A Concrete Security Treatment of Symmetric Encryption in a Quantum Computing World

Shahram Mossayebi

Thesis submitted to the University of London
for the degree of Doctor of Philosophy

School of Mathematics and Information Security
Royal Holloway, University of London

2015

Declaration

These doctoral studies were conducted under the supervision of Prof. Rüdiger Schack.

The work presented in this thesis is the result of original research carried out by myself, in collaboration with others, whilst enrolled in the Department of Mathematics as a candidate for the degree of Doctor of Philosophy. This work has not been submitted for any other degree or award in any other university or educational establishment.

Shahram Mossayebi
February, 2015

Acknowledgements

Even though I have had a life full of adventures so far, none are comparable to my Ph.D. This has been the most exciting, challenging, and interesting of all. For this, first and foremost, I would like to thank my supervisor Rüdiger Schack, who gave me the opportunity to pursue a Ph.D. He is an excellent teacher, and I loved all the discussions that we had together, especially those where he taught me quantum mechanics. Rüdiger has guided my research, and supported me throughout. Without him, I would not have reached this point. I would also like to extend this thanks to the entire Schack family (especially Dorothee Schack), for their support and encouragement.

I am indebted with Kenny Paterson, who has been a mentor and a friend ever since I started my M.Sc. at Royal Holloway. Without him I would not have met Rüdiger. Kenny always provided me with his vision and guidance whenever I needed it, which ultimately helped me to step in the right direction.

Also, I would like to thank the support staff of the School of Mathematics and Information Security. Jenny, Lisa, Liz, Jon, Tristan, and Valerie all work hard to provide every student with a great and comfortable workplace to study.

I am very grateful that I have crossed paths with many wonderful people who have helped me in many ways in pursuit of my Ph.D. at Royal Holloway. I would particularly like to thank my friends Pooya Farshim for his comments on a number of sections of this thesis, and Laurence O'Toole for proof-reading and constantly reminding me of the importance of commas. I feel privileged to have met and become friends with my colleagues at Royal Holloway, with whom I have had very interesting discussions and shared great times. Special thanks to Alessio, Dale, Eugenio, George, Gordon, James, Jean Paul, Kimberly, Michelle, Suaad, and Susan. I am also very grateful to my friend Rabih, who kept checking on me to make sure I was doing fine, and to Patricia, my amazing housemate, for being like a second sister to me and giving me support (and of course delicious Spanish foods). I would like to extend my gratitudes to Zahra, for her invaluable love and support. Thank you all for making this adventure so enjoyable.

Last but not least, I would like to thank my brilliant family. Thanks to Rana, Afshin, and little Emmitis, for their hospitality and support. I had a lot of fun staying with you guys. Thanks to my sister and brother-in-law for being very supportive and considerate. I appreciate all the good times, laughter, and of course burritos we had

together. And thanks to my parents. Their unwavering support and endless love get me through. I dedicate this to you.

Abstract

Even though it is not clear when (or if) quantum computers will be built, the theoretical existence of quantum computing has potentially far-reaching consequences for the future of cryptography. This thesis aims to provide an in-depth analysis of the security of existing (non-quantum) symmetric encryption schemes against an attacker with quantum capabilities.

Here, formal security models will be developed and justified in the provable security framework. Our results add to existing efforts in post-quantum cryptography by providing security proofs for cryptographic schemes within the concrete security paradigm. In practice, this is more relevant than the asymptotic security paradigm that is usually assumed in post-quantum cryptography.

We begin by exploring how existing classical confidentiality notions translate into a security model, where a quantum adversary is only allowed to make classical queries. Then we give a formal analysis of the security of encryption schemes, such as Counter mode, in this model.

Next we turn our attention to another natural and conservative security model where a quantum adversary is permitted to make quantum queries. To demonstrate the quantum adversary's power in this model, we show how it can break the security of block ciphers such as the Even-Mansour scheme. We further discuss the security of existing symmetric encryptions by defining security notions for confidentiality in this model. Specifically, we give a formal definition of the security of symmetric encryption schemes under both a quantum superposition chosen plaintext attack and a quantum superposition chosen ciphertext attack. Attention is also given to semantic security in this model.

Contents

Chapter 1

Introduction

Contents

This chapter gives an overview of the thesis, and presents its overall structure. We provide the motivation for our research and describe the contributions of our work. We also present the overall structure of the chapters to follow.

1.1 Motivation

Cryptography was once used only by authorities for confidential communication, but now it is part of our everyday lives. Any call we make using our mobile phones, most of web browsing we do, any payment we make using our electronic cards, etc., are all use some sort of cryptographic protocols to provide us with some sort of security.

Cryptographic protocols are built upon a deep theoretical background from both mathematics and computer science. In general, their security is rigorously analysed in a formal mathematical framework where computational power of a classical adversary (who is in possession of a classical computer) is considered. This approach is called *provable security*, and was proposed by Goldwasser and Micali [64] primarily to formalise the security of asymmetric schemes. Security proofs via this approach only hold within specific security definitions and with regard to a number of security parameters about the assessed scheme and assumptions about the adversary's capabilities. Therefore they do not rule out all classes of attack, and there may exist a practical attack against a scheme that is provably secure. To address this shortcoming, Bellare and Rogaway [15, 14, 13] introduced *practice-oriented provable* *security.* This applies the provable security approach to practical cryptographic schemes, and formulates its results in a concrete manner that is more meaningful to practice. This approach is also known as *concrete security.* Symmetric encryptions are fundamental cryptography primitives, and before the development of concrete security, it was not possible to analyse them in the provable security framework. This is because symmetric encryptions such as block ciphers have no security parameter, and hence it is not possible to define their security merely in terms of an adversary's computational power.

In 1981, while cryptography was mainly concerned with classical adversaries, in another part of the scientific world, Feynman [60] raised a question about the possibility of building quantum computers. Following Feynman's idea, a number of quantum algorithms such as Shor's algorithm [103] were developed that would, if fully realised on a quantum computer, break modern cryptosystems such as those based on the difficulty of factoring [94] or the discrete log problem. Emerging quantum technologies raise the question that if an adversary possesses a quantum computer, then which modern cryptosystems are secure, and which are not? In other words, since the security of modern cryptographic schemes is based on the computational power of the classical adversary, they must be re-examined for the case of a quantum adversary who is in possession of a quantum computer. Moreover, understanding which modern cryptosystems are secure against quantum adversaries is a relevant and important question because, for example, highly confidential information that is encrypted today should remain protected even if quantum computers are built in couple of decades. This is the subject of a new research field, called *post-quantum cryptography* [22]. This field attempts to design cryptographic schemes which will be secure even in the presence of quantum computers. To be secure in a quantum setting, a cryptosystem must have an underlying problem that is hard for a quantum computer, as well as a security reduction that shows how to solve this hard problem by using a quantum adversary that breaks the cryptosystem.

Quantum computing appears to have very little impact on symmetric encryption. The generic quantum attack on block ciphers using Grover's algorithm [65] requires $O\left(2^{n/2}\right)$ queries for key length n and thus can be countered by doubling the key length. For this reason, symmetric encryption has not been the subject of research in post-quantum cryptography, and there has not been a systematic exploration to see whether their security proofs carry over to this quantum setting. Therefore we will miss out on any case where a symmetric encryption scheme might have flaws in its construction that could be exploited by a quantum adversary.

In contrast to the above, there is also another natural and conservative security model that is beyond post-quantum cryptography. This is a model that allows the quantum adversary to issue quantum queries. This is possible if any cryptosystem is run on a quantum computer. So far, there has been little discussion, however, of the security of existing symmetric encryptions in this security model. Furthermore, the question has not been explored much of how existing classical security notions translate into this quantum setting, and whether they can be satisfied.

1.2 Contribution

This thesis considers the security of symmetric encryptions in two different models. These are: a quantum computation security model where a quantum adversary is only allowed to make classical queries, and a quantum superposition security model where quantum queries are permitted.

In the case of the quantum computation model, we explore how existing classical confidentiality notions translate into this model by formally analysing the security of encryption schemes such as Counter mode. We provide our security analysis in the concrete security framework. In this way, the security of an encryption scheme is quantified based on properties of its underlying primitive, such as the block length or key length, as well as resources available to the quantum adversary.

We then turn to the quantum superposition model, where we discuss the security of simple block ciphers such as the Even-Mansour scheme [57]. We show that a quantum adversary with superposition access to an encryption oracle can break the Even-Mansour block cipher with key length n using $O(n)$ queries. This should be compared to the lower bound given by Even and Mansour for their scheme, and to the generic quantum attack against symmetric encryption schemes using Grover's algorithm, both of which require $O\left(2^{n/2}\right)$ queries. Our $O(n)$ attack extends to two special cases of the multiple-round Even-Mansour scheme. These are the case of arbitrarily many rounds using a single permutation with identical round keys, and the case of two rounds using a single permutation with round keys derived from a very basic key schedule.

Furthermore, we explore how existing classical confidentiality notions translate into this model. We introduce a new confidentiality notion called *Real-or-Permutation* (RoP). We show that the notion of RoP in the quantum superposition model arises as its classical counterpart, as opposed to the other classical confidentiality notions which need to be rethought from scratch. We demonstrate that RoP is satisfiable in the quantum superposition model by proving the security of a generic symmetric encryption schemes under both a *quantum superposition chosen plaintext attack* and a *quantum superposition chosen ciphertext attack*. Since ultimately we are interested in the security of our schemes in the semantic security model [64, 12], we also propose a quantum analogue of semantic security and discuss its implication with RoP.

1.3 Thesis Structure

We start off Chapter 2 by explaining the structure of security definitions and security proofs in the provable security framework. This is followed by a discussion on asymptotic and concrete approaches in the provable security framework. We then turn to quantum mechanics, where we explain some of its basic principles through describing the Mach-Zehnder interferometer. These principles are: quantum superposition, unitary evolution, measurement, entanglement, and the density operator. In Chapter 3 we discuss the idea of quantum computers before explaining the quantum circuit model as a model of quantum computation. We also discuss the possibility of quantum computers being built, and we mention several advances in the field. The quantum algorithms of Simon and Grover are explained in Chapter 3. Some limitations of quantum computers are also pointed out. We give a definition for a quantum adversary, and explain two different models of quantum attack.

In Chapter 4, we define the basic building blocks of symmetric encryption such as block ciphers and modes of operation. We also provide definitions for pseudorandom function families, pseudorandom permutation families, and quantum pseudorandom function families. Furthermore, we discuss the basic security models used for symmetric encryption and state the relations which hold between these models. Considering these security definitions, we assess the security of Counter mode against quantum computation attacks. At the end of this chapter, we discuss other symmetric primitives that help us to achieve other cryptographic goals such as integrity.

Chapter 5 shows how powerful quantum superposition attacks could be. We explain the construction of the Even-Mansour scheme, and how the slide with a twist attacks work on it. Then we discuss the possibility of exploiting this attack using Simon's algorithm. We explain why this attack is successful even though it is only partially satisfies Simon's problem. At the end, an extension of this attack to other variations of the Even-Mansour scheme is discussed.

In Chapter 6, we define the basic security models used for symmetric encryption in the quantum superposition model. This chapter begins by arguing that existing classical security models for encryption schemes need to be rethought from scratch in the quantum superposition model. Moreover, we define a new confidentiality notion, Real-or-Permutation (RoP), and discuss its achievability under a quantum superposition chosen plaintext attack and a quantum superposition chosen ciphertext attack. We also define a quantum analogue of semantic security and show a reduction from RoP to semantic security.

In the final chapter, Chapter 7, we discuss the meaning of our results, further works, and open problems.

Chapter 2

Preliminaries

Contents

This chapter briefly explains a number of key ingredients of both cryptography and quantum mechanics. We start by giving the notation, followed by a discussion on the main cryptographic framework we will use in the thesis, and finally we finish off the chapter by giving a hint of quantum mechanics and its properties (mostly) via describing a physical experiment.

2.1 Notation

Let $\mathcal{X}$ be a set. Then $|\mathcal{X}|$ denotes its size and $x \leftarrow\!\$\, \mathcal{X}$ denotes sampling an element uniformly at random from $\mathcal{X}$ and assigning it to x. We use $\{0,1\}^*$ to denote the set of all finite binary strings. Let x be a binary string. We use $|x|$ to denote its bit length. Let $\{0,1\}^n$ represents the set of all binary strings of length n where n is any positive integer. For a bit b, let b^n denotes the strings of n consecutive b. For any two strings

x and y, $x \oplus y$ denotes their bitwise addition and $x \,||\, y$ denotes their concatenation. To show the set of all functions with domain $\mathcal{X}$ and range $\mathcal{Y}$ we use $\mathsf{Func}\,(\mathcal{X}, \mathcal{Y})$. We use $\mathsf{Perm}\,(\mathcal{X})$ to show the set of all permutations on domain $\mathcal{X}$. If $\mathcal{X} = \{0,1\}^l$ or $\mathcal{X} = \{0,1\}^*$, and $\mathcal{Y} = \{0,1\}^n$ for any positive integers l and n then for compactness of notation we will often use $\mathsf{Func}\,(l, n)$, $\mathsf{Func}\,(*, n)$ and $\mathsf{Perm}\,(l)$, respectively. An algorithm may be randomised, unless otherwise stated. An adversary is an algorithm. We use capital letters (such as A) to denote a classical adversary. For any algorithm A, we use $y \leftarrow A\,(x_1, x_2, \ldots)$ to denote the operation of running algorithm A on inputs $x_1, x_2, \ldots$ with fresh coins and assigning its output to y. When a definition involves multiple experiments, if the name of an experiment is surrounded by a box, the experiment includes $\boxed{\text{the boxed codes}}$, otherwise it dose not. Often the boxed code replaces the code adjacent to it.

2.2 Provable Security

Traditionally, the approach towards designing a cryptographic scheme has been seen as a cycle of ‘build’, ‘break’, and ‘fix’. That is, a cryptographic goal is recognised, and a cryptographic solution is proposed. One then tries to discover its weaknesses through concrete attacks. If any are found, the solution is amended to remove the weaknesses. If any new weaknesses are found at any later point in time, then the solution is amended again. There are a number of difficulties with this heuristic approach. Assume that at one point no new weaknesses are found. It is unclear whether this is because all the solution’s weaknesses have been found and fixed, or whether there are still weaknesses yet to be found. In the other words, it is unclear when the cycle has concluded. Hence, the cycle should be iterated until one feels confident that the solution is adequate. But, one should always consider that a design error might be discovered at any time.

Shannon [100] introduced a more systematic approach to cryptography by taking proofs and definitions into account. Shannon showed what it means for a scheme to be *perfectly secure*. Shannon set a goal to achieve privacy, then he defined a symmetric encryption scheme and proved that the scheme achieves the goal perfectly. The symmetric encryption scheme is known as ‘one-time pad’. Shannon argued that given two different messages, M_0 and M_1, and a ciphertext C, where C is the encryption of either M_0 or M_1 with equal probability, then the scheme is perfectly secure. The probability is taken over any randomised encryption algorithm and the choice of key. Perfect security, also called *information-theoretic security*, though very powerful and desirable, proved to be difficult to achieve in practice. To achieve perfect privacy, the size of the messages in bits should not exceed the size of the key bits. In cryptography, however, it is usually preferred to encrypt many message bits with a finite key bits instead of some arbitrary key bits. A finite key bits is easier to manage in practice. Moreover, parties do not need to know the total size of a message prior to encryption.

In contrast to Shannon's perfect security, there is another approach to modern cryptography. Instead of focusing on the impossibility of any attacks on a scheme, it focuses on the infeasibility of those attacks. Cryptographic schemes in this approach are considered breakable in theory but not in practice. In order to assess the security of a scheme, the amount of computing power available to an adversary is considered. As long as the adversary does not have too much computing time or other computational resources, the scheme is considered secure. Hence, it is called *computational security.* Hereafter by 'security' we mean computational security unless otherwise stated. Rabin [92] was one of the first cryptographers to came up with a mathematical proof of security for a scheme. But it was not until 1982 when Goldwasser and Micali [64] published their pioneering work providing a formal mathematical framework to rigorously analyse security of cryptographic schemes. Their proposed formal framework essentially consisted of a 'security definition', a 'cryptographic scheme', and a 'reduction proof technique'. This approach has come to be called *provable security.*

The first step in provable security is the formulation of security definitions in a precise mathematical form. That is, what it actually means to say that a cryptographic scheme is secure and what goal it intends to achieve. For instance, the goal of a scheme may be is to achieve confidentiality or to achieve ciphertext integrity. A security definition is usually expressed as an 'experiment' conducted by a 'challenger'. An adversary plays the experiment with regards to a cryptographic scheme while it might be given access to a set of 'oracles' maintained by the challenger. Provable security is concerned with the probability of an adversary 'winning' the experiment. The winning condition could take different forms, such as an adversary distinguishing between two experiments, or finding the plaintext corresponding to a given challenge ciphertext. The winning probability of an adversary is represented by the adversary's 'advantage'. The advantage is a measure of how much better an adversary can do to win the experiment compared to simply guessing. That is, the maximal winning probability over a class of adversaries which indicates the security of a scheme. The advantage over a class of adversaries lets us quantify the security of a scheme against all adversaries with bounded resources. If the advantage of a 'feasible' adversary exceeds some acceptable threshold by a substantial amount, then the scheme is considered broken. A classical adversary is a randomised algorithm parametrised by its computational resources. For a quantum adversary see Section 3.4. Usually the resources of interest are the number of oracle queries made by the adversary, the size of the oracle queries, and the running time of the adversary. By convention [15], the running time includes the space required to store the program that describes the adversary. The latter prevents the adversary from embedding arbitrary large look-up tables in its description.

The second step of the provable security approach is reduction proof technique. Here, the security of a scheme with regards to a particular security definition is proved via a reduction to the security of the underlying cryptographic primitive or some number-theoretic hard problem, such as factoring. The reduction proof technique enables us to focus on the security of the underlying primitive or the underlying hard problem instead of directly looking into cryptanalysis of the original scheme. The basic idea of reduction is the same as the theory of **NP**-completeness. Assume, without loss of generality, an adversary A attacks a scheme S with underlying primitive P. A reduction transforms A to another adversary A' which simulates the challenger for A against S and uses A's output to break the underlying primitive P. If A is able to break S then A' is able to break P. Hence, as long as the underlying primitive P is secure, then the scheme S is secure.

Note that security proofs only hold within specific security definitions and with regards to a number of assumptions about the adversary's capabilities. Therefore they will not rule out all classes of attack. For instance, there may exist an attack in practice against a provably secure scheme. Another important point is that the heuristic provable security approach and the complexity of the constructions make cryptographic schemes inefficient in practice. Koblitz and Menezes [76, 75] point out some of the above issues among other criticisms of provable security. The issues they raise make an important point that when one uses results from provable security, great attention must always be given to the context of the results and what they actually mean in practice. Bellare [10] discusses that the term 'provable security' is in some way misleading. Provable security does not prove a scheme secure. It is merely a demonstration of a reduction of the security of a scheme to the intractability of some underlying primitive. Bellare suggests the term *reductionist security* instead of provable security.

It is worth mentioning that the information-theoretic security falls into the provable security paradigm where the term 'provable' makes more sense because the security proofs are unconditional.

2.2.1 Asymptotic vs. Concrete Security

Since its inception, provable security has mainly evolved in a complexity theoretic framework where adversaries and transformations are 'efficient' if they run in *polynomial time*, and adversaries' 'success probabilities' are bounded by a *negligible* function. In this approach, a 'security parameter' is introduced which is relative to the polynomial running time and success probabilities. A scheme is secure if every polynomial time adversary obtains only negligible advantage in attacking the scheme. This is called the *asymptotic* approach, and is used in all early cryptographic definitions and security proofs [95]. Results expressed in the asymptotic framework proved to be unpopular among practitioners because they require precise numbers about the adversary's computational resources or the security parameters etc, which are only loosely captured by the asymptotic framework. For instance, block ciphers are widely used and very popular in practice, but they seemed to be outside of the domain of provable security for some time [10].

Bellare and Rogaway introduced 'practice-oriented provable security' via a series of papers [15, 14, 13]. The goal of this is to explicitly capture the quantitative nature of security, as opposed to the qualitative nature of security captured by the asymptotic framework, through a concrete treatment of security. This is often called *concrete* security. In this approach, the adversary's advantage is quantified by the adversary's resources. Hence the adversary's computational model becomes relevant in the concrete security. For instance, consider a scheme S with underlying primitive P. We assume that P is secure against an adversary A' running in time t', making q' oracle queries, totalling μ' bits. Then the scheme S may be proved to be secure against an adversary A running in time t, making q oracle queries, totalling μ bits. Bellare [10] also points out that the quality of a reduction is very important. For instance, tightness of security bounds in a reduction affects efficiency of the reduction in practice. A reduction with tighter security bounds is more efficient. Therefore the concrete security results help protocol designers to understand what security to expect in practice.

In this thesis, we are interested in concrete security treatment of symmetric schemes against quantum adversaries.

2.3 Quantum Mechanics

Quantum mechanics is a branch of physics relating to atomic and subatomic scale phenomenon which classical mechanics could not explain. Quantum mechanics began at the turn of the 20th century and it gradually gained acceptance and experimental verification between 1900 and 1930 through the contribution of multiple scientists to the foundation of its revolutionary principles.

One of the experiments which cannot be reasonably explained with classical mechanics, but can be easily explained with quantum mechanics, is the interference that happens in the *Mach-Zehnder* interferometer. It is a simple experiment which exemplifies a number of the main principles of quantum mechanics which we explain in this chapter. Depending on the hardware used in the experiment, it can be run with photons, electrons, neutrons, atoms, or even molecules. Here we assume the experiment is set up for photons.

The Mach-Zehnder interferometer, as depicted in the Figure 2.1, consists of two half-silvered mirrors (or beam splitters), two full mirrors, and two photon detectors. The interferometer is a device that allows us to measure the interference of photons following two different paths. We label the lower path with $|0\rangle$ and the upper path

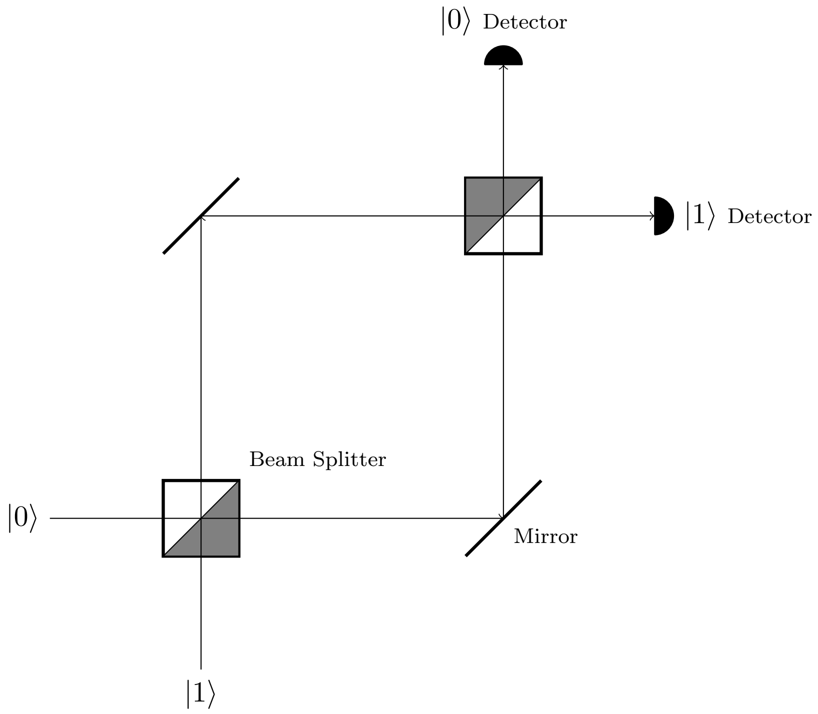

Figure 2.1: The Mach-Zehnder interferometer

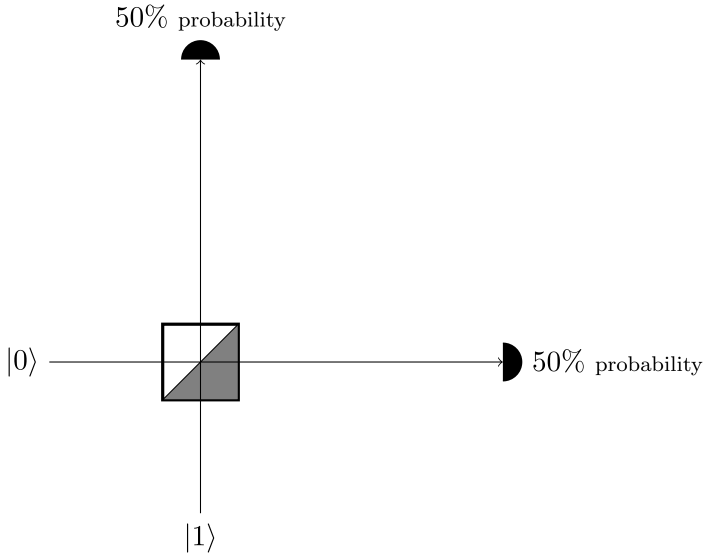

Figure 2.2: Each detector clicks with 50% probability

with $|1\rangle$. To see the behaviour of the half-silvered mirror, we simplify the experiment by replacing the full mirrors with two detectors instead, as shown in the Figure 2.2. Assume a single photon travelling in the lower path $|0\rangle$. Here we assume a photon always starts in the $|0\rangle$ path unless otherwise stated. When the photon passes through the half-silvered mirror, exactly one of the two detectors clicks. Note that the clicks we get are discrete. Precisely one of the detectors clicks at any one time. For instance, we never get two half-clicks, or any other fractions of a click. This discreteness is one of the main properties of quantum mechanics. By repeating the experiment we notice that each detector clicks about half of the time. The simplest explanation is that the half-silvered mirror acts as a classical coin flip, randomly sending each photon one way or the other with 50% probability. We now consider the full apparatus again as depicted in Figure 2.1. Note that the two paths to the detectors are the same length. We know that the first half-silvered mirror, with 50% probability, sends a photon in one of the two paths. Hence, we expect each of the $|0\rangle$ and $|1\rangle$ detectors to detect roughly half of the photons. It turns out that this classical intuition is false since all the photons are detected at the $|1\rangle$ detector. A rough explanation for now is that when exiting the first beam splitter, the reflected beam picks up a phase shift while the transmitted beam does not. This phase shift introduces a phase difference between the two paths. Since the two paths are the same length, there is constructive interference on the path to the $|1\rangle$ detector, and destructive interference on the path to the $|0\rangle$ detector. We can shift the probability distribution from 100% detection by the $|1\rangle$ detector to 100% detection by the $|0\rangle$ detector by placing a π-phase shifter along the $|0\rangle$ path. Therefore, by observing the detectors, we can distinguish whether this π-phase shifter exists or not. Moreover, if we place phase shifters ϕ_0 and ϕ_1 along the $|0\rangle$ and $|1\rangle$ paths respectively, the proportions are $\cos^2\left(\frac{\phi_0-\phi_1}{2}\right)$ detections by the $|0\rangle$ detector and $\sin^2\left(\frac{\phi_0-\phi_1}{2}\right)$ detections by the $|1\rangle$ detector, as shown in Figure 2.3.

We explain the role of the experiment's key ingredients in the following subsections.

2.3.1 Notation

Let $[n]$ denotes the set $\{1,\ldots,n\}$. A vector space is denoted with capital script letters such as $\mathcal{V}$. A vector is denoted with bold lower case letters such as $\mathbf{v}$. We use

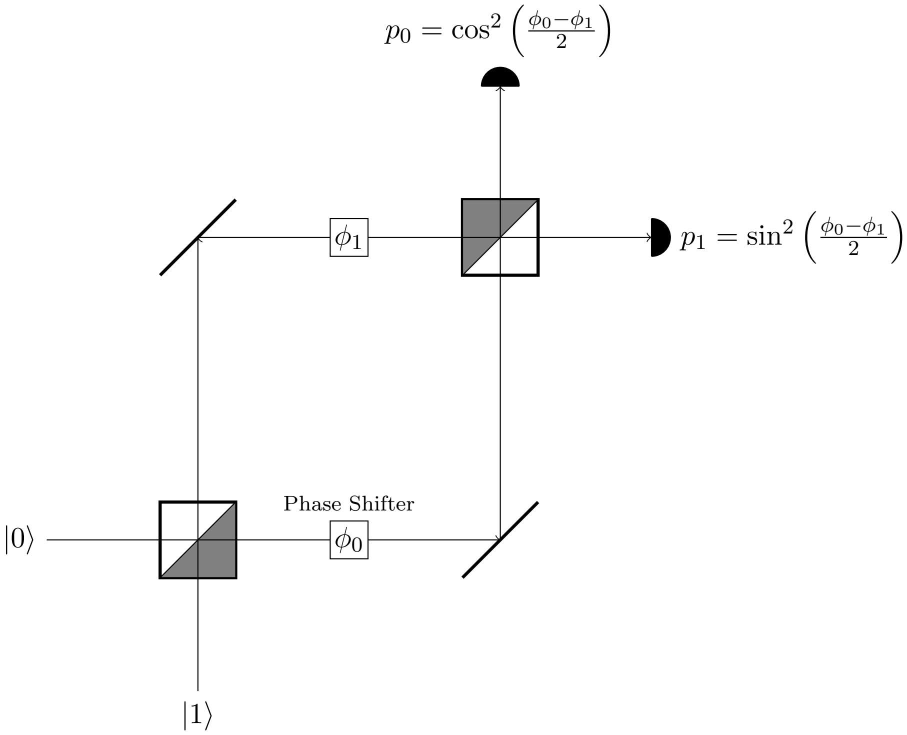
$p_0 = \cos^2\left(\frac{\phi_0-\phi_1}{2}\right)$
ϕ_1
$p_1 = \sin^2\left(\frac{\phi_0-\phi_1}{2}\right)$
Phase Shifter
$|0\rangle$
ϕ_0
$|1\rangle$

Figure 2.3: The Mach-Zehnder interferometer with two phase shifters. By adjusting the phase shifter, one can change the probabilities of photons striking the $|1\rangle$ detector and the $|0\rangle$ detector.

$\mathbb{C}^n$ to denote the n-dimensional complex vector space. We denote the components of a vector $\mathbf{v} \in \mathcal{V}^n$ by $v_i, i \in [n]$. We use bold capital letters such as $\mathbf{M}$ to denote matrices. The components of a matrix $\mathbf{M} \in \mathcal{V}^{m \times n}$ are denoted by $M_{i,j}$ where, $i \in [m]$ and $j \in [n]$. Let $\mathbf{I}_n$ denotes the $n \times n$ 'identity' matrix. We will often use merely $\mathbf{I}$ instead, when the dimension is clear from context. If $\mathbf{M}$ is square we use $\mathbf{M}^{-1}$ to denote the 'inverse' of the matrix $\mathbf{M}$. Let $\mathbf{M}^*$ denote 'complex conjugate' of the matrix $\mathbf{M}$, and $\mathbf{M}^T$ denotes the 'transpose' of the matrix $\mathbf{M}$. We use $\mathbf{M}^\dagger$ to denote the *Hermitian conjugate* or 'adjoint' of the matrix $\mathbf{M}$. Note that $\mathbf{M}^\dagger = \left(\mathbf{M}^T\right)^*$. For a matrix $\mathbf{M}$, let $\mathrm{Tr}\left(\mathbf{M}\right)$ denotes the trace of $\mathbf{M}$. For any two matrices $\mathbf{M}$ and $\mathbf{N}$, let $\mathbf{M} \otimes \mathbf{N}$ denote their tensor product. We use *Dirac* notation to denote quantum states. Let $|\mathbf{v}\rangle = \mathbf{v}$ and $\langle\mathbf{v}| = \mathbf{v}^*$. The latter is called a 'bra' and the former a 'ket'. For any two vectors $\mathbf{v}$ and $\mathbf{w}$, we use $\langle\mathbf{v}|\mathbf{w}\rangle$ to denote their inner product, and we use $|\mathbf{v}\rangle\langle\mathbf{w}|$ to denote their outer product. If $|\mathbf{v}\rangle = \sum_{i\in[n]} \alpha_i |v_i\rangle$ and $|\mathbf{w}\rangle = \sum_{j\in[n]} \alpha_j |w_j\rangle$, the tensor product $|\mathbf{v}\rangle \otimes |\mathbf{w}\rangle = \sum_{i\in[n]} \sum_{j\in[n]} \alpha_{ij} |v_i\rangle \otimes |w_j\rangle$. For abbreviation we also often use, $|v_i, w_j\rangle$, $|v_i w_j\rangle$ or $|v_i\rangle |w_j\rangle$ instead of $|v_i\rangle \otimes |w_j\rangle$. We use $\mathcal{H}$ to denote the *Hilbert* space, which is a complex vector space with the inner product $\langle\cdot|\cdot\rangle$. Finally, for two variables i and j, we denote the Kronecker delta by δ_{ij}.

2.3.2 Superposition

A classical state is a state in which a physical system can be found when it is observed. Consider a physical system that can be in n different, mutually exclusive classical states, for instance the states $|1\rangle, \ldots, |n\rangle$. A pure quantum state $|\psi\rangle$ is a linear combination or a *superposition* of the classical states

$$|\psi\rangle = \sum_{i\in[n]} \alpha_i |i\rangle \ , \tag{2.1}$$

where α_i is called the amplitude of $|i\rangle$ in $|\psi\rangle$. It is a complex number. The squared of the amplitude of $|i\rangle$, $|\alpha_i|^2$, tells us the probability of observing the system in the state $|i\rangle$. Hence

$$\sum_{i\in[n]} |\alpha_i|^2 = 1 \ . \tag{2.2}$$

We discuss how to obtain the probability in the next subsection.

A system in quantum state $|\psi\rangle$ can be in a superposition of all classical states. The states $|1\rangle, \dots, |n\rangle$ form an 'orthonormal basis' of an n-dimensional Hilbert space where a quantum state $|\psi\rangle$ is a vector in this space. Consider a 2-dimensional Hilbert space where the states (orthonormal basis) are $|0\rangle$ and $|1\rangle$. This orthonormal basis is also referred to as the *computational basis.* A *qubit* is a quantum state (a vector) which can be written as a linear combination of $|0\rangle$ and $|1\rangle$.

Recall the Mach-Zehnder interferometer. The labels $|0\rangle$ and $|1\rangle$ are classical states of the lower and the upper path, respectively. In the experiment, after striking the first beam splitter, a photon behaves as though it is propagated through all possible paths to the detectors. Therefore the state of a photon after exiting the first beam splitter is a superposition of the two states, $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$. Similarly, if a photon is sent through the first beam splitter starting in the $|1\rangle$ path, its state after exiting the beam splitter is $\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$. Note that the negative sign is the phase shift picked up by the reflected beam. If we bring the phase shifters ϕ_0 and ϕ_1 into the picture then they change the state in the following way

$$\begin{aligned} \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle \xrightarrow{\phi_0 \wedge \phi_1} & \frac{e^{i\phi_0}}{\sqrt{2}}|0\rangle + \frac{e^{i\phi_1}}{\sqrt{2}}|1\rangle \\ &= \frac{e^{i\frac{\phi_0+\phi_1}{2}}}{\sqrt{2}}\left(e^{i\frac{\phi_0-\phi_1}{2}}|0\rangle + e^{-i\frac{\phi_0-\phi_1}{2}}|1\rangle\right) \ . \end{aligned} \tag{2.3}$$

The second beam splitter operates in the same way as the first. Hence, it transforms

$$\begin{aligned} &\frac{e^{i\frac{\phi_0+\phi_1}{2}}}{\sqrt{2}}\left(e^{i\frac{\phi_0-\phi_1}{2}}|0\rangle + e^{-i\frac{\phi_0-\phi_1}{2}}|1\rangle\right) \\ &\longrightarrow e^{i\frac{\phi_0+\phi_1}{2}}\left(\cos\left(\frac{\phi_0-\phi_1}{2}\right)|0\rangle + i\sin\left(\frac{\phi_0-\phi_1}{2}\right)|1\rangle\right) \ . \end{aligned} \tag{2.4}$$

Equation 2.4 shows us that the experimental outcomes are influenced by the existence of different possible paths leading to detection. For instance, the equation tells us that a photon can strike the $|0\rangle$ detector from two different paths. One with the probability amplitude of $\cos\left(\frac{\phi_0-\phi_1}{2}\right)$, and the other with the probability amplitude of $i\sin\left(\frac{\phi_0-\phi_1}{2}\right)$.

2.3.3 Measurement

Consider the quantum state $|\psi\rangle$, which is a superposition of a number of classical states (see Equation 2.1). Measuring $|\psi\rangle$ in the computational basis yields one and only one classical state $|i\rangle$ with probability $|\alpha_i|^2$, where α_i is the corresponding amplitude. The outcome is not determined prior to the measurement. We merely can predict that the measurement outcome is the state $|i\rangle$ with probability $|\alpha_i|^2$. This means that measuring a quantum state induces a probability distribution on the classical states, which implies

$$\sum_{i\in[n]} |\alpha_i|^2 = 1 \ . \tag{2.5}$$

Moreover, measuring the quantum state $|\psi\rangle$ collapses $|\psi\rangle$ to the classical state $|i\rangle$ and destroys all other information that the superposition $|\psi\rangle$ might have contained. Thus, a measurement is 'irreversible'.

The above is an example of a *projective measurement.* In general, a projective measurement is described by a set of projectors $\mathsf{M}_1, \ldots, \mathsf{M}_m$ $(m \leq n)$ which satisfy the completeness equation

$$\sum_{i\in[m]} \mathsf{M}_i = \mathbf{I} \ . \tag{2.6}$$

Let V be a subspace of the Hilbert space $\mathcal{H}$. Let M_i project on subspace V_i. Then for every state $|\psi\rangle \in V$, we can write

$$|\psi\rangle = \sum_{i\in[m]} |\psi_i\rangle \ , \tag{2.7}$$

where $|\psi_i\rangle = \mathsf{M}_i |\psi\rangle$ and $|\psi_i\rangle \in V_i$. Applying this projective measurement to the pure quantum state $|\psi\rangle$ yields the outcome i with probability

$$\| |\psi_i\rangle \|^2 = \mathrm{Tr}\left(\mathsf{M}_i |\psi\rangle\langle\psi|\right) \ , \tag{2.8}$$

and the quantum state collapses to a new state

$$\frac{|\psi_i\rangle}{\| |\psi_i\rangle \|} = \frac{\mathsf{M}_i |\psi\rangle}{\| \mathsf{M}_i |\psi\rangle \|} \ . \tag{2.9}$$

Now we can show that measurement in the computational basis is a special case of the projective measurement, where M_i projects onto the computational basis state $|i\rangle$ and the corresponding subspace V_i is the space spanned by $|i\rangle$. Consider a single qubit $|\psi\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle$ where the two possible outcomes are defined by the two projectors $\mathsf{M}_0 = |0\rangle\langle 0|$ and $\mathsf{M}_1 = |1\rangle\langle 1|$. Note that the two projectors satisfy the completeness equation. The probability of obtaining 0 after the measurement is

$$\| \mathsf{M}_0 |\psi\rangle \|^2 = \| \alpha_0 |0\rangle \|^2 = |\alpha_0|^2 \ , \tag{2.10}$$

and the state $|\psi\rangle$ collapses to

$$\frac{\alpha_0 |0\rangle}{\| \alpha_0 |0\rangle \|} = \frac{\alpha_0}{|\alpha_0|} |0\rangle \ . \tag{2.11}$$

Note that $\alpha_0 / |\alpha_0|$ is an irrelevant phase factor. Similarly, the probability of obtaining 1 after the measurement is $|\alpha_1|^2$ and the quantum state collapses to

$$\frac{\alpha_1 |1\rangle}{\| \alpha_1 |1\rangle \|} = \frac{\alpha_1}{|\alpha_1|} |1\rangle \ , \tag{2.12}$$

where $\alpha_1/\left|\alpha_1\right|$ is an irrelevant phase factor.

We look back at the state of a photon after exiting the first beam splitter in the Mach-Zehnder interferometer:

$$\frac{1}{\sqrt{2}}\left|0\right\rangle + \frac{1}{\sqrt{2}}\left|1\right\rangle \ . \tag{2.13}$$

If we measure this state in the computational basis, then we find the photon in the $\left|0\right\rangle$ path with probability $\left|\frac{1}{\sqrt{2}}\right|^2$, and in the $\left|1\right\rangle$ path with probability $\left|\frac{1}{\sqrt{2}}\right|^2$. Similarly, regarding the state of a photon after passing through the second beam splitter,

$$e^{i\frac{\phi_0+\phi_1}{2}}\left(\cos\left(\frac{\phi_0-\phi_1}{2}\right)\left|0\right\rangle + i\sin\left(\frac{\phi_0-\phi_1}{2}\right)\left|1\right\rangle\right) \ , \tag{2.14}$$

a photon arrives at the detector $\left|0\right\rangle$ or $\left|1\right\rangle$ with probability $\cos^2\left(\frac{\phi_0-\phi_1}{2}\right)$ and $\sin^2\left(\frac{\phi_0-\phi_1}{2}\right)$, respectively.

The most general quantum measurement, called a *positive operator-valued measurement*, or POVM, is described by positive operators E_i satisfying

$$\sum_{i\in[m]} \mathsf{E}_i = 1 \ . \tag{2.15}$$

2.3.4 Unitary Evolution

Consider a closed quantum system, which is a system that is not interacting in any way with other systems. Quantum mechanics tells us that the state of a closed quantum system evolves during the time from, say, $\left|\psi\right\rangle$ at time t_1 to $\left|\varphi\right\rangle$ at time t_2. These states are related by a unitary operator $\mathbf{U}$ which depends only on times t_1 and t_2

$$|\varphi\rangle = \mathbf{U}\,|\psi\rangle\ . \tag{2.16}$$

This unitary transformation explains how the states of a closed quantum system at two different times are related. A linear operator $\mathbf{U}$ is unitary if and only if $\mathbf{U}^{-1}$ exists and $\mathbf{U}^{\dagger} = \mathbf{U}^{-1}$. Equivalently, $\mathbf{U}$ is unitary if and only if $\mathbf{U}\mathbf{U}^{\dagger} = \mathbf{I}$. Thus, any unitary operation on quantum states is 'reversible'. Hereafter we always consider closed quantum systems unless otherwise stated.

Unitary operators can be described in matrix form. For instance, the action of the beam splitter in the Mach-Zehnder interferometer is

$$\begin{aligned} |0\rangle &\longrightarrow \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) \\ |1\rangle &\longrightarrow \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right)\ , \end{aligned} \tag{2.17}$$

which can be described by the matrix

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

while the action of the phase shifters is described by the matrix

$$\begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix}\ . \tag{2.19}$$

In Chapter 3 we see the importance of unitary operations in quantum computation and that they allow us to consider time as 'discrete' and discuss 'computational steps'.

2.3.5 Entanglement

Consider a quantum system of more than 1 qubit. The quantum state of this system is described by the tensor product of its components. For instance, if the system consists of i qubits, each represented by $|\psi_i\rangle$ where $i \in [n]$, then the quantum state of the system is described by

$$|\psi_1\rangle \otimes \ldots \otimes |\psi_n\rangle \ . \tag{2.20}$$

Assume a 2-qubit system. Let $|\psi_1\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle$ and $|\psi_2\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle$ describe the state of first and second qubit, respectively. For both qubits we have that $|\alpha_0|^2 + |\alpha_1|^2 = 1$. Also, $|\psi_1\rangle \in \mathcal{H}_1$ and $|\psi_2\rangle \in \mathcal{H}_2$. The state of the 2-qubit system is

$$\begin{aligned} |\psi\rangle &= |\psi_1\rangle \otimes |\psi_2\rangle \\ &= \alpha_{00} |00\rangle + \alpha_{01} |01\rangle + \alpha_{10} |10\rangle + \alpha_{11} |11\rangle \ , \end{aligned} \tag{2.21}$$

where $|\alpha_{00}|^2 + |\alpha_{01}|^2 + |\alpha_{10}|^2 + |\alpha_{11}|^2 = 1$ and $|\psi\rangle \in \mathcal{H}_1 \otimes \mathcal{H}_2$. This implies that a 2-qubit system has 4 basis states, $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$. More generally, an n-qubit system has 2^n basis states of the form $|b_1 b_2 \ldots b_n\rangle$ where $b_i \in \{0, 1\}$.

