Revocable Cryptography from Learning with Errors

Prabhanjan Ananth[*] UCSB

Alexander Poremba[†] Caltech

Vinod Vaikuntanathan[‡] MIT

Abstract

Quantum cryptography leverages many unique features of quantum information in order to construct cryptographic primitives that are oftentimes impossible classically. In this work, we build on the no-cloning principle of quantum mechanics and design cryptographic schemes with *key-revocation capabilities*. We consider schemes where secret keys are represented as quantum states with the guarantee that, once the secret key is successfully revoked from a user, they no longer have the ability to perform the same functionality as before.

We define and construct several fundamental cryptographic primitives with *key-revocation capabilities*, namely pseudorandom functions, secret-key and public-key encryption, and even fully homomorphic encryption, assuming the quantum subexponential hardness of the learning with errors problem. Central to all our constructions is our approach for making the Dual-Regev encryption scheme (Gentry, Peikert and Vaikuntanathan, STOC 2008) revocable.

[*] prabhanjan@cs.ucsb.edu

[†] aporemba@caltech.edu

[‡] vinodv@mit.edu

Contents

1 Introduction

Quantum computing presents exciting new opportunities for cryptography, using remarkable properties of quantum information to construct cryptographic primitives that are unattainable classically. At the heart of quantum cryptography lies the *no-cloning principle* [WZ82, Die82] of quantum information which stipulates that it is fundamentally impossible to copy an unknown quantum state. Indeed, Wiesner [Wie83] in his seminal work from the 1970s, used the no-cloning principle to construct a quantum money scheme, wherein quantum states are used to construct banknotes that can be verified to be authentic (using a secret key) but cannot be counterfeited. Ever since this watershed moment, and especially so in the recent years, a wide variety of primitives referred to as *unclonable* primitives have been studied and constructed in the context of encryption [Got02, BL20, BI20b, GZ20], digital signatures [LLQZ22] and pseudorandom functions [CLLZ21].

Our Work: Revocable Cryptography. Delegation and recovation of privilege are problems of great importance in cryptography. Indeed, the problem of revocation in the context of digital signatures and certificates in the classical world is a thorny problem [Stu95, Riv98]. In this work, we undertake a systematic study of *revocable (quantum) cryptography* which allows us to delegate and revoke privileges in the context of several fundamental cryptographic primitives. This continues a recent line of work in quantum cryptography dealing with revoking (or certifiably deleting) states such as quantum ciphertexts or simple quantum programs [Unr13, BI20b, GZ20, AL21, HMNY21a, Por22, BK22].

As a motivating example, consider the setting of an employee at a company who takes a vacation and wishes to authorize a colleague to perform certain tasks on her behalf, tasks that involve handling sensitive data. Since the sensitive data is (required to be) encrypted, the employee must necessarily share her decryption keys with her colleague. When she returns from vacation, she would like to have her decryption key back; naturally, one would like to ensure that her colleague should not be able to decrypt future ciphertexts (which are encrypted under the same public key) once the key is "returned". Evidently, if the decryption key is a classical object, this is impossible to achieve.

In revocable (quantum) cryptography, we associate a cryptographic functionality, such as decryption using a secret key, with a quantum state in such a way that a user can compute this functionality if and only if they are in possession of the quantum state. We then design a revocation algorithm which enables the user to certifiably return the quantum state to the owner. Security requires that once the user returns the state (via our revocation algorithm), they should not have the ability to evaluate the functionality (e.g. decrypt ciphertexts) anymore. We refer to this new security notion as *revocation security.*

Another, possibly non-obvious, application is to detecting malware attacks. Consider a malicious party who hacks into an electronic device and manages to steal a user's decryption keys. If cryptographic keys are represented by classical bits, it is inherently challenging to detect *phishing attacks* that compromise user keys. For all we know, the intruder could have stolen the user's decryption keys without leaving a trace. Indeed, a few years ago, decryption keys which were used to protect cell-phone communications [Int15] were successfully stolen by spies without being detected. With revocable cryptography, a malicious user successfully stealing a user key would invariably revoke the decryption capability from the user. This latter event can be detected.

Our Results in a Nutshell. We construct revocable cryptographic objects under standard cryptographic assumptions. Our first main result constructs a key-revocable public-key encryption scheme, and our second main result constructs a key-revocable pseudorandom function. We obtain several corollaries and extensions, including key-revocable secret-key encryption and key-revocable fully homomorphic encryption. In all these primitives, secret keys are represented as quantum states that retain the functionality of the original secret keys. We design revocation procedures and guarantee that once a user successfully passes the procedure, they cannot compute the functionality any more.

All our constructions are secure under the quantum subexponential hardness of learning with errors [Reg05]. At the heart of all of our contributions lies our result which shows that the Dual-Regev public-key encryption scheme [GPV07] satisfies revocation security.

Related Notions. There are several recent notions in quantum cryptography that are related to revocability. Of particular relevance is the stronger notion of copy-protection introduced by Aaronson [Aar09]. Breaking the revocable security of a task gives the adversary a way to make two copies of a (possibly different) state both of which are capable of computing the same functionality. Thus, uncloneability is a stronger notion. However, the only known constructions of copy-protection [CLLZ21, LLQZ22] rely on the heavy hammer of post-quantum secure indistinguishability obfuscation for which there are no known constructions based on well-studied assumptions. Our constructions, in contrast, rely on the post-quantum hardness of the standard learning with errors problem. Another related notion is the significantly weaker definition of secure software leasing [AL21] which guarantees that once the quantum state computing a functionality is returned, the *honest evaluation algorithm* cannot compute the original functionality. Yet another orthogonal notion is that of certifiably deleting *ciphertexts*, originating from the works of Unruh [Unr13] and Broadbent and Islam [BI20b]. In contrast, our goal is to delegate and revoke *cryptographic capabilities* enabled by private keys. For detailed comparisons, we refer the reader to Section 1.4.

1.1 Our Contributions in More Detail

We present our results in more detail below. First, we introduce the notion of key-revocable public-key encryption. Our main result is that dual-Regev public-key encryption scheme [GPV07] satisfies revocation security. After that, we study revocation security in the context of fully homomorphic encryption and pseudorandom functions.

Key-Revocable Public-Key Encryption. We consider public-key encryption schemes where the decryption key, modeled as a quantum state, can be delegated to a third party and can later be revoked [GZ20]. The syntax of a key-revocable public-key scheme (Definition 5.1) is as follows:

- $\mathsf{KeyGen}(1^\lambda)$: this is a setup procedure which outputs a public key PK, a master secret key MSK and a decryption key ρ_{SK}. While the master secret key is typically a classical string, the decryption key is modeled as a quantum state. (The use cases of MSK and ρ_{SK} are different, as will be evident below.)
- $\mathsf{Enc}(\mathsf{PK}, x)$: this is the regular classical encryption algorithm which outputs a ciphertext CT.
- $\mathsf{Dec}(\rho_{\mathsf{SK}}, \mathsf{CT})$: this is a quantum algorithm which takes as input the quantum decryption key ρ_{SK} and a classical ciphertext, and produces a plaintext.

- $\mathsf{Revoke}(\mathsf{PK}, \mathsf{MSK}, \sigma)$: this is the revocation procedure that outputs Valid or $\mathsf{Invalid}$. If σ equals the decryption key ρ_{SK}, then Revoke is expected to output Valid with high probability.

After the decryption key is returned, we require that the sender loses its ability to decrypt ciphertexts. This is formalized as follows (see Definition 5.3): conditioned on revocation being successful, the adversary should not be able to predict whether it is given an encryption of a message versus uniform distribution over the ciphertext space with probability better than[1] $\frac{1}{2} + \mathsf{negl}(\lambda)$. We prove the following in Theorem 6.1.

Theorem 1.1 (Informal)**.** *Assuming that the* LWE *and* SIS *problems with subexponential modulus are hard against quantum adversaries running in subexponential time (see Section 2.2), there exists a key-revocable public-key encryption scheme.*

Due to the quantum reduction from SIS to LWE [SSTX09], the two assumptions are, in some sense, equivalent. Therefore, we can in principle rely on the subexponential hardness of LWE alone.

Our work improves upon prior works, which either use post-quantum secure indistinguishability obfuscation [GZ20, CLLZ21] or consider the weaker private-key setting [KN22].

Key-Revocable Fully Homomorphic Encryption. We go beyond the traditional public-key setting and design the first *fully homomorphic encryption* (FHE) scheme [Gen09, BV14b] with key-revocation capabilities. Our construction is based on a variant of the (leveled) FHE scheme of Gentry, Sahai and Waters [GSW13], which we extend to a key-revocable encryption scheme using Gaussian superpositions. The syntax of a key-revocable FHE scheme is the same as in the key-revocable public-key setting from before (Definition 5.1), except for the additional algorithm Eval which is the same as in a regular FHE scheme. We prove the following in Theorem 7.1.

Theorem 1.2 (Informal)**.** *Assuming that the* LWE *and* SIS *problems with subexponential modulus are hard against quantum adversaries running in subexponential time (see Section 2.2), there exists a key-revocable (leveled) fully homomorphic encryption scheme.*

We prove the theorem by invoking the security of our key-revocable Dual-Regev public-key encryption scheme in Section 6.

(Key-)Revocable Pseudorandom Functions. We consider other cryptographic primitives with key-revocation capabilities that go beyond decryption functionalities; specifically, we introduce the notion of *key-revocable* pseudorandom functions (PRFs) with the following syntax:

- $\mathsf{Gen}(1^{\lambda})$: outputs a PRF key k, a quantum key ρ_k and a master secret key MSK.
- $\mathsf{PRF}(k; x)$: on key k and input x, output a value y. This is a deterministic algorithm.
- $\mathsf{Eval}(\rho_k, x)$: on input a state ρ_k and an input x, output a value y.
- $\mathsf{Revoke}(\mathsf{MSK}, \sigma)$: on input verification MSK and state σ, outputs Valid or $\mathsf{Invalid}$.

[1] The definition is intentionally formulated as a 1-bit unpredictability game; this is inspired by the notion of *uncloneable-indistinguishable security* considered by Broadbent and Lord [BL20]. Unlike the traditional cryptography literature, in this setting, 1-bit unpredictability is not equivalent to computational indistinguishability; the reason is that we also incorporate whether revocation is successful in the security experiment. Nonetheless, our construction satisfies the indistinguishability-based security notion as well.

After the quantum key ρ_k is successfully returned, we require that the sender loses its ability to evaluate the PRF. This is formalized as follows (see Definition 8.3): any efficient adversary can simultaneously pass the revocation phase and succeed in predicting the output of a pseudorandom function on a challenge input x^* versus uniform with probability at most $\frac{1}{2} + \mathsf{negl}(\lambda)$. In fact, we consider a more general definition where the adversary receives many challenge inputs instead of just one challenge input.

We give the first construction of key-revocable pseudorandom functions (PRFs) from standard assumptions. Previous schemes implicit in [CLLZ21] either require indistinguishability obfuscation, or considered weaker notions of revocable PRFs in the form of *secure software leasing* [AL21, KNY21a], which merely prevents the possiblity of *honestly* evaluating the PRF once the key is revoked.

Since in the context of pseudorandom functions, it is clear what is being revoked, we instead simply call the notion revocable pseudorandom functions.

Theorem 1.3 (Informal)**.** *Assuming that the* LWE *and* SIS *problems with subexponential modulus are hard against quantum adversaries running in subexponential time (see Section 2.2), there exist key-revocable pseudorandom functions.*

Revocable pseudorandom functions immediately give us key-revocable (many time secure) secret-key encrypton schemes.

Discussion: Unclonable Cryptography from LWE. Over the years, the existence of many fundamental cryptographic primitives such as pseudorandom functions [BPR12], fully homomorphic encryption [BV14a], attribute-based encryption [BGG$^+$14] and succinct argument systems [CJJ22] have been based on the existence of learning with errors. In fact, as far as we know, there are only a few foundational primitives remaining (indistinguishability obfuscation is one such example) whose existence is not (yet) known to be based on learning with errors.

This situation is quite different in the world of unclonable cryptography. Most of the prominent results have information-theoretic guarantees but restricted functionalities [BI20b, BL20] or are based on the existence of post-quantum indistinguishability obfuscation [Zha21, CLLZ21]. While there are works [KNY21b] that do propose lattice-based constructions of unclonable primitives, there are still many primitives, such as quantum money and quantum copy-protection, whose feasibility we would like to establish based on the existence of learning with errors. We hope that our work presents new toolkits towards building more unclonable primitives from LWE.

Independent and Concurrent Work. Independently and concurrently, Agrawal et al. [AKN$^+$23], explored the notion of public-key encryption with secure leasing which is related to key-revocable public-key encryption. Their notion as such is stronger than ours: they achieve classical revocation whereas we achieve quantum revocation. On the one hand, they achieve a generic construction based on any post-quantum secure public-key encryption whereas our notion is based on the post-quantum hardness of learning with errors. They also explore other notions of advanced encryption with secure leasing including attribute-based encryption and functional encryption, which are not explored in our work.

On the other hand, their construction of revocable public-key encryption involves many abstractions whereas our construction is based on the versatile Dual-Regev public-key encryption scheme. Additionally, we obtain key-revocable *fully homomorphic encryption* and key-revocable *pseudorandom functions* which are unique to our work.

1.2 Overview

We now give a technical overview of our constructions and their high level proof ideas. We begin with the key-revocable public-key encryption construction. A natural idea would be to start with Regev's public-key encryption scheme [Reg05] and to then upgrade the construction in order to make it revocable. However, natural attempts to associate an unclonable quantum state with the decryption key fail and thus, we instead consider the Dual-Regev public-key encryption scheme and make it key-revocable. We describe the scheme below.

Key-Revocable Dual-Regev Public-Key Encryption. Our first construction is based on the *Dual-Regev* public-key encryption scheme [GPV07] and makes use of Gaussian superpositions which serve as a quantum decryption key. We give an overview of Construction 2 below.

- $\mathsf{KeyGen}(1^n)$: sample a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ along with a *short trapdoor basis* $\mathsf{td}_{\mathbf{A}}$. To generate the decryption key, we employ the following procedure[2]: Using the matrix $\mathbf{A}$ as input, first create a Gaussian superposition of short vectors in $\mathbb{Z}^m \cap (-\frac{q}{2}, \frac{q}{2}]^m$, denoted by[3]

$$|\psi\rangle = \sum_{\mathbf{x}\in\mathbb{Z}_q^m} \rho_\sigma(\mathbf{x})\,|\mathbf{x}\rangle \otimes |\mathbf{A}\cdot\mathbf{x} \pmod q\rangle$$

  where $\rho_\sigma(\mathbf{x}) = \exp(-\pi\|\mathbf{x}\|^2/\sigma^2)$ is the Gaussian measure, for some $\sigma > 0$. Next, measure the second register which partially collapses the superposition and results in the *coset state*

$$|\psi_{\mathbf{y}}\rangle = \sum_{\substack{\mathbf{x}\in\mathbb{Z}_q^m:\\ \mathbf{A}\mathbf{x}=\mathbf{y} \pmod q}} \rho_\sigma(\mathbf{x})\,|\mathbf{x}\rangle$$

  for some outcome $\mathbf{y} \in \mathbb{Z}_q^n$. Finally, we let $|\psi_{\mathbf{y}}\rangle$ be the decryption key ρ_{SK}, $(\mathbf{A}, \mathbf{y})$ be the public key PK, and we let the trapdoor $\mathsf{td}_{\mathbf{A}}$ serve as the master secret key MSK.

- $\mathsf{Enc}(\mathsf{PK}, \mu)$: to encrypt a bit $\mu \in \{0,1\}$, sample a random string $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n$ together with discrete Gaussian errors $\mathbf{e} \in \mathbb{Z}^m$ and $e' \in \mathbb{Z}$, and output the (classical) ciphertext CT given by

$$\mathsf{CT} = (\mathbf{s}^\intercal \mathbf{A} + \mathbf{e}^\intercal, \mathbf{s}^\intercal\mathbf{y} + e' + \mu\cdot\lfloor\frac{q}{2}\rfloor) \;\in \mathbb{Z}_q^m \times \mathbb{Z}_q.$$

- $\mathsf{Dec}(\rho_{\mathsf{SK}}, \mathsf{CT})$: to decrypt a ciphertext CT using the decryption key $\rho_{\mathsf{SK}} = |\psi_{\mathbf{y}}\rangle$, first apply the unitary $U : |\mathbf{x}\rangle|0\rangle \to |\mathbf{x}\rangle|\mathsf{CT}\cdot(-\mathbf{x},1)^\intercal\rangle$ on input $|\psi_{\mathbf{y}}\rangle|0\rangle$, and then measure the second register in the computational basis. Because $|\psi_{\mathbf{y}}\rangle$ is a superposition of short vectors $\mathbf{x}$ subject to $\mathbf{A}\cdot\mathbf{x} = \mathbf{y} \pmod q$, we obtain an approximation of $\mu\cdot\lfloor\frac{q}{2}\rfloor$ from which we can recover μ.[4]

- $\mathsf{Revoke}(\mathsf{PK}, \mathsf{MSK}, \rho)$: to verify the returned state ρ given as input the public key $(\mathbf{A}, \mathbf{y})$ and master secret key $\mathsf{td}_{\mathbf{A}}$, apply the projective measurement $\{|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|, I - |\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|\}$ onto ρ. Output Valid, if the measurement succeeds, and output $\mathsf{Invalid}$, otherwise.

[2] In Section 3.2, this is formalized as the procedure $\mathsf{GenGauss}$ (see Algorithm 1).

[3] Note that the state is not normalized for convenience.

[4] For approriate choices of parameters, decryption via rounding succeeds at outputting μ with overwhelming probability and hence we can invoke the *Almost as Good as New Lemma* [Aar16] to recover the original state $|\psi_{\mathbf{y}}\rangle$.

Implementing revocation, efficiently. Note that performing a projective measurement onto a fixed Gaussian state $|\psi_{\mathbf{y}}\rangle$ is, in general, computationally infeasable. In fact, if it were to be possible to efficiently perform this projection using $(\mathbf{A}, \mathbf{y})$ alone, then one could easily use such a procedure to solve the short integer solution (SIS) problem. Fortunately, we additionally have the trapdoor for $\mathbf{A}$ at our disposal to perform the projection.

One of our contributions is to design a *quantum discrete Gaussian sampler for q-ary lattices*[5] which, given as input $(\mathbf{A}, \mathbf{y}, \mathsf{td}_{\mathbf{A}}, \sigma)$, implements a unitary that efficiently prepares the Gaussian superposition $|\psi_{\mathbf{y}}\rangle$ from scratch with access to the trapdoor $\mathsf{td}_{\mathbf{A}}$. At a high level, our Gaussian sampler can be thought of as an explicit quantum reduction from the *inhomogenous* SIS problem [Ajt96] to the search variant of the LWE problem (see Section 3.3).

Proving security: Initial challenges. Let us first discuss some high level ideas behind proving the security of the above constrution. We would like to prove that if the above scheme is insecure in the presence of a particular adversary, then we can use such an adverary to contradict some well-known computational assumption. That is, there exists an adversary who can simultaneously pass the revocation step successfully and also predict whether it receives a ciphertext or a uniform element from the ciphertext space. Towards designing such a reduction, an initial attempt would be to use the predictor, predicting an encryption of a valid message versus uniform, to break some computational assumption. Indeed, since the ciphertexts look like samples from the LWE distribution, we might be tempted to directly invoke LWE to prove this. Unfortunately, this argument is flawed! For all we know, the adversary could be doing the following: given the state $|\psi_{\mathbf{y}}\rangle$, it clones it, returns the cloned version and, then uses the original copy to distinguish encryption of a valid message versus uniform. In this case, the predictor is running the decryption algorithm honestly and thus it is not feasible to use such an adversary to break LWE.

This suggests that we may be able to argue that a computationally bounded adversary cannot possibly clone the state $|\psi_{\mathbf{y}}\rangle$. Indeed, if the adversary did succeed at cloning $|\psi_{\mathbf{y}}\rangle$, then we should be able to measure the two copies separately in order to come up with a short solution in the kernel of $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ – thereby solving the short integer solution (SIS) problem [Ajt96]. However, it is not clear if the adversary needs to clone the state in order for it to succeed. Perhaps the adversary did not clone the state after all and nevertheless succeeded at distinguishing a valid ciphertext versus uniform ciphertext. For all we know, the adversary could have been successful in breaking LWE.

Since it is not possible to detect which scenario we are in (i.e. whether the adversary successfully cloned or whether it solved the LWE problem), it is important that the reduction leverages the fact that the adversary simultaneously returns the original state and yet at the same time violates the 1-bit unpredictability experiment, in order to break some computational assumption.

Insight: Reduction to SIS. Our goal is to use the state returned by the adversary and to leverage the 1-bit prediction guarantee in order to break some computational problem. It should seem suspicious whether such a reduction is even possible: after all the adversary is returning the state we gave them! *How could this possibly help?* Our main insight lies in the following observation: while the adversary does eventually return the state we give them, the only way it can later succeed in the prediction experiment is if it retains useful information about the state. If we could somehow extract this information from the adversary, then using the extracted information alongside the returned state, we could hope to break some computational assumption. For instance, suppose we

[5] In Section 3.3, this is formalized as the procedure QSampGauss (see Algorithm 2).

can extract a short vector $\mathbf{x}$ such that $\mathbf{A} \cdot \mathbf{x} = \mathbf{y} \pmod{q}$. By measuring the state returned by the adversary, we could then hope to get a second short vector $\mathbf{x}'$ such that $\mathbf{A} \cdot \mathbf{x}' = \mathbf{y} \pmod{q}$, and from this, we can recover a short solution in the kernel of $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$.

Even if, for a moment, we disregard the issue of being able to extract $\mathbf{x}$ from the adversary, there are still some important missing steps in the above proof template:

- Firstly, measuring the returned state should give a vector different from $\mathbf{x}$ with non-negligible probability. In order to prove this, we need to argue that the squared ampltidue of every term is bounded away from 1. We prove this statement (Lemma 2.9) holds as long as $\mathbf{A}$ is full rank.

- Secondly, the reduction to SIS would only get as input $\mathbf{A}$ and not a trapdoor for $\mathbf{A}$. This means that it will no longer be possible for the reduction to actually check whether the state returned by the adversary is valid. We observe that, instead of first verifying whether the returned state is valid and then measuring in the computational basis, we can in fact skip verification and immediately go ahead and measure the state in the computational basis; this is implicit in the analysis in the proof of Lemma 6.9.

- Finally, the adversary could have entangled the returned state with its residual state in such a way that measuring the returned state always yields the same vector $\mathbf{x}$ as the one extracted from the adversary. In the same analysis in the proof of Lemma 6.9, we prove that, even if the adversary entangles its state with the returned state, with non-negligible probability we get two distinct short vectors mapping $\mathbf{A}$ to $\mathbf{y}$.

All that is left is to argue that it is possible to extract $\mathbf{x}$ from the adversary while simultaneously verifying whether the returned state is correct or not. To show that we can indeed extract another short pre-image from the adversary's quantum side information, we prove what we call a *simultaneous search-to-decision reduction with quantum auxiliary input* with respect to the Dual-Regev scheme (see Theorem 6.8). This constitutes the main technical result of this work.

Main contribution: Simultaneous search-to-decision reduction with quantum advice. Informally, our theorem says the following: any successful Dual-Regev distinguisher with access to quantum side information $\textsc{Aux}$ (which depends on the decryption key) can be converted into a successful extractor that finds a key on input $\textsc{Aux}$ – even conditioned on Revoke succeeding on a seperate register R. We now present some intuition behind our proof.

Suppose there exists a successful Dual-Regev distinguisher $\mathcal{D}$ (as part of the adversary $\mathcal{A}$) that, given quantum auxiliary information $\textsc{Aux}$, can distinguish between $(\mathbf{s}^\intercal \mathbf{A} + \mathbf{e}^\intercal, \mathbf{s}^\intercal \mathbf{y} + e')$ and uniform $(\mathbf{u}, r) \in \mathbb{Z}_q^m \times \mathbb{Z}_q$ with advantage ϵ.

Ignoring register R: For now, let us ignore the fact that Revoke is simultaneously applied on system R. Inspired by techniques from the *leakage resilience* literature [DGT+10], we now make the following observation. Letting $\mathbf{y} = \mathbf{A} \cdot \mathbf{x}_0 \pmod{q}$, for some Gaussian vector $\mathbf{x}_0$ with distribution proportional to $\rho_\sigma(\mathbf{x}_0)$, the former sample can be written as $(\mathbf{s}^\intercal \mathbf{A} + \mathbf{e}^\intercal, (\mathbf{s}^\intercal \mathbf{A} + \mathbf{e}^\intercal) \cdot \mathbf{x}_0 + e')$. Here, we assume a *noise flooding* regime in which the noise magnitude of e' is significantly larger than that of $\mathbf{e}^\intercal \cdot \mathbf{x}_0$. Because the distributions are statistically close, the distinguisher $\mathcal{D}$ must succeed at distinguishing the sample from uniform with probability negligibly close to ϵ. Finally, we invoke the LWE assumption and claim that the same distinguishing advantage persists, even if we replace $(\mathbf{s}^\intercal \mathbf{A} + \mathbf{e}^\intercal)$ with a random string $\mathbf{u} \in \mathbb{Z}_q^m$. Here, we rely on the fact that the underlying LWE sample is, in some sense, independent of the auxiliary input Aux handed to the distinguisher $\mathcal{D}$. To show that this is the case, we need to argue that the reduction can generate the appropriate inputs to $\mathcal{D}$ on input $\mathbf{A}$; in particular it should be able to generate the auxiliary input Aux (which depends on a state $|\psi_{\mathbf{y}}\rangle$), while simultaneously producing a Gaussian vector $\mathbf{x}_0$ such that $\mathbf{A} \cdot \mathbf{x}_0 = \mathbf{y} \pmod{q}$. Note that this seems to violate the SIS assumption, since the ability to produce both a superposition $|\psi_{\mathbf{y}}\rangle$ of pre-images and a single pre-image $\mathbf{x}_0$ would allow one to obtain a collision for $\mathbf{y}$.

Invoking Gaussian-collapsing: To overcome this issue, we ask the reduction to generate the quantum auxiliary input in a different way; rather than computing Aux as a function of $|\psi_{\mathbf{y}}\rangle$, we compute it as a function of $|\mathbf{x}_0\rangle$, where $\mathbf{x}_0$ results from *collapsing* the state $|\psi_{\mathbf{y}}\rangle$ via a measurement in the computational basis. By invoking the *Gaussian collapsing property* [Por22], we can show that the auxiliary information computed using $|\psi_{\mathbf{y}}\rangle$ is computationally indistinguishable from the auxiliary information computed using $|\mathbf{x}_0\rangle$. Once we invoke the collapsed version of $|\psi_{\mathbf{y}}\rangle$, we can carry out the reduction and conclude that $\mathcal{D}$ can distinguish between the samples $(\mathbf{u}, \mathbf{u}^\intercal \mathbf{x}_0)$ and $(\mathbf{u}, r)$, where $\mathbf{u}$ and r are random and $\mathbf{x}_0$ is Gaussian, with advantage negligibly close to ϵ.[6] Notice that $\mathcal{D}$ now resembles a so-called *Goldreich-Levin* distinguisher [GL89].

Reduction to Goldreich-Levin: Assuming the existence of a quantum Goldreich-Levin theorem for the field $\mathbb{Z}_q$, one could then convert $\mathcal{D}$ into an extractor that extracts $\mathbf{x}_0$ with high probability. Prior to our work, a quantum Goldreich-Levin theorem was only known for $\mathbb{Z}_2$ [AC02, CLLZ21]. In particular, it is unclear how to extend prior work towards higher order fields $\mathbb{Z}_q$ because the interference pattern in the analysis of the quantum extractor does not seem to generalize beyond the case when $q = 2$. Fortunately, we can rely on the *classical* Goldreich-Levin theorem for finite fields due to Dodis et al. [DGT+10], as well as recent work by Bitansky, Brakerski and Kalai. [BBK22] which shows that a large class of classical reductions can be generically converted into a quantum reductions. This allows us to obtain the first quantum Goldreich-Levin theorem for large fields, which we prove in Section 4. Specifically, we can show that a distinguisher $\mathcal{D}$ that, given auxiliary input Aux, can distinguish between $(\mathbf{u}, \mathbf{u}^\intercal \mathbf{x}_0)$ and $(\mathbf{u}, r)$ with advantage ε can be converted into a quantum extractor that can extract $\mathbf{x}_0$ given Aux in time $\text{poly}(1/\varepsilon, q)$ with probability negligibly close to 1. The fact that the extractor succeeds with probability negligibly close to 1 is crucial in our analysis mentioned below.

Incorporating R: To complete the security proof behind our key-revocable Dual-Regev scheme, we need to show something *stronger*; namely, we need to argue that the Goldreich-Levin extractor succeeds on input Aux – even conditioned on the fact that Revoke outputs Valid when applied on a separate register R (which may be entangled with Aux). At first sight, it might seem as though all the previous ideas are of no use since the guarantee of the Goldreich-Levin extractor only holds when we ignore the register R.

Fortunately, the Goldreich-Levin extractor succeeds with probability negligibly close to 1. Since the probability that revocation succeeds is non-negligible, this implies that the extractor has to succeed with non-negligible probability – even if we condition on revocation succeeding on register R. Using this fact, we can successfully carry out the reduction to SIS.

[6] Technically, $\mathcal{D}$ can distinguish between $(\mathbf{u}, \mathbf{u}^\intercal \mathbf{x}_0 + e')$ and $(\mathbf{u}, r)$ for a Gaussian error e'. However, by defining a distinguisher $\tilde{\mathcal{D}}$ that first shifts $\mathbf{u}$ by a Gaussian vector e' and then runs $\mathcal{D}$, we obtain the desired distinguisher.

1.3 Applications

We leverage our result of key-revocable Dual-Regev encryption to get key-revocable fully homomorphic encryption and revocable pseudorandom functions.

Key-Revocable Dual-Regev Fully Homomorphic Encryption. Our first application of our key-revocable public-key encryption concerns fully homomorphic encryption schemes. We extend our key-revocable Dual-Regev scheme towards a (leveled) FHE scheme in Construction 3 by using the DualGSW variant of the FHE scheme by Gentry, Sahai and Waters [GSW13, Mah18].

To encrypt a bit $\mu \in \{0,1\}$ with respect to the public-key $(\mathbf{A}, \mathbf{y})$, sample a matrix $\mathbf{S} \xleftarrow{\$} \mathbb{Z}_q^{n \times N}$ together with a Gaussian error matrix $\mathbf{E} \in \mathbb{Z}^{m \times N}$ and row vector $\mathbf{e} \in \mathbb{Z}^N$, and output the ciphertext

$$\mathsf{CT} = \begin{bmatrix} \mathbf{A}^\intercal \mathbf{S} + \mathbf{E} \\ \hline \mathbf{y}^\intercal \mathbf{S} + \mathbf{e} \end{bmatrix} + \mu \cdot \mathbf{G} \pmod{q} \in \mathbb{Z}_q^{(m+1) \times N}.$$

Here, $\mathbf{G}$ is the *gadget matrix* which converts a binary vector in into its field representation over $\mathbb{Z}_q$. As before, the decryption key consists of a Gaussian superposition $|\psi_{\mathbf{y}}\rangle$ of pre-images of $\mathbf{y}$.

Note that the DualGSW ciphertext can be thought of as a column-wise concatenation of N-many independent Dual-Regev ciphertexts. In Theorem 7.1, we prove the security of our construction by invoking the security of our key-revocable Dual-Regev scheme.

Revocable Pseudorandom Functions. Our next focus is on leveraging the techniques behind key-revocable public-key encryption to obtain revocable pseudorandom functions. Recall that the revocation security of pseudorandom functions stipulates the following: any efficient adversary (after successfully revoking the state that enables it to evaluate pseudorandom functions) cannot predict whether it receives pseudorandom outputs on many challenge inputs versus strings picked uniformly at random with probability better than $\frac{1}{2} + \mathsf{negl}(\lambda)$. An astute reader might notice that revocation security does not even imply the traditional pseudorandomness guarantee! Hence, we need to additionally impose the requirement that a revocable pseudorandom function should also satisfy the traditional pseudorandomness guarantee.

Towards realizing a construction satisfying our definitions, we consider the following template:

1. First show that there exists a μ-revocable pseudorandom function for $\mu = 1$. Here, μ-revocation security means the adversary receives μ-many random inputs after revocation.

2. Next, we show that any 1-revocable pseudorandom function also satisfies the stronger notion of revocation security where there is no a priori bound on the number of challenge inputs received by the adversary.

3. Finally, we show that we can generically upgrade any revocable PRF in such a way that it also satisfies the traditional pseudorandomness property.

The second bullet is proven using a hybrid argument. The third bullet is realized by combining a revocable PRF with a post-quantum secure PRF (not necessarily satisfying revocation security).

Hence, we focus the rest of our attention on proving the first bullet.

1-revocation security. We start with the following warmup construction. The secret key k comprises of matrices $\mathbf{A}, \{\mathbf{S}_{i,0}, \mathbf{S}_{i,1}\}_{i \in [\ell], b \in \{0,1\}}$, where $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$, $\mathbf{S}_{i,b} \in \mathbb{Z}_q^{n \times n}$ such that all $\mathbf{S}_{i,b}$ are sampled from some error distribution and the output of the pseudorandom function on x is denoted to be $\lfloor \sum_{i \in [\ell]} \mathbf{S}_{i,x_i} \mathbf{A} \rceil_p$, where $q \gg p$ and $\lfloor \cdot \rceil_p$ refers to a particular rounding operation modulo p.

In addition to handing out a regular PRF key k, we also need to generate a quantum key ρ_k such that, given ρ_k and any input x, we can efficiently compute $\mathsf{PRF}(k,x)$. Moreover, ρ_k can be revoked such that any efficient adversary after revocation loses the ability to evaluate the pseudorandom function. To enable the generation of ρ_k, we first modify the above construction. We generate $\mathbf{y} \in \mathbb{Z}_q^n$ and include this as part of the key. The modified pseudorandom function, on input x, outputs $\lfloor \sum_{i \in [\ell]} \mathbf{S}_{i,x_i} \mathbf{y} \rceil_p$. We denote $\sum_{i \in [\ell]} \mathbf{S}_{i,x_i}$ by $\mathbf{S}_x$ and, with this new notation, the output of the pseudorandom function can be written as $\lfloor \mathbf{S}_x \mathbf{y} \rceil_p$.

With this modified construction, we now describe the elements as part of the quantum key ρ_k:

- For every $i \in [\ell]$, include $\mathbf{S}_{i,b}\mathbf{A} + \mathbf{E}_{i,b}$ in ρ_k, where $i \in [\ell]$ and $b \in \{0,1\}$. We sample $\mathbf{S}_{i,b}$ and $\mathbf{E}_{i,b}$ from a discrete Gaussian distribution with appropriate standard deviation $\sigma > 0$.

- Include $|\psi_{\mathbf{y}}\rangle$ which, as defined in the key-revocable Dual-Regev construction, is a Gaussian superposition of short solutions mapping $\mathbf{A}$ to $\mathbf{y}$.

