose the general measurements into binary measurements.

Lemma 5.16. *Let A be a* pure *quantum algorithm that makes t projective* binary *measurements described by $\{U_0, \mathcal{M}_1, \ldots, \mathcal{M}_t, U_t\}$ for unitaries $U_0, ..., U_t$ and measurements $\mathcal{M}_i = (|0\rangle\langle 0| \otimes I, |1\rangle\langle 1| \otimes I)$ as follows:*

$$A(\cdot) = U_t \circ \mathcal{M}_t \circ \cdots \circ U_1 \circ \mathcal{M}_1 \circ U_0(\cdot), \tag{5.11}$$

where the oracle queries may be included in U_i's.[7] *Suppose that for a pure input state ϕ, there exists an $\varepsilon > 0$, such that $\mathrm{Tr}(A(\phi)^2) \geq 1-\varepsilon$. Define $b_{i+1} := \arg\max_{b\in\{0,1\}} \mathrm{Tr}((|b\rangle\langle b| \otimes I)(U_i \circ \mathcal{M}_i \circ \cdots \circ \mathcal{M}_1$ Then, it holds that the algorithm A can be approximated by projecting only onto the most likely outcomes of the binary measurements*

$$\|U_t \circ (|b_t\rangle\langle b_t| \otimes I) \circ \cdots \circ U_1 \circ (|b_1\rangle\langle b_1| \otimes I) \circ U_0(\phi) - U_t \circ \mathcal{M}_t \circ \cdots \circ U_1 \circ \mathcal{M}_1 \circ U_0(\phi)\|_1 \leq t\varepsilon. \tag{5.12}$$

For any intermediate state ϕ_i right after applying U_i, it also holds that $\mathrm{Tr}((|b_{i+1}\rangle\langle b_{i+1}| \otimes I)\phi_i) \geq 1-\varepsilon$ for all i. Furthermore, assuming conjecture 5.1 is true, there exists an algorithm that learns $b_1, \ldots, b_t$, i.e., the query inputs of A without making any oracle queries with overwhelming probability.

Proof. We rewrite the algorithm A in simpler terms for the proof by considering

$$\mathcal{N}_i := (\Pi_i^0, \Pi_i^1), \quad \text{where} \quad \Pi_i^b := U_0^\dagger \cdots U_{i-1}^\dagger (|b\rangle\langle b| \otimes I) U_{i-1} \cdots U_0,$$

acting on any mixed input state ρ as $\mathcal{N}_i(\rho) = \Pi_i^0 \rho \Pi_i^0 + \Pi_i^1 \rho \Pi_i^1$. The algorithm A can be reformulated as follows[8]:

$$\begin{aligned} A(\rho) &= U_t \circ \cdots \circ U_0 \circ \mathcal{N}_t \circ \cdots \circ \mathcal{N}_1(\rho) \\ &= \sum_{b_1, \cdots, b_t \in \{0,1\}} U_t \cdots U_0 \Pi_t^{b_t} \cdots \Pi_1^{b_1} \rho \Pi_1^{b_1} \cdots \Pi_t^{b_t} U_0^\dagger \cdots U_t^\dagger. \end{aligned} \tag{5.13}$$

[7] This is possible for the isometry oracle as we assume that the output register is not touched before the oracle queries.

[8] Careful readers may be concerned about the isometry oracle implicit in U_i's when using $U_i^\dagger$. We note that the same proof applies to the original algorithm represented as in eq. (5.11); we only use eq. (5.13) for the simplicity of the proof of claim 5.17.

We also define the intermediate states $\{\phi_i\}_{i\in[t]}$ after measurement $\mathcal{N}_i$ as

$$\phi_i := \mathcal{N}_i \circ \cdots \circ \mathcal{N}_1(\phi).$$

The most probable outcomes for the original binary measurements are also simplified with this notation, in particular $b_{i+1} = \arg\max_{b\in\{0,1\}} \mathrm{Tr}\big(\Pi^b_{i+1}\phi_i\big)$, and we define the associated measurement operator

$$\Lambda_{i+1}(\rho) := \Pi^{b_{i+1}}_{i+1}\rho\Pi^{b_{i+1}}_{i+1}.$$

Since the trace-norm is invariant under unitaries, in order to prove the theorem it is enough to show that

$$\|\Lambda_t \circ \cdots \circ \Lambda_1(\phi) - \mathcal{N}_t \circ \cdots \circ \mathcal{N}_1(\phi)\|_{tr} \leq t\varepsilon.$$

It turns out that proving that "it also holds" part suffices for proving the above inequality. In the formulation of this proof, it can be written as follows.

Claim 5.17. *For every $i \in [t]$ and measurement operator $\Lambda_{i+1} := \Pi^{b_{i+1}}_{i+1}\rho\Pi^{b_{i+1}}_{i+1}$., we have*

$$\mathrm{Tr}(\Lambda_{i+1}(\phi_i)) \geq 1 - \varepsilon.$$

We prove that the claim implies the main inequality of the theorem, as the measurement channel and the operator associated with the most likely outcome are closely related. This is, their difference is just the operator associated with the least likely outcome, whose probability of occurring is bounded by claim 5.17:

$$\|\Lambda_{i+1}(\phi_i) - \mathcal{N}_{i+1}(\phi_i)\|_1 = \|\Pi^{1-b_{i+1}}_{i+1}\phi_i\Pi^{1-b_{i+1}}_{i+1}\|_1 = 1 - \mathrm{Tr}\Big(\Pi^{b_{i+1}}_{i+1}\phi_i\Big) \leq \varepsilon,$$

so that $\|\Lambda_{i+1}(\phi_i) - \mathcal{N}_{i+1}(\phi_i)\|_{tr} \leq \varepsilon$. The theorem follows by the triangle inequality as

$$\begin{aligned}
&\|\Lambda_t \circ \cdots \circ \Lambda_1(\phi) - \mathcal{N}_t \circ \cdots \circ \mathcal{N}_1(\phi)\|_{tr} \\
&\quad\leq \|\Lambda_t \circ \cdots \circ \Lambda_1(\phi) - \Lambda_t \circ \cdots \circ \mathcal{N}_1(\phi)\|_{tr} \\
&\qquad + \|\Lambda_t \circ \cdots \circ \Lambda_2 \circ \mathcal{N}_1(\phi) - \Lambda_t \circ \cdots \circ \mathcal{N}_2 \circ \mathcal{N}_1(\phi)\|_{tr} \\
&\qquad\quad + \cdots + \|\Lambda_t \circ \mathcal{N}_{t-1} \circ \cdots \circ \mathcal{N}_1(\phi) - \mathcal{N}_t \circ \mathcal{N}_{t-1} \circ \cdots \circ \mathcal{N}_1(\phi)\|_{tr} \\
&\quad\leq \sum_{i=0}^{t-1} \|\Lambda_{i+1}(\phi_i) - \mathcal{N}_{i+1}(\phi_i)\|_{tr} \leq \sum_{i=0}^{t-1} \varepsilon = t\varepsilon,
\end{aligned}$$

where we used the fact that a quantum channel does not increase the trace norm, see eq. (2.6), for the quantum channel Λ_j in the second inequality. □

Proof of claim 5.17. Note that measurement channels can only decrease purity, this is for all $i \in [t]$:

$$\begin{aligned}
\mathrm{Tr}\big(\phi_{i+1}^2\big) &= \mathrm{Tr}\big(\mathcal{N}_{i+1}(\phi_i)^2\big) \\
&= \mathrm{Tr}\Big(\big(\Pi_{i+1}^0\phi_i\Pi_{i+1}^0 + \Pi_{i+1}^1\phi_i\Pi_{i+1}^1\big)^2\Big) \\
&= \mathrm{Tr}\big(\Pi_{i+1}^0\phi_i\Pi_{i+1}^0\phi_i\Pi_{i+1}^0 + \Pi_{i+1}^1\phi_i\Pi_{i+1}^1\phi_i\Pi_{i+1}^1\big) \\
&\leq \mathrm{Tr}\big(\Pi_{i+1}^0\phi_i^2\big) + \mathrm{Tr}\big(\Pi_{i+1}^1\phi_i^2\big) \\
&= \mathrm{Tr}\big(\phi_i^2\big),
\end{aligned}$$

where we use $\mathrm{Tr}\big(C\rho C^\dagger\big) \leq \mathrm{Tr}(\rho)$ for any unnormalized state $\rho = \phi_i\Pi_{i+1}^b\phi_i$ and quantum channel $C(\cdot) = \Pi_{i+1}^b(\cdot)\Pi_{i+1}^b$, and the cyclicity of the trace.

Moreover, we know by hypothesis of lemma 5.16 that the outcome of the algorithm A is pure with high probability, i.e. $\mathrm{Tr}(\phi_t^2) \geq 1-\varepsilon$. In particular, the above implies that for every $i \in [t]$, the intermediate state ϕ_i is pure with high probability, and hence the channel described by the most probable measurement element must have high probability

$$\begin{aligned}
1-\varepsilon \leq \mathrm{Tr}\big(\phi_t^2\big) &\leq \mathrm{Tr}\big(\phi_{i+1}^2\big) \\
&\leq \mathrm{Tr}\big(\Pi_{i+1}^0\phi_i\Pi_{i+1}^0\big)^2 + \mathrm{Tr}\big(\Pi_{i+1}^1\phi_i\Pi_{i+1}^1\big)^2 \\
&\leq \mathrm{Tr}\Big(\Pi_{i+1}^{b_{i+1}}\phi_i\Pi_{i+1}^{b_{i+1}}\Big)\left(\mathrm{Tr}\big(\Pi_{i+1}^0\phi_i\Pi_{i+1}^0\big) + \mathrm{Tr}\big(\Pi_{i+1}^1\phi_i\Pi_{i+1}^1\big)\right) \\
&\leq \mathrm{Tr}\Big(\Pi_{i+1}^{b_{i+1}}\phi_i\Pi_{i+1}^{b_{i+1}}\Big)\,\mathrm{Tr}(\phi_i) \\
&= \mathrm{Tr}(\Lambda_{i+1}(\phi_i)).
\end{aligned}$$

□

5.7.3 Conditional separation

We now show the following theorem, which is the main result of this section.

Theorem 5.18. *Assuming conjecture 5.1 is true, there exists an isometry oracle $\mathcal{O}$ relative to which (classical-accessible) short-PRFSs exist, but long-PRSs with pure generation algorithms do not.*

The separating oracle $\mathcal{O}$ consists of two oracles: the classical-accessible isometry CHFS oracle O_ℓ for $\ell(\lambda) = \lfloor 2\log\lambda \rfloor$ and the $\mathsf{QPSPACE}$ oracle. The existence of short-PRFSs follows immediately from theorem 5.12. It remains to break long PRSs with pure generation algorithm.

Proof of theorem 5.18. By contradiction, assume there exists a PRS $\mathsf{Gen}(\cdot)$ with a pure generation algorithm relative to $\mathcal{O}$. We write $d(\lambda)$ and $\kappa(\lambda)$ to denote the output length and the key length of the PRS. Since the $\mathsf{QPSPACE}$ oracle is unitary, we can embed them in the unitaries and write the output state of the algorithm $\mathsf{Gen}^{\mathcal{O}}(k) = \rho^{(k)}$ by

$$(U_t^{(k)} \circ \tilde{O}_{n_t}^{(k)} \circ \mathcal{N}_t^{(k)}) \circ \ldots \circ (U_1^{(k)} \circ \tilde{O}_{n_1} \circ \mathcal{N}_1^{(k)}) \circ U_0^{(k)}(|0\rangle\langle 0|^{\otimes d(\lambda)}). \tag{5.14}$$

We omit the superscript (k) when it is clear from the context. Here $U_0, \ldots, U_t$ denote unitary operations and $\mathcal{N}_1, \ldots, \mathcal{N}_t$ are measurements on some registers $\mathbf{\Lambda}_1\mathbf{X}_1, \ldots, \mathbf{\Lambda}_t\mathbf{X}_t$, where $\mathbf{\Lambda}_j$ specifies the index for the CHFS oracle to be applied on $\mathbf{X}_j$. The values $n_1, \ldots, n_t$ denote the size of $\mathbf{\Lambda}_1, \ldots, \mathbf{\Lambda}_t$.

Let us denote by $\rho^{(k)} = \rho_t^{(k)}$ the final state, and we denote by $\rho_j^{(k)}$ the intermediate state right after applying the unitary U_j for $j = 0, \ldots, t-1$. We consider the following adversary $\mathcal{A}$, given the polynomial copies of either $\rho = \rho^{(k)}$ for some k (in which case it outputs 1) or Haar random state ρ (in which case it outputs 0). In the following, let $r = 10\lambda^2$ and $T = 20r^2(2td+1)^3$.