The state $|\psi\rangle$ is a 'product state' if and only if $\alpha_{00}\alpha_{11} = \alpha_{01}\alpha_{10}$. The product state can be factored into the product of two independent qubit states. However, the state of a bipartite system can also be in a form such that cannot be decomposed as the tensor product of two independent qubit states. For example the Einstein-Podolsky-Rosen (EPR) [56] state

$$\frac{1}{\sqrt{2}} |00\rangle + \frac{1}{\sqrt{2}} |11\rangle \ . \tag{2.22}$$

When this is the case, the two qubits are, in a way, strongly correlated. This introduces an important idea related to composite quantum systems, called *entanglement*. The EPR state is interesting because observing either the first qubit or the second qubit immediately fixes the unobserved qubit to a classical state. For instance, if the outcome of measuring the first qubit in Equation 2.22 is the classical state $|0\rangle$, then the EPR state collapses to $|00\rangle$. Thus, we obtain information about the second qubit by only observing the first qubit.

2.3.6 The Density Operator

Consider the state vector of a quantum system that evolves in time via a number of unitary operations until it is measured. At this time, one can employ projective measurement to predict the probabilities for different results. Assume we want to confirm the predictions. We need to prepare a known initial state, apply the unitary operations on it, and then measure it. If we iterate this process enough, the results should show statistical agreement with the predicted probabilities. But, in general, it is impossible to prepare the exact same quantum state every time. Instead, what we know is that the quantum system is in state $|\psi_i\rangle$ with probability p_i. This is known as 'statistical mixture'. The state of the quantum system is either $|\psi_1\rangle$ or $|\psi_2\rangle$ (or any others) but we do not know exactly which one. Note that this is not the same as saying the quantum system is in state $|\psi\rangle = |\psi_1\rangle + |\psi_2\rangle + \cdots$, which is known as 'coherent superposition'. In some sense, the quantum system is in both $|\psi_1\rangle$ and $|\psi_2\rangle$ (and any others) at the same time. In the case of statistical mixture, we have some probability distribution of states.

Assume a quantum system described by a set of state vectors $\{|\psi_1\rangle, \ldots, |\psi_n\rangle\}$ with their corresponding probabilities $\{p_1, \ldots, p_n\}$ such that $\sum_i p_i = 1$. All we know is that the quantum system's state is $|\psi_i\rangle$ with probability p_i. The *density operator* is an operator ρ associated with the ensembles $\{p_i, |\psi_i\rangle\}$ and is defined as

$$\rho = \sum_{i=1}^{n} p_i \, |\psi_i\rangle\langle\psi_i| \ . \tag{2.23}$$

Note that ρ is a positive operator and $\mathrm{Tr}(\rho) = 1$. The density operator, also called the density matrix, is an alternative mathematical approach to formulate quantum mechanics. For instance it can be regarded as the state of a quantum system. In order to measure the state, we can apply an operator M_i, $\mathsf{M}_i \geq 0$ and $\sum_i \mathsf{M}_i = 1$, to

ρ to obtain the probability of an outcome i:

$$\Pr\left[\text{outcome} = i | \text{state} = \rho\right] = \operatorname{Tr}\left(\rho \mathsf{M}_i\right) \ , \tag{2.24}$$

where $i \in [m]$. Hence, applying measurement to a density operator results in a probability distribution. If we are given two density operators ρ_0 and ρ_1 then after measuring them we obtain two probability distributions, $\mathcal{P}_0$ and $\mathcal{P}_1$ respectively. We can distinguish between these two probability distributions. In fact distinguishing between probability distributions is an important and well-studied problem in statistical science. This gives us an interesting intuition. Since we are able to distinguish between two probability distributions, $\mathcal{P}_0$ and $\mathcal{P}_1$, then we can also distinguish between their corresponding density operators ρ_0 and ρ_1. There are different measures for quantum distinguishability [61]. Here we discuss two of them which we will use in the thesis: 'trace distance[1]' and 'fidelity'. Both of these distance measures are 'static', meaning they quantify how close two quantum states are.

Given two quantum states ρ and σ, the trace distance between them is

$$\mathsf{D}\left(\rho, \sigma\right) = \frac{1}{2} \operatorname{Tr} \left|\rho - \sigma\right| \ . \tag{2.25}$$

The quantum trace distance can be related to the classical trace distance by considering the probability distributions induced by a measurement.

Result 1 ([89] Theorem 9.1) *Given two quantum states ρ and σ, and a* POVM $\{\mathsf{E}_m\}$*, let $p_m = \operatorname{Tr}\left(\rho \mathsf{E}_m\right)$ and $q_m = \operatorname{Tr}\left(\sigma \mathsf{E}_m\right)$ indicate the probabilities of obtaining a measurement outcome labelled by m. Then*

$$\mathsf{D}\left(\rho, \sigma\right) = \max_{\{\mathsf{E}_m\}} \mathsf{D}\left(\left\{p_m\right\}, \left\{q_m\right\}\right) \ ,$$

where the maximum is over all POVMs $\{\mathsf{E}_m\}$.

[1] This is also known as Kolmogorov distance.

The trace distance is a metric. If the two quantum states (probability distributions) are identical then $\mathsf{D} = 0$, and if they are orthogonal then $\mathsf{D} = 1$. The trace distance preserves the unitary transformations

$$\mathsf{D}\left(\mathbf{U}\rho\mathbf{U}^{\dagger}, \mathbf{U}\sigma\mathbf{U}^{\dagger}\right) = \mathsf{D}\left(\rho, \sigma\right) \ . \tag{2.26}$$

And the triangle inequality holds:

$$\mathsf{D}\left(\rho, \sigma\right) \leq \mathsf{D}\left(\rho, \tau\right) + \mathsf{D}\left(\tau, \sigma\right) \ . \tag{2.27}$$

The other distance measure is fidelity. Given two quantum states ρ and σ, their fidelity is defined to be

$$\mathsf{F}\left(\rho, \sigma\right) = \mathrm{Tr}\left(\sqrt{\rho^{1/2}\sigma\rho^{1/2}}\right) \ . \tag{2.28}$$

Similar to the trace distance, the fidelity preserves unitary transformation, and we can relate it to the probability distributions obtained by a measurement

$$\mathsf{F}\left(\rho, \sigma\right) = \min_{\{\mathsf{E}_m\}} \mathsf{F}\left(\{p_m\}, \{q_m\}\right) \ , \tag{2.29}$$

where the minimum is over all POVMs $\{\mathsf{E}_m\}$. The fidelity is not a metric but

$$\arccos\left(\mathsf{F}\left(\rho, \sigma\right)\right) \tag{2.30}$$

is a metric. This also satisfies the triangle inequality:

$$\arccos\left(\mathsf{F}\left(\rho, \sigma\right)\right) \leq \arccos\left(\mathsf{F}\left(\rho, \tau\right)\right) + \arccos\left(\mathsf{F}\left(\tau, \sigma\right)\right) \ . \tag{2.31}$$

The fidelity is qualitatively related to the trace distance. When two quantum states become more distinguishable, the fidelity decreases while the trace distance increases. On the other hand, when two quantum states become less distinguishable, the fidelity increases while the trace distance decreases [89]. This can be shown by

$$1 - \mathsf{F}\left(\rho, \sigma\right) \leq \mathsf{D}\left(\rho, \sigma\right) \leq \sqrt{1 - \mathsf{F}\left(\rho, \sigma\right)^2} \, . \tag{2.32}$$

In the next chapter, we explain how one can perform computation by exploiting the properties of quantum mechanics. Moreover, we discuss quantum algorithms and define a quantum adversary.

Chapter 3

Quantum Computation

Contents

In this chapter, we give an overview of quantum computation. Then a model of quantum computation is described. We also discuss the possibility of realisation of quantum computers. Moreover, we explain a number of quantum algorithms and their limited ability to solve problems. Finally we provide the definition of a quantum adversary to use in the next chapters.

3.1 Quantum Computers

The *Turing* machine [105] is a mathematical model for a 'universal' computer. According to the modern *Church-Turing thesis*, any 'reasonable' model of computation can be 'efficiently' simulated on a probabilistic Turing machine. By 'reasonable', we mean any model of computation that can be defined in a realistic physical framework. From a complexity theory perspective, an efficient simulation means that the amount of resources used by the Turing machine is polynomially bounded by the amount of resources used by the given realistic model of computation. In 1981, Feynman [60] raised a question about the possibility of simulating 'quantum physics' with a universal computer. He discussed as follows: Given an n-particle (say n qubits) quantum system that is evolving in time, then the amount of information required, in classical terms, for simulation of the quantum system grows exponentially in time. Therefore, in general, simulating the natural evolution of a quantum system on a probabilistic Turing machine involves exponential slowdown. But, that is in contrast with the notion of an efficient simulation. Feynman proposed the idea of a 'quantum computer' in order to efficiently simulate a general quantum evolution on a computing apparatus based on quantum physics.

Following the above idea, Deutsch [46] introduced a formal model for *universal quantum Turing machines*. He proposed that the quantum Turing machine might be faster than the classical Turing machine in solving certain classical problems. These are problems with classical input and output. He also proved that any given quantum machines can be simulated by universal quantum computers, subject to exponential slowdown. Later on, a separation between probabilistic classical and quantum models was shown by Bernstein and Vazirani [23]. They constructed the first universal quantum computer that could efficiently simulate a large class of quantum Turing machines with only a polynomial overhead. There are also other quantum computational models such as the *quantum circuit* model [47]. Most of the known quantum algorithms are defined in the quantum circuit model. Yao [111] showed that the two models, the quantum Turing machine and the quantum circuit, are mathematically equivalent in terms of their computing power. In this thesis we only discuss the quantum circuit model.

3.1.1 Quantum Circuit Model

A 'qubit' (see Subsection 2.3.2) is the quantum analogue of a 'bit', which is a fundamental elementary unit of classical computation. While a bit represents either 0 or 1, a qubit represents 0 and 1 simultaneously. To build a quantum computer, we need to prepare a number of qubits to operate on them. Assume we place n qubits in a quantum register, initially in state

$$|\psi_0\rangle = |0^n\rangle \ . \tag{3.1}$$

There exists an operator **U** such that if we apply it to the register only once, then we transform the state $|\psi_0\rangle$ to another quantum state $|\psi_1\rangle$ that is a superposition of all 2^n possible configurations of n qubits

$$|\psi_1\rangle = \frac{1}{\sqrt{2^n}} \sum_{b \in \{0,1\}^n} |b\rangle \ . \tag{3.2}$$

Therefore, any further operation applied to this quantum register will be applied to the all 2^n configurations in 'parallel'. Contrast this with the classical computation where, after an operation on n bits, a classical register only holds one of the 2^n configurations at a time. Hence, any further operation applied to the classical register will be applied to merely one of the 2^n configurations.

In general, during a quantum computation, a quantum system evolves in time according to the operation **U**

$$\sum_b \alpha_b |b\rangle \xrightarrow{\mathbf{U}} \sum_b \beta_b |b\rangle \ , \tag{3.3}$$

where $\sum |\alpha_b|^2 = 1$ and $\sum |\beta_b|^2 = 1$. The operation **U** is a linear transformation that maps the state on the left hand side to the state on the right hand side. As discussed in Subsection 2.3.4, the operation **U** is unitary. In fact, the operation must be unitary, otherwise we are not able to find the Hamiltonian corresponding to the closed quantum system by solving the Schrödinger equation [99]. Intuitively, quantum computation can be seen as a sequence of a number of unitary operations

$$\mathbf{U} = \mathbf{U}_1 \mathbf{U}_2 \ldots \mathbf{U}_m \ , \tag{3.4}$$

where m is polynomially bounded in n. The operators $\mathbf{U}_i$, $i \in [m]$, are elementary

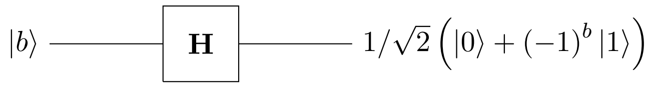

Figure 3.1: Schematic representation of **H** gate operating on a qubit in state $|b\rangle$

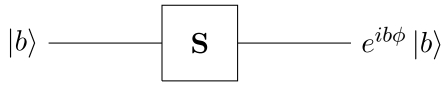

Figure 3.2: Schematic representation of **S** gate operating on a qubit in state $|b\rangle$

unitary transformations usually acting on 1, 2 or 3 qubits. They are known as *elementary quantum logic gates* [47].

We have already introduced two 1-qubit gates. Recall the beam splitter and the phase shifter in the Mach-Zehnder interferometer (see Subsection 2.3.4). They are unitary operations. The beam splitter operation is known as the *Hadamard* gate and is denoted by **H**:

$$\mathbf{H} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} . \tag{3.5}$$

The phase shifter operation is known as the *phase shift* gate, and is denoted by **S**:

$$\mathbf{S} = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix} . \tag{3.6}$$

Moreover, in general if we apply a Hadamard transformation to a state of, say, n qubits such as $|i\rangle$ where $i \in \{0,1\}^n$ then we get:

$$\mathbf{H}^{\otimes n} |i\rangle = \frac{1}{\sqrt{2^n}} \sum_{j \in \{0,1\}^n} (-1)^{i \cdot j} |j\rangle \ , \tag{3.7}$$

where $i \cdot j = \sum_{k \in [n]} i_k j_k$ denotes the inner product of the n-bit strings $i, j \in \{0,1\}^n$. Another important quantum gate is a 2-qubit gate called the *controlled-NOT* gate or '**CNOT**' for short:

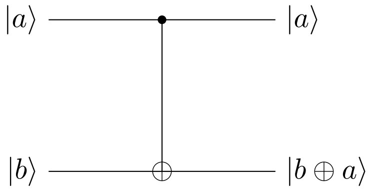

Figure 3.3: Schematic representation of **CNOT** gate operating on 2 qubits in state $|a\rangle\,|b\rangle$

$$\mathbf{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \tag{3.8}$$

Any unitary transformation can be built from a 'universal' set of quantum gates [47]. It turns out that such a universal set could consist of merely all 1-qubit gates together with the **CNOT** gate [8]. Moreover, it is proven that a set consisting of the Hadamard gate, controlled-NOT gate, and phase shift gate with a suitable phase ϕ, is universal [32]. In this thesis, by a 'universal set of gates', we mean the latter or any other possible universal set consisting of a number of arbitrary 1-qubit gates and the **CNOT** gate. Other universal sets of gates also exist [52, 80]. This description of quantum gates suffices for the purposes of this thesis, but for more details we refer to work of Nielsen and Chuang [89], and Kaye *et al.* [72].

A 'quantum circuit' [47] (also known as a 'quantum network' or an 'acyclic quantum gate array'), is a finite directed acyclic graph, that is formed by 'input nodes', 'elementary quantum gates', and 'output nodes'. There are n input nodes, which contain the input as classical bits that is essentially a basis state. The quantum gates each act on at most 2 qubits, and determine how the state of the input evolves over time. The quantum gates are chosen from a universal gate set. The outcome is achieved by measuring the output qubits in the output nodes. In general, that probabilistically yields a string of classical bits as the output of the quantum computation. The 'size' of a quantum circuit is the number of elementary quantum gates in the circuit. A quantum circuit is 'efficient' if the total size of the circuit is polynomially bounded in the number of bits of the input. The 'depth' of a quantum circuit is the maximum number of elementary quantum gates placed on any path from an input to an output, and indicates the required time to implement a quantum circuit if the quantum gates can perform in parallel. A quantum circuit is required to be ‘uniformly’ generated. This means that a classical Turing machine can efficiently output a description of the quantum circuit. For instance, a C program that can be run on a classical computer and efficiently output a description of the quantum circuit.

Note that we may use the term ‘register’ instead of ‘node’. A register consists of at least one node. If we say an n-qubit state is placed in a register, that means the register consists of n nodes.

Since quantum circuits are reversible, in order to imitate a classical computation on a quantum computer, the corresponding classical circuit must be reversible too. It turns out that any classical computation can be represented by a reversible classical circuit without losing much in efficiency [20]. Therefore, quantum computation can imitate any classical computation without losing much in efficiency, and possibly do more. For further reading on quantum gates and circuits, we refer to work of DiVincenzo [53].

3.1.2 Physical Realisation of Quantum Computers

Developing a large scale quantum computer is not an easy task [51]. To begin with, a suitable physical presentation of a collection of qubits is required. For instance, a qubit could be considered as the states of a vertical and horizontal polarised photon, or the two spin states of a spin 1/2 atom in the ground and excited states. In each of these cases one state denotes $|0\rangle$ and the other $|1\rangle$. Realisation of a suitable qubit means to accurately know a qubit’s physical parameters, such as its internal Hamiltonian, couplings to its other states, and the interaction with other qubits. We need to be able to physically prepare these qubits in their initial states. This is not a trivial task in some cases.

To perform a quantum computation, a physical implementation of a universal set of quantum gates is needed. We require these unitary transformations to be implemented in a way that each can act on a small number of qubits. This is usually done by identifying Hamiltonians which generate the unitary transformation. For instance $\mathbf{U}_j = e^{iH_j t/\hbar}$ where $j \in [n]$. During all these processes, the qubits, as a closed quantum system, must remain coherent. This means we must be able to keep the physical qubits from interacting with the environment, also known as 'quantum noise'. In practice, however, the qubits are susceptible to perturbation by quantum noise. This perturbation is called *decoherence*. Finally, to get the quantum computation outcome, we require the ability to measure specific qubits. In practice though, measurements are not 100% efficient. The state of nearby qubits or quantum noise have a negative effect on the measurement outcome.

There are different proposals for a natural presentation of a qubit. For instance, electromagnetically trapped particles such as 'trapped ions' [39, 66], and 'trapped single electrons' [85]. Others are 'molecular spins in liquids', also known as 'nuclear magnetic resonance' (NMR) [70], and 'nuclear and electron spins in silicon' [71, 86]. Moreover, there are proposals such as 'optical photon' [91], 'superconducting qubits' [102, 110, 90], and 'solid state qubits with quantum dots' [7, 81]. The problem is that none of these proposals overcame all the difficulties mentioned in the above. At least not until now.

Discussing the above proposals in detail is beyond the scope of this thesis. But for more details we refer to work of Nielsen and Chuang [89] and Chen *et al.* [35]. Here we merely point out a number of advances in the field. For example, IBM opts to develop superconducting qubits but, in general, the error rate of the qubits is too high to allow operation on them. But, IBM has been able to reduce the errors in elementary quantum computations [93]. Also recently, researchers built a superconducting multi-qubit processor with 99% reliability in performing 1-qubit and 2-qubit quantum gates [9].

There is another possible proposal known as 'adiabatic' [58, 37] quantum computing. This is mainly an approach to address the decoherence problem. It refers to an evolution in which the quantum system always remains in its instantaneous eigenstates. The ground state of a quantum system is very robust against decoherence. Hence, if one can perform adiabatic quantum computation when a system is in its ground state, then it remains in the ground state all the time and it is only the nature of the ground states which evolves into the final outcome of the computation. The D-Wave company currently builds a 512-qubit superconducting quantum computer which performs adiabatic computation to solve optimisation problems [42]. The company has been joined by Google and NASA for further developments of its quantum computer [38]. Despite this enthusiasm, D-Wave's quantum computer is the subject of much debate [101, 97]. The drawback of adiabatic quantum computing is that the lowest eigenstate of a quantum system might get very close to a higher state. Therefore, to suppress non-adiabatic transformation between them, the adiabatic evolutions have to be performed extremely slowly. This, may lead to other problems.

3.2 Quantum Algorithms

At the heart of quantum computation are quantum algorithms which enable us to harness the power of quantum computation.

Early quantum algorithms were designed after Deutsch [46] suggested that quantum computers might be faster than classical computers in solving certain problems. Deutsch and Jozsa [48] gave the first quantum algorithm that showed an exponential advantage over the best deterministic classical algorithm. Call a function $f : \{0,1\}^n \rightarrow \{0,1\}$ balanced if it has an equal number of 0 and 1 outputs. Given the promise that a function f is constant or balanced, the Deutsch and Jozsa algorithm determines whether it is constant or balanced. This was followed by the work of Brassard and Berthiaume [24], who recast this problem in complexity theoretic terms, that showed an exponential advantage over the best probabilistic classical algorithm with zero error probability. However, a probabilistic classical algorithm with exponentially small error probability could efficiently solve [104] the problems explored in [48] and [24].

Simon [104] designed an algorithm that demonstrated an exponential advantage over the best probabilistic classical algorithms. Later on, Shor [103], inspired by Simon's algorithm, discovered a quantum algorithm to solve the discrete logarithm and factoring problem in polynomial time. Shor's algorithm meant that a number of asymmetric cryptographic schemes such as RSA [94] could be broken in a reasonable amount of time using a quantum computer. This eventually led to the inception of a new research field, called *post-quantum cryptography* [22]. This field attempts to design cryptographic schemes which are secure even in the presence of quantum computers.

Grover [65] found another quantum algorithm for solving the database search problem. Grover's algorithm showed only a polynomial speed up over classical algorithms, but it is applicable to solving a wide range of problems (see Subsection 3.2.2).

The quantum algorithms we discuss here, and in fact most of the known quantum algorithms, can be described in the 'black-box model' [107, 40]. This is a model of computation where a problem is defined in terms of a black-box that can be applied. A black-box is also, equivalently, called an 'oracle' that can be queried. We use the terms 'black-box' and 'oracle' interchangeably. The only way to extract information from an oracle is to query it. That is, to supply an input and receive the corresponding output. The 'complexity' of a black-box algorithm that solves a black-box problem is the number of oracles used by the algorithm. Intuitively, an oracle is a sub-circuit that implements a function.

For the purpose of this thesis, we only explain quantum algorithms of Simon and Grover, but we refer to work of Aharonov [5] and Mosca [87] for more details of quantum algorithms.

3.2.1 Simon's Algorithm

Distinguishing between two different classes of computable functions is known to be hard in classical computation [104]. For instance, it takes a classical algorithm an exponential amount of time to determine whether a function, given as a black-box, is two-to-one or one-to-one. A classical lower bound for the number of function queries is $\Omega\left(2^{n/4}\right)$ [104]. Simon's algorithm, on the other hand, solves this problem by querying the function $O\left(n\right)$ times, where n is size of the domain. Therefore Simon's algorithm shows an exponential improvement over the best classical algorithm.

Problem 1 (Simon's Problem) *Given a function $f : \{0,1\}^n \rightarrow \{0,1\}^n$ for which there exists $s \in \{0,1\}^n$ such that for any n-bit strings i and j, $f(i) = f(j)$ if and* *only if* $i = j$ *or* $i = j \oplus s$*, find* s*.*

We now explain how Simon's algorithm solves Problem 1. We start by preparing two quantum registers in the initial state 0,

$$|\psi_0\rangle = |0^n\rangle |0^n\rangle \ . \tag{3.9}$$

Then we apply the Hadamard transformation to the first register to get:

$$|\psi_1\rangle = \frac{1}{\sqrt{2^n}} \sum_{i \in \{0,1\}^n} |i\rangle |0^n\rangle \ . \tag{3.10}$$

Now if we query function f (or equivalently, apply $\mathbf{U}_f$ to the two registers) we get:

$$|\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{i \in \{0,1\}^n} |i\rangle |x_i\rangle \ , \tag{3.11}$$

where $f(i) = x_i$. Note that the second register now stores all 2^n configurations of function $f(i)$. Applying a projective measurement, $\{(\mathbf{I} \otimes |i\rangle\langle i|)\}$, to the second register, yields:

$$|\psi_3\rangle = \frac{1}{\sqrt{2}} (|i\rangle + |i \oplus s\rangle) |x_i\rangle \ , \tag{3.12}$$

for a random $i \in \{0,1\}^n$. Now the second register stores the measurement outcome x_i while the first register collapses into a superposition of states corresponding to the measurement outcome. To find s, we only focus on the first register. Again we apply a Hadamard transformation to the first register. The result is:

$$
\begin{aligned}
|\psi_4\rangle &= \frac{1}{\sqrt{2^{n+1}}} \left(\sum_{j \in \{0,1\}^n} (-1)^{i \cdot j} |j\rangle + \sum_{j \in \{0,1\}^n} (-1)^{(i \oplus s) \cdot j} |j\rangle \right) \\
&= \frac{1}{\sqrt{2^{n+1}}} \left(\sum_{j \in \{0,1\}^n} (-1)^{i \cdot j} \left(1 + (-1)^{s \cdot j}\right) |j\rangle \right) \ .
\end{aligned}
\tag{3.13}
$$

Note that we used Equation 3.7 and the fact that $(i \oplus s) \cdot j = (i \cdot j) \oplus (s \cdot j)$. The amplitude of $|j\rangle$ in Equation 3.13 is non-zero if and only if $s \cdot j = 0 \bmod 2$. If we measure $|\psi_4\rangle$, we obtain, say, j_1. We repeat this algorithm n times to get $j_1, \ldots, j_n$. Therefore we obtain $n-1$ linear equations

$$
j_i \cdot s = 0 \mod 2 \text{ where } i \in [n] \ . \tag{3.14}
$$

By applying Gaussian elimination modulo 2, we can find a solution of the set of equations that is either 0^n or the correct s. Note that finding the solution also means determining whether the function f is two-to-one, in the case where $s \neq 0$, or one-to-one, when $s = 0$.

3.2.2 Grover's Algorithm

Assume we are given an unordered database that contains N entries. We are interested in an entry i that satisfies a number of properties. It is easy to verify whether the properties are satisfied. But it is hard to find i, if it exists. This is known as the database search problem or just the search problem. The best classical algorithm requires at least $\Omega(N/2)$ queries to solve this problem with a probability of 1/2. In a comparison, Grover's algorithm [65] solves this problem in $O\left(\sqrt{N}\right)$ queries with quadratic increase in speed.

Problem 2 (Search Problem) *Given an arbitrary function $f : \{0,1\}^n \to \{0,1\}$, find $x \in \{0,1\}^n$ such that $f(x) = 1$, otherwise output 'no solution'.*

In order to solve Problem 2, Grover's algorithm prepares a quantum superposition of states that consists of all 2^n possible configurations of f each with equal amplitude $1/\sqrt{2^n}$. The configurations are checked to see if they satisfy $f(x) = 1$, and their amplitudes are manipulated to produce the correct configuration with probability at least 1/2. The key to Grover's algorithm is the selective shifting of the phase of those configurations that satisfy the desired properties. This is known as *amplitude amplification*. Note that manipulating the phase of a state does not change the probability of being in that state. For instance, if we apply the unitary transformation $\mathbf{S}$, where $\phi = \pi$, to a state (see Equation 3.6) then we put a '-1' in front of $|1\rangle$ in the state. In Subsection 2.3.3, we mentioned that probability disregards the sign of the amplitude. Therefore the overall measurement probability distribution is intact. The phase of a state has no analogue in classical computation, and we can exploit it to our advantage. If we can manipulate the phase of each configuration in a quantum state in a way such that it induces destructive interference on 'bad' configurations and constructive interference on 'good' configurations, then we can increase the probability of finding the solutions.

We use s to denote the number of solutions for $f(x) = 1$. We define the oracle $\mathbf{O}$ as

$$|x\rangle \xrightarrow{\mathbf{O}} (-1)^{f(x)} |x\rangle \ . \tag{3.15}$$

Moreover, we define the unitary transformation $\mathbf{U}_g$ which manipulates the phase of $|0^n\rangle$ by placing '-1' in front of it

$$|x\rangle \xrightarrow{\mathbf{U}_g} (-1)^{\delta_{x0}} |x\rangle \ . \tag{3.16}$$

We use $\mathbf{G}$ to denote the *Grover iteration* such that $\mathbf{G} = -\mathbf{H}^{\otimes n}\mathbf{U}_g\mathbf{H}^{\otimes n}\mathbf{O}$.

Now we explain Grover's algorithm [65]. We prepare n qubits in the initial state $|0^n\rangle$. Then we apply the Hadamard transformation to it to obtain a uniform superposition of all x:

$$|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle \ . \tag{3.17}$$

Now we define the following 'good' and 'bad' states, denoted by $|g\rangle$ and $|b\rangle$ respectively, as:

$$|g\rangle = \frac{1}{\sqrt{s}} \sum_{f(x)=1} |x\rangle \text{ and } |b\rangle = \frac{1}{\sqrt{2^n - s}} \sum_{f(x)\neq 1} |x\rangle \ . \tag{3.18}$$

Therefore we can rewrite $|\psi\rangle$ as follows,

$$|\psi\rangle = \sqrt{\frac{2^n - s}{2^n}}\, |b\rangle + \sqrt{\frac{s}{2^n}}\, |g\rangle \ . \tag{3.19}$$

We apply $\mathbf{G}$ to $|\psi\rangle$ a number of times. The Grover iterate performs two reflections. The oracle $\mathbf{O}$ performs a reflection through $|b\rangle$ in the plane defined by $|b\rangle$ and $|g\rangle$, and

$$-\mathbf{H}^{\otimes n}\mathbf{U}_g\mathbf{H}^{\otimes n} = \mathbf{H}^{\otimes n}\left(2\,|0^n\rangle\langle 0^n| - \mathbf{I}\right)\mathbf{H}^{\otimes n} = 2\,|\psi\rangle\langle\psi| - \mathbf{I} \tag{3.20}$$

is a reflection through $|\psi\rangle$ in the plane defined by $|b\rangle$ and $|g\rangle$. Now if we measure the final state we obtain a solution.

Note that each $\mathbf{G}$ is considered as one oracle query. The number of required oracle queries depends on the number of solutions s. To see how many oracle queries are required, we rewrite the state $|\psi\rangle$ again

$$|\psi\rangle = \sin\theta\,|g\rangle + \cos\theta\,|b\rangle \ , \tag{3.21}$$

where $\theta = \arcsin\left(\sqrt{s/2^n}\right)$. After q oracle queries, the two reflections performed by $\mathbf{G}$ transform the state $|\psi\rangle$ to

$$\left|\psi'\right\rangle = \sin\left(2q+1\right)\theta\left|g\right\rangle + \cos\left(2q+1\right)\theta\left|b\right\rangle \ . \tag{3.22}$$

According to the state $|\psi'\rangle$, the probability of obtaining a solution after measurement is $p = \sin\left(\left(2q+1\right)\theta\right)^2$. To increase this probability we need to pick q such that $p = \sin\left(\left(2q+1\right)\theta\right)^2 = 1$. Note that if we choose q' such that

$$q' = \frac{\pi}{4\theta} - \frac{1}{2} \tag{3.23}$$

then $\left(2q'+1\right)\theta = \pi/2$ hence $p = 1$. Therefore, we need a discrete number of queries q. Assuming that $|q - q'| \leq 1/2$, and the number of solutions is $s \leq 2^n/2$, then [31]

$$p = \sin\left(\left(2q+1\right)\theta\right)^2 = \left(\sin\theta\right)^2 = \frac{s}{2^n} \ . \tag{3.24}$$

Because $\arcsin\theta \geq \theta$, then the number of oracle queries is [31]

$$q \leq \frac{\pi}{4\theta} \leq \frac{\pi}{4}\sqrt{\frac{2^n}{s}} \ . \tag{3.25}$$

Grover's algorithm only has a quadratic advantage over the best classical algorithm. Bennett *et al.* [21] show that the Grover bound is the best one could do to solve a search problem. Boyer *et al.* [31] give tight bounds on any possible quantum search algorithm and Zalka [113] shows that Equation 3.25 is optimal. Grover's algorithm has many application. For instance, it can be used to find the shared secret key of a symmetric encryption such as AES [3]. Moreover, the work of Brassard *et al.* [34, 33], based on Grover's algorithm, gives a quantum algorithm to find collisions in an arbitrary r-to-one function after $O\left(\sqrt[3]{N/r}\right)$ oracle query. The latter can be used to find collisions in hash functions.

3.3 Limitations of Quantum Computers

So far in this chapter, we have shown the power of quantum computation over classical computation. We mentioned problems that quantum algorithms can solve exponentially faster than classical algorithms, as well as problems that quantum algorithms can solve merely polynomially faster. But as it turns out, there are problems that quantum algorithms cannot solve efficiently [108]. They are just better than classical algorithms at solving certain types of problem.

In complexity theory, classes such as **P** and **PSPACE** refer to the set of decision problems that can be solved in polynomial time, and in polynomial space, using a deterministic Turing machine, respectively. The hardest problems in **NP** are in what is known as the **NP**-complete class. The problems in **NP** can be reduced to problems in **NP**-complete. The general belief is that $\mathbf{NP} \neq \mathbf{P}$, this has not been proven. If an **NP**-complete problem is found to be in **P**, then it implies that all **NP** problems are in **P**, which would mean $\mathbf{NP} = \mathbf{P}$.

Another important complexity class is one that consists of problems which can be solved in polynomial time, with bounded error probability, by a probabilistic Turing machine. These are called **BPP**. This definition explicitly allows for a small probability that the solution is wrong. It is conjectured that $\mathbf{P} = \mathbf{BPP}$ [68]. A quantum extension of **BPP** is called **BQP** [23]. The latter is the class of problems that can be solved in polynomial time, with bounded error probability, by a probabilistic quantum Turing machine. Aaronson [4] suggests that $\mathbf{P} \subset \mathbf{BQP}$. This means that quantum algorithms could solve, in polynomial time, problems that cannot be solved efficiently by classical algorithms.

3.4 Quantum Adversary

A *quantum adversary* is a quantum algorithm that runs on an ideal quantum computer. We use capital letters such as $\mathcal{A}$ to denote a quantum adversary. We define a general quantum adversary, that without loss of generality, complies with a given security definition. In Section 2.2, we discussed that a security definition is usually expressed as an ‘experiment’ conducted by a ‘challenger’. A quantum adversary plays the experiment with regards to a cryptographic scheme. To do so, the quantum adversary maintains a number of registers. Specifically, registers for input and output, a register for querying its ‘oracles’ that are maintained by the challenger, a register for storing its internal state between each oracle query, and a register for classical communication. Formally, all of these registers are quantum registers. The provided oracles could be classically or quantumly queried, depending on the security definition. Generally we assume that oracle queries made by a quantum adversary are quantum states. If a security definition merely allows classical access to the provided oracles, then the quantum queries are measured by the oracle before being answered. Note that measuring a quantum state yields a random classical string. On the other hand, if quantum queries are allowed, then the oracle applies a unitary transformation to the quantum queries.

Often at the beginning of an experiment, a number of variables with their initial values are introduced. A quantum adversary might use these variables as its input. Or it might just simply place $|0^n\rangle$, or any other string, in its input register. Then the quantum adversary queries its oracle. The oracle queries are placed in a quantum register prepared by the quantum adversary. The quantum register is shared between the quantum adversary and the oracle. The oracle prepares its response by applying a unitary transformation to the quantum register. If the oracle is merely classically accessible, it first measures the quantum register. Then the oracle writes its classical response, based on the measurement outcome, to the quantum register. Note that this is the same as, say, the quantum register being sent back and forth between the quantum adversary and the oracle. The quantum adversary performs a number of unitary transformations on its registers between each oracle query. It finally produces an output.

Definition 1 [**Quantum Adversary**] A quantum adversary $\mathcal{A}$ maintains a number of quantum registers. That is, specifically, there are two quantum registers *inp* and *out* for inputs and outputs respectively, a quantum register Q for the purpose of making quantum oracle queries, a quantum register S for storing its internal state between each oracle query, and a quantum register R for classical communication with the environment (such as specifying the type of a query, the length n, and any other query parameters). The quantum adversary always begins by preparing some initial quantum state in its input register. Then, $\mathcal{A}$ prepares an n-qubit quantum query in the register Q. The oracle takes the form of a unitary transformation, $\mathbf{U}_f$, defined by its action on the first $2n$ qubits of the register Q such that

$$\mathbf{U}_f \left|x, y\right\rangle = \left|x, y \oplus f\left(x\right)\right\rangle \ , \tag{3.26}$$

where x and y are n-bit strings. This defines the action of $\mathbf{U}_f$ for arbitrary quantum states in the register Q. This includes superposition of states, mixed states, and states entangled with state of the register S. If the oracle merely accepts classical queries, then it measures the first n qubits in the register Q to obtain a n-bit string x. The oracle then replaces the first n qubits in the register Q with its response $f\left(x\right)$. The quantum adversary's ability to store its internal state in the register S means that it can make interactive oracle queries. Formally, the action of the quantum adversary is a quantum operation. That is, a completely positive map, acting on its registers. Finally, $\mathcal{A}$ outputs by measuring the *out* register. We will quantify resources available to the adversary as follows. The *running time* of an adversary $\mathcal{A}$ is the time, in seconds, that elapses until $\mathcal{A}$ writes its final output and halts, including any initialisation steps. In addition to the *number of oracle queries* made by $\mathcal{A}$, we specify the *total size*, measured in number of qubits, of all oracle queries. In some cases, we will also quantify the *size of the classical output* of $\mathcal{A}$. ▮

The standard formal model for a quantum computer is the quantum circuit model described in Subsection 3.1.1. The above definition of a quantum adversary, however, deliberately avoids referring to a particular quantum computing model. The concrete security reductions given in this thesis are black-box reductions. They assume the existence of a specific quantum adversary attacking one scheme and, based on this, construct a specific quantum adversary attacking another scheme. For the reductions, the hardware realisation of the adversary or the quantum computing model on which it is based are irrelevant. The resources given in the definition above, which are those that play a role in the reductions, do not depend on the details of the adversary or the computational model. This should be regarded as a strength of the concrete security approach, since it is unknown what form a future quantum computer will take [97].

To fully characterise the resources used by a quantum adversary, one needs to quantify, in addition to the running time, the size of the adversary. In principle there are many ways of doing this. For instance, one could limit the physical volume, the size of the available Hilbert space, and the size of the classical memory available to the quantum adversary. Corresponding resource parameters could easily be added to our reduction theorems. But since they would simply appear unchanged on both sides of the equations, they would not add anything and have therefore been omitted in the above definition of a quantum adversary.

The theorems proved in this thesis are reductions of the following form. Given a specific adversary A that attacks some scheme S_A using certain physical resources, an adversary B is constructed that attacks another scheme S_B using broadly similar resources. In order to draw conclusions from such a reduction in the concrete security framework adopted here, one has to make (often heuristic) assumptions about the security of scheme S_B. For example, one might assume that no quantum adversary which can be physically realised in the next 30 years, and which runs for at most t seconds and makes at most q oracle queries, can break S_B with probability larger than ϵ, where t, q are suitably large numbers and ϵ is a suitably small number. This allows one to draw similarly concrete conclusions about the security of scheme S_A.

Now we give a couple of examples of the interaction between a quantum adversary and its oracles. When the register Q contains a quantum superposition of states $\sum_x \alpha_x |x, 0\rangle$, then the oracle's action, $\mathbf{U}_f$, on the register Q, is given by

$$\mathbf{U}_f \sum_x \alpha_x |x, 0\rangle = \sum_x \alpha_x |x, f(x)\rangle . \tag{3.27}$$

In a classical n-bit randomised encryption query, the oracle measures the first n qubits in the register Q to obtain a bit string x. It then replaces the first $(n + n_r)$ qubits of Q with the oracle response $(f(x, r), r)$, where n_r is the length of the random string r. In an n-bit randomised quantum encryption query, the oracle takes the form of a unitary operation, $\mathbf{U}_{f(\cdot,r)}$, defined by its action on the first $(2n + n_r)$ qubits in the register Q,

$$\mathbf{U}_{f(\cdot,r)} \left|x, y, z\right\rangle = \left|x, y \oplus f\left(x, r\right), z \oplus r\right\rangle \ , \tag{3.28}$$

where x and y are n-bit strings and z is a n_r-bit string. This defines the action of $\mathbf{U}_{f(\cdot,r)}$ for arbitrary states in the register Q, including superposition of states, mixed states, or states entangled with state of the register S. For instance, the action of $\mathbf{U}_{f(\cdot,r)}$ on the superposition state $\left|\psi\right\rangle = \sum_x \alpha_x \left|x, 0, 0\right\rangle$ is given by

$$\mathbf{U}_{f(\cdot,r)} \sum_x \alpha_x \left|x, 0, 0\right\rangle = \sum_x \alpha_x \left|x, f\left(x, r\right), r\right\rangle \ . \tag{3.29}$$

The resources required to apply the unitary $\mathbf{U}_{f(\cdot,r)}$ to a quantum register are independent of the initial state $\left|\psi\right\rangle$ of the register. Applying a unitary operator can be thought of as a single physical operation, for which the number of terms in the superposition state $\left|\psi\right\rangle$ is irrelevant. Since the encryption oracle does not ‘know’ whether it acts on a superposition or on a single basis state, we have assumed above that the random string r required for the randomised encryption is chosen exactly once every time $\mathbf{U}_{f(\cdot,r)}$ is applied, i.e., r is the same for all terms in the sum in Equation 3.29.

In the case where f is a permutation, an alternative way to define a quantum oracle query would be through a unitary $\mathbf{U}'_{f(\cdot,r)}$ acting on $(n + n_r)$ bits. This definition would be

$$\mathbf{U}'_{f(\cdot,r)} \left|x, z\right\rangle = \left|f\left(x, r\right), z \oplus r\right\rangle \ . \tag{3.30}$$

3.4.1 Quantum Computation vs. Quantum Superposition Attack

In the previous section, we discussed that the quantum adversary is given either classical or quantum access to its oracles with regards to the security definition. Assume the quantum adversary with classical access to its oracle. By making an oracle query, the quantum adversary can evaluate one instance of the oracle at a time. It then can take advantage of the quantum computation power between each oracle query to attack a cryptosystem. To describe this approach we use the term *quantum computation attack*.

For example, consider the case where the quantum adversary attacks a simple symmetric encryption scheme such as Even-Mansour [57] with a key size n. We explain this in details in Section 5.1. The quantum adversary queries the oracle on a number of messages to get their corresponding ciphertexts. After each oracle query, the quantum adversary possesses a message/ciphertext pair. Given a message and its corresponding ciphertext, the key can be recovered using Grover's algorithm [65] after $2^{n/2}$ quantum operations. This is in comparison with the classical 'exhaustive key search' attack that requires 2^n classical operations. Although the key can be recovered faster than a classical attack in this manner, it still takes the quantum adversary an exponential time to recover any key. Hence the Even-Mansour scheme is considered secure against a quantum computation attack. Another example is to attack an asymmetric encryption scheme such as RSA [94]. Given the public key and a message/ciphertext pair, the quantum adversary can recover the private key in polynomial time using Shor's algorithm [103].

Now assume that the quantum adversary is given quantum access to its oracle. The quantum adversary can make a quantum superposition query. For example, if the query is a superposition of all messages then the oracle response contains all the corresponding ciphertexts. This gives the quantum adversary an additional power besides its quantum computation power to attack a cryptosystem. We use the term *quantum superposition attack* to describe this property. For example, in Chapter 5, we show that the Even-Mansour scheme is insecure against the quantum superposition attack.

In general, a quantum computation attack against modern cryptosystems is the subject of the field of post-quantum cryptography [22]. Here, the assumption is that the honest parties use classical computation and communication, while the adversary might be in possession of quantum computers. The quantum adversary is then able to launch a quantum computation attack against the cryptosystems used by the honest parties. In contrast, the quantum superposition attack is beyond the field of post-quantum cryptography. A quantum superposition attack is possible when the honest parties use quantum computation. Different attacks are possible depending on whether the honest parties use classical or quantum communication. We have already discussed that any classical algorithm can be run on a quantum computer (see Subsection 3.1.1).

Note that the quantum superposition attack is more powerful that the quantum computation attack. Moreover, a security definition that allows quantum superposition queries is stronger than a security definition that is restricted to classical queries.