To evaluate on an input x using ρ_k, compute $\sum_i \mathbf{S}_{i,x_i}\mathbf{A} + \mathbf{E}_{i,x_i}$ and then using the state $|\psi_{\mathbf{y}}\rangle$, map this to $\sum_i \mathbf{S}_{i,x_i}\mathbf{y} + \mathbf{E}_{i,x_i}$. Finally, perform the rounding operation to get the desired result.

Our goal is to show that after the adversary revokes $|\psi_{\mathbf{y}}\rangle$, on input a challenge input x^* picked uniformly at random, it cannot predict whether it has received $\lfloor \sum_{i \in [N]} \mathbf{S}_{i,x_i^*} \mathbf{y} \rceil_p$ or a uniformly random vector in $\mathbb{Z}_p^n$.

Challenges in proving security: We would like to argue that when the state $|\psi_{\mathbf{y}}\rangle$ is revoked, the adversary loses its ability to evaluate the pseudorandom function. Unfortunately, this is not completely true. For all we know, the adversary could have computed the pseudorandom function on many inputs of its choice before the revocation phase and it could leverage this to break the security after revocation. For instance, suppose say the input is of length $O(\log \lambda)$ then in this case, the adversary could evaluate the pseudorandom function on all possible inputs before revocation. After revocation, on any challenge input x^*, the adversary can then successfully predict whether it receives a pseudorandom output or a uniformly chosen random output. Indeed, a pseudorandom function with $O(\log \lambda)$-length input is learnable and hence, there should be no hope of proving it to be key-revocable. This suggests that, at the very least, we need to explicitly incorporate the fact that x^* is of length $\omega(\log \lambda)$, and more importantly, should have enough entropy, in order to prove security.

Our insight: Our insight is to reduce the security of revocable pseudorandom function to the security of key-revocable Dual-Regev public-key encryption. At a high level, our goal is to set up the parameters in such a way that the following holds:

- $(\mathbf{A}, \mathbf{y})$, defined above, is set to be the public key corresponding to the Dual-Regev public-key encryption scheme,

- $|\psi_{\mathbf{y}}\rangle$, which is part of the pseudorandom function key, is set to be the decryption key of the Dual Regev scheme,

- Suppose that the challenge ciphertext, denoted by CT^*, comprises of two parts: $\mathsf{CT}_1^* \in \mathbb{Z}_q^{n \times m}$ and $\mathsf{CT}_2^* \in \mathbb{Z}_q^n$. Note that if $\mathsf{CT}_1^* \approx \mathbf{s}^\intercal \mathbf{A}$ and $\mathsf{CT}_2^* \approx \mathbf{s}^\intercal \mathbf{y}$, for some LWE secret vector $\mathbf{s}$, then

CT_1^* can be mapped onto CT_2^* using the state $|\psi_{\mathbf{y}}\rangle$. We use CT_1^* to set the challenge input x^* in such a way that CT_2^* is the output of the pseudorandom function on x^*. This implicitly resolves the entropy issue we discussed above; by the semantic security of Dual-Regev, there should be enough entropy in CT_1^* which translates to the entropy of x^*.

It turns our goal is quite ambitious: in particular, it is unclear how to set up the parameters in such that the output of the pseudorandom function on x is exactly CT_2^*. Fortunately, this will not be a deterrant, we can set up the parameters such that the output is $\approx \mathsf{CT}_2^* + \mathbf{u}$, where $\mathbf{u}$ is a vector that is known to reduction.

Once we set up the parameters, we can then reduce the security of revocable pseudorandom functions to revocable Dual Regev.

Implementation details: So far we established the proof template should work but the implementation details of the proof need to be fleshed out. Firstly, we set up the parameters in such a way that $\ell = nm\lceil \log q \rceil$. This means that there is a bijective function mapping $[n] \times [m] \times [\lceil \log q \rceil]$ to $[\ell]$. As a result, the quantum key ρ_k can be alternately viewed as follows:

- For every $i \in [n], j \in [m], \tau \in [\lceil \log q \rceil], b \in \{0,1\}$, include $\mathbf{S}_b^{(i,j,\tau)}\mathbf{A} + \mathbf{E}_b^{(i,j,\tau)}$ in ρ_k. We sample $\mathbf{S}_b^{(i,j,\tau)}$ and $\mathbf{E}_b^{(i,j,\tau)}$ from a discrete Gaussian with appropriate standard deviation $\sigma > 0$.

The output of the pseudorandom function on input x can now be interpreted as

$$\mathsf{PRF}(k,x) = \left\lceil \sum_{\substack{i \in [n], j \in [m] \\ \tau \in [\lceil \log q \rceil]}} \mathbf{S}_{x_i}^{(i,j,\tau)} \mathbf{y} \right\rceil_p$$

Next, we modify ρ_k as follows: instead of generating, $\mathbf{S}_b^{(i,j,\tau)}\mathbf{A} + \mathbf{E}_b^{(i,j,\tau)}$, we instead generate $\mathbf{S}_b^{(i,j,\tau)}\mathbf{A} + \mathbf{E}_b^{(i,j,\tau)} + \mathsf{M}_b^{(i,j,k)}$, for any set of matrices $\{\mathsf{M}_b^{(i,j,\tau)}\}$. The change should be undetectable to a computationally bounded adversary, thanks to the quantum hardness of learning with errors. In the security proof, we set up the challenge input x^* in such a way that summing up the matrices $\mathsf{M}_{x_i^*}^{(i,j,\tau)}$ corresponds to CT_1^*, where CT_1^* is part of the key-revocable Dual-Regev challenge ciphertext as discussed above. With this modification, when ρ_k is evaluated on x^*, we get an output that is close to $\mathsf{CT}_2^* + \mathbf{u}$, where $\mathbf{u} \approx \sum_{i \in [n], j \in [m], \tau \in [\lceil \log(q) \rceil]} \mathbf{y}$ is known to the reduction (discussed above) – thereby violating the security of key-revocable Dual-Regev scheme.

1.4 Related Work

Copy-Protection. Of particular relevance to our work is the foundational notion of copy-protection introduced by Aaronson [Aar09]. Informally speaking, a copy-protection scheme is a compiler that transforms programs into quantum states in such a way that using the resulting states, one can run the original program. Yet, the security guarantee stipulates that any adversary given one copy of the state cannot produce a bipartite state wherein both parts compute the original program.

While copy-protection is known to be impossible for arbitrary unlearnable functions [AL21, AK22], identifying interesting functionalities for which copy-protection is feasible has been an active research direction [CMP20, AKL+22, AKL23]. Of particular significance is the problem of copy-protecting cryptographic functionalities, such as decryption and signing functionalities. Coladangelo et al. [CLLZ21] took the first step in this direction and showed that it is feasible to copy-protect decryption functionalities and pseudorandom functions assuming the existence of post-quantum indistinguishability obfuscation. While a very significant first step, the assumption of post-quantum iO is unsatisfactory: there have been numerous post-quantum iO candidate proposals [BMSZ16, CVW18, BDGM20, DQV$^+$21, GP21, WW21], but not one of them have been based on well-studied assumptions[7].

Our work can be viewed as copy-protecting cryptographic functionalities based on learning with errors under a weaker yet meaningful security guarantee.

Secure Software Leasing. Another primitive relavent to revocable cryptography is secure software leasing [AL21]. The notion of secure software leasing states that any program can be compiled into a functionally equivalent program, represented as a quantum state, in such a way that once the compiled program is returned back[8], the honest evaluation algorithm on the residual state cannot compute the original functionality. Key-revocable encryption can be viewed as secure software leasing for decryption algorithms. However, unlike secure software leasing, key-revocable encryption satisfies a much stronger security guarantee, where there is no restriction on the adversary to run honestly after returning back the software. Secure leasing for different functionalities, namely, point functions [CMP20, BJL$^+$21], evasive functions [AL21, KNY21b] and pseudorandom functions [ALL$^+$21] have been studied by recent works.

Encryption Schemes with Revocable Ciphertexts. Unruh [Unr13] proposed a (private-key) quantum timed-release encryption scheme that is *revocable*, i.e. it allows a user to *return* the ciphertext of a quantum timed-release encryption scheme, thereby losing all access to the data. Unruh's scheme uses conjugate coding [Wie83, BB84] and relies on the *monogamy of entanglement* in order to guarantee that revocation necessarily erases information about the plaintext. Broadbent and Islam [BI20b] introduced the notion of *certified deletion*[9] and constructed a private-key quantum encryption scheme with the aforementioned feature which is inspired by the quantum key distribution protocol [BB84, TL17]. In contrast with Unruh's [Unr13] notion of revocable quantum ciphertexts which are eventually returned and verified, Broadbent and Islam [BI20b] consider certificates which are entirely classical. Moreover, the security definition requires that, once the certificate is successfully verified, the plaintext remains hidden even if the secret key is later revealed.

Using a hybrid encryption scheme, Hiroka, Morimae, Nishimaki and Yamakawa [HMNY21b] extended the scheme in [BI20a] to both public-key and attribute-based encryption with certified deletion via *receiver non-committing* encryption [JL00, CFGN96]. As a complementary result, the authors also gave a public-key encryption scheme with certified deletion which is *publicly verifiable* assuming the existence of one-shot signatures and extractable witness encryption. Using *Gaussian superpositions*, Poremba [Por22] proposed *Dual-Regev*-based public-key and fully homomorphic encryption schemes with certified deletion which are publicly verifiable and proven secure assuming a *strong Gaussian-collapsing conjecture* — a strengthening of the collapsing property of the Ajtai hash. Bartusek and Khurana [BK22] revisited the notion of certified deletion and presented a unified

[7] We remark that, there do exist post-quantum-insecure iO schemes based on well-founded assumptions [JLS21].

[8] Acording to the terminology of [AL21], this refers to finite term secure software leasing.

[9] This notion is incomparable with another related notion called unclonable encryption [BL20, AK21, AKL$^+$22], which informally guarantees that it should be infeasible to clone quantum ciphertexts without losing information about the encrypted message.

approach for how to generically convert any public-key, attribute-based, fully-homomorphic, timed-release or witness encryption scheme into an equivalent quantum encryption scheme with certified deletion. In particular, they considered a stronger notion called *certified everlasting security* which allows the adversary to be computationally unbounded once a valid deletion certificate is submitted.

Acknowledgements

P.A. thanks Fatih Kaleoglu for several insightful discussions.

This work was done (in part) while the authors were visiting the Simons Institute for the Theory of Computing. P.A. is supported by a research gift from Cisco. A.P. is partially supported by AFOSR YIP (award number FA9550-16-1-0495), the Institute for Quantum Information and Matter (an NSF Physics Frontiers Center; NSF Grant PHY-1733907) and by a grant from the Simons Foundation (828076, TV).

2 Preliminaries

Let $\lambda \in \mathbb{N}$ denote the security parameter throughout this work. We assume that the reader is familiar with the fundamental cryptographic concepts.

2.1 Quantum Computing

For a comprehensive background on quantum computation, we refer to [NC11, Wil13]. We denote a finite-dimensional complex Hilbert space by $\mathcal{H}$, and we use subscripts to distinguish between different systems (or registers). For example, we let $\mathcal{H}_A$ be the Hilbert space corresponding to a system A. The tensor product of two Hilbert spaces $\mathcal{H}_A$ and $\mathcal{H}_B$ is another Hilbert space denoted by $\mathcal{H}_{AB} = \mathcal{H}_A \otimes \mathcal{H}_B$. Let $\mathcal{L}(\mathcal{H})$ denote the set of linear operators over $\mathcal{H}$. A quantum system over the 2-dimensional Hilbert space $\mathcal{H} = \mathbb{C}^2$ is called a *qubit*. For $n \in \mathbb{N}$, we refer to quantum registers over the Hilbert space $\mathcal{H} = \left(\mathbb{C}^2\right)^{\otimes n}$ as n-qubit states. More generally, we associate *qudits* of dimension $d \geq 2$ with a d-dimensional Hilbert space $\mathcal{H} = \mathbb{C}^d$. For brevity, we write $\mathcal{H}_d^n = \mathcal{H}_d^{\otimes n}$, where $\mathcal{H}_d$ is d-dimensional. We use the word *quantum state* to refer to both pure states (unit vectors $|\psi\rangle \in \mathcal{H}$) and density matrices $\rho \in \mathcal{D}(\mathcal{H})$, where we use the notation $\mathcal{D}(\mathcal{H})$ to refer to the space of positive semidefinite matrices of unit trace acting on $\mathcal{H}$. Occasionally, we consider *subnormalized states*, i.e. states in the space of positive semidefinite operators over $\mathcal{H}$ with trace norm not exceeding 1.

The *trace distance* of two density matrices $\rho, \sigma \in \mathcal{D}(\mathcal{H})$ is given by

$$\mathrm{TD}(\rho, \sigma) = \frac{1}{2}\mathrm{Tr}\left[\sqrt{(\rho - \sigma)^\dagger(\rho - \sigma)}\right].$$

Let $q \geq 2$ be a modulus and $n \in \mathbb{N}$ and let $\omega_q = e^{\frac{2\pi i}{q}} \in \mathbb{C}$ denote the primitive q-th root of unity. The n-qudit *q-ary quantum Fourier transform* over the ring $\mathbb{Z}_q^n$ is defined by the operation,

$$\mathsf{FT}_q: \quad |\mathbf{x}\rangle \quad \mapsto \quad \sqrt{q^{-n}} \sum_{\mathbf{y} \in \mathbb{Z}_q^n} \omega_q^{\langle \mathbf{x}, \mathbf{y}\rangle} |\mathbf{y}\rangle, \qquad \forall \mathbf{x} \in \mathbb{Z}_q^n.$$

The q-ary quantum Fourier transform is *unitary* and can be efficiently performed on a quantum computer for any modulus $q \geq 2$ [HH00].

A quantum channel $\Phi : \mathcal{L}(\mathcal{H}_A) \to \mathcal{L}(\mathcal{H}_B)$ is a linear map between linear operators over the Hilbert spaces $\mathcal{H}_A$ and $\mathcal{H}_B$. Oftentimes, we use the compact notation $\Phi_{A\to B}$ to denote a quantum channel between $\mathcal{L}(\mathcal{H}_A)$ and $\mathcal{L}(\mathcal{H}_B)$. We say that a channel Φ is *completely positive* if, for a reference system R of arbitrary size, the induced map $I_R \otimes \Phi$ is positive, and we call it *trace-preserving* if $\mathrm{Tr}[\Phi(X)] = \mathrm{Tr}[X]$, for all $X \in \mathcal{L}(\mathcal{H})$. A quantum channel that is both completely positive and trace-preserving is called a quantum CPTP channel.

A polynomial-time *uniform* quantum algorithm (or QPT algorithm) is a polynomial-time family of quantum circuits given by $\mathcal{C} = \{C_\lambda\}_{\lambda\in\mathbb{N}}$, where each circuit $C \in \mathcal{C}$ is described by a sequence of unitary gates and measurements; moreover, for each $\lambda \in \mathbb{N}$, there exists a deterministic polynomial-time Turing machine that, on input 1^λ, outputs a circuit description of C_λ. Similarly, we also define (classical) probabilistic polynomial-time (PPT) algorithms. A quantum algorithm may, in general, receive (mixed) quantum states as inputs and produce (mixed) quantum states as outputs. Occasionally, we restrict QPT algorithms implicitly; for example, if we write $\Pr[\mathcal{A}(1^\lambda) = 1]$ for a QPT algorithm $\mathcal{A}$, it is implicit that $\mathcal{A}$ is a QPT algorithm that outputs a single classical bit.

A polynomial-time *non-uniform* quantum algorithm is a family $\{(C_\lambda, \nu_\lambda)\}_{\lambda\in\mathbb{N}}$, where $\{C_\lambda\}_{\lambda\in\mathbb{N}}$ is a polynomial-size (not necessarily uniformly generated) family of circuits where, for each $\lambda \in \mathbb{N}$, a subset of input qubits to C_λ consists of a polynomial-size auxiliary density matrix ν_λ. We use the following notion of indistinguishability of quantum states in the presence of auxiliary inputs.

Definition 2.1 (Indistinguishability of ensembles of quantum states, [Wat05])**.** *Let $p : \mathbb{N} \to \mathbb{N}$ be a polynomially bounded function, and let ρ_λ and σ_λ be $p(\lambda)$-qubit quantum states. We say that $\{\rho_\lambda\}_{\lambda\in\mathbb{N}}$ and $\{\sigma_\lambda\}_{\lambda\in\mathbb{N}}$ are quantum computationally indistinguishable ensembles of quantum states, denoted by $\rho_\lambda \approx_c \sigma_\lambda$, if, for any* QPT *distinguisher $\mathcal{D}$ with single-bit output, any polynomially bounded $q : \mathbb{N} \to \mathbb{N}$, any family of $q(\lambda)$-qubit auxiliary states $\{\nu_\lambda\}_{\lambda\in\mathbb{N}}$, and every $\lambda \in \mathbb{N}$,*

$$\big|\Pr[\mathcal{D}(\rho_\lambda \otimes \nu_\lambda) = 1] - \Pr[\mathcal{D}(\sigma_\lambda \otimes \nu_\lambda) = 1]\big| \leq \mathsf{negl}(\lambda)\,.$$

Lemma 2.2 ("Almost As Good As New" Lemma, [Aar16])**.** *Let $\rho \in \mathcal{D}(\mathcal{H})$ be a density matrix over a Hilbert space $\mathcal{H}$. Let U be an arbitrary unitary and let $(\mathbf{\Pi}_0, \mathbf{\Pi}_1 = \mathbf{I} - \mathbf{\Pi}_0)$ be projectors acting on $\mathcal{H}\otimes\mathcal{H}_{\mathsf{aux}}$. We interpret $(U, \mathbf{\Pi}_0, \mathbf{\Pi}_1)$ as a measurement performed by appending an ancillary system in the state $|0\rangle\langle 0|_{\mathsf{aux}}$, applying the unitary U and subsequently performing the two-outcome measurement $\{\mathbf{\Pi}_0, \mathbf{\Pi}_1\}$ on the larger system. Suppose that the outcome corresponding to $\mathbf{\Pi}_0$ occurs with probability $1 - \varepsilon$, for some $\varepsilon \in [0, 1]$. In other words, it holds that $\mathrm{Tr}[\mathbf{\Pi}_0(U\rho \otimes |0\rangle\langle 0|_{\mathsf{aux}} U^\dagger)] = 1 - \varepsilon$. Then,*

$$\mathrm{TD}(\rho, \widetilde{\rho}) \leq \sqrt{\varepsilon},$$

where $\widetilde{\rho}$ is the state after performing the measurement and applying $U^\dagger$, and after tracing out $\mathcal{H}_{\mathsf{aux}}$:

$$\widetilde{\rho} = \mathrm{Tr}_{\mathsf{aux}}\left[U^\dagger\left(\mathbf{\Pi}_0 U(\rho \otimes |0\rangle\langle 0|_{\mathsf{aux}})U^\dagger\mathbf{\Pi}_0 + \mathbf{\Pi}_1 U(\rho \otimes |0\rangle\langle 0|_{\mathsf{aux}})U^\dagger\mathbf{\Pi}_1\right)U\right].$$

Lemma 2.3 (Quantum Union Bound, [Gao15])**.** *Let $\rho \in \mathcal{D}(\mathcal{H})$ be a state and let $\mathbf{\Pi}_1, \ldots, \mathbf{\Pi}_n \geq 0$ be sequence of (orthogonal) projections acting on $\mathcal{H}$. Suppose that, for every $i \in [n]$, it holds that $\mathrm{Tr}[\mathbf{\Pi}_i\rho] = 1 - \varepsilon_i$, for $\varepsilon_i \in [0, 1]$. Then, if we sequentially measure ρ with projective measurements $\{\mathbf{\Pi}_1, \mathbf{I} - \mathbf{\Pi}_1\}, \ldots, \{\mathbf{\Pi}_n, \mathbf{I} - \mathbf{\Pi}_n\}$, the probability that all measurements succeed is at least*

$$\mathrm{Tr}[\mathbf{\Pi}_n \cdots \mathbf{\Pi}_1\rho\mathbf{\Pi}_1 \cdots \mathbf{\Pi}_n] \geq 1 - 4\sum_{i=1}^{n}\varepsilon_i.$$

2.2 Lattices and Cryptography

Let $n, m, p, q \in \mathbb{N}$ be positive integers. The rounding operation for $q \geq p \geq 2$ is the function

$$\lfloor \cdot \rceil_p \; : \; \mathbb{Z}_q \to \mathbb{Z}_p \; : \; x \mapsto \lfloor (p/q) \cdot x \rceil \pmod{p}.$$

A *lattice* $\Lambda \subset \mathbb{R}^m$ is a discrete subgroup of $\mathbb{R}^m$. Given a lattice $\Lambda \subset \mathbb{R}^m$ and a vector $\mathbf{t} \in \mathbb{R}^m$, we define the coset with respect to vector $\mathbf{t}$ as the lattice shift $\Lambda - \mathbf{t} = \{\mathbf{x} \in \mathbb{R}^m : \mathbf{x} + \mathbf{t} \in \Lambda\}$. Note that many different shifts $\mathbf{t}$ can define the same coset. The *dual* of a lattice $\Lambda \subset \mathbb{R}^m$, denoted by Λ^*, is the lattice of all $y \in \mathbb{R}^m$ that satisfy $\langle \mathbf{y}, \mathbf{x} \rangle \in \mathbb{Z}$, for every $\mathbf{x} \in \Lambda$. In other words, we let

$$\Lambda^* = \{\mathbf{y} \in \mathbb{R}^m \; : \; \langle \mathbf{y}, \mathbf{x} \rangle \in \mathbb{Z}, \text{ for all } \mathbf{x} \in \Lambda\}.$$

In this work, we mainly consider *q-ary lattices* Λ that that satisfy $q\mathbb{Z}^m \subseteq \Lambda \subseteq \mathbb{Z}^m$, for some integer modulus $q \geq 2$. Specifically, we consider the lattice generated by a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ for some $n, m \in \mathbb{N}$ that consists of all vectors which are perpendicular to the rows of $\mathbf{A}$, namely

$$\Lambda_q^{\perp}(\mathbf{A}) = \{\mathbf{x} \in \mathbb{Z}^m : \; \mathbf{A} \cdot \mathbf{x} = \mathbf{0} \pmod{q}\}.$$

For any *syndrome* $\mathbf{y} \in \mathbb{Z}_q^n$ in the column span of $\mathbf{A}$, we also consider the coset $\Lambda_q^{\mathbf{y}}(\mathbf{A})$ given by

$$\Lambda_q^{\mathbf{y}}(\mathbf{A}) = \{\mathbf{x} \in \mathbb{Z}^m : \; \mathbf{A} \cdot \mathbf{x} = \mathbf{y} \pmod{q}\} = \Lambda_q^{\perp}(\mathbf{A}) + \mathbf{c},$$

where $\mathbf{c} \in \mathbb{Z}^m$ is an arbitrary integer solution to the equation $\mathbf{A}\mathbf{c} = \mathbf{y} \pmod{q}$.

We use the following facts due to Gentry, Peikert and Vaikuntanathan [GPV07].

Lemma 2.4 ([GPV07], Lemma 5.1)**.** *Let $n \in \mathbb{N}$ and let $q \geq 2$ be a prime modulus with $m \geq 2n \log q$. Then, for all but a q^{-n} fraction of $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, the subset-sums of the columns of $\mathbf{A}$ generate $\mathbb{Z}_q^n$. In other words, a uniformly random matrix $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$ is full-rank with overwhelming probability.*

Gaussian Distribution. The *Gaussian measure* ρ_σ with parameter $\sigma > 0$ is defined as

$$\rho_\sigma(\mathbf{x}) = \exp(-\pi \|\mathbf{x}\|^2 / \sigma^2), \qquad \forall \mathbf{x} \in \mathbb{R}^m.$$

Let $\Lambda \subset \mathbb{R}^m$ be a lattice and let $\mathbf{t} \in \mathbb{R}^m$. We define the *Gaussian mass* of $\Lambda - \mathbf{t}$ as the quantity

$$\rho_\sigma(\Lambda - \mathbf{t}) = \sum_{\mathbf{y} \in \Lambda} \rho_\sigma(\mathbf{y} - \mathbf{t}).$$

The *discrete Gaussian distribution* $D_{\Lambda - \mathbf{t}, \sigma}$ assigns probability proportional to $e^{-\pi \|\mathbf{x} - \mathbf{t}\|^2 / \sigma^2}$ to every lattice point $\mathbf{x} \in \Lambda$. In other words, we have

$$D_{\Lambda - \mathbf{t}, \sigma}(\mathbf{x}) = \frac{\rho_\sigma(\mathbf{x} - \mathbf{t})}{\rho_\sigma(\Lambda - \mathbf{t})}, \qquad \forall \mathbf{x} \in \Lambda.$$

The following lemma follows from [PR06, Lemma 2.11] and [GPV07, Lemma 5.3].

Lemma 2.5. *Let $n \in \mathbb{N}$ and let q be a prime with $m \geq 2n \log q$. Let $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ be a matrix whose columns generate $\mathbb{Z}_q^n$. Let $\mathbf{y} \in \mathbb{Z}_q^n$ be arbitrary. Then, for any $\sigma \geq \omega(\sqrt{\log m})$, there exists a negligible function $\varepsilon(m)$ such that*

$$D_{\Lambda_q^{\mathbf{y}}(\mathbf{A}), \sigma}(\mathbf{x}) \; \leq \; 2^{-m} \cdot \frac{1 + \varepsilon}{1 - \varepsilon}, \qquad \forall \, \mathbf{x} \in \Lambda_q^{\perp}(\mathbf{A}).$$

Let $\mathcal{B}^m(\mathbf{0},r) = \{\mathbf{x} \in \mathbb{R}^m : \|\mathbf{x}\| \leq r\}$ denote the m-dimensional ball of radius $r > 0$. We use of the following tail bound for the Gaussian mass of a lattice [Ban93, Lemma 1.5 (ii)].

Lemma 2.6. *For any m-dimensional lattice Λ, shift $\mathbf{t} \in \mathbb{R}^m$, $\sigma > 0$ and $c \geq (2\pi)^{-\frac{1}{2}}$ it holds that*

$$\rho_\sigma\left((\Lambda - \mathbf{t}) \setminus \mathcal{B}^m(\mathbf{0}, c\sqrt{m}\sigma)\right) \leq (2\pi e c^2)^{\frac{m}{2}} e^{-\pi c^2 m} \rho_\sigma(\Lambda).$$

In addition, we also make use of the following tail bound for the discrete Gaussian which follows from [MR04, Lemma 4.4] and [GPV07, Lemma 5.3].

Lemma 2.7. *Let $n \in \mathbb{N}$ and let q be a prime with $m \geq 2n \log q$. Let $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ be a matrix whose columns generate $\mathbb{Z}_q^n$. Let $\mathbf{y} \in \mathbb{Z}_q^n$ be arbitrary. Then, for any $\sigma \geq \omega(\sqrt{\log m})$, there exists a negligible function $\varepsilon(m)$ such that*

$$\Pr_{\mathbf{x} \sim D_{\Lambda_q^{\mathbf{y}}(\mathbf{A}),\sigma}}\left[\|\mathbf{x}\| > \sigma\sqrt{m}\right] \leq 2^{-m} \cdot \frac{1+\varepsilon}{1-\varepsilon}.$$

Given a modulus $q \in \mathbb{N}$ and $\sigma \in (0, q/2\sqrt{m})$, the *truncated* discrete Gaussian distribution $D_{\mathbb{Z}_q^m,\sigma}$ over $\mathbb{Z}^m \cap (-\frac{q}{2}, \frac{q}{2}]^m$ with support $\{\mathbf{x} \in \mathbb{Z}_q^m : \|\mathbf{x}\| \leq \sigma\sqrt{m}\}$ is the density function defined below:

$$D_{\mathbb{Z}_q^m,\sigma}(\mathbf{x}) = \frac{\rho_\sigma(\mathbf{x})}{\displaystyle\sum_{\mathbf{z} \in \mathbb{Z}_q^m, \|\mathbf{z}\| \leq \sigma\sqrt{m}} \rho_\sigma(\mathbf{z})}.$$

Finally, we recall the following *noise smudging* property.

Lemma 2.8 (Noise smudging, [DGT+10])**.** *Let $y, \sigma > 0$. Then, the statistical distance between the distribution $D_{\mathbb{Z},\sigma}$ and $D_{\mathbb{Z},\sigma+y}$ is at most y/σ.*

We use the following technical lemma on the min-entropy of the truncated discrete Gaussian distribution, which we prove below.

Lemma 2.9. *Let $n \in \mathbb{N}$ and let q be a prime with $m \geq 2n \log q$. Let $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ be a matrix whose columns generate $\mathbb{Z}_q^n$. Then, for any $\sigma \geq \omega(\sqrt{\log m})$, there exists a negligible $\varepsilon(m)$ such that*

$$\max_{\mathbf{y} \in \mathbb{Z}_q^n} \max_{\substack{\mathbf{x} \in \mathbb{Z}_q^m,\, \|\mathbf{x}\| \leq \sigma\sqrt{m} \\ \mathbf{A}\mathbf{x} = \mathbf{y} \pmod q}} \left\{ \frac{\rho_\sigma(\mathbf{x})}{\displaystyle\sum_{\substack{\mathbf{z} \in \mathbb{Z}_q^m, \|\mathbf{z}\| \leq \sigma\sqrt{m} \\ \mathbf{A}\mathbf{z} = \mathbf{y} \pmod q}} \rho_\sigma(\mathbf{z})} \right\} \leq 2^{-m+1} \cdot \frac{1+\varepsilon}{1-\varepsilon}.$$

Proof. Suppose that $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ is a matrix whose columns generate $\mathbb{Z}_q^n$, i.e. $\mathbf{A}$ is full-rank. Then,

$$\begin{aligned}
&\max_{\mathbf{y}\in\mathbb{Z}_q^n} \max_{\substack{\mathbf{x}\in\mathbb{Z}_q^m,\, \|\mathbf{x}\|\le\sigma\sqrt{m} \\ \mathbf{Ax}=\mathbf{y} \pmod q}} \left\{ \frac{\rho_\sigma(\mathbf{x})}{\sum\limits_{\substack{\mathbf{z}\in\mathbb{Z}_q^m, \|\mathbf{z}\|\le\sigma\sqrt{m} \\ \mathbf{Az}=\mathbf{y} \pmod q}} \rho_\sigma(\mathbf{z})} \right\} \\
\le\ & \max_{\mathbf{y}\in\mathbb{Z}_q^n} \sup_{\mathbf{x}\in\Lambda_q^\perp(\mathbf{A})} D_{\Lambda_q^{\mathbf{y}}(\mathbf{A}),\sigma}(\mathbf{x}) \\
& + \max_{\mathbf{y}\in\mathbb{Z}_q^n} \max_{\substack{\mathbf{x}\in\mathbb{Z}_q^m,\, \|\mathbf{x}\|\le\sigma\sqrt{m} \\ \mathbf{Ax}=\mathbf{y} \pmod q}} \left| \frac{\rho_\sigma(\mathbf{x})}{\sum\limits_{\substack{\mathbf{z}\in\mathbb{Z}_q^m, \|\mathbf{z}\|\le\sigma\sqrt{m} \\ \mathbf{Az}=\mathbf{y} \pmod q}} \rho_\sigma(\mathbf{z})} - \frac{\rho_\sigma(\mathbf{x})}{\sum\limits_{\substack{\mathbf{z}\in\mathbb{Z}^m \\ \mathbf{Az}=\mathbf{y} \pmod q}} \rho_\sigma(\mathbf{z})} \right| \\
\le\ & \max_{\mathbf{y}\in\mathbb{Z}_q^n} \sup_{\mathbf{x}\in\Lambda_q^\perp(\mathbf{A})} D_{\Lambda_q^{\mathbf{y}}(\mathbf{A}),\sigma}(\mathbf{x}) \\
& + \max_{\mathbf{y}\in\mathbb{Z}_q^n} \max_{\substack{\mathbf{x}\in\mathbb{Z}_q^m,\, \|\mathbf{x}\|\le\sigma\sqrt{m} \\ \mathbf{Ax}=\mathbf{y} \pmod q}} \frac{\rho_\sigma(\mathbf{x})}{\sum\limits_{\substack{\mathbf{z}\in\mathbb{Z}_q^m, \|\mathbf{z}\|\le\sigma\sqrt{m} \\ \mathbf{Az}=\mathbf{y} \pmod q}} \rho_\sigma(\mathbf{z})} \cdot \frac{\rho_\sigma(\Lambda_q^{\mathbf{y}}(\mathbf{A}) \setminus \mathcal{B}^m(\mathbf{0}, \sigma\sqrt{m}))}{\rho_\sigma(\Lambda_q^{\mathbf{y}}(\mathbf{A}))}
\end{aligned}$$

where $B^m(\mathbf{0}, r) = \{\mathbf{x} \in \mathbb{R}^m : \|\mathbf{x}\| \le r\}$. Using the fact that

$$\frac{\rho_\sigma(\mathbf{x})}{\sum\limits_{\substack{\mathbf{z}\in\mathbb{Z}_q^m, \|\mathbf{z}\|\le\sigma\sqrt{m} \\ \mathbf{Az}=\mathbf{y} \pmod q}} \rho_\sigma(\mathbf{z})} \le 1,$$

for $\mathbf{x} \in \mathbb{Z}_q^m$ with $\mathbf{Ax} = \mathbf{y} \pmod q$, and the fact that

$$\Pr_{\mathbf{v}\sim D_{\Lambda_q^{\mathbf{y}}(\mathbf{A}),\sigma}} \left[\|\mathbf{v}\| > \sigma\sqrt{m} \right] = \frac{\rho_\sigma(\Lambda_q^{\mathbf{y}}(\mathbf{A}) \setminus \mathcal{B}^m(\mathbf{0}, \sigma\sqrt{m}))}{\rho_\sigma(\Lambda_q^{\mathbf{y}}(\mathbf{A}))}$$

we get that

$$\begin{aligned}
&\max_{\mathbf{y}\in\mathbb{Z}_q^n} \max_{\substack{\mathbf{x}\in\mathbb{Z}_q^m,\, \|\mathbf{x}\|\le\sigma\sqrt{m} \\ \mathbf{Ax}=\mathbf{y} \pmod q}} \left\{ \frac{\rho_\sigma(\mathbf{x})}{\sum\limits_{\substack{\mathbf{z}\in\mathbb{Z}_q^m, \|\mathbf{z}\|\le\sigma\sqrt{m} \\ \mathbf{Az}=\mathbf{y} \pmod q}} \rho_\sigma(\mathbf{z})} \right\} \\
&\le \max_{\mathbf{y}\in\mathbb{Z}_q^n} \left\{ \sup_{\mathbf{x}\in\Lambda_q^\perp(\mathbf{A})} D_{\Lambda_q^{\mathbf{y}}(\mathbf{A}),\sigma}(\mathbf{x}) + \Pr_{\mathbf{v}\sim D_{\Lambda_q^{\mathbf{y}}(\mathbf{A}),\sigma}} \left[\|\mathbf{v}\| > \sigma\sqrt{m} \right] \right\}.
\end{aligned}$$

Because $\sigma \ge \omega(\sqrt{\log m})$, the claim then follows from Lemma 2.5 and Lemma 2.7.