Algorithm 5.5. *$\mathcal{A}$ does to following on input multiple copies of a state ρ.*

1. *$\mathcal{A}$ executes the purity test $16T\lambda$ times on ρ. If the test fails at least 8λ times, $\mathcal{A}$ returns 1 and aborts. Otherwise, it proceeds to the next step.*
2. *$\mathcal{A}$ defines $\widetilde{U}_k = U_t^{(k)} \circ \cdots \circ U_0^{(k)}$. For each k, and $i = 0, \ldots, t-1$, let $(\lambda_i^{(k)}, x_i^{(k)}) = \arg\max_{\lambda,x} \mathrm{Tr}\Big(|\lambda, x\rangle\langle\lambda, x| \, \rho_{i-1}^{(k)}\Big)$. We define the following sub-protocol P_k that takes as input a state $\Psi = (\rho^{\otimes 2})^{\otimes r}$ for $r = 10\lambda^2$:*

   P_k: For each $i \in [M]$, compute $\widetilde{U}_k^\dagger \otimes \widetilde{U}_k^\dagger(\rho \otimes \rho)$ and apply the product test for $\ell(\lambda_1^{(k)}), \ldots, \ell(\lambda_t^{(k)}), 1, \ldots, 1$ qubits, where the number of 1 is $s_k = d - \sum_{i\in[t]} \lambda_i^{(k)}$. Let $m_k = t + s_k$ be the total number of swap tests used in the product test. Return 1 if all tests pass, and return 0 otherwise.

   Then $\mathcal{A}$ runs the quantum OR tester with $\{P_k\}_{k\in\{0,1\}^\kappa}$ on $\Psi = (\rho^{\otimes 2})^{\otimes r}$, and returns the same output.

We first argue that the sub-protocol P_k can be implemented in polynomial time. This is because the $(\lambda_i^{(k)}, x_i^{(k)})$ can be learned without making any query by lemma 5.14.

Claim 5.19. *If $\rho = \rho^{(k)}$ and $\mathrm{Tr}(\rho^2) \geq 1 - 1/T$, then $\Pr[P_k(\Phi)] \geq 4/5$.*

Claim 5.20. *If ρ is Haar random state, then $\Pr[P_k(\Phi) \to 1] \leq 1/2^{2\lambda}$ for all k with probability at least $1 - 1/2^{2\lambda}$.*

The correctness of the algorithm can be shown by the case analysis. If $\rho = \rho^{(k)}$ for some k and $\mathrm{Tr}(\rho^2) \leq 1 - 1/T$, then lemma 2.16 asserts that the first step outputs 1 with probability $1 - 2^{-\lambda}$.

The other case, i.e., $\rho = \rho^{(k)}$ and $\mathrm{Tr}(\rho^2) \geq 1 - 1/T$ or ρ is a true Haar random state is dealt with the quantum or lemma. In this case, by claim 5.19 and claim 5.20, the POVMs $\{P_k\}_{k\in\{0,1\}^\lambda}$ and Ψ satisfies the conditions of the quantum or lemma (lemma 2.19) unless with probability $1/2^\lambda\cdot$. Therefore, $\mathcal{A}$ outputs 1 with probability at least 1/8 if $\rho = \rho^{(k)}$ for some k, but it outputs 1 with probability at most $4/2^\lambda$ if $\rho \leftarrow \nu_n$, that is, $\mathcal{A}$ breaks the PRS security of $\mathsf{Gen}(\cdot)$.

□

Proof of claim 5.19. By the above claim, we can assume that $\mathrm{Tr}(\rho^2) \geq 1 - 1/T$, otherwise Algorithm 5.5 would have terminated at step 1 with probability at least $1 - 2^{-\lambda}$. We can decompose the measurement $\mathcal{N}_i$ by $\mathcal{M}_{i,d_i} \circ ... \circ \mathcal{M}_{i,1}$ for some binary measurements $\mathcal{M}_{i,1}, ..., \mathcal{M}_{i,d_i}$ where $d_i \leq d$, which is bounded by the number of qubits.

Let $\tilde{\rho}_t^{(k)}$ be defined as

$$(U_t^{(k)} \circ \tilde{O}_{n_t} \circ |\lambda_t, x_t\rangle\langle\lambda_t, x_t|) \circ \ldots \circ (U_1^{(k)} \circ \tilde{O}_{n_1} \circ |\lambda_1, x_1\rangle\langle\lambda_1, x_1|) \circ U_0^{(k)}(|0\rangle\langle 0|^{\otimes d(\lambda)}),$$

where we replaced $\mathcal{N}_i$ by $|\lambda_i, x_i\rangle\langle\lambda_i, x_i|$ in eq. (5.14). It is not hard to see that each bit of (λ_i, x_i) coincides with some of b_j defined in lemma 5.16 because $td/T < 1/2$. By lemma 5.16, we have

$$\|\tilde{\rho}_t^{(k)} - \rho_t^{(k)}\|_{tr} \leq \frac{td}{T}. \tag{5.15}$$

Now we give another representation of $\tilde{\rho}_t^{(k)}$. Given fixed (λ_i, x_i), the oracle $\tilde{O}_{n_i}$ generates $|\phi_{x_i}\rangle_{\mathbf{Y}_i}$ that is initialized by $|0\rangle$ and never changed, so we can write

$$\tilde{O}_{n_i} \circ |\lambda_i, x_i\rangle\langle\lambda_i, x_i|_{\mathbf{\Lambda}_i\mathbf{X}_i} \otimes |0\rangle\langle 0|_{\mathbf{Y}_i} = |\lambda_i, x_i\rangle\langle\lambda_i, x_i|_{\mathbf{\Lambda}_i\mathbf{X}_i} \otimes |\phi_{x_i}\rangle\langle 0|_{\mathbf{Y}_i},$$

which allows us to write $\tilde{\rho}_t^{(k)}$ as

$$U_t \circ |\lambda_t, x_t\rangle\langle\lambda_t, x_t| \circ \ldots \circ U_1 \circ |\lambda_1, x_1\rangle\langle\lambda_1, x_1| \circ U_0(|\phi_{x_t}, \ldots, \phi_{x_1}\rangle\langle\phi_{x_t}, \ldots, \phi_{x_1}| \otimes |0\rangle\langle 0|),$$

where $|\phi_{x_t}, \ldots, \phi_{x_1}\rangle$ is stored in the register $\mathbf{Y}_t \ldots \mathbf{Y}_1$. Now let $\tilde{\rho}_j^{(k)}$ be the state after applying U_j in the above equation. We have that $\|\tilde{\rho}_j^{(k)} - \rho_j^{(k)}\|_{tr} \leq \frac{2td}{T}$ using eq. (5.15) for all $j = 0, \ldots, t-1$ and the fact that the quantum channel never increases the trace distance.

By the part "it also holds" of lemma 5.16, for any projector $\Pi = |b\rangle\langle b| \otimes I$ induced from (λ_i, x_i)[9], it holds that

$$\mathrm{Tr}(\Pi\rho_{i-1}) \geq 1 - 1/T. \tag{5.16}$$

Using the triangular inequality, this gives $\mathrm{Tr}(\Pi\tilde{\rho}_{i-1}) \geq 1 - (2td+1)/T$. By applying corollary 2.10 for each binary measurement, we can replace each projectors by identity and use the triangular inequality to derive

$$\|\tilde{\rho}_t^{(k)} - U_t \circ \cdots \circ U_0(|\phi_{x_t}, \ldots, \phi_{x_1}\rangle\langle\phi_{x_t}, \ldots, \phi_{x_1}| \otimes |0\rangle\langle 0|)\|_{tr} \leq 2td \cdot \sqrt{\frac{2td+1}{T}}.$$

[9]In other words, $\Pi = |\lambda_{ij}\rangle\langle\lambda_{ij}| \otimes I$ for $\lambda_i = \lambda_{i1}...\lambda_{in}$ or $\Pi = |x_{ij}\rangle\langle x_{ij}| \otimes I$ for $x_i = x_{i1}...x_{im}$ with some rearrangement of the registers.

Together with eq. (5.15), this implies that

$$\|\rho_t^{(k)} - \tilde{\rho}_t^{(k)}\|_{tr} \leq \frac{td}{T} + 2td \cdot \sqrt{\frac{2td+1}{T}} \leq (2td+1) \cdot \sqrt{\frac{2td+1}{T}}. \tag{5.17}$$

Note that $\Pr\Big[P_k((\tilde{\rho}_t^{(k)})^{\otimes 2r}) \to 1\Big] = 1$ by lemma 2.17. This implies that P_k outputs 1 on input $\Phi = (\rho^{(k)})^{\otimes 2r}$ with probability at least $1 - 2r(2td+1) \cdot \sqrt{\frac{2td+1}{T}} \geq 4/5$. □

Proof of claim 5.20. Here, we need to show that the number of swap test done m_k in the product test is at least 13 for some large enough λ. This is because

$$m_k = t + s_k \geq \frac{t \cdot 2\log\lambda + s_k}{2\log\lambda} \geq \frac{\sum_{i=1}^{t} \ell(\lambda_i^{(k)}) + s_k}{2\log\lambda} = \frac{\omega(\log\lambda)}{2\log\lambda} = \omega(1),$$

where we used the fact that the candidate PRS generator has output dimension $d(\lambda) = \omega(\log\lambda)$.

By lemma 2.18, we have that a single product test (for key k) succeeds with expected probability at most $2 \cdot (3/4)^{13} \leq 0.05$. By the concentration inequality, we can show that with probability at least $1 - 1/2^{2\lambda}$ over Haar random states, a single product test for k succeeds with probability at most 0.1. Using Chernoff's inequality, we conclude that for each k, $\Pr[P_k(\Phi) \to 1] \leq 1/2^{2\lambda}$. □

Chapter 6

Conclusion

This work

In summary, our work explores the rich and complex landscape of quantum cryptographic primitives and highlights several fundamental differences from classical cryptography.

Quantum security of SPHINCS(+) The analysis of cryptographic primitives have been challenging in the quantum setting since its introduction, but remains possible. Our bounds for the subset cover problems and its variants mark a step towards proving formally the security of primitive standardized by the National Institute of Standards and Technology (NIST). There is still a lot of work left open in that direction, as the security of SPHINCS+ in the quantum settin remains to be proven and we believe this to be an important open problem.

Quantum versus Classical Cryptographic Primitives Unlike pseudorandom generators (PRGs) and one-way functions (OWFs), quantum primitives such as pseudorandom quantum states (PRSs) exhibit critical dependencies on their output length. We show that, contrary to the classical setting, shrinking PRSs to a shorter form (short-PRSs) is impossible in a black-box way, and make steps toward an impossibility result for expanding short-PRSs to long PRSs. These impossibility results reveals that the output size is a crucial parameter in quantum pseudorandomness.

Oracle Separations in Idealized Models Our work uses idealized quantum models, namely the Quantum Random Oracle Model (QROM) and the Common Haar Function-Like State model (CHFS), to demonstrate separations between different quantum cryptographic

assumptions. Our work show that key agreements where Alice and Bob exchange classical messages and only the final message is quantum, is unlikely to exist in the QROM.

Overall, our results not only establish new separations but also demonstrate the inherent challenges of constructing and relating classical and quantum cryptographic primitives. This work is a contribution to the quest of understanding the power of quantum computers at a theoretical level, in the domain of cryptography.

Future directions

SPHINCS and SPHINCS+ To assess the security of SPHINCS from [BHH+15, Theorem 1] for concrete parameters such as those proposed in [BHH+15] (namely $h = 60, q_s = 2^{30}$), it would also be necessary to upper-bound the success probabilities $Succ_A((2, k) - SC)$ and $Succ_A((3, k) - SC)$, which we leave for future work. While our work left this question open, the work of [YTA23] give a bound that matches our algorithms, hence it is possible to prove the security of SPHINCS against quantum adversaries. For SPHINCS+ however, there are still conjectures in the security proof against quantum adversaries, thus formally assessing the security of the scheme remains an open question.

Black-box separations in the QROM In the Quantum Random Oracle Model, quantum queries continue to introduce serious analytical challenges. In particular, it remains open whether querying the oracle in superposition can lead to more efficient attacks—so far, all known key-agreement and public-key encryption attacks use only classical queries.

On building quantum public key encryption More generally, there are still restrictions regarding the separations of quantum public key encryption from one-way functions (OWFs), and pseudorandom unitaries (PRUs). We highlight the state of the art of the impossibility and feasibility results of building public key encryption in Figure 6.1.

Minimal assumption for quantum cryptography Finally, there are still a lot of open questions in the field of quantum cryptographic primitives. New tools and assumptions are consistently being discovered, and we believe it is necessary to compare them to each other, to assess their strength. What is the minimal assumption needed to build quantum cryptography? More practically, which purely quantum computational problem can realize that assumption, and more generally, quantum cryptography? For example, Learning With Error can be used to construct post-quantum cryptography, but what would be a quantum assumption that allows to construct pseudorandom quantum states but not one-way functions?