In the next chapter, we discuss the security of a number of symmetric schemes and the achievability of a number of security definitions against quantum computation attacks.

Chapter 4

Symmetric Encryption

Contents

This chapter gives an overview of symmetric encryption. We mostly focus on primitives and security notions. While discussing the latter, we introduce a new indistinguishability-based security notion. Then we explain relations among all the given security notions in this chapter. Moreover, we discuss the security of symmetric encryption schemes against the quantum computation attack. We finish this chapter by discussing a security notion for achieving integrity.

4.1 Cryptographic Primitives

Cryptographic schemes are built on a number of smaller and simpler schemes called *primitives*. The cryptographic schemes are designed to achieve a goal. For instance, encrypting messages. Primitives can also be considered as simple cryptographic schemes, but they merely provide some sort of ‘hardness’ or ‘security’ properties that must be properly used to design more complex schemes to achieve a goal.

Cryptographic primitives can be drawn from two main groups: ‘symmetric’ and ‘asymmetric’. Block ciphers such as DES or AES can be used as symmetric primitives, and RSA, for instance, can be used as an asymmetric primitive. In this thesis we solely focus on symmetric primitives and symmetric cryptography.

4.1.1 Block Ciphers

At the centre of symmetric cryptography are *block ciphers* that play a very important role in modern cryptography. Block ciphers are one of the most widely used primitives. They are simple and adaptable with efficient implementation. Block ciphers are also essential to many cryptographic schemes that are used in practice. Widely used block ciphers such as DES and AES have been subject of intensive cryptanalysis, and to this day no major security flaw has been found in their design. This is another reason to consider block ciphers as reliable primitives.

A block cipher is a function $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ that takes a k-bit string and an n-bit string as input, and returns an n-bit string as its output. The variables k and n denote the *key length* and the *block length* respectively. They are parameters associated with the block cipher, and vary according to its design. For each $K \in \{0,1\}^k$ we denote $E_K : \{0,1\}^n \to \{0,1\}^n$ as the function defined by $E_K(x) = E(K,x)$. A block cipher can be considered as a family of functions where each key identifies a function. In general, for any block cipher E and any key K, the function E_K is required to be a permutation on $\{0,1\}^n$. This implies the existence of an inverse function E_K^{-1} for E_K, with $E_K^{-1}(E_K(x)) = x$.

The popular block ciphers are the ‘Data Encryption Standard’ (DES), triple DES (3DES), and the ‘Advanced Encryption Standard’ (AES). In the early 1970s, IBM designed DES. Later in 1976, DES was standardised by the United States National Bureau of Standards (now known as the National Institute of Standards and Technology, NIST) [1]. DES is a 64-bit block cipher designed based on a ‘Feistel cipher’ structure [59], with a 56-bit key. Its relatively short key size left DES vulnerable to key recovery attacks such as ‘brute-force’ attacks. In 1998 DES was broken. The Electronic Frontier Foundation built a specific machine with a cost of less than $250,000, and recovered a DES key in less than 3 days. A simple solution to address such attacks is to increase the size of the key. Triple DES (3DES) provides a relatively simple method of increasing the key size by using DES three times in a form of ‘encrypt-decrypt-encrypt’, with either two or three different keys. In 1985, 3DES was standardised, and it became part of the Data Encryption Standard in 1999.

Daemen and Rijmen designed the block cipher ‘Rijndael’ as part of a proposal submitted to NIST during the competition to find a successor for DES in 1997. The Rijndael block cipher is based on a ‘substitution-permutation network’, and has a 128-bit block size but three different key lengths of 128 bits, 192 bits, and 256 bits. It won the NIST competition that required the winner to provide at least the same level of security as 3DES but be substantially more efficient. In 2001, NIST standardised the Rijndael block cipher under the name of ‘Advanced Encryption Standard’ (AES) [3].

4.1.2 Pseudorandom Functions and Permutations

This subsection, as well as Subsections 4.2.1 and 4.2.2, contains background material from modern cryptography. These subsections do not refer to the quantum adversary defined in Chapter 3, but to a classical adversary definition based on a Turing machine model. Here we measure running time in discrete steps rather than seconds. We follow Bellare *et al.* [13] in including in the running time, the space required to store the program that describes the adversary. This prevents the adversary, e.g., from embedding arbitrary large look-up tables in the program. Measuring space and running time in the same units makes sense because the time required to read in a program is generally proportional to the program length. In this way the number of Turing machines bounded by a given running time is finite, so that taking a maximum over all classical adversaries bounded such is well defined.

Normally in the context of provable security, block ciphers are modelled as *pseudorandom functions* or *pseudorandom permutations* [82, 83, 15]. A pseudorandom function is a family of functions such that a function chosen uniformly at random from the family of functions is indistinguishable from a function chosen uniformly at random from the set of all functions [62, 63]. This is expressed via the ability of an adversary to distinguish between two experiments. We are interested in the probability of the

```
Experiment $\mathbf{Exp}_F^{prf-1}(A)$
    $K \xleftarrow{\$} \mathcal{K}$
    $b \leftarrow A^{F_K(\cdot)}$
    return $b$
```

```
Experiment $\mathbf{Exp}_F^{prf-0}(A)$
    $f \xleftarrow{\$} \mathsf{Func}(\mathcal{X},\mathcal{Y})$
    $b \leftarrow A^{f(\cdot)}$
    return $b$
```

Figure 4.1: The PRF definition

adversary doing so. We now provide a formal definition for pseudorandom functions.

Definition 2 [**Pseudorandom Functions (PRF)**] Let $F : \mathcal{K} \times \mathcal{X} \to \mathcal{Y}$ be a family of functions identified by the set $\mathcal{K}$. Define two experiments $\mathbf{Exp}_F^{prf-0}$ and $\mathbf{Exp}_F^{prf-1}$ for an adversary A as depicted in Figure 4.1. The adversary A has access to an oracle, and returns a bit as its output. The advantage of A is defined as

$$\mathbf{Adv}_F^{prf}(A) = \Pr\left[\mathbf{Exp}_F^{prf-1}(A) = 1\right] - \Pr\left[\mathbf{Exp}_F^{prf-0}(A) = 1\right] .$$

The advantage of the function family is given by

$$\mathbf{Adv}_F^{prf}(t, q) = \max_A \left\{\mathbf{Adv}_F^{prf}(A)\right\} , \tag{4.1}$$

for any integers t, q. Here the maximum is over all adversaries A with a running time of at most t, making at most q oracle queries. ∎

A low value of $\mathbf{Adv}_F^{prf}(t, q)$ indicates that F is a secure PRF for reasonable values of t and q.

In the previous subsection we explained that block ciphers can be regarded as a family of permutations. Therefore we can also model block ciphers as a family of pseudorandom permutations. They can be described in a similar way.

Definition 3 [**Pseudorandom Permutations (PRP)**] Let $F : \mathcal{K} \times \mathcal{X} \to \mathcal{X}$ be a family of permutations identified by the set $\mathcal{K}$. Define two experiments $\mathbf{Exp}_F^{prp-0}$

```
Experiment $\mathbf{Exp}_F^{prp-1}(A)$
    $K \leftarrow\!\$\, \mathcal{K}$
    $b \leftarrow A^{F_K(\cdot)}$
    return $b$
```

```
Experiment $\mathbf{Exp}_F^{prp-0}(A)$
    $\Pi \leftarrow\!\$\, \mathsf{Perm}(\mathcal{X})$
    $b \leftarrow A^{\Pi(\cdot)}$
    return $b$
```

Figure 4.2: The PRP definition

and $\mathbf{Exp}_F^{prp-1}$ for an adversary A as depicted in Figure 4.2. The adversary A has access to an oracle, and returns a bit as its output. The advantage of A is defined as

$$\mathbf{Adv}_F^{prp}(A) = \Pr\left[\mathbf{Exp}_F^{prp-1}(A) = 1\right] - \Pr\left[\mathbf{Exp}_F^{prp-0}(A) = 1\right] \; .$$

The advantage of the function family is given by

$$\mathbf{Adv}_F^{prp}(t, q) = \max_A \left\{\mathbf{Adv}_F^{prp}(A)\right\} \; ,$$

for any integers t, q. Here the maximum is over all adversaries A with a running time of at most t, making at most q oracle queries. ∎

A low value of $\mathbf{Adv}_F^{prp}(t, q)$ indicates that F is a secure PRP for reasonable values of t and q.

Block cipher constructions can be modelled as either PRFs or PRPs. Although PRPs better model a block cipher, analysis of a block cipher construction is sometimes easier if one assumes the underlying primitives are PRFs. The following lemma proves that the prf-advantage and the prp-advantage of a block cipher are always close to the amount given by the 'birthday attack'.

Result 2 (PRP/PRF Switching Lemma ([16] Proposition 2.5**))** *Let $F : \mathcal{K} \times \mathcal{X} \to \mathcal{X}$ be a family of functions identified by the set $\mathcal{K}$. For any adversary A that makes at most q oracle queries, then*

$$\left|\mathbf{Adv}_F^{prf}(t,q) - \mathbf{Adv}_F^{prp}(t,q)\right| \leq \frac{q(q-1)}{2|\mathcal{X}|} .$$

So far, the definitions given in this subsection merely consider classical adversaries. We now define *quantum pseudorandom functions* (QPRF). This is analogous to the definition of PRFs except that now a quantum adversary is given quantum superposition access to its oracle. Therefore, the quantum adversary can make quantum superposition queries, and the oracle responds to each query by applying a unitary transformation to the adversary's quantum register. The unitary transformation depends on the experiment that the quantum adversary is playing.

Note that Zhandry [114] shows how to construct a QPRF assuming that one-way functions exist. However, Zhandry's work is in the asymptotic setting, as opposed to concrete setting which applies in this thesis. From a concrete perspective a QPRF is simply a function family together with the definition of an advantage against a quantum adversary. Here is the formal definition:

Definition 4 [**Quantum Pseudorandom Functions (QPRF)**] Let $F : \mathcal{K}\times\mathcal{X} \to \mathcal{Y}$ be a family of functions identified by the set $\mathcal{K}$. Define two experiments $\mathbf{Exp}_F^{qprf-0}$ and $\mathbf{Exp}_F^{qprf-1}$ for a quantum adversary $\mathcal{A}$ as depicted in Figure 4.3. The adversary $\mathcal{A}$ has quantum superposition access to an oracle, and returns a bit as its output. The oracle responds to each query by applying a unitary transformation to the first $2n$ qubits of the adversary's quantum register, where n is the length of each query. The advantage of $\mathcal{A}$ is defined as

$$\mathbf{Adv}_F^{qprf}(\mathcal{A}) = \Pr\left[\mathbf{Exp}_F^{qprf-1}(\mathcal{A}) = 1\right] - \Pr\left[\mathbf{Exp}_F^{qprf-0}(\mathcal{A}) = 1\right] .$$

This advantage refers to a specific quantum adversary using resources as discussed in Section 3.4. These include the running time t, and the number of queries q. ∎

Notice that in the above we do not provide a definition of the advantage of the function family similar to Equation 4.1. The reason is that, the maximum of the

```
Experiment $\mathbf{Exp}_F^{qprf-1}(\mathcal{A})$
    $K \leftarrow\!\$\, \mathcal{K}$
    $b \leftarrow \mathcal{A}^{F_K(\cdot)}$
    return $b$
```

```
Experiment $\mathbf{Exp}_F^{qprf-0}(\mathcal{A})$
    $f \leftarrow\!\$\, \mathsf{Func}(\mathcal{X}, \mathcal{Y})$
    $b \leftarrow \mathcal{A}^{f(\cdot)}$
    return $b$
```

Figure 4.3: The QPRF definition

advantage over all adversaries limited by a set of given resources is only well defined with respect to a precise model of computation. But as we explained in Section 3.4, the concrete security reductions in this thesis do not require the specification of a formal quantum computing model. Such a specification might even limit the generality of our reductions unnecessarily.

In Chapter 6 we give constructions based on a QPRF. For these constructions to be secure, we need to assume that there exists a function family F such that its QPRF-advantage is very small for any quantum adversary using resources that are available now or might become available in the foreseeable future. Such function families exist in the form of standard block ciphers, for instance AES-256. The best currently known quantum attack against AES-256 is based on Grover's search algorithm [65], which requires of the order of 2^{128} queries to succeed with high probability. The security of the schemes discussed in Chapter 6 thus depends on the heuristic assumption that AES-256 or similar block ciphers cannot be broken by a quantum computer using realistic resources.

4.2 Encryption Schemes

Symmetric encryption provides 'privacy' for two parties that share a secret key. A symmetric encryption scheme consists of three algorithms: *key generation*, *encryption*, and *decryption*. A key generation algorithm produces a key that two parties need to share prior to any communication. The key is then used by an encryption algorithm that specifies how to produce the ciphertext from a plaintext. The ciphertext is transmitted between the parties. Then a decryption algorithm specifies how to recover the plaintext from the ciphertext by using the key.

We denote a symmetric encryption scheme by $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$. The key generation algorithm is denoted by $\mathcal{K}$. This is a randomised algorithm that takes no input, and returns a key K. The key is chosen uniformly at random from a set of keys and is usually a random bit string with an arbitrary size. We also often use the same notation, $\mathcal{K}$, with regards to the set of keys representing the key space. When the key K is generated, it needs to be securely exchanged between two parties. How the two parties securely exchange the key is beyond the scope of this thesis. However, we refer to the work of Bellare and Rogaway [18, 19] for more details on secure ways to exchange a key. Here we assume the two parties are in possession of the secret key K.

A message space $\mathcal{M} \subset \{0,1\}^*$ and a ciphertext space $\mathcal{C} \subset \{0,1\}^*$ are associated with the symmetric encryption scheme $\mathcal{SE}$. The encryption algorithm is denoted by $\mathcal{E}$. It may be either randomised or stateful. If the encryption algorithm $\mathcal{E}$ is stateful, then the key generation algorithm $\mathcal{K}$ outputs the initial encryption and decryption states, $\varrho_0 \in \Sigma$ and $\varsigma_0 \in \Sigma$ respectively, alongside the key K. The randomised encryption $\mathcal{E} : \mathcal{K} \times \mathcal{M} \to \mathcal{C}$ takes the key $K \in \mathcal{K}$ and a message $m \in \mathcal{M}$ as input, and returns a ciphertext $c \in \mathcal{C}$. Note that the randomised $\mathcal{E}$ uses fresh coins each time it is invoked. Therefore, invoking $\mathcal{E}$ on the same inputs twice may not yield the same results. The stateful encryption $\mathcal{E} : \mathcal{K} \times \mathcal{M} \times \Sigma \to \mathcal{C} \times \Sigma$ takes the key $K \in \mathcal{K}$, a message $m \in \mathcal{M}$, and the current encryption state $\varrho \in \Sigma$ as input, then returns a ciphertext $c \in \mathcal{C}$ and updates the encryption state.

The deterministic decryption algorithm is denoted by $\mathcal{D}$. The decryption algorithm $\mathcal{D} : \mathcal{K} \times \mathcal{C} \to \mathcal{M} \cup \{\perp\}$ takes the key $K \in \mathcal{K}$ and a ciphertext $c \in \mathcal{C}$ as input, and returns either the corresponding message $m \in \mathcal{M}$ or the symbol $\perp$ meaning the ciphertext is invalid. If the encryption scheme is stateful then the decryption algorithm $\mathcal{D} : \mathcal{K} \times \mathcal{C} \times \Sigma \to (\mathcal{M} \cup \{\perp\}) \times \Sigma$ takes the key $K \in \mathcal{K}$, a ciphertext $c \in \mathcal{C}$, and the current state $\varsigma \in \Sigma$ as input, and returns either the corresponding message $m \in \mathcal{M}$ or the symbol $\perp$, and updates the decryption state.

In this thesis we always consider a symmetric encryption scheme to be randomised, unless otherwise stated. For any key $K \in \mathcal{K}$, any sequence of messages $m_i \in \mathcal{M}$, $i \in [q]$, and any sequence of ciphertexts $c_i \leftarrow \mathcal{E}_K(m_i)$, we expect that $m'_i \leftarrow \mathcal{D}_K(c_i)$ where $m_i = m'_i$ with probability 1. In case of the stateful encryption scheme, for any key $K \in \mathcal{K}$ and initial states $\varrho_0, \varsigma_0 \in \Sigma$, any sequence of messages $m_i \in \mathcal{M}$, $i \in [q]$, and any sequence of ciphertexts

$$(c_i, \varrho_i) \leftarrow \mathcal{E}_K (m_i, \varrho_{i-1}) \ , \ i \in [q] \tag{4.2}$$

we expect that

$$\left(m'_i, \varsigma_i\right) \leftarrow \mathcal{D}_K (c_i, \varsigma_{i-1}) \tag{4.3}$$

where $m_i = m'_i$ with probability 1.

4.2.1 Notions of Confidentiality

The security of a symmetric encryption scheme is examined against an adversary that is not in possession of the secret key but has some prior information about the plaintext. For example, the adversary might know the length of the plaintext, or that it is an English word. The adversary also gets to see the ciphertext that is transmitted between the honest parties. This information must not enable the adversary to recover the secret key or gain any partial information about the plaintext. For example, a (stateless) deterministic encryption scheme is considered insecure in this sense because when an adversary observes two identical ciphertexts, it can conclude that they both correspond to the same plaintext. Therefore it obtains partial information about the plaintext without actually knowing the plaintext. This is not the case of course, if the symmetric encryption scheme is randomised or stateful. However, it is still possible that even a randomised or stateful symmetric encryption scheme, due to its design for instance, leaks information about the plaintext.

Goldwasser and Micali [64] were the first to formally model the security of encryption schemes. They introduced two notions of security, called *semantic security* and *polynomial security*, and proved them to be equivalent. Initially, semantic security was defined primarily for asymmetric encryption schemes in an asymptotic framework. If a given encryption scheme is semantically secure, then an adversary should be unable to obtain any partial information about the plaintext from the ciphertext. In practice, semantic security is what we desire. But in theory, semantic security is complex and difficult to work with. However, there is an *Indistinguishability*-based definition that is easier to work with. These two definitions are equivalent. This means that we can analyse the security of a scheme in the indistinguishability model while being convinced that the security properties we obtain are those that we expect from semantic security.

Bellare *et al.* [12] introduced several *Indistinguishability*-based (IND) security models for symmetric encryption schemes in a concrete framework. They presented two new indistinguishability notions called *Left-or-Right* (LoR) and *Real-or-Random* (RoR). The authors also gave an adaptation of semantic security notion (SEM), and an adaptation of polynomial security, called *Find-then-Guess* (FtG) in an indistinguishability-based security model. The four notions given by the authors consider two different types of attack, called *chosen plaintext attack* (CPA) and *chosen ciphertext attack* (CCA). In the former, an adversary is given an encryption oracle, while in the latter, an adversary is given both an encryption oracle and a decryption oracle. The adversary can query the encryption oracle on any plaintext to obtain a corresponding ciphertext. Moreover, in CCA model the adversary can query the decryption oracle, to obtain the corresponding plaintext.

In a model based on indistinguishability, an adversary is required to distinguish between the encryptions of two different plaintexts. That is, the adversary merely needs to find a bit corresponding to each plaintext rather than recovering the whole plaintext.

We explain the four notions LoR, RoR, FtG, and SEM under both chosen plaintext and chosen ciphertext attacks. Then we discuss the relations among these notions given by Bellare *et al.* [12]. Moreover, we present a new indistinguishability-based notion called *Real-or-Permutation* (RoP), and we prove that RoP and RoR are equivalent.

The results given in this subsection are in the symmetric setting but they carry over to the asymmetric setting.

4.2.1.1 Left-or-Right Indistinguishability

The adversary plays two experiments, one for LoR-CPA and the other for LoR-CCA. Both experiments begin by the challenger choosing a secret key $K \in \mathcal{K}$ and a random bit $b \in \{0, 1\}$. The adversary is given access to a left-or-right encryption oracle which it can adaptively query. The encryption oracle queries sent by the adversary are in the form of (m_0, m_1) such that $|m_0| = |m_1|$. Upon arriving each encryption query, the oracle encrypts one of the messages $c \leftarrow \mathcal{E}_K(m_b)$, and returns the ciphertext. We call this ciphertext the *challenge ciphertext*. Additionally, in the LoR-CCA experiment, the adversary is given access to a decryption oracle. The adversary may query the decryption oracle on any ciphertext except the challenge ciphertext. The decryption oracle returns a message $m \leftarrow \mathcal{D}_K(c)$ corresponding to each decryption query. At some point the adversary outputs a bit b', and the experiment returns b' as well.

Definition 5 [**LoR-CPA and LoR-CCA**] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. Define experiments $\mathbf{Exp}_{\mathcal{SE}}^{lor-cpa-b}(A)$ and $\mathbf{Exp}_{\mathcal{SE}}^{lor-cca-b}(A)$ for an adversary A and a bit b as depicted in Figure 4.4. In both experiments, the adversary A is given access to a left-or-right encryption oracle $\mathsf{LoR}(\cdot)$. It is additionally given access to a decryption oracle $\mathsf{Dec}(\cdot)$ in the latter experiment. No restriction is imposed on the adversary's queries, except, it is assumed that the probability that the adversary queries $\mathsf{Dec}(\cdot)$ on previously returned ciphertexts by $\mathsf{LoR}(\cdot)$ is zero.

In both experiments, the adversary's goal is to output a bit b' as its guess of the challenge bit b. The adversary wins if $b' = b$. The corresponding advantages of an adversary A are given by:

$$\begin{aligned}
\mathbf{Adv}_{\mathcal{SE}}^{lor-cpa}(A) &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{lor-cpa-1}(A) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{lor-cpa-0}(A) = 1\right] \ , \\
\mathbf{Adv}_{\mathcal{SE}}^{lor-cca}(A) &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{lor-cca-1}(A) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{lor-cca-0}(A) = 1\right] \ .
\end{aligned}$$

The advantage functions of the scheme are defined to be:

```
$\underline{\mathbf{Exp}^{lor-cpa-b}_{\mathcal{SE}}(A)}$ $\boxed{\underline{\mathbf{Exp}^{lor-cca-b}_{\mathcal{SE}}(A)}}$

    $K \leftarrow \mathcal{K}$
    $b' \leftarrow A^{\mathsf{LoR}(\cdot)}$ $\boxed{b' \leftarrow A^{\mathsf{LoR}(\cdot),\mathsf{Dec}(\cdot)}}$
    return $b'$


$\underline{\mathsf{LoR}((m_0, m_1))}$

    if $|m_0| \neq |m_1|$ then
        return $\perp$
    else
        $c \leftarrow \mathcal{E}_K(m_b)$
        return $c$
    end if


$\underline{\mathsf{Dec}(c)}$

    $m \leftarrow \mathcal{D}_K(c)$
    return $m$
```

Figure 4.4: The LoR-CPA and LoR-CCA confidentiality notions. The boxed codes are excluded in LoR-CPA experiment, whereas they replace the codes adjacent to them in LoR-CCA experiment.

$$\begin{aligned}\mathbf{Adv}^{lor-cpa}_{\mathcal{SE}}(t, q_e, \mu_e) &= \max_A \left\{\mathbf{Adv}^{lor-cpa}_{\mathcal{SE}}(A)\right\} ,\\ \mathbf{Adv}^{lor-cca}_{\mathcal{SE}}(t, q_e, \mu_e, q_d, \mu_d) &= \max_A \left\{\mathbf{Adv}^{lor-cca}_{\mathcal{SE}}(A)\right\}\end{aligned}$$

for any positive integers $t, q_e, \mu_e, q_d, \mu_d$. Here the maximum is over all adversaries A with a running time of at most t, making at most q_e queries to the encryption oracle, totalling at most μ_e bits, and in case of $\mathbf{Exp}^{lor-cca-b}_{\mathcal{SE}}(A)$, making at most q_d queries to the decryption oracle, totalling at most μ_d bits. ▮

We say that the scheme $\mathcal{SE}$ is LoR-CPA (t, q_e, μ_e)-secure (respectively LoR-CCA $(t, q_e, \mu_e, q_d, \mu_d)$-secure) if $\mathbf{Adv}^{lor-cpa}_{\mathcal{SE}}(A)$ (respectively $\mathbf{Adv}^{lor-cca}_{\mathcal{SE}}(A)$) is small for all adversaries A using reasonable resources.

4.2.1.2 Real-or-Random Indistinguishability

This notion can be seen as an adaptation of LoR notion. The difference is that the adversary adaptively queries the real-or-random encryption oracle each time on a single message m, instead of two messages (m_0, m_1). If $b = 1$, the encryption oracle encrypts m and returns $c \leftarrow \mathcal{E}_K(m)$, otherwise the encryption oracle chooses a random bit string r where $|r| = |m|$ and returns $c \leftarrow \mathcal{E}_K(r)$. Therefore, the adversary is required to distinguish between the ciphertext corresponding to its query and the ciphertext of a redundant string. Now we give the formal definition.

Definition 6 [**RoR-CPA and RoR-CCA**] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. Define experiments $\mathbf{Exp}^{ror-cpa-b}_{\mathcal{SE}}(A)$ and $\mathbf{Exp}^{ror-cca-b}_{\mathcal{SE}}(A)$ for an adversary A and a bit b as depicted in Figure 4.5. In both experiments the adversary A is given access to a real-or-random encryption oracle $\mathsf{RoR}(\cdot)$. It is additionally given access to a decryption oracle $\mathsf{Dec}(\cdot)$ in the latter experiment. No restriction is imposed on the adversary's queries, except, it is assumed that the probability that the adversary queries $\mathsf{Dec}(\cdot)$ on previously returned ciphertexts c by $\mathsf{RoR}(\cdot)$ is zero.

In both experiments, the adversary's goal is to output a bit b' as its guess of the challenge bit b. The adversary wins if $b' = b$. The corresponding advantages of an adversary A are given by:

$$\begin{aligned}
\mathbf{Adv}^{ror-cpa}_{\mathcal{SE}}(A) &= \Pr\left[\mathbf{Exp}^{ror-cpa-1}_{\mathcal{SE}}(A) = 1\right] - \Pr\left[\mathbf{Exp}^{ror-cpa-0}_{\mathcal{SE}}(A) = 1\right] \ , \\
\mathbf{Adv}^{ror-cca}_{\mathcal{SE}}(A) &= \Pr\left[\mathbf{Exp}^{ror-cca-1}_{\mathcal{SE}}(A) = 1\right] - \Pr\left[\mathbf{Exp}^{ror-cca-0}_{\mathcal{SE}}(A) = 1\right] \ .
\end{aligned}$$

The advantage functions of the scheme are defined to be:

$$\begin{aligned}
\mathbf{Adv}^{ror-cpa}_{\mathcal{SE}}(t, q_e, \mu_e) &= \max_A \left\{\mathbf{Adv}^{ror-cpa}_{\mathcal{SE}}(A)\right\} \ , \\
\mathbf{Adv}^{ror-cca}_{\mathcal{SE}}(t, q_e, \mu_e, q_d, \mu_d) &= \max_A \left\{\mathbf{Adv}^{ror-cca}_{\mathcal{SE}}(A)\right\}
\end{aligned}$$

```
$\underline{\mathbf{Exp}_{\mathcal{SE}}^{ror-cpa-b}(A)}$ $\boxed{\underline{\mathbf{Exp}_{\mathcal{SE}}^{ror-cca-b}(A)}}$

    $K \leftarrow \mathcal{K}$
    $b' \leftarrow A^{\mathsf{RoR}(\cdot)}$ $\boxed{b' \leftarrow A^{\mathsf{RoR}(\cdot),\mathsf{Dec}(\cdot)}}$
    return $b'$


$\underline{\mathsf{RoR}(m)}$

    if $b = 1$ then
        $c \leftarrow \mathcal{E}_K(m)$
    else
        $r \leftarrow_{\$} \{0,1\}^{|m|}$
        $c \leftarrow \mathcal{E}_K(r)$
    end if
    return $c$


$\underline{\mathsf{Dec}(c)}$

    $m \leftarrow \mathcal{D}_K(c)$
    return $m$
```

Figure 4.5: The RoR-CPA and RoR-CCA confidentiality notions. The boxed codes are excluded in RoR-CPA experiment, whereas they replace the codes adjacent to them in RoR-CCA experiment.

for any positive integers $t, q_e, \mu_e, q_d, \mu_d$. Here the maximum is over all adversaries A with a running time of at most t, making at most q_e queries to the encryption oracle, totalling at most μ_e bits, and in case of $\mathbf{Exp}_{\mathcal{SE}}^{ror-cca-b}(A)$, making at most q_d queries to the decryption oracle, totalling at most μ_d bits. ■

We say that the scheme $\mathcal{SE}$ is RoR-CPA (t, q_e, μ_e)-secure (respectively RoR-CCA $(t, q_e, \mu_e, q_d, \mu_d)$-secure) if $\mathbf{Adv}_{\mathcal{SE}}^{ror-cpa}(A)$ (respectively $\mathbf{Adv}_{\mathcal{SE}}^{ror-cca}(A)$) is small for all adversaries A using reasonable resources.

4.2.1.3 Find-then-Guess Indistinguishability

The adversary plays two experiments FtG-CPA and FtG-CCA. Both experiments begin by the challenger choosing a secret key $K \in \mathcal{K}$ and a random bit $b \in \{0, 1\}$. In the previous two notions, LoR and RoR, the adversary runs in only one phase.

In FtG, the adversary runs in two phases. It begins with the find phase, where the adversary is given access to an encryption oracle. The adversary queries the encryption oracle on adaptively chosen messages m, to which the encryption oracle returns $c \leftarrow \mathcal{E}_K(m)$ in response. The aim of the find phase is for the adversary to choose two equal length messages (m_0, m_1) upon which it wishes to be challenged. The adversary also may preserve some state information s that might help it in the later phase. Then the challenger sends the challenge ciphertext $c \leftarrow \mathcal{E}_K(m_b)$ to the adversary. In the guess phase the adversary tries to determine the message to which c decrypts. Additionally, the adversary is given access to a decryption oracle in FtG-CCA experiment. The decryption oracle can be queried by the adversary on any message except the challenge ciphertext. Finally, the adversary outputs a bit b', and the experiment returns b' as well.

Definition 7 [**FtG-CPA and FtG-CCA**] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. Define experiments $\mathbf{Exp}_{\mathcal{SE}}^{ftg-cpa-b}(A)$ and $\mathbf{Exp}_{\mathcal{SE}}^{ftg-cca-b}(A)$ for an adversary A and a bit b as depicted in Figure 4.6. In both experiments, the adversary A is given access to an encryption oracle $\mathcal{E}_K(\cdot)$. The adversary is additionally given access to a decryption oracle $\mathcal{D}_K(\cdot)$ in the latter experiment. The two messages (m_0, m_1), output by the adversary at the end of the find phase, must be the same length. No restriction is imposed on the adversary's queries, except, it is assumed that the probability that the adversary queries $\mathcal{D}_K(\cdot)$ on the challenge ciphertexts c is zero.

In both experiments, the adversary's goal is to output a bit b' as its guess of the challenge bit b. The adversary wins if $b' = b$. The corresponding advantages of an adversary A are given by:

$$\begin{aligned}
\mathbf{Adv}_{\mathcal{SE}}^{ftg-cpa}(A) &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{ftg-cpa-1}(A) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{ftg-cpa-0}(A) = 1\right] \ , \\
\mathbf{Adv}_{\mathcal{SE}}^{ftg-cca}(A) &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{ftg-cca-1}(A) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{ftg-cca-0}(A) = 1\right] \ .
\end{aligned}$$

The advantage functions of the scheme are defined to be:

```
Experiment $\mathbf{Exp}_{\mathcal{SE}}^{ftg-cpa-b}(A)$
    $K \leftarrow \mathcal{K}$
    $((m_0, m_1), s) \leftarrow A^{\mathcal{E}_K(\cdot)}(\mathsf{find})$
    $c \leftarrow \mathcal{E}_K(m_b)$
    $b' \leftarrow A^{\mathcal{E}_K(\cdot)}(\mathsf{guess}, c, s)$
    return $b'$
```

```
Experiment $\mathbf{Exp}_{\mathcal{SE}}^{ftg-cca-b}(A)$
    $K \leftarrow \mathcal{K}$
    $((m_0, m_1), s) \leftarrow A^{\mathcal{E}_K(\cdot), \mathcal{D}_K(\cdot)}(\mathsf{find})$
    $c \leftarrow \mathcal{E}_K(m_b)$
    $b' \leftarrow A^{\mathcal{E}_K(\cdot), \mathcal{D}_K(\cdot)}(\mathsf{guess}, c, s)$
    return $b'$
```

Figure 4.6: The FtG-CPA and FtG-CCA confidentiality notions in the left hand and the right hand side, respectively.

$$\mathbf{Adv}_{\mathcal{SE}}^{ftg-cpa}(t, q_e, \mu_e) = \max_A \left\{ \mathbf{Adv}_{\mathcal{SE}}^{ftg-cpa}(A) \right\} ,$$
$$\mathbf{Adv}_{\mathcal{SE}}^{ftg-cca}(t, q_e, \mu_e, q_d, \mu_d) = \max_A \left\{ \mathbf{Adv}_{\mathcal{SE}}^{ftg-cca}(A) \right\}$$

for any positive integers $t, q_e, \mu_e, q_d, \mu_d$. Here the maximum is over all adversaries A with a running time of at most t, making at most q_e queries to the encryption oracle, totalling at most $(\mu_e - |m_0|)$ bits, and in case of $\mathbf{Exp}_{\mathcal{SE}}^{ftg-cca-b}(A)$, making at most q_d queries to the decryption oracle, totalling at most μ_d bits. ∎

We say that the scheme $\mathcal{SE}$ is FtG-CPA (t, q_e, μ_e)-secure (respectively FtG-CCA $(t, q_e, \mu_e, q_d, \mu_d)$-secure) if $\mathbf{Adv}_{\mathcal{SE}}^{ftg-cpa}(A)$ (respectively $\mathbf{Adv}_{\mathcal{SE}}^{ftg-cca}(A)$) is small for all adversaries A using reasonable resources.

4.2.1.4 Semantic Security

This notion captures the idea of security for an encryption scheme defined by Shannon (see Section 2.2). This is, a secure encryption scheme should hide all information about an unknown plaintext. In other words, an encryption scheme is secure if an adversary is unable to obtain any partial information about the plaintext from the ciphertext. The security in Shannon's notion is computationally unconditional, but in semantic security it depends on an adversary's computational effort.

The adversary plays two experiments SEM-CPA and SEM-CCA. The experiments are characterised by a bit $b \in \{0, 1\}$. The adversary runs in two phases. First, in the select phase, the adversary is given access to an encryption oracle. It adaptively queries the encryption oracle, which returns the corresponding ciphertext $c \leftarrow \mathcal{E}_K(m)$. At the end of this phase, the adversary outputs a message space. The message space must be valid, this means all the messages with non-zero probability must have the same length. The adversary may also retain some state information s that might help it in the later phase. The challenger samples the message space to obtain two messages m_0 and m_1. It then sends the challenge ciphertext $c \leftarrow \mathcal{E}_K(m_1)$ to the adversary. In the second phase, the predict phase, the adversary is also given access to the encryption oracle. At the end of this phase, the adversary outputs a function f and a value α. Additionally, in SEM-CCA experiment the adversary is given access to a decryption oracle in both phases. The decryption oracle can be adaptively queried on any ciphertext except the challenge ciphertext. The adversary hopes that if $b = 1$ then $f(m_1) = \alpha$, otherwise $f(m_0) = \alpha$. Note that if the latter is the case, then the adversary has not seen the corresponding ciphertext. The encryption scheme is semantically secure if the adversary succeeds about as often in the latter case $(f(m_0) = \alpha)$ as the former case.

The message space can be considered as a probabilistic algorithm which the adversary outputs its code. Each time this program is executed, it samples and outputs two messages. Moreover, the function f is deterministic to which the adversary outputs its program.

Definition 8 [**SEM-CPA and SEM-CCA**] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. Define experiments $\mathbf{Exp}_{\mathcal{SE}}^{sem-cpa-b}(A)$ and $\mathbf{Exp}_{\mathcal{SE}}^{sem-cca-b}(A)$ that are characterised by a bit b for an adversary A as depicted in Figure 4.7. In both experiments the adversary A is given access to an encryption oracle $\mathcal{E}_K(\cdot)$. The adversary is additionally given access to a decryption oracle $\mathcal{D}_K(\cdot)$ in the latter experiment. No restriction is imposed on the adversary's queries, except, it is assumed that the probability that the adversary queries the challenge ciphertext c to $\mathcal{D}_K(\cdot)$ is zero. At the end of the select phase, the adversary outputs a valid message space $\mathcal{M}$. Moreover, the adversary outputs a function f and a value α at the end of the predict phase.

In both experiments, the adversary's goal is to output a function f and a value α such that $f(m_b) = \alpha$. Then the adversary wins. The corresponding advantages of an adversary A are given by:

Experiment $\mathbf{Exp}_{\mathcal{SE}}^{sem-cpa-b}(A)$

```
$K \leftarrow \mathcal{K}$
$(\mathcal{M}, s) \leftarrow A^{\mathcal{E}_K(\cdot)}(\mathsf{select})$
$m_0 \leftarrow \mathcal{M}$; $m_1 \leftarrow \mathcal{M}$
$c \leftarrow \mathcal{E}_K(m_1)$
$(f, \alpha) \leftarrow A^{\mathcal{E}_K(\cdot)}(\mathsf{predict}, c, s)$
if $f(m_b) = \alpha$ then
    $b' \leftarrow 1$
else
    $b' \leftarrow 0$
end if
return $b'$
```

Experiment $\mathbf{Exp}_{\mathcal{SE}}^{sem-cca-b}(A)$

```
$K \leftarrow \mathcal{K}$
$(\mathcal{M}, s) \leftarrow A^{\mathcal{E}_K(\cdot),\mathcal{D}_K(\cdot)}(\mathsf{select})$
$m_0 \leftarrow \mathcal{M}$; $m_1 \leftarrow \mathcal{M}$
$c \leftarrow \mathcal{E}_K(m_1)$
$(f, \alpha) \leftarrow A^{\mathcal{E}_K(\cdot),\mathcal{D}_K(\cdot)}(\mathsf{predict}, c, s)$
if $f(m_b) = \alpha$ then
    $b' \leftarrow 1$
else
    $b' \leftarrow 0$
end if
return $b'$
```

Figure 4.7: The SEM-CPA and SEM-CCA confidentiality notions in the left hand and the right hand side, respectively.

$$\begin{aligned}
\mathbf{Adv}_{\mathcal{SE}}^{sem-cpa}(A) &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-cpa-1}(A) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-cpa-0}(A) = 1\right] \ , \\
\mathbf{Adv}_{\mathcal{SE}}^{sem-cca}(A) &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-cca-1}(A) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-cca-0}(A) = 1\right] \ .
\end{aligned}$$

The advantage functions of the scheme are defined to be:

$$\begin{aligned}
\mathbf{Adv}_{\mathcal{SE}}^{sem-cpa}(t, q_e, \mu_e) &= \max_A \left\{\mathbf{Adv}_{\mathcal{SE}}^{sem-cpa}(A)\right\} \ , \\
\mathbf{Adv}_{\mathcal{SE}}^{sem-cca}(t, q_e, \mu_e, q_d, \mu_d) &= \max_A \left\{\mathbf{Adv}_{\mathcal{SE}}^{sem-cca}(A)\right\}
\end{aligned}$$

for any positive integers $t, q_e, \mu_e, q_d, \mu_d$. Here the maximum is over all adversaries A with a running time of at most t, making at most q_e queries to the encryption oracle, totalling at most μ_e bits, and in case of $\mathbf{Exp}_{\mathcal{SE}}^{sem-cca-b}(A)$, making at most q_d queries to the decryption oracle, totalling at most μ_d bits. Note that the running time t includes the maximum time required to sample from the message space, and the maximum time required to run the function f on any string. Moreover, the length of the encryption queries μ_e includes a sum over all sampled messages from the message space $\mathcal{M}$, also a sum over the size of the program for the message space, the function$f$, and the length of the value α. ■

We say that the scheme $\mathcal{SE}$ is SEM-CPA (t, q_e, μ_e)-secure (respectively SEM-CCA

$(t, q_e, \mu_e, q_d, \mu_d)$-secure) if $\mathbf{Adv}_{\mathcal{SE}}^{sem-cpa}(A)$ (respectively $\mathbf{Adv}_{\mathcal{SE}}^{sem-cca}(A)$) is small for all adversaries A using reasonable resources.

Semantic security is a very strong notion of security. Despite this, we show that indistinguishability-based notions imply semantic security.

4.2.1.5 Real-or-Permutation Indistinguishability

We introduce a new notion of security that is very similar to the notion of RoR. The adversary plays two experiments: RoP-CPA and RoP-CCA. Both experiments begin with the challenger choosing a key $K \leftarrow \mathcal{K}$ and a bit $b \in \{0, 1\}$. In both experiments, the adversary is given access to a real-or-permutation encryption oracle. The adversary adaptively requests the encryption of messages m. The encryption oracle response depends on the bit b. If $b = 1$ then the oracle returns the ciphertext $c \leftarrow \mathcal{E}_K(m)$. Otherwise, the challenger chooses a random permutation function with the same domain size as the message length, then applies the permutation function to the message. Finally, the encryption oracle encrypts the permuted message and returns the ciphertext. We call the ciphertext that is returned by the encryption oracle, the challenge ciphertext. Additionally, in the RoP-CCA experiment, the adversary is given access to a decryption oracle. The decryption oracle can be queried on any ciphertext except the challenge ciphertext. The adversary is required to distinguish between the encryption of a message and the encryption of a permutation of that message.

Definition 9 [**RoP-CPA and RoP-CCA**] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. Define experiments $\mathbf{Exp}_{\mathcal{SE}}^{rop-cpa-b}(A)$ and $\mathbf{Exp}_{\mathcal{SE}}^{rop-cca-b}(A)$ for an adversary A and a bit b as depicted in Figure 4.8. In both experiments, the adversary A is given access to a real-or-permutation encryption oracle $\mathsf{RoP}(\cdot)$, and it is additionally given access to a decryption oracle $\mathsf{Dec}(\cdot)$ in the latter experiment. No restriction is imposed on the adversary's queries, except, it is assumed that the probability that the adversary queries $\mathsf{Dec}(\cdot)$ on previously returned ciphertext by $\mathsf{RoP}(\cdot)$ is zero.

In both experiments, the adversary's goal is to output a bit b' as its guess of the challenge bit b. The adversary wins if $b' = b$. The corresponding advantages of an adversary A are given by:

$$\begin{aligned}\mathbf{Adv}_{\mathcal{SE}}^{rop-cpa}(A) &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-cpa-1}(A)=1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-cpa-0}(A)=1\right]\ ,\\ \mathbf{Adv}_{\mathcal{SE}}^{rop-cca}(A) &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-cca-1}(A)=1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-cca-0}(A)=1\right]\ .\end{aligned}$$

The advantage functions of the scheme are defined to be:

$$\begin{aligned}\mathbf{Adv}_{\mathcal{SE}}^{rop-cpa}(t, q_e, \mu_e) &= \max_{A}\left\{\mathbf{Adv}_{\mathcal{SE}}^{rop-cpa}(A)\right\}\ ,\\ \mathbf{Adv}_{\mathcal{SE}}^{rop-cca}(t, q_e, \mu_e, q_d, \mu_d) &= \max_{A}\left\{\mathbf{Adv}_{\mathcal{SE}}^{rop-cca}(A)\right\}\end{aligned}$$

for any positive integers $t, q_e, \mu_e, q_d, \mu_d$. Here the maximum is over all adversaries A with a running time of at most t, making at most q_e queries to the encryption oracle, totalling at most μ_e bits, and in case of $\mathbf{Exp}_{\mathcal{SE}}^{rop-cca-b}(A)$, making at most q_d queries to the decryption oracle, totalling at most μ_d bits. ▮

We say that the scheme $\mathcal{SE}$ is RoP-CPA (t, q_e, μ_e)-secure (respectively RoP-CCA $(t, q_e, \mu_e, q_d, \mu_d)$-secure) if $\mathbf{Adv}_{\mathcal{SE}}^{rop-cpa}(A)$ (respectively $\mathbf{Adv}_{\mathcal{SE}}^{rop-cca}(A)$) is small for all adversaries A using reasonable resources.