□

The Short Integer Solution problem. The *Short Integer Solution* (SIS) problem was introduced by Ajtai [Ajt96] in his seminal work on average-case lattice problems.

Definition 2.10 (Short Integer Solution problem, [Ajt96])**.** *Let $n, m \in \mathbb{N}$, $q \geq 2$ be a modulus and let $\beta > 0$ be a parameter. The Short Integer Solution problem ($\mathsf{SIS}^m_{n,q,\beta}$) problem is to find a short solution $\mathbf{x} \in \mathbb{Z}^m$ with $\|\mathbf{x}\| \leq \beta$ such that $\mathbf{A} \cdot \mathbf{x} = \mathbf{0} \pmod{q}$ given as input a matrix $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$.*

Micciancio and Regev [MR07] showed that the SIS problem is, on the average, as hard as approximating worst-case lattice problems to within small factors. Subsequently, Gentry, Peikert and Vaikuntanathan [GPV07] gave an improved reduction showing that, for parameters $m = \mathrm{poly}(n)$, $\beta = \mathrm{poly}(n)$ and prime $q \geq \beta \cdot \omega(\sqrt{n \log q})$, the average-case $\mathsf{SIS}^m_{n,q,\beta}$ problem is as hard as approximating the shortest independent vector problem (SIVP) problem in the worst case to within a factor $\gamma = \beta \cdot \tilde{O}(\sqrt{n})$. We assume that $\mathsf{SIS}^m_{n,q,\beta}$, for $m = \Omega(n \log q)$, $\beta = 2^{o(n)}$ and $q = 2^{o(n)}$, is hard against quantum adversaries running in time $\mathrm{poly}(q)$ with success probability $\mathrm{poly}(1/q)$.

The Learning with Errors problem. The *Learning with Errors* problem was introduced by Regev [Reg05] and serves as the primary basis of hardness of post-quantum cryptosystems. The problem is defined as follows.

Definition 2.11 (Learning with Errors problem, [Reg05])**.** *Let $n, m \in \mathbb{N}$ be integers, let $q \geq 2$ be a modulus and let $\alpha \in (0, 1)$ be a noise ratio parameter. The (decisional) Learning with Errors ($\mathsf{LWE}^m_{n,q,\alpha q}$) problem is to distinguish between the following samples*

$$(\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}, \mathbf{s}^\intercal \mathbf{A} + \mathbf{e}^\intercal \pmod{q}) \quad \text{and} \quad (\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}, \mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^m),$$

where $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n$ is a uniformly random vector and where $\mathbf{e} \sim D_{\mathbb{Z}^m, \alpha q}$ is a discrete Gaussian error vector. We rely on the quantum $\mathsf{LWE}^m_{n,q,\alpha q}$ assumption which states that the samples above are computationally indistinguishable for any QPT algorithm.

As shown in [Reg05], the $\mathsf{LWE}^m_{n,q,\alpha q}$ problem with parameter $\alpha q \geq 2\sqrt{n}$ is at least as hard as approximating the shortest independent vector problem (SIVP) to within a factor of $\gamma = \widetilde{O}(n/\alpha)$ in worst case lattices of dimension n. In this work we assume the subexponential hardness of $\mathsf{LWE}^m_{n,q,\alpha q}$ which relies on the worst case hardness of approximating short vector problems in lattices to within a subexponential factor. We assume that the $\mathsf{LWE}^m_{n,q,\alpha q}$ problem, for $m = \Omega(n \log q)$, $q = 2^{o(n)}$, $\alpha = 1/2^{o(n)}$, is hard against quantum adversaries running in time $\mathrm{poly}(q)$. We note that this parameter regime implies $\mathsf{SIS}^m_{n,q,\beta}$ [SSTX09].

Trapdoors for lattices. We use the following *trapdoor* property for the LWE problem.

Theorem 2.12 ([MP11], Theorem 5.1)**.** *Let $n, m \in \mathbb{N}$ and $q \in \mathbb{N}$ be a prime with $m = \Omega(n \log q)$. There exists a randomized algorithms with the following properties:*

- $\mathsf{GenTrap}(1^n, 1^m, q)$*: on input $1^n, 1^m$ and q, returns a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and a trapdoor $\mathsf{td}_{\mathbf{A}}$ such that the distribution of $\mathbf{A}$ is negligibly (in the parameter n) close to uniform.*
- $\mathsf{Invert}(\mathbf{A}, \mathsf{td}_{\mathbf{A}}, \mathbf{b})$*: on input $\mathbf{A}$, $\mathsf{td}_{\mathbf{A}}$ and $\mathbf{b} = \mathbf{s}^\intercal \cdot \mathbf{A} + \mathbf{e}^\intercal \pmod{q}$, where $\|\mathbf{e}\| \leq q/(C_T \sqrt{n \log q})$ and $C_T > 0$ is a universal constant, returns $\mathbf{s}$ and $\mathbf{e}$ with overwhelming probability over $(\mathbf{A}, \mathsf{td}_{\mathbf{A}}) \leftarrow \mathsf{GenTrap}(1^n, 1^m, q)$.*

3 Quantum Discrete Gaussian Sampling for q-ary Lattices

In this section, we review some basic facts about Gaussian superpositions and present our *quantum discrete Gaussian sampler* which is used to revoke the decryption keys for our schemes.

3.1 Gaussian Superpositions

In this section, we review some basic facts about *Gaussian superpositions*. Given $q \in \mathbb{N}$, $m \in \mathbb{N}$ and $\sigma \in (\sqrt{2m}, q/\sqrt{2m})$, we consider Gaussian superpositions over $\mathbb{Z}^m \cap (-\frac{q}{2}, \frac{q}{2}]^m$ of the form

$$|\psi\rangle = \sum_{\mathbf{x} \in \mathbb{Z}_q^m} \rho_\sigma(\mathbf{x}) \, |\mathbf{x}\rangle \, .$$

Note that the state $|\psi\rangle$ is not normalized for convenience and ease of notation. The tail bound in Lemma 2.6 implies that (the normalized variant of) $|\psi\rangle$ is within negligible trace distance of a *truncated* discrete Gaussian superposition $|\tilde{\psi}\rangle$ with support $\{\mathbf{x} \in \mathbb{Z}_q^m : \|\mathbf{x}\| \leq \sigma\sqrt{\frac{m}{2}}\}$, where

$$|\tilde{\psi}\rangle = \sum_{\mathbf{x} \in \mathbb{Z}_q^m} \sqrt{D_{\mathbb{Z}_q^m, \frac{\sigma}{\sqrt{2}}}(\mathbf{x})} \, |\mathbf{x}\rangle = \left(\sum_{\mathbf{z} \in \mathbb{Z}_q^m, \|\mathbf{z}\| \leq \sigma\sqrt{\frac{m}{2}}} \rho_{\frac{\sigma}{\sqrt{2}}}(\mathbf{z}) \right)^{-\frac{1}{2}} \sum_{\mathbf{x} \in \mathbb{Z}_q^m : \|\mathbf{x}\| \leq \sigma\sqrt{\frac{m}{2}}} \rho_\sigma(\mathbf{x}) \, |\mathbf{x}\rangle \, .$$

In this work, we consider Gaussian superpositions with parameter $\sigma = \Omega(\sqrt{m})$ which can be efficiently implemented using standard quantum state preparation techniques; for example using *quantum rejection sampling* and the *Grover-Rudolph algorithm* [GR02, Reg05, Bra18, BCM+21].

Gaussian coset states. Our key-revocable encryption schemes in Section 6 and Section 7 rely on Gaussian superpositions over $\mathbf{x} \in \mathbb{Z}_q^m$ subject to a constraint of the form $\mathbf{A} \cdot \mathbf{x} = \mathbf{y} \pmod q$, for some matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and image $\mathbf{y} \in \mathbb{Z}_q^n$. In Algorithm 1, we give a procedure called $\mathsf{GenGauss}$ that, on input $\mathbf{A}$ and $\sigma > 0$, generates a Gaussian superposition state of the form

$$|\psi_{\mathbf{y}}\rangle = \sum_{\substack{\mathbf{x} \in \mathbb{Z}_q^m: \\ \mathbf{A}\mathbf{x} = \mathbf{y}}} \rho_\sigma(\mathbf{x}) \, |\mathbf{x}\rangle \, ,$$

for some $\mathbf{y} \in \mathbb{Z}_q^n$ which is statistically close to uniform whenever $m \geq 2n \log q$ and $\sigma \geq \omega(\sqrt{\log m})$. Because $|\psi_{\mathbf{y}}\rangle$ corresponds to a (truncated) Gaussian superposition over a particular lattice coset,

$$\Lambda_q^{\mathbf{y}}(\mathbf{A}) = \{\mathbf{x} \in \mathbb{Z}^m : \mathbf{A} \cdot \mathbf{x} = \mathbf{y} \pmod q\},$$

of the q-ary lattice $\Lambda_q^\perp(\mathbf{A}) = \{\mathbf{x} \in \mathbb{Z}^m : \mathbf{A} \cdot \mathbf{x} = \mathbf{0} \pmod q\}$, we refer to it as a *Gaussian coset state*.

Finally, we recall an important property of Gaussian coset states.

Gaussian-collapsing hash functions. Unruh [Unr15] introduced the notion of collapsing hash functions in his seminal work on computationally binding quantum commitments. Informally, a hash function is called *collapsing* if it is computationally difficult to distinguish between a superposition of pre-images and a single (measured) pre-image.

In recent work, Poremba [Por22] proposed a special variant of the collapsing property with respect to *Gaussian superpositions*. Previously, Liu and Zhandry [LZ19] implicitly showed that the *Ajtai* hash function $h_{\mathbf{A}}(\mathbf{x}) = \mathbf{A}\mathbf{x} \pmod{q}$ is collapsing – and thus *Gaussian-collapsing* – via the notion of *lossy functions* and by assuming the superpolynomial hardness of (decisional) LWE.

We use the following result on the Gaussian-collapsing property of the Ajtai hash function.

Theorem 3.1 (Gaussian-collapsing property, [Por22], Theorem 4)**.** *Let $n \in \mathbb{N}$ and q be a prime with $m \geq 2n \log q$, each parameterized by $\lambda \in \mathbb{N}$. Let $\sigma \in (\sqrt{2m}, q/\sqrt{2m})$. Then, the following samples are computationally indistinguishable assuming the quantum hardness of decisional $\mathsf{LWE}^m_{n,q,\alpha q}$, for any noise ratio $\alpha \in (0,1)$ with relative noise magnitude $1/\alpha = \sigma \cdot 2^{o(n)}$:*

$$\left(\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m},\ |\psi_{\mathbf{y}}\rangle = \sum_{\substack{\mathbf{x}\in\mathbb{Z}_q^m\\ \mathbf{A}\mathbf{x}=\mathbf{y}}} \rho_\sigma(\mathbf{x})\ |\mathbf{x}\rangle\,,\ \mathbf{y}\in\mathbb{Z}_q^n\right) \approx_c \left(\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m},\ |\mathbf{x}_0\rangle\,,\ \mathbf{A}\cdot\mathbf{x}_0 \in \mathbb{Z}_q^n\right)$$

where $(|\psi_{\mathbf{y}}\rangle\,, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A},\sigma)$ and where $\mathbf{x}_0 \sim D_{\mathbb{Z}_q^m,\frac{\sigma}{\sqrt{2}}}$ is a discrete Gaussian error.

3.2 Algorithm: GenGauss

The state preparation procedure $\mathsf{GenGauss}(\mathbf{A},\sigma)$ is defined as follows.

Algorithm 1: $\mathsf{GenGauss}(\mathbf{A},\sigma)$

```
Input: Matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and parameter $\sigma = \Omega(\sqrt{m})$.
Output: Gaussian state $|\psi_{\mathbf{y}}\rangle$ and $\mathbf{y} \in \mathbb{Z}_q^n$.
Prepare a Gaussian superposition in system $X$ with parameter $\sigma > 0$:
    $|\psi\rangle_{XY} = \sum_{\mathbf{x}\in\mathbb{Z}_q^m} \rho_\sigma(\mathbf{x})\,|\mathbf{x}\rangle_X \otimes |\mathbf{0}\rangle_Y\,.$
Apply the unitary $U_{\mathbf{A}} : |\mathbf{x}\rangle\,|\mathbf{0}\rangle \rightarrow |\mathbf{x}\rangle\,|\mathbf{A}\cdot\mathbf{x} \pmod{q}\rangle$ on systems $X$ and $Y$:
    $|\psi\rangle_{XY} = \sum_{\mathbf{x}\in\mathbb{Z}_q^m} \rho_\sigma(\mathbf{x})\,|\mathbf{x}\rangle_X \otimes |\mathbf{A}\cdot\mathbf{x} \pmod{q}\rangle_Y\,.$
Measure system $Y$ in the computational basis, resulting in the state
    $|\psi_{\mathbf{y}}\rangle_{XY} = \sum_{\substack{\mathbf{x}\in\mathbb{Z}_q^m:\\ \mathbf{A}\mathbf{x}=\mathbf{y}}} \rho_\sigma(\mathbf{x})\,|\mathbf{x}\rangle_X \otimes |\mathbf{y}\rangle_Y\,.$
Output the state $|\psi_{\mathbf{y}}\rangle$ in system $X$ and the outcome $\mathbf{y} \in \mathbb{Z}_q^n$ in system $Y$.
```

3.3 Algorithm: QSampGauss

Recall that, in Algorithm 1, we gave a procedure called $\mathsf{GenGauss}(\mathbf{A},\sigma)$ that prepares a Gaussian coset state $|\psi_{\mathbf{y}}\rangle$, for a randomly generated $\mathbf{y} \in \mathbb{Z}_q^n$. In general, however, generating a specific Gaussian coset state on input $(\mathbf{A},\mathbf{y})$ requires a *short trapdoor basis* $\mathsf{td}_{\mathbf{A}}$ for the matrix $\mathbf{A}$. This task can be thought of as a quantum analogue of the *discrete Gaussian sampling problem* [GPV07], where the goal is to output a sample $\mathbf{x} \sim D_{\mathbb{Z}^m,\sigma}$ such that $\mathbf{A}\cdot\mathbf{x} = \mathbf{y} \pmod q$ on input $(\mathbf{A},\mathbf{y})$ and $\sigma > 0$.

In Algorithm 2, we give a procedure called $\mathsf{QSampGauss}$ which, on input $(\mathbf{A},\mathsf{td}_{\mathbf{A}},\mathbf{y},\sigma)$ generates a specific Gaussian coset state $|\psi_{\mathbf{y}}\rangle$ of the form

$$|\psi_{\mathbf{y}}\rangle = \sum_{\substack{\mathbf{x}\in\mathbb{Z}_q^m\\ \mathbf{A}\mathbf{x}=\mathbf{y}}} \rho_\sigma(\mathbf{x})\,|\mathbf{x}\rangle\,.$$

Our procedure $\mathsf{SampGauss}$ in Algorithm 2 can be thought of as an explicit quantum reduction from $\mathsf{ISIS}^m_{n,q,\sigma\sqrt{m/2}}$ to $\mathsf{LWE}^m_{n,q,q/\sqrt{2}\sigma}$ which is inspired by the quantum reduction of Stehlé et al. [SSTX09] which reduces SIS to LWE. To obtain the aforementioned reduction, one simply needs to replace the procedure $\mathsf{Invert}(\mathbf{A},\mathsf{td}_{\mathbf{A}},\cdot)$ in Step 4 in Algorithm 2 with a solver for the $\mathsf{LWE}^m_{n,q,q/\sqrt{2}\sigma}$ problem.

In Theorem 3.3, we prove the correctness of Algorithm 2. As a technical ingredient, we rely on a *duality lemma* [Por22] that characterizes the Fourier transform of a Gaussian coset state in terms of its dual state. Note that $|\psi_{\mathbf{y}}\rangle$ corresponds to a Gaussian superposition over a lattice coset,

$$\Lambda_q^{\mathbf{y}}(\mathbf{A}) = \{\mathbf{x}\in\mathbb{Z}^m : \mathbf{A}\cdot\mathbf{x} = \mathbf{y} \pmod q\},$$

of the q-ary lattice $\Lambda_q^\perp(\mathbf{A}) = \{\mathbf{x}\in\mathbb{Z}^m : \mathbf{A}\cdot\mathbf{x} = \mathbf{0} \pmod q\}$. Here, the *dual* of $\Lambda_q^\perp(\mathbf{A})$ satisfies $q\cdot\Lambda_q^\perp(\mathbf{A})^* = \Lambda_q(\mathbf{A})$, where $\Lambda_q(\mathbf{A})$ corresponds to the lattice generated by $\mathbf{A}^\intercal$, i.e.

$$\Lambda_q(\mathbf{A}) = \{\mathbf{z}\in\mathbb{Z}^m : \mathbf{z} = \mathbf{A}^\intercal\cdot\mathbf{s} \pmod q, \text{ for some } \mathbf{s}\in\mathbb{Z}^n\}.$$

The following lemma relates the Fourier transform of $|\psi_{\mathbf{y}}\rangle$ with a superposition of LWE samples with respect to a matrix $\mathbf{A}\in\mathbb{Z}_q^{n\times m}$ and a phase which depends on $\mathbf{y}$. In other words, the resulting state can be thought of as a superposition of Gaussian balls around random lattice vectors in $\Lambda_q(\mathbf{A})$.

Lemma 3.2 ([Por22], Lemma 16)**.** *Let $m\in\mathbb{N}$, $q\geq 2$ be a prime and $\sigma\in(\sqrt{2m},q/\sqrt{2m})$. Let $\mathbf{A}\in\mathbb{Z}_q^{n\times m}$ be a matrix whose columns generate $\mathbb{Z}_q^n$ and let $\mathbf{y}\in\mathbb{Z}_q^n$ be arbitrary. Then, the q-ary quantum Fourier transform of the (normalized variant of the) Gaussian coset state*

$$|\psi_{\mathbf{y}}\rangle = \sum_{\substack{\mathbf{x}\in\mathbb{Z}_q^m\\ \mathbf{A}\mathbf{x}=\mathbf{y} \pmod q}} \rho_\sigma(\mathbf{x})\,|\mathbf{x}\rangle$$

is within negligible (in $m\in\mathbb{N}$) trace distance of the (normalized variant of the) Gaussian state

$$|\hat{\psi}_{\mathbf{y}}\rangle = \sum_{\mathbf{s}\in\mathbb{Z}_q^n}\sum_{\mathbf{e}\in\mathbb{Z}_q^m} \rho_{q/\sigma}(\mathbf{e})\,\omega_q^{-\langle\mathbf{s},\mathbf{y}\rangle}\,|\mathbf{s}^\intercal\mathbf{A}+\mathbf{e}^\intercal \pmod q\rangle\,.$$

The procedure $\mathsf{QSampGauss}(\mathbf{A},\mathsf{td}_{\mathbf{A}},\mathbf{y},\sigma)$ is defined as follows.

```
Algorithm 2: QSampGauss($\mathbf{A}, \mathsf{td}_{\mathbf{A}}, \mathbf{y}, \sigma$)
    Input: Matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, a trapdoor $\mathsf{td}_{\mathbf{A}}$, an image $\mathbf{y} \in \mathbb{Z}_q^n$ and parameter $\sigma = O(\frac{q}{\sqrt{m}})$.
    Output: Gaussian state $|\psi_{\mathbf{y}}\rangle$.
    Prepare the following superposition with parameter $q/\sigma > 0$:
        $$\sum_{\mathbf{s} \in \mathbb{Z}_q^n} |\mathbf{s}\rangle \otimes \sum_{\mathbf{e} \in \mathbb{Z}_q^m} \rho_{q/\sigma}(\mathbf{e}) |\mathbf{e}\rangle \otimes |\mathbf{0}\rangle$$
    Apply the generalized Pauli operator $\mathbf{Z}_q^{-\mathbf{y}}$ on the first register, resulting in the state
        $$\sum_{\mathbf{s} \in \mathbb{Z}_q^n} \omega_q^{-\langle \mathbf{s}, \mathbf{y} \rangle} |\mathbf{s}\rangle \otimes \sum_{\mathbf{e} \in \mathbb{Z}_q^m} \rho_{q/\sigma}(\mathbf{e}) |\mathbf{e}\rangle \otimes |\mathbf{0}\rangle$$
    Apply the unitary $U_{\mathbf{A}} : |\mathbf{s}\rangle |\mathbf{e}\rangle |\mathbf{0}\rangle \to |\mathbf{s}\rangle |\mathbf{e}\rangle |\mathbf{s}^\intercal \mathbf{A} + \mathbf{e}^\intercal \pmod{q}\rangle$, resulting in the state
        $$\sum_{\mathbf{s} \in \mathbb{Z}_q^n} \sum_{\mathbf{e} \in \mathbb{Z}_q^m} \rho_{q/\sigma}(\mathbf{e})\, \omega_q^{-\langle \mathbf{s}, \mathbf{y} \rangle} |\mathbf{s}\rangle |\mathbf{e}\rangle |\mathbf{s}^\intercal \mathbf{A} + \mathbf{e}^\intercal \pmod{q}\rangle$$
    Coherently run Invert($\mathbf{A}, \mathsf{td}_{\mathbf{A}}, \cdot$) on the third register in order to uncompute the first and the second register, resulting in a state that is close in trace distance to the following state:
        $$\sum_{\mathbf{s} \in \mathbb{Z}_q^n} \sum_{\mathbf{e} \in \mathbb{Z}_q^m} \rho_{q/\sigma}(\mathbf{e})\, \omega_q^{-\langle \mathbf{s}, \mathbf{y} \rangle} |0\rangle |0\rangle |\mathbf{s}^\intercal \mathbf{A} + \mathbf{e}^\intercal \pmod{q}\rangle$$
    Discard the first two registers. Apply the (inverse) quantum Fourier transform and output the resulting state.
```

Let us now prove the correctness of Algorithm 2.

Theorem 3.3 (Quantum Discrete Gaussian Sampler)**.** *Let $n \in \mathbb{N}$, q be a prime with $m \geq 2n \log q$ and $\sigma \in (\sqrt{2m}, q/\sqrt{2m})$. Let $(\mathbf{A}, \mathsf{td}_{\mathbf{A}}) \leftarrow \mathsf{GenTrap}(1^n, 1^m, q)$ be sampled as in Theorem 2.12 and let $\mathbf{y} \in \mathbb{Z}_q^n$ be arbitrary. Then, with overwhelming probability, $\mathsf{QSampGauss}(\mathbf{A}, \mathsf{td}_{\mathbf{A}}, \mathbf{y}, \sigma)$ in Algorithm 2 outputs a state which is within negligible trace distance of the (normalized variant of the) state,*

$$|\psi_{\mathbf{y}}\rangle = \sum_{\substack{\mathbf{x} \in \mathbb{Z}_q^m \\ \mathbf{A}\mathbf{x} = \mathbf{y} \pmod{q}}} \rho_\sigma(\mathbf{x}) |\mathbf{x}\rangle .$$

Proof. From Lemma 2.4 and Theorem 2.12, it follows that $(\mathbf{A}, \mathsf{td}_{\mathbf{A}}) \leftarrow \mathsf{GenTrap}(1^n, 1^m, q)$ yields a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ be a matrix whose columns generate $\mathbb{Z}_q^n$ with overwhelming probability. Moreover, since $\sigma \in (\sqrt{2m}, q/\sqrt{2m})$, the inversion procedure $\mathsf{Invert}(\mathbf{A}, \mathsf{td}_{\mathbf{A}}, \cdot)$ from Theorem 2.12 in Step 4 in

Algorithm 2 succeeds with overwhelming probability at generating the Gaussian state

$$|\hat{\psi}_{\mathbf{y}}\rangle = \sum_{\mathbf{s}\in\mathbb{Z}_q^n} \sum_{\mathbf{e}\in\mathbb{Z}_q^m} \rho_{q/\sigma}(\mathbf{e})\, \omega_q^{-\langle \mathbf{s},\mathbf{y}\rangle} \left|\mathbf{s}^\intercal \mathbf{A} + \mathbf{e}^\intercal \pmod q\right\rangle$$

Applying the (inverse) quantum Fourier transform $\mathsf{FT}_q^\dagger$, the claim then follows from Lemma 3.2. □

4 Quantum Goldreich-Levin Theorem for Large Fields

In this section, we give a proof of the first quantum Goldreich-Levin theorem for large fields $\mathbb{Z}_q$.

4.1 Post-Quantum Reductions and Quantum Rewinding

We first review some recent work by Bitansky, Brakerski and Kalai [BBK22] that enables us to convert a wide range of classical reductions into post-quantum reductions (which allow for quantum auxiliary input) in a constructive manner. We first review some basic terminology from [BBK22].

Let $\lambda \in \mathbb{N}$ be a parameter. A *non-interactive assumption* $\mathsf{P} = (\mathsf{G}, \mathsf{V}, c)$ with respect to a set of polynomials $d(\lambda), n(\lambda)$ and $m(\lambda)$ is characterized as follows:

- The generator G takes as input 1^λ and $r \in \{0,1\}^d$, and returns $x \in \{0,1\}^n$.
- The verifier V takes as input 1^λ and $(r, y) \in \{0,1\}^d \times \{0,1\}^m$, and returns a single bit output.
- $c(\lambda)$ is the threshold associated with the assumption.

Given a (possibly randomized) *solver*, we characterize the *advantage* in solving an assumption P in terms of the absolute distance between the solving probability (or, *value*) and the threshold c; for example, for a *decision assumption* P (with $m = 1$) we characterize the value in solving P in terms of $\frac{1}{2} + \varepsilon$, where the threshold is given by $c(\lambda) = \frac{1}{2}$ and $\varepsilon > 0$ is corresponds to the *advantage*. We say that a reduction is *black-box* if it is oblivious to the representation and inner workings of the solver that is being used. Moreover, we say that a reduction is *non-adaptive* if all queries to the solver are known ahead of time.

We use the following theorem.

Theorem 4.1 ([BBK22], adapted from Theorem 7.1)**.** *Let $c \in \mathbb{R}$. Suppose that there exists a classical reduction from solving a non-interactive assumption Q to solving a non-interactive assumption P such that the following holds: if the P-solver has advantage $\varepsilon > 0$ then the Q-solver has advantage c (independent of ε) with running time* $\mathrm{poly}(1/\varepsilon, c, \lambda)$.

Then, there exists a quantum reduction from solving Q to quantumly solving P such that the following holds: if the quantum P-solver (with non-uniform quantum advice) has advantage given by $\varepsilon > 0$, then the the Q-solver has advantage c (the same as the classical reduction) with running time $\mathrm{poly}(1/\varepsilon, c, \lambda)$.

Remark 4.2. *We note that [BBK22] consider a more general theorem where the advantage of the classical Q-solver can depend on the advantage of the P-solver. But in the case when the classical Q-solver's advantage is independent of the P-solver's advantage then, as reflected in the above theorem, it turns out the advantage of the quantum Q-solver is the same as the classical Q-solver.*

4.2 Goldreich-Levin Theorems for Large Fields

The following result is implicit in the work of Dodis et al. [DGT+10].

Theorem 4.3 (Classical Goldreich-Levin Theorem for Large Fields, [DGT+10], Theorem 1)**.** *Let q be a prime and $m \in \mathbb{N}$. Let $\sigma \in (2\sqrt{m}, q/2\sqrt{m})$ and let $H = \{\mathbf{x} \in \mathbb{Z}_q^m \, : \, \|\mathbf{x}\| \leq \sigma\sqrt{m}\}$ be a subset of $\mathbb{Z}_q^m$. Let $\mathsf{aux} : H \to \{0,1\}^*$ be any (possibly randomized) auxiliary information. Suppose there exists a distinguisher $\mathcal{D}$ which runs in time $T(\mathcal{D})$ such that*

$$\left|\Pr\left[\mathcal{D}\big(\mathbf{u},\mathbf{u}^\intercal\mathbf{x},\mathsf{aux}(\mathbf{x})\big)=1 \ : \ \substack{\mathbf{u}\xleftarrow{\$}\mathbb{Z}_q^m\\ \mathbf{x}\sim D_{\mathbb{Z}_q^m,\sigma}}\right] - \Pr\left[\mathcal{D}\big(\mathbf{u},r,\mathsf{aux}(\mathbf{x})\big)=1 \ : \ \substack{\mathbf{u}\xleftarrow{\$}\mathbb{Z}_q^m,\, r\xleftarrow{\$}\mathbb{Z}_q\\ \mathbf{x}\sim D_{\mathbb{Z}_q^m,\sigma}}\right]\right| = \varepsilon.$$

Then, there exists a (classical) non-adaptive black-box extractor $\mathcal{E}$ whose running time is given by $T(\mathcal{E}) = T(\mathcal{D}) \cdot \mathrm{poly}(m, \sigma, 1/\varepsilon)$ and succeeds with probability at least

$$\Pr\left[\mathcal{E}\big(\mathsf{aux}(\mathbf{x})\big) = \mathbf{x} \ : \ \mathbf{x} \sim D_{\mathbb{Z}_q^m,\sigma}\right] \geq \frac{\varepsilon^3}{512 \cdot m \cdot q^2}.$$

Using the constructive post-quantum reduction from Theorem 4.1, we can convert Theorem 4.3 into a quantum Goldreich-Levin Theorem for finite fields, and obtain the following.

Theorem 4.4 (Quantum Goldreich-Levin Theorem for Large Fields)**.** *Let q be a prime and $m \in \mathbb{N}$. Let $\sigma \in (2\sqrt{m}, q/2\sqrt{m})$ and let $\Phi : \mathcal{L}(\mathcal{H}_q^m) \to \mathcal{L}(\mathcal{H}_{\mathrm{Aux}})$ be any* CPTP *map with auxiliary system $\mathcal{H}_{\mathrm{Aux}}$. Suppose there exists a distinguisher $\mathcal{D}$ which runs in time $T(\mathcal{D})$ such that*

$$\left|\Pr\left[\mathcal{D}\big(\mathbf{u},\mathbf{u}^\intercal\mathbf{x},\mathsf{aux}(\mathbf{x})\big)=1 \ : \ \substack{\mathbf{u}\xleftarrow{\$}\mathbb{Z}_q^m\\ \mathbf{x}\sim D_{\mathbb{Z}_q^m,\sigma}\\ \mathsf{aux}(\mathbf{x})\leftarrow\Phi(|\mathbf{x}\rangle\langle\mathbf{x}|)}\right] - \Pr\left[\mathcal{D}\big(\mathbf{u},r,\mathsf{aux}(\mathbf{x})\big)=1 \ : \ \substack{\mathbf{u}\xleftarrow{\$}\mathbb{Z}_q^m,\, r\xleftarrow{\$}\mathbb{Z}_q\\ \mathbf{x}\sim D_{\mathbb{Z}_q^m,\sigma}\\ \mathsf{aux}(\mathbf{x})\leftarrow\Phi(|\mathbf{x}\rangle\langle\mathbf{x}|)}\right]\right| = \varepsilon.$$

Then, there exists a quantum extractor $\mathcal{E}$ that runs in time $T(\mathcal{E}) = \mathrm{poly}(m, T(\mathcal{D}), \sigma, q, 1/\varepsilon)$ with

$$\Pr\left[\mathcal{E}\big(\mathsf{aux}(\mathbf{x})\big) = \mathbf{x} \ : \ \substack{\mathbf{x}\sim D_{\mathbb{Z}_q^m,\sigma}\\ \mathsf{aux}(\mathbf{x})\leftarrow\Phi(|\mathbf{x}\rangle\langle\mathbf{x}|)}\right] \geq \mathrm{poly}\big(1/m, 1/T(\mathcal{D}), 1/\sigma, 1/q, \varepsilon\big).$$

Proof. The proof follows immediately by combining Theorem 4.3 and Theorem 4.1. □

4.3 Amplification

We now show that it is possible to *boost* the success probability of the Goldreich-Levin extractor, assuming a particular kind of leakage on the hidden vector. Consider the following algorithm.

```
Algorithm 3: BoostedExtractor($\mathbf{A}, \mathbf{y}, \mathsf{aux}(\mathbf{x})$)
  Input: Matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, vector $\mathbf{y} \in \mathbb{Z}_q^n$ and auxiliary input $\mathsf{aux}(\mathbf{x}) \in \{0,1\}^*$.
  Parameters: $\nu, \delta \in (0,1)$.
  Output: Vector $\mathbf{x} \in \mathbb{Z}_q^m$.
  for $i = 1, \ldots, \lceil \frac{1}{\nu} \ln\left(\frac{1}{\delta}\right) \rceil$ do
      run $\mathbf{x}_i \leftarrow \mathcal{E}(\mathsf{aux}(\mathbf{x}))$, where $\mathcal{E}$ is the Goldreich-Levin extractor in Theorem 4.3.
      if $\mathbf{x}_i \in \Lambda_q^{\mathbf{y}}(\mathbf{A}) \cap \mathcal{B}^m(\mathbf{0}, \sigma\sqrt{m})$ then
          output $\mathbf{x}_i$
      else
          continue
      end
  end
```

Theorem 4.5 (Boosted Classical Goldreich-Levin Theorem for Large Fields)**.** *Let $n, m \in \mathbb{N}$ be integers and let q be a prime. Let $\sigma \in (2\sqrt{m}, q/2\sqrt{m})$ and let $H = \{\mathbf{x} \in \mathbb{Z}_q^m : \|\mathbf{x}\| \leq \sigma\sqrt{m}\}$ be a subset of $\mathbb{Z}_q^m$. Let $\mathsf{aux} : H \to \{0,1\}^*$ be any (possibly randomized) auxiliary information. Suppose that there exists a distinguisher $\mathcal{D}$ which runs in time $T(\mathcal{D})$ such that*

$$\left| \Pr\left[\mathcal{D}\big(\mathbf{A}, \mathbf{y}, \mathbf{u}, \mathbf{u}^\intercal\mathbf{x}, \mathsf{aux}(\mathbf{x})\big) = 1 : \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m} \\ \mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^m \\ \mathbf{x} \sim D_{\mathbb{Z}_q^m, \sigma} \\ \mathbf{y} \leftarrow \mathbf{A}\mathbf{x} (\bmod q) \end{array} \right] - \Pr\left[\mathcal{D}\big(\mathbf{A}, \mathbf{y}, \mathbf{u}, r, \mathsf{aux}(\mathbf{x})\big) = 1 : \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m} \\ \mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^m \\ r \xleftarrow{\$} \mathbb{Z}_q \\ \mathbf{x} \sim D_{\mathbb{Z}_q^m, \sigma} \\ \mathbf{y} \leftarrow \mathbf{A}\mathbf{x} (\bmod q) \end{array} \right] \right| = \varepsilon.$$

Let $\nu = 512mq^2/\varepsilon^3$ and $\delta = \exp(-\Omega(n))$ be parameters. Then, $\mathsf{BoostedExtractor}(\mathbf{A}, \mathbf{y}, \mathsf{aux}(\mathbf{x}))$ in Algorithm 3 is a non-adaptive black-box extractor that runs in time $T(\mathcal{D}) \cdot \mathrm{poly}(n, m, \sigma, q, 1/\varepsilon)$ and outputs a short vector in the coset $\Lambda_q^{\mathbf{y}}(\mathbf{A})$ with probability at least

$$\Pr\left[\mathsf{BoostedExtractor}(\mathbf{A}, \mathbf{y}, \mathsf{aux}(\mathbf{x})) \in \Lambda_q^{\mathbf{y}}(\mathbf{A}) \cap \mathcal{B}^m(\mathbf{0}, \sigma\sqrt{m}) \ : \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m} \\ \mathbf{x} \sim D_{\mathbb{Z}_q^m, \sigma} \\ \mathbf{y} \leftarrow \mathbf{A}\mathbf{x} \pmod q \end{array} \right] \geq 1 - \exp(-\Omega(n)).$$

Proof. Recall that the Goldreich-Levin extractor $\mathcal{E}$ in Theorem 4.3 is a non-adaptive black-box extractor running in time $T(\mathcal{E}) = T(\mathcal{D}) \cdot \mathrm{poly}(m, \sigma, 1/\varepsilon)$ that, on input $\mathsf{aux}(\mathbf{x})$, outputs $\mathbf{x}$ with probability at least $\varepsilon^3/512mq^2$. Let $L = \lceil \frac{1}{\nu} \ln\left(\frac{1}{\delta}\right) \rceil$ with $\nu = 512mq^2/\varepsilon^3$ and $\delta = \exp(-\Omega(n))$. Therefore, the probability that $\mathsf{BoostedExtractor}(\mathbf{A}, \mathbf{y}, \mathsf{aux}(\mathbf{x}))$ in Algorithm 3 fails is at most

$$(1-\nu)^L \leq \exp(-L \cdot \nu) \leq \exp(-\Omega(n)).$$

This proves the claim. □

Using the constructive post-quantum reduction from Theorem 4.1, we can convert Theorem 4.5 into a (boosted) quantum Goldreich-Levin Theorem for finite fields, and obtain the following.