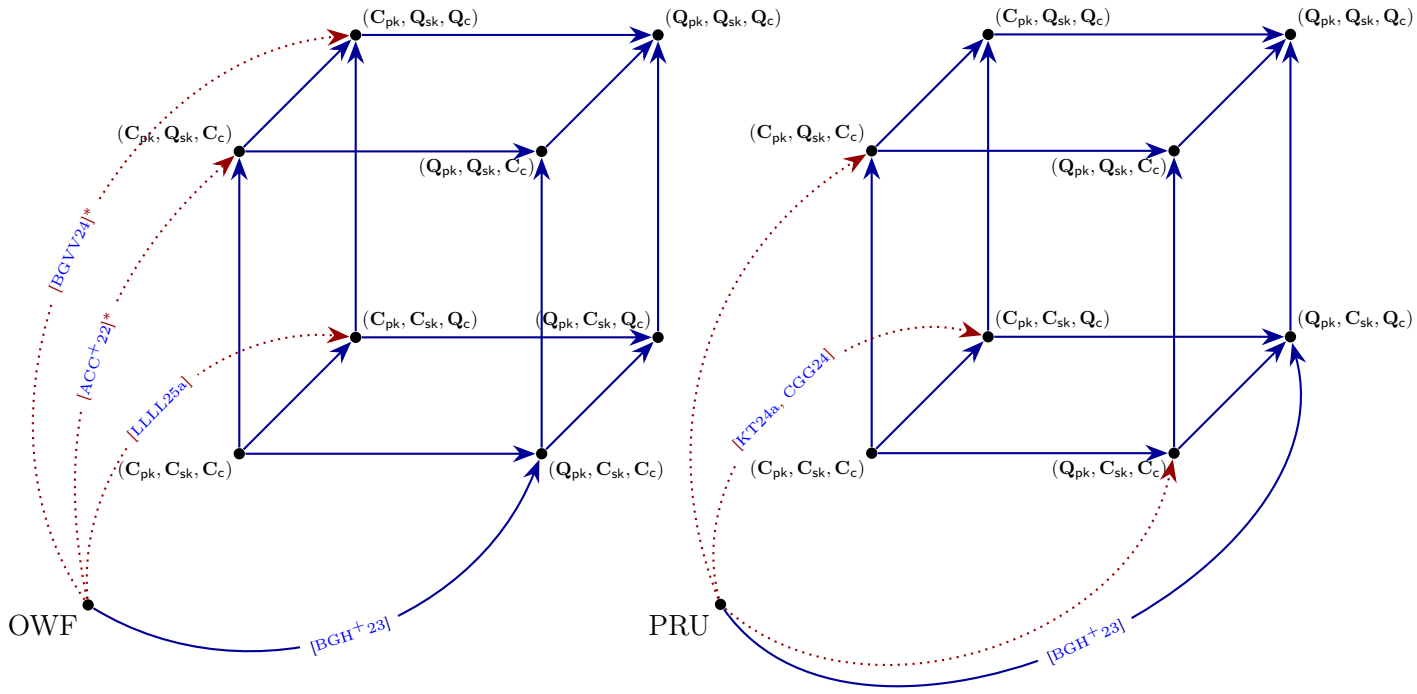


Figure 6.1: Summary of feasibility and impossibility result of public key encryption (PKE) from one-way functions (OWFs) and pseudorandom unitaries (PRUs). Each triplet corresponds to a different kind of PKE, where the order is $(\mathsf{pk}, \mathsf{sk}, \mathsf{c})$, and a $\mathbf{C}$ indicates a classical string, and a $\mathbf{Q}$ indicates a quantum state. A blue arrow indicates a black-box construction, and a dotted red arrow indicates a black-box separation. An asterisk denotes a conditioned result, i.e. assuming Conjecture 2.1 holds.

Bibliography

[AAB+19] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando G. S. L. Brandao, David A. Buell, Brian Burkett, Yu Chen, Zijun Chen, Ben Chiaro, Roberto Collins, William Courtney, Andrew Dunsworth, Edward Farhi, Brooks Foxen, Austin Fowler, Craig Gidney, Marissa Giustina, Rob Graff, Keith Guerin, Steve Habegger, Matthew P. Harrigan, Michael J. Hartmann, Alan Ho, Markus Hoffmann, Trent Huang, Travis S. Humble, Sergei V. Isakov, Evan Jeffrey, Zhang Jiang, Dvir Kafri, Kostyantyn Kechedzhi, Julian Kelly, Paul V. Klimov, Sergey Knysh, Alexander Korotkov, Fedor Kostritsa, David Landhuis, Mike Lindmark, Erik Lucero, Dmitry Lyakh, Salvatore Mandrà, Jarrod R. McClean, Matthew McEwen, Anthony Megrant, Xiao Mi, Kristel Michielsen, Masoud Mohseni, Josh Mutus, Ofer Naaman, Matthew Neeley, Charles Neill, Murphy Yuezhen Niu, Eric Ostby, Andre Petukhov, John C. Platt, Chris Quintana, Eleanor G. Rieffel, Pedram Roushan, Nicholas C. Rubin, Daniel Sank, Kevin J. Satzinger, Vadim Smelyanskiy, Kevin J. Sung, Matthew D. Trevithick, Amit Vainsencher, Benjamin Villalonga, Theodore White, Z. Jamie Yao, Ping Yeh, Adam Zalcman, Hartmut Neven, and John M. Martinis. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, Oct 2019. doi:10.1038/s41586-019-1666-5.

[Aar04] Scott Aaronson. Limitations of quantum advice and one-way communication. In *Proceedings. 19th IEEE Annual Conference on Computational Complexity, 2004.*, pages 320–332. IEEE, 2004.

[Aar05] S. Aaronson. Quantum computing, postselection, and probabilistic polynomial-time. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 461:3473 – 3482, 2005.

[Aar16] Scott Aaronson. The complexity of quantum states and transformations: From quantum money to black holes. *Electron. Colloquium Comput. Complex.*, TR16,

2016. URL: https://api.semanticscholar.org/CorpusID:1869239.

[AARA+25] Morteza Aghaee, Alejandro Alcaraz Ramirez, Zulfi Alam, Rizwan Ali, Mariusz Andrzejczuk, Andrey Antipov, Mikhail Astafev, Amin Barzegar, Bela Bauer, Jonathan Becker, Umesh Kumar Bhaskar, Alex Bocharov, Srini Boddapati, David Bohn, Jouri Bommer, Leo Bourdet, Arnaud Bousquet, Samuel Boutin, Lucas Casparis, Benjamin J. Chapman, Sohail Chatoor, Anna Wulff Christensen, Cassandra Chua, Patrick Codd, William Cole, Paul Cooper, Fabiano Corsetti, Ajuan Cui, Paolo Dalpasso, Juan Pablo Dehollain, Gijs de Lange, Michiel de Moor, Andreas Ekefjärd, Tareq El Dandachi, Juan Carlos Estrada Saldaña, Saeed Fallahi, Luca Galletti, Geoff Gardner, Deshan Govender, Flavio Griggio, Ruben Grigoryan, Sebastian Grijalva, Sergei Gronin, Jan Gukelberger, Marzie Hamdast, Firas Hamze, Esben Bork Hansen, Sebastian Heedt, Zahra Heidarnia, Jesús Herranz Zamorano, Samantha Ho, Laurens Holgaard, John Hornibrook, Jinnapat Indrapiromkul, Henrik Ingerslev, Lovro Ivancevic, Thomas Jensen, Jaspreet Jhoja, Jeffrey Jones, Konstantin V. Kalashnikov, Ray Kallaher, Rachpon Kalra, Farhad Karimi, Torsten Karzig, Evelyn King, Maren Elisabeth Kloster, Christina Knapp, Dariusz Kocon, Jonne V. Koski, Pasi Kostamo, Mahesh Kumar, Tom Laeven, Thorvald Larsen, Jason Lee, Kyunghoon Lee, Grant Leum, Kongyi Li, Tyler Lindemann, Matthew Looij, Julie Love, Marijn Lucas, Roman Lutchyn, Morten Hannibal Madsen, Nash Madulid, Albert Malmros, Michael Manfra, Devashish Mantri, Signe Brynold Markussen, Esteban Martinez, Marco Mattila, Robert McNeil, Antonio B. Mei, Ryan V. Mishmash, Gopakumar Mohandas, Christian Mollgaard, Trevor Morgan, George Moussa, Chetan Nayak, Jens Hedegaard Nielsen, Jens Munk Nielsen, William Hvidtfelt Padkar Nielsen, Bas Nijholt, Mike Nystrom, Eoin O'Farrell, Thomas Ohki, Keita Otani, Brian Paquelet Wütz, Sebastian Pauka, Karl Petersson, Luca Petit, Dima Pikulin, Guen Prawiroatmodjo, Frank Preiss, Eduardo Puchol Morejon, Mohana Rajpalke, Craig Ranta, Katrine Rasmussen, David Razmadze, Outi Reentila, David J. Reilly, Yuan Ren, Ken Reneris, Richard Rouse, Ivan Sadovskyy, Lauri Sainiemi, Irene Sanlorenzo, Emma Schmidgall, Cristina Sfiligoj, Mustafeez Bashir Shah, Kevin Simoes, Shilpi Singh, Sarat Sinha, Thomas Soerensen, Patrick Sohr, Tomas Stankevic, Lieuwe Stek, Eric Stuppard, Henri Suominen, Judith Suter, Sam Teicher, Nivetha Thiyagarajah, Raj Tholapi, Mason Thomas, Emily Toomey, Josh Tracy, Michelle Turley, Shivendra Upadhyay, Ivan Urban, Kevin Van Hoogdalem, David J. Van Woerkom, Dmitrii V. Viazmitinov, Dominik Vogel, John Watson, Alex Webster, Joseph Weston, Georg W. Winkler, Di Xu, Chung Kai Yang, Emrah Yucelen, Roland Zeisel, Guoji Zheng, Justin Zilke, and Microsoft Azure Quantum. Interferometric single-shot parity measurement in inas–al hybrid devices. *Nature*, 638(8051):651–655, Feb 2025. doi:10.1038/s41586-024-08445-2.

[ACC+22] Per Austrin, Hao Chung, Kai-Min Chung, Shiuan Fu, Yao-Ting Lin, and

Mohammad Mahmoody. On the impossibility of key agreements from quantum random oracles. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part II*, volume 13508 of *LNCS*, pages 165–194. Springer, Cham, August 2022. doi:10.1007/978-3-031-15979-4_6.

[AE17] Jean-Philippe Aumasson and Guillaume Endignoux. Clarifying the subset-resilience problem. Cryptology ePrint Archive, Report 2017/909, 2017. URL: https://eprint.iacr.org/2017/909.

[AGKL24] Prabhanjan Ananth, Aditya Gulati, Fatih Kaleoglu, and Yao-Ting Lin. Pseudorandom isometries. In Marc Joye and Gregor Leander, editors, *EUROCRYPT 2024, Part IV*, volume 14654 of *LNCS*, pages 226–254. Springer, Cham, May 2024. doi:10.1007/978-3-031-58737-5_9.

[AGL24] Prabhanjan Ananth, Aditya Gulati, and Yao-Ting Lin. Cryptography in the common Haar state model: Feasibility results and separations. In Elette Boyle and Mohammad Mahmoody, editors, *TCC 2024, Part II*, volume 15365 of *LNCS*, pages 94–125. Springer, Cham, December 2024. doi:10.1007/978-3-031-78017-2_4.

[AGQY22] Prabhanjan Ananth, Aditya Gulati, Luowen Qian, and Henry Yuen. Pseudorandom (function-like) quantum state generators: New definitions and applications. In Eike Kiltz and Vinod Vaikuntanathan, editors, *TCC 2022, Part I*, volume 13747 of *LNCS*, pages 237–265. Springer, Cham, November 2022. doi:10.1007/978-3-031-22318-1_9.

[AIK22] Scott Aaronson, DeVon Ingram, and William Kretschmer. The Acrobatics of BQP. In Shachar Lovett, editor, *37th Computational Complexity Conference (CCC 2022)*, volume 234 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 20:1–20:17, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. URL: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.CCC.2022.20, doi:10.4230/LIPIcs.CCC.2022.20.

[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.

[ALY24] Prabhanjan Ananth, Yao-Ting Lin, and Henry Yuen. Pseudorandom strings from pseudorandom quantum states. In Venkatesan Guruswami, editor, *ITCS 2024*, volume 287, pages 6:1–6:22. LIPIcs, January / February 2024. doi:10.4230/LIPIcs.ITCS.2024.6.

[AQY22] Prabhanjan Ananth, Luowen Qian, and Henry Yuen. Cryptography from pseudorandom quantum states. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part I*, volume 13507 of *LNCS*, pages 208–236. Springer, Cham, August 2022. doi:10.1007/978-3-031-15802-5_8.