Next we show that RoP and RoR are equivalent. In Chapter 6 we will describe the quantum analogue of distinguishability notions when the adversary is given quantum superposition access to its oracles. We prove that the quantum analogue of the RoP notion of security is achievable even against such a strong adversarial model. Moreover, we will go on to prove that the quantum analogue of RoP implies the quantum analogue of SEM.

```
$\underline{\mathbf{Exp}^{rop-cpa-b}_{\mathcal{SE}}(A)}$ $\boxed{\underline{\mathbf{Exp}^{rop-cca-b}_{\mathcal{SE}}(A)}}$

    $K \leftarrow \mathcal{K}$
    $b' \leftarrow A^{\mathsf{RoP}(\cdot)}$ $\boxed{b' \leftarrow A^{\mathsf{RoP}(\cdot),\mathsf{Dec}(\cdot)}}$
    return $b'$


$\underline{\mathsf{RoP}(m)}$

    if $b = 1$ then
        $c \leftarrow \mathcal{E}_K(m)$
    else
        $\Pi \leftarrow\!\!\$\ \mathsf{Perm}(|m|)$
        $m' \leftarrow \Pi(m)$
        $c \leftarrow \mathcal{E}_K(m')$
    end if
    return $c$


$\underline{\mathsf{Dec}(c)}$

    $m \leftarrow \mathcal{D}_K(c)$
    return $m$
```

Figure 4.8: The RoP-CPA and RoP-CCA confidentiality notions. The boxed codes are excluded in RoP-CPA experiment, whereas they replace the codes adjacent to them in RoP-CCA experiment.

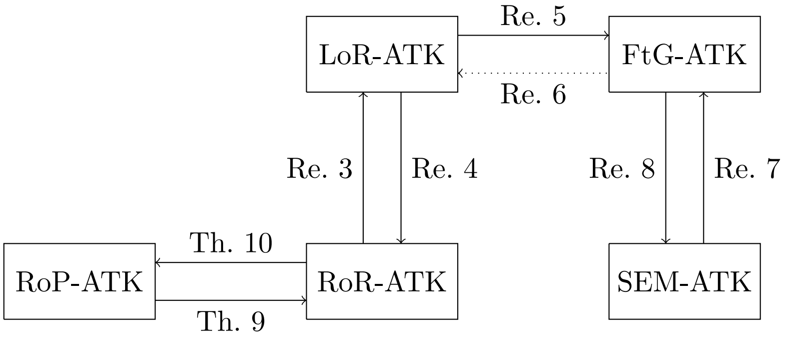

Figure 4.9: Relations among confidentiality notions where ATK $\in \{\text{CPA}, \text{CCA}\}$. A solid line denotes a security-preserving reduction from a notion to another, while a dotted line denotes a reduction that is not security-preserving.

4.2.1.6 Relations Among Notions

Here we discuss the relations among different notions of security from a concrete security perspective. The concrete results help us to see how strong a notion of security is, compared to other notions to which it is asymptotically equivalent. Bellare *et al.* [12] show the reduction among LoR, RoR, FtG, and SEM. It turns out that LoR security implies other notions of security. On top of that we prove that RoP implies RoR which means, as we can see below, RoP also implies LoR. The relations among notions are illustrated in Figure 4.9.

We adopt the notations used by Bellare *et al.* [12]. We use $A \Rightarrow B$ to denote a security-preserving reduction from A to B, i.e., a reduction where the advantage of an adversary against B is bounded by the advantage of an adversary against A multiplied by a small constant factor. If a reduction from A to B is not security preserving, we follow the work of Bellare *et al.* [12] and use the notation $A \rightarrow B$. Notice that even a not security-preserving reduction can lead to a secure scheme if the security parameters are chosen suitably large. We examine the relations simultaneously with respect to CPA and CCA. Therefore, we use the symbol ATK instead of CPA and CCA.

The first two results show that LoR and RoR are equivalent.

Result 3 (RoR-ATK $\Rightarrow$ LoR-ATK ([12] Theorem 1)) *For any symmetric en-*

cryption scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$*,*

$$\begin{aligned}\mathbf{Adv}^{lor-cpa}_{\mathcal{SE}}(t, q_e, \mu_e) &\leq 2 \cdot \mathbf{Adv}^{ror-cpa}_{\mathcal{SE}}(t, q_e, \mu_e) \;\; \textit{and} \\ \mathbf{Adv}^{lor-cca}_{\mathcal{SE}}(t, q_e, \mu_e, q_d, \mu_d) &\leq 2 \cdot \mathbf{Adv}^{ror-cca}_{\mathcal{SE}}(t, q_e, \mu_e, q_d, \mu_d) \;\; .\end{aligned}$$

Result 4 (LoR-ATK $\Rightarrow$ RoR-ATK ([12] Theorem 2)) *For any symmetric encryption scheme* $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$*,*

$$\begin{aligned}\mathbf{Adv}^{ror-cpa}_{\mathcal{SE}}(t, q_e, \mu_e) &\leq \mathbf{Adv}^{lor-cpa}_{\mathcal{SE}}(t, q_e, \mu_e) \;\; \textit{and} \\ \mathbf{Adv}^{ror-cca}_{\mathcal{SE}}(t, q_e, \mu_e, q_d, \mu_d) &\leq \mathbf{Adv}^{lor-cca}_{\mathcal{SE}}(t, q_e, \mu_e, q_d, \mu_d) \;\; .\end{aligned}$$

The next two results show that LoR security implies FtG security but the reduction from FtG to LoR is not security-preserving. This is because the advantage of an adversary against LoR security is bounded by the advantage of an adversary against FtG security multiplied by the total number of encryption oracle queries, as shown in Result 6. Hence the level of LoR security that can be achieved decreases as the total number of encryption oracle queries increases.

Result 5 (LoR-ATK $\Rightarrow$ FtG-ATK ([12] Theorem 3)) *For any symmetric encryption scheme* $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$*,*

$$\begin{aligned}\mathbf{Adv}^{ftg-cpa}_{\mathcal{SE}}(t, q_e, \mu_e) &\leq \mathbf{Adv}^{lor-cpa}_{\mathcal{SE}}(t, q_e + 1, \mu_e) \;\; \textit{and} \\ \mathbf{Adv}^{ftg-cca}_{\mathcal{SE}}(t, q_e, \mu_e, q_d, \mu_d) &\leq \mathbf{Adv}^{lor-cca}_{\mathcal{SE}}(t, q_e + 1, \mu_e, q_d, \mu_d) \;\; .\end{aligned}$$

Result 6 (FtG-ATK $\rightarrow$ LoR-ATK ([12] Theorem 4)) *For any symmetric encryption scheme* $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$*,*

$$\begin{aligned}\mathbf{Adv}_{\mathcal{SE}}^{lor-cpa}(t, q_e, \mu_e) &\leq q_e \cdot \mathbf{Adv}_{\mathcal{SE}}^{ftg-cpa}(t, q_e, \mu_e) \;\; \textit{and} \\ \mathbf{Adv}_{\mathcal{SE}}^{lor-cca}(t, q_e, \mu_e, q_d, \mu_d) &\leq q_e \cdot \mathbf{Adv}_{\mathcal{SE}}^{ftg-cca}(t, q_e, \mu_e, q_d, \mu_d) \;\; .\end{aligned}$$

The following two results show that FtG security and SEM security are equivalent. From the latter we can deduce that LoR security implies SEM security. This is an important result because SEM security reflects the security we want in practice. On the other hand, it is easier to analyse our scheme using the LoR notion of security. Therefore, if we can prove that our scheme is LoR secure, then SEM security automatically follows.

Result 7 (SEM-ATK ⇒ FtG-ATK ([12] Theorem 6)) *For any symmetric encryption scheme* $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$,

$$\begin{aligned}\mathbf{Adv}_{\mathcal{SE}}^{ftg-cpa}(t, q_e, \mu_e) &\leq \mathbf{Adv}_{\mathcal{SE}}^{sem-cpa}(t, q_e, \mu_e) \;\; \textit{and} \\ \mathbf{Adv}_{\mathcal{SE}}^{ftg-cca}(t, q_e, \mu_e, q_d, \mu_d) &\leq \mathbf{Adv}_{\mathcal{SE}}^{sem-cca}(t, q_e, \mu_e, q_d, \mu_d) \;\; .\end{aligned}$$

Result 8 (FtG-ATK ⇒ SEM-ATK ([12] Theorem 7)) *For any symmetric encryption scheme* $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$,

$$\begin{aligned}\mathbf{Adv}_{\mathcal{SE}}^{sem-cpa}(t, q_e, \mu_e) &\leq 2 \cdot \mathbf{Adv}_{\mathcal{SE}}^{ftg-cpa}(t, q_e, \mu_e) \;\; \textit{and} \\ \mathbf{Adv}_{\mathcal{SE}}^{sem-cca}(t, q_e, \mu_e, q_d, \mu_d) &\leq 2 \cdot \mathbf{Adv}_{\mathcal{SE}}^{ftg-cca}(t, q_e, \mu_e, q_d, \mu_d) \;\; .\end{aligned}$$

In the next two theorems we prove that RoR security and RoP security are equivalent.

Theorem 9 (RoP-ATK ⇒ RoR-ATK) *For any symmetric encryption scheme* $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$,

$$\mathbf{Adv}_{\mathcal{SE}}^{ror-cpa}(t, q_e, \mu_e) \leq \mathbf{Adv}_{\mathcal{SE}}^{rop-cpa}(t, q_e, \mu_e) \;\; and$$
$$\mathbf{Adv}_{\mathcal{SE}}^{ror-cca}(t, q_e, \mu_e, q_d, \mu_d) \leq \mathbf{Adv}_{\mathcal{SE}}^{rop-cca}(t, q_e, \mu_e, q_d, \mu_d) \; .$$

Proof Assume that A is an adversary attacking $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ in RoR sense. We construct a new adversary B, using A, that attacks $\mathcal{SE}$ in RoP sense.

B uses its oracles, $\mathsf{RoP}(\cdot)$ and $\mathsf{Dec}(\cdot)$, to provide a simulation of A's oracles, $\mathsf{LoR}(\cdot)$ and $\mathsf{Dec}(\cdot)$. The adversary B runs A.

When A makes an encryption oracle query, B will respond with the output from its encryption oracle $\mathsf{RoP}(\cdot)$. The output depends on the bit b. Note that in RoP experiments, a permutation Π is chosen uniformly at random for each encryption query if $b = 0$. Then applying Π to any message leads to a uniform probability distribution of all messages. Therefore, the output is a random message which gets encrypted by the encryption oracle. When A makes a decryption oracle query, B will respond with the corresponding plaintext. It is assumed that the probability that the adversary queries the decryption oracle on ciphertexts previously returned by the encryption oracle is zero.

At some point, the adversary A outputs a bit b'.

For either case of $b = 0$ or $b = 1$, B provides a perfect simulation of RoR-CPA and RoR-CCA experiments for A. So B succeeds with the same probability as A. Hence, for B's advantage we have:

$$\begin{aligned}\mathbf{Adv}_{\mathcal{SE}}^{rop\text{-}atk}(B) &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-atk-1}(B) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-atk-0}(B) = 1\right] \\ &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{ror-atk-1}(A) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{ror-atk-0}(A) = 1\right] \\ &= \mathbf{Adv}_{\mathcal{SE}}^{ror-atk}(A) \; . \qquad (4.4)\end{aligned}$$

Both B and A use the same resources. They are running in time at most t, making q_e encryption and q_d decryption oracle queries, totalling at most μ_e and μ_d bits respectively. Since A is an arbitrary adversary, then we have proven the claimed relation between RoP-ATK and RoR-ATK. ▮

Theorem 10 (RoR-ATK ⇒ RoP-ATK) *For any symmetric encryption scheme* $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$*,*

$$\mathbf{Adv}_{\mathcal{SE}}^{rop-cpa}(t, q_e, \mu_e) \leq \mathbf{Adv}_{\mathcal{SE}}^{ror-cpa}(t, q_e, \mu_e) \;\; \textit{and}$$
$$\mathbf{Adv}_{\mathcal{SE}}^{rop-cca}(t, q_e, \mu_e, q_d, \mu_d) \leq \mathbf{Adv}_{\mathcal{SE}}^{ror-cca}(t, q_e, \mu_e, q_d, \mu_d) \;\; .$$

Proof Assume that B is an adversary attacking $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ in RoP sense. We construct a new adversary A, using B, that attacks $\mathcal{SE}$ in RoR sense.

A uses its oracles, $\mathsf{RoR}(\cdot)$ and $\mathsf{Dec}(\cdot)$, to provide a simulation of B's oracles, $\mathsf{RoP}(\cdot)$ and $\mathsf{Dec}(\cdot)$. The adversary A runs B.

When B makes an encryption oracle query, A will respond with the output from its encryption oracle $\mathsf{RoR}(\cdot)$. The output depends on the bit b. When B makes a decryption oracle query, A will respond with the corresponding plaintext. It is assumed that the probability that the adversary queries the decryption oracle on ciphertexts previously returned by the encryption oracle is zero.

At some point the adversary B outputs a bit b'.

For either case of $b = 0$ or $b = 1$, A provides a perfect simulation of RoP-CPA and RoP-CCA experiments for B. So A succeeds with the same probability as B. Hence, for A's advantage we have:

$$\begin{aligned}
\mathbf{Adv}_{\mathcal{SE}}^{ror-atk}(A) &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{ror-atk-1}(A) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{ror-atk-0}(A) = 1\right] \\
&= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-atk-1}(B) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-atk-0}(B) = 1\right] \\
&= \mathbf{Adv}_{\mathcal{SE}}^{rop-atk}(B) \;\; . \qquad (4.5)
\end{aligned}$$

Both A and B use the same resources. They are running at most in time t, making q_e encryption and q_d decryption oracle queries, totalling at most μ_e and μ_d bits respectively. Since B is an arbitrary adversary, then we have proven the claimed relation between RoR-ATK and RoP-ATK. ∎

4.2.2 Modes of Operation

In Subsection 4.1.1, we discussed block ciphers such as AES. Consider an AES scheme with a fixed key. If we encrypt the same 128-bit block of message by AES twice, we get the same ciphertext. Hence, an adversary could gain partial information about the encrypted message. This is because AES, like all block ciphers, is deterministic. Therefore, they neither satisfy the semantic security model nor any indistinguishability-based security model, unless a new key is used to encrypt each block of a message. That is very hard to achieve in practice, and rather an unrealistic assumption. To be able to encrypt and decrypt multiple blocks of data, with the same key, using block ciphers, we build encryption schemes, known as *modes of operation.* A mode of operation is essentially a way of encrypting/decryption arbitrary length plaintext/ciphertext using a block cipher. It can provide a cryptographic goal such as confidentiality, authenticity, or both. Here we merely concentrate on confidentiality modes of operation, but we refer to work of Rogaway [96] for further details.

A number of popular confidentiality modes of operation were standardised in 2001 by NIST in SP 800-38A [55]. Among them are *Electronic Code Book* (ECB), *Cipher Block Chaining* (CBC), and *Counter* (CTR). The first one, ECB, is deterministic and therefore it does not satisfy our SEM-CPA or LoR-CPA security models. Bellare *et al.* [12] proved that CBC and CTR are LoR-CPA secure, therefore, SEM-CPA secure also. However, they are not LoR-CCA secure, therefore not SEM-CCA secure. Here we only describe CTR mode, which is regarded as the best choice among the set of the confidentiality modes of operation [96]. We then evaluate its security against the quantum computing attack.

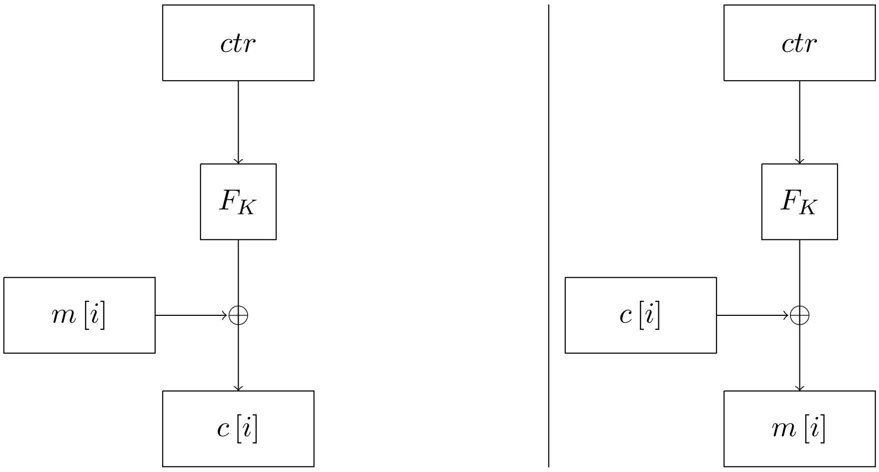

Figure 4.10: Counter mode of operation. Encryption and decryption processes are shown in the left and right hand side, respectively.

4.2.2.1 Counter Mode

CTR mode, as depicted in Figure 4.10, turns a block cipher into a stream cipher. That is, a counter value is encrypted using the block cipher, and then the result is XORed with the plaintext. After encryption of each block, the counter value is updated. It is usually incremented by one.

CTR mode has two variants: *stateful* and *randomised*. We use CTR to denote stateful counter mode, where the counter is maintained as the state of encryption. We use $\mathsf{CTR\$}$ to denote randomised counter mode, where the counter is a bit string chosen uniformly at random for each ciphertext. In either of these variants, given a counter value and an arbitrary length message, a key stream is created. This is done by calculating the message's number of block, and then iterating encryption of successive values of the counter using the block cipher accordingly. Concatenation of the block cipher outputs yields a key stream at least as long as the arbitrary length message. To encrypt, the message is XORed with the key stream in the one-time pad fashion. To decrypt, the same procedure is taken to produce a key stream which is then XORed with the ciphertext. We describe randomised CTR mode first.

Construction 1 (Randomised CTR Mode ($\mathsf{CTR\$}$)**)** *Let* $F : \{0,1\}^k \times \{0,1\}^l \rightarrow \{0,1\}^l$ *be a family of functions, possibly a block cipher but not necessarily.* CTR *mode over* F *with a random starting point is a probabilistic, stateless symmetric encryption*

```
CTR$-$\mathcal{K}$

    $K \leftarrow_{\$} \{0,1\}^k$
    return $K$
```

```
CTR$-$\mathcal{E}_K(m)$

    $r \leftarrow_{\$} \{0,1\}^l$
    Parse $m$ as $m[1] \cdots m[n]$
    for $i = 1, \ldots, n$ do
      $c[i] = F_K(r+i) \oplus m[i]$
    end for
    $c \leftarrow c[1] \cdots c[n]$
    return $(r, c)$
```

```
CTR$-$\mathcal{D}_K(c)$

    Parse $c$ as $c[1] \cdots c[n]$
    for $i = 1, \ldots, n$ do
      $m[i] = F_K(r+i) \oplus c[i]$
    end for
    $m \leftarrow m[1] \cdots m[n]$
    return $m$
```

Figure 4.11: Randomised CTR mode

scheme $\mathsf{CTR\$}[F] = (\mathsf{CTR\$}\text{-}\mathcal{K}, \mathsf{CTR\$}\text{-}\mathcal{E}, \mathsf{CTR\$}\text{-}\mathcal{D})$ *as shown in Figure 4.11. The message m to be encrypted is regarded as a sequence of l-bit blocks, $m = m[1] \cdots m[n]$. Accordingly, the ciphertext c is a sequence of l-bit blocks $c = c[1] \cdots c[n]$.*

The following result shows LoR-CPA (t, q, μ)-security of $\mathsf{CTR\$}$ mode.

Result 11 (Security of $\mathsf{CTR\$}$ Mode Using a PRF ([12] Theorem 11)) *Let $F : \{0,1\}^k \times \{0,1\}^l \to \{0,1\}^l$ be a block cipher. For any $\mathsf{CTR\$}[F]$ scheme, we have:*

$$\mathbf{Adv}^{lor\text{-}cpa}_{\mathsf{CTR\$}[F]}(t, q, \mu) \leq 2 \cdot \mathbf{Adv}^{prf}_{F}(t', q') + \frac{\mu^2}{2^l} \ ,$$

where $t' = t + (q + l\mu)$ and $q' = \mu$.

We now describe stateful CTR mode.

Construction 2 (Stateful CTR Mode (CTR)**)** *Let $F : \{0,1\}^k \times \{0,1\}^l \to \{0,1\}^l$ be a family of functions, possibly a block cipher but not necessarily.* CTR *mode over F with a counter starting point is a stateful symmetric encryption scheme* $\mathsf{CTR}[F] = (\mathsf{CTR}\text{-}\mathcal{K}, \mathsf{CTR}\text{-}\mathcal{E}, \mathsf{CTR}\text{-}\mathcal{D})$ *as shown in Figure 4.12. The message m to be encrypted is regarded as a sequence of l-bit blocks, $m = m[1] \cdots m[n]$. Accordingly, the ciphertext c is a sequence of l-bit blocks $c = c[1] \cdots c[n]$. The encryption counter ϱ_0 and decryption counter ς_0 are initially zero. Total number of encrypted blocks is restricted to be at most 2^l.*

```
CTR-$\mathcal{K}$

  $K \leftarrow\$ \{0,1\}^k$
  return $(K, 0, \varepsilon)$
```

```
CTR-$\mathcal{E}_K(m, \varrho)$

  $c[0] \leftarrow \varrho$
  Parse $m$ as $m[1] \cdots m[n]$
  if $c[0] + n \geq 2^n$ then
    return $\perp$
  else
    for $i = 1, \ldots, n$ do
      $c[i] = F_K(c[0] + i) \oplus m[i]$
    end for
    $c \leftarrow c[0]\, c[1] \cdots c[n]$
    return $(\varrho + n, c)$
  end if
```

```
CTR-$\mathcal{D}_K(c, \varsigma)$

  Parse $c$ as $c[0]\, c[1] \cdots c[n]$
  if $\varsigma + n \geq 2^n$ then
    return $\perp$
  else
    for $i = 1, \ldots, n$ do
      $m[i] = F_K(c[0] + i) \oplus c[i]$
    end for
    $m \leftarrow m[1] \cdots m[n]$
    return $(\varsigma, m)$
  end if
```

Figure 4.12: Stateful CTR mode

The following result shows LoR-CPA (t, q, μ)-security of CTR mode that is different from Result 11.

Result 12 (Security of CTR Mode Using a PRF ([12] Theorem 13)) *Let $F : \{0,1\}^k \times \{0,1\}^l \to \{0,1\}^l$ be a block cipher. For any $\mathsf{CTR}[F]$ scheme we have:*

$$\mathbf{Adv}^{\textit{lor-cpa}}_{\mathsf{CTR}[F]}(t, q, \mu) \leq 2 \cdot \mathbf{Adv}^{prf}_{F}\left(t', q'\right) ,$$

where $t' = t + (q + l\mu)$ and $q' = \mu$.

Note the difference between Result 11 and Result 12. The former shows that CTR\$ is insecure regardless of the security of F as a PRF if the scheme encrypts more than $\mu = 2^{l/2}$ blocks of messages with the same key. This is due to the birthday attack on block ciphers where the prf-advantage of A may be as large as $\mu^2/2^l$. In contrast, this is not the case for CTR mode where it might be secure as long as the number of blocks queried is at most 2^l.

4.2.3 Quantum Computation Attack

In Section 3.2 and Subsection 3.4.1, we explained that quantum algorithms can be a threat to the supposed security of modern cryptosystems. This threat is more serious against asymmetric cryptosystems where quantum algorithms such as Shor's algorithm [103] might solve their underlying hard problems using reasonable resources. In contrast, this threat has never been seriously considered against symmetric cryptosystems, where the best known quantum algorithms need an exponentially large amount of resources to break them. For instance, quantum algorithms such as Grover's algorithm [65], and that of Brassard *et al.* [34, 33], despite being faster than their classical counterparts, still need $\sqrt{N}$ oracle queries to recover the secret key and $\left(\sqrt[3]{N/r}\right)$ oracle queries to find a collision for an r-to-one function, respectively. We believe this is the reason for the lack of a formal security analysis of symmetric schemes against quantum computation attacks. At first glance, this might make sense, but we argue that this approach takes a modern cryptosystem as a black-box, therefore it might miss out on the flaws the scheme might have in its construction. With the latter in mind, we give a formal security analysis of CTR mode against quantum computation attacks. We first discuss LoR-CPA security of CTR mode, and then security of $\mathsf{CTR\$}$ mode. Note that, as it turns out, the following security proofs are essentially identical to the security proofs of Counter mode in classical setting given by Bellare *et al.* [12].

Theorem 13 (Security of CTR Mode Using a QPRF) *Let $F : \{0,1\}^k \times \{0,1\}^l \to \{0,1\}^l$ be a block cipher. For any $\mathsf{CTR}[F]$, assume $\mathcal{A}$ is a quantum adversary attacking $\mathsf{CTR}[F]$ in a LoR-CPA sense, with a running time of at most t, making at most q queries to the encryption oracle, and the size of the classical output μ bits, and having advantage*

$$\mathbf{Adv}^{\mathit{lor\text{-}cpa}}_{\mathsf{CTR}[F]}(\mathcal{A}) \geq \epsilon \ .$$

Then there exists a quantum adversary $\mathcal{B}$ attacking F with a running time of at most $t' = t + (q + l\mu)$, making at most $q' = \mu$ queries to the oracle, and having advantage

$$\mathbf{Adv}^{qprf}_{F}(\mathcal{B}) \geq \frac{\epsilon}{2} \ .$$

Proof We first prove the security of $\mathsf{CTR}\left[F\right]$ against a quantum adversary $\mathcal{A}$ when F is replaced by a random function f. Then we look at $\mathcal{A}$'s probability of success when F is our given family of functions. Finally we reduce the security of the scheme to QPRF security of F.

Recall Construction 2. Consider the construction $\mathsf{CTR}\left[f\right]$ where the function F is replaced with a random function $f \leftarrow\!\!\$\, \mathsf{Func}\left(l, l\right)$. The function f takes distinct counter values as input. The output of f on successive counter values yields a truly random and unpredictable sequence of bits. This bit string is XORed with the message in an one-time pad fashion. Therefore the quantum adversary $\mathcal{A}$ does not gain any information about the encrypted messages. This is an information theoretic result which stands regardless of the computing power and computing time of the adversary. Hence,

$$\mathbf{Adv}^{\textit{lor-cpa}}_{\mathsf{CTR}[f]}\left(\mathcal{A}\right) = 0\ . \tag{4.6}$$

Next, we look at the security of $\mathsf{CTR}\left[F\right]$ where F is the given family of functions. The adversary plays the experiment $\mathbf{Exp}^{\textit{lor-cpa}}_{\mathsf{CTR}[F]}$. The adversary's advantage is:

$$\mathbf{Adv}^{\textit{lor-cpa}}_{\mathsf{CTR}[F]} = \Pr\left[\mathbf{Exp}^{\textit{lor-cpa-}1}_{\mathsf{CTR}[F]}\left(\mathcal{A}\right) = 1\right] - \Pr\left[\mathbf{Exp}^{\textit{lor-cpa-}0}_{\mathsf{CTR}[F]}\left(\mathcal{A}\right) = 1\right]\ . \tag{4.7}$$

Now assume $\mathcal{A}$ is a quantum adversary attacking LoR-CPA security of $\mathsf{CTR}\left[F\right]$. We construct a new quantum adversary $\mathcal{B}$, using $\mathcal{A}$, to attack QPRF security of F. $\mathcal{B}$ uses its oracle to provide a simulation of $\mathcal{A}$'s oracle.

The quantum adversary $\mathcal{B}$ runs $\mathcal{A}$. The adversary $\mathcal{B}$ maintains a counter (we assume that $\mathcal{B}$ does this perfectly) and a bit $d \leftarrow\!\!\$\ \{0, 1\}$. Then, upon receiving an encryption query (m_0, m_1) from $\mathcal{A}$, $\mathcal{B}$ queries its oracle on a counter value and XORs the result with the message m_d. It then sends the ciphertext to $\mathcal{A}$. Eventually $\mathcal{A}$ outputs a bit b'. The quantum adversary $\mathcal{B}$ outputs 1 if $b' = d$, otherwise it outputs 0.

When $b = 1$ we have:

$$\Pr\left[\mathbf{Exp}_F^{qprf\text{-}1}(\mathcal{B})=1\right]=\frac{1}{2}+\frac{1}{2}\cdot\mathbf{Adv}_{\mathsf{CTR}[F]}^{lor\text{-}cpa}(\mathcal{A}) \ . \tag{4.8}$$

And when $b=0$ we have:

$$\Pr\left[\mathbf{Exp}_F^{qprf\text{-}0}(\mathcal{B})=1\right]=\frac{1}{2}+\frac{1}{2}\cdot\mathbf{Adv}_{\mathsf{CTR}[f]}^{lor\text{-}cpa}(\mathcal{A}) \ . \tag{4.9}$$

Hence,

$$\begin{aligned}\mathbf{Adv}_F^{qprf}(\mathcal{B}) &= \Pr\left[\mathbf{Exp}_F^{qprf\text{-}1}(\mathcal{B})=1\right]-\Pr\left[\mathbf{Exp}_F^{qprf\text{-}0}(\mathcal{B})=1\right]\\ &= \frac{1}{2}\cdot\mathbf{Adv}_{\mathsf{CTR}[F]}^{lor\text{-}cpa}(\mathcal{A})-\frac{1}{2}\cdot\mathbf{Adv}_{\mathsf{CTR}[f]}^{lor\text{-}cpa}(\mathcal{A})\\ &= \frac{1}{2}\cdot\mathbf{Adv}_{\mathsf{CTR}[F]}^{lor\text{-}cpa}(\mathcal{A}) \ . \end{aligned} \tag{4.10}$$

This concludes the proof. The adversary $\mathcal{B}$ needs to query its oracle $q'=|m_b|/l$ times, which is equal to μ. $\mathcal{B}$ runs in time at most $t'=t+(q+l\mu)$ that is equal to the maximum running time of $\mathcal{A}$ plus the overhead for answering the encryption oracle queries. ▮

We now discuss LoR-CPA security of CTR\$ mode.

Theorem 14 (Security of CTR\$ Mode Using a QPRF) *Let $F:\{0,1\}^k\times\{0,1\}^l\rightarrow\{0,1\}^l$ be a block cipher. For any $\mathsf{CTR\$}[F]$ scheme, assume $\mathcal{A}$ is a quantum adversary attacking $\mathsf{CTR\$}[F]$ with a running time of at most t, making at most q queries to the encryption oracle, and the size of the classical output μ bits, and having advantage*

$$\mathbf{Adv}_{\mathsf{CTR\$}[F]}^{lor\text{-}cpa}(\mathcal{A})\geq\epsilon \ .$$

Then there exists a quantum adversary $\mathcal{B}$ attacking F with a running time of at most $t'=t+(q+l\mu)$, making at most $q'=\mu$ queries to the oracle, and having advantage

$$\mathbf{Adv}_F^{qprf}(\mathcal{B}) \geq \frac{1}{2}\left(\epsilon - \frac{\mu^2}{2^l}\right) .$$

Proof We first prove the security of $\mathsf{CTR\$}[F]$ against a quantum adversary $\mathcal{A}$ when F is replaced by a random function f. Then we look at $\mathcal{A}$'s probability of success when F is our given family of functions. Finally we reduce the security of the scheme to QPRF security of F.

Recall Construction 1. Consider the construction $\mathsf{CTR\$}[f]$ where $f \leftarrow\!\$\, \mathsf{Func}(l, l)$ is a random function. The function f takes counter values, that may not be distinct, as input. Let r is the initial encryption counter. The security is achieved (or precisely, the advantage of the adversary is 0) as long as each block of a message is XORed with the output of $f(r+i)$ where the value of $r+i$ was never taken by f as input. Therefore, this encryption has the same effect as encrypting with one-time pad. To prove LoR-CPA security of $\mathsf{CTR\$}[f]$ we explore the probability of the value $r+i$ repeating more than once, which would mean that the encryption could not be considered as one-time pad.

The adversary makes q oracle queries in the form of (m_0, m_1) where $|m_0| = |m_1|$. We use $(m_{i,0}, m_{i,1})$ to denote the i-th encryption query. Each $m_{i,0}$ or $m_{i,1}$ contains n_i number of blocks. We use $m_{i,b}[j]$ to denote the value of the j-th l-bit block $m_{i,b}$ where $b \in \{0, 1\}$. The challenge ciphertext is denoted by c_i. We can show the encryption of messages as

$$m_{i,b} = m_{i,b}[1]\, m_{i,b}[2] \cdots m_{i,b}[n_i] \quad \text{and} \tag{4.11}$$

$$c_i = (r_i, c_i[1]\, c_i[2] \cdots c_i[n_i]) \ , \tag{4.12}$$

where $i \in [q]$, and $r_i \leftarrow\!\$\, \{0,1\}^l$ is chosen by the encryption oracle. Now we define Col to be the event that the following $n_1 + \cdots + n_q$ values contain at least two values that are the same:

$$\begin{array}{llll} r_1, & r_1+1, & \cdots, & r_1+n_1-1 \\ r_2, & r_2+1, & \cdots, & r_2+n_2-1 \\ \vdots & & & \vdots \\ r_q, & r_q+1, & \cdots, & r_q+n_q-1 \; . \end{array} \tag{4.13}$$

Also we define $\neg\mathsf{Col}$ to be the event that the above values are all distinct. We can see that in LoR-CPA game, Col might happen regardless of which message is encrypted, because r is chosen by the encryption oracle independently of the encrypted message. Moreover, we can see that in the case of $\neg\mathsf{Col}$, the advantage of the adversary is 0, because the encryption oracle encrypts messages in a one-time pad fashion. We now calculate the advantage of the adversary if Col is true.

$$\begin{aligned} \mathbf{Adv}^{\textit{lor-cpa}}_{\mathsf{CTR\$}[f]}(\mathcal{A}) &= \Pr\left[\mathbf{Exp}^{\textit{lor-cpa-}1}_{\mathsf{CTR\$}[f]}(\mathcal{A})=1\right] - \Pr\left[\mathbf{Exp}^{\textit{lor-cpa-}0}_{\mathsf{CTR\$}[f]}(\mathcal{A})=1\right] \\ &= \Big(\Pr\left[\mathbf{Exp}^{\textit{lor-cpa-}1}_{\mathsf{CTR\$}[f]}(\mathcal{A})=1\middle|\mathsf{Col}\right]\cdot\Pr\left[\mathsf{Col}\right] \\ &\qquad +\Pr\left[\mathbf{Exp}^{\textit{lor-cpa-}1}_{\mathsf{CTR\$}[f]}(\mathcal{A})=1\middle|\neg\mathsf{Col}\right]\cdot\Pr\left[\neg\mathsf{Col}\right]\Big) \\ &\qquad - \Big(\Pr\left[\mathbf{Exp}^{\textit{lor-cpa-}0}_{\mathsf{CTR\$}[f]}(\mathcal{A})=1\middle|\mathsf{Col}\right]\cdot\Pr\left[\mathsf{Col}\right] \\ &\qquad +\Pr\left[\mathbf{Exp}^{\textit{lor-cpa-}0}_{\mathsf{CTR\$}[f]}(\mathcal{A})=1\middle|\neg\mathsf{Col}\right]\cdot\Pr\left[\neg\mathsf{Col}\right]\Big) \\ &= \Big(\Pr\left[\mathbf{Exp}^{\textit{lor-cpa-}1}_{\mathsf{CTR\$}[f]}(\mathcal{A})=1\middle|\mathsf{Col}\right] \\ &\qquad -\Pr\left[\mathbf{Exp}^{\textit{lor-cpa-}0}_{\mathsf{CTR\$}[f]}(\mathcal{A})=1\middle|\mathsf{Col}\right]\Big)\cdot\Pr\left[\mathsf{Col}\right] \\ &\le \Pr\left[\mathsf{Col}\right] \; . \end{aligned} \tag{4.14}$$

Above, the parenthesised term has an upper bound of 1. Now we need to calculate $\Pr\left[\mathsf{Col}\right]$. Recall Equation 4.13. We use Col_i to denote the event that a collision exists among the first i rows of Equation 4.13. We also use $\neg\mathsf{Col}_i$ to denote the event that no collision exists in the first i rows. Then we have:

$$\begin{aligned}
\Pr[\mathsf{Col}] &= \Pr[\mathsf{Col}_q] \\
&= \Pr[\mathsf{Col}_{q-1}] + \Pr[\mathsf{Col}_q|\neg\mathsf{Col}_{q-1}] \cdot \Pr[\neg\mathsf{Col}_{q-1}] \\
&\leq \Pr[\mathsf{Col}_{q-1}] + \Pr[\mathsf{Col}_q|\neg\mathsf{Col}_{q-1}] \\
&\leq \vdots \\
&\leq \Pr[\mathsf{Col}_1] + \sum_{i=2}^{q} \Pr[\mathsf{Col}_i|\neg\mathsf{Col}_{i-1}] \\
&= \sum_{i=2}^{q} \Pr[\mathsf{Col}_i|\neg\mathsf{Col}_{i-1}] \ . \qquad (4.15)
\end{aligned}$$

We are now required to find an upper bound for the probability of a collision upon receiving the i-th query, given that no collision happened in the first $i-1$ queries. We begin with a simple case when $i = 1, 2$. Upon receiving the first query, the probability of a collision is 0, because there is no previous row in Equation 4.13. Upon receiving the second query, we need to find out the probability that one of the values $r_2+1, \cdots, r_2+n_2$ is equal to one of the values in the first row $r_1+1, \cdots, r_1+n_1$. Note that r_1 is fixed. Therefore we can see that a collision can happen if and only if,

$$r_1 - n_2 + 1 \leq r_2 \leq r_1 + n_1 - 1 \ . \qquad (4.16)$$

Hence,

$$(r_1 + n_1 - 1) - (r_1 - n_2 + 1) + 1 = n_1 + n_2 - 1 \qquad (4.17)$$

choices of r_2 exist that could yield a collision. Then we can calculate the following probability,

$$\Pr[\mathsf{Col}_2|\neg\mathsf{Col}_1] \leq \frac{(n_1 + n_2 - 1)}{2^l} \ . \qquad (4.18)$$

Given this intuition, we now extend Equation 4.18 for the case where $2 \leq i \leq q$ and we assume that no collision happened in the first $i-1$ rows. A collision might happen between row i and each of the first $i-1$ rows, therefore:

$$\begin{aligned}\Pr\left[\mathsf{Col}_i|\neg\mathsf{Col}_{i-1}\right] &\leq \frac{(n_i+n_1-1)+(n_i+n_2-1)+\cdots+(n_i+n_{i-1}-1)}{2^l} \\ &= \frac{(i-1)\,n_i+n_{i-1}+\cdots+n_1-(i-1)}{2^l}\;. \end{aligned} \tag{4.19}$$

We drop the last negative term in the above equation to get:

$$\begin{aligned}\Pr\left[\mathsf{Col}\right] &\leq \sum_{i=2}^{q}\Pr\left[\mathsf{Col}_i|\neg\mathsf{Col}_{i-1}\right] \\ &\leq \sum_{i=2}^{q}\frac{(i-1)\,n_i+n_{i-1}+\cdots+n_1}{2^l}\;. \end{aligned} \tag{4.20}$$

Note that in the above equation, n_i occurs with weight $i-1$ in the i-th term of the sum. Also it occurs with weight 1 in the j-th term of the sum where $j=i+1,\ldots,q$. Therefore its total weight is $(i-1)+(q-i)=q-1$, so we get:

$$\Pr\left[\mathsf{Col}\right] = \frac{(q-1)\,(n_1+\cdots+n_q)}{2^l}\;. \tag{4.21}$$

Finally the advantage of the adversary is

$$\begin{aligned}\mathbf{Adv}^{\textit{lor-cpa}}_{\mathsf{CTR\$}[f]}\left(\mathcal{A}\right) &\leq \Pr\left[\mathsf{Col}\right] \\ &\leq \frac{(q-1)\,(n_1+\cdots+n_q)}{2^l} \\ &\leq \frac{\mu^2}{2^l}\;. \end{aligned} \tag{4.22}$$

Next, we look at the security of $\mathsf{CTR\$}\left[F\right]$ where F is the given family of functions. The adversary plays LoR-CPA game. The adversary's advantage is:

$$\mathbf{Adv}^{\textit{lor-cpa}}_{\mathsf{CTR\$}[F]}(\mathcal{A}) = \Pr\left[\mathbf{Exp}^{\textit{lor-cpa-1}}_{\mathsf{CTR\$}[F]}(\mathcal{A}) = 1\right] - \Pr\left[\mathbf{Exp}^{\textit{lor-cpa-0}}_{\mathsf{CTR\$}[F]}(\mathcal{A}) = 1\right] . \tag{4.23}$$

Now assume $\mathcal{A}$ is a quantum adversary attacking LoR-CPA security of $\mathsf{CTR\$}[F]$. We construct a new quantum adversary $\mathcal{B}$, using $\mathcal{A}$, to attack QPRF security of F. $\mathcal{B}$ uses its oracle to provide a simulation of $\mathcal{A}$'s oracle.

The quantum adversary $\mathcal{B}$ runs $\mathcal{A}$. The adversary $\mathcal{B}$ chooses a bit $d \leftarrow\$ \{0,1\}$. Then, upon receiving an encryption query (m_0, m_1), $\mathcal{B}$ queries its oracle on $(r+i)$ where $i \in [n]$, $r \leftarrow\$ \{0,1\}^l$, and then XORs the result with the message m_d. It then sends the ciphertext to $\mathcal{A}$. The adversary $\mathcal{B}$ chooses a fresh r for each query. Here we assume that $\mathcal{B}$ simulates the encryption oracle for $\mathcal{A}$ perfectly. Eventually $\mathcal{A}$ outputs a bit b'. The adversary $\mathcal{B}$ outputs 1 if $b' = d$, otherwise it outputs 0.

When $b = 1$ we have:

$$\Pr\left[\mathbf{Exp}^{\textit{qprf-1}}_{F}(\mathcal{B}) = 1\right] = \frac{1}{2} + \frac{1}{2} \cdot \mathbf{Adv}^{\textit{lor-cpa}}_{\mathsf{CTR\$}[F]}(\mathcal{A}) . \tag{4.24}$$

And when $b = 0$ we have:

$$\Pr\left[\mathbf{Exp}^{\textit{qprf-0}}_{F}(\mathcal{B}) = 1\right] = \frac{1}{2} + \frac{1}{2} \cdot \mathbf{Adv}^{\textit{lor-cpa}}_{\mathsf{CTR\$}[f]}(\mathcal{A}) . \tag{4.25}$$

Hence,

$$\begin{aligned}\mathbf{Adv}^{\textit{qprf}}_{F}(\mathcal{B}) &= \Pr\left[\mathbf{Exp}^{\textit{qprf-1}}_{F}(\mathcal{B}) = 1\right] - \Pr\left[\mathbf{Exp}^{\textit{qprf-0}}_{F}(\mathcal{B}) = 1\right] \\ &= \frac{1}{2} \cdot \mathbf{Adv}^{\textit{lor-cpa}}_{\mathsf{CTR\$}[F]}(\mathcal{A}) - \frac{1}{2} \cdot \mathbf{Adv}^{\textit{lor-cpa}}_{\mathsf{CTR\$}[f]}(\mathcal{A}) \\ &= \frac{1}{2}\left(\mathbf{Adv}^{\textit{lor-cpa}}_{\mathsf{CTR\$}[F]}(\mathcal{A}) - \frac{\mu^2}{2^l}\right) . \end{aligned} \tag{4.26}$$

This concludes the proof. The adversary $\mathcal{B}$ needs to query its oracle $|m_b|/l = \mu$ times. $\mathcal{B}$ runs in time at most t' that is equal to the maximum running time of $\mathcal{A}$

plus the overhead for answering the encryption oracle queries. ▮

We note that proofs of Theorems 13 and 14, given above, are similar to their classical counterparts. In other words, classical security proofs of CTR\$ mode and CTR mode carry over to a quantum setting where, quantum computation attacks are considered. If we look at the characterisation of these types of security proofs, then we can break them into two parts. In the first part, the security of an idealised scheme is assessed. This analysis is merely a probabilistic process and the type of adversary, whether it is classical or quantum, has nothing to do with it. The second part is a reduction. In this, the advantage of the adversary distinguishing between the ideal scheme and the real scheme which we are interested in its security, is bounded by the advantage of the adversary breaking the underlying primitive using comparable resources. This is where the type of the adversary matters and one must consider possible attacks that are unique to a quantum adversary.

