Post quantum cryptographic constructions

Li, Zhe

2020

Zhe, L. (2020). Post quantum cryptographic constructions. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/146200>

<https://doi.org/10.32657/10356/146200>

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Downloaded on 26 Apr 2023 19:49:53 SGT

**NANYANG
TECHNOLOGICAL
UNIVERSITY**

SINGAPORE

Post Quantum Cryptographic Constructions

ZHE LI

SCHOOL OF PHYSICAL & MATHEMATICAL SCIENCES

July 2020

Post Quantum Cryptographic Constructions

ZHE LI

School of Physical & Mathematical Sciences

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Doctor of Philosophy

July 2020

STATEMENT OF ORIGINALITY

I hereby certify that the work embodied in this thesis is the result of original research done by me except where otherwise stated in this thesis. The thesis work has not been submitted for a degree or professional qualification to any other university or institution. I declare that this thesis is written by myself and is free of plagiarism and of sufficient grammatical clarity to be examined. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

Date Aug-04-2020

.....

Name Zhe Li

.....

Signature

.....

SUPERVISOR DECLARATION STATEMENT

I have reviewed the content and presentation style of this thesis and declare it of sufficient grammatical clarity to be examined. To the best of my knowledge, the thesis is free of plagiarism and the research and writing are those of the candidate's except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

Date 4 August 2020

Name Prof Ling San

Signature

.....

.....

.....

AUTHORSHIP ATTRIBUTION STATEMENT

This thesis contains material from *three* papers published in the following peer-reviewed journal(s) / from papers accepted at conferences or in submission in which I am listed as an author.

[LXY19] Zhe Li, Chaoping Xing and Sze Ling Yeo. Reducing the Key Size of McEliece Cryptosystem from Automorphism-induced Goppa Codes via Permutations. *PKC 2019:599-617*.

[LLXY20] Zhe Li, San Ling, Chaoping Xing and Sze Ling Yeo. On the Bounded Distance Decoding Problem for Lattices Constructed and Their Cryptographic Applications. *IEEE Trans. Information Theory 66(4):2588-2598 (2020)*.

[HLY] Jingnan He, Zhe Li and Sze Ling Yeo. A Variant of Subset Sum Problem with Provably Secure Cryptographic Applications. *In submission*.

Chapter 2 is reproduced from the paper [LXY19]. The contributions of the co-authors are listed as follows:

- Chaoping Xing proposed to consider the idea.
- Zhe Li and Sze Ling Yeo realized the idea to form a scheme.

Chapter 3 is reproduced from the paper [LLXY20]. The contributions of the co-authors are listed as follows:

- San Ling and Chaoping Xing proposed to consider the special lattice construction.

- Zhe Li and Sze Ling Yeo proved the properties of the lattices, constructed the cryptosystem and analyzed the security.

Chapter 4 is reproduced from the paper [HLY]. The contributions of the co-authors are listed as follows:

- Zhe Li proposed to consider the rounded subset sum problem and proved the first reduction.
- Jingnan He proved the weak and leaky secret property of the rounded subset sum problem.
- Sze Ling Yeo constructed the cryptosystem and proved the security of the cryptosystem.

Date **Aug-04-2020**

.....

Name **Zhe Li**

.....

Signature

.....

ABSTRACT

With the past and ongoing developments of quantum computers, the existing asymmetric schemes are no longer secure. In a report of 2006, the National Institute of Standards and Technology (NIST) predicts that large scale quantum computers will be built within three decades. Thus, we need to actively find quantum resistant cryptographic constructions. To date, there are many proposals based on different hard mathematical problems. This thesis seeks to enrich the diversity of the quantum resistant schemes and improve the efficiency of existing schemes. We present constructions from coding theory, lattices and subset sum problems. Concretely, our quantum-safe contributions are listed as follows.