[BB84] Charles Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and coin tossing. volume 560, pages 175–179, 01 1984. doi:10.1016/j.tcs.2011.08.039.

[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.

[BBHT98] Michel Boyer, Gilles Brassard, Peter Høyer, and Alain Tapp. Tight bounds on quantum searching. *Fortschritte der Physik*, 46(4-5):493–505, jun 1998. URL: https://doi.org/10.1002%2F%28sici%291521-3978%28199806%2946%3A4%2F5%3C493%3A%3Aaid-prop493%3E3.0.co%3B2-p, doi:10.1002/(sici)1521-3978(199806)46:4/5<493::aid-prop493>3.0.co;2-p.

[BBO+24] Mohammed Barhoush, Amit Behera, Lior Ozer, Louis Salvail, and Or Sattath. Signatures from pseudorandom states via ⊥-prfs, 2024. arXiv:2311.00847.

[BBSS23] Amit Behera, Zvika Brakerski, Or Sattath, and Omri Shmueli. Pseudorandomness with proof of destruction and applications. In Guy N. Rothblum and Hoeteck Wee, editors, *TCC 2023, Part IV*, volume 14372 of *LNCS*, pages 125–154. Springer, Cham, November / December 2023. doi:10.1007/978-3-031-48624-1_5.

[BCKM21] James Bartusek, Andrea Coladangelo, Dakshita Khurana, and Fermi Ma. On the round complexity of secure quantum computation. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part I*, volume 12825 of *LNCS*, pages 406–435, Virtual Event, August 2021. Springer, Cham. doi:10.1007/978-3-030-84242-0_15.

[BCN24] John Bostanci, Boyang Chen, and Barak Nehoran. Oracle separation between quantum commitments and quantum one-wayness. *arXiv preprint arXiv:2410.03358*, 2024.

[BCQ23] Zvika Brakerski, Ran Canetti, and Luowen Qian. On the computational hardness needed for quantum cryptography. In Yael Tauman Kalai, editor, *ITCS 2023*, volume 251, pages 24:1–24:21. LIPIcs, January 2023. doi:10.4230/LIPIcs.ITCS.2023.24.

[BDF+11] Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random oracles in a quantum world. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 41–69. Springer, Berlin, Heidelberg, December 2011. doi:10.1007/978-3-642-25385-0_3.

[BGH+23] Khashayar Barooti, Alex B. Grilo, Loïs Huguenin-Dumittan, Giulio Malavolta, Or Sattath, Quoc-Huy Vu, and Michael Walter. Public-key encryption with quantum keys. In Guy N. Rothblum and Hoeteck Wee, editors, *TCC 2023,*

Part IV, volume 14372 of *LNCS*, pages 198–227. Springer, Cham, November / December 2023. doi:10.1007/978-3-031-48624-1_8.

[BGV23] Samuel Bouaziz-Ermann, Alex B. Grilo, and Damien Vergnaud. Quantum security of subset cover problems. In Kai-Min Chung, editor, *ITC 2023*, volume 267 of *LIPIcs*, pages 9:1–9:17. Schloss Dagstuhl, June 2023. doi:10.4230/LIPIcs.ITC.2023.9.

[BGVV24] Samuel Bouaziz-Ermann, Alex B. Grilo, Damien Vergnaud, and Quoc-Huy Vu. Towards the impossibility of quantum public key encryption with classical keys from one-way functions. *CiC*, 1(1):32, 2024. doi:10.62056/ahvr-11zn4.

[BHH+15] Daniel J. Bernstein, Daira Hopwood, Andreas Hülsing, Tanja Lange, Ruben Niederhagen, Louiza Papachristodoulou, Michael Schneider, Peter Schwabe, and Zooko Wilcox-O'Hearn. SPHINCS: Practical stateless hash-based signatures. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part I*, volume 9056 of *LNCS*, pages 368–397. Springer, Berlin, Heidelberg, April 2015. doi:10.1007/978-3-662-46800-5_15.

[BHK+19] Gilles Brassard, Peter Høyer, Kassem Kalach, Marc Kaplan, Sophie Laplante, and Louis Salvail. Key establishment à la Merkle in a quantum world. *Journal of Cryptology*, 32(3):601–634, July 2019. doi:10.1007/s00145-019-09317-z.

[BHMV25] Samuel Bouaziz--Ermann, Minki Hhan, Garazi Muguruza, and Quoc-Huy Vu. On limits on the provable consequences of quantum pseudorandomness. 2025.

[BHT98] Gilles Brassard, Peter Høyer, and Alain Tapp. Quantum cryptanalysis of hash and claw-free functions. In Claudio L. Lucchesi and Arnaldo V. Moura, editors, *LATIN '98: Theoretical Informatics, Third Latin American Symposium, Campinas, Brazil, April, 20-24, 1998, Proceedings*, volume 1380 of *Lecture Notes in Computer Science*, pages 163–169. Springer, 1998. doi:10.1007/BFb0054319.

[BJ24] Rishabh Batra and Rahul Jain. Commitments are equivalent to statistically-verifiable one-way state generators. In *65th FOCS*, pages 1178–1192. IEEE Computer Society Press, October 2024. doi:10.1109/FOCS61266.2024.00077.

[BM09] Boaz Barak and Mohammad Mahmoody-Ghidary. Merkle puzzles are optimal - an $O(n^2)$-query attack on any key exchange from a random oracle. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 374–390. Springer, Berlin, Heidelberg, August 2009. doi:10.1007/978-3-642-03356-8_22.

[BM24] Samuel Bouaziz--Ermann and Garazi Muguruza. Quantum pseudorandomness cannot be shrunk in a black-box way. *arXiv preprint arXiv:2402.13324*, 2024. arXiv:2402.13324.

[BMM+24] Amit Behera, Giulio Malavolta, Tomoyuki Morimae, Tamer Mour, and Takashi Yamakawa. A new world in the depths of microcrypt: Separating owsgs and quantum money from qefid. *arXiv preprint arXiv:2410.03453*, 2024.

[BR93] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In Dorothy E. Denning, Raymond Pyle, Ravi Ganesan, Ravi S. Sandhu, and Victoria Ashby, editors, *ACM CCS 93*, pages 62–73. ACM Press, November 1993. doi:10.1145/168588.168596.

[BS20] Zvika Brakerski and Omri Shmueli. Scalable pseudorandom quantum states. In Daniele Micciancio and Thomas Ristenpart, editors, *CRYPTO 2020, Part II*, volume 12171 of *LNCS*, pages 417–440. Springer, Cham, August 2020. doi:10.1007/978-3-030-56880-1_15.

[CCS24] Boyang Chen, Andrea Coladangelo, and Or Sattath. The power of a single haar random state: constructing and separating quantum pseudorandomness. *arXiv preprint arXiv:2404.03295*, 2024.

[CFHL21] Kai-Min Chung, Serge Fehr, Yu-Hsuan Huang, and Tai-Ning Liao. On the compressed-oracle technique, and post-quantum security of proofs of sequential work. In Anne Canteaut and François-Xavier Standaert, editors, *EUROCRYPT 2021, Part II*, volume 12697 of *LNCS*, pages 598–629. Springer, Cham, October 2021. doi:10.1007/978-3-030-77886-6_21.

[CGG+23] Bruno Cavalar, Eli Goldin, Matthew Gray, Peter Hall, Yanyi Liu, and Angelos Pelecanos. On the computational hardness of quantum one-wayness. *arXiv preprint arXiv:2312.08363*, 2023.

[CGG24] Kai-Min Chung, Eli Goldin, and Matthew Gray. On central primitives for quantum cryptography with classical communication. In Leonid Reyzin and Douglas Stebila, editors, *CRYPTO 2024, Part VII*, volume 14926 of *LNCS*, pages 215–248. Springer, Cham, August 2024. doi:10.1007/978-3-031-68394-7_8.

[CGGH25] Bruno Pasqualotto Cavalar, Eli Goldin, Matthew Gray, and Peter Hall. A meta-complexity characterization of quantum cryptography. In Serge Fehr and Pierre-Alain Fouque, editors, *EUROCRYPT 2025, Part VII*, volume 15607 of *LNCS*, pages 82–107. Springer, Cham, May 2025. doi:10.1007/978-3-031-91098-2_4.

[CLM23] Kai-Min Chung, Yao-Ting Lin, and Mohammad Mahmoody. Black-box separations for non-interactive classical commitments in a quantum world. In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT 2023, Part I*, volume 14004 of *LNCS*, pages 144–172. Springer, Cham, April 2023. doi:10.1007/978-3-031-30545-0_6.

[CM24] Andrea Coladangelo and Saachi Mutreja. On black-box separations of quantum digital signatures from pseudorandom states. In Elette Boyle and Mohammad Mahmoody, editors, *TCC 2024, Part III*, volume 15366 of *LNCS*, pages 289–317. Springer, Cham, December 2024. doi:10.1007/978-3-031-78020-2_10.

[Col23] Andrea Coladangelo. Quantum trapdoor functions from classical one-way functions. *CoRR*, abs/2302.12821, 2023. arXiv:2302.12821, doi:10.48550/arXiv.2302.12821.

[GLSV21] Alex B. Grilo, Huijia Lin, Fang Song, and Vinod Vaikuntanathan. Oblivious transfer is in MiniQCrypt. In Anne Canteaut and François-Xavier Standaert, editors, *EUROCRYPT 2021, Part II*, volume 12697 of *LNCS*, pages 531–561. Springer, Cham, October 2021. doi:10.1007/978-3-030-77886-6_18.

[GMMY24] Eli Goldin, Tomoyuki Morimae, Saachi Mutreja, and Takashi Yamakawa. Countcrypt: Quantum cryptography between qcma and pp, 2024. URL: https://arxiv.org/abs/2410.14792, arXiv:2410.14792.

[Gol90] Oded Goldreich. A note on computational indistinguishability. *Information Processing Letters*, 34(6):277–281, 1990. URL: https://www.sciencedirect.com/science/article/pii/002001909090010U, doi:https://doi.org/10.1016/0020-0190(90)90010-U.

[GRM78] S. L. Graham, R. L. Rivest, and Ralph C. Merkle. Secure communications over insecure channels, 1978.

[Gro96] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *28th ACM STOC*, pages 212–219. ACM Press, May 1996. doi:10.1145/237814.237866.

[Gro97] Lov K. Grover. Quantum mechanics helps in searching for a needle in a haystack. *Physical Review Letters*, 79(2):325–328, Jul 1997. URL: http://dx.doi.org/10.1103/PhysRevLett.79.325, doi:10.1103/physrevlett.79.325.

[GZ25] Eli Goldin and Mark Zhandry. Translating between the common Haar random state model and the unitary model. Cryptology ePrint Archive, Report 2025/489, 2025. URL: https://eprint.iacr.org/2025/489.

[HBD+22] Andreas Hülsing, Daniel J. Bernstein, Christoph Dobraunig, Maria Eichlseder, Scott Fluhrer, Stefan-Lukas Gazdag, Panos Kampanakis, Stefan Kölbl, Tanja Lange, Martin M. Lauridsen, Florian Mendel, Ruben Niederhagen, Christian Rechberger, Joost Rijneveld, Peter Schwabe, Jean-Philippe Aumasson, Bas Westerbaan, and Ward Beullens. SPHINCS$^+$. Technical report, National Institute of Standards and Technology, 2022. available at https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022.

[HKOT23] Jeongwan Haah, Robin Kothari, Ryan O'Donnell, and Ewin Tang. Query-optimal estimation of unitary channels in diamond distance. In *64th FOCS*, pages 363–390. IEEE Computer Society Press, November 2023. doi:10.1109/FOCS57990.2023.00028.

[HLM17] Aram W Harrow, Cedric Yen-Yu Lin, and Ashley Montanaro. Sequential measurements, disturbance and property testing. In *Proceedings of the Twenty-*

Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, pages 1598–1611. SIAM, 2017.

[HM10] Aram Wettroth Harrow and Ashley Montanaro. An efficient test for product states with applications to quantum Merlin-Arthur games. In *51st FOCS*, pages 633–642. IEEE Computer Society Press, October 2010. doi:10.1109/FOCS.2010.66.

[HM24] Taiga Hiroka and Tomoyuki Morimae. Quantum cryptography and meta-complexity, 2024. URL: https://arxiv.org/abs/2410.01369, arXiv:2410.01369.

[HMY24] Minki Hhan, Tomoyuki Morimae, and Takashi Yamakawa. A note on output length of one-way state generators and efis, 2024. arXiv:2312.16025.

[HR07] Iftach Haitner and Omer Reingold. Statistically-hiding commitment from any one-way function. In David S. Johnson and Uriel Feige, editors, *39th ACM STOC*, pages 1–10. ACM Press, June 2007. doi:10.1145/1250790.1250792.

[Imp95] R. Impagliazzo. A personal view of average-case complexity. In *Proceedings of Structure in Complexity Theory. Tenth Annual IEEE Conference*, pages 134–147, 1995. doi:10.1109/SCT.1995.514853.