More rigorously, Theorems 13 and 14 can be seen as *black-box reductions* [69]. That is, if the theorem states that the security of P implies the security of S, then S can be constructed from P, merely using P as a black-box and regardless of the specifics of how P works. Moreover, the security reduction is also black-box. Because, an algorithm for breaking P can be constructed from a black-box for breaking S.

To elaborate, and for the sake of concrete security framework, we give an example. Consider CTR\$ mode in Construction 1. According to Result 11, the security bound of this scheme against a classical adversary A is:

$$\mathbf{Adv}^{lor\text{-}cpa}_{\mathsf{CTR\$}[F]}(A) \leq 2 \cdot \mathbf{Adv}^{prf}_{F}(B) + \frac{\mu^2}{2^l} . \tag{4.27}$$

Theorem 14 shows the security bound of this scheme against an adversary $\mathcal{A}$ that can mount quantum computation attack. This is:

$$\mathbf{Adv}^{lor\text{-}cpa}_{\mathsf{CTR\$}[F]}(\mathcal{A}) \leq 2 \cdot \mathbf{Adv}^{qprf}_{F}(\mathcal{B}) + \frac{\mu^2}{2^l} . \tag{4.28}$$

Now suppose the function $F : \{0,1\}^l \to \{0,1\}^l$ where $l = 128$. Also assume the adversary, whether classical or quantum, makes $q = 2^{30}$ encryption oracle queries. If each query is 2^{13} bits long (which is a kilobyte) then the total amount of encrypted data is 2^{43} bits which is $\mu = 2^{36}$ 128-bit blocks. The question is whether $\mathsf{CTR\$}[F]$ mode is secure against the adversary, given this information. To calculate the advantage of the adversary, we need to calculate prf-advantage or qprf-advantage of F. We first consider the classical case. Result 11 tells us that the security of the scheme against a classical adversary A is bounded above by the prf-advantage of F against another classical adversary B. The classical adversary B makes $q = 2^{36}$ queries, and to best of our knowledge, the best known classical attack against PRF security of F is the birthday attack. Therefore we assume B's advantage is no more than $q^2/2^{128}$. Hence we can get:

$$\begin{aligned}\mathbf{Adv}^{\textit{lor-cpa}}_{\mathsf{CTR\$}[F]}(A) &\leq 2 \cdot \frac{\mu^2}{2^{128}} + \frac{\mu^2}{2^{128}} \\ &\leq \frac{1}{2^{55}} \, . \end{aligned} \tag{4.29}$$

Now we consider the quantum case. Analogously, Theorem 14 tells us that the security of the scheme against a quantum adversary $\mathcal{A}$ is bounded above by the qprf-advantage of F against another quantum adversary $\mathcal{B}$. Similarly to the classical case, the best attack the adversary $\mathcal{B}$ can mount against QPRF security of F is the birthday attack. Therefore we get the same security bound as above for the quantum adversary:

$$\begin{aligned}\mathbf{Adv}^{\textit{lor-cpa}}_{\mathsf{CTR\$}[F]}(\mathcal{A}) &\leq 2 \cdot \frac{\mu^2}{2^{128}} + \frac{\mu^2}{2^{128}} \\ &\leq \frac{1}{2^{55}} \, . \end{aligned} \tag{4.30}$$

Note, one might think that the best quantum attack on QPRF security of F is the quantum collision finding algorithm [34, 33]. In Subsection 3.2.2, we explained that the work of Brassard *et al.* [34, 33], based on Grover's algorithm, gives a quantum algorithm to find collisions in an arbitrary r-to-one function after $O\left(\sqrt[3]{N/r}\right)$ oracle queries. Hence, to find a collision for a two-to-one function F, one only needs $O\left(\sqrt[3]{N}\right)$ evaluations of the function F. With regards to the latter and Equation 3.25, one could say that $\mathcal{B}$'s advantage is no more than $4 \cdot q^3/2^l$, under the assumption that the best attack the quantum adversary $\mathcal{B}$ can mount against QPRF security of F is the quantum collision finding algorithm. Therefore in the case of F, where $l = 128$, we would get:

$$\begin{aligned}
\mathbf{Adv}^{lor\text{-}cpa}_{\mathsf{CTR\$}[F]}(\mathcal{A}) &\leq 8 \cdot \frac{\mu^3}{2^{128}} + \frac{\mu^2}{2^{128}} \\
&\leq 8 \cdot \frac{2^{108}}{2^{128}} + \frac{2^{72}}{2^{128}} \\
&\leq \frac{1}{2^{17}} \,. \qquad (4.31)
\end{aligned}$$

Equation 4.31 suggests that $\mathsf{CTR\$}\,[F]$ provides little security against this quantum adversary. However, we emphasize that this is not true. The collision finding algorithm [34, 33] uses Grover's algorithm to evaluate the function. To do so, the algorithm needs superposition access to the function. In this case the adversary needs superposition access to F. Since the adversary is only given classical access to

this example we can deduce that the security bounds of $\mathsf{CTR\$}$ mode are the same for both classical and quantum adversaries. The same follows for CTR mode.

A natural question regarding the above would be whether all similar classical security proofs carry over to this quantum setting. Or what class of classical security proofs carry over to this quantum setting. There are a number of works with regards to this question. For instance, Crépeau [41] and Yao [112] showed that the quantum oblivious transfer can be seen as a construction of quantum oblivious transfer from a black-box for bit commitment. Damgaard and Lunemann [44], and Lunemann and Nielsen [84] prove that a few classical protocols are quantum secure. Hallgren *et al.* [67] formalise a family of classical security proofs that carry over to the quantum setting against efficient quantum adversaries under reasonable computational assumptions. Moreover, Watrous [109] and Unruh [106] discuss quantum zero-knowledge and quantum proofs of knowledge, respectively. In the case of a classical symmetric encryption scheme, we believe the classical security proofs carry over as we have shown it for $\mathsf{CTR\$}$ mode and CTR mode. But a general formalisation of that needs further work and we leave it as an open problem.

4.3 Message Authentication Code

Until now, we have explored cryptographic ways and tools that enable us to achieve confidentiality. That is, encryption can help two parties to establish a private communication which prevents an eavesdropper or an active adversary from gaining partial information about messages sent over an unprotected communication channel. However, this does not guarantee the identity of the origin of encrypted messages. Moreover, this does not prevent adversaries from tampering with encrypted messages as long as the results decrypt to valid messages. For example, assume we want to send an order to our bank to transfer some money to another account. All the communication between us is encrypted using the secret key we have shared with the bank. It turns out that privacy is not enough to protect our order, as an adversary can tamper with ciphertexts sent to the bank. For instance, it can flip some bits in ciphertexts which then might directly effects the corresponding bits in the decrypted messages. In this way, the adversary might be able to, say, change the amount of money in our order. It does not know the new amount but it surely has changed our original order. Therefore, upon receiving an order, the bank's goal should be to check two things. First, did the order really comes from us? And second, is it the exact order that we issued? These goals are called *message integrity* (or *message authentication*). And *message authentication code* (MAC) is a mechanism to achieve it.

Here we consider symmetric message authentication. Formally, a message authentication scheme $\mathcal{MA} = (\mathcal{K}, \mathcal{T}, \mathcal{V})$ consists of three algorithms. The randomised key generation algorithm $\mathcal{K}$ takes no input and output a key K. The tagging algorithm $\mathcal{T}$, which may be randomised or stateful, takes a key $K \in \mathcal{K}$ and an arbitrary length message m from the message space $\mathcal{M}$ as input, and returns a tag $\tau \in \{0,1\}^*$. The deterministic verification algorithm $\mathcal{V}$ takes as input the secret key K, the message m, and a candidate tag τ', and returns a bit v. For completeness, we require that for any key $K \in \mathcal{K}$ and any message $m \in \mathcal{M}$

$$\Pr\left[\tau \leftarrow \mathcal{T}_K\left(m\right) : \mathcal{V}_K\left(m, \tau\right) = 1\right] = 1 \; . \tag{4.32}$$

We merely consider message authentication schemes whose tagging algorithm is deterministic and stateless such as MAC. A generally accepted security definition for MACs is called *existential unforgeability under an adaptive chosen message attack*. That is, an adversary using reasonable resources should not be able to create a valid tag for a new message that was not previously tagged (or authenticated) by honest parties. In this security model, the adversary is given access to a MAC oracle which it can query on any message to see the corresponding tag. A message authentication scheme is considered broken if the adversary can produce a valid tag τ for a message m where $\mathcal{V}_K\left(m, \tau\right) = 1$ and the message m was not queried to MAC oracle before.

Boneh and Zhandry [30] give a security definition for quantum-secure MACs. That is, an adaptation of the existential unforgeability notion where a quantum adversary is given quantum superposition access to the MAC oracle, but it submits classical pairs of (m, τ) to the challenger. A MAC is said to be *quantum-secure* if after q queries to the MAC oracle, the adversary cannot submit $q + 1$ valid and distinct classical pairs of (m, τ) to the challenger.

In general, a MAC can be constructed from a PRF where the PRF takes the role of the tagging algorithm. In practice, this approach leads to constructions such as HMAC [11] and CBC-MAC [2].

4.4 Authenticated Encryption

We discussed how to separately achieve the cryptographic goals of privacy and integrity. But there are scenarios where we need both simultaneously. For instance, the example that we discussed in the previous section about communicating with our bank is one scenario where both confidentiality and authenticity of data is required. Encryption schemes include authenticity assurances are called *authenticated encryption*. In practice, there are many protocols, such as SSL/TLS [49] and IPSec [73], that use authenticated encryption to provide secure private communication. Many attacks on these protocols are due to misuse of authenticated encryption schemes or lack of it [45].

A simple way to construct an authenticated encryption scheme is to combine an encryption scheme with a MAC. Bellare and Namprempre [17] formally analyse the security of three generic compositions of a given symmetric encryption and a given MAC. The compositions are: *Encrypt-and-MAC* (EaM), *MAC-then-Encrypt* (MtE), and *Encrypt-then-MAC* (EtM).

In EaM, the sender produces a MAC tag for the message and sends it along with the ciphertext of the message to the receiver. The receiver first decrypts the ciphertext and then checks whether the tag verifies correctly on the resulting message. If so, it returns the message. Otherwise it returns $\perp$. In MtE, the sender produces a MAC tag of the message, then concatenates the tag and the message together and encrypts the result. The receiver decrypts the ciphertext to recover the message and its tag. If the tag verifies correctly on the message, then the receiver outputs the message. Otherwise it outputs $\perp$. In EtM, the sender first encrypts the message, then produces a MAC tag on the ciphertext. The receiver checks whether the tag verifies correctly on the ciphertext. If so, it decrypts the ciphertext and outputs the resulting message. Otherwise it returns $\perp$. Bellare and Namprempre [17] show that EtM provides LoR-CCA security, given that both the encryption and authentication schemes meet the required security properties.

Chapter 5

Quantum Superposition Attacks on the Even-Mansour Scheme

Contents

In this chapter, we show how powerful a quantum adversary might get when it is given quantum superposition access to its oracles. While block ciphers are considered to be secure against quantum computation attacks, we illustrate that a class of them will not provide any security whatsoever if the quantum adversary is given quantum superposition access to them. We first discuss an extremely simple block cipher and a known classical attack on it. Then we show how we can exploit this attack in a quantum setting. Finally, we discuss an extension of our attack to apply to other variants of block ciphers.

5.1 The Even-Mansour Scheme

Block ciphers, such as AES (see Subsection 4.1.1), mostly have an iterated structure. Their structure consists of XORing a secret key with their internal state, and then applying some publicly known permutation that is chosen randomly. The number of

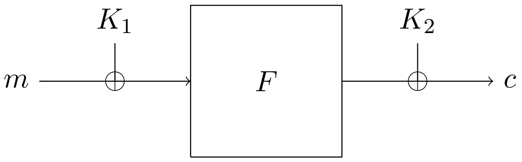

Figure 5.1: The Even-Mansour scheme

iterations varies depending on each specific block cipher and the security properties they offer.

Even and Mansour [57] defined and analysed the simplest possible construction of a block cipher. In the *Even-Mansour scheme* (EM), depicted in Figure 5.1, the ciphertext is obtained by first XORing the plaintext with an n-bit key K_1, then applying a publicly known random permutation F and XORing the output with a second n-bit key K_2, *i.e.*,

$$E(m) = F(m \oplus K_1) \oplus K_2 \, . \tag{5.1}$$

Even and Mansour assumed that the adversary is allowed to perform two types of queries: (i) queries to a full encryption/decryption oracle that computes either $E(m)$ or $E^{-1}(m)$; and (ii) queries to a permutation oracle that computes either $F(m)$ or $F^{-1}(m)$. Given this assumption, they proved that in order to attack the scheme with a given probability of success, one must have $DT = O(2^n)$, where D is the number of queries to the encryption/decryption oracle and T is the number of queries to the F-oracle. Even and Mansour gave a lower bound for the number of queries needed to break their scheme, thus providing a formal security proof. Moreover, Dunkelman *et al.* [54] showed that EM can even be further simplified into a single-key variant with half as many key bits, while still having exactly the same provable security.

Despite its simplicity, the EM scheme is not merely a theoretical construct, but is implicit in other ciphers. For instance, there are other works that study the security of iterated EM with more than one round [78, 50, 25]. These works, in their security analysis, consider different numbers of rounds and keys. A similar construction is also used to construct *tweakable* block ciphers [79]. Moreover, a generalised variant of EM known as *key-alternating cipher* is given by Daemen and Rijmen [43]. A general

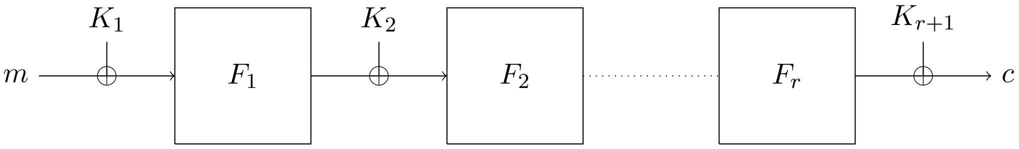

Figure 5.2: An r-round key-alternating cipher

r-round key-alternating cipher, depicted in Figure 5.2, consists of $F_1, \ldots, F_r$ public random permutations and $r + 1$ distinct secret n-bit keys $K_1, \ldots, K_{r+1}$,

$$E(m) = F_r(F_{r-1}(\cdots F_2(F_1(m \oplus K_1) \oplus K_2) \cdots) \oplus K_r) \oplus K_{r+1} \ . \tag{5.2}$$

Bogdanov *et al.* [28] give a formal security proof for the latter scheme, demonstrating that an adversary needs to make at least $2^{2n/3}$ queries to the underlying permutations to be able to distinguish this scheme from random.

5.2 Slide with a Twist Attack

Biryukov and Wagner [26] introduce a cryptanalytic attack, called *slide attack*, to break iterated cryptosystems with an arbitrary number of rounds by exploiting their self similarity under small shift. They then extend the basic slide attack to make it applicable to larger classes of ciphers. One of the extended methods introduced by Biryukov and Wagner [27] is called the *slide with a twist* attack. They describe the slide with a twist attack on the EM scheme, and show that it achieves the lower bound up to a factor of $\sqrt{2}$. Here we explain the main idea of the slide with a twist attack on the EM scheme, which is also discussed in work of Dunkelman *et al.* [54].

Assume that two plaintexts m, m' satisfy

$$m \oplus m' = K_1 \ . \tag{5.3}$$

Therefore we can write their encryptions as,

$$E\left(m\right)=F\left(m\oplus K_1\right)\oplus K_2=F\left(m'\right)\oplus K_2\text{ , and} \tag{5.4}$$

$$E\left(m'\right)=F\left(m'\oplus K_1\right)\oplus K_2=F\left(m\right)\oplus K_2\text{ .} \tag{5.5}$$

By XORing the above equations (see also Figure 5.3) we get

$$E\left(m\right)\oplus E\left(m'\right)=F\left(m\right)\oplus F\left(m'\right)\text{ ,} \tag{5.6}$$

or equivalently,

$$E\left(m\right)\oplus F\left(m\right)=E\left(m'\right)\oplus F\left(m'\right)\text{ .} \tag{5.7}$$

Given these relations, an adversary can query the E-oracle and F-oracle on the same $2^{(n+1)/2}$ values of known plaintexts $m_1, m_2, \ldots$. The adversary then stores the results of $E\left(m_i\right)\oplus F\left(m_i\right)$ in a table sorted by this value. The adversary looks for collisions $E\left(m_i\right)\oplus F\left(m_i\right)=E\left(m_j\right)\oplus F\left(m_j\right)$. When found, it checks the guess $K_1=m_i\oplus m_j$ and $K_2=E\left(m_i\right)\oplus F\left(m_j\right)$.

Each pair of plaintexts $\left(m_i, m_j\right)$ that satisfies $m_i\oplus m_j=K_1$ is called a *slid pair*. The probability that the collision happens for a random pair of plaintexts is 2^{-n}. Therefore the table is expected to contain only a few collisions such that with regards to the birthday paradox at least one of them with high probability is induced by the slid pair which yields the correct values of K_1 and K_2. The data complexity of the attack is $DT=2^{n+1}$ where $D=2^{(n+1)/2}$ is the number of known plaintexts and $T=2^{(n+1)/2}$ is the number of queries to the F-oracle.

5.3 Quantum Superposition Attack

Assuming the existence of a scalable quantum computer, Shor's algorithm [103] breaks the most widely used public key encryption schemes, including RSA [94].

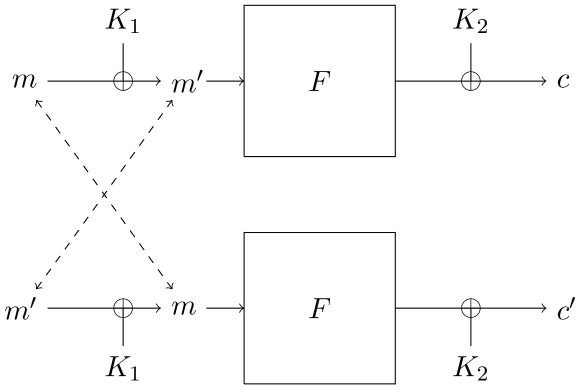

Figure 5.3: The slide with a twist attack on the EM scheme

On the other hand, quantum computing appears to have very little impact on symmetric cryptography. The generic quantum attack on block ciphers using Grover's algorithm [65] requires $O(2^{n/2})$ queries for key length n and thus can be countered by doubling the key length.

To mount a generic Grover attack on a block cipher, an adversary does not need access to an encryption oracle, but only to (i) a single valid plaintext/ciphertext pair, and (ii) an implementation of the encryption algorithm on a quantum computer. Since any classical algorithm can be converted efficiently into a quantum algorithm (see Subsection 3.1.1), an adversary in possession of a scalable quantum computer can satisfy requirement (ii) as long as the encryption algorithm is publicly known.

In contrast to the above, here we assume a security model where the adversary is given quantum superposition access to an encryption oracle. See Section 3.4 for the definition of quantum adversary. So far there has been little discussion, however, of the security of existing symmetric schemes in this security model. Our work is a contribution to this question. We show that some specific symmetric constructions offer no security at all against an adversary with superposition access to an encryption oracle.

A prerequisite for superposition access is that the encryption oracle must be implemented on a quantum computer. Our result therefore poses no threat to existing classical implementations of block ciphers. More generally, the security model on which our result is based is of no practical relevance for present-day security environments. It is conceivable that this could change in a future environment (the ‘quantum internet’) in which communication channels between quantum computers need to be secured. Or, classical encryption schemes running on a quantum computer might need to be secured against, say, quantum malwares running on the same quantum computer. At present, the main interest of our result is that it establishes that some specific symmetric schemes are vulnerable against quantum adversaries independently of the generic attack using Grover’s algorithm.

Our attack against the Even-Mansour scheme makes use of a slight generalisation of Simon’s algorithm [104]. Simon’s algorithm is also at the heart of the quantum related key attack against a general block cipher discovered by Rötteler and Steinwandt [98]. They show that the cipher’s secret key can be extracted efficiently if the quantum adversary is allowed to query superposition of related keys. In contrast, in our attack, the quantum adversary queries superposition of messages, but the attack works only against specific schemes. Our work is done independently from work of Kuwakado and Morii [77] which also discusses security of the EM scheme if run on a quantum computer. Both works exploit Simon’s algorithm to break the EM scheme. However, Kuwakado and Morii do not question the assumption that Simon’s algorithm can be used, while we calculate the precise probability of getting a slid pair (see Subsection 5.3.2), showing that Simon’s problem is only partially satisfied. We also extend our results to two variants of iterated EM with more than one round.

5.3.1 Quantum Oracle for the Even-Mansour Scheme

The aim of our quantum attack will be to recover the secret key K_1. Since $K_2 = E(m) \oplus F(m \oplus K_1)$ and E and F are known, finding K_2 is trivial once K_1 is known. We assume that the quantum adversary is allowed to make superposition queries to both the encryption oracle $E(m)$ and the permutation oracle $F(m)$. Formally this means that the two oracles act as unitary transformations satisfying

$$|m\rangle \otimes |0\rangle \longrightarrow |m\rangle \otimes |E(m)\rangle \quad \text{and} \quad |m\rangle \otimes |0\rangle \longrightarrow |m\rangle \otimes |F(m)\rangle \tag{5.8}$$

for all computational basis states $|m\rangle$. The action of the encryption oracle on an arbitrary superposition with coefficients c_m is then

$$\sum_m c_m |m\rangle \otimes |0\rangle \longrightarrow \sum_m c_m |m\rangle \otimes |E(m)\rangle \tag{5.9}$$

and similarly for the permutation oracle.

5.3.2 Partially Satisfying the Assumptions of Simon's Problem

Our quantum attack is based on Simon's problem which we explained in Subsection 3.2.1. In order for Simon's algorithm to give us the desired answer for K_1, we exclude $K_1 = 0^n$ from the set of possible values for K_1. We make use of the following fact, that is given by the slide with a twist attack on the Even-Mansour scheme. Define the function

$$X(m) = F(m) \oplus E(m) \ . \tag{5.10}$$

Since

$$X(m) = F(m) \oplus F(m \oplus K_1) \oplus K_2 \ , \tag{5.11}$$

we have that, for all $m \in \{0,1\}^n$,

$$X(m \oplus K_1) = X(m) \ . \tag{5.12}$$

The function X thus satisfies part of the assumptions made in Simon's problem. To fully satisfy the assumptions of Simon's problem, one also needs that $X(m) = X(m')$ implies $m' \in \{m, m \oplus K_1\}$. This is not true in our case because, for any given string l, there can be more than two solutions to

$$X\left(m\right) = l\ . \tag{5.13}$$

The solutions to Equation 5.13 do come in pairs $\{m, m \oplus K_1\}$, however. Let M be a subset of $\{0,1\}^n$ of size 2^{n-1} such that

$$\{0,1\}^n = \bigcup_{m \in M} \{m, m \oplus K_1\}\ . \tag{5.14}$$

Equation 5.11 and the fact that the permutation F is chosen randomly imply that the probability that $X\left(m\right) = l$ for given $m \in M$ and $l \in \{0,1\}^n$ is equal to 2^{-n}. Assuming that $X\left(m\right)$ can be approximated by a random function (see [88] for a justification), the probability p_1 that Equation 5.13 has exactly one solution $m \in M$ is therefore given by

$$p_1 = 2^{n-1}\, 2^{-n} \left(1 - 2^{-n}\right)^{2^{n-1}-1} \simeq \frac{1}{2\sqrt{e}}\ . \tag{5.15}$$

This equation holds for any $K_1 \neq 0^n$. Similarly, the probability p_r that Equation 5.13 has exactly r solutions $m \in M$ can be found [88] to be approximately

$$p_r \simeq \frac{1}{2^r\, r!\, \sqrt{e}}\ . \tag{5.16}$$

We have $p_1 > 0.3$ for any value of n. This means that, for given l, the probability that Equation 5.13 has exactly two solutions m and $m \oplus K_1$ is greater than 0.3. It turns out that the existence of this bound allows us to apply Simon's algorithm to the problem of extracting the key K_1.

5.3.3 The Quantum Attack

The quantum adversary begins by preparing four n-qubit registers in the state

$$|\psi_1\rangle = \frac{1}{\sqrt{2^n}} \sum_{m\in\{0,1\}^n} |m\rangle \otimes |0^n\rangle \otimes |0^n\rangle \otimes |0^n\rangle \ . \tag{5.17}$$

Applying first the permutation oracle and then the encryption oracle to the appropriate registers results in the state

$$|\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{m\in\{0,1\}^n} |m\rangle \otimes |F\left(m\right)\rangle \otimes |E\left(m\right)\rangle \otimes |0^n\rangle \ . \tag{5.18}$$

Applying XOR to the second and third register and placing the result in the fourth register gives

$$|\psi_3\rangle = \frac{1}{\sqrt{2^n}} \sum_{m\in\{0,1\}^n} |m\rangle \otimes |F(m)\rangle \otimes |E(m)\rangle \otimes |X(m)\rangle \ . \tag{5.19}$$

Now let r_l denote the number of solutions $m \in M$ to Equation 5.13 and define

$$L_r = \{l \in \{0,1\}^n \ : \ r_l = r\}. \tag{5.20}$$

The expected value of $|L_r|$ is given by

$$\mathsf{E}\left(|L_r|\right) = p_r \, 2^n \ , \tag{5.21}$$

which decreases rapidly as r increases, and is effectively zero for $r \geq n$. We can rewrite the state in terms of the set L_r as follows:

$$|\psi_3\rangle = \frac{1}{\sqrt{2^n}} \sum_{r>0} \sum_{l\in L_r} \sum_{i=1}^{r} \Big(|m_i^l\rangle + |m_i^l \oplus K_1\rangle \Big) \otimes |F(m_i^l)\rangle \otimes |E(m_i^l)\rangle \otimes |l\rangle \ , \tag{5.22}$$

where $m_i^l \in M$ and $X(m_i^l) = X(m_i^l \oplus K_1) = l$ for all l and $i = 1, \ldots, r_l$.

The next step is a measurement of the fourth register in the computational basis. We denote the measurement outcome by l^*. The state of the first register after the measurement is

$$|\psi_4\rangle = \frac{1}{\sqrt{2r^*}} \sum_{i=1}^{r^*} (|m_i^*\rangle + |m_i^* \oplus K_1\rangle) \ , \tag{5.23}$$

where we have used the abbreviated notation $r^* = r_{l^*}$ and $m_i^* = m_i^{l^*}$. For any $r \geq 1$, the probability $\Pr(r^* = r)$ of getting an outcome l^* such that $r^* = r$ is equal to 2^{-n} times the expectation value of the number of terms in the sum (see Equation 5.22) for the given value of r. We have thus

$$\Pr(r^* = r) = 2^{-n}\, \mathsf{E}(|L_r|) \times 2r = 2rp_r \simeq \frac{1}{2^{r-1}\,(r-1)!\,\sqrt{e}} \ . \tag{5.24}$$

In particular, $\Pr(r^* = 1) = 2p_1 \geq 0.6$. Now the adversary applies the n-qubit Hadamard transformation to the first register, resulting in the state

$$\begin{aligned}
|\psi_5\rangle &= \frac{1}{\sqrt{2r^*\,2^n}} \sum_{i=1}^{r^*} \Big(\sum_{a\in\{0,1\}^n} (-1)^{m_i^* \cdot a} |a\rangle + \sum_{a\in\{0,1\}^n} (-1)^{(m_i^* \oplus K_1)\cdot a} |a\rangle \Big) && (5.25)\\
&= \frac{1}{\sqrt{r^*\,2^{n+1}}} \sum_{i=1}^{r^*} \sum_{a\in\{0,1\}^n} (-1)^{m_i^* \cdot a} \Big(1 + (-1)^{K_1\cdot a}\Big) |a\rangle && (5.26)\\
&= \frac{1}{\sqrt{r^*\,2^{n-1}}} \sum_{i=1}^{r^*} \sum_{a: K_1\cdot a = 0} (-1)^{m_i^* \cdot a} |a\rangle \ . && (5.27)
\end{aligned}$$

The last step is a measurement of the first register in the computational basis. As in the standard Simon algorithm, we are guaranteed to obtain a bit string a such that

$$K_1 \cdot a = 0 \mod 2 \ . \tag{5.28}$$

In the standard version of Simon's problem, we always have that $r^* = 1$. This means that the string a resulting from the measurement is random, subject to the constraint

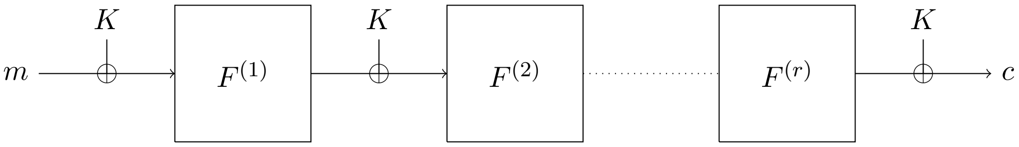

Figure 5.4: The r-round EM scheme with a single permutation and identical round keys

given by Equation 5.28. The algorithm is run repeatedly until among the strings a so obtained, there are $n-1$ linearly independent ones. The key K_1 can then be extracted from the system of linear Equations 5.28 using Gaussian elimination. Given a set of strings a which span a subspace of dimension less than $n-1$, the probability that the next (random) string is outside that subspace is at least $1/2$. This means that $O(n)$ repetitions of Simon's algorithm will, with probability exponentially close to 1, result in a set of equations that determines K_1.

In the Even-Mansour case, for every run of the algorithm, we also get a string a such that Equation 5.28 is satisfied. Whenever r^* turns out to be equal to 1, which happens with probability greater than 0.6, the string a will be random. This means that the analysis of the previous paragraph still applies: given a set of strings which span a subspace of dimension less than $n-1$, the probability that the next string returned by the algorithm is outside that subspace is now bounded below by 0.3. After $O(n)$ repetitions of the algorithm we will, with probability exponentially close to one, have a set of equations that determines K_1.

5.3.4 Generalisation to Multiple Rounds

A natural question is whether our $O(n)$ attack extends to more general ciphers with multiple rounds. Although the answer appears to be 'no' in general, it turns out that our attack can be applied to two special cases of the multiple-round Even-Mansour scheme. These are the case of arbitrarily many rounds using a single permutation and identical round keys, and the case of two rounds using a single permutation and round keys derived from a very basic key schedule. Chen *et al.* [36] recently described slide with a twist attacks against these schemes.

Consider first the r-round Even-Mansour scheme with a single permutation F and identical round keys as shown in Figure 5.4. The encryption of an arbitrary message m is

$$E(m) = F^{(r)}(F^{(r-1)}(\cdots F^{(2)}(F^{(1)}(m \oplus K) \oplus K) \cdots) \oplus K) \oplus K \; , \tag{5.29}$$

where K denotes the common key, and

$$F^{(1)} = F^{(2)} = \ldots = F^{(r)} = F \tag{5.30}$$

denote identical permutations, labelled to distinguish between rounds for clarity. We have

$$\begin{aligned} F(E(m)) &= F(F^{(r)}(F^{(r-1)}(\cdots F^{(2)}(F^{(1)}(m \oplus K) \oplus K) \cdots) \oplus K) \oplus K) \\ &= F^{(r)}(F^{(r-1)}(\cdots F^{(1)}(F(m \oplus K) \oplus K) \cdots) \oplus K) \oplus K) \\ &= E(F(m \oplus K)) \oplus K \; , \end{aligned} \tag{5.31}$$

where we have relabelled the permutations using Equation 5.30. Now define

$$X(m) = E(F(m)) \oplus F(E(m)) \; . \tag{5.32}$$

Given access to both E and F oracles, $X(m)$ can be evaluated by the adversary. Using Equation 5.31, this gives

$$X(m) = E(F(m)) \oplus E(F(m \oplus K)) \oplus K \; . \tag{5.33}$$

Therefore $X(m) = X(m \oplus K)$. The rest of the analysis and the details of the quantum attack are almost identical to the single round case above. As before, the key K can be recovered with constant probability using $O(n)$ queries.

We now move on to the single-permutation two-round Even-Mansour scheme with a key-schedule where the round keys K_0, K_1, K_2 are derived from a secret n-bit master key K and public n-bit constants t_0, t_1, t_2 via a simple XOR, $K_i = K \oplus t_i$. This is depicted in Figure 5.5.

The encryption of an arbitrary message m is

$$E(m) = F(F(m \oplus K_0) \oplus K_1) \oplus K_2 \ . \tag{5.34}$$

We have

$$\begin{aligned} E(F(m) \oplus t_0 \oplus t_1) &= F(F(F(m) \oplus t_0 \oplus t_1 \oplus K_0) \oplus K_1) \oplus K_2 \\ &= F(F(F(m) \oplus t_1 \oplus K) \oplus K_1) \oplus K_2 \\ &= F(F(F(m) \oplus K_1) \oplus K_1) \oplus K_2 \ , \end{aligned} \tag{5.35}$$

and

$$\begin{aligned} F(E(m) \oplus t_1 \oplus t_2) &= F(F(F(m \oplus K_0) \oplus K_1) \oplus K_2 \oplus t_1 \oplus t_2) \\ &= F(F(F(m \oplus K_0) \oplus K_1) \oplus K \oplus t_1) \\ &= F(F(F(m \oplus K_0) \oplus K_1) \oplus K_1) \ . \end{aligned} \tag{5.36}$$

Hence

$$E(F(m \oplus K_0) \oplus t_0 \oplus t_1) = F(E(m) \oplus t_1 \oplus t_2) \oplus K_2 \ . \tag{5.37}$$

Now define

$$X(m) = E(F(m) \oplus t_0 \oplus t_1) \oplus F(E(m) \oplus t_1 \oplus t_2) \ . \tag{5.38}$$

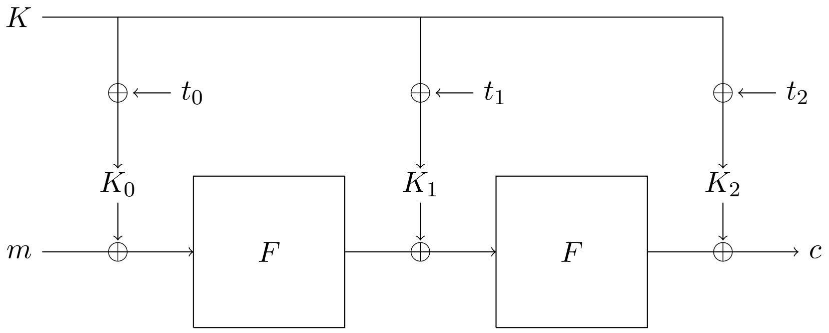

Figure 5.5: The 2-round EM scheme with a single permutation and a simple key-schedule

Given access to the constants t_0, t_1, t_2 as well as both E and F oracles, $X(m)$ can be evaluated by the adversary. Using Equation 5.37, the function can be rewritten as

$$X(m) = E(F(m) \oplus t_0 \oplus t_1) \oplus E(F(m \oplus K_0) \oplus t_0 \oplus t_1) \oplus K_2 \; . \tag{5.39}$$

It follows that $X(m) = X(m \oplus K_0)$. Again, the analysis and the details of the quantum attack are almost identical to the single round case above, and the key $K = K_0 \oplus t_0$ can be recovered with constant probability using $O(n)$ queries.

Our attack depends crucially on Simon's algorithm, and to apply it, a property equivalent to Equation 5.12 needs to hold for a function X that can be evaluated by the adversary. This is no longer the case in more general ciphers such as key-alternating schemes with more than one permutation [43, 28]. Already when the publicly known permutation in the single-round Even-Mansour scheme is replaced by a keyed permutation as in DESX [74], the adversary loses the ability to evaluate $F(m)$ and therefore $X(m)$ in Equation 5.10 and Equations 5.32 or 5.38. This means that additional techniques would be required in order to extend the methods described here to more general encryption schemes.

Quantum superposition attacks are very powerful, but in the next chapter we show a number of notions of confidentiality that are achievable against them.

Chapter 6

Notions of Confidentiality in a Quantum Setting

Contents

In this chapter, we address the security of symmetric encryption schemes against quantum superposition attacks. We will consider both quantum superposition chosen plaintext attack (qsCPA), where a quantum adversary is given superposition access to an encryption oracle, and quantum superposition chosen ciphertext attack (qsCCA), where the adversary in addition has superposition access to a decryption oracle. We will discuss achievability of different confidentiality notions in this setting. We prove that RoP-qsCPA and RoP-qsCCA are achievable by showing two generic symmetric schemes that satisfy these notions. We also discuss semantic security in this setting and prove a reduction from RoP to SEM. Our security analysis is in a concrete security framework. Here we only discuss symmetric cryptosystems, but the discussion for asymmetric schemes is similar and our results apply.

6.1 Introduction

We studied notions of confidentiality in the quantum computation setting in Chapter 4. In this chapter we study notions of confidentiality in the quantum superposition setting where quantum queries are allowed. There has not been a systematic exploration of how existing classical security notions translate into this quantum world. A natural question in this setting is to see whether notions of confidentiality arise from their classical counterparts, or whether they are needed to be rethought from scratch.

We explore two routes to define notions of confidentiality in the quantum superposition setting. One route is to start from a generalisation of semantic security to this setting. To have a meaningful semantic security notion, it is required to properly define the message space in the quantum superposition setting. And also it is essential to take necessary restrictions into account to prevent the quantum adversary from winning trivially. In Section 6.3 we give our definition of semantic security against a quantum adversary.

Another route is to start from a generalisation of indistinguishability notions to the quantum superposition setting.

Boneh and Zhandry [29] discuss a notion of CPA security where quantum encryption queries are allowed. They define a quantum analogue of LoR-CPA arisen from its classical counterpart (Definition 4.1 in [29]) where the adversary is allowed to make chosen message queries on superpositions of message pairs. For a given symmetric encryption scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ and a chosen bit $b \leftarrow\$ \{0,1\}$, the encryption queries are in the form of:

$$\sum_{m_0,m_1,c} \alpha_{m_0,m_1,c} \left|m_0, m_1, c\right\rangle \longrightarrow \sum_{m_0,m_1,c} \alpha_{m_0,m_1,c} \left|m_0, m_1, c \oplus \mathcal{E}_K\left(m_b\right)\right\rangle \ . \tag{6.1}$$

Boneh and Zhandry go on to prove that this notion of CPA security is not achievable.

Result 15 ([29] Theorem 4.2) *No symmetric encryption scheme $\mathcal{SE}$ satisfies the quantum analogue of the LoR-CPA notion defined in [29].*

The nature of their proof for Result 15 is that depending on which message gets encrypted, the register containing that message is entangled with the ciphertext response. Therefore, the quantum adversary can exploit this entanglement to distinguish between encrypted messages. Based on the same intuition, it turns out that quantum analogues of RoR-CPA and FtG-CPA, if arisen from their classical counterparts, are also not achievable in the quantum superposition setting. For instance, in RoR-CPA, for a given symmetric encryption scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ and a chosen bit $b \leftarrow\$ \{0, 1\}$, the encryption queries are either in the form of:

$$\sum_{m,c} \alpha_{m,c} |m, c\rangle \longrightarrow \sum_{m,c} \alpha_{m,c} |m, c \oplus \mathcal{E}_K(m)\rangle \ , \tag{6.2}$$

if $b = 1$; or in the form of:

$$\sum_{m,c} \alpha_{m,c} |m, c\rangle \longrightarrow \sum_{m,c} \alpha_{m,c} |m, c \oplus \mathcal{E}_K(r)\rangle \ , \tag{6.3}$$

if $b = 0$; where $r \leftarrow\$ \{0, 1\}^{|m|}$.

Therefore, in RoR-CPA, the message queried by the adversary is either encrypted or not. In either case, the quantum adversary can distinguish between encrypted messages by just checking whether the ciphertext response is entangled with the register or not. Hence, we can conclude that LoR-CPA, RoR-CPA, and FtG-CPA need to be rethought from scratch in the quantum superposition setting. This is as opposed to the RoP notion of confidentiality in this setting which arises from its classical counterpart as we prove in the next section.

In this chapter, we define indistinguishability notions as well as a semantic security notion where all queries, including challenge queries, allow quantum superpositions. We show that our notions are achievable both under a *quantum superposition chosen* *plaintext attack* (qsCPA) and a *quantum superposition chosen ciphertext attack* (qsCCA), and our semantic security notion implies our indistinguishability notion.

6.2 Real-or-Permutation Indistinguishability in a Quantum Setting

We introduced the classical version of RoP in Subsection 4.2.1.5. We also proved that RoP and RoR are equivalent in a classical setting (see Theorems 9 and 10 in Subsection 4.2.1.6). It follows that RoP also implies semantic security SEM. We now introduce the quantum analogue of RoP, and we prove that it is achievable even against a quantum superposition adversary.

6.2.1 Quantum Superposition Chosen Plaintext Attack

Assume a quantum adversary that plays the experiment RoP-qsCPA shown in Figure 6.1. The experiment begins with choosing a key $K \leftarrow \mathcal{K}$ and a bit $b \in \{0,1\}$. The quantum adversary is given quantum superposition access to an encryption oracle. The quantum adversary adaptively requests encryptions of quantum queries of its choice. The encryption oracle responds to each encryption query by applying a unitary transformation to the first $(2n + n_r)$ qubits of the adversary's quantum register, where n is the length of the encryption query and n_r is the length of the randomness used by the oracle to encrypt this query. The transformation depends on the bit b. If $b = 1$, the encryption oracle applies the unitary $\mathcal{E}_K(\cdot)$:

$$\sum_{m,x} \lambda_{m,x} \left|m, x\right\rangle \xrightarrow{\mathbf{U}_{\mathcal{E}_K(\cdot)}} \sum_{m,x} \lambda_{m,x} \left|m, x \oplus \mathcal{E}_K(m)\right\rangle \ . \tag{6.4}$$

Otherwise, the challenger chooses a permutation Π uniformly at random from the set of all permutations of $\{0,1\}^n$, and then the encryption oracle applies the unitary $\mathcal{E}_K(\Pi(\cdot))$:

$$\sum_{m,x} \lambda_{m,x} \left| m, x \right\rangle \xrightarrow{\mathbf{U}_{\mathcal{E}_K(\cdot)}\,\mathbf{U}_{\Pi(\cdot)}} \sum_{m,x} \lambda_{m,x} \left| m, x \oplus \mathcal{E}_K\left(\Pi\left(m\right)\right) \right\rangle \ . \tag{6.5}$$

We call the ciphertext returned by the encryption oracle the *challenge ciphertext.* At some point, the quantum adversary outputs a bit b'.

The goal of the quantum adversary is to distinguish between the encryption of its query and the encryption of its permuted query. This can also be interpreted as the goal of the quantum adversary being to find out whether the ciphertext is the encryption of its query directly or whether a unitary transformation was applied to its query before encryption.

Definition 10 [**RoP-qsCPA**] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. Define experiment $\mathbf{Exp}_{\mathcal{SE}}^{rop-qscpa-b}(\mathcal{A})$ for a quantum adversary $\mathcal{A}$ and a bit b as depicted in Figure 6.1. In the experiment, the adversary $\mathcal{A}$ is given quantum superposition access to a real-or-permutation encryption oracle $\mathsf{RoP}_{Q_\mathcal{A}}()$. The encryption oracle responds to each query by applying a unitary transformation to the first $(2n + n_r)$ qubits of the adversary's quantum register $Q_\mathcal{A}$.