For the code-based construction, we propose a new general construction to reduce the public key size of McEliece cryptosystems constructed from automorphism-induced Goppa codes. In particular, we generalize the ideas of automorphism-induced Goppa codes by considering nontrivial subsets of automorphism groups to construct Goppa codes with a nice block structure. By considering additive and multiplicative automorphism subgroups, we provide explicit constructions to demonstrate our technique. We show that our technique can be applied to automorphism-induced Goppa codes based cryptosystems to further reduce their key sizes.

For the lattice-based construction, we propose new classes of trapdoor functions to solve the bounded distance decoding problem in lattices. Specifically, we construct lattices based on properties of polynomials for which the bounded distance decoding problem is hard to solve unless some trapdoor information is revealed. We thoroughly analyze the security of our proposed functions using state-of-the-art attacks and results on lattice reductions. Then, we describe how

our functions can be used to design quantum-safe encryption schemes with reasonable public key sizes. Our encryption schemes are efficient with respect to key generation, encryption and decryption.

For the subset sum problem, we propose a variant of the subset sum problem which we name the *rounded subset sum problem*. We prove that this problem is as hard as the subset sum problem. We then present a public key cryptosystem that achieves semantic security based on the rounded subset sum problem. Next we show the robustness of the rounded subset sum assumption in the presence of auxiliary input of the secret. Based on the robustness result, we present a public key encryption that is resilient to adaptive leakage of the secret key, and this solves an open problem in the work of Lyubashevsky, Palacio, and Segev (TCC 2010). The subset sum assumption we used to construct adaptive leakage-resilient scheme is weaker than the assumption underlying existing lattice-based leakage-resilient schemes such as Dodis et al.(TCC 2010). We also present other applications of the rounded subset sum problem.

ACKNOWLEDGEMENTS

I am very thankful to my supervisors Professor San Ling, Professor Chaoping Xing, and Dr. Sze Ling Yeo. I would like to thank Professor San Ling for providing me with the opportunity to study in NTU, as well as for his encouragement and support during my post-graduate years. I owe many thanks to Professor Chaoping Xing. First, he gave me the freedom to choose my own research topics. Second, I benefited much from his deep expertise in mathematics and coding theory. More importantly, he gave me invaluable advice and guidance and provided me with many suggestions for my future career. He even suggested me to do physical exercises every week. For all the things I learned from Professor Chaoping Xing, I will be forever grateful to him. I would like to thank Dr. Sze Ling Yeo for sharing her ideas with me and the helpful discussions. We worked together on many different topics. I want to thank Sze Ling, especially for her patience. Every time when I share an idea with her, she will attempt to understand it and give some advice no matter how ridiculous the idea sounds. She also helped me to improve my writing.

Thanks to Jingnan He, San Ling, Chaoping Xing and Sze Ling Yeo for the discussions and collaborations with them.

In addition, I am grateful to Professor Huaxiong Wang and Dr. Khoa Nguyen for allowing me to take part in their weekly cryptographic seminars. I learned a lot from the seminar.

Thanks to all the members in MAS-04-21. I spent almost all the day time in that office in the past four years. Thanks to my friends Nankun Hong and Haoyang Wang in NTU. I cherish the time we spent together. Especially during the summer holiday, Haoyang and I lived in

Nankun's hall together. The time spent together will always leave a deep impression on me.

Most important of all, I want to thank my wife Yancun Chen and my parents for their continuous support and patience.

I am writing this thesis during the circuit breaker in Singapore due to Covid 19. We do not know when the disease has a cure or when there will be no new infection. To me, the feeling is similar to research life with many uncertainties. We can only hope that Covid 19 will be over as soon as possible. Maybe the uncertainty of the future is the nature of life. This summer is indeed memorable!

LIST OF WORKS

Below is the list of works done during my graduate studies in Nanyang Technological University.