Theorem 4.6 (Boosted Quantum Goldreich-Levin Theorem for Large Fields)**.** *Let $n, m \in \mathbb{N}$ and q be a prime. Let $\sigma \in (2\sqrt{m}, q/2\sqrt{m})$. Let $\Phi : \mathcal{L}(\mathcal{H}_q^m) \to \mathcal{L}(\mathcal{H}_{\mathrm{AUX}})$ be any CPTP map with auxiliary system $\mathcal{H}_{\mathrm{AUX}}$. Suppose that there exists a distinguisher $\mathcal{D}$ which runs in time $T(\mathcal{D})$ such that*

$$\left| \Pr\left[\mathcal{D}\big(\mathbf{A}, \mathbf{y}, \mathbf{u}, \mathbf{u}^\intercal\mathbf{x}, \mathsf{aux}(\mathbf{x})\big) = 1 : \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m} \\ \mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^m \\ \mathbf{x} \sim D_{\mathbb{Z}_q^m, \sigma} \\ \mathbf{y} \leftarrow \mathbf{A}\mathbf{x} (\bmod q) \end{array} \right] - \Pr\left[\mathcal{D}\big(\mathbf{A}, \mathbf{y}, \mathbf{u}, r, \mathsf{aux}(\mathbf{x})\big) = 1 : \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m} \\ \mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^m \\ r \xleftarrow{\$} \mathbb{Z}_q \\ \mathbf{x} \sim D_{\mathbb{Z}_q^m, \sigma} \\ \mathbf{y} \leftarrow \mathbf{A}\mathbf{x} (\bmod q) \end{array} \right] \right| = \varepsilon,$$

where $\mathsf{aux}(\mathbf{x}) \leftarrow \Phi(|\mathbf{x}\rangle\langle\mathbf{x}|)$. Then, there exists a quantum extractor $\mathcal{E}$ that has a running time of $T(\mathcal{E}) = T(\mathcal{D}) \cdot \mathrm{poly}(n, m, \sigma, q, 1/\varepsilon)$ and outputs a short vector in $\Lambda_q^{\mathbf{y}}(\mathbf{A})$ with probability at least

$$\Pr\left[\mathcal{E}(\mathbf{A}, \mathbf{y}, \mathsf{aux}(\mathbf{x})) \in \Lambda_q^{\mathbf{y}}(\mathbf{A}) \cap \mathcal{B}^m(\mathbf{0}, \sigma\sqrt{m}) \ : \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m} \\ \mathbf{x} \sim D_{\mathbb{Z}_q^m, \sigma} \\ \mathbf{y} \leftarrow \mathbf{A}\mathbf{x} \pmod q \end{array} \right] \geq 1 - \exp(-\Omega(n)).$$

Proof. The proof follows immediately by combining Theorem 4.5 and Theorem 4.1. □

5 Definition: Key-Revocable Public-Key Encryption

Let us now give a formal definition of key-revocable public-key encryption schemes.

Definition 5.1 (Key-Revocable Public-Key Encryption). *A key-revocable public-key encryption scheme consists efficient algorithms* $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Revoke})$*, where* Enc *is a* PPT *algorithm and* $\mathsf{KeyGen}, \mathsf{Dec}$ *and* Revoke *are* QPT *algorithms defined as follows:*

- $\mathsf{KeyGen}(1^\lambda)$*: given as input a security parameter* λ*, output a public key* PK*, a master secret key* MSK *and a quantum decryption key* ρ_{SK}*.*
- $\mathsf{Enc}(\mathsf{PK}, x)$*: given a public key* PK *and plaintext* $x \in \{0,1\}^\ell$*, output a ciphertext* CT*.*
- $\mathsf{Dec}(\rho_{\mathsf{SK}}, \mathsf{CT})$*: given a decryption key* ρ_{SK} *and ciphertext* CT*, output a message* y*.*
- $\mathsf{Revoke}\left(\mathsf{PK}, \mathsf{MSK}, \sigma\right)$*: given as input a master secret key* MSK*, a public key* PK *and quantum state* σ*, output* Valid *or* $\mathsf{Invalid}$*.*

Correctness of Decryption. For every $x \in \{0,1\}^\ell$, the following holds:

$$\Pr\left[x \leftarrow \mathsf{Dec}(\rho_{\mathsf{SK}}, \mathsf{CT}) \;:\; \substack{(\mathsf{PK}, \mathsf{MSK}, \rho_{\mathsf{SK}}) \leftarrow \mathsf{KeyGen}(1^\lambda) \\ \mathsf{CT} \leftarrow \mathsf{Enc}(\mathsf{PK}, x)}\right] \geq 1 - \nu(\lambda),$$

where $\nu(\cdot)$ is a negligible function.

Correctness of Revocation. The following holds:

$$\Pr\left[\mathsf{Valid} \leftarrow \mathsf{Revoke}\left(\mathsf{PK}, \mathsf{MSK}, \rho_{\mathsf{SK}}\right) \;:\; (\mathsf{PK}, \mathsf{MSK}, \rho_{\mathsf{SK}}) \leftarrow \mathsf{KeyGen}(1^\lambda)\right] \geq 1 - \nu(\lambda),$$

where $\nu(\cdot)$ is a negligible function.

Remark 5.2. *Using the well-known "Almost As Good As New Lemma" (Lemma 2.2), the procedure* Dec *can be purified to obtain another quantum circuit* $\widetilde{\mathsf{Dec}}$ *such that* $\widetilde{\mathsf{Dec}}(\rho_{\mathsf{SK}}, \mathsf{CT})$ *yields* (x, ρ'_{SK}) *with probability at least* $1 - \nu(\lambda)$ *and moreover,* CT *is an encryption of* x *and* $\mathrm{TD}(\rho'_{\mathsf{SK}}, \rho_{\mathsf{SK}}) \leq \nu'(\lambda)$ *with* $\nu'(\lambda)$ *is another negligible function.*

5.1 Security Definition

Let $\Sigma = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Revoke})$ be a key-revocable public-key encryption scheme. We consider the following security experiment, defined below.

Definition 5.3. *A key-revocable public-key encryption scheme* $\Sigma = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Revoke})$ *is secure if, for every* QPT *adversary* $\mathcal{A}$*, the following holds:*

$$\Pr\left[b \leftarrow \mathsf{Expt}_\Sigma^{\mathcal{A}}(1^\lambda, b) \;:\; b \xleftarrow{\$} \{0,1\}\right] \leq \frac{1}{2} + \mathsf{negl}(\lambda),$$

where $\mathsf{Expt}_\Sigma^{\mathcal{A}}(1^\lambda, b)$ *is as defined in Figure 1.*

$\underline{\mathsf{Expt}_{\Sigma}^{\mathcal{A}}\left(1^{\lambda}, b\right)}$:

Initialization Phase:

- The challenger runs $(\mathsf{PK}, \mathsf{MSK}, \rho_{\mathsf{SK}}) \leftarrow \mathsf{KeyGen}(1^{\lambda})$ and sends $(\mathsf{PK}, \rho_{\mathsf{SK}})$ to $\mathcal{A}$.

Revocation Phase:

- The challenger sends the message `REVOKE` to $\mathcal{A}$.
- The adversary $\mathcal{A}$ returns a state σ.
- The challenger aborts if $\mathsf{Revoke}(\mathsf{PK}, \mathsf{MSK}, \sigma)$ outputs $\mathsf{Invalid}$.

Guessing Phase:

- $\mathcal{A}$ submits a plaintext $x \in \{0,1\}^{\ell}$ to the challenger.
- If $b = 0$: The challenger sends $\mathsf{CT} \leftarrow \mathsf{Enc}(\mathsf{PK}, x)$ to $\mathcal{A}$. Else, if $b = 1$, the challenger sends $\mathsf{CT} \xleftarrow{\$} \mathcal{C}$, where $\mathcal{C}$ is the ciphertext space of ℓ bit messages.
- Output $b_{\mathcal{A}}$ if the output of $\mathcal{A}$ is $b_{\mathcal{A}}$.

Figure 1: Security Experiment

Remark 5.4. *In the traditional setting, 1-bit unpredictability and computational indistinguishability are equivalent in the following sense: if there are two distributions D_0 and D_1 such that an efficient adversary can distinguish these two distributions with advantage ϵ then the same adversary can predict D_0 versus D_1 with probability $\frac{1}{2} + \frac{\epsilon}{2}$.*

This observation no longer applies to the above setting where we simultaneously need to consider the success probability of Revoke. *As a result, our definition is incomparable with a variant of the above definition where we instead require the adversary to distinguish a valid ciphertext versus uniform.*

Hybrid Lemma. We present a hybrid lemma for 1-bit unpredictability below. This lemma will be useful in applications where we can employ hybrid argument in a similar vein as done in the computational indistinguishability setting.

Lemma 5.5 (Hybrid Lemma for 1-Bit Unpredictability)**.** *Suppose there exists a sequence of hybrid experiments $\mathsf{H}_1, \ldots, \mathsf{H}_k$ such that any* QPT *predictor $\mathcal{A}$ can predict H_i versus H_{i+1} with advantage at most ϵ_i. Then, $\mathcal{A}$ can only predict hybrid H_1 versus H_k with advantage of at most $\sum_{i=1}^{k-1} \epsilon_i$.*

Proof. Let $\mathcal{A}$ be a QPT adversary and suppose that for $i \in [k-1]$:

$$\frac{1}{2}\mathsf{Pr}[0 \leftarrow \mathsf{H}_i^{\mathcal{A}}] + \frac{1}{2}\mathsf{Pr}[1 \leftarrow \mathsf{H}_{i+1}^{\mathcal{A}}] = \frac{1}{2} + \epsilon_i.$$

We give a proof by induction. First, note that the base case for $k = 2$ follows immediately by the definition of ϵ_1. Now fix an arbitrary $k \geq 2$, and suppose that $\mathcal{A}$ can predict hybrid H_1 versus H_k with advantage at most $\sum_{i=1}^{k-1} \epsilon_i$. In other words, by the induction hypothesis we have

$$\frac{1}{2}\mathsf{Pr}[0 \leftarrow \mathsf{H}_1^{\mathcal{A}}] + \frac{1}{2}\mathsf{Pr}[1 \leftarrow \mathsf{H}_k^{\mathcal{A}}] = \frac{1}{2} + \sum_{i=1}^{k-1} \epsilon_i.$$

Suppose also that

$$\frac{1}{2}\mathsf{Pr}[0 \leftarrow \mathsf{H}_k^{\mathcal{A}}] + \frac{1}{2}\mathsf{Pr}[1 \leftarrow \mathsf{H}_{k+1}^{\mathcal{A}}] = \frac{1}{2} + \epsilon_k.$$

By taking the sum of the two equations above, we get

$$\frac{1}{2}\mathsf{Pr}[0 \leftarrow \mathsf{H}_1^{\mathcal{A}}] + \frac{1}{2}\mathsf{Pr}[1 \leftarrow \mathsf{H}_k^{\mathcal{A}}] + \frac{1}{2}\mathsf{Pr}[0 \leftarrow \mathsf{H}_k^{\mathcal{A}}] + \frac{1}{2}\mathsf{Pr}[1 \leftarrow \mathsf{H}_{k+1}^{\mathcal{A}}] = 1 + \sum_{i=1}^{k} \epsilon_i.$$

Using the identity $\mathsf{Pr}[0 \leftarrow \mathsf{H}_k^{\mathcal{A}}] + \mathsf{Pr}[1 \leftarrow \mathsf{H}_k^{\mathcal{A}}] = 1$, we obtain the desired identity for $k+1$:

$$\frac{1}{2}\mathsf{Pr}[0 \leftarrow \mathsf{H}_1^{\mathcal{A}}] + \frac{1}{2}\mathsf{Pr}[1 \leftarrow \mathsf{H}_{k+1}^{\mathcal{A}}] = \frac{1}{2} + \sum_{i=1}^{k} \epsilon_i.$$

This proves the claim. □

5.2 Key-Revocable Public-Key Fully Homomorphic Encryption

A key-revocable public-key fully homomorphic encryption scheme defined for a class of functions $\mathcal{F}$, in addition to $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Revoke})$, consists of the following PPT algorithm:

- $\mathsf{Eval}(\mathsf{PK}, f, \mathsf{CT})$: on input a public key PK, function $f \in \mathcal{F}$, ciphertext CT, outputs another ciphertext CT'.

Remark 5.6. *Sometimes we allow* KeyGen *to additionally take as input different parameters associated with the implementations of the functions in* $\mathcal{F}$*. For example, we allow* KeyGen *to take as input a parameter* L *in such a way that all the parameters in the system depend on* L *and moreover, the homomorphic evaluation is only supported on circuits (in* $\mathcal{F}$*) of depth at most* L*.*

Correctness of Evaluation and Decryption. For every $f \in \mathcal{F}$ with ℓ-bit inputs, every $x \in \{0,1\}^\ell$, the following holds:

$$\mathsf{Pr}\left[f(x) \leftarrow \mathsf{Dec}(\rho_{\mathsf{SK}}, \mathsf{CT}') \ : \ \begin{array}{c} (\mathsf{PK},\mathsf{MSK},\rho_{\mathsf{SK}}) \leftarrow \mathsf{KeyGen}(1^\lambda) \\ \mathsf{CT} \leftarrow \mathsf{Enc}(\mathsf{PK},x) \\ \mathsf{CT}' \leftarrow \mathsf{Eval}(\mathsf{PK},f,\mathsf{CT}) \end{array} \right] \geq 1 - \nu(\lambda),$$

where $\nu(\cdot)$ is a negligible function.

Correctness of Revocation. Defined as before.

Security. Defined as before (Definition 5.3).

5.3 From Single-Bit to Multi-Bit Security

Consider the following transformation from a single-bit key-revocable public-key encryption scheme to a multi-bit scheme. While such a transformation was known for indistinguishability-based encryption schemes, we show that the same transformation also works in the 1-bit unpredictability setting.

Construction 1 (Single-Bit to Multi-Bit Transformation)**.** *Let* $\Sigma = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Revoke})$ *be a single-bit key-revocable public-key encryption scheme. Then, for* $k \in \mathbb{N}$*, we define the corresponding multi-bit transformation* $\Sigma^k = \left(\mathsf{KeyGen}^k, \mathsf{Enc}^k, \mathsf{Dec}^k, \mathsf{Revoke}^k\right)$ *as follows:*

- $\mathsf{KeyGen}^k(1^\lambda)$*: given as input a security parameter* λ*, run* $\mathsf{KeyGen}(1^\lambda)$ *to output a public key* PK*, a master secret key* MSK *and a quantum decryption key* ρ_{SK}*.*
- $\mathsf{Enc}^k(\mathsf{PK}, x)$*: given a public key* PK *and plaintext* $x \in \{0,1\}^k$*, output the ciphertext*
$$\mathsf{CT} = (\mathsf{Enc}(\mathsf{PK}, x_1), \ldots, \mathsf{Enc}(\mathsf{PK}, x_k)).$$
- $\mathsf{Dec}^k(\rho_{\mathsf{SK}}, \mathsf{CT})$*: given a decryption key* ρ_{SK} *and a ciphertext* $\mathsf{CT} = \mathsf{CT}_1, \ldots, \mathsf{CT}_k$*, decrypt each of the ciphertexts separately by running the purified variant*[10] *of* Dec *and re-using the key.*
- $\mathsf{Revoke}^k\left(\mathsf{PK}, \mathsf{MSK}, \sigma\right)$*: given as input a master secret key* MSK*, a public key* PK *and quantum state* σ*, run* $\mathsf{Revoke}\left(\mathsf{PK}, \mathsf{MSK}, \sigma\right)$ *to output* Valid *or* $\mathsf{Invalid}$*.*

The following claim follows immediately from the "Almost As Good As New Lemma" (Lemma 2.2) mentioned in Remark 5.2.

Claim 5.7. *Let* $\lambda \in \mathbb{N}$ *be the security parameter. If* $\Sigma = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Revoke})$ *satisfies correctness of decryption and revocation, then so does* Σ^k *in Construction 1 for any* $k = \mathrm{poly}(\lambda)$*.*

Finally, we show the following.

Claim 5.8. *Let* $\lambda \in \mathbb{N}$ *be the security parameter. If* $\Sigma = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Revoke})$ *is a secure key-revocable public-key encryption scheme, then so is* Σ^k *in Construction 1 for any* $k = \mathrm{poly}(\lambda)$*.*

Proof. Let $\lambda \in \mathbb{N}$ and $k = \mathrm{poly}(\lambda)$. Let $\mathcal{A}$ be a QPT adversary and suppose that

$$\Pr\left[b \leftarrow \mathsf{Expt}_{\Sigma^k}^{\mathcal{A}}(1^\lambda, b) \ : \ b \xleftarrow{\$} \{0,1\}\right] = \frac{1}{2} + \epsilon(\lambda),$$

for some $\varepsilon(\lambda)$ with respect to $\mathsf{Expt}_{\Sigma^k}^{\mathcal{A}}(1^\lambda, b)$ in Figure 1. We show that $\varepsilon(\lambda)$ is negligible.

For $i \in [k]$, we now consider the following sequence of intermediate hybrid experiments $\mathsf{H}_i^{\mathcal{A}}$ defined in Figure 2, where $\mathsf{H}_1 = \mathsf{Expt}_{\Sigma^k}^{\mathcal{A}}(1^\lambda, 0)$ and $\mathsf{H}_k = \mathsf{Expt}_{\Sigma^k}^{\mathcal{A}}(1^\lambda, 1)$. Because the single-bit scheme Σ is secure, there exist negligible functions $\epsilon_i(\lambda)$ such that for each $i \in [k-1]$,

$$\frac{1}{2}\mathsf{Pr}[0 \leftarrow \mathsf{H}_i^{\mathcal{A}}(1^\lambda)] + \frac{1}{2}\mathsf{Pr}[1 \leftarrow \mathsf{H}_{i+1}^{\mathcal{A}}(1^\lambda)] = \frac{1}{2} + \epsilon_i(\lambda).$$

Using Lemma 5.5, we get that $\epsilon(\lambda) \leq \sum_{i=1}^{k-1} \epsilon_i(\lambda) \leq \mathsf{negl}(\lambda)$. This proves the claim. □

[10] See Remark 5.2.

$\underline{\mathsf{H}_i^{\mathcal{A}}\left(1^\lambda\right)}$:

Initialization Phase:

- The challenger runs $(\mathsf{PK}, \mathsf{MSK}, \rho_{\mathsf{SK}}) \leftarrow \mathsf{KeyGen}^k(1^\lambda)$ and sends $(\mathsf{PK}, \rho_{\mathsf{SK}})$ to $\mathcal{A}$.

Revocation Phase:

- The challenger sends the message `REVOKE` to $\mathcal{A}$.
- The adversary $\mathcal{A}$ returns a state σ.
- The challenger aborts if $\mathsf{Revoke}^k(\mathsf{PK}, \mathsf{MSK}, \sigma)$ outputs $\mathsf{Invalid}$.

Guessing Phase:

- $\mathcal{A}$ submits a plaintext $x \in \{0,1\}^k$ to the challenger.
- The challenger samples $C_1, \dots, C_{i-1} \xleftarrow{\$} \mathcal{C}$, where $\mathcal{C}$ is the space of all ciphertexts of single-bit messages, and sends the following to $\mathcal{A}$:
$$\mathsf{CT} = (C_1, C_2, \dots, C_{i-1}, \mathsf{Enc}(\mathsf{PK}, x_i), \dots \mathsf{Enc}(\mathsf{PK}, x_k)).$$
- Output $b_{\mathcal{A}}$ if the output of $\mathcal{A}$ is $b_{\mathcal{A}}$.

Figure 2: The hybrid experiment $\mathsf{H}_i^{\mathcal{A}}\left(1^\lambda\right)$.

6 Key-Revocable Dual-Regev Encryption

In this section, we present the first construction of key-revocable public-key encryption from standard assumptions. Our construction involves making the Dual Regev public-key encryption of Gentry, Peikert and Vaikuntanathan [GPV07] key revocable.

6.1 Construction

We define our Dual-Regev construction below.

Construction 2 (Key-Revocable Dual-Regev Encryption)**.** *Let $n \in \mathbb{N}$ be the security parameter and $m \in \mathbb{N}$. Let $q \geq 2$ be a prime and let $\alpha, \beta, \sigma > 0$ be parameters. The key-revocable public-key scheme* $\mathsf{RevDual} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Revoke})$ *consists of the following* QPT *algorithms:*

- $\mathsf{KeyGen}(1^\lambda) \to (\mathsf{PK}, \rho_{\mathsf{SK}}, \mathsf{MSK})$: *sample* $(\mathbf{A} \in \mathbb{Z}_q^{n \times m}, \mathsf{td}_{\mathbf{A}}) \leftarrow \mathsf{GenTrap}(1^n, 1^m, q)$ *and generate a Gaussian superposition* $(|\psi_{\mathbf{y}}\rangle, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma)$ *with*
$$|\psi_{\mathbf{y}}\rangle \;=\; \sum_{\substack{\mathbf{x} \in \mathbb{Z}_q^m \\ \mathbf{A}\mathbf{x} = \mathbf{y}}} \rho_\sigma(\mathbf{x}) \; |\mathbf{x}\rangle \, ,$$

for some $\mathbf{y} \in \mathbb{Z}_q^n$*. Output* $\mathsf{PK} = (\mathbf{A}, \mathbf{y})$*,* $\rho_{\mathsf{SK}} = |\psi_{\mathbf{y}}\rangle$ *and* $\mathsf{MSK} = \mathsf{td}_{\mathbf{A}}$*.*

- $\mathsf{Enc}(\mathsf{PK}, \mu) \to \mathsf{CT}$: *to encrypt a bit* $\mu \in \{0, 1\}$*, sample a random vector* $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n$ *and errors* $\mathbf{e} \sim D_{\mathbb{Z}^m, \alpha q}$ *and* $e' \sim D_{\mathbb{Z}, \beta q}$*, and output the ciphertext pair*

$$\mathsf{CT} = \Big(\mathbf{s}^\intercal \mathbf{A} + \mathbf{e}^\intercal \pmod q, \mathbf{s}^\intercal \mathbf{y} + e' + \mu \cdot \lfloor \frac{q}{2} \rfloor \pmod q\Big) \in \mathbb{Z}_q^m \times \mathbb{Z}_q.$$

- $\mathsf{Dec}(\rho_{\mathsf{SK}}, \mathsf{CT}) \to \{0, 1\}$: *to decrypt* CT*, apply the unitary* $U : |\mathbf{x}\rangle |0\rangle \to |\mathbf{x}\rangle |\mathsf{CT} \cdot (-\mathbf{x}, 1)^\intercal\rangle$ *on input* $|\psi_{\mathbf{y}}\rangle |0\rangle$*, where* $\rho_{\mathsf{SK}} = |\psi_{\mathbf{y}}\rangle$*, and measure the second register in the computational basis. Output* 0*, if the measurement outcome is closer to* 0 *than to* $\lfloor \frac{q}{2} \rfloor$*, and output* 1*, otherwise.*

- $\mathsf{Revoke}(\mathsf{MSK}, \mathsf{PK}, \rho) \to \{\top, \bot\}$*: on input* $\mathsf{td}_{\mathbf{A}} \leftarrow \mathsf{MSK}$ *and* $(\mathbf{A}, \mathbf{y}) \leftarrow \mathsf{PK}$*, apply the measurement* $\{|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|, I - |\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|\}$ *onto the state* ρ *using the procedure* $\mathsf{QSampGauss}(\mathbf{A}, \mathsf{td}_{\mathbf{A}}, \mathbf{y}, \sigma)$ *in Algorithm 2. Output* $\top$*, if the measurement is successful, and output* $\bot$ *otherwise.*

Correctness of Decryption. Follows from the correctness of Dual-Regev public-key encryption.

Correctness of Revocation. Follows from Theorem 3.3.

Let us now prove the security of our key-revocable Dual-Regev construction.

Theorem 6.1. *Let* $n \in \mathbb{N}$ *and* q *be a prime modulus with* $q = 2^{o(n)}$ *and* $m \geq 2n \log q$*, each parameterized by the security parameter* $\lambda \in \mathbb{N}$*. Let* $\sigma \in (\sqrt{2m}, q/\sqrt{2m})$ *and let* $\alpha, \beta \in (0, 1)$ *be noise ratios chosen such that* $\beta/\alpha = 2^{o(n)}$ *and* $1/\alpha = 2^{o(n)} \cdot \sigma$*. Then, assuming the subexponential hardness of the* $\mathsf{LWE}^m_{n,q,\alpha q}$ *and* $\mathsf{SIS}^m_{n,q,\sigma\sqrt{2m}}$ *problems, the scheme* $\mathsf{RevDual} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Revoke})$ *in Construction 2 is a secure key-revocable public-key encryption scheme according to Definition 5.3.*

Remark 6.2. *Note that our construction only handles 1-bit messages. However, we can apply the transformation in Definition 5.3 to obtain a key-revocable public-key encryption scheme for multi-bit messages.*

Guide for proving Theorem 6.1.

- The first step towards proving Theorem 6.1 is the simultaneous search-to-decision reduction (Theorem 6.8). Here, we show how to extract a short vector mapping $\mathbf{A}$ to $\mathbf{y}$ from an efficient adversary who has a non-negligible advantage in Definition 5.3.

- Next, we exploit the search-to-reduction to extract two distinct short vectors mapping $\mathbf{A}$ to $\mathbf{y}$. This is proven in Section 6.3.

- Finally, we put all the pieces together in Section 6.4 and show how to use the result from Section 6.3 in order to break the SIS assumption.

6.2 Simultaneous Search-to-Decision Reduction with Quantum Auxiliary Input

Our first result concerns distinguishers with quantum auxiliary input that can distinguish between Dual-Regev samples and uniformly random samples with high probability. In Theorem 6.3, we give a search-to-decision reduction: we show that such distinguishers can be converted into a quantum extractor that can obtain a Dual-Regev secret key with overwhelming probability. We then improve on the result and give a *simultaneous* search-to-decision reduction in Theorem 6.8 which holds even if additionally require that a *revocation* procedure succeeds on a separate register.

We first show the following result.

Theorem 6.3 (Search-to-Decision Reduction with Quantum Auxiliary Input)**.** *Let $n \in \mathbb{N}$ and q be a prime modulus with $q = 2^{o(n)}$ and let $m \geq 2n \log q$, each parameterized by the security parameter $\lambda \in \mathbb{N}$. Let $\sigma \in (\sqrt{2m}, q/\sqrt{2m})$ and let $\alpha, \beta \in (0,1)$ be noise ratios with $\beta/\alpha = 2^{o(n)}$ and $1/\alpha = 2^{o(n)} \cdot \sigma$. Let $\mathcal{A} = \{(\mathcal{A}_{\lambda,\mathbf{A}}, \nu_\lambda)\}_{\lambda \in \mathbb{N}}$ be any non-uniform quantum algorithm consisting of a family of polynomial-sized quantum circuits*

$$\left\{\mathcal{A}_{\lambda,\mathbf{A}} : \mathcal{L}(\mathcal{H}_q^m \otimes \mathcal{H}_{B_\lambda}) \rightarrow \mathcal{L}(\mathcal{H}_{R_\lambda} \otimes \mathcal{H}_{\textsc{Aux}_\lambda})\right\}_{\mathbf{A} \in \mathbb{Z}_q^{n \times m}}$$

and polynomial-sized advice states $\nu_\lambda \in \mathcal{D}(\mathcal{H}_{B_\lambda})$ which are independent of $\mathbf{A}$. Then, assuming the quantum hardness of the $\mathsf{LWE}_{n,q,\alpha q}^m$ assumption, the following holds for every QPT distinguisher $\mathcal{D}$. Suppose that there exists a function $\varepsilon(\lambda) = 1/\mathrm{poly}(\lambda)$ such that

$$\left|\mathsf{Pr}\left[1 \leftarrow \mathsf{SearchToDecisionExpt}^{\mathcal{A},\mathcal{D}}(1^\lambda, 0)\right] - \mathsf{Pr}\left[1 \leftarrow \mathsf{SearchToDecisionExpt}^{\mathcal{A},\mathcal{D}}(1^\lambda, 1)\right]\right| = \varepsilon(\lambda).$$

$\underline{\mathsf{SearchToDecisionExpt}^{\mathcal{A},\mathcal{D}}\left(1^\lambda, b\right)}$:

- If $b = 0$: output $\mathsf{lwe.Dist}^{\mathcal{A},\mathcal{D}}\left(1^\lambda\right)$ defined in Figure 4.
- If $b = 1$: output $\mathsf{unif.Dist}^{\mathcal{A},\mathcal{D}}\left(1^\lambda\right)$ defined in Figure 5.

Figure 3: The experiment $\mathsf{SearchToDecisionExpt}^{\mathcal{A},\mathcal{D}}\left(1^\lambda, b\right)$.

Then, there exists a quantum extractor $\mathcal{E}$ that takes as input $\mathbf{A}$, $\mathbf{y}$ and system $\textsc{Aux}$ of the state $\rho_{R,\textsc{Aux}}$ and outputs a short vector in the coset $\Lambda_q^{\mathbf{y}}(\mathbf{A})$ in time $\mathrm{poly}(\lambda, m, \sigma, q, 1/\varepsilon)$ such that

$$\Pr\left[\begin{array}{c} \mathcal{E}(\mathbf{A},\mathbf{y},\rho_{\textsc{Aux}})=\mathbf{x} \\ \bigwedge \\ \mathbf{x} \in \Lambda_q^{\mathbf{y}}(\mathbf{A}) \cap \mathcal{B}^m(\mathbf{0},\sigma\sqrt{\frac{m}{2}}) \end{array} \middle| \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m} \\ (|\psi_{\mathbf{y}}\rangle,\mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A},\sigma) \\ \rho_{R,\textsc{Aux}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}| \otimes \nu_\lambda) \end{array}\right] \geq 1 - \mathsf{negl}(\lambda).$$

Proof. Let $\lambda \in \mathbb{N}$ be the security parameter and let $\mathcal{A} = \{(\mathcal{A}_{\lambda,\mathbf{A}}, \nu_\lambda)\}_{\mathbf{A} \in \mathbb{Z}_q^{n \times m}}$ be a non-uniform quantum algorithm. Suppose that $\mathcal{D}$ is a QPT distinguisher with advantage $\varepsilon = 1/\mathrm{poly}(\lambda)$.

To prove the claim, we consider the following sequence of hybrid distributions.

H_0: This is the distribution $\mathsf{lwe.Dist}^{\mathcal{A},\mathcal{D}}\left(1^\lambda\right)$ in Figure 4.

H_1: This is the following distribution:

$\underline{\mathsf{lwe.Dist}^{\mathcal{A},\mathcal{D}}\left(1^{\lambda}\right)}$:

1. Sample $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$.
2. Generate $(|\psi_{\mathbf{y}}\rangle, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma)$.
3. Generate $\rho_{R,\textsc{Aux}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}| \otimes \nu_\lambda)$.
4. Sample $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n, \mathbf{e} \sim D_{\mathbb{Z}^m,\alpha q}$ and $e' \sim D_{\mathbb{Z},\beta q}$.
5. Generate $\rho_{R,\textsc{Aux}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}| \otimes \nu_\lambda)$.
6. Run $b' \leftarrow \mathcal{D}(\mathbf{A}, \mathbf{y}, \mathbf{s}^\intercal\mathbf{A} + \mathbf{e}^\intercal, \mathbf{s}^\intercal\mathbf{y} + e', \rho_{\textsc{Aux}})$ on the reduced state. Output b'.

Figure 4: The distribution $\mathsf{lwe.Dist}^{\mathcal{A},\mathcal{D}}\left(1^{\lambda}\right)$.

1. Sample a random matrix $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$.
2. Sample $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n$, $\mathbf{e} \sim D_{\mathbb{Z}^m,\alpha q}$ and $e' \sim D_{\mathbb{Z},\beta q}$.
3. Sample a Gaussian vector $\mathbf{x}_0 \sim D_{\mathbb{Z}_q^m,\frac{\sigma}{\sqrt{2}}}$ and let $\mathbf{y} = \mathbf{A} \cdot \mathbf{x}_0 \pmod{q}$.
4. Run $\mathcal{A}_{\lambda,\mathbf{A}}(|\mathbf{x}_0\rangle\langle\mathbf{x}_0| \otimes \nu_\lambda)$ to generate a state $\rho_{R,\textsc{Aux}}$ in systems R and $\textsc{Aux}$.
5. Run the distinguisher $\mathcal{D}(\mathbf{A}, \mathbf{y}, \mathbf{s}^\intercal\mathbf{A} + \mathbf{e}^\intercal, \mathbf{s}^\intercal\mathbf{y} + e', \rho_{\textsc{Aux}})$ on the reduced state $\rho_{\textsc{Aux}}$.