[IR89] Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *21st ACM STOC*, pages 44–61. ACM Press, May 1989. doi:10.1145/73007.73012.

[JLS18] Zhengfeng Ji, Yi-Kai Liu, and Fang Song. Pseudorandom quantum states. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part III*, volume 10993 of *LNCS*, pages 126–152. Springer, Cham, August 2018. doi:10.1007/978-3-319-96878-0_5.

[KMNY23] Fuyuki Kitagawa, Tomoyuki Morimae, Ryo Nishimaki, and Takashi Yamakawa. Quantum public-key encryption with tamper-resilient public keys from one-way functions. Cryptology ePrint Archive, Paper 2023/490, 2023. https://eprint.iacr.org/2023/490. URL: https://eprint.iacr.org/2023/490.

[KMNY24] Fuyuki Kitagawa, Tomoyuki Morimae, Ryo Nishimaki, and Takashi Yamakawa. Quantum public-key encryption with tamper-resilient public keys from one-way functions. In Leonid Reyzin and Douglas Stebila, editors, *CRYPTO 2024, Part VII*, volume 14926 of *LNCS*, pages 93–125. Springer, Cham, August 2024. doi:10.1007/978-3-031-68394-7_4.

[KQST23] William Kretschmer, Luowen Qian, Makrand Sinha, and Avishay Tal. Quantum cryptography in algorithmica. In Barna Saha and Rocco A. Servedio, editors, *55th ACM STOC*, pages 1589–1602. ACM Press, June 2023. doi:10.1145/3564246.3585225.

[KQT24] William Kretschmer, Luowen Qian, and Avishay Tal. Quantum-computable one-way functions without one-way functions. *arXiv preprint arXiv:2411.02554*, 2024.

[Kre21] William Kretschmer. Quantum Pseudorandomness and Classical Complexity. In Min-Hsiu Hsieh, editor, *16th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2021)*, volume 197 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 2:1–2:20, Dagstuhl, Germany, 2021. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. URL: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.TQC.2021.2, doi:10.4230/LIPIcs.TQC.2021.2.

[KT24a] Dakshita Khurana and Kabir Tomer. Commitments from quantum one-wayness. In Bojan Mohar, Igor Shinkar, and Ryan O'Donnell, editors, *56th ACM STOC*, pages 968–978. ACM Press, June 2024. doi:10.1145/3618260.3649654.

[KT24b] Dakshita Khurana and Kabir Tomer. Founding quantum cryptography on quantum advantage, or, towards cryptography from #p-hardness, 2024. URL: https://arxiv.org/abs/2409.15248, arXiv:2409.15248.

[Lam79] L. Lamport. Constructing digital signatures from a one-way function. Technical Report SRI-CSL-98, SRI International Computer Science Laboratory, october 1979.

[LDK+22] Vadim Lyubashevsky, Léo Ducas, Eike Kiltz, Tancrède Lepoint, Peter Schwabe, Gregor Seiler, Damien Stehlé, and Shi Bai. CRYSTALS-DILITHIUM. Technical report, National Institute of Standards and Technology, 2022. available at https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022.

[LLLL24] Longcheng Li, Qian Li, Xingjian Li, and Qipeng Liu. How (not) to build quantum PKE in minicrypt. In Leonid Reyzin and Douglas Stebila, editors, *CRYPTO 2024, Part VII*, volume 14926 of *LNCS*, pages 152–183. Springer, Cham, August 2024. doi:10.1007/978-3-031-68394-7_6.

[LLLL25a] Longcheng Li, Qian Li, Xingjian Li, and Qipeng Liu. Cryptomania v.s. minicrypt in a quantum world. Cryptology ePrint Archive, Paper 2025/639, 2025. URL: https://eprint.iacr.org/2025/639.

[LLLL25b] Longcheng Li, Qian Li, Xingjian Li, and Qipeng Liu. Toward the impossibility of perfect complete quantum PKE from OWFs. In *ITCS 2025*, pages 71:1–71:16. LIPIcs, January 2025. doi:10.4230/LIPIcs.ITCS.2025.71.

[Lub78] Elihu Lubkin. Entropy of an n-system from its correlation with a k-reservoir. *Journal of Mathematical Physics*, 19(5):1028–1031, 05 1978. arXiv:https://pubs.aip.org/aip/jmp/article-pdf/19/5/1028/19149012/1028_1_online.pdf, doi:10.1063/1.523763.

[LV24] Romi Levy and Thomas Vidick. Prs length expansion, 2024. URL: https://arxiv.org/abs/2411.03215, arXiv:2411.03215.

[LZ19] Qipeng Liu and Mark Zhandry. On finding quantum multi-collisions. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019, Part III*, volume 11478 of *LNCS*, pages 189–218. Springer, Cham, May 2019. doi:10.1007/978-3-030-17659-4_7.

[Mec19] Elizabeth S Meckes. *The random matrix theory of the classical compact groups*, volume 218. Cambridge University Press, 2019.

[Mel24] Antonio Anna Mele. Introduction to Haar Measure Tools in Quantum Information: A Beginner's Tutorial. *Quantum*, 8:1340, May 2024. doi:10.22331/q-2024-05-08-1340.

[MFP+24] Nicolas Maring, Andreas Fyrillas, Mathias Pont, Edouard Ivanov, Petr Stepanov, Nico Margaria, William Hease, Anton Pishchagin, Aristide Lemaître, Isabelle Sagnes, Thi Huong Au, Sébastien Boissier, Eric Bertasi, Aurélien Baert, Mario Valdivia, Marie Billard, Ozan Acar, Alexandre Brieussel, Rawad Mezher, Stephen C. Wein, Alexia Salavrakos, Patrick Sinnott, Dario A. Fioretto, Pierre-Emmanuel Emeriau, Nadia Belabas, Shane Mansfield, Pascale Senellart, Jean Senellart, and Niccolo Somaschi. A versatile single-photon-based quantum computing platform. *Nature Photonics*, 18(6):603–609, Jun 2024. doi:10.1038/s41566-024-01403-4.

[MH25] Fermi Ma and Hsin-Yuan Huang. How to construct random unitaries. In *57th ACM STOC*, pages 806–809. ACM Press, June 2025. doi:10.1145/3717823.3718254.

[MW24] Giulio Malavolta and Michael Walter. Robust quantum public-key encryption with applications to quantum key distribution. In Leonid Reyzin and Douglas Stebila, editors, *CRYPTO 2024, Part VII*, volume 14926 of *LNCS*, pages 126–151. Springer, Cham, August 2024. doi:10.1007/978-3-031-68394-7_5.

[MY22a] Tomoyuki Morimae and Takashi Yamakawa. One-wayness in quantum cryptography. Cryptology ePrint Archive, Report 2022/1336, 2022. URL: https://eprint.iacr.org/2022/1336.

[MY22b] Tomoyuki Morimae and Takashi Yamakawa. Quantum commitments and signatures without one-way functions. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part I*, volume 13507 of *LNCS*, pages 269–295. Springer, Cham, August 2022. doi:10.1007/978-3-031-15802-5_10.

[MY25] Tomoyuki Morimae and Shogo Yamada. Black-box separation between pseudorandom unitaries and pseudorandom function-like states. 2025.

[MYY24] Tomoyuki Morimae, Shogo Yamada, and Takashi Yamakawa. Quantum unpredictability. In Kai-Min Chung and Yu Sasaki, editors, *ASIACRYPT 2024,*

Part IX, volume 15492 of *LNCS*, pages 3–32. Springer, Singapore, December 2024. doi:10.1007/978-981-96-0947-5_1.

[NC10] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge university press, 2010.

[NR98] Moni Naor and Omer Reingold. From unpredictability to indistinguishability: A simple construction of pseudo-random functions from MACs (extended abstract). In Hugo Krawczyk, editor, *CRYPTO'98*, volume 1462 of *LNCS*, pages 267–282. Springer, Berlin, Heidelberg, August 1998. doi:10.1007/BFb0055734.

[PFH+22] Thomas Prest, Pierre-Alain Fouque, Jeffrey Hoffstein, Paul Kirchner, Vadim Lyubashevsky, Thomas Pornin, Thomas Ricosset, Gregor Seiler, William Whyte, and Zhenfei Zhang. FALCON. Technical report, National Institute of Standards and Technology, 2022. available at https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022.

[Rom90] John Rompel. One-way functions are necessary and sufficient for secure signatures. In *22nd ACM STOC*, pages 387–394. ACM Press, May 1990. doi:10.1145/100216.100269.

[RR02] Leonid Reyzin and Natan Reyzin. Better than BiBa: Short one-time signatures with fast signing and verifying. In Lynn Margaret Batten and Jennifer Seberry, editors, *ACISP 02*, volume 2384 of *LNCS*, pages 144–153. Springer, Berlin, Heidelberg, July 2002. doi:10.1007/3-540-45450-0_11.

[RSA78] Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the Association for Computing Machinery*, 21(2):120–126, February 1978. doi:10.1145/359340.359342.

[Sho97] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, oct 1997. URL: https://doi.org/10.1137%2Fs0097539795293172, doi:10.1137/s0097539795293172.

[Wie83] Stephen Wiesner. Conjugate coding. *SIGACT News*, 15(1):78–88, 1983. doi:10.1145/1008908.1008920.

[Yan22] Jun Yan. General properties of quantum bit commitments (extended abstract). In Shweta Agrawal and Dongdai Lin, editors, *ASIACRYPT 2022, Part IV*, volume 13794 of *LNCS*, pages 628–657. Springer, Cham, December 2022. doi:10.1007/978-3-031-22972-5_22.

[YTA22] Quan Yuan, Mehdi Tibouchi, and Masayuki Abe. On subset-resilient hash function families. *DCC*, 90(3):719–758, 2022. doi:10.1007/s10623-022-01008-4.

[YTA23] Quan Yuan, Mehdi Tibouchi, and Masayuki Abe. Quantum-access security of hash-based signature schemes. In Leonie Simpson and Mir Ali Reza-

zadeh Baee, editors, *Information Security and Privacy*, pages 343–380, Cham, 2023. Springer Nature Switzerland.

[YZ21] Takashi Yamakawa and Mark Zhandry. Classical vs quantum random oracles. In Anne Canteaut and François-Xavier Standaert, editors, *EUROCRYPT 2021, Part II*, volume 12697 of *LNCS*, pages 568–597. Springer, Cham, October 2021. doi:10.1007/978-3-030-77886-6_20.

[Zha19] Mark Zhandry. How to record quantum queries, and applications to quantum indifferentiability. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 239–268. Springer, Cham, August 2019. doi:10.1007/978-3-030-26951-7_9.

Appendices

A Grover's Algorithm and Quantum Fourier Transform

A.1 Grover's Algorithm

Here we quickly recall Grover's algorithm. We start by defining the search problem.

Definition .1 (Search problem)**.** *We are given a function $F : \mathcal{X} \to \{0,1\}$. The search problem consists of finding an $x \in \mathcal{X}$ such that $F(x) = 1$, in the least amount of queries to F possible.*

Grover's algorithm solves the search problem in $O\left(\sqrt{\frac{|\mathcal{X}|}{t}}\right)$, where t is the number of x such that $F(x) = 1$. The result is stated as follows:

Theorem A.1 ([Gro96][BBHT98])**.** *Let $F : \mathcal{X} \to \{0,1\}$ be a function, $t = |\{x|F(x) = 1\}|$, and $N = |\mathcal{X}|$. Then, Grover's algorithm finds an x such that $F(x) = 1$ with constant probability with $O\left(\sqrt{\frac{N}{t}}\right)$ queries to F. Moreover, this algorithm is optimal.*

Remark .1. *When constructing quantum algorithms in the* Quantum Random Oracle Model*, we are given a black box access to a function $H : \mathcal{X} \to \mathcal{Y}$. To use Grover's algorithm in this model, we need to construct the function $F : \mathcal{X} \to \{0,1\}$ from the function H. Then, to count the number of queries to H, it is sufficient to compute the number of queries to F.*

A.2 The Quantum Fourier Transform

Let $Y = \{0,1\}^n$, for some $n \in \mathbb{N}$. We recall that the computational basis is $\{|y\rangle\}_{y \in Y}$. The **Quantum Fourier Transform** is a unitary that, given an input state $|\phi\rangle = \sum_{k=0}^{2^n-1} x_k |k\rangle$, outputs $\sum_{k=0}^{2^n} y_k |k\rangle$ where the y_k's are computed with the following formula:

$$y_k = \frac{1}{2^{n/2}} \sum_{\ell=0}^{2^n-1} x_\ell \omega_N^{k\ell}$$

where $\omega_N = e^{2\pi i/2^n}$ thus ω_N^ℓ is a 2^n-th root of unity.

This unitary can be efficiently implemented, and we write it QFT.

Applying the QFT to the computational basis yields the *Fourier basis* $\{|\hat{y}\rangle\}_{y\in Y}$.