1. Zhe Li, Chaoping Xing and Sze Ling Yeo. Reducing the Key Size of McEliece Cryptosystem from Automorphism-induced Goppa Codes via Permutations. *PKC 2019:599-617*.
2. Zhe Li, San Ling, Chaoping Xing and Sze Ling Yeo. On the Bounded Distance Decoding Problem for Lattices Constructed and Their Cryptographic Applications. *IEEE Trans. Information Theory 66(4):2588-2598 (2020)*.
3. Jingnan He, Zhe Li and Sze Ling Yeo. A Variant of Subset Sum Problem with Provably Secure Cryptographic Applications. *In submission*.

CONTENTS

1. Introduction	1
1.1 Post Quantum Cryptography	2
1.1.1 Code-based Cryptography	2
1.1.2 Lattice-based Cryptography	5
1.1.3 Hash-based Cryptography	11
1.1.4 Isogeny-based Cryptography	12
1.1.5 Multivariate-quadratic Cryptography	14
1.2 Security of Cryptosystems	14
1.3 Contributions of the Thesis	16
1.3.1 Contribution 1	16
1.3.2 Contribution 2	19
1.3.3 Contribution 3	21
1.4 Organization of the Thesis	28
2. Preliminaries	29
2.1 Notations	29
2.2 Probability Distribution Results	29
2.2.1 Min-entropy	29
2.2.2 Discrete Gaussian Distribution	30
2.2.3 Some Results About Bounds	31

2.2.4 Rényi Divergence	31
2.3 Lattices	32
2.4 Goppa Codes	35
2.4.1 Permutations of Goppa Codes	37
2.4.2 Automorphism-induced Goppa Codes	40
2.5 The Subset Sum Problem	45
2.6 Security Definitions	46
2.7 Cryptosystems	47
2.7.1 The McEliece Cryptosystem	47
2.7.2 The GGH Cryptosystem	48
3. Goppa Codes With Reduced Key Size	51
3.1 Introduction	51
3.1.1 Organization of the Chapter	52
3.2 Partial Quasi-cyclic and Partial Quasi-dyadic Goppa Codes	53
3.3 Security Discussion	57
3.4 Practical Implementation Considerations	59
3.4.1 Practical Implementation Scheme	59
3.4.2 Concrete Parameters	61
4. Polynomial Lattices	63
4.1 Introduction	63
4.1.1 Comparison	63
4.1.2 Organization of the Chapter	64
4.2 Polynomial Lattices	65
4.3 Construction of Our Trapdoor Functions	72
4.4 Security Analysis of Our Trapdoor Functions	75

4.4.1	Error Search	75
4.4.2	Search for the Trapdoor	76
4.4.3	Embedding Attack to Solve BDD	77
4.4.4	Attack When $c(x)$ Is Irreducible	78
4.4.5	Attack Based on Chor-Rivest Scheme	79
4.5	A Practical Encryption Scheme	80
4.5.1	Description	80
4.5.2	Choosing the Parameters	83
5.	Rounded Subset Sum	85
5.1	Introduction	85
5.1.1	Comparisons With Lattice-based Schemes and Related Work	85
5.1.2	Organization	86
5.2	Rounded Subset Sum	87
5.2.1	One-wayness of Rounded Subset Sum	89
5.2.2	Pseudorandomness of Rounded Subset Sum	95
5.3	Public Key Cryptosystem Based on Rounded Subset Sum	98
5.4	Rounded Subset Sum Problem With Weak and Leaky Secrets	100
5.5	Adaptive Leakage-Resilient Secure PKE	106
5.6	Reusable Extractors	109
6.	Conclusions	111
6.1	Future Work	112
	Bibliography	114