H_2 : This is the following distribution:

1. Sample a uniformly random matrix $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$.
2. Sample $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n$, $\mathbf{e} \sim D_{\mathbb{Z}^m,\alpha q}$ and $e' \sim D_{\mathbb{Z},\beta q}$. Let $\mathbf{u} = \mathbf{A}^\intercal\mathbf{s} + \mathbf{e}$.
3. Sample a Gaussian vector $\mathbf{x}_0 \sim D_{\mathbb{Z}_q^m,\frac{\sigma}{\sqrt{2}}}$ and let $\mathbf{y} = \mathbf{A} \cdot \mathbf{x}_0 \pmod{q}$.
4. Run $\mathcal{A}_{\lambda,\mathbf{A}}(|\mathbf{x}_0\rangle\langle\mathbf{x}_0| \otimes \nu_\lambda)$ to generate a state $\rho_{R,\textsc{Aux}}$ in systems R and $\textsc{Aux}$.
5. Run the distinguisher $\mathcal{D}(\mathbf{A}, \mathbf{y}, \mathbf{u}, \mathbf{u}^\intercal\mathbf{x}_0 + e', \rho_{\textsc{Aux}})$ on the reduced state $\rho_{\textsc{Aux}}$.

H_3 : This is the following distribution:

1. Sample a uniformly random matrix $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$.
2. Sample $\mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^m$ and $e' \sim D_{\mathbb{Z},\beta q}$.
3. Sample a Gaussian vector $\mathbf{x}_0 \sim D_{\mathbb{Z}_q^m,\frac{\sigma}{\sqrt{2}}}$ and let $\mathbf{y} = \mathbf{A} \cdot \mathbf{x}_0 \pmod{q}$.
4. Run $\mathcal{A}_{\lambda,\mathbf{A}}(|\mathbf{x}_0\rangle\langle\mathbf{x}_0| \otimes \nu_\lambda)$ to generate a state $\rho_{R,\textsc{Aux}}$ in systems R and $\textsc{Aux}$.
5. Run the distinguisher $\mathcal{D}(\mathbf{A}, \mathbf{y}, \mathbf{u}, \mathbf{u}^\intercal\mathbf{x}_0 + e', \rho_{\textsc{Aux}})$ on the reduced state $\rho_{\textsc{Aux}}$.

H_4: This is the following distribution:

1. Sample a uniformly random matrix $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$.
2. Sample $\mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^m$ and $r \xleftarrow{\$} \mathbb{Z}_q$.

3. Sample a Gaussian vector $\mathbf{x}_0 \sim D_{\mathbb{Z}_q^m, \frac{\sigma}{\sqrt{2}}}$ and let $\mathbf{y} = \mathbf{A} \cdot \mathbf{x}_0 \pmod{q}$.
4. Run $\mathcal{A}_{\lambda,\mathbf{A}}(|\mathbf{x}_0\rangle\langle\mathbf{x}_0| \otimes \nu_\lambda)$ to generate a state $\rho_{R,\textsc{aux}}$ in systems R and $\textsc{aux}$.
5. Run the distinguisher $\mathcal{D}(\mathbf{A}, \mathbf{y}, \mathbf{u}, r, \rho_{\textsc{aux}})$ on the reduced state $\rho_{\textsc{aux}}$.

H_5: This is the distribution $\mathsf{unif.Dist}^{\mathcal{A},\mathcal{D}}(1^\lambda)$ in Figure 5.

```
unif.Dist^{A,D}(1^λ):

1. Sample $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m}$.
2. Sample $\mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^m$ and $r \xleftarrow{\$} \mathbb{Z}_q$.
3. Run $(|\psi_{\mathbf{y}}\rangle, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma)$.
4. Generate $\rho_{R,\text{Aux}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}| \otimes \nu_\lambda)$.
5. Run $b' \leftarrow \mathcal{D}(\mathbf{A}, \mathbf{y}, \mathbf{u}, r, \rho_{\text{Aux}})$ on the reduced state. Output $b'$.
```

Figure 5: The distribution $\mathsf{unif.Dist}^{\mathcal{A},\mathcal{D}}(1^\lambda)$.

We now show the following:

Claim 6.4. *Assuming* $\mathsf{LWE}^m_{n,q,\alpha q}$, *the hybrids* H_0 *and* H_1 *are computationally indistinguishable,*

$$\mathsf{H}_0 \approx_c \mathsf{H}_1.$$

Proof. Here, we invoke the *Gaussian-collapsing property* in Theorem 3.1 which states that the following samples are indistinguishable under $\mathsf{LWE}^m_{n,q,\alpha q}$,

$$\left(\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m},\ |\psi_{\mathbf{y}}\rangle = \sum_{\substack{\mathbf{x}\in\mathbb{Z}_q^m \\ \mathbf{A}\mathbf{x}=\mathbf{y}}} \rho_\sigma(\mathbf{x})\,|\mathbf{x}\rangle,\ \mathbf{y} \in \mathbb{Z}_q^n\right) \approx_c \left(\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m},\ |\mathbf{x}_0\rangle,\ \mathbf{A}\cdot\mathbf{x}_0 \in \mathbb{Z}_q^n\right)$$

where $(|\psi_{\mathbf{y}}\rangle, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma)$ and where $\mathbf{x}_0 \sim D_{\mathbb{Z}_q^m, \frac{\sigma}{\sqrt{2}}}$ is a sample from the discrete Gaussian distribution. Because $\mathcal{A}_{\lambda,\mathbf{A}}$ is a family efficient quantum algorithms, this implies that

$$\mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}| \otimes \nu_\lambda) \quad \approx_c \quad \mathcal{A}_{\lambda,\mathbf{A}}(|\mathbf{x}_0\rangle\langle\mathbf{x}_0| \otimes \nu_\lambda),$$

for any polynomial-sized advice state $\nu_\lambda \in \mathcal{D}(\mathcal{H}_{B_\lambda})$ which is independent of $\mathbf{A}$. □

Claim 6.5. *Hybrids* H_1 *and* H_2 *are statistically indistinguishable. In other words,*

$$\mathsf{H}_1 \approx_s \mathsf{H}_2.$$

Proof. Here, we invoke the *noise flooding* property in Lemma 2.8 to argue that $\mathbf{e}^\intercal \mathbf{x}_0 \ll e'$ holds with overwhelming probability for our choice of parameters. Therefore, the distributions in H_1 and H_2 are computationally indistinguishable. □

Claim 6.6. *Assuming* $\mathsf{LWE}^m_{n,q,\alpha q}$*, the hybrids* H_2 *and* H_3 *are computationally indistinguishable,*

$$\mathsf{H}_2 \approx_c \mathsf{H}_3.$$

Proof. This follows from the $\mathsf{LWE}^m_{n,q,\alpha q}$ assumption since the reduction can sample $\mathbf{x}_0 \sim D_{\mathbb{Z}^m, \frac{\sigma}{\sqrt{2}}}$ itself and generate $\rho_{R,\text{AUX}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\mathbf{x}_0\rangle\langle\mathbf{x}_0| \otimes \nu_\lambda)$ on input $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and ν_λ. □

Finally, we show the following:

Claim 6.7. *Assuming* $\mathsf{LWE}^m_{n,q,\alpha q}$*, the hybrids* H_4 *and* H_5 *are computationally indistinguishable,*

$$\mathsf{H}_4 \approx_c \mathsf{H}_5.$$

Proof. Here, we invoke the *Gaussian-collapsing property* in Theorem 3.1 again. □

Recall that H_0 and H_5 can be distinguished with probability $\varepsilon = 1/\text{poly}(\lambda)$. We proved that the hybrids H_0 and H_3 are computationally indistinguishable and moreover, hybrids H_4 and H_5 are computationally indistinguishable. As a consequence, it holds that hybrids H_3 and H_4 can be distinguished with probability at least $\varepsilon - \mathsf{negl}(\lambda)$.

We leverage this to obtain a Goldreich-Levin reduction. Consider the following distinguisher.

$\underline{\tilde{\mathcal{D}}(\mathbf{A}, \mathbf{y}, \mathbf{u}, v, \rho)}$:

Input: $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, $\mathbf{y} \in \mathbb{Z}_q^n$, $\mathbf{u} \in \mathbb{Z}_q^n$, $v \in \mathbb{Z}_q$ and $\rho \in L(\mathcal{H}_{\text{AUX}})$.
Output: A bit $b' \in \{0,1\}$.

Procedure:

1. Sample $e' \sim D_{\mathbb{Z},\beta q}$.
2. Output $b' \leftarrow \mathcal{D}(\mathbf{A}, \mathbf{y}, \mathbf{u}, v + e', \rho)$.

Figure 6: The distinguisher $\tilde{\mathcal{D}}(\mathbf{A}, \mathbf{y}, \mathbf{u}, v, \rho)$.

Note that $r + e' \pmod q$ is uniform whenever $r \xleftarrow{\$} \mathbb{Z}_q$ and $e' \sim D_{\mathbb{Z},\beta q}$. Therefore, our previous argument shows that there exists a negligible function η such that:

$$\left| \Pr\left[\tilde{\mathcal{D}}(\mathbf{A}, \mathbf{y}, \mathbf{u}, \mathbf{u}^\intercal \mathbf{x}_0, \rho_{\text{AUX}}) = 1 \,\middle|\, \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m}, \mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^m \\ \mathbf{x}_0 \sim D_{\mathbb{Z}_q^m, \frac{\sigma}{\sqrt{2}}}, \mathbf{y} \leftarrow \mathbf{A}\cdot\mathbf{x}_0 \pmod q \\ \rho_{R\,\text{AUX}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\mathbf{x}_0\rangle\langle\mathbf{x}_0| \otimes \nu_\lambda) \end{array} \right] \right.$$
$$\left. - \Pr\left[\tilde{\mathcal{D}}(\mathbf{A}, \mathbf{y}, \mathbf{u}, \mathbf{r}, \rho_{\text{AUX}}) = 1 \,\middle|\, \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m} \\ \mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^m, r \xleftarrow{\$} \mathbb{Z}_q \\ \mathbf{x}_0 \sim D_{\mathbb{Z}_q^m, \frac{\sigma}{\sqrt{2}}}, \mathbf{y} \leftarrow \mathbf{A}\cdot\mathbf{x}_0 \pmod q \\ \rho_{R\,\text{AUX}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\mathbf{x}_0\rangle\langle\mathbf{x}_0| \otimes \nu_\lambda) \end{array} \right] \right| \geq \varepsilon - \eta(\lambda).$$

Using Theorem 4.6, it follows that there exists a quantum Goldreich-Levin extractor $\mathcal{E}$ running in time $T(\mathcal{E}) = \mathrm{poly}(\lambda, n, m, \sigma, q, 1/\varepsilon)$ that outputs a short vector in $\Lambda_q^{\mathbf{y}}(\mathbf{A})$ with probability at least

$$\Pr\left[\begin{array}{c} \mathcal{E}(\mathbf{A},\mathbf{y},\rho_{\mathrm{Aux}})=\mathbf{x} \\ \bigwedge \\ \mathbf{x} \in \Lambda_q^{\mathbf{y}}(\mathbf{A}) \cap \mathcal{B}^m(\mathbf{0},\sigma\sqrt{\frac{m}{2}}) \end{array} \quad : \quad \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m} \\ \mathbf{x}_0 \sim D_{\mathbb{Z}_q^m, \frac{\sigma}{\sqrt{2}}} \\ \mathbf{y} \leftarrow \mathbf{A}\mathbf{x}_0 \pmod{q} \\ \rho_{R\,\mathrm{AUX}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\mathbf{x}_0\rangle\langle\mathbf{x}_0| \otimes \nu_\lambda) \end{array}\right] \geq 1 - \exp(-\Omega(n)).$$

Assuming the $\mathsf{LWE}_{n,q,\alpha q}^m$ assumption, we can invoke the Gaussian-collapsing property in Theorem 3.1 once again which implies that the quantum extractor $\mathcal{E}$ satisfies

$$\Pr\left[\begin{array}{c} \mathcal{E}(\mathbf{A},\mathbf{y},\rho_{\mathrm{Aux}})=\mathbf{x} \\ \bigwedge \\ \mathbf{x} \in \Lambda_q^{\mathbf{y}}(\mathbf{A}) \cap \mathcal{B}^m(\mathbf{0},\sigma\sqrt{\frac{m}{2}}) \end{array} \quad : \quad \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m} \\ (|\psi_{\mathbf{y}}\rangle,\mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A},\sigma) \\ \rho_{R\,\mathrm{AUX}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}| \otimes \nu_\lambda) \end{array}\right] \geq 1 - \mathsf{negl}(\lambda).$$

This proves the claim. □

Next, we improve on the result in Theorem 6.3 and give a *simultaneous* search-to-decision reduction with quantum auxiliary input which holds even if additionally require that a *revocation* procedure succeeds on a separate register.

To formalize the notion that revocation is applied on a separate register, we introduce the following procedure called $\mathsf{IneffRevoke}$ which is defined below.

$\underline{\mathsf{IneffRevoke}(\mathbf{A}, \mathbf{y}, \sigma, \rho_R)}$:

Input: $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, $\mathbf{y} \in \mathbb{Z}_q^n$ and $\rho \in L(\mathcal{H}_R)$.
Output: Accept ($\top$) or reject ($\bot$).

Procedure:

1. Apply the (inefficient) projective measurement

$$\{|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|, I - |\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|\}$$

   where $|\psi_{\mathbf{y}}\rangle$ is the Gaussian coset state

$$|\psi_{\mathbf{y}}\rangle = \sum_{\substack{\mathbf{x}\in\mathbb{Z}_q^m: \\ \mathbf{A}\mathbf{x}=\mathbf{y} \pmod{q}}} \rho_\sigma(\mathbf{x})\,|\mathbf{x}\rangle\,.$$

2. If the measurement succeeds, output $\top$. Else, output $\bot$.

Figure 7: The procedure $\mathsf{IneffRevoke}(\mathbf{A}, \mathbf{y}, \sigma, \rho_R)$.

Finally, we prove the following theorem which constitutes the main technical result of this work.

Theorem 6.8 (Simultaneous Search-to-Decision Reduction with Quantum Auxiliary Input)**.** *Let $n \in \mathbb{N}$ and q be a prime modulus with $q = 2^{o(n)}$ and let $m \geq 2n \log q$, each parameterized by the security parameter $\lambda \in \mathbb{N}$. Let $\sigma \in (\sqrt{2m}, q/\sqrt{2m})$ and let $\alpha, \beta \in (0,1)$ be noise ratios with $\beta/\alpha = 2^{o(n)}$ and $1/\alpha = 2^{o(n)} \cdot \sigma$. Let $\mathcal{A} = \{(\mathcal{A}_{\lambda,\mathbf{A}}, \nu_\lambda)\}_{\lambda \in \mathbb{N}}$ be any non-uniform quantum algorithm consisting of a family of polynomial-sized quantum circuits*

$$\left\{\mathcal{A}_{\lambda,\mathbf{A}} : \mathcal{L}(\mathcal{H}_q^m \otimes \mathcal{H}_{B_\lambda}) \rightarrow \mathcal{L}(\mathcal{H}_{R_\lambda} \otimes \mathcal{H}_{\textsc{aux}_\lambda})\right\}_{\mathbf{A} \in \mathbb{Z}_q^{n \times m}}$$

and polynomial-sized advice states $\nu_\lambda \in \mathcal{D}(\mathcal{H}_{B_\lambda})$ which are independent of $\mathbf{A}$. Then, assuming the quantum hardness of the $\mathsf{LWE}^m_{n,q,\alpha q}$ assumption, the following holds for every QPT distinguisher $\mathcal{D}$. Suppose that there exists a function $\varepsilon(\lambda) = 1/\mathrm{poly}(\lambda)$ such that

$$\Pr\left[b \leftarrow \mathsf{SimultSearchToDecisionExpt}^{\mathcal{A},\mathcal{D}}(1^\lambda, b) \;:\; b \xleftarrow{\$} \{0,1\}\right] = \frac{1}{2} + \varepsilon(\lambda).$$

$\underline{\mathsf{SimultSearchToDecisionExpt}^{\mathcal{A},\mathcal{D}}\left(1^\lambda, b\right)}$:

- If $b = 0$: output $\mathsf{simult.lwe.Dist}^{\mathcal{A},\mathcal{D}}\left(1^\lambda\right)$ defined in Figure 9.
- If $b = 1$: output $\mathsf{simult.unif.Dist}^{\mathcal{A},\mathcal{D}}\left(1^\lambda\right)$ defined in Figure 10.

Figure 8: The experiment $\mathsf{SimultSearchToDecisionExpt}^{\mathcal{A},\mathcal{D}}\left(1^\lambda, b\right)$.

Then, there exists a quantum extractor $\mathcal{E}$ that takes as input $\mathbf{A}$, $\mathbf{y}$ and system $\textsc{Aux}$ *of the state $\rho_{R,\textsc{Aux}}$ and outputs a short vector in the coset $\Lambda_q^{\mathbf{y}}(\mathbf{A})$ in time $\mathrm{poly}(\lambda, m, \sigma, q, 1/\varepsilon)$ such that*

$$\Pr\left[\begin{array}{c} (\mathsf{IneffRevoke}(\mathbf{A},\mathbf{y},\sigma,\cdot) \otimes \mathcal{E}(\mathbf{A},\mathbf{y},\cdot))(\rho_{R,\textsc{Aux}}) = (\top, \mathbf{x}) \\ \bigwedge \\ \mathbf{x} \in \Lambda_q^{\mathbf{y}}(\mathbf{A}) \cap \mathcal{B}^m(\mathbf{0}, \sigma\sqrt{\frac{m}{2}}) \end{array} \;:\; \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m} \\ (|\psi_{\mathbf{y}}\rangle, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma) \\ \rho_{R,\textsc{Aux}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}| \otimes \nu_\lambda) \end{array}\right]$$

$$\geq \Pr\left[(\mathsf{IneffRevoke}(\mathbf{A}, \mathbf{y}, \sigma, \rho_R) = \top \;:\; \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m} \\ (|\psi_{\mathbf{y}}\rangle, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma) \\ \rho_{R,\textsc{Aux}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}| \otimes \nu_\lambda) \end{array}\right] - \mathsf{negl}(\lambda).$$

Proof. Let $\mathcal{A} = \{(\mathcal{A}_{\lambda,\mathbf{A}}, \nu_\lambda)\}_{\lambda \in \mathbb{N}}$ be a non-uniform quantum algorithm and let $\mathcal{D}$ be any QPT distinguisher. Let $\mathsf{simult.lwe.Dist}^{\mathcal{A},\mathcal{D}}$ and $\mathsf{simult.unif.Dist}^{\mathcal{A},\mathcal{D}}$ be the two distributions which are defined in Figure 9 and Figure 10, respectively.

By assumption, we have that there exists a function $\varepsilon(\lambda) = 1/\mathrm{poly}(\lambda)$ such that

$$\begin{aligned} &\Pr\left[b \leftarrow \mathsf{SimultSearchToDecisionExpt}^{\mathcal{A},\mathcal{D}}(1^\lambda, b) \;:\; b \xleftarrow{\$} \{0,1\}\right] \\ &= \frac{1}{2}\Pr[0 \leftarrow \mathsf{simult.lwe.Dist}^{\mathcal{A},\mathcal{D}}(1^\lambda)] + \frac{1}{2}\Pr[1 \leftarrow \mathsf{simult.unif.Dist}^{\mathcal{A},\mathcal{D}}(1^\lambda)] = \frac{1}{2} + \varepsilon(\lambda). \end{aligned}$$

```
$\underline{\mathsf{simult.lwe.Dist}^{\mathcal{A},\mathcal{D}}\left(1^{\lambda}\right)}$:

1. Sample $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$.
2. Generate $(|\psi_{\mathbf{y}}\rangle, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma)$.
3. Generate $\rho_{R,\,\textsc{Aux}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}| \otimes \nu_\lambda)$.
4. Sample $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n, \mathbf{e} \sim D_{\mathbb{Z}^m,\alpha q}$ and $e' \sim D_{\mathbb{Z},\beta q}$.
5. Generate $\rho_{R,\textsc{Aux}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}| \otimes \nu_\lambda)$.
6. Run $\mathsf{IneffRevoke}(\mathbf{A}, \mathbf{y}, \sigma, \cdot)$ on system $R$. If it outputs $\top$, continue. Otherwise,
   output $\mathsf{Invalid}$.
7. Run $b' \leftarrow \mathcal{D}(\mathbf{A}, \mathbf{y}, \mathbf{s}^\intercal \mathbf{A} + \mathbf{e}^\intercal, \mathbf{s}^\intercal \mathbf{y} + e', \cdot)$ on system $\textsc{Aux}$. Output $b'$.
```

Figure 9: The distribution $\mathsf{simult.lwe.Dist}^{\mathcal{A},\mathcal{D}}\left(1^{\lambda}\right)$.

```
$\underline{\mathsf{simult.unif.Dist}^{\mathcal{A},\mathcal{D}}\left(1^{\lambda}\right)}$:

1. Sample $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$.
2. Generate $(|\psi_{\mathbf{y}}\rangle, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma)$.
3. Sample $\mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^m$ and $r \xleftarrow{\$} \mathbb{Z}_q$.
4. Generate $\rho_{R,\textsc{Aux}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}| \otimes \nu_\lambda)$.
5. Run $\mathsf{IneffRevoke}(\mathbf{A}, \mathbf{y}, \sigma, \cdot)$ on system $R$. If it outputs $\top$, continue. Otherwise,
   output $\mathsf{Invalid}$.
6. Run $b' \leftarrow \mathcal{D}(\mathbf{A}, \mathbf{y}, \mathbf{u}, r, \cdot)$ on system $\textsc{Aux}$. Output $b'$.
```

Figure 10: The distribution $\mathsf{simult.unif.Dist}^{\mathcal{A},\mathcal{D}}\left(1^{\lambda}\right)$.

Recall that the distributions $\mathsf{lwe.Dist}$ (Figure 4) and $\mathsf{unif.Dist}$ (Figure 5) are the same as the distributions $\mathsf{simult.lwe.Dist}$ and $\mathsf{simult.Dist}$, except that the procedure $\mathsf{IneffRevoke}$ is not performed in the experiment; instead, the register R is simply traced out and $\mathcal{D}$ is run on the reduced state in system $\textsc{Aux}$.

Using the fact that dropping $\mathsf{IneffRevoke}$ can only increase the success probability, we get

$$\begin{aligned}&\frac{1}{2}\mathsf{Pr}[0 \leftarrow \mathsf{lwe.Dist}^{\mathcal{A},\mathcal{D}}(1^\lambda)] + \frac{1}{2}\mathsf{Pr}[1 \leftarrow \mathsf{unif.Dist}^{\mathcal{A},\mathcal{D}}(1^\lambda)]\\ \geq\; &\frac{1}{2}\mathsf{Pr}[0 \leftarrow \mathsf{simult.lwe.Dist}^{\mathcal{A},\mathcal{D}}(1^\lambda)] + \frac{1}{2}\mathsf{Pr}[1 \leftarrow \mathsf{simult.unif.Dist}^{\mathcal{A},\mathcal{D}}(1^\lambda)] \;=\; \frac{1}{2} + \varepsilon(\lambda).\end{aligned}$$

In other words, the QPT algorithm $\mathcal{D}$ can successfully predict whether it has received a Dual-Regev sample or a uniformly random sample. Therefore,[11] we can now invoke Theorem 6.3 to argue there exists a quantum extractor $\mathcal{E}$ that takes as input $\mathbf{A}$, $\mathbf{y}$ and system $\textsc{Aux}$ of the state $\rho_{R,\textsc{Aux}}$ and outputs a short vector in the coset $\Lambda_q^{\mathbf{y}}(\mathbf{A})$ in time $\mathrm{poly}(\lambda, m, \sigma, q, 1/\varepsilon)$ such that

$$\Pr\left[\begin{array}{c}\mathcal{E}(\mathbf{A},\mathbf{y},\rho_{\textsc{Aux}})=\mathbf{x}\\ \bigwedge\\ \mathbf{x}\in\Lambda_q^{\mathbf{y}}(\mathbf{A})\cap\mathcal{B}^m(\mathbf{0},\sigma\sqrt{\frac{m}{2}})\end{array}\middle|\begin{array}{c}\mathbf{A}\xleftarrow{\$}\mathbb{Z}_q^{n\times m}\\ (|\psi_{\mathbf{y}}\rangle,\mathbf{y})\leftarrow\mathsf{GenGauss}(\mathbf{A},\sigma)\\ \rho_{R,\textsc{Aux}}\leftarrow\mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|\otimes\nu_\lambda)\end{array}\right]\geq 1-\mathsf{negl}(\lambda).$$

By expanding the above probability in terms of conditional probabilities with respect to whether $\mathsf{IneffRevoke}$ succeeds (or fails), we get that

$$\begin{aligned}&\Pr\left[\begin{array}{c}(\mathsf{IneffRevoke}(\mathbf{A},\mathbf{y},\sigma,\cdot)\otimes\mathcal{E}(\mathbf{A},\mathbf{y},\cdot))(\rho_{R,\textsc{Aux}})=(\top,\mathbf{x})\\ \bigwedge\\ \mathbf{x}\in\Lambda_q^{\mathbf{y}}(\mathbf{A})\cap\mathcal{B}^m(\mathbf{0},\sigma\sqrt{\frac{m}{2}})\end{array} : \begin{array}{c}\mathbf{A}\xleftarrow{\$}\mathbb{Z}_q^{n\times m}\\ (|\psi_{\mathbf{y}}\rangle,\mathbf{y})\leftarrow\mathsf{GenGauss}(\mathbf{A},\sigma)\\ \rho_{R,\textsc{Aux}}\leftarrow\mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|\otimes\nu_\lambda)\end{array}\right]\\ &\geq \Pr\left[(\mathsf{IneffRevoke}(\mathbf{A},\mathbf{y},\sigma,\rho_R)=\top \;:\; \begin{array}{c}\mathbf{A}\xleftarrow{\$}\mathbb{Z}_q^{n\times m}\\ (|\psi_{\mathbf{y}}\rangle,\mathbf{y})\leftarrow\mathsf{GenGauss}(\mathbf{A},\sigma)\\ \rho_{R,\textsc{Aux}}\leftarrow\mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|\otimes\nu_\lambda)\end{array}\right]-\mathsf{negl}(\lambda).\end{aligned}$$

□

6.3 Distinct Pair Extraction

The following lemma allows us to analyze the probability of simultaneously extracting two distinct preimages in terms of the success probability of revocation and the success probability of extracting a preimage from the adversary's state.

Lemma 6.9 (Projection onto Distinct Pairs)**.** *Let $\rho \in \mathcal{D}(\mathcal{H}_X \otimes \mathcal{H}_Y)$ be an any density matrix, for some Hilbert spaces $\mathcal{H}_X$ and $\mathcal{H}_Y$. Let $|\psi\rangle = \sum_{x\in\mathcal{S}} \alpha_x |x\rangle \in \mathcal{H}_X$ be any state supported on a subset $\mathcal{S} \subseteq \mathcal{X}$, and let $\mathbf{\Pi} = |\psi\rangle\langle\psi|$ denote its associated projection. Let $\mathbf{\Pi}_{\mathcal{S}}$ be the projector onto $\mathcal{S}$ with*

$$\mathbf{\Pi}_{\mathcal{S}} = \sum_{x\in\mathcal{S}} |x\rangle\langle x|.$$

Let $\mathcal{E} : \mathcal{L}(\mathcal{H}_Y) \to \mathcal{L}(\mathcal{H}_{X'})$ be any CPTP map of the form

$$\mathcal{E}_{Y\to X'}(\sigma) = \mathrm{Tr}_E\left[V_{Y\to X'E}\,\sigma\,V^\dagger_{Y\to X'E}\right], \quad \forall\sigma\in\mathcal{D}(\mathcal{H}_Y),$$

for some unitary $V_{Y\to X'E}$. Consider the projector $\mathbf{\Gamma}$ given by

$$\mathbf{\Gamma} = \sum_{x,x'\in\mathcal{S}:x\neq x'} |x\rangle\langle x|_X \otimes V^\dagger_{Y\to X'E}(|x'\rangle\langle x'|_{X'} \otimes I_E)V_{Y\to X'E}.$$

[11] Here, we use the following fact: Suppose that D_0 and D_1 are two distributions. Then, any QPT algorithm can predict b when given a sample from D_b, where $b \xleftarrow{\$} \{0,1\}$, with probability $\frac{1}{2}+\frac{\varepsilon}{2}$ if and only if the algorithm can distinguish between the distributions D_0 and D_1 with probability ε.

Let $\rho_X = \mathrm{Tr}_Y[\rho_{XY}]$ denote the reduced state. Then, it holds that

$$\mathrm{Tr}[\mathbf{\Gamma}\rho] \geq \left(1 - \max_{x \in \mathcal{S}} |\alpha_x|^2\right) \cdot \mathrm{Tr}[\mathbf{\Pi}\rho_X] \cdot \mathrm{Tr}\left[\mathbf{\Pi}_{\mathcal{S}}\, \mathcal{E}_{Y \to X'}(\sigma)\right],$$

where $\sigma = \mathrm{Tr}[(\mathbf{\Pi} \otimes I)\rho]^{-1} \cdot \mathrm{Tr}_X[(\mathbf{\Pi} \otimes I)\rho]$ is a reduced state in system Y.

Proof. Because the order in which we apply $\mathbf{\Gamma}$ and $(\mathbf{\Pi} \otimes I)$ does not matter, we have the inequality

$$\mathrm{Tr}\left[\mathbf{\Gamma}\rho\right] \geq \mathrm{Tr}\left[(\mathbf{\Pi} \otimes I)\,\mathbf{\Gamma}\rho\right] = \mathrm{Tr}\left[\mathbf{\Gamma}(\mathbf{\Pi} \otimes I)\rho\right]. \tag{1}$$

Notice also that $(\mathbf{\Pi} \otimes I)\rho(\mathbf{\Pi} \otimes I)$ lies in the image of $(\mathbf{\Pi} \otimes I)$ with $\mathbf{\Pi} = |\psi\rangle\langle\psi|$, and thus

$$(\mathbf{\Pi} \otimes I)\rho(\mathbf{\Pi} \otimes I) = \mathrm{Tr}[(\mathbf{\Pi} \otimes I)\rho] \cdot (|\psi\rangle\langle\psi| \otimes \sigma), \tag{2}$$

for some $\sigma \in \mathcal{D}(\mathcal{H}_Y)$. Putting everything together, we get that

$$\begin{aligned}
\mathrm{Tr}\left[\mathbf{\Gamma}\rho\right] &\geq \mathrm{Tr}\left[\mathbf{\Gamma}(\mathbf{\Pi} \otimes I)\rho\right] && \text{(using inequality (1))}\\
&= \mathrm{Tr}\left[\mathbf{\Gamma}(\mathbf{\Pi} \otimes I)\rho(\mathbf{\Pi} \otimes I)\mathbf{\Gamma}\right] && \text{(since } \mathbf{\Gamma}(\mathbf{\Pi} \otimes I) \text{ is a projector)}\\
&= \mathrm{Tr}[(\mathbf{\Pi} \otimes I)\rho] \cdot \mathrm{Tr}\left[\mathbf{\Gamma}\left(|\psi\rangle\langle\psi| \otimes \sigma\right)\mathbf{\Gamma}\right] && \text{(using equation (2))}\\
&= \mathrm{Tr}[(\mathbf{\Pi} \otimes I)\rho] \cdot \mathrm{Tr}\left[\mathbf{\Gamma}\left(|\psi\rangle\langle\psi| \otimes \sigma\right)\right] && \text{(since } \mathbf{\Gamma} \text{ is a projector)}\\
&= \mathrm{Tr}[\mathbf{\Pi}\rho_X] \cdot \mathrm{Tr}\left[\sum_{x,x' \in \mathcal{S}: x \neq x'} |x\rangle\langle x|_X \otimes V^{\dagger}_{Y \to X'E}\left(|x'\rangle\langle x'|_{X'} \otimes I_E\right) V_{Y \to X'E}\left(|\psi\rangle\langle\psi| \otimes \sigma\right)\right]\\
&= \mathrm{Tr}[\mathbf{\Pi}\rho_X] \cdot \sum_{x' \in \mathcal{S}} \left(\sum_{x \in \mathcal{S}: x \neq x'} |\langle x|\psi\rangle|^2\right) \mathrm{Tr}\left[V^{\dagger}_{Y \to X'E}(|x'\rangle\langle x'|_{X'} \otimes I_E)V_{Y \to X'E}\,\sigma\right]\\
&= \mathrm{Tr}[\mathbf{\Pi}\rho_X] \cdot \sum_{x' \in \mathcal{S}} \left(1 - |\alpha_{x'}|^2\right) \mathrm{Tr}\left[(|x'\rangle\langle x'|_{X'} \otimes I_E)V_{Y \to X'E}\,\sigma\, V^{\dagger}_{Y \to X'E}\right]\\
&\geq \mathrm{Tr}[\mathbf{\Pi}\rho_X] \cdot \left(1 - \max_{x \in \mathcal{S}} |\alpha_x|^2\right) \cdot \sum_{x' \in \mathcal{S}} \mathrm{Tr}\left[(|x'\rangle\langle x'|_{X'} \otimes I_E)V_{Y \to X'E}\,\sigma\, V^{\dagger}_{Y \to X'E}\right]\\
&= \mathrm{Tr}[\mathbf{\Pi}\rho_X] \cdot \left(1 - \max_{x \in \mathcal{S}} |\alpha_x|^2\right) \cdot \sum_{x' \in \mathcal{S}} \mathrm{Tr}\left[|x'\rangle\langle x'|_{X'}\, \mathrm{Tr}_E\left[V_{Y \to X'E}\,\sigma\, V^{\dagger}_{Y \to X'E}\right]\right]\\
&= \mathrm{Tr}[\mathbf{\Pi}\rho_X] \cdot \left(1 - \max_{x \in \mathcal{S}} |\alpha_x|^2\right) \cdot \mathrm{Tr}\left[\mathbf{\Pi}_{\mathcal{S}}\, \mathcal{E}_{Y \to X'}(\sigma)\right].
\end{aligned}$$

This proves the claim. □

6.4 Proof of Theorem 6.1

Proof. Let $\mathcal{A}$ be a QPT adversary and suppose that

$$\Pr\left[b \leftarrow \mathsf{Expt}^{\mathcal{A}}(1^\lambda, b) \;:\; b \xleftarrow{\$} \{0,1\}\right] = \frac{1}{2} + \epsilon(\lambda),$$

for some $\varepsilon(\lambda)$ with respect to $\mathsf{Expt}^{\mathcal{A}}(1^\lambda, b)$ in Figure 11. We show that $\varepsilon(\lambda)$ is negligible.