The adversary's goal is to output a bit b' as its guess of the challenge bit b, and the experiment returns b' as well. The corresponding advantage of a quantum adversary $\mathcal{A}$ is given by:

$$\mathbf{Adv}_{\mathcal{SE}}^{rop-qscpa}(\mathcal{A}) = \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-qscpa-1}(\mathcal{A}) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-qscpa-0}(\mathcal{A}) = 1\right] \ .$$

This advantage refers to a specific quantum adversary using resources as discussed in Section 3.4. ∎

We now give a symmetric encryption construction, and we prove that it can achieve RoP-qsCPA.

```
$\mathbf{Exp}_{\mathcal{SE}}^{rop-qscpa-1}(\mathcal{A})$
    $K \leftarrow \mathcal{K}$
    $b' \leftarrow \mathcal{A}^{\mathsf{RoP}_{Q_{\mathcal{A}}}()}$
    return $b'$
```

```
$\mathbf{Exp}_{\mathcal{SE}}^{rop-qscpa-0}(\mathcal{A})$
    $K \leftarrow \mathcal{K}$
    $b' \leftarrow \mathcal{A}^{\mathsf{RoP}_{Q_{\mathcal{A}}}()}$
    return $b'$
```

```
$\mathsf{RoP}_{Q_{\mathcal{A}}}()$
    if $b = 1$ then
        Apply $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to $Q_{\mathcal{A}}$
    else
        $\Pi \leftarrow\$ \mathsf{Perm}(n)$
        Apply $\mathbf{U}_{\Pi(\cdot)}$ to $Q_{\mathcal{A}}$
        Apply $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to $Q_{\mathcal{A}}$
    end if
    return
```

Figure 6.1: The RoP-qsCPA confidentiality notion

Construction 3 *Let F be a family of pseudorandom functions. We construct the following symmetric encryption scheme $\mathcal{SE} = (\mathcal{E}, \mathcal{D})$ where:*

$$\begin{aligned}
\mathcal{E}(K, m) :\ & r \leftarrow\$ \{0,1\}^n \\
& c \leftarrow F_K(r) \oplus m \\
& \textbf{output } (r, c) \\
\mathcal{D}(K, r, c) :\ & m \leftarrow F_K(r) \oplus c \\
& \textbf{output } (m)
\end{aligned}$$

In Construction 3, the encryption algorithm is randomised. Moreover, if it is implemented on a quantum computer, then the encryption algorithm uses a single fresh randomness for the entire superposition query, see Section 3.4.

The following theorem establishes that RoP-qsCPA security is achievable. In the concrete security framework adopted here this means the following. The theorem provides a straightforward reduction: if our Construction 3 can be broken by a specific quantum adversary, the reduction establishes the existence of a quantum adversary using similar resources that can break the underlying QPRF. But as we discuss in Subsection 4.1.2, a QPRF based on a suitably chosen block cipher is currently thought to be secure against quantum attacks.

Theorem 16 (RoP-qsCPA security is achievable) *Consider the scheme $\mathcal{SE}$ in Construction 3 based on a family of pseudorandom functions F. Assume $\mathcal{A}$ is a quantum adversary attacking $\mathcal{SE}$ in RoP-qsCPA sense, with a running time of at most t, making at most q queries to the encryption oracle, and having advantage*

$$\mathbf{Adv}_{\mathcal{SE}}^{rop-qscpa}(\mathcal{A}) \geq \epsilon\ .$$

Then there exists a quantum adversary $\mathcal{B}$ attacking F with a running time of at most $t' = t + q \cdot T_{\Pi}$, making at most q queries to the encryption oracle, and having advantage

$$\mathbf{Adv}_{F}^{qprf}(\mathcal{B}) \geq \frac{1}{2}\left(\epsilon - \frac{q^2}{2^{n_r}}\right)\ .$$

Here, T_{Π} is the maximum required time to apply a permutation Π.

Proof We first prove the security of the scheme when F is replaced by a truly random function f. Next, we claim that if the scheme is insecure when F was used, then there exists a quantum adversary which can distinguish F from a truly random function f.

We use $\widetilde{\mathcal{SE}} = \left(\widetilde{\mathcal{E}}, \widetilde{\mathcal{D}}\right)$ to denote an encryption scheme that is the same as $\mathcal{SE}$ in Construction 3, except that a truly random function f is used instead of F. Consider the following RoP-qsCPA experiment that the quantum adversary $\mathcal{A}$ plays.

The challenger maintains an encryption oracle to which the quantum adversary is given quantum superposition access.

$\mathcal{A}$ adaptively requests encryption of quantum queries of its choice. The encryption oracle responds to each encryption query by choosing a random $r \leftarrow\!\!\$ \{0,1\}^{n_r}$ and then applying a unitary transformation to the first $(2n + n_r)$ qubits of the adversary's quantum register. If $b = 1$, then the encryption oracle applies

$$\sum_m \alpha_m |m, 0, 0\rangle \xrightarrow{\mathbf{U}_{f(r)}} \sum_m \alpha_m |m, f(r) \oplus m, r\rangle . \tag{6.6}$$

Otherwise the encryption oracle chooses a permutation $\Pi \leftarrow\!\!\$ \,\mathsf{Perm}(n)$, and then applies $\mathbf{U}_{\Pi(m)}$ on the first $2n$ qubits of the adversary's quantum register, followed by applying $\mathbf{U}_{f(r)}$ on the first $(2n + n_r)$ qubits of the quantum register:

$$\sum_m \alpha_m |m, 0, 0\rangle \xrightarrow{\mathbf{U}_{f(r)}\,\mathbf{U}_{\Pi(m)}} \sum_m \alpha_m |m, f(r) \oplus \Pi(m), r\rangle . \tag{6.7}$$

Note that each encryption query receives a single r for the entire query superposition, meaning that the encryption oracle uses the same randomness r for every message in the superposition. Hence the encryption oracle can answer any encryption query by making a single query to f on r. At some point, $\mathcal{A}$ outputs a guess b' for b.

We claim that the advantage of the quantum adversary $\mathcal{A}$ is:

$$\mathbf{Adv}^{rop-qscpa}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) \leq \frac{q^2}{2^{n_r}} . \tag{6.8}$$

To justify Equation 6.8, see that

$$\mathbf{Adv}^{rop-qscpa}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) = \Pr\left[\mathbf{Exp}^{rop-qscpa-1}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) = 1\right] - \Pr\left[\mathbf{Exp}^{rop-qscpa-0}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) = 1\right] . \tag{6.9}$$

A random r^* might be used more than once by the encryption oracle, giving the quantum adversary partial information about the encrypted message. To denote this event, we define Repeat. Therefore we have

$$\Pr\left[\mathbf{Exp}_{\widetilde{\mathcal{SE}}}^{rop-qscpa-1}(\mathcal{A})=1\right]=\Pr\left[\mathbf{Exp}_{\widetilde{\mathcal{SE}}}^{rop-qscpa-1}(\mathcal{A})=1\wedge\mathsf{Repeat}\right]+\Pr\left[\mathbf{Exp}_{\widetilde{\mathcal{SE}}}^{rop-qscpa-1}(\mathcal{A})=1\wedge\neg\mathsf{Repeat}\right]. \tag{6.10}$$

Similarly,

$$\Pr\left[\mathbf{Exp}_{\widetilde{\mathcal{SE}}}^{rop-qscpa-0}(\mathcal{A})=1\right]=\Pr\left[\mathbf{Exp}_{\widetilde{\mathcal{SE}}}^{rop-qscpa-0}(\mathcal{A})=1\wedge\mathsf{Repeat}\right]+\Pr\left[\mathbf{Exp}_{\widetilde{\mathcal{SE}}}^{rop-qscpa-0}(\mathcal{A})=1\wedge\neg\mathsf{Repeat}\right]. \tag{6.11}$$

Therefore,

$$\begin{aligned}\mathbf{Adv}_{\widetilde{\mathcal{SE}}}^{rop-qscpa}(\mathcal{A})=\Big(&\Pr\left[\mathbf{Exp}_{\widetilde{\mathcal{SE}}}^{rop-qscpa-1}(\mathcal{A})=1\wedge\mathsf{Repeat}\right]\\&-\Pr\left[\mathbf{Exp}_{\widetilde{\mathcal{SE}}}^{rop-qscpa-0}(\mathcal{A})=1\wedge\mathsf{Repeat}\right]\Big)\\&+\Big(\Pr\left[\mathbf{Exp}_{\widetilde{\mathcal{SE}}}^{rop-qscpa-1}(\mathcal{A})=1\wedge\neg\mathsf{Repeat}\right]\\&-\Pr\left[\mathbf{Exp}_{\widetilde{\mathcal{SE}}}^{rop-qscpa-0}(\mathcal{A})=1\wedge\neg\mathsf{Repeat}\right]\Big).\end{aligned} \tag{6.12}$$

The first difference is at most the probability of the event Repeat happening. Since r^* is chosen uniformly at random from $\{0,1\}^{n_r}$, it follows by the birthday bound that the probability of the event Repeat is bounded by $q^2/2^{n_r}$ where q is the number of encryption queries made by the quantum adversary. The second difference, on the other hand, is zero, because with a true random function there is a one-to-one mapping between every random choice, which makes the value of $f(r)\oplus m$ or $f(r)\oplus\Pi(m)$ completely random, and hence indistinguishable for the quantum adversary. Note that the encryption acts the same as a one-time pad, and is thus information theoretically secure (see Section 2.2), even against a quantum adversary, as long as the value of $f(r)$ is not repeated for the other superposition queries during the encryption.

Now, assume $\mathcal{A}$ is attacking $\mathcal{SE}$ of Construction 3 in the RoP-qsCPA sense. We construct a quantum adversary $\mathcal{B}$, using $\mathcal{A}$, to attack the QPRF security of F. The quantum adversary $\mathcal{B}$ uses its oracles to provide a simulation of $\mathcal{A}$'s oracles. $\mathcal{B}$ runs $\mathcal{A}$.

The challenger maintains QPRF experiment. The quantum adversary $\mathcal{B}$ chooses a bit $d \leftarrow\$ \{0,1\}$. When $\mathcal{A}$ adaptively makes quantum encryption queries, $\mathcal{B}$ will respond with the output from its oracle. $\mathcal{B}$ does this by choosing an $r \leftarrow\$ \{0,1\}^{n_r}$ and then queries its oracle on r. The oracle responds by either $s = F_K(r)$ if $b = 1$, or, $s = f(r)$ otherwise, where $f \leftarrow\$ \mathsf{Func}(n,n)$. If $d = 1$, then $\mathcal{B}$ applies a unitary transformation to the first $(2n + n_r)$ qubits of $\mathcal{A}$'s quantum register to prepare $\sum_m \alpha_m |m, s \oplus m, r\rangle$. Otherwise $\mathcal{B}$ chooses a permutation $\Pi \leftarrow\$ \mathsf{Perm}(n)$, and applies the unitary $\mathbf{U}_{\Pi(\cdot)}$ to the first $2n$ qubits of $\mathcal{A}$'s quantum register. This is followed by applying a unitary transformation to the first $(2n + n_r)$ qubits of $\mathcal{A}$'s quantum register to prepare $\sum_m \alpha_m |m, s \oplus \Pi(m), r\rangle$.

Eventually $\mathcal{A}$ outputs a bit d' for d. If $d = d'$, then $\mathcal{B}$ outputs 1. Otherwise it outputs 0.

For the advantage of $\mathcal{B}$, we have:

$$\mathbf{Adv}_F^{qprf}(\mathcal{B}) = \Pr\left[\mathbf{Exp}_F^{qprf-1}(\mathcal{B}) = 1\right] - \Pr\left[\mathbf{Exp}_F^{qprf-0}(\mathcal{B}) = 1\right] . \tag{6.13}$$

When $b = 0$ it is easy to see that $\mathcal{B}$ simulates the RoP-qsCPA experiment for $\mathcal{A}$ when it is attacking $\widetilde{\mathcal{SE}}$. Therefore:

$$\Pr\left[\mathbf{Exp}_F^{qprf-0}(\mathcal{B}) = 1\right] = \frac{1}{2} + \frac{1}{2} \cdot \mathbf{Adv}_{\widetilde{\mathcal{SE}}}^{rop-qscpa}(\mathcal{A}) . \tag{6.14}$$

Moreover, when $b = 1$ we can see that $\mathcal{B}$ simulates the RoP-qsCPA experiment for $\mathcal{A}$ when it is attacking $\mathcal{SE}$. Therefore:

$$\Pr\left[\mathbf{Exp}_F^{qprf-1}(\mathcal{B}) = 1\right] = \frac{1}{2} + \frac{1}{2} \cdot \mathbf{Adv}_{\mathcal{SE}}^{rop-qscpa}(\mathcal{A}) . \tag{6.15}$$

Hence,

$$\begin{aligned}
\mathbf{Adv}_{F}^{qprf}(\mathcal{B}) &= \frac{1}{2}\cdot \mathbf{Adv}_{\mathcal{SE}}^{rop-qscpa}(\mathcal{A}) - \frac{1}{2}\cdot \mathbf{Adv}_{\widetilde{\mathcal{SE}}}^{rop-qscpa}(\mathcal{A}) \\
2\cdot \mathbf{Adv}_{F}^{qprf}(\mathcal{B}) &\geq \mathbf{Adv}_{\mathcal{SE}}^{rop-qscpa}(\mathcal{A}) - \frac{q^2}{2^{n_r}} \\
2\cdot \mathbf{Adv}_{F}^{qprf}(\mathcal{B}) + \frac{q^2}{2^{n_r}} &\geq \mathbf{Adv}_{\mathcal{SE}}^{rop-qscpa}(\mathcal{A}) \ . \qquad (6.16)
\end{aligned}$$

Since it is assumed that

$$\mathbf{Adv}_{\mathcal{SE}}^{rop-qscpa}(\mathcal{A}) \geq \epsilon \ , \qquad (6.17)$$

then

$$\mathbf{Adv}_{F}^{qprf}(\mathcal{B}) \geq \frac{1}{2}\left(\epsilon - \frac{q^2}{2^{n_r}}\right) \ . \qquad (6.18)$$

$\mathcal{B}$ runs in time at most $t' = t + q \cdot T_{\Pi}$ where t is the upper bound for the running time of $\mathcal{A}$ and T_{Π} is the maximum required time to apply a permutation Π by $\mathcal{B}$. $\mathcal{A}$ and $\mathcal{B}$ make at most q oracle queries. This concludes the proof. ▮

6.2.2 Quantum Superposition Chosen Ciphertext Attack

For indistinguishability under chosen ciphertext attack definitions in the classical setting, one could assume that the adversary does not query the decryption oracle on ciphertexts that it receives from the encryption oracle. In the other words, the probability that the adversary makes the challenge decryption query is 0. However, this assumption is not enough to prevent an adversary from trivially winning in the quantum setting. This is because a quantum adversary can make a decryption query (a quantum superposition query) that is different from the challenge ciphertext (that received from the encryption oracle) but still very close to it, helping the quantum adversary to win the game with high probability.

In order to define indistinguishability under a quantum superposition chosen ciphertext attack (IND-qsCCA), we restrict the quantum adversary to make sure that it cannot query the decryption oracle with any of the challenge ciphertexts. Given that the quantum challenge ciphertexts are of the form

$$\sum_c \lambda_c \left|m_c, c\right\rangle \ , \tag{6.19}$$

then for each challenge ciphertext we define the projector as:

$$\mathsf{Proj}_c = \sum_x \left|x, c\right\rangle\!\left\langle x, c\right| \ . \tag{6.20}$$

We also use ρ to denote the state of the adversary's quantum register, Q, before any decryption query. We assume the following condition holds for any quantum adversary:

$$\mathrm{Tr}\left(\mathsf{Proj}_c\, \rho\right) = 0 \ \ \forall c \text{ such that } \lambda_c \neq 0 \ . \tag{6.21}$$

We first present the notion of RoP-qsCCA security. Then we prove that it is achievable.

Assume a quantum adversary that plays the experiment RoP-qsCCA shown in Figure 6.2. The experiment begins with choosing a key $K \leftarrow \mathcal{K}$ and a bit $b \in \{0, 1\}$. The quantum adversary is given quantum superposition access to an encryption oracle. The quantum adversary adaptively requests encryptions of quantum queries of its choice. The encryption oracle responds to each encryption query by applying a unitary transformation to the first $(2n + n_r)$ qubits of the quantum adversary's quantum register, where n is the length of the encryption query and n_r is the length of the randomness used by the encryption oracle to encrypt the query. The transformation depends on the bit b. If $b = 1$, then the encryption oracle applies the unitary $\mathcal{E}_K\left(\cdot\right)$:

$$\sum_{m,x} \lambda_{m,x} \left|m, x\right\rangle \xrightarrow{\mathbf{U}_{\mathcal{E}_K(\cdot)}} \sum_{m,x} \lambda_{m,x} \left|m, x \oplus \mathcal{E}_K\left(m\right)\right\rangle \ . \tag{6.22}$$

Otherwise, the challenger chooses a permutation Π uniformly at random from the set of all permutations of $\{0,1\}^n$, and then the encryption oracle applies the unitary $\mathcal{E}_K\left(\Pi\left(\cdot\right)\right)$:

$$\sum_{m,x} \lambda_{m,x} \left|m, x\right\rangle \xrightarrow{\mathbf{U}_{\mathcal{E}_K(\cdot)}\,\mathbf{U}_{\Pi(\cdot)}} \sum_{m,x} \lambda_{m,x} \left|m, x \oplus \mathcal{E}_K\left(\Pi\left(m\right)\right)\right\rangle \ . \tag{6.23}$$

We call the ciphertext returned by the encryption oracle the *challenge ciphertext*. Additionally, the quantum adversary is given quantum superposition access to a decryption oracle. The quantum adversary can query the decryption oracle on any ciphertext as long as the condition given in Equation 6.21 is satisfied. At some point the quantum adversary outputs a bit b'.

Definition 11 [**RoP-qsCCA**] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. Define experiment $\mathbf{Exp}_{\mathcal{SE}}^{rop-qscca-b}\left(\mathcal{A}\right)$ for a quantum adversary $\mathcal{A}$ and a bit b as depicted in Figure 6.2. In the experiment, the adversary $\mathcal{A}$ is given quantum superposition access to a real-or-permutation encryption oracle $\mathsf{RoP}_{Q_\mathcal{A}}\left(\right)$. The encryption oracle responds to each query by applying a unitary transformation to the first $(2n + n_r)$ qubits of the adversary's quantum register $Q_\mathcal{A}$. The adversary $\mathcal{A}$ is additionally given quantum superposition access to a decryption oracle, $\mathsf{Dec}_{Q_\mathcal{A}}\left(\right)$. For any challenge ciphertext $\sum_c \lambda_c \left|m_c, c\right\rangle$ we define the projector $\mathsf{Proj}_c = \sum_x |x, c\rangle\langle x, c|$. No restriction is imposed on the quantum adversary's queries, rather than it is assumed that

$$\Pr\left[\exists c : \lambda_c \neq 0 \text{ and } \mathrm{Tr}\left(\mathsf{Proj}_c\, \rho\right) \neq 0\right] = 0 \ ,$$

where ρ is the state of $Q_\mathcal{A}$ before making any decryption query.

The adversary's goal is to output a bit b' as its guess of the challenge bit b, and the experiment returns b' as well. The advantage of a quantum adversary $\mathcal{A}$ is given by:

```
$\mathbf{Exp}_{\mathcal{SE}}^{rop-qscca-1}(\mathcal{A})$
    $K \leftarrow \mathcal{K}$
    $b' \leftarrow \mathcal{A}^{\mathsf{RoP}_{Q_\mathcal{A}}(),\mathsf{Dec}_{Q_\mathcal{A}}()}$
    return $b'$
```

```
$\mathbf{Exp}_{\mathcal{SE}}^{rop-qscca-0}(\mathcal{A})$
    $K \leftarrow \mathcal{K}$
    $b' \leftarrow \mathcal{A}^{\mathsf{RoP}_{Q_\mathcal{A}}(),\mathsf{Dec}_{Q_\mathcal{A}}()}$
    return $b'$
```

```
$\mathsf{RoP}_{Q_\mathcal{A}}()$
    if $b = 1$ then
        Apply $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to $Q_\mathcal{A}$
    else
        $\Pi \leftarrow\$ \mathsf{Perm}(n)$
        Apply $\mathbf{U}_{\Pi(\cdot)}$ to $Q_\mathcal{A}$
        Apply $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to $Q_\mathcal{A}$
    end if
    return
```

```
$\mathsf{Dec}_{Q_\mathcal{A}}()$
    Apply $\mathbf{U}_{\mathcal{D}_K(\cdot)}$ to $Q_\mathcal{A}$
    return
```

Figure 6.2: The RoP-qsCCA confidentiality notion

$$\mathbf{Adv}_{\mathcal{SE}}^{rop-qscca}(\mathcal{A}) = \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-qscca-1}(\mathcal{A}) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-qscca-0}(\mathcal{A}) = 1\right] .$$

This advantage refers to a specific quantum adversary using resources as discussed in Section 3.4. ∎

We now give a symmetric encryption construction, and prove that it can achieve RoP-qsCCA security. Here we construct an RoP-qsCCA secure symmetric encryption scheme using the Encrypt-then-MAC (EtM) paradigm.

Construction 4 *Let $\mathcal{SE} = (\mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme and let F be a family of pseudorandom functions. We construct the following encryption scheme $\mathcal{SE}' = (\mathcal{E}', \mathcal{D}')$ where:*

$$\begin{aligned}
\mathcal{E}'\left(\left(K_1, K_2\right), m\right) : & \, c \leftarrow \mathcal{E}_{K_1}\left(m\right), \, \tau \leftarrow F_{K_2}\left(c\right) \\
& \textbf{output } \left(c, \tau\right) \\
\mathcal{D}'\left(\left(K_1, K_2\right), c, \tau\right) : & \, \tau' \leftarrow F_{K_2}\left(c\right), \, m \leftarrow \mathcal{D}_{K_1}\left(c\right) \\
& \textbf{if } \tau = \tau', \textbf{ output } \left(m\right) \\
& \textbf{otherwise, output } \perp
\end{aligned}$$

In Construction 4, the encryption algorithm is randomised. Moreover, if it is implemented on a quantum computer, then the encryption algorithm uses a single fresh randomness for the entire superposition query.

Theorem 17 (RoP-qsCCA security is achievable) *Consider the scheme $\mathcal{SE}'$ in Construction 4 based on a family of pseudorandom functions F and an encryption scheme $\mathcal{SE}$. Assume $\mathcal{A}$ is a quantum adversary attacking $\mathcal{SE}'$ in the RoP-qsCCA sense with a running time of at most t, making at most q_e encryption and q_d decryption queries to the oracle, and having advantage*

$$\mathbf{Adv}_{\mathcal{SE}'}^{rop-qscca}\left(\mathcal{A}\right) \geq \epsilon \, .$$

Then there exist quantum adversaries $\mathcal{B}$ and $\mathcal{J}$ attacking $\mathcal{SE}$ and F respectively, as follows. $\mathcal{B}$ has running time of at most t and makes at most q_e encryption oracle queries. $\mathcal{J}$ has running time of at most t and makes at most q_d oracle queries. The advantages satisfy:

$$\mathbf{Adv}_{\mathcal{SE}}^{rop-qscpa}\left(\mathcal{B}\right) + 2 \cdot \mathbf{Adv}_{F}^{qprf}\left(\mathcal{J}\right) \geq \epsilon - 2\left(1 + 2q_d^2\right) 2^{-n_\tau/4} \, ,$$

where n_τ is the length of tag τ as defined in Construction 4.

Proof To prove Theorem 17, we first modify Construction 4 by replacing F with f which is a true random function. Next we show that the modified construction is indistinguishable from the original construction. If the quantum adversary can distinguish between these two constructions, then it can break QPRF security of F.

We first construct a new scheme $\widetilde{\mathcal{SE}}$ which is similar to $\mathcal{SE}'$ in Construction 4 except that F_{K_2} is replaced by f, where $f \leftarrow\$ \, \mathsf{Func}(n,n)$ is a true random function. Hence the $\widetilde{\mathcal{SE}} = \left(\widetilde{\mathcal{E}}, \widetilde{\mathcal{D}}\right)$ where:

$$\begin{aligned}
\widetilde{\mathcal{E}}\left(\left(K_1\right), m\right) &: c \leftarrow \mathcal{E}_{K_1}\left(m\right), \, \tau \leftarrow f\left(c\right) \\
&\quad \textbf{output } \left(c, \tau\right) \\
\widetilde{\mathcal{D}}\left(\left(K_1\right), c, \tau\right) &: \tau' \leftarrow f\left(c\right), \, m \leftarrow \mathcal{D}_{K_1}\left(c\right) \\
&\quad \textbf{if } \tau = \tau', \textbf{ output } \left(m\right) \\
&\quad \textbf{otherwise, output } \perp
\end{aligned}$$

Consider the following RoP-qsCCA experiment that the quantum adversary $\mathcal{A}$ plays with regards to $\widetilde{\mathcal{SE}}$.

The challenger maintains the experiment. The quantum adversary, $\mathcal{A}$, makes adaptive quantum queries of its choice. The queries can be either encryption or decryption queries. The oracles respond to each query by applying a unitary transformation to the adversary's quantum register.

In the case of encryption queries, the unitary transformation depends on the bit b. If $b = 1$, then the encryption oracle applies the unitary $\mathbf{U}_{\mathcal{E}_{K_1}}$, followed by $\mathbf{U}_f$.

$$\sum_m \alpha_m \left|m, 0, 0\right\rangle \xrightarrow{\mathbf{U}_{f(\cdot)}\,\mathbf{U}_{\mathcal{E}_{K_1}(\cdot)}} \sum_m \alpha_m \left|m, \mathcal{E}_{K_1}\left(m\right), f\left(\mathcal{E}_{K_1}\left(m\right)\right)\right\rangle \,, \tag{6.24}$$

where $c = \mathcal{E}_{K_1}\left(m\right)$ and $\tau = f\left(\mathcal{E}_{K_1}\left(m\right)\right)$.

If $b = 0$, the encryption oracle chooses a permutation $\Pi \leftarrow\$ \, \mathsf{Perm}\left(n\right)$, and applies the unitary $\mathbf{U}_{\mathcal{E}_{K_1}(\Pi(\cdot))}$, followed by $\mathbf{U}_f$.

$$\sum_m \alpha_m \left|m,0,0\right\rangle \xrightarrow{\mathbf{U}_{f(\cdot)}\,\mathbf{U}_{\mathcal{E}_{K_1}(\Pi(\cdot))}} \sum_m \alpha_m \left|m,\mathcal{E}_{K_1}\left(\Pi\left(m\right)\right),f\left(\mathcal{E}_{K_1}\left(\Pi\left(m\right)\right)\right)\right\rangle \ , \qquad (6.25)$$

where $c = \mathcal{E}_{K_1}\left(\Pi\left(m\right)\right)$ and $\tau = f\left(\mathcal{E}_{K_1}\left(\Pi\left(m\right)\right)\right)$.

In the case of decryption queries, the decryption oracle applies the unitary $\mathbf{U}_{\widetilde{\mathcal{D}}_{K_1}(\cdot,\cdot)}$ to the adversary's quantum register

$$\sum_{c,\tau} \alpha_{c,\tau} \left|c,\tau,0\right\rangle \xrightarrow{\mathbf{U}_{\widetilde{\mathcal{D}}_{K_1}(\cdot,\cdot)}} \sum_{c,\tau} \alpha_{c,\tau} \left|c,\tau,\widetilde{\mathcal{D}}_{K_1}\left(c,\tau\right)\right\rangle \ , \qquad (6.26)$$

where

$$\widetilde{\mathcal{D}}_{K_1}\left(c,\tau\right) = \begin{cases} m \leftarrow \mathcal{D}_{K_1}\left(c\right) & \textbf{if } f\left(c\right) = \tau \\ \perp & \textbf{otherwise} \end{cases} .$$

Eventually the quantum adversary $\mathcal{A}$ outputs a guess b' for b.

Claim 1 *The advantage of $\mathcal{A}$ in the RoP-qsCCA experiment with regards to $\widetilde{\mathcal{SE}}$ is:*

$$\mathbf{Adv}^{rop-qscca}_{\widetilde{\mathcal{SE}}}\left(\mathcal{A}\right) \leq \mathbf{Adv}^{rop-qscpa}_{\mathcal{SE}}\left(\mathcal{B}\right) + 2\left(1 + 2\,q_d^2\right)\,2^{-n_\tau/4} \ . \qquad (6.27)$$

Proof of Claim 1. In the classical setting, since f is a true random function, the probability that an adversary forges a valid tag for a ciphertext, which it has not been given by the encryption oracle before, is $q_d/2^{n_\tau}$. Hence, the classical adversary, with high probability, gets $\perp$ most of the time in response to its decryption queries. In this case, the decryption oracle is not useful for the classical adversary and the security of the construction reduces to RoP-CPA security of the $\mathcal{SE} = (\mathcal{E},\mathcal{D})$. However this is not the case in the quantum setting, where quantum superposition queries are allowed. For instance, the quantum adversary could query all the possible tags for a ciphertext, by just one superposition query. Then it might be able to somehow extract the valid tag from the encryption oracle's response. This leads the quantum adversary to be able to decrypt a ciphertext that helps it to win the game.

We use two scenarios, denoted by Q0 and Q1, to see the advantage of $\mathcal{A}$ in the RoP-qsCCA experiment with regards to $\widetilde{\mathcal{SE}}$. For every ciphertext c, the tag $\tau = f\left(c\right)$ is a string randomly chosen from the set $\{0,1\}^{n_\tau}$, i.e., the tag is a random n_τ-bit string. Assume the quantum adversary makes q_e encryption queries and q_d decryption queries.

Q0: In this scenario, we first assume the decryption oracle of the construction $\widetilde{\mathcal{SE}}$ always returns $\perp$ in response to the quantum adversary's decryption queries. Let the unitary $\tilde{\mathbf{V}}_i$ denote the decryption oracle's action on the adversary's quantum register in the i-th decryption query, where $i = 1, \ldots, q_d$. The action of the quantum adversary can be written as:

$$\mathbf{U}_{q_d}\tilde{\mathbf{V}}_{q_d}\ldots\mathbf{U}_2\tilde{\mathbf{V}}_2\mathbf{U}_1\tilde{\mathbf{V}}_1\mathbf{U}_0\left|s\right\rangle \ . \tag{6.28}$$

This is followed by a binary measurement whose outcome is the guess b'. The input state $|s\rangle$ is the result of some initialisation. The unitaries $\mathbf{U}_i$ describe the evolution of the quantum adversary between decryption queries and include the actions of the encryption oracle.

The quantum register state consists of three sections, the first one for the message, the second one for the ciphertext, and the third one for the tag. The action of the decryption oracle $\tilde{\mathbf{V}}_i$ on a register state $|m, c, \tau\rangle$ is:

$$\tilde{\mathbf{V}}_i\left|m, c, \tau\right\rangle = \left|m \oplus \perp, c, \tau\right\rangle \ . \tag{6.29}$$

Note that $\perp$ is some fixed string that is outside the message space.

Since the decryption oracle always return $\perp$, it is not useful for $\mathcal{A}$. Therefore, the security of the construction in scenario Q0 is reduced to RoP-qsCPA security of $\mathcal{SE}$.

Assume $\mathcal{A}$ attacks RoP-qsCCA security of $\widetilde{\mathcal{SE}}$. We construct another adversary $\mathcal{B}$, using $\mathcal{A}$, to attack RoP-qsCPA security of $\mathcal{SE}$.

$\mathcal{B}$ runs $\mathcal{A}$, and uses its oracles to provide a simulation of $\mathcal{A}$'s oracles in the RoP-qsCCA experiment. The game is straight forward and it is easy to see the advantage of the quantum adversary in the scenario Q0:

$$\begin{aligned}\mathbf{Adv}_{\widetilde{\mathcal{SE}}}^{Q0}(\mathcal{A}) &= \Pr\left[\mathbf{Exp}_{\widetilde{\mathcal{SE}}}^{rop-qscca-1}(\mathcal{A})=1\right]-\Pr\left[\mathbf{Exp}_{\widetilde{\mathcal{SE}}}^{rop-qscca-0}(\mathcal{A})=1\right]\\ &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-qscpa-1}(\mathcal{B})=1\right]-\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-qscpa-0}(\mathcal{B})=1\right]\\ &\le \mathbf{Adv}_{\mathcal{SE}}^{rop-qscpa}(\mathcal{B}) \ . \end{aligned} \tag{6.30}$$

Q1: Now assume the decryption oracle of the construction $\widetilde{\mathcal{SE}}$ works as it is supposed to, meaning, it checks the tag for each ciphertext and decrypts if the tag was valid. Let the unitary $\mathbf{V}_i$ denote the decryption oracle's action on the quantum adversary's quantum register in the i-th decryption query, where $i = 1, \ldots, q_d$. The action of the quantum adversary can be written as:

$$\mathbf{U}_{q_d}\mathbf{V}_{q_d}\ldots\mathbf{U}_2\mathbf{V}_2\mathbf{U}_1\mathbf{V}_1\mathbf{U}_0\left|s\right\rangle \ . \tag{6.31}$$

This is followed by a binary measurement whose outcome is the guess b'. The input state $|s\rangle$ is the result of some initialisation. The unitaries $\mathbf{U}_i$ describe the evolution of the adversary between decryption queries and include the actions of the encryption oracle.

The quantum register state consists of three sections, the first one for the message, the second one for the ciphertext, and the third one for the tag. The action of the decryption oracle $\mathbf{V}_i$ on a register state $|m, c, \tau\rangle$ is:

$$\mathbf{V}_i\left|m,c,\tau\right\rangle = \left|m\oplus\mathcal{D}_{K_1}(c),c,\tau\right\rangle \quad \textbf{if } f(c)=\tau \ , \tag{6.32}$$

$$\mathbf{V}_i\left|m,c,\tau\right\rangle = \left|m\oplus\bot,c,\tau\right\rangle \ . \tag{6.33}$$

Here $\mathcal{D}_{K_1}(c)$ is the decryption of the ciphertext c, and $\perp$ is some fixed string that is outside the message space.

Since, in general, the unitaries $\mathbf{U}_i$ entangle the adversary's quantum register with its internal registers, one cannot assume that the quantum register is in a pure state during decryption queries. Denote by $\mathcal{C}$ the set of all ciphertexts c. For any ciphertext $c \in \mathcal{C}$, define the projector

$$\mathsf{Proj}_c = \sum_{m,\tau} |m, c, \tau\rangle\langle m, c, \tau| = I \otimes |c\rangle\langle c| \otimes I \ . \tag{6.34}$$

Denote by ρ_i^e the state of the quantum register after the i-th encryption query in scenario Q1. Let $\mathcal{C}'$ be the set of all ciphertexts that do not result from any encryption query. That is, ciphertexts that have zero weight in all encryption queries. Formally,

$$\mathcal{C}' = \{c \in \mathcal{C} \ : \ \mathrm{Tr}\left(\mathsf{Proj}_c \rho_i^e\right) = 0 \ , i = 1, \ldots, q_e\} \ . \tag{6.35}$$

We can now define the set $\mathcal{C}_{\text{valid}}$ as the set of pairs (c, τ) that do not result from any encryption query,

$$\mathcal{C}_{\text{valid}} = \left\{(c, \tau) \ : \ c \in \mathcal{C}'\right\} \ . \tag{6.36}$$

Given a ciphertext $c \in \mathcal{C}'$, trying to guess $\tau = f(c)$ leads to a valid pair with very small probability:

$$|\mathcal{C}_{\text{valid}}| = 2^{-n_\tau} \left|\mathcal{C}' \times \{0,1\}^{n_\tau}\right| \ . \tag{6.37}$$

The results of the q_e encryption queries contain no information about the set $\mathcal{C}_{\text{valid}}$.

Now let ρ_i^d be the state of the quantum register before the i-th decryption query in scenario Q1, and define

$$\mathsf{Proj}_{\text{valid}} = \sum_{(c,\tau)\in\mathcal{C}_{\text{valid}}} |c,\tau\rangle\langle c,\tau| \ . \tag{6.38}$$

We can now define $\mathsf{W}_{\text{val},i}$ as the total weight of terms belonging to $\mathcal{C}_{\text{valid}}$ in the i-th decryption query ($i = 1, \ldots, q_d$),

$$\mathsf{W}_{\text{val},i} = \mathrm{Tr}\left(\rho_i^d\,\mathsf{Proj}_{\text{valid}}\right) \ . \tag{6.39}$$

Equivalently, let $\left|\psi_i^d\right\rangle$ be the state of the totality of the adversary's quantum registers immediately before the i-th decryption query in scenario Q1,

$$\begin{aligned} \left|\psi_i^d\right\rangle &= \mathbf{U}_{i-1}\mathbf{V}_{i-1}\ldots\mathbf{U}_1\mathbf{V}_1\mathbf{U}_0\,|s\rangle & (6.40)\\ &= \sum_{j,m,c,\tau} \lambda_{j,m,c,\tau}\,|j,m,c,\tau\rangle \ , & (6.41)\end{aligned}$$

where j labels the computational basis states of all internal registers (i.e., all registers in addition to the quantum register). We then have

$$\mathsf{W}_{\text{val},i} = \left\langle\psi_i^d\right|\mathsf{Proj}_{\text{valid}}\left|\psi_i^d\right\rangle = \sum_{j,m,c}\left|\lambda_{j,m,c,f(c)}\right|^2 = 1 - \sum_{j,m,c,\tau\neq f(c)}\left|\lambda_{j,m,c,\tau}\right|^2 \ . \tag{6.42}$$

The probability that a direct measurement after the i-th decryption query gives a string $(c,\tau)\in\mathcal{C}_{\text{valid}}$ is then given by the expectation value $\mathsf{E}\left(\mathsf{W}_{\text{val},i}\right)$.