1. INTRODUCTION

Currently, two main public key cryptosystems are widely deployed to secure communications, namely, the RSA [RSA78] and the ElGamal [Gam84] schemes. The security of these two cryptosystems builds on the assumptions that the underlying mathematical problems, specifically, integer factorization and discrete logarithm over different groups, are hard for classical computers. However, it is well-known that Peter Shor [Sho94] designed quantum algorithms that can solve both of the problems in quantum polynomial time. In the 1990s, Shor's algorithm was mainly regarded as a theoretical algorithm which showed the advantages of quantum computers. To implement Shor's quantum algorithm, a quantum computer equipped with thousands of qubits is needed. In 2016, the American National Institute for Standards and Technology (NIST) [CCJ⁺16] predicted that such large-scale quantum computers will be created within one decade to three decades. Indeed, quantum technology has developed rapidly in the past few years and has thus intensified people's concerns on the security of existing public key cryptosystems. This is especially true after Google has achieved *quantum supremacy* [AAB⁺19]. As a result, the security community is eager to seek for alternative public key cryptosystems that can resist large scale quantum computers. In December 2016, NIST [NIS17a] initiated a multi-year project to standardize one or more post-quantum public key cryptosystems. As of the writing of this thesis, it is in the process to evaluate the round three submissions [NIS20].

1.1 Post Quantum Cryptography

Cryptographic schemes that are resistant to quantum algorithms belong to the realm of *post quantum cryptography* (PQC). Note that the notion of PQC is different from the notion of *quantum cryptography* (QC). PQC works in the classical Turing machine model. It is conjectured to withstand future large scale quantum computers and no quantum mechanism is used in the PQC . However, QC works in the quantum Turing machine model and quantum mechanism is fully exploited in the QC protocols. At present, there are five classes of cryptographic schemes predicted to endure large scale quantum computers. The five classes of cryptosystems include code-based cryptosystems, hash-based cryptosystems, isogeny-based cryptosystems, lattice-based cryptosystems, and multivariate-quadratic cryptosystems. Some of these systems have roots dating back to the time that RSA was invented and a number of them are experiencing swift developments in recent years.

As in the classical case, the main focus of PQC , such as in the NIST submissions, is to design practical and secure public key encryption schemes and digital signature schemes. We will next briefly review the five classes of PQC . For more information, readers may refer to [BL17, NDR⁺19].

1.1.1 Code-based Cryptography

Code-based constructions are divided into two classes depending on whether the construction hides a private code structure or not. For instance, the security of McEliece constructions relies on the assumption that an adversary is unable to seek out the code structure from the public key. On the other hand, in LPN based constructions, random linear codes are involved. Thus no structure is hidden against an adversary. We introduce the two classes of constructions in what follows.

The history of code-based cryptosystem traces back to the time of invention of RSA. In 1978, McEliece proposed the first code-based cryptosystem [McE78]. The system established a

general framework for code-based cryptosystems. Concretely, the McEliece encryption scheme works as follows. Let G be a generator matrix of an $[n, k, 2t + 1]$ -linear code $\mathcal{C}$ with an efficient decoder $\mathcal{D}$. Let S be a $k \times k$ invertible matrix and let P be an $n \times n$ permutation matrix. Let $G' = SG P$. Then the public key is G' and the private key consists of the matrices S, M, G and the decoder $\mathcal{D}$. To encrypt a message $\mathbf{x}$, one chooses an error vector $\mathbf{e}$ with Hamming weight at most t and computes $\mathbf{y} = \mathbf{x}G' + \mathbf{e}$. To recover the message $\mathbf{x}$ from the ciphertext $\mathbf{y}$, one first computes $\mathbf{y}' = \mathbf{y}P^{-1}$, uses the decoder $\mathcal{D}$ to decode $\mathbf{y}'$ to obtain $\mathbf{y}'' \in \mathcal{C}$, and then recover $\mathbf{x}$ from $\mathbf{y}''$.

An alternative but equivalent code-based encryption scheme was proposed by Niederreiter in [Nie86] in which the parity-check matrix instead of the generator matrix was used. In this scheme, a message is converted into an error vector of Hamming weight t and encryption is performed by multiplying this vector with the parity check matrix. Once again, decryption is accomplished via the fast decoder $\mathcal{D}$ while the security is based on the difficulty of the syndrome decoding problem.