B Technical Proofs

B.1 Proof of Lemma 2.5

As previously mentioned, the proof closely follows the proof of Corollary 11 from [LZ19].

We write $P'_{\ell-col-h_1}$ the set of databases that contain *at least* ℓ *distinct* collisions on h_1.

Let $i \in \mathbb{N}$. Let $|\phi_i\rangle$ be the state just before the i^{th} query to $H = (h_1, h_2)$, namely

$$|\phi_i\rangle = \sum_{x,\hat{y},z,D} \alpha_{x,\hat{y},z,D} |x, \hat{y}, z\rangle \otimes |D\rangle ,$$

where x is the query register, y is the answer register, z is the work register and D is the database register. Let $|\psi_i\rangle$ be the state right after the i^{th} query to H, namely

$$|\psi_i\rangle = \sum_{\substack{x,\hat{y},z,D \\ D(x)=\perp}} \frac{1}{\sqrt{N^2}} \sum_{y'} \omega_N^{yy'} \alpha_{x,\hat{y},z,D} |x, \hat{y}, z\rangle \otimes |D \cup (x, y')\rangle + \mathsf{cO} \sum_{\substack{x,\hat{y},z,D \\ D(x)\neq\perp}} \alpha_{x,\hat{y},z,D} |x, \hat{y}, z\rangle \otimes |D\rangle .$$

From Lemma 2.4, we have that:

$$\left\| P'_{\ell-col-h_1} |\psi_i\rangle \right\| \leq \left\| P'_{\ell-col-h_1} |\phi_i\rangle \right\| + \left\| P'_{\ell-col-h_1} \mathsf{cO} (I - P'_{\ell-col-h_1}) |\phi_i\rangle \right\| . \tag{1}$$

Writing $f_{i,\ell}^{col} = \left\| P'_{\ell-col-h_1} |\psi_i\rangle \right\|$ and similarly to the proof of Lemma 3.2, using Lemma 2.2 and Lemma 2.3 we obtain the following recursive inequality:

$$
\begin{aligned}
f_{i,\ell}^{col} &\leq f_{i-1,\ell}^{col} + 4\frac{\sqrt{i-1}}{\sqrt{N}} f_{i-1,\ell-1}^{col} \\
&\leq \sum_{j=0}^{i-1} 4\frac{\sqrt{j}}{\sqrt{N}} f_{j,\ell-1}^{col} \\
&\leq \sum_{j_1=0}^{i-1} 4\frac{\sqrt{j_1}}{\sqrt{N}} \sum_{j_2=0}^{j_1-1} 4\frac{\sqrt{j_2}}{\sqrt{N}} f_{j_2,\ell-2}^{col} \\
&\vdots \\
&\leq \sum_{0\leq j_\ell < j_{\ell-1} < \cdots < j_1 < i} \prod_{k=1}^{\ell} 4\frac{\sqrt{j_k}}{\sqrt{N}} \\
&\leq \frac{1}{\ell!} \sum_{0\leq j_\ell, j_{\ell-1}, \ldots, j_1 < i} \prod_{k=1}^{\ell} 4\frac{\sqrt{j_k}}{\sqrt{N}} \\
&= \frac{1}{\ell!} \left(\sum_{0<j<i} 4\frac{\sqrt{i-1}}{\sqrt{N}} \right)^j \\
&\leq \left(\frac{4e \cdot i^{3/2}}{j\sqrt{N}} \right)^j,
\end{aligned}
$$

where the computation follows from the proof of Lemma 11 in [LZ19].

B.2 Proof of Equation 3.5

Writing $D_{y'} = D \cup (x, y')$,

$$
\begin{aligned}
&\left| P_2 \sum_{y'} \frac{1}{\sqrt{N^2}} \sum_{\substack{x,\hat{y},z \\ D:\neg P_2 \\ D(x)=\perp}} \omega_N^{yy'} \alpha_{x,\hat{y},z,D} \left| x, \hat{y}, z, D_{y'} \right\rangle \right| \\
\leq &\left| \sum_{\ell \geq 0} \frac{\ell}{N} \sum_{b \in \{1,2\}} \sum_{\substack{x,\hat{y},z \\ D:\neg P_2 \\ \text{exactly } \ell \\ \text{collisions on } h_b}} \sum_{y'} \frac{1}{\sqrt{N^2}} \omega_N^{yy'} \alpha_{x,\hat{y},z,D} \left| x, \hat{y}, z, D_{y'} \right\rangle \right. \\
&\left. + \frac{(i-1)^2}{N^2} \sum_{\substack{x,\hat{y},z \\ D:\neg P_2}} \sum_{y'} \frac{1}{\sqrt{N^2}} \omega_N^{yy'} \alpha_{x,\hat{y},z,D} \left| x, \hat{y}, z, D_{y'} \right\rangle \right| \\
\leq &\left| 2 \cdot \sum_{\ell \geq 0} \frac{\ell}{N} \sum_{\substack{x,\hat{y},z \\ D:\neg P_2 \\ \text{exactly } \ell \\ \text{collisions on } h_1}} \sum_{y'} \frac{1}{\sqrt{N^2}} \omega_N^{yy'} \alpha_{x,\hat{y},z,D} \left| x, \hat{y}, z, D_{y'} \right\rangle \right| \\
&+ \left| \frac{(i-1)^2}{N^2} \sum_{\substack{x,\hat{y},z \\ D:\neg P_2}} \sum_{y'} \frac{1}{\sqrt{N^2}} \omega_N^{yy'} \alpha_{x,\hat{y},z,D} \left| x, \hat{y}, z, D_{y'} \right\rangle \right| \\
\leq &\left(2 \cdot \sum_{\ell \geq 0} \frac{\ell}{N} \sum_{\substack{x,\hat{y},z \\ D:\neg P_2 \\ \text{exactly } \ell \\ \text{collisions on } h_1}} \left| \alpha_{x,\hat{y},z,D} \right|^2 \right)^{1/2} + \left(\frac{(i-1)^2}{N^2} \sum_{\substack{x,\hat{y},z \\ D:\neg P_2}} \left| \alpha_{x,\hat{y},z,D} \right|^2 \right)^{1/2} \\
\leq &\left(2 \cdot \sum_{\ell \geq 0} \frac{\ell}{N} \left| P_{\ell - col - h_1} \left| \phi_i \right\rangle \right|^2 \right)^{1/2} + \frac{(i-1)}{N},
\end{aligned}
$$

where in the second inequality, we used the symmetry of finding collisions on h_1 and collisions on h_2, and used the definition of $\left| P_{\ell - col - h_1} \left| \phi_i \right\rangle \right|^2$ in the last inequality.

B.3 Proof of Lemma 3.6

Proof. We have that

$$\begin{aligned}
A_i &\leq \sum_{\ell:\mu(\ell)=8e\frac{\ell^{3/2}}{\sqrt{N}}} \sqrt{2}\cdot\frac{\sqrt{8e\ell^{3/2}}}{N^{3/4}} + \sum_{\ell:\mu_3(\ell)=10N^{1/8}} \sqrt{2}\cdot\frac{\sqrt{10N^{1/8}}}{N^{1/2}} + \sum_{\ell=0}^{i-1} 4\cdot\frac{\ell-1}{N} \\
&\leq \sum_{\ell=1}^{i-1} \sqrt{2}\cdot\frac{\sqrt{8e\ell^{3/2}}}{N^{3/4}} + \sum_{\ell:\mu_3(\ell)=10N^{1/8}} \sqrt{2}\cdot\frac{\sqrt{10N^{1/8}}}{N^{1/2}} + \sum_{\ell=0}^{i-1} 4\cdot\frac{\ell-1}{N} \\
&\leq 4\sqrt{e}\frac{i^{7/4}}{N^{3/4}} + \sqrt{2}\cdot\left(\frac{10}{8e}\right)^{2/3}\cdot N^{5/12}\cdot\frac{\sqrt{10N^{1/8}}}{N^{1/2}} + 4\cdot\frac{i^2}{N} \\
&\leq 4\sqrt{e}\cdot\frac{i^{7/4}}{N^{3/4}} + 4\cdot\frac{i^2}{N} + O\left(N^{-1/48}\right),
\end{aligned}$$

where the third inequality comes from counting the number of ℓ such that $\mu_3(\ell) = 10N^{1/8}$, which is equal to the number of ℓ such that $8e\frac{\ell^{3/2}}{\sqrt{N}} \leq 10N^{1/8}$. □

B.4 Proof of Equation 3.13

Here, we give a proof of Equation (3.13). Starting from Equation (3.12), we have that:

$$\begin{aligned}
g_{i,k} &\leq g_{i-1,k} + \sqrt{2}\left(\sqrt{\frac{\mu_3(i-1)}{N}} + \sqrt{8}\frac{i-1}{N}\right) g_{i-1,k-1} + \sqrt{2}\cdot f^{col}_{i-1,\mu_3(i-1)} \\
&\vdots \\
&\leq \sqrt{2}\sum_{\ell=0}^{i-1}\left(\left(\sqrt{\frac{\mu_3(\ell)}{N}} + \sqrt{8}\frac{\ell}{N}\right) g_{\ell,k-1} + f^{col}_{\ell,\mu_3(\ell)}\right) \\
&\leq \sqrt{2}\sum_{\ell=0}^{i-1}\left(\left(\sqrt{\frac{\mu_3(\ell)}{N}} + \sqrt{8}\frac{\ell}{N}\right) g_{\ell,k-1} + \left(\frac{1}{2}\right)^{10N^{1/8}}\right) \\
&\leq \sum_{\ell=0}^{i-1}\sqrt{2}\left(\sqrt{\frac{\mu_3(\ell)}{N}} + \sqrt{8}\frac{\ell}{N}\right) g_{\ell,k-1} + \sqrt{2}\cdot 2^{-10N^{1/8}}\cdot N^{1/2} \\
&\leq \sum_{\ell=0}^{i-1}\sqrt{2}\left(\sqrt{\frac{\mu_3(\ell)}{N}} + \sqrt{8}\frac{\ell}{N}\right) g_{\ell,k-1} + \sqrt{2}\cdot 2^{-9.5N^{1/8}},
\end{aligned}$$

where the second inequality comes from the recursion on the first term $g_{i-1,k}$, and using the fact that $g_{0,k} = 0$. For the third inequality, we used Lemma 2.5 and the definition of μ_3.

Expanding recursively inside the sum, we have:

$$
\begin{aligned}
g_{i,k} &\leq \sum_{\ell=0}^{i-1} \sqrt{2}\left(\sqrt{\frac{\mu_3(\ell)}{N}} + \sqrt{8}\frac{\ell}{N}\right) g_{\ell,k-1} + \sqrt{2}\cdot 2^{-9.5N^{1/8}} \\
&\leq \sum_{\ell_1=0}^{i-1} \sqrt{2}\left(\sqrt{\frac{\mu_3(\ell_1)}{N}} + \sqrt{8}\frac{\ell_1}{N}\right)\left(\sum_{\ell_2=0}^{\ell_1} \sqrt{2}\left(\sqrt{\frac{\mu_3(\ell_2)}{N}} + \sqrt{8}\frac{\ell_2}{N}\right) g_{\ell_2,k-2}\right. \\
&\quad \left. + \sqrt{2}\cdot 2^{-9.5N^{1/8}}\right) + \sqrt{2}\cdot 2^{-9.5N^{1/8}} \\
&\vdots \\
&\leq \sum_{0\leq \ell_k < \ell_{k-1} < \cdots < \ell_1 < i} \prod_{j=1}^{k} \sqrt{2}\left(\sqrt{\frac{\mu_3(\ell_j)}{N}} + \sqrt{8}\frac{\ell_j}{N}\right) \\
&\quad + \sqrt{2}\cdot 2^{-9.5N^{1/8}} \sum_{t=0}^{k-1} \sum_{0\leq \ell_t < \ell_{t-1} < \cdots < \ell_1 < i} \prod_{j=1}^{t} \sqrt{2}\left(\sqrt{\frac{\mu_3(\ell_j)}{N}} + \sqrt{8}\frac{\ell_j}{N}\right) \\
&\leq \frac{A_i^k}{k!} + \sqrt{2}\cdot 2^{9.5N^{1/8}} \sum_{t=0}^{k-1} \frac{A_i^t}{t!} \\
&\leq \frac{A_i^k}{k!} + \sqrt{2}\cdot e^{A_i} 2^{9.5N^{1/8}},
\end{aligned}
$$

where the third inequality comes from expanding recursively all of the terms $g_{\ell_t,k-t}$, and using the fact that $g_{\ell,0} = 1$. The fourth inequality comes from the fact that:

$$
\begin{aligned}
&\sum_{0\leq \ell_k < \ell_{k-1} < \cdots < \ell_1 < i} \prod_{j=1}^{k} \sqrt{2}\left(\sqrt{\frac{\mu_3(\ell_j)}{N}} + \sqrt{8}\frac{\ell_j}{N}\right) \\
&\leq \frac{1}{k!} \sum_{0\leq \ell_k, \ell_{k-1}, \ldots, \ell_1 < i} \prod_{j=1}^{k} \sqrt{2}\left(\sqrt{\frac{\mu_3(\ell_j)}{N}} + \sqrt{8}\frac{\ell_j}{N}\right) \\
&= \frac{1}{k!} \prod_{j=1}^{k} \sum_{0\leq \ell_j < i} \sqrt{2}\left(\sqrt{\frac{\mu_3(\ell_j)}{N}} + \sqrt{8}\frac{\ell_j}{N}\right) \\
&= \frac{1}{k!} \prod_{j=1}^{k} A_i \\
&= \frac{A_i^k}{k!}.
\end{aligned}
$$