$\underline{\mathsf{Expt}^{\mathcal{A}}(1^\lambda, b)}$:

1. The challenger samples $(\mathbf{A} \in \mathbb{Z}_q^{n\times m}, \mathsf{td}_{\mathbf{A}}) \leftarrow \mathsf{GenTrap}(1^n, 1^m, q)$ and generates
$$|\psi_{\mathbf{y}}\rangle \;=\; \sum_{\substack{\mathbf{x}\in\mathbb{Z}_q^m \\ \mathbf{A}\mathbf{x}=\mathbf{y} \pmod q}} \rho_\sigma(\mathbf{x})\; |\mathbf{x}\rangle\,,$$
for some $\mathbf{y} \in \mathbb{Z}_q^n$, by running $(|\psi_{\mathbf{y}}\rangle, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma)$. The challenger lets $\mathsf{MSK} \leftarrow \mathsf{td}_{\mathbf{A}}$ and $\mathsf{PK} \leftarrow (\mathbf{A}, \mathbf{y})$ and sends $\rho_{\mathsf{SK}} \leftarrow |\psi_{\mathbf{y}}\rangle$ to the adversary $\mathcal{A}$.
2. $\mathcal{A}$ generates a (possibly entangled) bipartite state $\rho_{R,\textsc{Aux}}$ in systems $\mathcal{H}_R \otimes \mathcal{H}_{\textsc{Aux}}$ with $\mathcal{H}_R = \mathcal{H}_q^m$, returns system R and holds onto the auxiliary system $\textsc{Aux}$.
3. The challenger runs $\mathsf{Revoke}(\mathsf{PK}, \mathsf{MSK}, \rho_R)$, where ρ_R is the reduced state in system R. If the outcome is $\top$, the game continues. Otherwise, output $\mathsf{Invalid}$.
4. $\mathcal{A}$ submits a plaintext bit $\mu \in \{0, 1\}$.
5. The challenger does the following depending on $b \in \{0, 1\}$:
   - if $b = 0$: the challenger samples a vector $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n$ and errors $\mathbf{e} \sim D_{\mathbb{Z}^m, \alpha q}$ and $e' \sim D_{\mathbb{Z}, \beta q}$, and sends a Dual-Regev encryption of $\mu \in \{0, 1\}$ to $\mathcal{A}$:
   $$\mathsf{CT} = \left(\mathbf{s}^\intercal \mathbf{A} + \mathbf{e}^\intercal, \mathbf{s}^\intercal \mathbf{y} + e' + \mu \cdot \lfloor \frac{q}{2} \rfloor\right) \in \mathbb{Z}_q^m \times \mathbb{Z}_q.$$
   - if $b = 1$: the challenger samples $\mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^m$ and $r \xleftarrow{\$} \mathbb{Z}_q$ uniformly at random and sends the following pair to $\mathcal{A}$:
   $$(\mathbf{u}, r) \in \mathbb{Z}_q^m \times \mathbb{Z}_q.$$
6. $\mathcal{A}$ returns a bit $b' \in \{0, 1\}$.

Figure 11: The key-revocable security experiment according to Definition 5.3.

Suppose for the sake of contradiction that $\epsilon(\lambda)$ is non-negligible. We show that we can use $\mathcal{A}$ to break the $\mathsf{SIS}^m_{n,q,\sigma\sqrt{2m}}$ problem. Without loss of generality, we assume that $\mathcal{A}$ submits the plaintext $x = 0$. By the assumption that $\epsilon(\lambda) \geq 1/\mathrm{poly}(\lambda)$, it follows from Theorem 6.8 that there exists a quantum Goldreich-Levin extractor $\mathcal{E}$ that takes as input $\mathbf{A}$, $\mathbf{y}$ and system $\textsc{Aux}$ of the state $\rho_{R,\textsc{Aux}}$ and outputs a short vector in the coset $\Lambda_q^{\mathbf{y}}(\mathbf{A})$ in time $\mathrm{poly}(\lambda, m, \sigma, q, 1/\varepsilon)$ such that

$$\Pr\left[\begin{array}{c}(\mathsf{Revoke}(\mathbf{A},\mathsf{td}_{\mathbf{A}},\mathbf{y},\cdot)\otimes\mathcal{E}(\mathbf{A},\mathbf{y},\cdot))(\rho_{R,\textsc{Aux}})=(\top,\mathbf{x})\\ \bigwedge \\ \mathbf{x}\in\Lambda_q^{\mathbf{y}}(\mathbf{A})\cap\mathcal{B}^m(\mathbf{0},\sigma\sqrt{\frac{m}{2}})\end{array} \ : \ \begin{array}{c}(\mathbf{A},\mathsf{td}_{\mathbf{A}})\leftarrow\mathsf{GenTrap}(1^n,1^m,q)\\ (|\psi_{\mathbf{y}}\rangle,\mathbf{y})\leftarrow\mathsf{GenGauss}(\mathbf{A},\sigma)\\ \rho_{R,\textsc{Aux}}\leftarrow\mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|\otimes\nu_\lambda)\end{array}\right] \geq 1/\mathrm{poly}(\lambda).$$

Here, we rely on the correctness of $\mathsf{GenTrap}$ in Theorem 2.12 and $\mathsf{QSampGauss}$ in Theorem 3.3, as well as the fact that revocation must necessarily succeed with inverse-polyomial probability. Consider the following procedure in Algorithm 4.

Algorithm 4: $\mathsf{SIS\_Solver}(\mathbf{A})$

```
Input: Matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$.
Output: Vector $\mathbf{x} \in \mathbb{Z}^m$.
Generate a Gaussian state $(|\psi_{\mathbf{y}}\rangle, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma)$ with
    $$|\psi_{\mathbf{y}}\rangle = \sum_{\substack{\mathbf{x}\in\mathbb{Z}_q^m \\ \mathbf{A}\mathbf{x}=\mathbf{y} \pmod q}} \rho_\sigma(\mathbf{x})\,|\mathbf{x}\rangle$$
  for some vector $\mathbf{y} \in \mathbb{Z}_q^n$.
Run $\mathcal{A}$ to generate a bipartite state $\rho_{R\,\textsc{Aux}}$ in systems $\mathcal{H}_R \otimes \mathcal{H}_{\textsc{Aux}}$ with $\mathcal{H}_R = \mathcal{H}_q^m$.
Measure system $R$ in the computational basis, and let $\mathbf{x}_0 \in \mathbb{Z}_q^n$ denote the outcome.
Run the quantum Goldreich-Levin extractor $\mathcal{E}(\mathbf{A}, \mathbf{y}, \rho_{\textsc{Aux}})$ from Theorem 6.8, where $\rho_{\textsc{Aux}}$ is
  the reduced state in system $\mathcal{H}_{\textsc{Aux}}$, and let $\mathbf{x}_1 \in \mathbb{Z}_q^n$ denote the outcome.
Output the vector $\mathbf{x} = \mathbf{x}_1 - \mathbf{x}_0$.
```

To conclude the proof, we show that $\mathsf{SIS\_Solver}(\mathbf{A})$ in Algorithm 4 breaks the $\mathsf{SIS}^m_{n,q,\sigma\sqrt{2m}}$ problem whenever $\varepsilon(\lambda) = 1/\mathrm{poly}(\lambda)$. In order to guarantee that $\mathsf{SIS\_Solver}(\mathbf{A})$ is successful, we use the distinct pair extraction result of Lemma 6.9. This allows us to analyze the probability of simultaneously extracting two distinct short pre-images $\mathbf{x}_0 \neq \mathbf{x}_1$ such that $\mathbf{A}\mathbf{x}_0 = \mathbf{y} = \mathbf{A}\mathbf{x}_1 \pmod q$ – both in terms of the success probability of revocation and the success probability of extracting a pre-image from the adversary's state $\rho_{\textsc{Aux}}$ in system $\mathcal{H}_{\textsc{Aux}}$. Assuming that $\mathbf{x}_0, \mathbf{x}_1$ are distinct short pre-images such that $\|\mathbf{x}_0\| \leq \sigma\sqrt{\frac{m}{2}}$ and $\|\mathbf{x}_1\| \leq \sigma\sqrt{\frac{m}{2}}$, it then follows that the vector $\mathbf{x} = \mathbf{x}_1 - \mathbf{x}_0$ output by $\mathsf{SIS\_Solver}(\mathbf{A})$ has norm at most $\sigma\sqrt{2m}$, and thus yields a solution to $\mathsf{SIS}^m_{n,q,\sigma\sqrt{2m}}$.

We remark that the state $|\psi_{\mathbf{y}}\rangle$ prepared by Algorithm 4 is not normalized for ease of notation. Note that the tail bound in Lemma 2.6 implies that (the normalized variant of) $|\psi_{\mathbf{y}}\rangle$ is within negligible trace distance of the state with support $\{\mathbf{x} \in \mathbb{Z}_q^m : \|\mathbf{x}\| \leq \sigma\sqrt{\frac{m}{2}}\}$. Therefore, for the sake of Lemma 6.9, we can assume that $|\psi_{\mathbf{y}}\rangle$ is a normalized state of the form

$$|\psi_{\mathbf{y}}\rangle = \left(\sum_{\substack{\mathbf{z} \in \mathbb{Z}_q^m, \|\mathbf{z}\| \leq \sigma\sqrt{\frac{m}{2}} \\ \mathbf{A}\mathbf{z} = \mathbf{y} \pmod{q}}} \rho_{\frac{\sigma}{\sqrt{2}}}(\mathbf{z}) \right)^{-\frac{1}{2}} \sum_{\substack{\mathbf{x} \in \mathbb{Z}_q^m, \|\mathbf{x}\| \leq \sigma\sqrt{\frac{m}{2}} \\ \mathbf{A}\mathbf{x} = \mathbf{y} \pmod{q}}} \rho_{\sigma}(\mathbf{x}) \, |\mathbf{x}\rangle \, .$$

Before we analyze Algorithm 4, we first make two technical remarks. First, since $\sigma \geq \omega(\sqrt{\log m})$, it follows from Lemma 2.9 that, for any full-rank $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and for any $\mathbf{y} \in \mathbb{Z}_q^n$, we have

$$\max_{\substack{\mathbf{x} \in \mathbb{Z}_q^m, \, \|\mathbf{x}\| \leq \sigma\sqrt{\frac{m}{2}} \\ \mathbf{A}\mathbf{x} = \mathbf{y} \pmod{q}}} \left\{ \frac{\rho_{\frac{\sigma}{\sqrt{2}}}(\mathbf{x})}{\sum_{\substack{\mathbf{z} \in \mathbb{Z}_q^m, \|\mathbf{z}\| \leq \sigma\sqrt{\frac{m}{2}} \\ \mathbf{A}\mathbf{z} = \mathbf{y} \pmod{q}}} \rho_{\frac{\sigma}{\sqrt{2}}}(\mathbf{z})} \right\} \leq \; 2^{-\Omega(m)}.$$

Second, we can replace the procedure $\mathsf{Revoke}(\mathbf{A}, \mathsf{td}_{\mathbf{A}}, \mathbf{y}, \rho_R)$ by an (inefficient) projective measurement $\{|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|, I - |\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|\}$, since they produce statistically close outcomes. This follows from the fact that $\mathsf{Revoke}(\mathbf{A}, \mathsf{td}_{\mathbf{A}}, \mathbf{y}, \rho_R)$ applies the procedure $\mathsf{QSampGauss}$ in Algorithm 2 as a subroutine, which is correct with overwhelming probability acccording to Theorem 3.3.

Let us now analyze the success probability of Algorithm 4. Putting everything together, we get

$$\begin{aligned}
&\mathsf{Pr}\left[\begin{array}{c} \mathbf{x} \leftarrow \mathsf{SIS\_Solver}(\mathbf{A}) \\ \wedge \\ \mathbf{x} \neq \mathbf{0} \text{ s.t. } \|\mathbf{x}\| \leq \sigma\sqrt{2m} \end{array} \; : \; \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m} \right] \\
&\geq \left(1 - \max_{\substack{\mathbf{x} \in \mathbb{Z}_q^m, \, \|\mathbf{x}\| \leq \sigma\sqrt{\frac{m}{2}} \\ \mathbf{A}\mathbf{x} = \mathbf{y} \pmod{q}}} \left\{ \frac{\rho_{\frac{\sigma}{\sqrt{2}}}(\mathbf{x})}{\sum_{\substack{\mathbf{z} \in \mathbb{Z}_q^m, \|\mathbf{z}\| \leq \sigma\sqrt{\frac{m}{2}} \\ \mathbf{A}\mathbf{z} = \mathbf{y} \pmod{q}}} \rho_{\frac{\sigma}{\sqrt{2}}}(\mathbf{z})} \right\} \right) \\
&\quad \cdot \Pr\left[\mathsf{IneffRevoke}(\mathbf{A}, \mathbf{y}, \rho_R) = \top \; : \; \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m} \text{ s.t. } \mathbf{A} \text{ is full-rank} \\ (|\psi_{\mathbf{y}}\rangle, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma) \\ \rho_{R,\textsc{Aux}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}| \otimes \nu_\lambda) \end{array} \right] \\
&\quad \cdot \Pr\left[\mathcal{E}\big(\mathbf{A}, \mathbf{y}, \rho_{\textsc{Aux}}\big) \;\in\; \Lambda_q^{\mathbf{y}}(\mathbf{A}) \cap \mathcal{B}^m(\mathbf{0}, \sigma\sqrt{m/2}) \; : \; \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m} \text{ s.t. } \mathbf{A} \text{ is full-rank} \\ (|\psi_{\mathbf{y}}\rangle, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma) \\ \rho_{R,\textsc{Aux}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}| \otimes \nu_\lambda) \\ \top \leftarrow \mathsf{IneffRevoke}(\mathbf{A}, \mathbf{y}, \rho_R) \end{array} \right] \\
&\geq \left(1 - 2^{-\Omega(m)}\right) \cdot \Pr\left[\begin{array}{c} (\mathsf{IneffRevoke}(\mathbf{A},\mathbf{y},\cdot) \otimes \mathcal{E}(\mathbf{A},\mathbf{y},\cdot))(\rho_{R\,\textsc{aux}}) = (\top, \mathbf{x}_1) \\ \wedge \\ \mathbf{x}_1 \in \in \Lambda_q^{\mathbf{y}}(\mathbf{A}) \cap \mathcal{B}^m(\mathbf{0}, \sigma\sqrt{\frac{m}{2}}) \end{array} \; : \; \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m} \\ \text{s.t. } \mathbf{A} \text{ is full-rank} \\ (|\psi_{\mathbf{y}}\rangle, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma) \\ \rho_{R,\textsc{Aux}} \leftarrow \mathcal{A}_{\lambda,\mathbf{A}}(|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}| \otimes \nu_\lambda) \end{array} \right] \\
&\geq \left(1 - 2^{-\Omega(m)}\right) \cdot \left(1/\mathrm{poly}(\lambda) - q^{-n}\right) \quad \geq \quad 1/\mathrm{poly}(\lambda).
\end{aligned}$$

In the last line, we applied the simultaneous search-to-decision reduction from Theorem 6.8 and Lemma 2.4. Therefore, $\mathsf{SIS\_Solver}(\mathbf{A})$ in Algorithm 4 runs in time $\mathrm{poly}(q, 1/\varepsilon)$ and solves $\mathsf{SIS}_{n,q,\sigma\sqrt{2m}}^m$ whenever $\varepsilon = 1/\mathrm{poly}(\lambda)$. Therefore, we conclude that $\varepsilon(\lambda)$ must be negligible. □

7 Key-Revocable Fully Homomorphic Encryption

In this section, we describe our key-revocable (leveled) fully homomorphic encryption scheme from LWE which is based on the so-called $\mathsf{DualGSW}$ scheme used by Mahadev [Mah18] which itself is a variant of the homomorphic encryption scheme by Gentry, Sahai and Waters [GSW13].

Let $\lambda \in \mathbb{N}$ be the security parameter. Suppose we would like to evaluate L-depth circuits consisting of NAND gates. We choose $n(\lambda, L) \gg L$ and a prime $q = 2^{o(n)}$. Then, for integer parameters $m \geq 2n \log q$ and $N = (m+1) \cdot \lceil \log q \rceil$, we let $\mathbf{I}$ be the $(m+1) \times (m+1)$ identity matrix and let $\mathbf{G} = [\mathbf{I} \,\|\, 2\mathbf{I} \,\|\, \dots \,\|\, 2^{\lceil \log q \rceil - 1}\mathbf{I}] \in \mathbb{Z}_q^{(m+1) \times N}$ denote the so-called *gadget matrix* which converts a binary representation of a vector back to its original vector representation over the field $\mathbb{Z}_q$. Note that the associated (non-linear) inverse operation $\mathbf{G}^{-1}$ converts vectors in $\mathbb{Z}_q^{m+1}$ to their binary representation in $\{0,1\}^N$. In other words, we have that $\mathbf{G} \circ \mathbf{G}^{-1}$ acts as the identity operator.

7.1 Construction

Construction 3 (Key-Revocable $\mathsf{DualGSW}$ encryption)**.** *Let $\lambda \in \mathbb{N}$ be the security parameter. The scheme* $\mathsf{RevDualGSW} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval}, \mathsf{Revoke})$ *consists of the following* QPT *algorithms:*

$\mathsf{KeyGen}(1^\lambda, 1^L) \to (\mathsf{PK}, \rho_{\mathsf{SK}})$: *sample a pair* $(\mathbf{A} \in \mathbb{Z}_q^{n \times m}, \mathsf{td}_{\mathbf{A}}) \leftarrow \mathsf{GenTrap}(1^n, 1^m, q)$ *and generate a Gaussian superposition* $(|\psi_{\mathbf{y}}\rangle, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma)$ *with*

$$|\psi_{\mathbf{y}}\rangle = \sum_{\substack{\mathbf{x} \in \mathbb{Z}_q^m \\ \mathbf{A}\mathbf{x} = \mathbf{y}}} \rho_\sigma(\mathbf{x}) \, |\mathbf{x}\rangle ,$$

for some $\mathbf{y} \in \mathbb{Z}_q^n$*. Output* $\mathsf{PK} = (\mathbf{A}, \mathbf{y})$*,* $\rho_{\mathsf{SK}} = |\psi_{\mathbf{y}}\rangle$ *and* $\mathsf{MSK} = \mathsf{td}_{\mathbf{A}}$*.*

$\mathsf{Enc}(\mathsf{PK}, \mu)$: *to encrypt* $\mu \in \{0,1\}$*, parse* $(\mathbf{A}, \mathbf{y}) \leftarrow \mathsf{PK}$*, sample a random matrix* $\mathbf{S} \xleftarrow{\$} \mathbb{Z}_q^{n \times N}$ *and* $\mathbf{E} \sim D_{\mathbb{Z}^{m \times N}, \alpha q}$ *and row vector* $\mathbf{e} \sim D_{\mathbb{Z}^N, \beta q}$*, and output the ciphertext*

$$\mathsf{CT} = \left[\frac{\mathbf{A}^\intercal \mathbf{S} + \mathbf{E}}{\mathbf{y}^\intercal \mathbf{S} + \mathbf{e}}\right] + \mu \cdot \mathbf{G} \pmod q \in \mathbb{Z}_q^{(m+1) \times N}.$$

$\mathsf{Eval}(\mathsf{CT}_0, \mathsf{CT}_1)$: *to apply a* NAND *gate on a ciphertext pair* CT_0 *and* CT_1*, output the matrix*

$$\mathbf{G} - \mathsf{CT}_0 \cdot \mathbf{G}^{-1}(\mathsf{CT}_1) \pmod q \in \mathbb{Z}_q^{(m+1) \times N}.$$

$\mathsf{Dec}(\rho_{\mathsf{SK}}, \mathsf{CT}) \to \{0,1\}$: *to decrypt* CT*, apply the unitary* $U : |\mathbf{x}\rangle |0\rangle \to |\mathbf{x}\rangle |(-\mathbf{x}, 1) \cdot \mathsf{CT}_N\rangle$ *on input* $|\psi_{\mathbf{y}}\rangle \leftarrow \rho_{\mathsf{SK}}$*, where* $\mathsf{CT}_N \in \mathbb{Z}_q^{m+1}$ *is the* N*-th column of* CT*, and measure the second register in the computational basis. Output* 0*, if the measurement outcome is closer to* 0 *than to* $\lfloor \frac{q}{2} \rfloor$*, and output* 1*, otherwise.*

$\mathsf{Revoke}(\mathsf{MSK}, \mathsf{PK}, \rho) \to \{\top, \bot\}$*: on input* $\mathsf{td}_{\mathbf{A}} \leftarrow \mathsf{MSK}$ *and* $(\mathbf{A}, \mathbf{y}) \leftarrow \mathsf{PK}$*, apply the projective measurement* $\{|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|, I - |\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|\}$ *onto* ρ *using* $\mathsf{SampGauss}(\mathbf{A}, \mathsf{td}_{\mathbf{A}}, \mathbf{y}, \sigma)$ *in Algorithm 2. Output* $\top$ *if the measurement is successful, and output* $\bot$ *otherwise.*

$\underline{\mathsf{Expt}^{\mathcal{A}}(1^\lambda, b)}$:

1. The challenger samples $(\mathbf{A} \in \mathbb{Z}_q^{n\times m}, \mathsf{td}_{\mathbf{A}}) \leftarrow \mathsf{GenTrap}(1^n, 1^m, q)$ and generates
$$|\psi_{\mathbf{y}}\rangle \;=\; \sum_{\substack{\mathbf{x}\in\mathbb{Z}_q^m \\ \mathbf{A}\mathbf{x}=\mathbf{y} \pmod q}} \rho_\sigma(\mathbf{x})\; |\mathbf{x}\rangle\,,$$
for some $\mathbf{y} \in \mathbb{Z}_q^n$, by running $(|\psi_{\mathbf{y}}\rangle, \mathbf{y}) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma)$. The challenger lets $\mathsf{MSK} \leftarrow \mathsf{td}_{\mathbf{A}}$ and $\mathsf{PK} \leftarrow (\mathbf{A}, \mathbf{y})$ and sends $\rho_{\mathsf{SK}} \leftarrow |\psi_{\mathbf{y}}\rangle$ to the adversary $\mathcal{A}$.
2. $\mathcal{A}$ generates a (possibly entangled) bipartite state $\rho_{R,\textsc{Aux}}$ in systems $\mathcal{H}_R \otimes \mathcal{H}_{\textsc{Aux}}$ with $\mathcal{H}_R = \mathcal{H}_q^m$, returns system R and holds onto the auxiliary system $\textsc{Aux}$.
3. The challenger runs $\mathsf{Revoke}(\mathsf{MSK}, \mathsf{PK}, \rho_R)$, where ρ_R is the reduced state in system R. If the outcome is $\top$, the game continues. Otherwise, output $\mathsf{Invalid}$.
4. $\mathcal{A}$ submits a plaintext bit $\mu \in \{0,1\}$.
5. The challenger does the following depending on $b \in \{0,1\}$:
   - if $b = 0$: The challenger samples a random matrix $\mathbf{S} \xleftarrow{\$} \mathbb{Z}_q^{n\times N}$ and errors $\mathbf{E} \sim D_{\mathbb{Z}^{m\times N}, \alpha q}$ and row vector $\mathbf{e} \sim D_{\mathbb{Z}^N, \beta q}$, and outputs the ciphertext
   $$\mathsf{CT} = \left[\frac{\mathbf{A}^\intercal\mathbf{S}+\mathbf{E}}{\mathbf{y}^\intercal\mathbf{S}+\mathbf{e}}\right] + \mu\cdot\mathbf{G} \;\in \mathbb{Z}_q^{(m+1)\times N}.$$
   - if $b = 1$: the challenger samples a matrix $\mathbf{U} \xleftarrow{\$} \mathbb{Z}_q^{m\times N}$ and row vector $r \xleftarrow{\$} \mathbb{Z}_q^N$ uniformly at random, and sends the following to $\mathcal{A}$:
   $$\left[\frac{\mathbf{U}}{\mathbf{r}}\right] \;\in \mathbb{Z}_q^{(m+1)\times N}.$$
6. $\mathcal{A}$ returns a bit $b' \in \{0,1\}$.

Figure 12: The key-revocable security experiment according to Definition 5.3.

7.2 Proof of Theorem 7.1

Theorem 7.1. *Let L be an upper bound on the* NAND*-depth of the circuit which is to be evaluated. Let $n \in \mathbb{N}$ and q be a prime modulus with $n = n(\lambda, L) \gg L$, $q = 2^{o(n)}$ and $m \geq 2n\log q$, each parameterized by the security parameter $\lambda \in \mathbb{N}$. Let $N = (m+1)\cdot\lceil\log q\rceil$ be an integer. Let $\sigma \in (\sqrt{2m}, q/\sqrt{2m})$ and let $\alpha, \beta \in (0,1)$ be parameters such that $\beta/\alpha = 2^{o(n)}$ and $1/\alpha = 2^{o(n)}\cdot\sigma$. Then, assuming the subexponential hardness of the* $\mathsf{LWE}^m_{n,q,\alpha q}$ *and* $\mathsf{SIS}^m_{n,q,\sigma\sqrt{2m}}$ *problems, the scheme* $\mathsf{RevDualGSW} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval}, \mathsf{Revoke})$ *in Construction 3 is a secure key-revocable (leveled)* *fully homomorphic encryption scheme according to Definition 5.3.*

Proof. Let $\mathcal{A}$ be a QPT adversary and suppose that

$$\Pr\left[b \leftarrow \mathsf{Expt}^{\mathcal{A}}(1^\lambda, b) \ : \ b \xleftarrow{\$} \{0,1\}\right] = \frac{1}{2} + \epsilon(\lambda),$$

for some $\varepsilon(\lambda)$ with respect to $\mathsf{Expt}^{\mathcal{A}}(1^\lambda, b)$ in Figure 12. Note that the $\mathsf{RevDualGSW}$ ciphertext can (up to an additive shift) be thought of as a column-wise concatenation of N-many independent ciphertexts of our key-revocable Dual-Regev scheme in Construction 2. Therefore, we can invoke Claim 5.8 and Theorem 6.1 in order to argue that $\varepsilon(\lambda)$ is at most negligible.

□

8 Revocable Pseudorandom Functions

In this section, we introduce the notion of *key-revocable* pseudorandom functions (or simply, called *revocable*) and present the first construction from (quantum hardness of) learning with errors.

8.1 Definition

Let us first recall the traditional notion of PRF security [GGM86], defined as follows.

Definition 8.1 (Pseudorandom Function). *Let $\lambda \in \mathbb{N}$ and $\kappa(\lambda), \ell(\lambda)$ and $\ell'(\lambda)$ be polynomials. A (post-quantum) pseudorandom function (pqPRF) is a pair ($\mathsf{Gen}, \mathsf{PRF}$) of PPT algorithms given by*

- $\mathsf{Gen}(1^\lambda)$: *On input 1^λ, it outputs a key $k \in \{0,1\}^\kappa$.*
- $\mathsf{PRF}(k, x)$: *On input $k \in \{0,1\}^\kappa$ and $x \in \{0,1\}^\ell$, it outputs a value $y \in \{0,1\}^{\ell'}$.*

with the property that, for any QPT distinguisher $\mathcal{D}$, we have

$$\left|\Pr\left[\mathcal{D}^{\mathsf{PRF}(k,\cdot)}(1^\lambda) = 1\right] \ : \ k \leftarrow \mathsf{Gen}(1^\lambda)\right] - \Pr\left[\mathcal{D}^{F(\cdot)}(1^\lambda) = 1\right] \ : \ F \xleftarrow{\$} \mathcal{F}^{\ell,\ell'}\right]\right| \leq \mathsf{negl}(\lambda),$$

where $\mathcal{F}^{\ell,\ell'}$ is the set of all functions with domain $\{0,1\}^\ell$ and range $\{0,1\}^{\ell'}$.

We now present a formal definition of revocable pseudorandom functions below.

Definition 8.2 (Revocable Pseudorandom Function). *Let $\lambda \in \mathbb{N}$ be the security parameter and let $\kappa(\lambda), \ell(\lambda)$ and $\ell'(\lambda)$ be polynomials. A revocable pseudorandom function (rPRF) is a scheme ($\mathsf{Gen}, \mathsf{PRF}, \mathsf{Eval}, \mathsf{Revoke}$) consisting of the following efficient algorithms:*

- $\mathsf{Gen}(1^\lambda)$*: on input the security parameter $\lambda \in \mathbb{N}$, it outputs a PRF key $k \in \{0,1\}^\kappa$, a quantum state ρ_k and a master secret key MSK.*
- $\mathsf{PRF}(k, x)$*: on input a key $k \in \{0,1\}^\kappa$ and an input string $x \in \{0,1\}^\ell$, it outputs a value $y \in \{0,1\}^{\ell'}$. This is a deterministic algorithm.*
- $\mathsf{Eval}(\rho_k, x)$*: on input a state ρ_k and an input $x \in \{0,1\}^\ell$, it outputs a value $y \in \{0,1\}^{\ell'}$.*
- $\mathsf{Revoke}(\mathsf{MSK}, \sigma)$*: on input key MSK and a state σ, it outputs Valid or $\mathsf{Invalid}$.*

We additionally require that the following holds:

Correctness. For each $(k, \rho_k, \mathsf{MSK})$ in the support of $\mathsf{Gen}(1^\lambda)$ and for every $x \in \{0,1\}^\ell$:

- (Correctness of evaluation:)
$$\Pr\left[\mathsf{PRF}(k,x) = \mathsf{Eval}(\rho_k, x)\right] \geq 1 - \mathsf{negl}(\lambda).$$

- (Correctness of revocation:)
$$\Pr\left[\mathsf{Valid} \leftarrow \mathsf{Revoke}(\mathsf{MSK}, \rho_k)\right] \geq 1 - \mathsf{negl}(\lambda).$$

8.2 Security

We define revocable PRF security below.

$\underline{\mathsf{Expt}^{\mathcal{A},\mu}(1^\lambda, b)}$:

Initialization Phase:

- The challenger computes $(k, \rho_k, \mathsf{MSK}) \leftarrow \mathsf{Gen}(1^\lambda)$ and sends ρ_k to $\mathcal{A}$.

Revocation Phase:

- The challenger sends the message `REVOKE` to $\mathcal{A}$.
- The adversary $\mathcal{A}$ sends a state σ to the challenger.
- The challenger aborts if $\mathsf{Revoke}(\mathsf{MSK}, \sigma)$ outputs $\mathsf{Invalid}$.

Guessing Phase:

- The challenger samples bit $b \leftarrow \{0,1\}$.
- The challenger samples random inputs $x_1, \ldots, x_\mu \xleftarrow{\$} \{0,1\}^\ell$ and then sends the values $(x_1, \ldots, x_\mu)$ and $(y_1, \ldots, y_\mu)$ to $\mathcal{A}$, where:
  - If $b = 0$, set $y_1 = \mathsf{PRF}(k, x_1)$, ..., $y_\mu = \mathsf{PRF}(k, x_\mu)$ and,
  - If $b = 1$, set $y_1, \ldots, y_\mu \xleftarrow{\$} \{0,1\}^{\ell'}$.
- $\mathcal{A}$ outputs a bit b' and wins if $b' = b$.

Figure 13: Revocable PRF security

Definition 8.3 (Revocable PRF Security). *A revocable pseudorandom function* (rPRF) *satisfies revocable* PRF *security if, for every QPT adversary $\mathcal{A}$ and every polynomial $\mu = \mu(\lambda) \in \mathbb{N}$,*
$$\Pr\left[b \leftarrow \mathsf{Expt}^{\mathcal{A},\mu}(1^\lambda, b) \ : \ b \xleftarrow{\$} \{0,1\}\right] \leq \frac{1}{2} + \mathsf{negl}(\lambda),$$
where $\mathsf{Expt}^{\mathcal{A},\mu}$ is as defined in Figure 13. If the above property holds for a fixed polynomial $\mu(\lambda)$, then we say that rPRF *satisfies μ-revocable* PRF *security.*

From one-query to multi-query security. We show that proving security with respect to $\mu = 1$ is sufficient. That is, we show the following.

Claim 8.4. *Supoose an* rPRF *scheme* $(\mathsf{Gen}, \mathsf{PRF}, \mathsf{Eval}, \mathsf{Revoke})$ *satisfies* 1*-revocable* PRF *security. Then,* rPRF *also satisfies the stronger notion of (multi-query) revocable PRF security.*

Proof. We consider a sequence of hybrids defined as follows. Let $\mathcal{A}$ be a QPT adversary participating in the revocable PRF security experiment and let $(x_1, y_1), \ldots, (x_\mu, y_\mu)$ denote the challenge input-output pairs, for some polynomial $\mu = \mu(\lambda)$. We also denote by k the PRF key sampled using Gen by the challenger in Figure 13.

H_i, for $i \in [\mu + 1]$: In this hybrid, $y_1, \ldots, y_{i-1}$ are sampled uniformly at random from $\{0,1\}^{\ell'}$ and $y_i, \ldots, y_\mu$ are generated as follows: $y_j = \mathsf{PRF}(k, x_j)$ for $j \geq i$.

We claim that $\mathcal{A}$ can win the 1-bit unpredictability game between hybrids H_i and H_{i+1}, for all $i \in [\mu]$, with probability $\frac{1}{2} + \mathsf{negl}(\lambda)$. That is, a bit b is sampled uniformly at random and if $b = 0$ then $\mathcal{A}$ participates in H_i and if $b = 1$ then $\mathcal{A}$ participates in H_{i+1}. We claim that $\mathcal{A}$ can predict b with probability $\frac{1}{2} + \mathsf{negl}(\lambda)$. Once we show this, we can then invoke the hybrid lemma for 1-bit unpredictability (Lemma 5.5) to complete the proof.

Suppose the above claim is not true. Let the prediction probability of $\mathcal{A}$ be $\frac{1}{2} + \varepsilon$, where ε is inverse polynomial. Then we use $\mathcal{A}$ to break 1-revocation security of rPRF. Specifically, we construct a reduction $\mathcal{B}$ that does the following:

- Get ρ_k from the challenger.
- Sample $x_{i+1}, \ldots, x_\mu$ uniformly at random from $\{0,1\}^\ell$. Denote $\rho_k^{(i+1)} = \rho_k$. Do the following for $j = i+1, \ldots, \mu$: $\mathsf{Eval}(\rho_k^{(j)}, x_j)$ to obtain y_j. Using Almost as good as new lemma [Aar16], recover $\rho_k^{(j+1)}$, where $\rho_k^{(j+1)}$ is negligibly[12] close to ρ_k in trace distance.
- Forward $\rho_k^{(\mu+1)}$ to $\mathcal{A}$.
- When the challenger sends the message $\texttt{REVOKE}$ then forward this message to $\mathcal{A}$.
- If $\mathcal{A}$ sends σ. Forward this to the challenger.
- If the revocation did not fail, the guessing phase begins. The challenger sends (x^*, y^*). Then, sample $x_1, \ldots, x_{i-1}$ uniformly at random from $\{0,1\}^\ell$ and $y_1, \ldots, y_{i-1}$ uniformly at random from $\{0,1\}^{\ell'}$. Set $x_i = x^*$ and $y_i = y^*$. Send $(x_1, y_1), \ldots, (x_\mu, y_\mu)$ to $\mathcal{A}$.
- Output b, where b is the output of $\mathcal{A}$.