Now the optimal way of searching for a string $(c,\tau)\in\mathcal{C}_{\text{valid}}$ is Grover's algorithm. As long as i is less than the minimum number of queries required for Grover's algorithm to succeed with certainty (which is approximately $\frac{\pi}{4}\sqrt{2^{n_\tau}}$), the best probability with which any quantum algorithm can find a string $(c,\tau)\in\mathcal{C}_{\text{valid}}$ using i queries is exactly the probability $\Pr_{\text{Grover}}$ achieved by running Grover's algorithm with i queries [113, 6] (also see Subsection 3.2.2). That probability is equal to $\Pr_{\text{Grover}} = \sin^2\left(\left(i+\frac{1}{2}\right)\theta\right)$, where $\sin\frac{\theta}{2} = \sqrt{2^{-n_\tau}}$ [89]. To a very good approximation,

$$\Pr_{\text{Grover}} = 4i^2\, 2^{-n_\tau} \ . \tag{6.43}$$

By measuring the quantum register after the i-th query and then stopping, the quantum adversary can find a string $(c, \tau) \in \mathcal{C}_{\text{valid}}$ with probability $\mathsf{E}\left(\mathsf{W}_{\text{val},i}\right)$. Therefore we must have

$$\mathsf{E}\left(\mathsf{W}_{\text{val},i}\right) \leq 4i^2\, 2^{-n_\tau} \tag{6.44}$$

for $i = 1, \ldots, q_d$. What we actually need is a bound on the probabilities for $\sqrt{\mathsf{W}_{\text{val},i}}$. For any random variable $X \geq 0$, it holds that $\mathsf{E}\left(\sqrt{X}\right) \leq \sqrt{\mathsf{E}\left(X\right)}$. This follows from $\mathsf{E}\left(X\right) - \left(\mathsf{E}\left(\sqrt{X}\right)\right)^2 = \mathsf{Var}\left(\sqrt{X}\right) \geq 0$. Hence,

$$\mathsf{E}\left(\sqrt{\mathsf{W}_{\text{val},i}}\right) \leq 2i\, 2^{-n_\tau/2} \ . \tag{6.45}$$

Now we want to compare the probability of outputting the guess $b' = 1$ in scenario Q0 and the probability of outputting the guess $b' = 1$ in scenario Q1. Let $\left|\psi_i^d\right\rangle$ denote the state immediately before the i-th decryption query in scenario Q1 as before and, similarly, let

$$\left|\widetilde{\psi_i^d}\right\rangle = \mathbf{U}_{i-1}\tilde{\mathbf{V}}_{i-1}\ldots\mathbf{U}_1\tilde{\mathbf{V}}_1\mathbf{U}_0\left|s\right\rangle \tag{6.46}$$

denote the state immediately before the i-th decryption query in scenario Q0. We have

$$\begin{aligned}
\mathbf{V}_i\left|\psi_i^d\right\rangle &= \sum_{j,m,c,\tau\neq f(c)} \lambda_{j,m,c,\tau}\mathbf{V}_i\left|j,m,c,\tau\right\rangle + \sum_{j,m,c,\tau=f(c)} \lambda_{j,m,c,\tau}\mathbf{V}_i\left|j,m,c,\tau\right\rangle \\
&= \sum_{j,m,c,\tau\neq f(c)} \lambda_{j,m,c,\tau}\left|j,m\oplus\perp,c,\tau\right\rangle \\
&\qquad + \sum_{j,m,c,\tau=f(c)} \lambda_{j,m,c,\tau}\left|j,m\oplus\mathcal{D}_{K_1}\left(c\right),c,\tau\right\rangle \ ,
\end{aligned} \tag{6.47}$$

and

$$\begin{aligned}
\tilde{\mathbf{V}}_i \left|\psi_i^d\right\rangle &= \sum_{j,m,c,\tau\neq f(c)} \lambda_{j,m,c,\tau} \tilde{\mathbf{V}}_i \left|j,m,c,\tau\right\rangle + \sum_{j,m,c,\tau=f(c)} \lambda_{j,m,c,\tau} \tilde{\mathbf{V}}_i \left|j,m,c,\tau\right\rangle \\
&= \sum_{j,m,c,\tau\neq f(c)} \lambda_{j,m,c,\tau} \left|j,m\oplus\perp,c,\tau\right\rangle \\
&\qquad + \sum_{j,m,c,\tau=f(c)} \lambda_{j,m,c,\tau} \left|j,m\oplus\perp,c,\tau\right\rangle \ . \qquad (6.48)
\end{aligned}$$

Putting these together and using Equation 6.42 twice, we get the following for the fidelity of these two states:

$$\begin{aligned}
\left|\left\langle\psi_i^d\right| \tilde{\mathbf{V}}_i^{\dagger}\mathbf{V}_i \left|\psi_i^d\right\rangle\right| &= \Bigg| \sum_{j,m,c,\tau\neq f(c)} \left|\lambda_{j,m,c,\tau}\right|^2 \\
&\qquad + \sum_{j,m,m',c} \lambda^*_{j,m',c,f(c)}\lambda_{j,m,c,f(c)} \left\langle m'\oplus\perp \middle| m\oplus\mathcal{D}_{K_1}(c)\right\rangle \Bigg| \\
&= \left| \sum_{j,m,c,\tau\neq f(c)} \left|\lambda_{j,m,c,\tau}\right|^2 + \sum_{j,m,c} \lambda^*_{j,m\oplus\mathcal{D}_{K_1}(c)\oplus\perp,c,f(c)}\lambda_{j,m,c,f(c)} \right| \\
&\geq \left| \sum_{j,m,c,\tau\neq f(c)} \left|\lambda_{j,m,c,\tau}\right|^2 \right| - \left| \sum_{j,m,c} \lambda^*_{j,m\oplus\mathcal{D}_{K_1}(c)\oplus\perp,c,f(c)}\lambda_{j,m,c,f(c)} \right| \\
&= 1 - \mathsf{W}_{\mathrm{val},i} - \left| \sum_{j,m,c} \lambda^*_{j,m\oplus\mathcal{D}_{K_1}(c)\oplus\perp,c,f(c)}\lambda_{j,m,c,f(c)} \right| \\
&\geq 1 - \mathsf{W}_{\mathrm{val},i} - \sqrt{\sum_{j,m,c} \left|\lambda_{j,m\oplus\mathcal{D}_{K_1}(c)\oplus\perp,c,f(c)}\right|^2} \sqrt{\sum_{j,m,c} \left|\lambda_{j,m,c,f(c)}\right|^2} \\
&= 1 - \mathsf{W}_{\mathrm{val},i} - \sqrt{\mathsf{W}_{\mathrm{val},i}}\sqrt{\mathsf{W}_{\mathrm{val},i}} \\
&= 1 - 2\,\mathsf{W}_{\mathrm{val},i} \ . \qquad (6.49)
\end{aligned}$$

This implies that the trace distance (see Subsection 2.3.6) of these two states is bounded as

$$\mathsf{D}\left(\mathbf{V}_i \left|\psi_i^d\right\rangle, \tilde{\mathbf{V}}_i \left|\psi_i^d\right\rangle\right) \leq \sqrt{1-(1-2\,\mathsf{W}_{\text{val},i})^2} \leq 2\sqrt{\mathsf{W}_{\text{val},i}}\;. \tag{6.50}$$

Before the first decryption query, the states of the adversary in both scenario Q0 and Q1 is identical, $\left|\psi_1^d\right\rangle = \mathbf{U}_0 \left|s\right\rangle$. Before the second decryption query, the states are $\left|\widetilde{\psi_2^d}\right\rangle = \mathbf{U}_1\tilde{\mathbf{V}}_1 \left|\psi_1^d\right\rangle$ and $\left|\psi_2^d\right\rangle = \mathbf{U}_1\mathbf{V}_1 \left|\psi_1^d\right\rangle$, respectively. Therefore, for the trace distance we have

$$\mathsf{D}\left(\left|\psi_2^d\right\rangle, \left|\widetilde{\psi_2^d}\right\rangle\right) = \mathsf{D}\left(\mathbf{V}_1 \left|\psi_1^d\right\rangle, \tilde{\mathbf{V}}_1 \left|\psi_1^d\right\rangle\right) \leq 2\sqrt{\mathsf{W}_{\text{val},1}}\;. \tag{6.51}$$

For arbitrary $i > 0$, the triangle inequality gives us

$$\begin{aligned}
\mathsf{D}\left(\left|\psi_{i+1}\right\rangle, \left|\widetilde{\psi_{i+1}}\right\rangle\right) &= \mathsf{D}\left(\mathbf{U}_i\mathbf{V}_i \left|\psi_i\right\rangle, \mathbf{U}_i\tilde{\mathbf{V}}_i \left|\widetilde{\psi_i}\right\rangle\right) \\
&= \mathsf{D}\left(\mathbf{V}_i \left|\psi_i\right\rangle, \tilde{\mathbf{V}}_i \left|\widetilde{\psi_i}\right\rangle\right) \\
&\leq \mathsf{D}\left(\mathbf{V}_i \left|\psi_i\right\rangle, \tilde{\mathbf{V}}_i \left|\psi_i\right\rangle\right) + \mathsf{D}\left(\tilde{\mathbf{V}}_i \left|\psi_i\right\rangle, \tilde{\mathbf{V}}_i \left|\widetilde{\psi_i}\right\rangle\right) \\
&= \mathsf{D}\left(\mathbf{V}_i \left|\psi_i\right\rangle, \tilde{\mathbf{V}}_i \left|\psi_i\right\rangle\right) + \mathsf{D}\left(\left|\psi_i\right\rangle, \left|\widetilde{\psi_i}\right\rangle\right) \\
&\leq 2\sqrt{\mathsf{W}_{\text{val},i}} + \mathsf{D}\left(\left|\psi_i\right\rangle, \left|\widetilde{\psi_i}\right\rangle\right)\;.
\end{aligned} \tag{6.52}$$

By induction, it follows that

$$\mathsf{D}\left(\left|\psi_{q_d}\right\rangle, \left|\widetilde{\psi_{q_d}}\right\rangle\right) \leq 2\sum_{i=1}^{q_d-1} \sqrt{\mathsf{W}_{\text{val},i}}\;. \tag{6.53}$$

This implies that, for any measurement, the probabilities for $b' = 1$ in both scenario can not differ by more than $2\sum_{i=1}^{q_d-1}\sqrt{\mathsf{W}_{\text{val},i}}$.

$$\Pr\left[\mathbf{Exp}_{\widetilde{\mathcal{SE}}}^{Q1}(\mathcal{A}) = 1\right] \leq \Pr\left[\mathbf{Exp}_{\widetilde{\mathcal{SE}}}^{Q0}(\mathcal{A}) = 1\right] + 2\sum_{i=1}^{q_d-1}\sqrt{\mathsf{W}_{\text{val},i}}\;. \tag{6.54}$$

Now the expectation of that quantity is

$$\begin{aligned} \mathsf{E}\left(2\sum_{i=1}^{q_d-1}\sqrt{\mathsf{W}_{\mathrm{val},i}}\right) &= 2\sum_{i=1}^{q_d-1}\mathsf{E}\left(\sqrt{\mathsf{W}_{\mathrm{val},i}}\right) \\ &\leq 2^{-n_\tau/2}4\sum_{i=1}^{q_d-1} i \\ &\leq 2q_d^2\,2^{-n_\tau/2}\,. \end{aligned} \tag{6.55}$$

Using the Markov inequality, this implies

$$\begin{aligned} \Pr\left(2\sum_{i=1}^{q_d-1}\sqrt{\mathsf{W}_{\mathrm{val},i}} \geq \xi\right) &\leq \frac{1}{\xi}\mathsf{E}\left(2\sum_{i=1}^{q_d-1}\sqrt{\mathsf{W}_{\mathrm{val},i}}\right) \\ &\leq \frac{2}{\xi}q_d^2\,2^{-n_\tau/2}\,. \end{aligned} \tag{6.56}$$

That is, with probability at least $1 - \frac{2}{\xi}q_d^2\,2^{-n_\tau/2}$, we have that

$$\begin{aligned} \Pr\left[\mathbf{Exp}^{Q1}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) = 1\right] &= \left(\Pr\left[\mathbf{Exp}^{Q1}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) = 1 \middle| 2\sum_{i=1}^{q_d-1}\sqrt{\mathsf{W}_{\mathrm{val},i}} \geq \xi\right]\right. \\ &\qquad \left.\cdot\Pr\left[2\sum_{i=1}^{q_d-1}\sqrt{\mathsf{W}_{\mathrm{val},i}} \geq \xi\right]\right) \\ &\qquad + \left(\Pr\left[\mathbf{Exp}^{Q1}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) = 1 \middle| 2\sum_{i=1}^{q_d-1}\sqrt{\mathsf{W}_{\mathrm{val},i}} < \xi\right]\right. \\ &\qquad \left.\cdot\Pr\left[2\sum_{i=1}^{q_d-1}\sqrt{\mathsf{W}_{\mathrm{val},i}} < \xi\right]\right) \\ &\leq 1\cdot\frac{2}{\xi}q_d^2\,2^{-n_\tau/2} + \left(\Pr\left[\mathbf{Exp}^{Q0}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) = 1\right] + \xi\right)\cdot 1\,. \end{aligned} \tag{6.57}$$

Hence

$$\Pr\left[\mathbf{Exp}^{Q1}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) = 1\right] \leq \Pr\left[\mathbf{Exp}^{Q0}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) = 1\right] + \xi + \frac{2}{\xi}q_d^2\,2^{-n_\tau/2}\,. \tag{6.58}$$

We can now choose ξ so that this has the best form. One possibility is $\xi = 2^{-n_\tau/4}$, which leads to

$$\Pr\left[\mathbf{Exp}^{Q1}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) = 1\right] \leq \Pr\left[\mathbf{Exp}^{Q0}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) = 1\right] + \left(1 + 2\, q_d^2\right)\, 2^{-n_\tau/4}\,. \tag{6.59}$$

This bound holds irrespective of the chosen bit b in the experiment. Since the advantages are defined as

$$\mathbf{Adv}^{Q1}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) = 2\Pr\left[\mathbf{Exp}^{Q1}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) = 1\right] - 1\,, \tag{6.60}$$

and similarly

$$\mathbf{Adv}^{Q0}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) = 2\Pr\left[\mathbf{Exp}^{Q0}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) = 1\right] - 1\,. \tag{6.61}$$

Therefore

$$\mathbf{Adv}^{Q1}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) \leq \mathbf{Adv}^{Q0}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) + 2\left(1 + 2\, q_d^2\right)\, 2^{-n_\tau/4}\,. \tag{6.62}$$

From Equation 6.30 we get

$$\mathbf{Adv}^{rop-qscca}_{\widetilde{\mathcal{SE}}}(\mathcal{A}) \leq \mathbf{Adv}^{rop-qscpa}_{\widetilde{\mathcal{SE}}}(\mathcal{B}) + 2\left(1 + 2\, q_d^2\right)\, 2^{-n_\tau/4} \tag{6.63}$$

which concludes the proof of the claim. ∎

Now we look at the original experiment. Assume that $\mathcal{A}$ is attacking RoP-qsCCA security of $\mathcal{SE}'$ in Construction 4. The challenger maintains the experiment.

The quantum adversary, $\mathcal{A}$, makes adaptive quantum queries of its choice. The queries can be either encryption or decryption queries. The oracles responds to each query by applying a unitary transformation to the adversary's quantum register.

In the case of encryption queries, the unitary transformation depends on the bit b. If $b = 1$, then the encryption oracle applies the unitary $\mathbf{U}_{\mathcal{E}_{K_1}}$, followed by $\mathbf{U}_{F_{K_2}}$:

$$\sum_m \alpha_m |m, 0, 0\rangle \xrightarrow{\mathbf{U}_{F_{K_2}(\cdot)}\mathbf{U}_{\mathcal{E}_{K_1}(\cdot)}} \sum_m \alpha_m |m, \mathcal{E}_{K_1}(m), F_{K_2}(\mathcal{E}_{K_1}(m))\rangle \ , \tag{6.64}$$

where $c = \mathcal{E}_{K_1}(m)$ and $\tau = F_{K_2}(\mathcal{E}_{K_1}(m))$.

If $b = 0$, the encryption oracle chooses a permutation $\Pi \leftarrow\!\!\$\ \mathsf{Perm}(n)$, and applies the unitary $\mathbf{U}_{\mathcal{E}_{K_1}(\Pi(\cdot))}$, followed by $\mathbf{U}_{F_{K_2}}$:

$$\sum_m \alpha_m |m, 0, 0\rangle \xrightarrow{\mathbf{U}_{F_{K_2}(\cdot)}\mathbf{U}_{\mathcal{E}_{K_1}(\Pi(\cdot))}} \sum_m \alpha_m |m, \mathcal{E}_{K_1}(\Pi(m)), F_{K_2}(\mathcal{E}_{K_1}(\Pi(m)))\rangle \ , \tag{6.65}$$

where $c = \mathcal{E}_{K_1}(\Pi(m))$ and $\tau = F_{K_2}(\mathcal{E}_{K_1}(\Pi(m)))$.

In the case of decryption queries, the decryption oracle applies the unitary $\mathbf{U}_{\mathcal{D}'_{K_1,K_2}(\cdot,\cdot)}$ to the adversary's quantum register:

$$\sum_{c,\tau} \alpha_{c,\tau} |c, \tau, 0\rangle \xrightarrow{\mathbf{U}_{\mathcal{D}'_{K_1,K_2}(\cdot,\cdot)}} \sum_{c,\tau} \alpha_{c,\tau} \left|c, \tau, \mathcal{D}'_{K_1,K_2}(c, \tau)\right\rangle \ , \tag{6.66}$$

where

$$\mathcal{D}'_{K_1,K_2}(c, \tau) = \begin{cases} m \leftarrow \mathcal{D}_{K_1}(c) & \textbf{if } F_{K_2}(c) = \tau \\ \perp & \textbf{otherwise} \end{cases} .$$

Eventually the quantum adversary $\mathcal{A}$ outputs a guess b' for b. The advantage of the adversary is

$$\mathbf{Adv}_{\mathcal{SE}'}^{rop-qscca}(\mathcal{A}) = \Pr\left[\mathbf{Exp}_{\mathcal{SE}'}^{rop-qscca-1}(\mathcal{A}) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}'}^{rop-qscca-0}(\mathcal{A}) = 1\right] \ . \tag{6.67}$$

Assume that $\mathcal{A}$ is attacking RoP-qsCCA security of $\mathcal{SE}'$ in Construction 4. We construct a new quantum adversary $\mathcal{J}$, using $\mathcal{A}$, to attack QPRF security of F. $\mathcal{J}$ runs $\mathcal{A}$, and uses its oracle to provide a simulation of $\mathcal{A}$'s oracles in RoP-qsCCA experiment.

The challenger maintains the QPRF experiment. The quantum adversary $\mathcal{J}$ chooses a bit $d \leftarrow\$ \{0,1\}$ and a key $K_1 \leftarrow\$ \mathcal{K}$. It is also assumed that $\mathcal{J}$ simulates $\mathcal{E}$ and $\mathcal{D}$ of Construction 4 perfectly. $\mathcal{A}$ adaptively makes encryption or decryption quantum queries.

In the case of encryption queries, if $d = 1$ then $\mathcal{J}$ applies $\mathbf{U}_{\mathcal{E}_{K_1}}$ on the first $2n$ qubits of $\mathcal{A}$'s quantum register. Otherwise, when $d = 0$, $\mathcal{J}$ applies $\mathbf{U}_{\mathcal{E}_{K_1}(\Pi(\cdot))}$ to $\mathcal{A}$'s quantum register, where $\Pi \leftarrow\$ \mathsf{Perm}(n)$ is chosen by $\mathcal{J}$. Then $\mathcal{J}$ sends the quantum register to its oracle. The oracle applies the unitary $\mathbf{U}_{F_{K_2}}$ or $\mathbf{U}_f$ to the last $2n$ bits of the quantum register when $b = 1$ or $b = 0$ respectively.

In the case of decryption queries, $\mathcal{J}$ sends $\mathcal{A}$'s quantum register to the oracle where it applies $\mathbf{U}_{F_{K_2}}$ or $\mathbf{U}_f$ to the register when $b = 1$ or $b = 0$ respectively,

$$\sum_c \alpha_c \left|0, c, \tau, 0\right\rangle \xrightarrow{\mathbf{U}_{f(c)} \text{ or } \mathbf{U}_{F_{K_2}(c)}} \sum_c \alpha_c \left|0, c, \tau, \tau'\right\rangle \ . \tag{6.68}$$

Then $\mathcal{J}$ applies $\mathbf{U}_{\widetilde{\mathcal{D}}_{K_1}(c)}$ to the quantum register:

$$\sum_c \alpha_c \left|0, c, \tau, \tau'\right\rangle \xrightarrow{\mathbf{U}_{\widetilde{\mathcal{D}}_{K_1}(c)}} \sum_c \alpha_c \left|\widetilde{\mathcal{D}}_{K_1}(c), c, \tau, \tau'\right\rangle \ , \tag{6.69}$$

where

$$\widetilde{\mathcal{D}}_{K_1}(c) = \begin{cases} m \leftarrow \mathcal{D}_{K_1}(c) & \textbf{if } \tau = \tau' \\ \perp & \textbf{otherwise} \end{cases} .$$

Eventually $\mathcal{A}$ outputs a bit d' for d. If $d = d'$, $\mathcal{J}$ outputs 1. Otherwise it outputs 0. For the advantage of $\mathcal{J}$ we have:

$$\mathbf{Adv}_F^{qprf}(\mathcal{J}) = \Pr\left[\mathbf{Exp}_F^{qprf-1}(\mathcal{J}) = 1\right] - \Pr\left[\mathbf{Exp}_F^{qprf-0}(\mathcal{J}) = 1\right] . \tag{6.70}$$

When $b = 0$, we can see that $\mathcal{J}$ simulates the RoP-qsCCA experiment for $\mathcal{A}$ when $\mathcal{A}$ is attacking $\widetilde{\mathcal{SE}}$. Then

$$\Pr\left[\mathbf{Exp}_F^{qprf-0}(\mathcal{J}) = 1\right] = \frac{1}{2} + \frac{1}{2} \cdot \mathbf{Adv}_{\widetilde{\mathcal{SE}}}^{rop-qscca}(\mathcal{A}) . \tag{6.71}$$

When $b = 1$, we can see that $\mathcal{J}$ simulates the RoP-qsCCA experiment for $\mathcal{A}$ when $\mathcal{A}$ is attacking $\mathcal{SE}'$. Then,

$$\Pr\left[\mathbf{Exp}_F^{qprf-1}(\mathcal{A}) = 1\right] = \frac{1}{2} + \frac{1}{2} \cdot \mathbf{Adv}_{\mathcal{SE}'}^{rop-qscca}(\mathcal{A}) . \tag{6.72}$$

Hence,

$$\begin{aligned} \mathbf{Adv}_F^{qprf}(\mathcal{J}) &= \frac{1}{2} \cdot \mathbf{Adv}_{\mathcal{SE}'}^{rop-qscca}(\mathcal{A}) - \frac{1}{2} \cdot \mathbf{Adv}_{\widetilde{\mathcal{SE}}}^{rop-qscca}(\mathcal{A}) \\ 2 \cdot \mathbf{Adv}_F^{qprf}(\mathcal{J}) &\geq \mathbf{Adv}_{\mathcal{SE}'}^{rop-qscca}(\mathcal{A}) - \mathbf{Adv}_{\widetilde{\mathcal{SE}}}^{rop-qscca}(\mathcal{A}) \\ \mathbf{Adv}_{\mathcal{SE}'}^{rop-qscca}(\mathcal{A}) &\leq 2 \cdot \mathbf{Adv}_F^{qprf}(\mathcal{J}) + \mathbf{Adv}_{\mathcal{SE}}^{rop-qscpa}(\mathcal{B}) + 2\left(1 + 2q_d^2\right) 2^{-n_\tau/4} . \end{aligned} \tag{6.73}$$

Finally we get

$$\mathbf{Adv}_{\mathcal{SE}}^{rop-qscpa}(\mathcal{B}) + 2 \cdot \mathbf{Adv}_F^{qprf}(\mathcal{J}) \geq \mathbf{Adv}_{\mathcal{SE}'}^{rop-qscca}(\mathcal{A}) - 2\left(1 + 2q_d^2\right) 2^{-n_\tau/4} , \tag{6.74}$$

which concludes the proof. ▮

6.3 Semantic Security in a Quantum Setting

The idea of semantic security (see Subsection 4.2.1.4) is that having access to a ciphertext should not provide any advantage to an adversary $\mathcal{A}$ who is trying to obtain information about the plaintext. This is formalised in a game where the adversary chooses a function f and tries to predict the value $f(m)$ for a message m chosen at random from a set $\mathcal{M}$, where $\mathcal{M}$ is also chosen by the adversary. The adversary's advantage is defined by comparing its success probability in two experiments, one in which $\mathcal{A}$ is provided with the encryption of m, and one in which $\mathcal{A}$ is provided instead with the encryption of a message m' which is also chosen randomly, and independently of m, from the set $\mathcal{M}$. For this definition to be meaningful, the set $\mathcal{M}$ must be valid in the sense that all messages in $\mathcal{M}$ must have the same length.

We now give a closely analogous definition of semantic security against a quantum adversary. The main difference from the original definition is that we allow the set $\mathcal{M}$ to contain quantum superpositions of messages. As in the classical definition, it will be necessary to restrict the set $\mathcal{M}$ in order to arrive at a meaningful definition.

Definition 12 Let $\mathcal{M}$ be a set of superpositions of n-bit messages of the form

$$|\psi\rangle = \sum_{m=0}^{2^n-1} \alpha_m |m\rangle \ , \tag{6.75}$$

and let $\mathsf{Perm}\,(n)$ be the set of all permutations of $\{0,1\}^n$. The set $\mathcal{M}$ is called *valid* if there is a state $|\psi\rangle$ and a subset $P \subseteq \mathsf{Perm}\,(n)$ such that

$$\mathcal{M} = \{\mathbf{U}_\Pi |\psi\rangle \ : \ \Pi \in P\} \ , \tag{6.76}$$

where $\mathbf{U}_\Pi$ is a unitary transformation. In other words, $\mathcal{M}$ is valid if all its elements are permutations of a given quantum state $|\psi\rangle$. ▮

Our definition of valid sets is just one of many possibilities. It is strictly larger than the set of classical messages, which here would correspond to choosing a computational basis state for $|\psi\rangle$. It is probably not the largest possible set that leads to an achievable notion. Which choices of message space lead to an achievable notion of semantic security remains an open question.

Here is an example that shows why there must be some restriction on the allowed sets $\mathcal{M}$. Let m_0 be some message, and let $|\psi_+\rangle = 2^{-n/2} \sum_m |m\rangle$ be the equal superposition of all 2^n messages. Then, for a permutation $\Pi \leftarrow\$ \, \mathsf{Perm}\,(n)$, we have

$$\mathbf{U}_\Pi \, |\psi_+\rangle = |\psi_+\rangle \ , \tag{6.77}$$

i.e., $|\psi_+\rangle$ is invariant under any permutation. On the other hand,

$$\langle m_0 | \mathbf{U}_\Pi | m_0 \rangle = 0 \tag{6.78}$$

with probability close to 1. Thus, a quantum adversary can easily tell which of the two states $|m_0\rangle$ and $|\psi_+\rangle$ was encrypted. This example is similar to the proof of Theorem 4.2 in [29].

Before explaining the definition, we define a number of notations. In Subsection 3.1.1, we explained that a quantum circuit is a quantum gate sequence. The size of a quantum circuit is the number of the elementary quantum gates in the circuit, where the elementary quantum gates are chosen from a universal set of gates. We stipulate that a quantum circuit is executed by a particular *universal quantum circuit evaluator*, or **UQE**. The action of our **UQE** consists of applying a quantum operation specified by a string $x \in \{0,1\}^*$ to a quantum register Q. We will denote this action by $\mathbf{UQE}\,(x, Q)$. Optionally, the universal quantum circuit evaluator returns an output string y, which we will indicate by $y \leftarrow \mathbf{UQE}\,(x, Q)$. The output y depends on the input quantum circuit, but in general, y will be randomised, simply because in order to get the output from a quantum computation, one has to make a measurement.

Assume a quantum adversary that plays the experiments SEM-qsCPA and SEM-qsCCA shown in Figure 6.3. Both experiments begin by choosing a bit $b \in \{0, 1\}$ that parametrises the experiments, and a key $K \leftarrow \mathcal{K}$. The quantum adversary runs in two phases, select and predict, where it is given quantum superposition access to its oracles.

During the select phase, the quantum adversary adaptively requests encryptions of quantum queries of its choice. The encryption oracle responds to each encryption query by applying the unitary $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to the first $(2n + n_r)$ qubits of the quantum adversary's quantum register, where n is the length of the encryption query and n_r is the length of the randomness used by the encryption oracle to encrypt the query. Additionally, the quantum adversary is given quantum superposition access to a decryption oracle in the SEM-qsCCA experiment. The quantum adversary can query the decryption oracle on any ciphertext. At the end of the select phase, the quantum adversary outputs a classical description of a set P of permutations as well as a quantum circuit description R of a state $|\psi\rangle$.

At the beginning of the predict phase, the challenger executes $\mathbf{UQE}\left(R, Q_{\mathcal{A}}\right)$, preparing the state $|\psi\rangle$ in the adversary's quantum register $Q_{\mathcal{A}}$. The challenger then chooses two permutations Π_0 and Π_1 at random from the set P, executes $\mathbf{UQE}\left(\Pi_0, Q_{\mathcal{A}}\right)$, thus applying the unitary $\mathbf{U}_{\Pi_0}$ to $Q_{\mathcal{A}}$, and finally applies the encryption oracle $\mathcal{E}_K$ to $Q_{\mathcal{A}}$. Note that Π_0 and Π_1 are in the form of quantum circuit descriptions. The adversary's quantum register now contains the state $\mathbf{U}_{\mathcal{E}_K}|\psi_0\rangle$, where $|\psi_b\rangle = \mathbf{U}_{\Pi_b}|\psi\rangle \in \mathcal{M}$ and $b \in \{0, 1\}$. We call this the *challenge ciphertext*. During the predict phase, the quantum adversary is again given superposition access to the encryption oracle. Additionally, the quantum adversary is given quantum superposition access to a decryption oracle in the SEM-qsCCA experiment. The quantum adversary can query the decryption oracle on any ciphertext as long as the condition given in Equation 6.21 is satisfied. At the end of this phase, the quantum adversary outputs the description of a quantum circuit V, and a bit z.

The challenger now executes $\mathbf{UQE}\left(R, Q_{\mathcal{A}}\right)$, again preparing the state $|\psi\rangle$ in the register $Q_{\mathcal{A}}$. The challenger then runs $\mathbf{UQE}\left(\Pi_b, Q_{\mathcal{A}}\right)$, thus applying the unitary $\mathbf{U}_{\Pi_b}$ to $Q_{\mathcal{A}}$, which means that $Q_{\mathcal{A}}$ now contains the state $|\psi_b\rangle \in \mathcal{M}$. Finally, the challenger runs $z' \leftarrow \mathbf{UQE}\left(V, Q_{\mathcal{A}}\right)$, thus generating an output bit z'. The experiment then returns 1 ('success') if $z' = z$, i.e., if the adversary guessed z' correctly, and 0 otherwise.

Definition 13 [**SEM-qsCPA and SEM-qsCCA**] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. Define experiment $\mathbf{Exp}_{\mathcal{SE}}^{sem-qscpa-b}(\mathcal{A})$ and experiment $\mathbf{Exp}_{\mathcal{SE}}^{sem-qscca-b}(\mathcal{A})$ for a quantum adversary $\mathcal{A}$ and a bit b as depicted in Figure 6.3. In the experiments, the adversary $\mathcal{A}$ is given quantum superposition access to an encryption oracle. The encryption oracle responds to each query by applying a unitary transformation to the first $(2n + n_r)$ qubits of the adversary's quantum register $Q_{\mathcal{A}}$. The adversary $\mathcal{A}$ is additionally given quantum superposition access to a decryption oracle in the latter experiment. For any challenge ciphertext $\sum_c \lambda_c |m_c, c\rangle$, we define the projector $\mathsf{Proj}_c = \sum_x |x, c\rangle\langle x, c|$. No restriction is imposed on the quantum adversary's queries, except, it is assumed that

$$\Pr\left[\exists c : \lambda_c \neq 0 \text{ and } \mathrm{Tr}\left(\mathsf{Proj}_c\, \rho\right) \neq 0\right] = 0 \ ,$$

where ρ is the state of $Q_{\mathcal{A}}$ before making any decryption query.

The corresponding advantages of a quantum adversary $\mathcal{A}$ are given by:

$$\mathbf{Adv}_{\mathcal{SE}}^{sem-qscpa}(\mathcal{A}) = \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qscpa-1}(\mathcal{A}) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qscpa-0}(\mathcal{A}) = 1\right] \ ,$$
$$\mathbf{Adv}_{\mathcal{SE}}^{sem-qscca}(\mathcal{A}) = \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qscca-1}(\mathcal{A}) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qscca-0}(\mathcal{A}) = 1\right] \ .$$

These advantages refer to a specific quantum adversary using resources as discussed in Section 3.4. ∎

The relevant resources for the quantum adversary $\mathcal{A}$ include the running time t, which includes the maximum running time of V where the maximum is taken over all states $|\psi\rangle$ in $\mathcal{M}$, the numbers q_e of encryption and q_d of decryption oracle queries, and the size of the classical output, $\mu = \mu_V + \mu_R + \mu_P$ bits, where μ_V and μ_R are the maximum number of bits required for the description of V and R respectively and $\mu_P = 2 \cdot \mu_\Pi$, where μ_Π is the maximum number of bits required for a permutation Π output by P.

```
$\underline{\text{Experiment } \mathbf{Exp}_{\mathcal{SE}}^{sem-qscpa-b}(\mathcal{A})}$ $\boxed{\underline{\text{Experiment } \mathbf{Exp}_{\mathcal{SE}}^{sem-qscca-b}(\mathcal{A})}}$
    $K \leftarrow \mathcal{K}$
    $(R, P) \leftarrow \mathcal{A}^{\mathcal{E}_{K,Q_{\mathcal{A}}}()}(\mathsf{select})$ $\boxed{(R, P) \leftarrow \mathcal{A}^{\mathcal{E}_{K,Q_{\mathcal{A}}}(),\mathcal{D}_{K,Q_{\mathcal{A}}}()}(\mathsf{select})}$
    $\Pi_0 \leftarrow_{\$} P$; $\Pi_1 \leftarrow_{\$} P$
    Run $\mathbf{UQE}(R, Q_{\mathcal{A}})$
    Run $\mathbf{UQE}(\Pi_0, Q_{\mathcal{A}})$
    Apply $\mathbf{U}_{\mathcal{E}_K}$ to $Q_{\mathcal{A}}$
    $(V, z) \leftarrow \mathcal{A}^{\mathcal{E}_{K,Q_{\mathcal{A}}}()}(\mathsf{predict})$ $\boxed{(V, z) \leftarrow \mathcal{A}^{\mathcal{E}_{K,Q_{\mathcal{A}}}(),\mathcal{D}_{K,Q_{\mathcal{A}}}()}(\mathsf{predict})}$
    Run $\mathbf{UQE}(R, Q_{\mathcal{A}})$
    Run $\mathbf{UQE}(\Pi_b, Q_{\mathcal{A}})$
    $z' \leftarrow \mathbf{UQE}(V, Q_{\mathcal{A}})$
    if $z = z'$ then
        $b' \leftarrow 1$
    else
        $b' \leftarrow 0$
    end if
    return $b'$

$\underline{\mathcal{E}_{K,Q_{\mathcal{A}}}()}$
    Apply $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to $Q_{\mathcal{A}}$
    return

$\underline{\mathcal{D}_{K,Q_{\mathcal{A}}}()}$
    Apply $\mathbf{U}_{\mathcal{D}_K(\cdot)}$ to $Q_{\mathcal{A}}$
    return
```

Figure 6.3: The SEM-qsCPA and SEM-qsCCA confidentiality notions. The boxed codes are excluded in SEM-qsCPA experiment, whereas they replace the codes adjacent to them in SEM-qsCCA experiment.

6.4 Relations Among Notions

In Subsection 4.2.1.6, we proved that classical RoP and RoR security notions are equivalent. From this, we can deduce that classical RoP also implies SEM security. Here we prove that the quantum analogue of RoP also implies our quantum analogue of SEM security.

Theorem 18 (RoP-qsATK ⇒ SEM-qsATK) *For any scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, assume that $\mathcal{A}_2$ is a quantum adversary attacking $\mathcal{SE}$ in the SEM-qsATK sense, with a running time of at most t_2, making at most q_e encryption and (in the CCA case) q_d decryption queries to the oracle, and the size of the classical output of μ_2 bits, and having advantage*

$$\begin{aligned}\mathbf{Adv}_{\mathcal{SE}}^{sem-qscpa}(\mathcal{A}_2) &\geq \epsilon_2 \ ,\\ \mathbf{Adv}_{\mathcal{SE}}^{sem-qscca}(\mathcal{A}_2) &\geq \epsilon_2 \ .\end{aligned}$$

Then there exists a quantum adversary $\mathcal{A}_1$ attacking $\mathcal{SE}$ in the RoP-qsATK sense, with a running time t_1 of at most $t_2 + q_e c\left(\frac{3}{2}\mu_2 + \mu_{\Pi'}\right)$, making at most q_e encryption and (in the CCA case) q_d decryption queries to the oracle, and the size of the classical output of $\mu_1 = \mu_2 + 2\mu_{\Pi'}$ bits, and having advantage

$$\begin{aligned}\mathbf{Adv}_{\mathcal{SE}}^{rop-qscpa}(\mathcal{A}_1) &\geq \frac{\epsilon_2}{4} \ ,\\ \mathbf{Adv}_{\mathcal{SE}}^{rop-qscca}(\mathcal{A}_1) &\geq \frac{\epsilon_2}{4} \ .\end{aligned}$$

Here, c is a constant, and $\mu_{\Pi'}$ is the maximum number of bits required for the description of a permutation Π'.

Before proving Theorem 18, we propose two more notions that will help us in the proof. We name these new notions ‘FtG’ and ‘LoR’.

Assume a quantum adversary that plays the experiments FtG-qsCPA and FtG-qsCCA shown in Figure 6.4. Both experiments begin with choosing a key $K \leftarrow \mathcal{K}$ and a bit $b \in \{0,1\}$. The quantum adversary runs in two phases, find and guess, where it is given quantum superposition access to its oracles.

During the find phase the quantum adversary adaptively requests encryptions of quantum queries of its choice. The encryption oracle responds to each encryption query by applying the unitary $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to the first $(2n + n_r)$ qubits of the quantum adversary's quantum register, where n is the length of the encryption query and n_r is the length of the randomness used by the encryption oracle to encrypt the query. Additionally, the quantum adversary is given quantum superposition access to a decryption oracle in the FtG-qsCCA experiment. The quantum adversary can query the decryption oracle on any ciphertext. At the end of the find phase, the quantum adversary outputs quantum circuit descriptions of two permutations Π_0, $\Pi_1 : \{0,1\}^n \rightarrow \{0,1\}^n$, and also it prepares an n qubit quantum query in its quantum register.

At the beginning of the guess phase, the challenger executes $\mathbf{UQE}\left(\Pi_b, Q_{\mathcal{A}}\right)$ which applies $\mathbf{U}_{\Pi_b}$ to the adversary's quantum register. Then the challenger applies the unitary $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to the adversary's quantum register. We call the result the *challenge ciphertext*. During the guess phase, the quantum adversary is again given superposition access to the encryption oracle. Additionally, the quantum adversary is given quantum superposition access to a decryption oracle in the FtG-qsCCA experiment. The quantum adversary can query the decryption oracle on any ciphertext as long as the condition given in Equation 6.21 is satisfied. At the end of this phase, the quantum adversary outputs a bit b'.

Definition 14 [**FtG-qsCPA and FtG-qsCCA**] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. Define experiment $\mathbf{Exp}_{\mathcal{SE}}^{ftg-qscpa-b}(\mathcal{A})$ and experiment $\mathbf{Exp}_{\mathcal{SE}}^{ftg-qscca-b}(\mathcal{A})$ for a quantum adversary $\mathcal{A}$ and a bit b as depicted in Figure 6.4. In the experiments, the adversary $\mathcal{A}$ is given quantum superposition access to an encryption oracle. The encryption oracle responds to each query by applying a unitary transformation to the first $(2n + n_r)$ qubits of the adversary's quantum register $Q_{\mathcal{A}}$. $\mathcal{A}$ is additionally given quantum superposition access to a decryption oracle in the latter experiment. For any challenge ciphertext $\sum_c \lambda_c \left|m_c, c\right\rangle$, we define the projector

$\mathsf{Proj}_c = \sum_x |x, c\rangle\langle x, c|$. No restriction is imposed on the quantum adversary's queries, except, it is assumed that

$$\Pr\left[\exists c : \lambda_c \neq 0 \text{ and } \mathrm{Tr}\left(\mathsf{Proj}_c\, \rho\right) \neq 0\right] = 0 \ ,$$

where ρ is the state of $Q_{\mathcal{A}}$ before making any decryption query.

The adversary's goal is to output a bit b' as its guess of the challenge bit b, and the experiment returns b' as well. The corresponding advantages of a quantum adversary $\mathcal{A}$ are given by:

$$\begin{aligned}\mathbf{Adv}^{ftg-qscpa}_{\mathcal{SE}}(\mathcal{A}) &= \Pr\left[\mathbf{Exp}^{ftg-qscpa-1}_{\mathcal{SE}}(\mathcal{A}) = 1\right] - \Pr\left[\mathbf{Exp}^{ftg-qscpa-0}_{\mathcal{SE}}(\mathcal{A}) = 1\right] \ , \\ \mathbf{Adv}^{ftg-qscca}_{\mathcal{SE}}(\mathcal{A}) &= \Pr\left[\mathbf{Exp}^{ftg-qscca-1}_{\mathcal{SE}}(\mathcal{A}) = 1\right] - \Pr\left[\mathbf{Exp}^{ftg-qscca-0}_{\mathcal{SE}}(\mathcal{A}) = 1\right] \ .\end{aligned}$$

These advantages refer to a specific quantum adversary using resources as discussed in Section 3.4. ▮

We now give the LoR notion. Assume a quantum adversary that plays the experiments LoR-qsCPA and LoR-qsCCA shown in Figure 6.5. Both experiments begin with choosing a key $K \leftarrow \mathcal{K}$ and a bit $b \in \{0, 1\}$. The quantum adversary is given quantum superposition access to an encryption oracle. The quantum adversary adaptively requests encryptions of quantum queries of its choice. Also, for each query, the quantum adversary outputs quantum circuit description of two permutations Π_0, $\Pi_1 : \{0,1\}^n \to \{0,1\}^n$. The challenger executes $\mathbf{UQE}\left(\Pi_b, Q_{\mathcal{A}}\right)$, which applies $\mathbf{U}_{\Pi_b}$ to the adversary's quantum register. Then the challenger applies the unitary $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to the first $(2n + n_r)$ qubits of the adversary's quantum register. We call the result, the *challenge ciphertext.*

Additionally, the quantum adversary is given quantum superposition access to a decryption oracle in the LoR-qsCCA experiment. The quantum adversary can query the decryption oracle on any ciphertext as long as the condition given in Equation 6.21 is satisfied. At some point the quantum adversary outputs a bit b'.

```
Experiment $\mathbf{Exp}_{\mathcal{SE}}^{ftg-qscpa-b}(\mathcal{A})$ $\boxed{\text{Experiment } \mathbf{Exp}_{\mathcal{SE}}^{ftg-qscca-b}(\mathcal{A})}$
    $K \leftarrow \mathcal{K}$
    $((\Pi_0,\Pi_1),Q_{\mathcal{A}}) \leftarrow \mathcal{A}^{\mathcal{E}_{K,Q_{\mathcal{A}}}()}$ (find)
    $\boxed{((\Pi_0,\Pi_1),Q_{\mathcal{A}}) \leftarrow \mathcal{A}^{\mathcal{E}_{K,Q_{\mathcal{A}}}(),\mathcal{D}_{K,Q_{\mathcal{A}}}()}\ \text{(find)}}$
    Run $\mathbf{UQE}(\Pi_b, Q_{\mathcal{A}})$
    Apply $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to $Q_{\mathcal{A}}$
    $b' \leftarrow \mathcal{A}^{\mathcal{E}_{K,Q_{\mathcal{A}}}()}$ (guess) $\boxed{b' \leftarrow \mathcal{A}^{\mathcal{E}_{K,Q_{\mathcal{A}}}(),\mathcal{D}_{K,Q_{\mathcal{A}}}()}\ \text{(guess)}}$
    return $b'$

$\mathcal{E}_{K,Q_{\mathcal{A}}}()$
    Apply $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to $Q_{\mathcal{A}}$
    return

$\mathcal{D}_{K,Q_{\mathcal{A}}}()$
    Apply $\mathbf{U}_{\mathcal{D}_K(\cdot)}$ to $Q_{\mathcal{A}}$
    return
```

Figure 6.4: The FtG-qsCPA and FtG-qsCCA confidentiality notions. The boxed codes are excluded in FtG-qsCPA experiment, whereas they replace the codes adjacent to them in FtG-qsCCA experiment.

Definition 15 [**LoR-qsCPA and LoR-qsCCA**] Let $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be a symmetric encryption scheme. Define experiment $\mathbf{Exp}_{\mathcal{SE}}^{lor-qscpa-b}(\mathcal{A})$ and experiment $\mathbf{Exp}_{\mathcal{SE}}^{lor-qscca-b}(\mathcal{A})$ for an adversary $\mathcal{A}$ and a bit b as depicted in Figure 6.5. In the experiments, $\mathcal{A}$ is given quantum superposition access to a left-or-right encryption oracle $\mathsf{LoR}_{Q_{\mathcal{A}}}(\cdot)$. The encryption oracle responds to each query by applying a unitary transformation to the first $(2n + n_r)$ bits of the adversary's quantum register $Q_{\mathcal{A}}$. The adversary $\mathcal{A}$ is additionally given quantum superposition access to a decryption oracle, $\mathsf{Dec}_{Q_{\mathcal{A}}}()$, in the latter experiment. For any challenge ciphertext $\sum_c \lambda_c |m_c, c\rangle$, we define the projector $\mathsf{Proj}_c = \sum_x |x, c\rangle\langle x, c|$. No restriction is imposed on the quantum adversary's queries except, it is assumed that

$$\Pr\left[\exists c : \lambda_c \neq 0 \text{ and } \mathrm{Tr}\left(\mathsf{Proj}_c\, \rho\right) \neq 0\right] = 0 \ ,$$

where ρ is the state of $Q_{\mathcal{A}}$ before making any decryption query.

The adversary's goal is to output a bit b' as its guess of the challenge bit b, and the experiment returns b' as well. The corresponding advantages of a quantum adversary $\mathcal{A}$ are given by:

```
$\underline{\mathbf{Exp}^{lor-qscpa-b}_{\mathcal{SE}}(\mathcal{A})}$ $\boxed{\underline{\mathbf{Exp}^{lor-qscca-b}_{\mathcal{SE}}(\mathcal{A})}}$

    $K \leftarrow \mathcal{K}$
    $b' \leftarrow \mathcal{A}^{\mathsf{LoR}_{Q_{\mathcal{A}}}(\cdot)}$ $\boxed{b' \leftarrow \mathcal{A}^{\mathsf{LoR}_{Q_{\mathcal{A}}}(\cdot),\mathsf{Dec}_{Q_{\mathcal{A}}}()}}$
    return $b'$

$\underline{\mathsf{LoR}_{Q_{\mathcal{A}}}(\Pi_0, \Pi_1)}$

    Run $\mathbf{UQE}(\Pi_b, Q_{\mathcal{A}})$
    Apply $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to $Q_{\mathcal{A}}$
    return

$\underline{\mathsf{Dec}_{Q_{\mathcal{A}}}()}$

    Apply $\mathbf{U}_{\mathcal{D}_K(\cdot)}$ to $Q_{\mathcal{A}}$
    return
```

Figure 6.5: The LoR-qsCPA and LoR-qsCCA confidentiality notions. The boxed codes are excluded in LoR-CPA experiment, whereas they replace the codes adjacent to them in LoR-CCA experiment.

$$\mathbf{Adv}^{lor-qscpa}_{\mathcal{SE}}(\mathcal{A}) = \Pr\left[\mathbf{Exp}^{lor-qscpa-1}_{\mathcal{SE}}(\mathcal{A}) = 1\right] - \Pr\left[\mathbf{Exp}^{lor-qscpa-0}_{\mathcal{SE}}(\mathcal{A}) = 1\right] ,$$
$$\mathbf{Adv}^{lor-qscca}_{\mathcal{SE}}(\mathcal{A}) = \Pr\left[\mathbf{Exp}^{lor-qscca-1}_{\mathcal{SE}}(\mathcal{A}) = 1\right] - \Pr\left[\mathbf{Exp}^{lor-qscca-0}_{\mathcal{SE}}(\mathcal{A}) = 1\right] .$$

These advantages refer to a specific quantum adversary using resources as discussed in Section 3.4. ■

Proof of Theorem 18. We prove this theorem in four steps.