B.5 Proof of Lemma 3.11

Proof. We have that:

$$A_{i,s} = \sum_{\ell=0}^{i-1} \left(\sqrt{(s-1)\cdot \frac{\mu_s(\ell)}{N}} + 4\left(\frac{\ell}{N}\right)^{s/2} + \left(\sum_{r=2}^{s} \frac{\ell}{N^r}\right)^{1/2} \right)$$
$$= \sqrt{s-1}\sum_{\ell=0}^{i-1} \sqrt{\frac{\mu_s(\ell)}{N}} + 4\sum_{\ell=0}^{i-1}\left(\frac{\ell}{N}\right)^{s/2} + \sum_{\ell=0}^{i-1}\left(\sum_{r=2}^{s}\frac{\ell}{N^r}\right)^{1/2}. \tag{2}$$

Notice that

$$\sum_{\ell=0}^{i-1} \sqrt{\frac{\mu_s(\ell)}{N}}$$
$$= \sum_{\ell:\mu_s(\ell)=40\cdot s^2\cdot \Pi_{s-1}\cdot N^{1/2^s}} \sqrt{\frac{40\cdot s^2\cdot \Pi_{s-1}\cdot N^{1/2^s}}{N}} \tag{3}$$
$$+ \sum_{\ell:\mu_s(\ell)>40\cdot s^2\cdot \Pi_{s-1}\cdot N^{1/2^s}} \sqrt{\frac{\mu_s(\ell)}{N}}$$
$$\leq \sum_{\ell:\mu_s(\ell)=40\cdot s^2\cdot \Pi_{s-1}\cdot N^{1/2^s}} \sqrt{\frac{40\cdot s^2\cdot \Pi_{s-1}\cdot N^{1/2^s}}{N}} \tag{4}$$
$$+ \sum_{\ell=0}^{i-1} (8e)^{\frac{2^{s-2}-1}{2^{s-2}}} \frac{\ell^{(2^{s-1}-1)/2^{s-1}}}{N^{(2^{s-2}-1)/2^{s-1}}}\cdot N^{-1/2}\cdot \sqrt{\Pi_{s-1}},$$

where we replaced $\mu_s(\ell)$ by its value, and the inequality comes from the fact that there cannot be more than i values such that $\mu_s(\ell) > 40\cdot s^2\cdot \Pi_{s-1}\cdot N^{1/2^s}$. The second summation is at most $(8e)^{\frac{2^{s-2}-1}{2^{s-2}}} \frac{i^{(2^s-1)/2^{s-1}}}{N^{(2^{s-1}-1)/2^{s-1}}}\cdot \sqrt{\Pi_{s-1}}$.

For the first summation of Equation (4), we need to count the values of ℓ such that $\mu_s(l) = 40s^2\cdot \Pi_{s-1}\cdot N^{1/2^s}$. By using the definition of $\mu_s(\ell)$, this quantity corresponds to the number of ℓ that satisfies:

$$\Pi_{s-1}\cdot (8e)^{\frac{2^{s-2}-1}{2^{s-3}}} \frac{\ell^{(2^{s-1}-1)/2^{s-2}}}{N^{(2^{s-2}-1)/2^{s-2}}} \leq 40\cdot s^2\cdot \Pi_{s-1}\cdot N^{1/2^s}$$
$$\Leftrightarrow \ell \leq \left(\frac{40}{(8e)^{\frac{2^{s-2}-1}{2^{s-3}}}}\right)^{2^{s-2}/(2^{s-1}-1)} \cdot N^{\left(\frac{1}{2^s}+\frac{2^{s-2}-1}{2^{s-2}}\right)\frac{2^{s-2}}{2^{s-1}-1}} \cdot s^{\frac{2^s}{2^{s-1}-1}}$$
$$\Leftrightarrow \ell \leq O\left(s^{\frac{2^s}{2^{s-1}-1}} \cdot N^{\left(\frac{1}{2^s}+\frac{2^{s-2}-1}{2^{s-2}}\right)\frac{2^{s-2}}{2^{s-1}-1}}\right).$$

Thus the first summation of Equation (4) is upper-bounded by:

$$\sum_{\ell:\mu_s(\ell)=10\cdot\Pi_{s-1}\cdot N^{1/2^s}} \sqrt{\frac{10\cdot s^2\cdot\Pi_{s-1}\cdot N^{1/2^s}}{N}} =$$
$$\sqrt{\frac{10\cdot s^2\cdot\Pi_{s-1}\cdot N^{1/2^s}}{N}}\cdot O\left(s^{\frac{2^s}{2^{s-1}-1}}\cdot N^{\left(\frac{1}{2^s}+\frac{2^{s-2}-1}{2^{s-2}}\right)\frac{2^{s-2}}{2^{s-1}-1}}\right)$$
$$\leq O\left(N^{-\frac{1}{2}+\frac{1}{2^{s+1}}+\frac{2^{s-3}}{4(2^{s-1}-1)}}\cdot s^4\cdot\sqrt{\Pi_{s-1}}\right)$$
$$\leq O\left(N^{\frac{-2^{2s-1}+2^s+2^{s-1}-1+2^{2s-4}}{2(2^s-2)}}\cdot s^4\cdot\sqrt{\Pi_{s-1}}\right)$$
$$\leq O\left(N^{-1/(2^s(2^s-2))}\cdot s^4\cdot\sqrt{\Pi_{s-1}}\right)=O\left(N^{-1/(2^s(2^s-2))}\cdot s^4\cdot\Pi_s\right),$$

where for the first inequality we use that $\frac{2^s}{2^{s-1}-1}+1\leq 4$ for all $s\geq 3$.
Therefore, we have that:

$$\sum_{\ell=0}^{i-1}\sqrt{\frac{\mu_s(\ell)}{N}}\leq(2e)^{\frac{2^{s-2}-1}{2^{s-2}}}\frac{i^{(2^s-1)/2^{s-1}}}{N^{(2^{s-1}-1)/2^{s-1}}}\sqrt{\Pi_{s-1}}+O\left(N^{-1/(2^s(2^s-2))}\cdot s^4\cdot\Pi_s\right). \tag{5}$$

For the second term of Equation (2), we have:

$$4\sum_{\ell=0}^{i-1}\left(\frac{\ell}{N}\right)^{s/2}\leq 4\sum_{\ell=0}^{i-1}\left(\frac{\ell}{N}\right)$$
$$\leq 4\sum_{\ell=0}^{i-1}\left(\frac{\ell}{N}\right)^{(2^{s-1}-1)/2^{s-1}}, \tag{6}$$

where we use that $s\geq 3$ and $1\geq(2^{s-1}-1)/2^{s-1}$. And for the third term,

$$\sum_{\ell=0}^{i-1}\left(\sum_{r=2}^{s}\frac{\ell}{N^r}\right)^{1/2}\leq\sum_{\ell=0}^{i-1}\left((s-1)\frac{\ell}{N^2}\right)^{1/2}$$
$$\leq\sum_{\ell=0}^{i-1}\left(\sqrt{s-1}\frac{\ell}{N}\right)$$
$$\leq\sum_{\ell=0}^{i-1}\left(\sqrt{s-1}\left(\frac{\ell}{N}\right)^{(2^{s-1}-1)/2^{s-1}}\right), \tag{7}$$

where we used that $r\geq 2$ for the second inequality, and that $1\geq(2^{s-1}-1)/2^{s-1}$ for the last inequality. Thus, using that $2\cdot(8e)^{\frac{2^{s-2}-1}{2^{s-2}}}\geq 6$ and combining Equation (2), Equation (5),

Equation (6) and Equation (7) yields that:

$$\begin{aligned} A_{i,s} &\leq 2 \cdot (8e)^{\frac{2^{s-2}-1}{2^{s-2}}} \frac{i^{(2^s-1)/2^{s-1}}}{N^{(2^{s-1}-1)/2^{s-1}}} \cdot \sqrt{s-1} \cdot \sqrt{\Pi_{s-1}} \\ &\quad + O\left(N^{-1/(2^s(2^s-2))} \cdot s^4 \cdot \Pi_s\right) \\ &= (8e)^{\frac{2^{s-2}-1}{2^{s-2}}} \frac{i^{(2^s-1)/2^{s-1}}}{N^{(2^{s-1}-1)/2^{s-1}}} \cdot \Pi_s + O\left(s^4 \cdot \Pi_s \cdot N^{-1/(2^s(2^s-2))}\right). \end{aligned}$$

□

B.6 Proof of Equation 3.18

We have

$$\begin{aligned} g_{i,k} &\leq \left(\sum_{\ell=0}^{i-1} B_{\ell,s} \cdot g_{\ell,k-1}\right) + s^{3/2} \cdot 2^{-9.5 \cdot 4 \cdot (s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}} \\ &\leq \left(\sum_{\ell_1=0}^{i-1} B_{\ell_1,s} \left(\sum_{\ell_2=\ell_1}^{i-1} B_{\ell_2,s} \cdot g_{\ell_2,k-1} + s^{3/2} \cdot 2^{-9.5 \cdot 4 \cdot (s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}}\right)\right) \\ &\qquad + s^{3/2} \cdot 2^{-9.5 \cdot 4 \cdot (s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}}. \end{aligned}$$

We get by induction

$$\begin{aligned} g_{i,k} \leq &\left(\sum_{\ell_1=0}^{i-1} B_{\ell_1,s} \left(\sum_{\ell_2=\ell_1}^{i-1} B_{\ell_2,s} \left(\sum_{\ell_3=\ell_2}^{i-1} B_{\ell_3,s} \cdots \right.\right.\right. \\ &\left.\left.\left. + s^{3/2} \cdot 2^{-9.5 \cdot 4 \cdot (s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}}\right) + s^{3/2} \cdot 2^{-9.5 \cdot 4 \cdot (s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}}\right)\right) \\ &\qquad + s^{3/2} \cdot 2^{-9.5 \cdot 4 \cdot (s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}}. \end{aligned}$$

We thus obtain

$$\begin{aligned} g_{i,k} \leq &\left(\sum_{0 \leq \ell_k < \ell_{k-1} < \cdots < \ell_1 < i} \prod_{j=1}^{k} B_{\ell_j,s}\right) \\ &\quad + s^{3/2} \cdot 2^{-9.5 \cdot 4 \cdot (s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}} \cdot \sum_{t=0}^{k-1} \sum_{0 \leq \ell_t < \ell_{t-1} < \cdots < \ell_1 < i} \prod_{j=1}^{t} B_{\ell_j,s}, \end{aligned}$$

and finally

$$\begin{aligned} g_{i,k} \leq & \frac{A_{i,s+1}^k}{k!} + \sum_{\ell=0}^{k-1} \frac{A_{i,s+1}^\ell}{\ell!} \cdot s^{3/2} \cdot 2^{-9.5 \cdot 4 \cdot (s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}} \\ \leq & \frac{A_{i,s+1}^k}{k!} + s^{3/2} \cdot e^{A_{i,s+1}} \cdot 2^{-9.5 \cdot 4 \cdot (s+1)^2 \cdot \Pi_s \cdot N^{1/2^{s+1}}}. \end{aligned}$$

B.7 Proof of Lemma 3.20

Proof. Similarly to the proof of Lemma 3.18, we can consider that there are $O\left(N^{k-j+1}\right)$ marked elements in the function F_1. Hence, using Theorem A.1, the second step of the algorithm is expected to make

$$O\left(t \cdot \sqrt{\frac{N^k}{N^{k-j+1} \cdot \binom{k}{j}}}\right) = O\left(\frac{t}{\sqrt{\binom{k}{j}}} \cdot N^{(j-1)/2}\right)$$

quantum queries to the oracle.

Similarly to the proof of Lemma 3.18, we can consider that there are $t \cdot N^{j-1} \cdot \frac{k!}{(j-1)!}$ marked elements in the function F_2. Hence, using Theorem A.1, the fourth step of the algorithm is expected to make

$$O\left(\sqrt{\frac{N^k}{t \cdot N^{j-1} \cdot \frac{k!}{(j-1)!}}}\right) = O\left(\frac{N^{(k-j+1)/2}}{\sqrt{t}} \cdot \sqrt{\frac{(j-1)!}{k!}}\right)$$

quantum queries to the oracle.