From the quantum union bound Lemma 2.3, "Almost As Good As New" lemma (Lemma 2.2) and the correctness of rPRF, it follows that $\mathrm{TD}(\rho_k, \rho_k^{(\mu+1)}) \leq \mathsf{negl}(\lambda)$ and thus, the success probability of $\mathcal{A}$ when given $\rho_k^{(\mu+1)}$ instead of ρ_k is now at least $\frac{1}{2} + \varepsilon - \mathsf{negl}(\lambda)$. Moreover, by the design of $\mathcal{B}$, it follows that the success probability of $\mathcal{B}$ in breaking 1-revocation security of rPRF is exactly the

[12] Technically, this depends on the correctness error and we start with a rPRF that is correct with probability negligibly close to 1.

same as the success probability of $\mathcal{A}$ in breaking revocation security of rPRF. This contradicts the fact that rPRF satisfies 1-revocation security.

□

Remark 8.5. *As in the case of key revocable public-key encryption, we could consider an alternate definition defined with respect to computational indistinguishability: instead of requiring the adversary (in the guessing phase) to predict whether it receives a pseudorandom output or a string sampled uniformly at random, we could instead require the adversary to* **distinguish** *a pseudorandom sample from the uniform distribution. For a reason similar to the revocable PKE case, these two definitions are incomparable. We leave the investigation of the indistinguishability-based definition to the future works.*

Remark 8.6. *Our notion of revocable* PRF *security from Definition 8.3 does not directly imply traditional notion of* pqPRF *security*[13] *from Definition 8.1. The reason is that the definition does not preclude the possibility of there being an input x (say an all zeroes string) on which,* PRF *outputs x itself (or the first bit of x if the output of* PRF *is a single bit).*

Motivated by Remark 8.6, we now introduce the following notion of a *strong* rPRF.

Definition 8.7 (Strong rPRF)**.** *We say that a scheme* $(\mathsf{Gen}, \mathsf{PRF}, \mathsf{Eval}, \mathsf{Revoke})$ *is a strong revocable pseudorandom function (or, strong* rPRF*) if the following two properties hold:*

1. $(\mathsf{Gen}, \mathsf{PRF}, \mathsf{Eval}, \mathsf{Revoke})$ *satisfy revocable* PRF *security according to Definition 8.3, and*
2. $(\mathsf{Gen}, \mathsf{PRF})$ *satisfy* pqPRF *security according to Definition 8.1.*

Remark 8.8. *Instantiating pseudorandom functions in the textbook construction of private-key encryption [Gol06] from revocable pseudorandom functions, we get a private-key revocable encryption scheme.*

We show that the issue raised in Remark 8.6 is not inherent. In fact, we give a simple generic transformation that allows us to obtain strong rPRFs by making use of traditional pqPRFs.

Claim 8.9 (Generic Transformation for Strong rPRFs)**.** *Let* $(\mathsf{Gen}, \mathsf{PRF}, \mathsf{Eval}, \mathsf{Revoke})$ *be an* rPRF *scheme which satisfies revocable* PRF *security, and let* $(\overline{\mathsf{Gen}}, \overline{\mathsf{PRF}})$ *be a* pqPRF*. Then, the scheme* $(\widetilde{\mathsf{Gen}}, \widetilde{\mathsf{PRF}}, \widetilde{\mathsf{Eval}}, \widetilde{\mathsf{Revoke}})$ *is a strong* rPRF *which consists of the following algorithms:*

- $\widetilde{\mathsf{Gen}}(1^\lambda)$*: on input the security parameter* 1^λ*, first run* $(k, \rho_k, \mathsf{MSK}) \leftarrow \mathsf{Gen}(1^\lambda)$ *and then output* $((K, k), (K, \rho_k), \mathsf{MSK})$*, where* $K \leftarrow \overline{\mathsf{Gen}}(1^\lambda)$ *is a* pqPRF *key.*
- $\widetilde{\mathsf{PRF}}((K, k), x)$*: on input a key* (K, k) *and string* $x \in \{0,1\}^\ell$*, output* $\overline{\mathsf{PRF}}(K, x) \oplus \mathsf{PRF}(k, x)$*.*
- $\widetilde{\mathsf{Eval}}((K, \rho_k), x)$*: on input* (K, ρ_k) *and* $x \in \{0,1\}^\ell$*, output* $\overline{\mathsf{PRF}}(K, x) \oplus \mathsf{Eval}(\rho_k, x)$*.*
- $\widetilde{\mathsf{Revoke}}(\mathsf{MSK}, (K, \sigma))$*: on input a master secret key* MSK *and a pair* (K, ρ_k)*, first discard the key K and then run* $\mathsf{Revoke}(\mathsf{MSK}, \sigma)$*.*

[13] Although any revocable PRF is a *weak* PRF. Recall that a weak PRF is one where the adversary receives as input $(x_1, y_1), \ldots, (x_\mu, y_\mu)$, where x_is are picked uniformly at random. The goal of the adversary is to distinguish the two cases: all y_is are pseudorandom or all y_is are picked uniformly at random.

Proof. Let us first show that the scheme $(\widetilde{\mathsf{Gen}}, \widetilde{\mathsf{PRF}}, \widetilde{\mathsf{Eval}}, \widetilde{\mathsf{Revoke}})$ maintains revocable PRF security. Suppose that there exists a QPT adversary $\mathcal{A}$ and a polynomial $\mu = \mu(\lambda) \in \mathbb{N}$ such that

$$\Pr\left[b \leftarrow \mathsf{Expt}^{\mathcal{A},\mu}(1^\lambda, b) \ : \ b \xleftarrow{\$} \{0,1\}\right] = \frac{1}{2} + \epsilon(\lambda),$$

for some function $\epsilon(\lambda) = 1/\text{poly}(\lambda)$ and $\mathsf{Expt}^{\mathcal{A},\mu}$ as defined in Figure 13. We show that this implies the existence of a QPT distinguisher $\mathcal{D}$ that breaks the revocable PRF security of the scheme $(\mathsf{Gen}, \mathsf{PRF}, \mathsf{Eval}, \mathsf{Revoke})$. The distinguisher $\mathcal{D}$ proceeds as follows:

1. $\mathcal{D}$ receives as input a quantum state ρ_k, where $(k, \rho_k, \mathsf{MSK}) \leftarrow \mathsf{Gen}(1^\lambda)$ is generated by the challenger. Then, $\mathcal{D}$ generates a pqPRF key $K \leftarrow \overline{\mathsf{Gen}}(1^\lambda)$ and sends (K, ρ_k) to $\mathcal{A}$.

2. When $\mathcal{A}$ returns a state ρ, $\mathcal{D}$ forwards it to the challenger as part of the revocation phase.

3. When $\mathcal{D}$ receives the challenge input $(x_1, \dots, x_\mu)$ and $(y_1, \dots, y_\mu)$ from the challenger, $\mathcal{D}$ sends $(x_1, \dots, x_\mu)$ and $(\overline{\mathsf{PRF}}(K, x_1) \oplus y_1, \dots, \overline{\mathsf{PRF}}(K, x_\mu) \oplus y_\mu)$ to $\mathcal{A}$.

4. When $\mathcal{A}$ outputs b', so does the distinguisher $\mathcal{D}$.

Note that the simulated challenge distribution above precisely matches the challenge distribution from the experiment $\mathsf{Expt}^{\mathcal{A},\mu}$ from Figure 13. Therefore, if $\mathcal{A}$ succeeds with inverse polynomial advantage $\epsilon(\lambda) = 1/\text{poly}(\lambda)$, so does $\mathcal{D}$ – thereby breaking the revocable PRF security of the scheme $(\mathsf{Gen}, \mathsf{PRF}, \mathsf{Eval}, \mathsf{Revoke})$. Consequently, $(\widetilde{\mathsf{Gen}}, \widetilde{\mathsf{PRF}}, \widetilde{\mathsf{Eval}}, \widetilde{\mathsf{Revoke}})$ satisfies revocable PRF security.

To see why $(\widetilde{\mathsf{Gen}}, \widetilde{\mathsf{PRF}})$ satisfy pqPRF security according to Definition 8.1, we can follow a similar argument as above to break the pqPRF security of $(\overline{\mathsf{Gen}}, \overline{\mathsf{PRF}})$. Here, we rely on the fact that the keys $(k, \rho_k, \mathsf{MSK}) \leftarrow \mathsf{Gen}(1^\lambda)$ and $K \leftarrow \overline{\mathsf{Gen}}(1^\lambda)$ are sampled independently from another.

□

Remark 8.10. *We note that previous works [CLLZ21, KNY21b] do not explicitly require in their definitions that either secure software leasing or copy-protection of pseudorandom functions to preserve the pseudorandomness property (although their constructions could still satisfy the traditional pseudorandomness property).*

8.3 Construction

We construct a PRF satisfying 1-revocation security (Definition 8.3).

Shift-Hiding Construction. We construct a *shift-hiding* function which is loosely inspired by shift-hiding shiftable functions introduced by Peikert and Shiehian [PS18].

Let $n, m \in \mathbb{N}$, $q \in \mathbb{N}$ be a modulus and let $\ell = nm\lceil \log q \rceil$. In the following, we consider matrix-valued functions $F : \{0,1\}^\ell \to \mathbb{Z}_q^{n \times m}$, where F is one of the following functions:

- $\mathcal{Z} : \{0,1\}^\ell \to \mathbb{Z}_q^{n \times m}$ which, on input $x \in \{0,1\}^\ell$, outputs an all zeroes matrix $\mathbf{0} \in \mathbb{Z}_q^{n \times m}$, or:

- $H_r : \{0,1\}^\ell \to \mathbb{Z}_q^{n \times m}$ which, on input $x \in \{0,1\}^\ell$, outputs $\mathbf{M} \in \mathbb{Z}_q^{n \times m}$, where $r \in \{0,1\}^\ell$ and $x = r \oplus \mathsf{bindecomp}(\mathbf{M})$, where $\mathbf{M} \in \mathbb{Z}_q^{n \times m}$ and $\mathsf{bindecomp}(\cdot)$ takes as input a matrix and outputs a binary string that is obtained by concatenating the binary decompositions of all the elements in the matrix (in some order).

We show that there exist PPT algorithms $(\mathcal{KG}, \mathcal{E})$ (formally defined in Construction 4) with the following properties:

- $\mathcal{KG}(1^n, 1^m, q, \mathbf{A}, F)$: on input $1^n, 1^m$, a modulus $q \in \mathbb{N}$, a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and a function $F \in \{\mathcal{Z}\} \cup \{H_r : r \in \{0,1\}^\ell\}$, it outputs a pair of keys (pk_F, sk_F).
- $\mathcal{E}(pk_F, x)$: on input pk_F, $x \in \{0,1\}^\ell$, it outputs $\mathbf{S}_x\mathbf{A} + \mathbf{E}_x + F(x)$, where $\mathbf{S}_x \in \mathbb{Z}_q^{n \times n}$ and $\mathbf{E}_x \in \mathbb{Z}_q^{n \times m}$, where $||\mathbf{E}_x||_\infty \leq (m\sigma)^2 \cdot (nm\lceil\log(q)\rceil)$. Moreover, there is an efficient algorithm that recovers $\mathbf{S}_x$ given sk_F and x.

We show that our construction of $(\mathcal{KG}, \mathcal{E})$ satisfies a *shift-hiding property*; namely, for any $r \in \{0,1\}^\ell$,

$$\{pk_{\mathcal{Z}}\} \approx_c \{pk_{H_r}\},$$

for any pk_F with $(pk_F, sk_F) \leftarrow \mathcal{KG}(1^n, 1^m, q, \mathbf{A}, F)$, where $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$, and $F \in \{\mathcal{Z}, H_r\}$.

In the construction below, we consider a bijective function $\phi : [n] \times [m] \times [\lceil\log(q)\rceil] \to [\ell]$.

Construction 4. *Consider the* PPT *algorithms* $(\mathcal{KG}, \mathcal{E})$ *defined as follows:*

- $\mathcal{KG}(1^n, 1^m, q, \mathbf{A}, F)$*: on input* $1^n, 1^m$*, a modulus* $q \in \mathbb{N}$*, a matrix* $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ *and function* $F \in \{\mathcal{Z}\} \cup \{H_r : r \in \{0,1\}^\ell\}$*, it outputs a pair of keys* $\kappa_F = (pk_F, sk_F)$ *generated as follows:*

  1. *For every* $i, j \in [n], \tau \in [\lceil\log(q)\rceil]$*, define* $\{\mathsf{M}_b^{(i,j,\tau)}\}$ *as follows:*
     - *If* $F = \mathcal{Z}$*, then for every* $i \in [n], j \in [m], \tau \in [\lceil\log(q)\rceil]$*, let* $\mathsf{M}_b^{(i,j,\tau)} = \mathbf{0} \in \mathbb{Z}_q^{n \times n}$*,*
     - *If* $F = H_r$*, then for every* $i \in [n], j \in [m], \tau \in [\lceil\log(q)\rceil]$*, let* $\mathsf{M}_b^{(i,j,\tau)} = (b \oplus r_{\phi(i,j,\tau)}) \cdot \mathbf{I}_{n \times n}$*.*
  2. *For every* $i \in [n], j \in [m], \tau \in [\lceil\log(q)\rceil]$*,* $b \in \{0,1\}$*, compute:*

     $$pk_b^{(i,j,\tau)} = \mathbf{S}_b^{(i,j,\tau)}\mathbf{A} + \mathbf{E}_b^{(i,j,\tau)} + \mathsf{M}_b^{(i,j,\tau)},$$

     $$sk_b^{(i,j,\tau)} = \left(\left\{\mathbf{S}_b^{(i,j,\tau)}, \mathbf{E}_b^{(i,j,\tau)}\right\}\right),$$

     where for every $i \in [n], j \in [m], \tau \in [\lceil\log(q)\rceil]$*,* $b \in \{0,1\}$*:*
     - $\mathbf{S}_b^{(i,j,\tau)} \leftarrow D_{\mathbb{Z}_q,\sigma}^{n \times n}$*,*
     - $\mathbf{E}_b^{(i,j,\tau)} \leftarrow D_{\mathbb{Z}_q,\sigma}^{n \times m}$
  3. *Output* $pk_F = \left(\mathbf{A}, \left\{pk_b^{(i,j,\tau)}\right\}_{\substack{i\in[n],j\in[m],\\ \tau\in[\lceil\log(q)\rceil],b\in\{0,1\}}}\right)$ *and* $sk_F = \left\{sk_b^{(i,j,\tau)}\right\}_{\substack{i\in[n],j\in[m],\\ \tau\in[\lceil\log(q)\rceil],b\in\{0,1\}}}$*.*

- $\mathcal{E}(pk_F, x)$*: on input* pk_F *and* $x \in \{0,1\}^\ell$*, proceed as follows:*

  1. *Parse* $pk_F = \left(\mathbf{A}\left\{pk_b^{(i,j,\tau)}\right\}_{\substack{i\in[n],j\in[m],\\ \tau\in[\lceil\log(q)\rceil],b\in\{0,1\}}}\right)$
  2. *Output* $\sum_{\substack{i\in[n],j\in[m],\\ \tau\in[\lceil\log(q)\rceil]}} pk_{x_{\phi(i,j,\tau)}}^{(i,j,\tau)}$*.*

Claim 8.11 (Correctness). *Let $(\mathcal{KG}, \mathcal{E})$ be the pair of* PPT *algorithms in Construction 4. Let $(pk_F, sk_F) \leftarrow \mathcal{KG}(1^n, 1^m, q, \mathbf{A}, F)$ with $F \in \{\mathcal{Z}\} \cup \{H_r : r \in \{0,1\}^\ell\}$. Then, the output of $\mathcal{E}(pk_F, x)$ is of the form:*

$$\mathcal{E}(pk_F, x) = \mathbf{S}_x\mathbf{A} + \mathbf{E}_x + F(x),$$

where $\mathbf{S}_x \in \mathbb{Z}_q^{n\times n}$ and $\mathbf{E}_x \in \mathbb{Z}_q^{n\times m}$ with $||\mathbf{E}_x||_\infty \leq (m\sigma)^2 \cdot (nm\lceil\log(q)\rceil)$. Moreover, there is an efficient algorithm that recovers $\mathbf{S}_x$ given (pk_F, sk_F).

Proof. Let $(pk_F, sk_F) \leftarrow \mathcal{KG}(1^n, 1^m, q, \mathbf{A}, F)$. Parse $pk_F = \left(\mathbf{A}, \left\{pk_b^{(i,j,\tau)}\right\}_{\substack{i\in[n],j\in[m],\\ \tau\in[\lceil\log(q)\rceil],b\in\{0,1\}}}\right)$ and $sk_F = \left\{sk_b^{(i,j,\tau)}\right\}_{\substack{i\in[n],j\in[m],\\ \tau\in[\lceil\log(q)\rceil],b\in\{0,1\}}}$, where:

$$pk_b^{(i,j,\tau)} = \mathbf{S}_b^{(i,j,\tau)}\mathbf{A} + \mathbf{E}_b^{(i,j,\tau)} + \mathsf{M}_b^{(i,j,\tau)},$$

$$sk_b^{(i,j,\tau)} = \left(\{\mathbf{S}_b^{(i,j,\tau)}, \mathbf{E}_b^{(i,j,\tau)}\}\right)$$

There are two cases to consider here:

Case 1. $F = \mathcal{Z}$: in this case, $\mathsf{M}_b^{(i,j,\tau)} = \mathbf{0}$, for every $i \in [n], j \in [m], \tau \in [\lceil\log(q)\rceil], b \in \{0,1\}$. Thus, the following holds:

$$\begin{aligned}\sum_{\substack{i\in[n],j\in[m],\\ \tau\in[\lceil\log(q)\rceil]}} pk_{x_{\phi(i,j,\tau)}}^{(i,j,\tau)} &= \underbrace{\left(\sum_{\substack{i\in[n],j\in[m],\\ \tau\in[\lceil\log(q)\rceil]}} \mathbf{S}_{x_{\phi(i,j,\tau)}}^{(i,j,\tau)}\right)}_{\mathbf{S}_x}\mathbf{A} + \underbrace{\left(\sum_{\substack{i\in[n],j\in[m],\\ \tau\in[\lceil\log(q)\rceil]}} \mathbf{E}_{x_{\phi(i,j,\tau)}}^{(i,j,\tau)}\right)}_{\mathbf{E}_x} + \left(\sum_{\substack{i\in[n],j\in[m],\\ \tau\in[\lceil\log(q)\rceil]}} \mathsf{M}_{x_{\phi(i,j,\tau)}}^{(i,j,\tau)}\right)\\ &= \mathbf{S}_x\mathbf{A} + \mathbf{E}_x + \mathcal{Z}(x)\end{aligned}$$

Moreover, $||\mathbf{E}_b^{(i,j,\tau)}||_\infty \leq (m\sigma)^2$ and thus, $||\mathbf{E}_x||_\infty \leq (m\sigma)^2 \cdot (nm\lceil\log(q)\rceil)$.

Case 2. $F = H_r$:

$$\begin{aligned}\sum_{\substack{i\in[n],j\in[m],\\ \tau\in[\lceil\log(q)\rceil]}} pk_{x_{\phi(i,j,\tau)}}^{(i,j,\tau)} &= \mathbf{S}_x\mathbf{A} + \mathbf{E}_x + \left(\sum_{\substack{i\in[n],j\in[m],\\ \tau\in[\lceil\log(q)\rceil]}} \mathsf{M}_{x_{\phi(i,j,\tau)}}^{(i,j,\tau)}\right)\\ &= \mathbf{S}_x\mathbf{A} + \mathbf{E}_x + H_r(x),\end{aligned}$$

where $\mathbf{S}_x$ and $\mathbf{E}_x$ are as defined above. The second equality holds because of the fact that $\mathsf{M}_{x_{\phi(i,j,\tau)}}^{(i,j,\tau)}$ has the value $(b \oplus r_{\phi(i,j,\tau)}) \cdot 2^\tau$ in the $(i,j)^{th}$ position and zero, everywhere else. Thus, summing up all the $\mathsf{M}_{x_{\phi(i,j,\tau)}}^{(i,j,\tau)}$ matrices results in the matrix M, where $x \oplus r$ is the binary decomposition of M.

Finally, it is clear that $\mathbf{S}_x$ can be efficiently recovered from sk_F and x. □

Claim 8.12 (Shift-hiding property)**.** *Assuming the quantum hardness of learning with errors, the pair $(\mathcal{KG}, \mathcal{E})$ in Construction 4 has the property that*

$$\{pk_{\mathcal{Z}}\} \approx_c \{pk_{H_r}\},$$

for any pk_F with $(pk_F, sk_F) \leftarrow \mathcal{KG}(1^n, 1^m, q, \mathbf{A}, F)$, where $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m}$, $r \in \{0,1\}^\ell$ and $F \in \{\mathcal{Z}, H_r\}$.

Proof. For every $i \in [n], j \in [m], \tau \in [\lceil \log(q) \rceil]$, $b \in \{0,1\}$, let $\mathsf{M}_b^{(i,j,\tau)} = (b \oplus r_{\phi(i,j,\tau)}) \cdot \mathbf{I}_{n\times n}$. Then from the quantum hardness of learning with errors, the following holds for every (i,j,τ) and $b \in \{0,1\}$:

$$\{\mathbf{S}_b^{(i,j,\tau)}\mathbf{A} + \mathbf{E}_b^{(i,j,\tau)}\} \approx_c \{\mathbf{S}_b^{(i,j,\tau)}\mathbf{A} + \mathbf{E}_b^{(i,j,\tau)} + \mathsf{M}_b^{(i,j,\tau)}\}$$

Since $\{\mathbf{S}_b^{(i,j,\tau)}\}$ and $\{\mathbf{E}_b^{(i,j,\tau)}\}$ are sampled independently for every (i,j,τ) and $b \in \{0,1\}$, the proof of the claim follows. □

Remark 8.13. *When consider the all-zeroes function $\mathcal{Z}$, we drop the notation from the parameters. For instance, we denote $pk_{\mathcal{Z}}$ to be simply pk.*

Construction. We consider the following parameters which are relevant to our PRF construction. Let $n, m \in \mathbb{N}$ and let $q \in \mathbb{N}$ be a modulus with $q = 2^{o(n)}$. Let $\ell = nm\lceil \log q \rceil$. Let $\sigma \in (\sqrt{2m}, q/\sqrt{2m})$ and let $p \ll q$ be a sufficiently large rounding parameter with

$$n \cdot m^{3.5}\sigma^3 \lceil \log q \rceil = (q/p) \cdot 2^{-o(n)}.$$

We describe our construction below.

Construction 5 (Revocable PRF scheme)**.** *Let $n \in \mathbb{N}$ be the security parameter and $m \in \mathbb{N}$. Let $q \geq 2$ be a prime and let $\sigma > 0$ be a parameter. Let $(\mathcal{KG}, \mathcal{E})$ be the procedure in Construction 4. Our revocable PRF scheme is defined as follows:*

- $\mathsf{Gen}(1^\lambda)$: *This is the following key generation procedure:*

  1. *Sample $(\mathbf{A}, \mathsf{td}_{\mathbf{A}}) \leftarrow \mathsf{GenTrap}(1^n, 1^m, q)$.*
  2. *Compute $\kappa_{\mathcal{Z}} \leftarrow \mathcal{KG}(1^n, 1^m, q, \mathbf{A}, \mathcal{Z})$, where $\mathcal{Z} : \{0,1\}^\ell \to \mathbb{Z}_q^{n\times m}$ is the such that $\mathcal{Z}(x)$ outputs an all zero matrix for every $x \in \{0,1\}^\ell$. Parse $\kappa_{\mathcal{Z}}$ as (pk, sk).*
  3. *Generate a Gaussian superposition $(|\psi_{\mathbf{y}}\rangle, \mathbf{y} \in \mathbb{Z}_q^n) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma)$ with*

  $$|\psi_{\mathbf{y}}\rangle = \sum_{\substack{\mathbf{x}\in\mathbb{Z}_q^m \\ \mathbf{Ax}=\mathbf{y}}} \rho_\sigma(\mathbf{x}) \, |\mathbf{x}\rangle \, .$$

  Output $k = (pk, sk, \mathbf{y})$, $\rho_k = (pk, |\psi_{\mathbf{y}}\rangle)$ and $\mathsf{MSK} = \mathsf{td}_{\mathbf{A}}$.

- $\mathsf{PRF}(k, x)$: *this is the following procedure:*

  1. *Parse the key k as a tuple $(pk, sk), \mathbf{y})$.*
  2. *Output $\lfloor \mathbf{S}_x \mathbf{y} \rceil_p$. Here, $\mathbf{S}_x \in \mathbb{Z}_q^{n\times n}$ is a matrix that can be efficiently recovered from sk as stated in Claim 8.11.*

- $\mathsf{Eval}(\rho_k, x)$: *this is the following evaluation algorithm:*
  1. *Parse* ρ_k *as* (pk, ρ).
  2. *Compute* $\mathsf{M}_x \leftarrow \mathcal{E}(pk, x)$.
  3. *Measure the register* Aux *of the state* $U(\rho \otimes |0\rangle\langle 0|_{\mathsf{Aux}})U^\dagger$. *Denote the resulting outcome to be* $\mathbf{z}$, *where* U *is defined as follows:*
  $$U\,|\mathbf{t}\rangle\,|0\rangle_{\mathsf{Aux}} \rightarrow |\mathbf{t}\rangle\,|\lfloor \mathsf{M}_x \cdot \mathbf{t}\rceil_p\rangle_{\mathsf{Aux}}$$
  4. *Output* $\mathbf{z}$.
- $\mathsf{Revoke}(\mathsf{MSK}, \rho)$: *given as input the trapdoor* $\mathsf{td}_{\mathbf{A}} \leftarrow \mathsf{MSK}$, *apply the projective measurement* $\{|\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|\,, I - |\psi_{\mathbf{y}}\rangle\langle\psi_{\mathbf{y}}|\}$ *onto the state* ρ *using the procedure* $\mathsf{QSampGauss}(\mathbf{A}, \mathsf{td}_{\mathbf{A}}, \mathbf{y}, \sigma)$ *in Algorithm 2. Output* Valid *if the measurement is successful, and* $\mathsf{Invalid}$ *otherwise.*

Lemma 8.14. *The above scheme satisfies correctness for our choice of parameters.*

Proof. The correctness of revocation follows immediately from the correctness of $\mathsf{QSampGauss}$ in Algorithm 2, which we showed in Theorem 3.3. Next, we show the correctness of evaluation. Let $\kappa_{\mathcal{Z}} \leftarrow \mathcal{KG}(1^n, 1^m, q, \mathbf{A}, \mathcal{Z})$ with $\kappa_{\mathcal{Z}} = (\mathsf{PK}, \mathsf{SK})$. From Claim 8.11, we have for any $x \in \{0,1\}^\ell$:

$$\mathcal{E}(\mathsf{PK}, x) = \mathbf{S}_x \mathbf{A} + \mathbf{E}_x \pmod q,$$

where $\mathbf{S}_x \in \mathbb{Z}_q^{n\times n}$ and $\mathbf{E}_x \in \mathbb{Z}_q^{n\times m}$ with $||\mathbf{E}_x||_\infty \leq (m\sigma)^2 \cdot (nm\lceil \log(q)\rceil)$. Recall that $\mathsf{GenGauss}(\mathbf{A}, \sigma)$ outputs a state $|\psi_{\mathbf{y}}\rangle$ that is overwhelmingly supported on vectors $\mathbf{t} \in \mathbb{Z}_q^m$ such that $\|\mathbf{t}\| \leq \sigma\sqrt{\frac{m}{2}}$ with $\mathbf{A}\cdot\mathbf{t} = \mathbf{y} \pmod q$. Therefore, we have for any input $x \in \{0,1\}^\ell$:

$$\lfloor \mathcal{E}(\mathsf{PK}, x)\cdot \mathbf{t}\rceil_p = \lfloor \mathbf{S}_x\mathbf{A}\cdot\mathbf{t} + \mathbf{E}_x\cdot\mathbf{t}\rceil_p = \lfloor \mathbf{S}_x\cdot\mathbf{y} + \mathbf{E}_x\cdot\mathbf{t}\rceil_p = \lfloor \mathbf{S}_x\cdot\mathbf{y}\rceil_p\,,$$

where the last equality follows from the fact that

$$\|\mathbf{E}_x\cdot\mathbf{t}\|_\infty \leq \|\mathbf{E}_x\|_\infty\cdot\|\mathbf{t}\|_\infty \leq (m\sigma)^2\cdot(nm\lceil\log(q)\rceil)\cdot\sigma\sqrt{m/2}.$$

and $n\cdot m^{3.5}\sigma^3\lceil\log q\rceil = (q/p)\cdot 2^{-o(n)}$ for our choice of parameters. □

Theorem 8.15. *Let* $n \in \mathbb{N}$ *and* q *be a prime modulus with* $q = 2^{o(n)}$ *and* $m \geq 2n\log q$, *each parameterized by* $\lambda \in \mathbb{N}$. *Let* $\ell = nm\lceil\log q\rceil$. *Let* $\sigma \in (\sqrt{2m}, q/\sqrt{2m})$ *and* $\alpha \in (0,1)$ *be any noise ratio with* $1/\alpha = \sigma\cdot 2^{o(n)}$, *and let* $p \ll q$ *be a sufficiently large rounding parameter with*

$$n\cdot m^{3.5}\sigma^3\lceil\log q\rceil = (q/p)\cdot 2^{-o(n)}.$$

Then, assuming the quantum subexponential hardness of $\mathsf{LWE}^m_{n,q,\alpha q}$ *and* $\mathsf{SIS}^m_{n,q,\sigma\sqrt{2m}}$, *our revocable* PRF *scheme* $(\mathsf{Gen}, \mathsf{PRF}, \mathsf{Eval}, \mathsf{Revoke})$ *defined in Construction 5 satisfies 1-revocation security according to Definition 8.3.*

Proof. Let $\mathcal{A}$ be a QPT adversary and suppose that

$$\Pr\left[b \leftarrow \mathsf{Expt}^{\mathcal{A}}(1^\lambda, b)\ :\ b \xleftarrow{\$} \{0,1\}\right] = \frac{1}{2} + \epsilon(\lambda),$$

$\underline{\mathsf{Expt}^{\mathcal{A}}(1^\lambda, b)}$:

Initialization Phase:

- The challenger runs the procedure $\mathsf{Gen}(1^\lambda)$:

  1. Sample $(\mathbf{A}, \mathsf{td}_{\mathbf{A}}) \leftarrow \mathsf{GenTrap}(1^n, 1^m, q)$.
  2. Generate $\mathbf{A}_N \in \mathbb{Z}_q^{(n+m)\times m}$ with $\overline{\mathbf{A}_N} \xleftarrow{\$} \mathbb{Z}_q^{m\times m}$ and $\underline{\mathbf{A}_N} = \mathbf{A}$.
  3. Compute $\kappa_{\mathcal{Z}} \leftarrow \mathcal{KG}(1^n, 1^m, 1^q, \mathbf{A}_N, \mathcal{Z})$, where $\mathcal{KG}$ is as defined in Construction 4 and $\mathcal{Z} : \{0,1\}^\ell \to \mathbb{Z}_q^{n\times m}$ is such that $\mathcal{Z}(x)$ outputs an all zero matrix for every $x \in \{0,1\}^\ell$. Parse $\kappa_{\mathcal{Z}}$ as (pk, sk).
  4. Generate $(|\psi_{\mathbf{y}}\rangle, \mathbf{y} \in \mathbb{Z}_q^n) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma)$ with
  $$|\psi_{\mathbf{y}}\rangle \;=\; \sum_{\substack{\mathbf{x}\in\mathbb{Z}_q^m \\ \mathbf{A}\mathbf{x}=\mathbf{y} \pmod q}} \rho_\sigma(\mathbf{x})\,|\mathbf{x}\rangle\,.$$
  5. Let $k = (pk, sk, \mathbf{y})$, $\rho_k = (pk, |\psi_{\mathbf{y}}\rangle)$ and $\mathsf{MSK} = \mathsf{td}_{\mathbf{A}}$.

- The challenger sends $\rho_k = (pk, |\psi_{\mathbf{y}}\rangle)$ to $\mathcal{A}$.

Revocation Phase:

- The challenger sends the message `REVOKE` to $\mathcal{A}$.

- $\mathcal{A}$ generates a (possibly entangled) bipartite quantum state $\rho_{R,\textsc{Aux}}$ in systems $\mathcal{H}_R \otimes \mathcal{H}_{\textsc{Aux}}$ with $\mathcal{H}_R = \mathcal{H}_q^m$, returns system R and holds onto the auxiliary system $\textsc{Aux}$.

- The challenger runs $\mathsf{Revoke}(\mathsf{MSK}, \rho_R)$, where ρ_R is the reduced state in system R. If the outcome is $\mathsf{Invalid}$, the challenger aborts.

Guessing Phase:

- The challenger samples $x \leftarrow \{0,1\}^\ell$ and sends (x, y) to $\mathcal{A}$, where
  - If $b = 0$: compute $\mathbf{S}_x$ from sk as in Claim 8.11. Set $y = \lfloor \mathbf{S}_x \mathbf{y} \rceil_p$.
  - If $b = 1$: sample $y \leftarrow \{0,1\}^n$.

- $\mathcal{A}$ outputs a string b' and wins if $b' = b$.

Figure 14: The revocable PRF experiment $\mathsf{Expt}^{\mathcal{A}}(1^\lambda, b)$ for Construction 5.

for some $\varepsilon(\lambda)$ with respect to experiment $\mathsf{Expt}^{\mathcal{A}}(1^\lambda, b)$ in Figure 14. Let us now show that $\varepsilon(\lambda)$ is negligible.

Suppose for the sake of contradition that $\epsilon(\lambda) = 1/\mathrm{poly}(\lambda)$. Let us now introduce a sequence of hybrid experiments which will be relevant for the remainder of the proof.

Let $\mathsf{RevDual} = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Revoke})$ be the n-bit key-revocable Dual-Regev scheme from Construction 2. Fix $\mu = 0^n$, where μ is the challenge message in the dual-Regev encryption security.

H_0: This is $\mathsf{Expt}^{\mathcal{A}}(1^\lambda, 0)$ in Figure 14.

H_1: This is the same experiment as $\mathsf{Expt}^{\mathcal{A}}(1^\lambda, 0)$, except for the following changes:

- Sample a random string $r \leftarrow \{0,1\}^\ell$.
- Run the procedure $\mathsf{RevDual.KeyGen}(1^\lambda)$ instead of $\mathsf{GenTrap}(1^n, 1^m, q)$ and $\mathsf{GenGauss}(\mathbf{A}, \sigma)$ to obtain $(\mathbf{A} \in \mathbb{Z}_q^{n \times m}, \mathbf{y} \in \mathbb{Z}_q^n, \mathsf{MSK}, \rho_{\mathsf{SK}})$.
- Compute $(\mathsf{CT}_1, \mathsf{CT}_2) \leftarrow \mathsf{RevDual.Enc}(\mathbf{A}, \mathbf{y}, \mu)$, where $\mathsf{CT}_1 \in \mathbb{Z}_q^{n \times m}$ and $\mathsf{CT}_2 \in \mathbb{Z}_q^n$.
- Set $x = r \oplus \mathsf{bindecomp}(\mathsf{CT}_1)$.

The rest of the hybrid is the same as before.