Step 1 (RoP-qsATK ⇒ LoR-qsATK): For any scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, assume that $\mathcal{A}_4$ is a quantum adversary attacking $\mathcal{SE}$ in the LoR-qsATK sense, with a running time of at most t_4, making at most q_e encryption and (in the CCA case) q_d decryption queries to the oracle, and the size of the classical output of μ_4 bits, and having advantage

$$\mathbf{Adv}_{\mathcal{SE}}^{lor-qscpa}\left(\mathcal{A}_4\right) \geq \epsilon_4 \ ,$$
$$\mathbf{Adv}_{\mathcal{SE}}^{lor-qscca}\left(\mathcal{A}_4\right) \geq \epsilon_4 \ .$$

Then there exists a quantum adversary $\mathcal{A}_1$ attacking $\mathcal{SE}$ in the RoP-qsATK sense, with a running time of at most $t_1 = t_4 + q_e c \frac{\mu_4}{2}$, making at most q_e encryption and (in the CCA case) q_d decryption queries to the oracle, and the size of the classical output of $\mu_1 = \mu_4$ bits, and having advantage

$$\mathbf{Adv}_{\mathcal{SE}}^{rop-qscpa}\left(\mathcal{A}_1\right) \geq \frac{\epsilon_4}{2} \ ,$$
$$\mathbf{Adv}_{\mathcal{SE}}^{rop-qscca}\left(\mathcal{A}_1\right) \geq \frac{\epsilon_4}{2} \ .$$

Assume $\mathcal{A}_4$ is a quantum adversary attacking $\mathcal{SE}$ in the LoR-qsATK sense. We construct a new quantum adversary $\mathcal{A}_1$, using $\mathcal{A}_4$, that attacks $\mathcal{SE}$ in the RoP-qsATK sense. $\mathcal{A}_1$ runs $\mathcal{A}_4$, using its oracles to provide a simulation of $\mathcal{A}_4$'s oracles. The RoP challenger maintains the experiment.

$\mathcal{A}_1$ selects a bit $b' \leftarrow\$ \{0,1\}$, independently from bit b. $\mathcal{A}_4$ adaptively requests encryptions of quantum queries of its choice. Also for each query, it places quantum circuit descriptions of two permutations Π'_0, $\Pi'_1 : \{0,1\}^n \rightarrow \{0,1\}^n$ in its classical register. $\mathcal{A}_1$ reads $\mathcal{A}_4$'s classical register. Then it executes $\mathbf{UQE}\left(\mathbf{U}_{\Pi'_{b'}}, Q_{\mathcal{A}_4}\right)$, and invokes $\mathsf{RoP}_{Q_{\mathcal{A}_4}}(\cdot)$. If $b = 0$, the encryption oracle chooses a permutation $\Pi \leftarrow\$ \mathsf{Perm}(n)$. Then it applies $\mathbf{U}_{\mathcal{E}_K(\Pi(\cdot))}$ to the first $(2n + n_r)$ qubits of the given quantum register. If $b = 1$, the encryption oracle applies the unitary $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to the first $(2n + n_r)$ qubits of the given quantum register. We call the result of this type of query the challenge ciphertexts.

Moreover, $\mathcal{A}_4$ adaptively requests decryption of quantum queries of its choice. When this happens, the decryption oracle $\mathsf{Dec}(\cdot)$ applies the unitary $\mathbf{U}_{\mathcal{D}_K(\cdot)}$ to the quantum register.

$\mathcal{A}_4$ eventually outputs a bit d. If $b' = d$ then $\mathcal{A}_1$ outputs 1. Otherwise it outputs 0. Note that, for any challenge ciphertext $\sum_c \lambda_c \left|m_c, c\right\rangle$, we define the projector

$\mathsf{Proj}_c = \sum_x |x,c\rangle\langle x,c|$. We use ρ to denote the state of $\mathcal{A}_4$'s quantum register before making any decryption query. Then we assume that

$$\Pr\left[\exists c : \lambda_c \neq 0 \text{ and } \mathrm{Tr}\left(\mathsf{Proj}_c\, \rho\right) \neq 0\right] = 0 \ ,$$

for all quantum adversaries $\mathcal{A}_4$. For $\mathcal{A}_1$'s advantage we have:

$$\mathbf{Adv}_{\mathcal{SE}}^{rop-qsatk}\left(\mathcal{A}_1\right) = \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-qsatk-1}\left(\mathcal{A}_1\right) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-qsatk-0}\left(\mathcal{A}_1\right) = 1\right] . \tag{6.79}$$

In the case that $b = 1$, $\mathcal{A}_1$ provides a perfect simulation for $\mathcal{A}_4$. Hence, $\mathcal{A}_1$ succeeds with the same probability as $\mathcal{A}_4$. Therefore we have to calculate

$$\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-qsatk-1}\left(\mathcal{A}_1\right) = 1\right] \tag{6.80}$$

which we can rewrite based on $\mathcal{A}_4$'s probability of success,

$$\begin{aligned}\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-qsatk-1}\left(\mathcal{A}_1\right) = 1\right] = \frac{1}{2} \cdot \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{lor-qsatk-1}\left(\mathcal{A}_4\right) = 1\right] \\ + \frac{1}{2} \cdot \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{lor-qsatk-0}\left(\mathcal{A}_4\right) = 1\right] \ .\end{aligned} \tag{6.81}$$

On the other hand, in the case where $b = 0$, the encryption oracle first applies the permutation Π to the given quantum register, which results in a random permutation in the register regardless of whether it was maintained in the case $b' = 0$ or $b' = 1$. Therefore, $\mathcal{A}_1$ provides a simulation for $\mathcal{A}_4$ where $\mathcal{A}_4$'s encryption oracle in $\mathbf{Exp}^{lor-qsatk-0}$ and in $\mathbf{Exp}^{lor-qsatk-1}$ returns identically distributed answers. Hence, $\mathcal{A}_1$ outputs a random bit and succeeds with probability $\frac{1}{2}$. Therefore,

$$\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-qsatk-0}\left(\mathcal{A}_1\right) = 1\right] = \frac{1}{2} \ . \tag{6.82}$$

Then we have that,

$$
\begin{aligned}
\mathbf{Adv}_{\mathcal{SE}}^{rop-qsatk}(\mathcal{A}_1) &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-qsatk-1}(\mathcal{A}_1)=1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-qsatk-0}(\mathcal{A}_1)=1\right] \\
&= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{rop-qsatk-1}(\mathcal{A}_1)=1\right] - \frac{1}{2} \\
&= \frac{1}{2}\cdot\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{lor-qsatk-1}(\mathcal{A}_4)=1\right] \\
&\qquad\qquad + \frac{1}{2}\cdot\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{lor-qsatk-0}(\mathcal{A}_4)=0\right] - \frac{1}{2} \\
&= \frac{1}{2}\cdot\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{lor-qsatk-1}(\mathcal{A}_4)=1\right] \\
&\qquad\qquad + \frac{1}{2}\cdot\left(1-\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{lor-qsatk-0}(\mathcal{A}_4)=1\right]\right) - \frac{1}{2} \\
&= \frac{1}{2}\cdot\left(\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{lor-qsatk-1}(\mathcal{A}_4)=1\right]\right. \\
&\qquad\qquad \left. - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{lor-qsatk-0}(\mathcal{A}_4)=1\right]\right) \\
&= \frac{1}{2}\cdot\mathbf{Adv}_{\mathcal{SE}}^{lor-qsatk}(\mathcal{A}_4)\ . \qquad (6.83)
\end{aligned}
$$

The running time of $\mathcal{A}_1$ is at most $t_1 = t_4 + T_{\Pi'}$, where $T_{\Pi'} = q_e c \frac{\mu_4}{2}$ is the maximum required time to apply a permutation Π'. $\mathcal{A}_1$ makes at most q_e encryption and (in the CCA case) q_d decryption oracle queries, the size of the classical output is $\mu_1 = \mu_4$ bits where $\mu_4 = 2 \cdot \mu_{\Pi'}$.

Step 2 (LoR-qsATK ⇒ FtG-qsATK): For any scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, assume that $\mathcal{A}_3$ is a quantum adversary attacking $\mathcal{SE}$ in the FtG-qsATK sense, with a running time of at most t_3, making at most q_e encryption and (in the CCA case) q_d decryption queries to the oracle, and the size of the classical output of μ_3 bits, and having advantage

$$
\begin{aligned}
\mathbf{Adv}_{\mathcal{SE}}^{ftg-qscpa}(\mathcal{A}_3) &\geq \epsilon_3\ , \\
\mathbf{Adv}_{\mathcal{SE}}^{ftg-qscca}(\mathcal{A}_3) &\geq \epsilon_3\ .
\end{aligned}
$$

Then there exists a quantum adversary $\mathcal{A}_4$ attacking $\mathcal{SE}$ in the LoR-qsATK sense, with a running time of at most $t_4 = t_3$, making at most q_e encryption and (in the

CCA case) q_d decryption queries to the oracle, and the size of the classical output of $\mu_4 = \mu_3 + 2\mu_{\Pi'}$ bits, and having advantage

$$\begin{aligned}\mathbf{Adv}_{\mathcal{SE}}^{lor-qscpa}(\mathcal{A}_4) &\geq \epsilon_3 \;, \\ \mathbf{Adv}_{\mathcal{SE}}^{lor-qscca}(\mathcal{A}_4) &\geq \epsilon_3 \;.\end{aligned}$$

Assume $\mathcal{A}_3$ is a quantum adversary attacking $\mathcal{SE}$ in the FtG-qsATK sense. We construct a new quantum adversary $\mathcal{A}_4$, using $\mathcal{A}_3$, that attacks $\mathcal{SE}$ in the LoR-qsATK sense. $\mathcal{A}_4$ runs $\mathcal{A}_3$, using its oracles to provide a simulation of $\mathcal{A}_3$'s oracles. The LoR challenger maintains the experiment.

$\mathcal{A}_4$ runs $\mathcal{A}_3$ in the find phase. $\mathcal{A}_3$ adaptively makes quantum queries of its choice. These can be either encryption or decryption queries. In the case of encryption queries, for each quantum query, $\mathcal{A}_4$ places quantum circuit descriptions of two permutations Π'_0 and Π'_1 in its classical register. We assume these permutations are identity functions. The encryption oracle $\mathsf{LoR}_{Q_{\mathcal{A}_3}}(\cdot)$ is invoked. The encryption oracle responds to each query by first executing $\mathbf{UQE}(\Pi_b, Q_{\mathcal{A}_3})$, and then applying the unitary $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to the first $(2n + n_r)$ qubits of the adversary's quantum register. At the end of the find phase, $\mathcal{A}_3$ requests encryption of a quantum query for which it also places quantum circuit descriptions of two permutations $\Pi_0, \Pi_1 : \{0,1\}^n \to \{0,1\}^n$ in its classical register. To respond, the encryption oracle executes $\mathbf{UQE}(\Pi_b, Q_{\mathcal{A}_3})$ and then applies the unitary $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to the first $(2n + n_r)$ qubits of the quantum register $Q_{\mathcal{A}_3}$. Without loss of generality, we call the results of all encryption queries made by $\mathcal{A}_3$ the challenge ciphertexts. In the case of decryption queries, the decryption oracle $\mathsf{Dec}_{Q_{\mathcal{A}_3}}()$ applies the unitary $\mathbf{U}_{\mathcal{D}_K(\cdot)}$ to the quantum register. At some point, $\mathcal{A}_3$ returns a bit d which $\mathcal{A}_4$ outputs as its guess.

Note that for any challenge ciphertext $\sum_c \lambda_c |m_c, c\rangle$, we define the projector $\mathsf{Proj}_c = \sum_x |x, c\rangle\langle x, c|$. We use ρ to denote the state of $\mathcal{A}_3$'s quantum register before making any decryption query. Then we assume that

$$\Pr\left[\exists c : \lambda_c \neq 0 \text{ and } \mathrm{Tr}\left(\mathsf{Proj}_c\, \rho\right) \neq 0\right] = 0 \;,$$

for all quantum adversaries $\mathcal{A}_3$. For either case of $b = 0$ or $b = 1$, $\mathcal{A}_4$ provides a perfect simulation for $\mathcal{A}_3$. Therefore, $\mathcal{A}_4$ succeeds with the same probability as $\mathcal{A}_3$. Hence, for $\mathcal{A}_4$'s advantage we have

$$\begin{aligned}\mathbf{Adv}_{\mathcal{SE}}^{lor-qsatk}(\mathcal{A}_4) &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{lor-qsatk-1}(\mathcal{A}_4) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{lor-qsatk-0}(\mathcal{A}_4) = 1\right]\\ &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{ftg-qsatk-1}(\mathcal{A}_3) = 1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{ftg-qsatk-0}(\mathcal{A}_3) = 1\right]\\ &= \mathbf{Adv}_{\mathcal{SE}}^{ftg-qsatk}(\mathcal{A}_3)\,. \end{aligned} \tag{6.84}$$

$\mathcal{A}_4$ runs in time at most $t_4 = t_3$, and makes at most q_e encryption and (in the CCA case) q_d decryption queries, and the size of the classical output is $\mu_4 = \mu_3 + 2\mu_{\Pi'}$ bits.

Step 3 (FtG-qsATK $\Rightarrow$ SEM-qsATK): For any scheme $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, assume that $\mathcal{A}_2$ is a quantum adversary attacking $\mathcal{SE}$ in the SEM-qsATK sense, with a running time of at most t_2, making at most q_e encryption and (in the CCA case) q_d decryption queries to the oracle, and the size of the classical output of μ_2 bits, and having advantage

$$\begin{aligned}\mathbf{Adv}_{\mathcal{SE}}^{sem-qscpa}(\mathcal{A}_2) &\geq \epsilon_2\ ,\\ \mathbf{Adv}_{\mathcal{SE}}^{sem-qscca}(\mathcal{A}_2) &\geq \epsilon_2\ .\end{aligned}$$

Then there exists a quantum adversary $\mathcal{A}_3$ attacking $\mathcal{SE}$ in the FtG-qsATK sense, with a running time t_3 of at most $t_2 + q_e c \mu_2$, making at most q_e encryption and (in the CCA case) q_d decryption queries to the oracle, and the size of the classical output of $\mu_2 = \mu_3$ bits, and having advantage

$$\begin{aligned}\mathbf{Adv}_{\mathcal{SE}}^{ftg-qscpa}(\mathcal{A}_3) &\geq \frac{\epsilon_2}{2}\ ,\\ \mathbf{Adv}_{\mathcal{SE}}^{ftg-qscca}(\mathcal{A}_3) &\geq \frac{\epsilon_2}{2}\ .\end{aligned}$$

Assume $\mathcal{A}_2$ is a quantum adversary attacking $\mathcal{SE}$ in the SEM-qsATK sense. We construct a new quantum adversary $\mathcal{A}_3$, using $\mathcal{A}_2$, that attacks $\mathcal{SE}$ in the FtG-qsATK sense. $\mathcal{A}_3$ runs $\mathcal{A}_2$, and uses its oracles to provide a simulation of $\mathcal{A}_2$'s oracles. The FtG challenger maintains the experiment.

$\mathcal{A}_3$ runs $\mathcal{A}_2$ in its select phase. $\mathcal{A}_2$ adaptively makes quantum queries of its choice. The queries can be either encryption or decryption queries. In the case of encryption queries, the encryption oracle responds to each query by applying the unitary $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to the first $(2n + n_r)$ qubits of the adversary's quantum register. At the end of the select phase, $\mathcal{A}_2$ outputs a classical description of P, a set of permutations, as well as R a quantum circuit description of a state $|\psi\rangle$. Together, R and P define a distribution $\mathcal{M}$ of quantum queries. $\mathcal{A}_3$ reads $\mathcal{A}_2$'s classical register. It then samples permutations $\Pi_0 \leftarrow\$ P$ and $\Pi_1 \leftarrow\$ P$. Also, $\mathcal{A}_3$ executes $\mathbf{UQE}\left(R, Q_{\mathcal{A}_2}\right)$ to prepare its quantum register in the state $|\psi\rangle$. $\mathcal{A}_3$ places the description of two permutations Π_0 and Π_1 in its classical register. The encryption oracle executes $\mathbf{UQE}\left(\Pi_b, Q_{\mathcal{A}_2}\right)$ and then applies the unitary $\mathbf{U}_{\mathcal{E}_K(\cdot)}$ to the first $(2n + n_r)$ bits of $\mathcal{A}_2$'s quantum register. We call the result of this type of query, the challenge ciphertexts.

In the case of decryption queries, the decryption oracle applies the unitary $\mathbf{U}_{\mathcal{D}_K(\cdot)}$. Note that for any challenge ciphertext $\sum_c \lambda_c |m_c, c\rangle$, we define the projector $\mathsf{Proj}_c = \sum_x |x, c\rangle\langle x, c|$. We use ρ to denote the state of $\mathcal{A}_2$'s quantum register before making any decryption query. Then we assume that

$$\Pr\left[\exists c : \lambda_c \neq 0 \text{ and } \mathrm{Tr}\left(\mathsf{Proj}_c\, \rho\right) \neq 0\right] = 0 \ ,$$

for all quantum adversaries $\mathcal{A}_2$. At some point, $\mathcal{A}_3$ runs $\mathcal{A}_2$ in its $\mathsf{predict}$ phase. $\mathcal{A}_2$ outputs a description of a quantum circuit V and a guess z. Then, $\mathcal{A}_3$ executes $\mathbf{UQE}\left(R, Q_{\mathcal{A}_2}\right)$ to prepare the quantum register in the state $|\psi\rangle$, and executes $\mathbf{UQE}\left(V, Q_{\mathcal{A}_2}\right)$ to obtain a value z'. If $z = z'$, $\mathcal{A}_3$ returns 0. Otherwise, it returns a random bit.

When $b = 0$ we have

$$\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qsatk}\left(\mathcal{A}_2\right)=1\right]=\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qsatk-0}\left(\mathcal{A}_2\right)=1\right]\ . \tag{6.85}$$

Then,

$$\begin{aligned}\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{ftg-qsatk-0}\left(\mathcal{A}_3\right)=1\right]&=\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qsatk-0}\left(\mathcal{A}_2\right)=1\right]\\&\quad+\frac{1}{2}\cdot\left(1-\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qsatk-0}\left(\mathcal{A}_2\right)=1\right]\right)\\&=\frac{1}{2}\cdot\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qsatk-0}\left(\mathcal{A}_2\right)=1\right]+\frac{1}{2}\ . \qquad (6.86)\end{aligned}$$

When $b=1$ we have

$$\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qsatk}\left(\mathcal{A}_2\right)=1\right]=\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qsatk-1}\left(\mathcal{A}_2\right)=1\right]\ . \tag{6.87}$$

Then,

$$\begin{aligned}\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{ftg-qsatk-1}\left(\mathcal{A}_3\right)=1\right]&=\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qsatk-1}\left(\mathcal{A}_2\right)=1\right]\\&\quad+\frac{1}{2}\cdot\left(1-\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qsatk-1}\left(\mathcal{A}_2\right)=1\right]\right)\\&=\frac{1}{2}\cdot\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qsatk-1}\left(\mathcal{A}_2\right)=1\right]+\frac{1}{2}\ . \qquad (6.88)\end{aligned}$$

Hence, from the above equations, for the advantage of $\mathcal{A}_3$ we have,

$$\begin{aligned}
\mathbf{Adv}_{\mathcal{SE}}^{ftg-qsatk}(\mathcal{A}_3) &= \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{ftg-qsatk-1}(\mathcal{A}_3)=1\right] - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{ftg-qsatk-0}(\mathcal{A}_3)=1\right] \\
&= \frac{1}{2} - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{ftg-qsatk-0}(\mathcal{A}_3)=1\right] \\
&= \frac{1}{2}\cdot\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qsatk-1}(\mathcal{A}_2)=1\right] + \frac{1}{2} \\
&\qquad\qquad - \frac{1}{2}\cdot\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qsatk-0}(\mathcal{A}_2)=1\right] + \frac{1}{2} \\
&= \frac{1}{2}\cdot\Big(\Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qsatk-1}(\mathcal{A}_2)=1\right] \\
&\qquad\qquad - \Pr\left[\mathbf{Exp}_{\mathcal{SE}}^{sem-qsatk-0}(\mathcal{A}_2)=1\right]\Big) \\
&= \frac{1}{2}\cdot\mathbf{Adv}_{\mathcal{SE}}^{sem-qsatk}(\mathcal{A}_2)\ . \qquad (6.89)
\end{aligned}$$

$\mathcal{A}_3$ runs in time at most $t_3 = t_2 + T_P + T_R$ where $T_P = q_e c \frac{\mu_P}{2}$ is the maximum time required for a permutation Π output by P, and $T_R = q_e c \mu_R$ is the maximum time required to prepare a quantum register in the state $|\psi\rangle$. Also $\mathcal{A}_3$ makes at most q_e encryption and (in the CCA case) q_d decryption oracle queries, and the size of the classical output is $\mu_3 = \mu_2$ bits.

Step 4 (RoP-qsATK ⇒ SEM-qsATK): Assume $\mathcal{A}_2$ is a quantum adversary attacking $\mathcal{SE} = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ in the SEM-qsATK sense. We construct a new quantum adversary $\mathcal{A}_1$, using $\mathcal{A}_2$, that attacks $\mathcal{SE}$ in the RoP-qsATK sense.

From the previous three steps we can see that RoP-qsATK ⇒ SEM-qsATK. For the advantage of $\mathcal{A}_2$ we have that,

$$\begin{aligned}
\mathbf{Adv}_{\mathcal{SE}}^{sem-qsatk}(\mathcal{A}_2) &\leq 2\cdot\mathbf{Adv}_{\mathcal{SE}}^{ftg-qsatk}(\mathcal{A}_3) \\
&\leq 2\cdot\mathbf{Adv}_{\mathcal{SE}}^{lor-qsatk}(\mathcal{A}_4) \\
&\leq 4\cdot\mathbf{Adv}_{\mathcal{SE}}^{rop-qsatk}(\mathcal{A}_1)\ . \qquad (6.90)
\end{aligned}$$

Since it is assumed that

$$\mathbf{Adv}_{\mathcal{SE}}^{sem-qsatk}(\mathcal{A}_2) \geq \epsilon_2\ , \qquad (6.91)$$

then we can show that

$$\mathbf{Adv}_{\mathcal{SE}}^{rop-qsatk}(\mathcal{A}_1) \geq \frac{\epsilon_2}{4} \; . \tag{6.92}$$

$\mathcal{A}_1$ runs in time at most $t_1 = t_2 + T_P + T_{\Pi'} + T_R$ where

$$\begin{aligned} t_1 &= t_2 + T_P + T_{\Pi'} + T_R \\ &\leq t_2 + q_e c\left(\frac{\mu_4}{2} + \mu_2\right) \\ &\leq t_2 + q_e c\left(\frac{\mu_2}{2} + \mu_2 + \mu_{\Pi'}\right) \\ &\leq t_2 + q_e c\left(\frac{3}{2}\mu_2 + \mu_{\Pi'}\right) \; . \end{aligned} \tag{6.93}$$

Moreover, $\mathcal{A}_1$ makes at most q_e encryption and (in the CCA case) q_d decryption oracle queries, and the size of the classical output is $\mu_1 = \mu_2 + 2\mu_{\Pi'}$. This concludes the proof. ▮

Chapter 7

Conclusion

In the case of quantum computation, we explored how existing classical confidentiality notions translate into this model. We showed that the security proofs of Counter mode carry over to the quantum computation model. This serves two goals. First, this means that existing security notions, such as LoR-CPA, are achievable in this model. And second, the proofs are a showcase of a class of classical black-box security proofs that can go through in the quantum computation model.

Our results of quantum superposition attacks show that some cryptographic schemes, while secure even against generic quantum computation attacks, might fall apart in this model. We discussed that block ciphers such as the Even-Mansour scheme offer no security in the quantum superposition model. It would be interesting to see if this were the case for other symmetric cryptosystems such as hash functions. Therefore we stress that the security of modern cryptosystems need to be reassessed in the quantum superposition model, given that one day we might use a 'quantum internet' or run our cryptosystems on quantum computers.

To be able to formally assess the security of modern cryptosystems in the quantum superposition model, meaningful notions of security are required. We discussed why the existing classical confidentiality notions need to be modified in this model. Then we defined a new notion of confidentiality, named Real-or-Permutation (RoP). We showed the implication between RoP and the existing classical security notions to prove that they are equivalent. But then we proved that the quantum analogues of RoP, such as RoP-qsCPA and RoP-qsCCA, are achievable in the quantum superposition model. Moreover, we defined a notion of semantic security (SEM) in this model, and proved that RoP implies SEM in the quantum superposition model.

These notions can serve us as tools to formally analyse the security of any cryptosystem, symmetric or asymmetric, in this model. Therefore we can have a meaningful understanding of the security of modern cryptosystems in the quantum superposition model. It is also interesting to see whether other existing classical notions of security, such as integrity, can be translated into the quantum superposition model. By having notions of confidentiality and integrity in the quantum superposition model, one can discuss how to construct quantum-secure secure channels in this model.

Bibliography

[1] Data encryption standard. Technical report, United States National Bureau of Standards, 1977.

[2] ISO/IEC9797. Data cryptographic techniques - Data integrity mechanism using a cryptographic check function employing a block cipher algorithm, 1989.

[3] Announcing the advanced encryption standard. Technical report, United States National Institute of Standards and Technology, 2001.

[4] S. Aaronson. BQP and the Polynomial Hierarchy. In *STOC '10 Proceedings of the Forty-second ACM Symposium on Theory of Computing*, pages 141–150. ACM, 2010.

[5] Dorit Aharonov. Quantum computation. *Annual Reviews of Computational Physics*, VI, 1998.

[6] A. Ambianis. Quantum search algorithms. *CoRR*, 2005.

[7] A. Barenco, D. Deutsch, A. Ekert, and R. Jozsa. Conditional Quantum Dynamics and Logic Gates. *Phys. Rev. Lett.*, 74(20):4083–4086, 1995.

[8] Adriano Barenco, Charles Bennett, Richard Cleve, David DiVincenzo, Norman Margolus, Peter Shor, Tycho Sleator, John Smolin, and Harald Weinfurter. Elementary gates for quantum computation. *Physical Review A*, 52(5):3457–3467, November 1995.

[9] R Barends, J Kelly, A Megrant, A Veitia, D Sank, E Jeffrey, T C White, J Mutus, a G Fowler, B Campbell, Y Chen, Z Chen, B Chiaro, A Dunsworth, C Neill, P O'Malley, P Roushan, A Vainsencher, J Wenner, a N Korotkov, a N Cleland, and John M Martinis. Superconducting quantum circuits at the surface code threshold for fault tolerance. *Nature*, 508:500–3, 2014.

[10] M Bellare. Practice-Oriented Provable-Security. In I. B. Damgaard, editor, *Information Security. First International Workshop, ISW'97*, volume 1561 of *Lecture Notes in Computer Science*, pages 221– 31, Berlin, Heidelberg, March 1999. Springer Berlin Heidelberg.

[11] M. Bellare. New Proofs for NMAC and HMAC: Security Without Collision-Resistance. In C. Dwork, editor, *Advances in Cryptology - CRYPTO 2006*, volume 4117, pages 602–619. Springer Berlin Heidelberg, 2006.

[12] M. Bellare, A. Desai, E. Jokipii, and P. Rogaway. A concrete security treatment of symmetric encryption. In *Proceedings 38th Annual Symposium on Foundations of Computer Science*, pages 394–403. IEEE Comput. Soc, 1997.

[13] M Bellare, A Desai, E Jokipii, and P Rogaway. A Concrete Security Treatment of Symmetric Encryption : Analysis of the DES Modes of Operation. In *The 38th Symposium on Foundations of Computer Science, IEEE*, 1997.

[14] M. Bellare, R. Guerin, and P. Rogaway. XOR MACs: New Methods for Message Authentication Using Finite Pseudorandom Functions. In Don Coppersmith, editor, *Advances in Cryptology CRYPTO 95*, volume 963 of *Lecture Notes in Computer Science*, pages 15–28, Berlin, Heidelberg, July 1995. Springer Berlin Heidelberg.

[15] M Bellare, J Kilian, and P Rogaway. The security of cipher block chaining. In Y. G. Desmedt, editor, *Advances in Cryptology - Crypto*. Springer, 1994.

[16] Mihir Bellare, Joe Kilian, and Phillip Rogaway. The Security of the Cipher Block Chaining Message Authentication Code. *Journal of Computer and System Sciences*, 399(61):362–399, 2000.

[17] Mihir Bellare and Chanathip Namprempre. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. *Advances in CryptologyASIACRYPT 2000*, pages 531–545, 2000.

[18] Mihir Bellare and Phillip Rogaway. Entity Authentication and Key Distribution. In *Advances in Cryptology - Crypto*. Springer-Verlag, 1993.

[19] Mihir Bellare and Phillip Rogaway. Provably Secure Session Key Distribution - The Three Party Case. *ACM*, 1995.

[20] C. H. Bennett. Logical Reversibility of Computation. *IBM Journal of Research and Development*, 17(6):525–532, November 1973.

[21] CH Bennett, Ethan Bernstein, Gilles Brassard, and U Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 1997.

[22] D. J. Bernstein, J. Buchmann, and E. Dahmen. *Post-Quantum Cryptography*. Springer Berlin Heidelberg, 2009.

[23] Ethan Bernstein and Umesh Vazirani. Quantum Complexity Theory. *SIAM Journal on Computing*, 26(5):1411–1473, October 1997.

[24] A. Berthiaume and G. Brassard. The quantum challenge to structural complexity theory. In *Proceedings of the Seventh Annual Structure in Complexity Theory Conference*. IEEE, 1992.

[25] Eli Biham, Yaniv Carmeli, Itai Dinur, Orr Dunkelman, Nathan Keller, and Adi Shamir. Cryptanalysis of Iterated Even-Mansour Schemes with Two Keys. *IACR Cryptology ePrint Archive*, 2013, 2013.

[26] A. Biryukov and D. Wagner. Slide Attacks. In L. Knudsen, editor, *Fast Software Encryption*, pages 245–259. Springer Berlin Heidelberg, 1999.

[27] A. Biryukov and D. Wagner. Advanced Slide Attacks. In Bart Preneel, editor, *Advances in Cryptology EUROCRYPT 2000*, volume 1807 of *Lecture Notes in Computer Science*, pages 589–606, Berlin, Heidelberg, May 2000. Springer Berlin Heidelberg.

[28] Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Francois-xavier Standaert, John Steinberger, and Elmar Tischhauser. Key-Alternating Ciphers in a Provable Setting : Encryption Using a Small Number. In D. Pointcheval and T. Johansson, editors, *Advances in Cryptology - Eurocrypt 2012*, pages 45–62. Springer, 2012.

[29] D Boneh and M Zhandry. Secure Signatures and Chosen Ciphertext Security in a Post-Quantum World. In R. Canetti and J. A. Garay, editors, *Advances in Cryptology CRYPTO 2013*, volume 8043, pages 361–379. Springer Berlin Heidelberg, 2013.

[30] Dan Boneh and M Zhandry. Quantum-Secure Message Authentication Codes. In T. Johansson and P. Q. Nguyen, editors, *Advances in Cryptology EUROCRYPT 2013*, pages 592–608. Springer Berlin Heidelberg, 2013.

[31] Michel Boyer, Gilles Brassard, and Peter Hø yer. Tight bounds on quantum searching. 1996.

[32] P. O. Boykin, T. Mor, M. Pulver, V. Roychowdhury, and F. Vatan. On Universal and Fault-Tolerant Quantum Computing. In *Proc. 40th FOCS*, pages 486–494. Society Press, 1999.

[33] G Brassard, P Hø yer, and A Tapp. Quantum cryptanalysis of hash and claw-free functions. *LATIN'98: Theoretical Informatics*, 20244(20244), 1998.

[34] Gilles Brassard, Peter Hoyer, and Alain Tapp. Quantum Algorithm for the Collision Problem, 1997.

[35] G Chen, DA Church, and BG Englert. *Quantum computing devices: principles, designs, and analysis.* 2010.

[36] S. Chen, R. Lampe, J. Lee, Y. Seurin, and J. Steinberger. Minimizing the Two-Round Even-Mansour Cipher. In Juan A. Garay and Rosario Gennaro, editors, *Advances in Cryptology CRYPTO 2014*, volume 8616 of *Lecture Notes in Computer Science*, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg.

[37] Andrew M. Childs, Edward Farhi, and John Preskill. Robustness of adiabatic quantum computation. *Phys. Rev. A*, 65(1), 2001.

[38] C. Choi. Google and NASA Launch Quantum Computing AI Lab, 2013.

[39] J. Cirac and P. Zoller. Quantum Computations with Cold Trapped Ions. *Physical Review Letters*, 74:4091–4094, 1995.

[40] Richard Cleve. An Introduction to Quantum Complexity Theory, 1999.

[41] Claude Crépeau. Quantum Oblivious Transfer. *Journal of Modern Optics*, 41(12):2445–2454, December 1994.

[42] D-Wave. D-Wave TwoTM Quantum Computer Selected for New Quantum Artificial Intelligence Initiative, System to be Installed at NASA's Ames Research Center, and Operational in Q3, 2013.

[43] Joan Daemen and Vincent Rijmen. Probability distributions of correlation and differentials in block ciphers. *Journal of Mathematical Cryptology*, 1:221–242, 2007.

[44] I. Damgaard and C. Lunemann. Quantum-secure coin-flipping and applications. In M. Matsui, editor, *Advances in Cryptology ASIACRYPT 2009*, volume 5912 LNCS, pages 52–69. Springer Berlin Heidelberg, 2009.

[45] J. P. Degabriele. *Authenticated Encryption in Theory and in Practice.* PhD thesis, Royal Holloway University of London, 2014.

[46] D. Deutsch. Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 400(1818):97–117, July 1985.

[47] D. Deutsch. Quantum Computational Networks. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 425(1868):73–90, September 1989.

[48] D. Deutsch and R. Jozsa. Rapid Solution of Problems by Quantum Computation. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 439:553–558, 1992.

[49] T. Dierks and E. Rescorla. The Transport Layer Security (TLS) Protocol, 2008.

[50] Itai Dinur, Orr Dunkelman, Nathan Keller, and Adi Shamir. Key Recovery Attacks on 3-round Even-Mansour, 8-step LED-128, and Full AES. In Kazue Sako and Palash Sarkar, editors, *Advances in Cryptology - ASIACRYPT 2013*, volume 8269 of *Lecture Notes in Computer Science*, pages 337–356, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[51] D. P. DiVincenzo. The Physical Implementation of Quantum Computation. 2000.

[52] David DiVincenzo. Two-bit gates are universal for quantum computation. *Physical Review A*, 51(2):1015–1022, February 1995.

[53] David P. DiVincenzo. Quantum Gates and Circuits. *Proceedings of The Royal Society A: Mathematical Physical and Engineering Sciences*, 1997.

[54] O. Dunkelman, N. Keller, and A. Shamir. Minimalism in Cryptography: The Even-mansour Scheme Revisited. In *Proceedings of the 31st Annual International Conference on Theory and Applications of Cryptographic Techniques*, pages 336–354, Cambridge, UK, 2012. Springer-Verlag.

[55] M. Dworkin. Recommendation for block cipher modes of operation: Methods and techniques. Technical report, NIST Special Publication 800-38A, 2001.

[56] A. Einstein, B. Podolsky, and N. Rosen. Can Quantum-Mechanical Description of Physical Reality Be Considered Complete? *Physical Review*, 47(10):777–780, May 1935.

[57] Shimon Even and Yishay Mansour. A construction of a cipher from a single pseudorandom permutation. In *Advances in Cryptology ASIACRYPT '91*, pages 210–224, 1993.

[58] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum Computation by Adiabatic Evolution. page 24, 2000.

[59] H. Feistel. Cryptography and Computer Privacy. *Scientific American*, 228:15–23, 1973.

[60] Richard P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6-7):467–488, 1982.

[61] C.A. Fuchs and J. van de Graaf. Cryptographic distinguishability measures for quantum-mechanical states. *IEEE Transactions on Information Theory*, 45(4):1216–1227, May 1999.

[62] O. Goldreich, S. Goldwasser, and S. Micali. On the Cryptographic Applications of Random Functions. In G. R. Blakley and D. Chaum, editors, *Proceedings of CRYPTO 84 on Advances in Cryptology*, pages 276–288. Springer-Verlag New York, 1985.

[63] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *Journal of the ACM*, 33:792–807, 1986.

[64] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299, April 1984.

[65] L. K. Grover. A fast quantum mechanical algorithm for database search. In *28th Annual ACM Symposium on the Theory of Computing*, page 212, 1996.

[66] H. Häffner, C. F. Roos, and R. Blatt. Quantum computing with trapped ions. *Physics Reports*, 469:155–203, 2008.

[67] Sean Hallgren, Adam Smith, and Fang Song. Classical cryptographic protocols in a quantum world. In Phillip Rogaway, editor, *Advances in Cryptology CRYPTO*, pages 411–428. Springer Berlin / Heidelberg, 2011.

[68] R. Impagliazzo and A. Wigderson. P=BPP unless E has sub-exponential circuits: Derandomizing the XOR Lemma. In *STOC '97 Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 220–229. ACM, 1997.

[69] Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing - STOC '89*, pages 44–61, New York, New York, USA, 1989. ACM Press.

[70] J. A. Jones. NMR Quantum Computation: a Critical Evaluation, 2000.

[71] B E Kane. A silicon-based nuclear spin quantum computer. *Nature*, 393:133–137, 1998.

[72] P. Kaye, R. Laflamme, and M. Mosca. *An Introduction to Quantum Computing.* Oxford University Press, first edition, 2006.

[73] S. Kent and R. Atkinson. IP Encapsulating Security Payload (ESP), 1998.

[74] J. Kilian and P. Rogaway. How to Protect DES Against Exhaustive Key Search. In N. Koblitz, editor, *Advances in Cryptology CRYPTO 1996*, volume 1109. Springer Berlin Heidelberg, 1996.

[75] Neal Koblitz and Alfred J. Menezes. Another look at "provable security". II, 2006.

[76] Neal Koblitz and Alfred J. Menezes. Another look at "provable security". *Journal of Cryptology*, 20(1):3–37, 2007.

[77] H. Kuwakado and M. Morii. Security on the quantum-type Even-Mansour cipher. In *International Symposium on Information Theory and its Applications (ISITA)*, pages 312 – 316, 2012.

[78] Rodolphe Lampe, Jacques Patarin, and Yannick Seurin. An asymptotically tight security analysis of the iterated Even-Mansour cipher. In X. Wang and K. Sako, editors, *Advances in Cryptology ASIACRYPT 2012*, volume 7658 LNCS, pages 278–295. Springer Berlin Heidelberg, 2012.

[79] M. Liskov, R. L. Rivest, and D. Wagner. Tweakable Block Ciphers. In M. Yung, editor, *Advances in Cryptology - CRYPTO 2002*, pages 31—-46. Springer, 2002.

[80] Seth Lloyd. Almost Any Quantum Logic Gate is Universal. *Physical Review Letters*, 75(2):346–349, July 1995.

[81] Daniel Loss and David P. DiVincenzo. Quantum Computation with Quantum Dots. *Phys. Rev. A*, 57(1):120–126, 1997.

[82] M. Luby and C. Rackoff. How to Construct Pseudo-random Permutations from Pseudo-random Functions. In H. C. Williams, editor, *Advances in Cryptology CRYPTO 85 Proceedings*, pages 447–447. Springer Berlin Heidelberg, 1986.

[83] M. Luby and C. Rackoff. A Study of Password Security. In C. Pomerance, editor, *Advances in Cryptology CRYPTO 87*, pages 392–397. Springer Berlin Heidelberg, 1988.

[84] Carolin Lunemann and Jesper Buus Nielsen. Fully simulatable quantum-secure coin-flipping and applications. In A. Nitaj and D. Pointcheval, editors, *Progress in Cryptology AFRICACRYPT 2011*, volume 6737 LNCS, pages 21–40. Springer Berlin Heidelberg, 2011.

[85] S. A. Lyon. Spin-based quantum computing using electrons on liquid helium. *Physical Review A - Atomic, Molecular, and Optical Physics*, 74, 2006.

[86] John J. L. Morton, Alexei M. Tyryshkin, Richard M. Brown, Shyam Shankar, Brendon W. Lovett, Arzhang Ardavan, Thomas Schenkel, Eugene E. Haller, Joel W. Ager, and S. A. Lyon. Solid state quantum memory using the 31P nuclear spin. *Nature*, 455:1085–1088, 2008.

[87] Michele Mosca. Quantum Algorithms, 2008.

[88] S. Murphy and M. J. B. Robshaw. Key-Dependent S-Boxes and Differential Cryptanalysis. *Designs, Codes and Cryptography*, 27(3):229–255, 2002.

[89] Michael A. Nielsen and Isaac L. Chuang. *Quantum computation and quantum information.* 2000.

[90] A O Niskanen, K Harrabi, F Yoshihara, Y Nakamura, S Lloyd, and J S Tsai. Quantum coherent tunable coupling of superconducting qubits. *Science*, 316:723–726, 2007.

[91] Jeremy L O'Brien. Optical quantum computing. *Science*, 318:1567–1570, 2007.

[92] M. O. Rabin. Digitalized signatures and public-key functions as intractable as factorization. Technical report, Massachusetts Institute of Technology, 1979.

[93] IBM Research. IBM Research Advances Device Performance for Quantum Computing, 2012.

[94] R. L. Rivest, A. Shamir, and L. Adleman. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[95] P. Rogaway. On the role of definitions in and beyond cryptography. In M. J. Maher, editor, *Advances in Computer Science-ASIAN 2004*, pages 13–32. Springer Berlin Heidelberg, 2004.

[96] P. Rogaway. Evaluation of Some Blockcipher Modes of Operation. Technical report, Cryptography Research and Evaluation Committees (CRYPTREC) for the Government of Japan, 2011.

[97] T. F. Ronnow, Zhihui Wang, Joshua Job, Sergio Boixo, Sergei V. Isakov, David Wecker, John M. Martinis, Daniel A. Lidar, and Matthias Troyer. Defining and detecting quantum speedup. *Science*, 345:420–424, 2014.

[98] M. Rötteler and R. Steinwandt. A note on quantum related-key attacks. *Information Processing Letters*, 115:40–44, 2014.

[99] J. J. Sakurai. *Modern Quantum Mechanics.* Addison Wesley, 1993.

[100] C. E. Shannon. Communication Theory of Secrecy Systems. *Bell System Technical Journal*, 28(4):656–715, October 1949.

[101] SW Shin, Graeme Smith, JA Smolin, and Umesh Vazirani. How "Quantum" is the D-Wave Machine?, 2014.

[102] Alexander Shnirman, Gerd Schoen, and Ziv Hermon. Quantum Manipulations of Small Josephson Junctions. *Phys. Rev. Lett.*, 79(12):2371–2374, 1997.

[103] Peter Shor. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM Review*, 41(2):303–332, 1999.

[104] Daniel R. Simon. On the Power of Quantum Computation. *SIAM Journal on Computing*, 26:116–123, 1994.

[105] A. M. Turing. On Computable Numbers, with an Application to the Entscheidungs Problem. *Proceedings of the London Mathematical Society*, s2-42(1):230–265, January 1937.

[106] Dominique Unruh. Quantum proofs of knowledge. *Advances in CryptologyEUROCRYPT 2012*, 2012.

[107] U Vazirani. On the power of quantum computation. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 356(1743):1759–1768, August 1998.

[108] U Vazirani. A survey of quantum complexity theory. In *Proceedings of Symposia in Applied Mathematics*, pages 193–220, 2002.

[109] John Watrous. Zero-Knowledge against Quantum Attacks. *SIAM Journal on Computing*, 39(1):25–58, January 2009.

[110] T. Yamamoto, A. Yu, O. Astafiev, Y. Nakamura, and J. S. Tsai. Demonstration of conditional gate operation using superconducting charge qubits. *Nature*, 425:941–944, 2003.

[111] A. Chi-Chih Yao. Quantum circuit complexity. *Proceedings of 1993 IEEE 34th Annual Foundations of Computer Science*, 1993.

[112] Andrew Chi-Chih Yao. Security of quantum protocols against coherent measurements. In *Proceedings of the twenty-seventh annual ACM symposium on Theory of computing - STOC '95*, pages 67–75, New York, New York, USA, 1995. ACM Press.

[113] Christof Zalka. Grover's quantum searching algorithm is optimal. page 13, 1997.

[114] Mark Zhandry. How to Construct Quantum Random Functions. In *IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 679–687. IEEE, October 2012.