By picking $t = N^{(k-2j+2)/3}$ with $j \leq \frac{k+2}{2}$, the complexity of the algorithm is

$$O\left(N^{(k-2j+2)/3} \cdot N^{(j-1)/2} \cdot \left(\sqrt{\frac{1}{\binom{k}{j}}} + \sqrt{\frac{j!}{k!}}\right)\right) = O\left(\frac{N^{(2k-j+1)/6}}{\sqrt{\binom{k}{j}}}\right)$$

quantum queries to the oracle. □

B.8 Proof of Lemma 3.22

Proof. We first prove the result when k is divisible by $r+1$. The result holds for $r = 1$ (using Algorithm 3.1 and Lemma 3.18).

Fix $r > 2$, and assume the result holds for $r - 1$.

The first step of the algorithm is expected to make $O\left(t \cdot N^{k'/2r}\right)$ quantum queries to the oracle if k' is divisible by r.

Similarly to the proof of Lemma 3.18, we can consider that there are $t \cdot N^{k'}$ marked elements in the function F_1. Hence, using Theorem A.1, the third step of the algorithm is expected to make $O\left(\sqrt{\frac{N^k}{t \cdot N^{k'}}}\right)$ quantum queries to the oracle.

Picking $t = N^{(rk-rk'-k')/3r}$ gives a complexity $O\left(N^{(2rk+(1-2r)k')/6r}\right)$.

By picking $k' = \frac{r}{r+1}k$, k' is an integer since k is divisible by $r+1$. Moreover, k' is divisible by r and the complexity becomes $O\left(N^{k/(2+2r)}\right)$.

If k is not divisible by $r+1$, then there is a k' between k and $k+r+1$ such that k' is divisible by $r+1$. Then, we can use Algorithm 3.3 to find a (r,k')–SC with the same functions $h_1, \ldots, h_k$ and new random functions $h_{k+1}, \ldots, h_{k'}$. This gives us a (r,k)–SC for the functions $h_1, \ldots, h_k$, and the quantum query complexity is

$$O\left(N^{k'/(2+2r)}\right) \leq O\left(N^{(k+r+1)/(2+2r)}\right).$$

□

C Geometric Interpretation of the Conjecture

Recall the conjecture for convenience.

Conjecture 5.1. *Let $X = \mathbb{S}(2^{n_1}) \times \cdots \times \mathbb{S}(2^{n_k})$ and the corresponding product Haar measure $\sigma = \sigma_{n_1} \times \cdots \times \sigma_{n_k}$, and let S_0, S_1 be two measurable subsets of X. If $d_{tr}(S_0, S_1) \geq \Delta$ and $\sigma(S_0), \sigma(S_1) \geq \Gamma$, then $\sigma(X \setminus (S_0 \cup S_1)) = \Omega(\Delta^a \Gamma^b)$ for some constant $a, b > 0$.*

Note that the space of pure random states $\mathbb{S}(N)$ can be understood as an N-dimensional unit hypersphere with complex coordinates with the quotient structure; we will use this idea to illustrate the diagrams.

Without loss of generality $\sigma(S_0) < 1/2$, and the most extreme case for S_1 is when $S_1^* = \{x \in X : d_{tr}(S_0, x) \geq \Delta\}$, since for any other S_1 we have that $\sigma(X \setminus (S_0 \cup S_1)) \geq \sigma(X \setminus (S_0 \cup S_1^*))$. We will show that the conjecture holds in this extreme situation for some natural scenarios.

We will also make use of the following lemma, which is proven in [AK07, Lemma 3.6].

Lemma C.1. *For any $\varepsilon \in [0, 1]$ and any n-qubit quantum state $|\phi\rangle$, it holds*

$$\Pr_{|\psi\rangle \leftarrow \sigma_n} \left[|\langle \psi | \phi \rangle|^2 \geq 1 - \varepsilon \right] = \varepsilon^{2^n - 1}.$$

We can rephrase the lemma in terms of trade-distance, so that

$$\Pr_{|\psi\rangle \leftarrow \sigma_n} \left[d_{tr}(|\psi\rangle, |\phi\rangle) \leq \varepsilon \right] = \varepsilon^{2(2^n - 1)}.$$

Single pure quantum state. We first consider the case of $k = 1$, i.e. $X = \mathbb{S}(2^n)$. Let $\Gamma \ll 1$ and $\Delta \ll 1$. Consider $S \subseteq X$ with $\sigma_n(S) \geq \Gamma$ and $T = \{|\psi\rangle \in X : d_{tr}(|\psi\rangle, S) \leq \Delta\}$.

Consider two extreme cases for $S \subset X$: when it is concentrated around a fixed state and when it is in the form of a "band".

Case 1: For some $\varepsilon > 0$, the set S is concentrated around a fixed state $|\phi\rangle$:

$$S := \{|\psi\rangle \in X : d_{tr}(|\psi\rangle, |\phi\rangle) \leq \varepsilon\}.$$

Based on lemma C.1 we can compute $\sigma_n(S) = \varepsilon^{2(2^n - 1)}$, and the measure of the associated T by

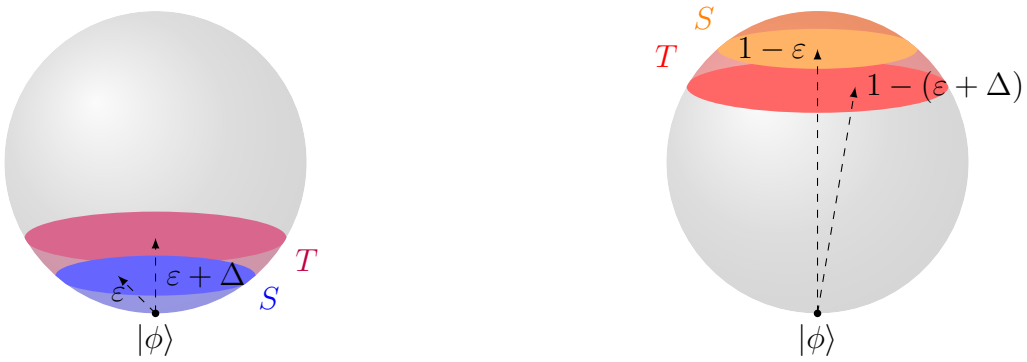
$$\sigma_n(T) = \Pr_{|\psi\rangle \leftarrow \sigma_n} \left[d_{tr}(|\psi\rangle, |\phi\rangle) \leq \varepsilon + \Delta \right] = (\varepsilon + \Delta)^{2(2^n - 1)},$$

which implies that the measure of the difference is

$$\sigma_n(T \setminus S) = \sigma_n(T) - \sigma_n(S) = (\varepsilon + \Delta)^{2(2^n - 1)} - \varepsilon^{2(2^n - 1)},$$

where we used the additivity of measures for $S \subset T$. For $\Delta \ll 1$ the above expression is the finite difference of $f(\varepsilon) = \varepsilon^{2(2^n - 1)}$, with derivative $f'(\varepsilon) = 2(2^n - 1)\varepsilon^{2(2^n - 1) - 1}$. Therefore, assuming $\sigma_n(S) \geq \Gamma$, we have

$$\sigma_n(T \setminus S) \geq 2(2^n - 1)\sigma_n(S)\varepsilon^{-1}\Delta \geq \Gamma\Delta. \tag{8}$$



(a) (S, T) from Case 1.

(b) (S, T) from Case 2.

Figure C.1: Geometric representation of the conjecture for $X = \mathbb{S}(2)$.

Case 2: For some $\varepsilon > 0$, the set S is concentrated far from the state $|\phi\rangle$:

$$S := \{|\psi\rangle \in X : \ d_{tr}(|\psi\rangle, |\phi\rangle) \geq 1 - \varepsilon\}.$$

Based on lemma C.1 we can compute $\sigma_n(S) = 1 - (1-\varepsilon)^{2(2^n-1)}$, and the measure of the associated T by

$$\begin{aligned}\sigma_n(T) &= \Pr_{|\psi\rangle \leftarrow \sigma_n}[d_{tr}(|\psi\rangle, |\phi\rangle) \geq 1 - \varepsilon - \Delta] \\ &= 1 - \Pr_{|\psi\rangle \leftarrow \sigma_n}[d_{tr}(|\psi\rangle, |\phi\rangle) \leq 1 - \varepsilon - \Delta] \\ &= 1 - (1 - \varepsilon - \Delta)^{2(2^n-1)},\end{aligned}$$

which implies that for $\Delta \ll 1$, the measure of the difference $\sigma_n(T \setminus S)$ is the finite difference of $f(\varepsilon) = 1 - (1-\varepsilon)^{2(2^n-1)}$, with derivative $f'(\varepsilon) = 2(2^n - 1)(1-\varepsilon)^{2(2^n-1)-1}$. Therefore, assuming $\sigma_n(S) \leq 1/2$, we have

$$\sigma_n(T \setminus S) \geq 2(2^n - 1)(1 - \sigma_n(S))(1-\varepsilon)^{-1}\Delta \geq \Delta.$$

Product space. We now consider the case of $k = 2$, i.e. $X = \mathbb{S}(2^{n_1}) \times \mathbb{S}(2^{n_2})$. Let $\varepsilon_1, \varepsilon_2 > 0$ and two fixed states $|\phi_1\rangle, |\phi_2\rangle \in X$. Consider $S \subset X$ as a product of two subsets $S = S_1 \times S_2$, where

$$\begin{aligned}S_1 &:= \{|\psi\rangle \in X : \ d_{tr}(|\psi\rangle, |\phi_1\rangle) \leq \varepsilon_1\}, \\ S_2 &:= \{|\psi\rangle \in X : \ d_{tr}(|\psi\rangle, |\phi_2\rangle) \leq \varepsilon_2\}.\end{aligned}$$

Let us denote by $N_i = 2^{n_i}$ for $i \in \{0, 1\}$. We also consider $T \subset X$ as before, i.e. $T = \{|\psi\rangle \in X : \ d_{tr}(|\psi\rangle, S) \leq \Delta\}$.

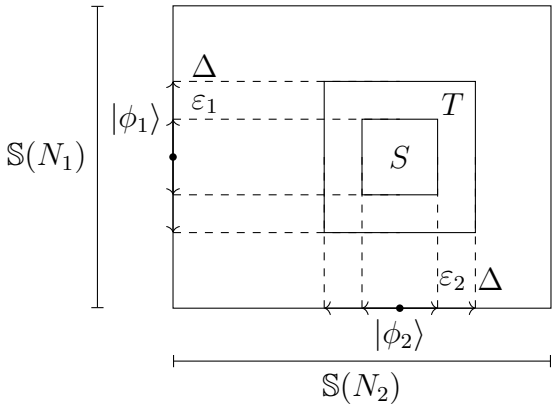


Figure C.2: Geometric representation of the conjecture for $X = \mathbb{S}(N_1) \times \mathbb{S}(N_2)$, $S = S_1 \times S_2$, and $S \subset T$.

Let $T_1 = \{|\psi\rangle \in X : \ d_{tr}(|\psi\rangle, S_1) \leq \Delta\}$ and $T_2 = \{|\psi\rangle \in X : \ d_{tr}(|\psi\rangle, S_2) \leq \Delta\}$, then $T = T_1 \times T_2$. Note that the Haar-measure is a product measure and $S \subset T_1 \times T_2$, therefore from the calculations of the previous example we obtain

$$\begin{aligned}\sigma(T \setminus S) &\geq \sigma_{n_1}(T_1) \cdot \sigma_{n_2}(T_2) - \sigma_{n_1}(S_1) \cdot \sigma_{n_2}(S_2) \\ &\geq (\varepsilon_1 + \Delta)^{2(N_1-1)}(\varepsilon_2 + \Delta)^{2(N_2-1)} - \varepsilon_1^{2(N_1-1)}\varepsilon_2^{2(N_2-1)}.\end{aligned}$$

In the case that $\varepsilon_1 = \varepsilon_2$ this can be interpreted as a finite difference of $f(\varepsilon) = \varepsilon^{2(N_1+N_2-2)}$ as before, with derivative $f'(\varepsilon) = 2(N_1 + N_2 - 2)\varepsilon^{2(N_1+N_2-2)-1}$, thus assuming $\sigma(S) = \sigma_{n_1}(S_1) \cdot \sigma_{n_2}(S_2) \geq \Gamma$ we get

$$\sigma(T \setminus S) \geq 2(N_1 + N_2 - 2)\sigma_{n_1}(S_1)\sigma_{n_2}(S_2)\varepsilon^{-1}\Delta \geq \Gamma\Delta.$$

Actually, note that this obeys the same inequality as in eq. (8); with the multiplicative overhead just increasing from $N - 1$ to $N_1 + N_2 - 2$.