Note that Hybrids H_0 and H_1 are identically distributed.

H_2: This is the same experiment as before, except that the challenger now uses an alternative key-generation algorithm:

- As before, run the procedure $\mathsf{RevDual.KeyGen}(1^\lambda)$ instead of $\mathsf{GenTrap}(1^n, 1^m, q)$ and $\mathsf{GenGauss}(\mathbf{A}, \sigma)$ to obtain $(\mathbf{A} \in \mathbb{Z}_q^{n \times m}, \mathbf{y} \in \mathbb{Z}_q^n, \mathsf{MSK}, \rho_{\mathsf{SK}})$. Sample $r \leftarrow \{0,1\}^\ell$.
- Let $H_r : \{0,1\}^\ell \to \mathbb{Z}_q^{n \times m}$ be as defined in the beginning of Section 8.3.
- Run the alternate algorithm $\kappa_H \leftarrow \mathcal{KG}(1^n, 1^m, 1^q, \mathbf{A}, H_r)$ instead of $\kappa_{\mathcal{Z}} \leftarrow \mathcal{KG}(1^n, 1^m, 1^q, \mathbf{A}, \mathcal{Z})$.
- Compute the ciphertext $(\mathsf{CT}_1^*, \mathsf{CT}_2^*) \leftarrow \mathsf{RevDual.Enc}(\mathbf{A}, \mathbf{y}, \mu)$, where $\mathsf{CT}_1^* \in \mathbb{Z}_q^{n \times m}$. Then, set $x^* = r \oplus \mathsf{bindecomp}(\mathsf{CT}_1^*)$. Send x^* to the adversary in the guessing phase.

H_3: This is the same hybrid as before, except that we choose $\mathsf{CT}_1^* \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$ and $\mathsf{CT}_2^* \xleftarrow{\$} \mathbb{Z}_q^n$.

H_4: This is the $\mathsf{Expt}^{\mathcal{A}}(1^\lambda, 1)$ in Figure 14.

Note that hybrids H_3 and H_4 are identically distributed.

Suppose that the following holds:

$$\frac{1}{2}\Pr[0 \leftarrow \mathsf{H}_1^{\mathcal{A}}(1^\lambda)] + \frac{1}{2}\Pr[1 \leftarrow \mathsf{H}_2^{\mathcal{A}}(1^\lambda)] = \frac{1}{2} + \delta_1(\lambda), \quad \text{and}$$
$$\frac{1}{2}\Pr[0 \leftarrow \mathsf{H}_2^{\mathcal{A}}(1^\lambda)] + \frac{1}{2}\Pr[1 \leftarrow \mathsf{H}_3^{\mathcal{A}}(1^\lambda)] = \frac{1}{2} + \delta_2(\lambda)$$

for some functions $\delta_1(\lambda)$ and $\delta_2(\lambda)$. We claim that either $\delta_1(\lambda) \geq 1/\mathrm{poly}(\lambda)$ or $\delta_2(\lambda) \geq 1/\mathrm{poly}(\lambda)$ must hold. This is easily seen as follows. By taking the sum of the two expressions above, we get

$$\begin{aligned}&\frac{1}{2}\mathsf{Pr}[0 \leftarrow \mathsf{H}_1^{\mathcal{A}}(1^\lambda)] + \frac{1}{2}\mathsf{Pr}[1 \leftarrow \mathsf{H}_2^{\mathcal{A}}(1^\lambda)] + \frac{1}{2}\mathsf{Pr}[0 \leftarrow \mathsf{H}_2^{\mathcal{A}}(1^\lambda)] + \frac{1}{2}\mathsf{Pr}[1 \leftarrow \mathsf{H}_3^{\mathcal{A}}(1^\lambda)]\\ &= \frac{1}{2} + \delta_1(\lambda) + \frac{1}{2} + \delta_2(\lambda).\end{aligned}$$

Note that we have the identity $\mathsf{Pr}[0 \leftarrow \mathsf{H}_2^{\mathcal{A}}(1^\lambda)] + \mathsf{Pr}[1 \leftarrow \mathsf{H}_2^{\mathcal{A}}(1^\lambda)] = 1$. Moreover, because the hybrids H_0 and H_1, as well as hybrids H_3 and H_4, are identically distributed, we have

$$\begin{aligned}\mathsf{Pr}[0 \leftarrow \mathsf{H}_1^{\mathcal{A}}(1^\lambda)] &= \mathsf{Pr}[0 \leftarrow \mathsf{H}_0^{\mathcal{A}}(1^\lambda)] = \mathsf{Pr}[0 \leftarrow \mathsf{Expt}^{\mathcal{A}}(1^\lambda, 0)] \quad \text{and}\\ \mathsf{Pr}[1 \leftarrow \mathsf{H}_3^{\mathcal{A}}(1^\lambda)] &= \mathsf{Pr}[1 \leftarrow \mathsf{H}_4^{\mathcal{A}}(1^\lambda)] = \mathsf{Pr}[1 \leftarrow \mathsf{Expt}^{\mathcal{A}}(1^\lambda, 1)].\end{aligned}$$

Plugging the identities above into the equation from before, we get

$$\begin{aligned}\frac{1}{2} + \varepsilon(\lambda) &= \frac{1}{2}\mathsf{Pr}[0 \leftarrow \mathsf{Expt}^{\mathcal{A}}(1^\lambda, 0)] + \frac{1}{2}\mathsf{Pr}[1 \leftarrow \mathsf{Expt}^{\mathcal{A}}(1^\lambda, 1)]\\ &= \frac{1}{2} + \delta_1(\lambda) + \delta_2(\lambda).\end{aligned}$$

In other words, we get $\delta_1(\lambda) + \delta_2(\lambda) = \epsilon(\lambda)$, which implies that either $\delta_1(\lambda) \geq 1/\mathrm{poly}(\lambda)$ or $\delta_2(\lambda) \geq 1/\mathrm{poly}(\lambda)$. To complete the proof, we show that both $\delta_1(\lambda)$ and $\delta_2(\lambda)$ are negligible, which yields a contradiction to our assumption that $\varepsilon = 1/\mathrm{poly}(\lambda)$.

Claim 8.16. *By the shift-hiding property*[14] *of $(\mathcal{KG}, \mathcal{E})$ in Claim 8.12, we have $\delta_1(\lambda) \leq \mathsf{negl}(\lambda)$.*

Proof. We first define alternate hybrids $\widetilde{\mathsf{H}_1}$ and $\widetilde{\mathsf{H}_2}$ as follows:

- $\widetilde{\mathsf{H}_1}$ is the same as H_1 except that Revoke is not applied on the returned state,
- $\widetilde{\mathsf{H}_2}$ is the same as H_2 except that Revoke is not applied on the returned state.

Since ignoring Revoke only increases the success probability of the adversary, the following holds:

$$\frac{1}{2}\mathsf{Pr}[0 \leftarrow \widetilde{\mathsf{H}_1}^{\mathcal{A}}(1^\lambda)] + \frac{1}{2}\mathsf{Pr}[1 \leftarrow \widetilde{\mathsf{H}_2}^{\mathcal{A}}(1^\lambda)] \geq \frac{1}{2} + \delta_1(\lambda)$$

We now argue that $\delta_1(\lambda) \leq \mathsf{negl}(\lambda)$.

Suppose not. We design a reduction $\mathcal{B}$ that violates the shift-hiding property as follows.

- Sample $r \xleftarrow{\$} \{0,1\}^\ell$. Send $(\mathcal{Z}, H_r)$ to the challenger.
- The challenger responds with $pk = \left(\mathbf{A}, \left\{\mathsf{CT}_b^{(i,j,\tau)}\right\}_{\substack{i \in [n], j \in [m]\\ \tau \in [\lceil \log(q) \rceil], b \in \{0,1\}}}\right)$
- Compute $(|\psi_{\mathbf{y}}\rangle, \mathbf{y} \in \mathbb{Z}_q^n) \leftarrow \mathsf{GenGauss}(\mathbf{A}, \sigma)$ from the challenger.
- Set $\rho_k = (pk, \rho)$.

[14] Technically, we are invoking the 1-bit indistinguishability variant of the shift-hiding property of $(\mathcal{KG}, \mathcal{E})$ in Claim 8.12, which is implied by the regular indistinguishability notion.

- Compute $(\mathsf{CT}_1, \mathsf{CT}_2) \leftarrow \mathsf{RevDual.Enc}(\mathbf{A}, \mathbf{y}, \mu)$, where $\mathsf{CT}_1 \in \mathbb{Z}_q^{n \times m}$ and $\mathsf{CT}_2 \in \mathbb{Z}_q^n$. Set $x^* = r \oplus \mathsf{bindecomp}(\mathsf{CT}_1)$.
- Compute $\mathsf{Eval}(\rho_k, x^*)$ to obtain y^* while recovering ρ_k^* (using almost as good as new lemma [Aar09]) such that $\mathsf{TD}(\rho_k^*, \rho_k) \leq \mathsf{negl}(\lambda)$.
- Send ρ_k^* to $\mathcal{A}$.
- $\mathcal{A}$ computes a state on two registers R and $\textsc{aux}$. It returns the state on the register R.
- $\mathcal{A}$, on input the register $\textsc{aux}$ and (x^*, y^*), outputs a bit b'.
- Output b'.

If pk is generated using $\mathcal{KG}(1^n, 1^m, q, \mathbf{A}, \mathcal{Z})$ then we are in the hybrid, $\widetilde{\mathsf{H}_1}^{\mathcal{A}}$. If pk is generated using $\mathcal{KG}(1^n, 1^m, q, \mathbf{A}, H_r)$ then we are in the hybrid, $\widetilde{\mathsf{H}_2}^{\mathcal{A}}$. Thus, we violate the shift-hiding property with advantage $\delta_1(\lambda)$. This completes the proof.

□

Next, we invoke the security of the n-bit variant of our key-revocable Dual-Regev scheme which follows from Claim 5.8 and Theorem 6.1.

Claim 8.17. *By the security of our n-bit key-revocable Dual-Regev encryption scheme which is based on the transformation in Construction 1, we have that $\delta_2(\lambda) \leq \mathsf{negl}(\lambda)$.*

Proof. Suppose $\delta_2(\lambda) = 1/\text{poly}(\lambda)$. Using $\mathcal{A}$, we design a reduction $\mathcal{B}$ that violates the revocation security of Construction 1, thus contradicting Theorem 6.1.

The reduction $\mathcal{B}$ proceeds as follows.

- First, it receives as input $\mathbf{A}, \mathbf{y}$ and a quantum state

$$|\psi_{\mathbf{y}}\rangle = \sum_{\substack{\mathbf{x} \in \mathbb{Z}_q^m \\ \mathbf{A}\mathbf{x} = \mathbf{y}}} \rho_\sigma(\mathbf{x}) \, |\mathbf{x}\rangle \, .$$

- The reduction generates a quantum state ρ_k as follows:
  - Sample a random string $r \xleftarrow{\$} \{0,1\}^\ell$.
  - Let $H_r : \{0,1\}^\ell \to \mathbb{Z}_q^{n \times m}$ be as defined in the beginning of Section 8.3.
  - Run the algorithm $\kappa_H \leftarrow \mathcal{KG}(1^n, 1^m, 1^q, \mathbf{A}, H_r)$ and parse κ_H as (pk, sk).
  - Set $\rho_k = (pk, |\psi_{\mathbf{y}}\rangle)$.

  Send ρ_k to $\mathcal{A}$.
- $\mathcal{A}$ outputs a state on two registers R and $\textsc{aux}$. The register R is returned. The reduction forwards the register R to the challenger.

- The reduction then gets the challenge ciphertext $\mathsf{CT} = [\mathsf{CT}_1, \mathsf{CT}_2]^\intercal \in \mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^n$. The reduction then sets

  $$x^* = r \oplus \mathsf{bindecomp}(\mathsf{CT}_1)$$

  and sends x^* to $\mathcal{A}$ in the guessing phase, together with $y = \lfloor \mathbf{S}_{x^*}\mathbf{y} + \mathsf{CT}_2 \rceil_p$ which is computed using the secret key SK (c.f. Claim 8.11).

- $\mathcal{A}$ outputs a bit b'. $\mathcal{B}$ outputs b'.

There are two cases to consider here. In the first case, we have $\mathsf{CT} = [\mathsf{CT}_1, \mathsf{CT}_2]^\intercal \in \mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^n$ is a Dual-Regev ciphertext. Here, $y = \lfloor \mathbf{S}_{x^*}\mathbf{y} + \mathsf{CT}_2 \rceil_p$ precisely corresponds to the output of the pseudorandom function on ρ_k and x. In the second case, we have $\mathsf{CT} = [\mathsf{CT}_1, \mathsf{CT}_2]^\intercal \in \mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^n$, where $\mathsf{CT}_1 \xleftarrow{\$} \mathbb{Z}_q^{n\times m}$ and $\mathsf{CT}_2 \xleftarrow{\$} \mathbb{Z}_q^m$. Here, $y = \lfloor \mathbf{S}_{x^*}\mathbf{y} + \mathsf{CT}_2 \rceil_p$ is (negligibly close) to a uniform distribution on $\mathbb{Z}_p^m$.

Thus, the first case precisely corresponds to H_2 and the second case corresponds to H_3. As a result, $\mathcal{B}$ violates the revocation security of Construction 1 with advantage $\delta_2(\lambda)$. This completes the proof. □

Putting everything together, we have shown that

$$\Pr\left[b \leftarrow \mathsf{Expt}^{\mathcal{A}}(1^\lambda, b) \;:\; b \xleftarrow{\$} \{0,1\}\right] \leq \frac{1}{2} + \mathsf{negl}(\lambda).$$

□

References

[Aar09] Scott Aaronson. "Quantum copy-protection and quantum money". In: *2009 24th Annual IEEE Conference on Computational Complexity*. IEEE. 2009, pp. 229–242 (cit. on pp. 4, 13, 60).

[Aar16] Scott Aaronson. *The Complexity of Quantum States and Transformations: From Quantum Money to Black Holes*. 2016. arXiv: 1607.05256 [quant-ph] (cit. on pp. 7, 16, 50).

[AC02] Mark Adcock and Richard Cleve. "A quantum Goldreich-Levin theorem with cryptographic applications". In: *STACS 2002*. Ed. by Helmut Alt and Afonso Ferreira. Springer, 2002, pp. 323–334. ISBN: 978-3-540-45841-8. DOI: 10.1007/3-540-45841-7_26. arXiv: quant-ph/0108095 (cit. on p. 10).

[Ajt96] Miklós Ajtai. "Generating Hard Instances of Lattice Problems (Extended Abstract)". In: *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996*. Ed. by Gary L. Miller. ACM, 1996, pp. 99–108. DOI: 10.1145/237814.237838. URL: https://doi.org/10.1145/237814.237838 (cit. on pp. 8, 20).

[AK21] Prabhanjan Ananth and Fatih Kaleoglu. "Unclonable encryption, revisited". In: *Theory of Cryptography: 19th International Conference, TCC 2021, Raleigh, NC, USA, November 8–11, 2021, Proceedings, Part I*. Springer. 2021, pp. 299–329 (cit. on p. 14).

[AK22] Prabhanjan Ananth and Fatih Kaleoglu. "A Note on Copy-Protection from Random Oracles". In: *arXiv preprint arXiv:2208.12884* (2022) (cit. on p. 13).

[AKL$^+$22] Prabhanjan Ananth, Fatih Kaleoglu, Xingjian Li, Qipeng Liu, and Mark Zhandry. *On the Feasibility of Unclonable Encryption, and More*. Cryptology ePrint Archive, Paper 2022/884. https://eprint.iacr.org/2022/884. 2022. URL: https://eprint.iacr.org/2022/884 (cit. on pp. 13, 14).

[AKL23] Prabhanjan Ananth, Fatih Kaleoglu, and Qipeng Liu. "Cloning Games: A General Framework for Unclonable Primitives". In: *arXiv preprint arXiv:2302.01874* (2023) (cit. on p. 13).

[AKN$^+$23] Shweta Agrawal, Fuyuki Kitagawa, Ryo Nishimaki, Shota Yamada, and Takashi Yamakawa. "Public Key Encryption with Secure Key Leasing". In: *arXiv preprint arXiv:2302.11663* (2023) (cit. on p. 6).

[AL21] Prabhanjan Ananth and Rolando L La Placa. "Secure Software Leasing". In: *Eurocrypt* (2021) (cit. on pp. 3, 4, 6, 13, 14).

[ALL$^+$21] Scott Aaronson, Jiahui Liu, Qipeng Liu, Mark Zhandry, and Ruizhe Zhang. "New approaches for quantum copy-protection". In: *Annual International Cryptology Conference*. Springer. 2021, pp. 526–555 (cit. on p. 14).

[Ban93] W. Banaszczyk. "New bounds in some transference theorems in the geometry of numbers." In: *Mathematische Annalen* 296.4 (1993), pp. 625–636. URL: http://eudml.org/doc/165105 (cit. on p. 18).

[BB84] C. H. Bennett and G. Brassard. "Quantum cryptography: Public key distribution and coin tossing". In: *Proceedings of IEEE International Conference on Computers, Systems, and Signal Processing*. Bangalore, 1984, p. 175 (cit. on p. 14).

[BBK22] Nir Bitansky, Zvika Brakerski, and Yael Tauman Kalai. *Constructive Post-Quantum Reductions*. 2022. DOI: 10.48550/ARXIV.2203.02314. URL: https://arxiv.org/abs/2203.02314 (cit. on pp. 10, 25).

[BCM$^+$21] Zvika Brakerski, Paul Christiano, Urmila Mahadev, Umesh Vazirani, and Thomas Vidick. *A Cryptographic Test of Quantumness and Certifiable Randomness from a Single Quantum Device*. 2021. arXiv: 1804.00640 [quant-ph] (cit. on p. 21).

[BDGM20] Zvika Brakerski, Nico Döttling, Sanjam Garg, and Giulio Malavolta. "Factoring and pairings are not necessary for io: Circular-secure lwe suffices". In: *Cryptology ePrint Archive* (2020) (cit. on p. 14).

[BGG$^+$14] Dan Boneh, Craig Gentry, Sergey Gorbunov, Shai Halevi, Valeria Nikolaenko, Gil Segev, Vinod Vaikuntanathan, and Dhinakaran Vinayagamurthy. *Fully Key-Homomorphic Encryption, Arithmetic Circuit ABE, and Compact Garbled Circuits*. Cryptology ePrint Archive, Paper 2014/356. https://eprint.iacr.org/2014/356. 2014. URL: https://eprint.iacr.org/2014/356 (cit. on p. 6).

[BI20a] Anne Broadbent and Rabib Islam. "Quantum Encryption with Certified Deletion". In: *Lecture Notes in Computer Science* (2020), pp. 92–122. ISSN: 1611-3349. DOI: 10.1007/978-3-030-64381-2_4. URL: http://dx.doi.org/10.1007/978-3-030-64381-2_4 (cit. on p. 14).

[BI20b] Anne Broadbent and Rabib Islam. “Quantum encryption with certified deletion”. In: *Theory of Cryptography Conference*. Springer. 2020, pp. 92–122 (cit. on pp. 3, 4, 6, 14).

[BJL$^+$21] Anne Broadbent, Stacey Jeffery, Sébastien Lord, Supartha Podder, and Aarthi Sundaram. “Secure software leasing without assumptions”. In: *Theory of Cryptography Conference*. Springer. 2021, pp. 90–120 (cit. on p. 14).

[BK22] James Bartusek and Dakshita Khurana. *Cryptography with Certified Deletion*. 2022. DOI: 10.48550/ARXIV.2207.01754. URL: https://arxiv.org/abs/2207.01754 (cit. on pp. 3, 14).

[BL20] Anne Broadbent and Sébastien Lord. “Uncloneable Quantum Encryption via Oracles”. In: *15th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2020)*. Ed. by Steven T. Flammia. Vol. 158. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2020, 4:1–4:22. DOI: 10.4230/LIPIcs.TQC.2020.4 (cit. on pp. 3, 5, 6, 14).

[BMSZ16] Saikrishna Badrinarayanan, Eric Miles, Amit Sahai, and Mark Zhandry. “Post-zeroizing Obfuscation: New Mathematical Tools, and the Case of Evasive Circuits”. In: *Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part II*. 2016, pp. 764–791 (cit. on p. 14).

[BPR12] Abhishek Banerjee, Chris Peikert, and Alon Rosen. “Pseudorandom functions and lattices”. In: *Advances in Cryptology–EUROCRYPT 2012: 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15-19, 2012. Proceedings 31*. Springer. 2012, pp. 719–737 (cit. on p. 6).

[Bra18] Zvika Brakerski. *Quantum FHE (Almost) As Secure As Classical*. Cryptology ePrint Archive, Report 2018/338. https://ia.cr/2018/338. 2018 (cit. on p. 21).

[BV14a] Zvika Brakerski and Vinod Vaikuntanathan. “Efficient fully homomorphic encryption from (standard) LWE”. In: *SIAM Journal on Computing* 43.2 (2014), pp. 831–871. DOI: 10.1137/120868669 (cit. on p. 6).

[BV14b] Zvika Brakerski and Vinod Vaikuntanathan. “Efficient fully homomorphic encryption from (standard) LWE”. In: *SIAM Journal on computing* 43.2 (2014), pp. 831–871 (cit. on p. 5).

[CFGN96] Ran Canetti, Uri Feige, Oded Goldreich, and Moni Naor. “Adaptively Secure Multi-Party Computation”. In: *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*. STOC ’96. Philadelphia, Pennsylvania, USA: Association for Computing Machinery, 1996, pp. 639–648. ISBN: 0897917855. DOI: 10.1145/237814.238015. URL: https://doi.org/10.1145/237814.238015 (cit. on p. 14).

[CJJ22] Arka Rai Choudhuri, Abhihsek Jain, and Zhengzhong Jin. “Snargs for P from LWE”. In: *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2022, pp. 68–79 (cit. on p. 6).

[CLLZ21] Andrea Coladangelo, Jiahui Liu, Qipeng Liu, and Mark Zhandry. "Hidden cosets and applications to unclonable cryptography". In: *Annual International Cryptology Conference*. Springer. 2021, pp. 556–584 (cit. on pp. 3–6, 10, 14, 52).

[CMP20] Andrea Coladangelo, Christian Majenz, and Alexander Poremba. *Quantum copy-protection of compute-and-compare programs in the quantum random oracle model.* 2020. arXiv: 2009.13865 [quant-ph] (cit. on pp. 13, 14).

[CVW18] Yilei Chen, Vinod Vaikuntanathan, and Hoeteck Wee. "GGH15 beyond permutation branching programs: proofs, attacks, and candidates". In: *Advances in Cryptology–CRYPTO 2018: 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19–23, 2018, Proceedings, Part II 38*. Springer. 2018, pp. 577–607 (cit. on p. 14).

[DGT$^+$10] Yevgeniy Dodis, Shafi Goldwasser, Yael Tauman Kalai, Chris Peikert, and Vinod Vaikuntanathan. "Public-Key Encryption Schemes with Auxiliary Inputs". In: *Theory of Cryptography*. Ed. by Daniele Micciancio. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 361–381. ISBN: 978-3-642-11799-2 (cit. on pp. 9, 10, 18, 26).

[Die82] DGBJ Dieks. "Communication by EPR devices". In: *Physics Letters A* 92.6 (1982), pp. 271–272 (cit. on p. 3).

[DQV$^+$21] Lalita Devadas, Willy Quach, Vinod Vaikuntanathan, Hoeteck Wee, and Daniel Wichs. "Succinct LWE sampling, random polynomials, and obfuscation". In: *Theory of Cryptography: 19th International Conference, TCC 2021, Raleigh, NC, USA, November 8–11, 2021, Proceedings, Part II 19*. Springer. 2021, pp. 256–287 (cit. on p. 14).

[Gao15] Jingliang Gao. "Quantum union bounds for sequential projective measurements". In: *Physical Review A* 92.5 (2015), p. 052331 (cit. on p. 16).

[Gen09] Craig Gentry. "Fully homomorphic encryption using ideal lattices". In: *Proceedings of the forty-first annual ACM symposium on Theory of computing*. 2009, pp. 169–178 (cit. on p. 5).

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. "How to construct random functions". In: *Journal of the ACM* 33.4 (1986), pp. 792–807. ISSN: 0004-5411. DOI: 10.1145/6490.6503 (cit. on p. 48).

[GL89] O. Goldreich and L. A. Levin. "A Hard-Core Predicate for All One-Way Functions". In: *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*. STOC '89. Seattle, Washington, USA: Association for Computing Machinery, 1989, pp. 25–32. ISBN: 0897913078. DOI: 10.1145/73007.73010. URL: https://doi.org/10.1145/73007.73010 (cit. on p. 10).

[Gol06] Oded Goldreich. *Foundations of Cryptography: Volume 1, Basic Tools*. Cambridge University Press, 2006 (cit. on p. 51).

[Got02] Daniel Gottesman. "Uncloneable encryption". In: *arXiv preprint quant-ph/0210062* (2002) (cit. on p. 3).

[GP21] Romain Gay and Rafael Pass. "Indistinguishability obfuscation from circular security". In: *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*. 2021, pp. 736–749 (cit. on p. 14).

[GPV07] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. *Trapdoors for Hard Lattices and New Cryptographic Constructions*. Cryptology ePrint Archive, Report 2007/432. https://eprint.iacr.org/2007/432. 2007 (cit. on pp. 4, 7, 17, 18, 20, 23, 32).

[GR02] Lov K. Grover and Terry Rudolph. "Creating superpositions that correspond to efficiently integrable probability distributions". In: *arXiv: Quantum Physics* (2002) (cit. on p. 21).

[GSW13] Craig Gentry, Amit Sahai, and Brent Waters. *Homomorphic Encryption from Learning with Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based*. Cryptology ePrint Archive, Report 2013/340. https://ia.cr/2013/340. 2013 (cit. on pp. 5, 11, 46).

[GZ20] Marios Georgiou and Mark Zhandry. "Unclonable decryption keys". In: *Cryptology ePrint Archive* (2020) (cit. on pp. 3–5).

[HH00] L. Hales and S. Hallgren. "An improved quantum Fourier transform algorithm and applications". In: *Proceedings 41st Annual Symposium on Foundations of Computer Science*. 2000, pp. 515–525. DOI: 10.1109/SFCS.2000.892139 (cit. on p. 15).

[HMNY21a] Taiga Hiroka, Tomoyuki Morimae, Ryo Nishimaki, and Takashi Yamakawa. *Certified Everlasting Zero-Knowledge Proof for QMA*. 2021. arXiv: 2109.14163 [quant-ph] (cit. on p. 3).

[HMNY21b] Taiga Hiroka, Tomoyuki Morimae, Ryo Nishimaki, and Takashi Yamakawa. *Quantum Encryption with Certified Deletion, Revisited: Public Key, Attribute-Based, and Classical Communication*. 2021. arXiv: 2105.05393 [quant-ph] (cit. on p. 14).

[Int15] Intercept. "How Spies Stole The Keys To The Encryption Castle". In: *https://theintercept.com/2015/02/19/great-sim-heist/*. 2015 (cit. on p. 3).

[JL00] Stanisław Jarecki and Anna Lysyanskaya. "Adaptively Secure Threshold Cryptography: Introducing Concurrency, Removing Erasures". In: *Proceedings of the 19th International Conference on Theory and Application of Cryptographic Techniques*. EUROCRYPT'00. Bruges, Belgium: Springer-Verlag, 2000, pp. 221–242. ISBN: 3540675175 (cit. on p. 14).

[JLS21] Aayush Jain, Huijia Lin, and Amit Sahai. "Indistinguishability obfuscation from well-founded assumptions". In: *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*. 2021, pp. 60–73 (cit. on p. 14).

[KN22] Fuyuki Kitagawa and Ryo Nishimaki. "Functional Encryption with Secure Key Leasing". In: *ASIACRYPT*. 2022 (cit. on p. 5).

[KNY21a] Fuyuki Kitagawa, Ryo Nishimaki, and Takashi Yamakawa. "Secure Software Leasing from Standard Assumptions". In: *Theory of Cryptography*. Springer International Publishing, 2021, pp. 31–61. DOI: 10.1007/978-3-030-90459-3_2. URL: https://doi.org/10.1007%2F978-3-030-90459-3_2 (cit. on p. 6).

[KNY21b] Fuyuki Kitagawa, Ryo Nishimaki, and Takashi Yamakawa. "Secure software leasing from standard assumptions". In: *Theory of Cryptography Conference*. Springer. 2021, pp. 31–61 (cit. on pp. 6, 14, 52).

[LLQZ22] Jiahui Liu, Qipeng Liu, Luowen Qian, and Mark Zhandry. *Collusion Resistant Copy-Protection for Watermarkable Functionalities.* Cryptology ePrint Archive, Paper 2022/1429. https://eprint.iacr.org/2022/1429. 2022. URL: https://eprint.iacr.org/2022/1429 (cit. on pp. 3, 4).

[LZ19] Qipeng Liu and Mark Zhandry. *Revisiting Post-Quantum Fiat-Shamir.* Cryptology ePrint Archive, Paper 2019/262. https://eprint.iacr.org/2019/262. 2019. URL: https://eprint.iacr.org/2019/262 (cit. on p. 22).

[Mah18] Urmila Mahadev. *Classical Verification of Quantum Computations.* 2018. arXiv: 1804.01082 [quant-ph] (cit. on pp. 11, 46).

[MP11] Daniele Micciancio and Chris Peikert. *Trapdoors for Lattices: Simpler, Tighter, Faster, Smaller.* Cryptology ePrint Archive, Report 2011/501. https://eprint.iacr.org/2011/501. 2011 (cit. on p. 20).

[MR04] D. Micciancio and O. Regev. "Worst-case to average-case reductions based on Gaussian measures". In: *45th Annual IEEE Symposium on Foundations of Computer Science.* 2004, pp. 372–381. DOI: 10.1109/FOCS.2004.72 (cit. on p. 18).

[MR07] Daniele Micciancio and Oded Regev. "Worst-Case to Average-Case Reductions Based on Gaussian Measures". In: *SIAM J. Comput.* 37.1 (2007), pp. 267–302. DOI: 10.1137/S0097539705447360. URL: https://doi.org/10.1137/S0097539705447360 (cit. on p. 20).

[NC11] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition.* 10th. USA: Cambridge University Press, 2011. ISBN: 1107002176 (cit. on p. 15).

[Por22] Alexander Poremba. *Quantum Proofs of Deletion for Learning with Errors.* 2022. DOI: 10.48550/ARXIV.2203.01610. URL: https://arxiv.org/abs/2203.01610 (cit. on pp. 3, 10, 14, 22, 23).

[PR06] Chris Peikert and Alon Rosen. "Efficient Collision-Resistant Hashing from Worst-Case Assumptions on Cyclic Lattices". In: *Theory of Cryptography.* Ed. by Shai Halevi and Tal Rabin. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 145–166. ISBN: 978-3-540-32732-5 (cit. on p. 17).

[PS18] Chris Peikert and Sina Shiehian. "Privately constraining and programming PRFs, the LWE way". In: *Public-Key Cryptography–PKC 2018: 21st IACR International Conference on Practice and Theory of Public-Key Cryptography, Rio de Janeiro, Brazil, March 25-29, 2018, Proceedings, Part II 21.* Springer. 2018, pp. 675–701 (cit. on p. 52).

[Reg05] Oded Regev. "On lattices, learning with errors, random linear codes, and cryptography". In: *Journal of the ACM* 56.6 (2005), 34:1–34:40. ISSN: 0004-5411. DOI: 10.1145/1568318.1568324 (cit. on pp. 4, 7, 20, 21).

[Riv98] Ronald L. Rivest. "Can We Eliminate Certificate Revocation Lists?" In: *Proceedings Financial Cryptography '98.* FC'98 (Anguilla, British West Indies, Feb. 23–25, 1998). Ed. by Rafael Hirschfeld. Vol. 1465. Lecture Notes in Computer Science. Springer, Feb. 1998, pp. 178–183. ISBN: 978-3-540-64951-9. DOI: 10.1007/BFb0055482 (cit. on p. 3).

[SSTX09] Damien Stehlé, Ron Steinfeld, Keisuke Tanaka, and Keita Xagawa. *Efficient Public Key Encryption Based on Ideal Lattices*. Cryptology ePrint Archive, Paper 2009/285. https://eprint.iacr.org/2009/285. 2009. URL: https://eprint.iacr.org/2009/285 (cit. on pp. 5, 20, 23).

[Stu95] S. Stubblebine. "Recent-secure authentication: enforcing revocation in distributed systems". In: *2012 IEEE Symposium on Security and Privacy*. Los Alamitos, CA, USA: IEEE Computer Society, May 1995, p. 0224 (cit. on p. 3).

[TL17] Marco Tomamichel and Anthony Leverrier. "A largely self-contained and complete security proof for quantum key distribution". In: *Quantum* 1 (July 2017), p. 14. ISSN: 2521-327X. DOI: 10.22331/q-2017-07-14-14. URL: https://doi.org/10.22331/q-2017-07-14-14 (cit. on p. 14).

[Unr13] Dominique Unruh. *Revocable quantum timed-release encryption*. Cryptology ePrint Archive, Report 2013/606. https://ia.cr/2013/606. 2013 (cit. on pp. 3, 4, 14).

[Unr15] Dominique Unruh. *Computationally binding quantum commitments*. Cryptology ePrint Archive, Paper 2015/361. https://eprint.iacr.org/2015/361. 2015. URL: https://eprint.iacr.org/2015/361 (cit. on p. 21).

[Wat05] John Watrous. *Zero-knowledge against quantum attacks*. 2005. DOI: 10.48550/ARXIV.QUANT-PH/0511020. URL: https://arxiv.org/abs/quant-ph/0511020 (cit. on p. 16).

[Wie83] Stephen Wiesner. "Conjugate Coding". In: *SIGACT News* 15.1 (Jan. 1983), pp. 78–88. ISSN: 0163-5700. DOI: 10.1145/1008908.1008920. URL: https://doi.org/10.1145/1008908.1008920 (cit. on pp. 3, 14).

[Wil13] Mark M. Wilde. *Quantum Information Theory*. 1st. USA: Cambridge University Press, 2013. ISBN: 1107034256 (cit. on p. 15).

[WW21] Hoeteck Wee and Daniel Wichs. "Candidate obfuscation via oblivious LWE sampling". In: *Advances in Cryptology–EUROCRYPT 2021: 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17–21, 2021, Proceedings, Part III*. Springer. 2021, pp. 127–156 (cit. on p. 14).

[WZ82] William K Wootters and Wojciech H Zurek. "A single quantum cannot be cloned". In: *Nature* 299.5886 (1982), pp. 802–803 (cit. on p. 3).

[Zha21] Mark Zhandry. "Quantum Lightning Never Strikes the Same State Twice. Or: Quantum Money from Cryptographic Assumptions". In: *J. Cryptol.* 34.1 (Jan. 2021). ISSN: 0933-2790. DOI: 10.1007/s00145-020-09372-x. URL: https://doi.org/10.1007/s00145-020-09372-x (cit. on p. 6).