The above descriptions only outline the essential ideas of code-based schemes. Variants of these schemes have been proposed to achieve different forms of security such as the CCA2 security [KI01, HHK17]. The main advantage of code-based cryptosystems lies in its efficient operations leading to very efficient encryption and decryption. As such, it continues to draw much interest to design new code-based cryptographic primitives. For instance, in the NIST round one submissions [NIS17b], several proposals on code-based key encapsulation mechanisms were proposed, including [BBB⁺18b, BBB⁺18a]. In the latest NIST round three results [NIS20], the classic McEliece scheme [BCL⁺17] is chosen as one of the round three finalists, and BIKE [ABB⁺18], HQC [MAB⁺19a] are selected as two of alternate candidates in the round three lists.

Essentially, any code $\mathcal{C}$ used in the schemes must satisfy the main property, namely, it has an efficient decoder using a particular matrix M but multiplying this matrix with a random matrix transforms the corresponding code into a random code. In addition, the code used should not

exhibit any structural weakness to recover the private key. The first family of codes suggested by McEliece in [McE78] is the family of *Goppa codes*. Subsequently, other families of codes are proposed, including algebraic geometric codes [JM96], Reed-Muller codes [Sid94], Reed Solomon codes [Nie86], and more recently, medium density parity check(MDPC) codes [MTSB13]. While all these codes have an efficient decoding algorithm, the structures exhibited by some of the codes make them vulnerable to other attacks. Ideally, one hopes to construct public keys with reasonably short lengths such that the underlying structures are properly concealed. Quasi-cyclic MDPC(QC-MDPC) [MTSB13] codes are more recent designs that seem promising to achieve these two goals simultaneously. On the other hand, codes such as Reed-Muller codes and Reed Solomon codes, while providing short public keys, have all been broken [MS07,MSOS92].

The QC-MDPC scheme employs a non-negligible decryption failure probability, which concretely relies on the chosen decoding algorithm and the parameters. Such a decryption failure probability is fully exploited to form a key recovery attack [GJS16]. In summary, the McEliece cryptosystem is accomplished by hiding the structure of a carefully chosen code such as the Goppa code. The security of McEliece cryptosystem is built on the assumption that one cannot infer the private code structure from the public key and the assumption that without knowing the structure of the code, one cannot efficiently decode a given ciphertext.

By considering a different approach, Alekhnovich [Ale03] proposed two encryption schemes, the security of which reduces to the problem of decoding random linear codes. These two encryption schemes are inspired by Ajtai-Dwork(AD) cryptosystems [AD97] which are built from hard worst-case lattice problems. As mentioned in Section 1.1.2 below, the *learning with errors* problem is also inspired by the AD cryptosystem. In fact, the LWE problem is a q -ary analog of the *learning parity with noise(LPN)* problem which was proposed in [BFKL93]. See Section 1.1.2 for the AD cryptosystem and the LWE problem. Here we briefly review the LPN problem. Given the length of the secret vector n , a noise rate $\tau < 0.5$, and the number of instances m , the LPN is defined as the problem to find a vector $\mathbf{s}$ from instances in the form of $(\mathbf{a}, e + \langle \mathbf{a}, \mathbf{s} \rangle)$,

where $\mathbf{a}$ is a random vector with the same dimension as the secret vector and e is sampled from a Bernoulli distribution parameterized by τ . From the perspective of coding theory, by writing the LPN instances in matrix form, the LPN problem can equivalently be considered as: for a generator matrix G of a random binary linear code and a noisy word $\mathbf{y}$ (that is a vector perturbed by a noise vector), decode the private message $\mathbf{s}$ from G and $\mathbf{y}$. Equipping the LPN problem with a ring structure, one gets the compact Ring-LPN problem [HKL⁺12, DP12]. Further, concatenating two random elements under the ring structure yields the quasi-cyclic random codes. Aguilar et al. [MBD⁺18] considered such a generalization of the Ring-LPN problem and propose the Hamming Quasi-Cyclic(HQC) [MAB⁺19a] cryptosystem, which has been submitted to NIST and is one of the alternate candidates in round three [NIS20].

All the schemes mentioned so far are so-called Hamming-based schemes in the sense that they are constructed based on the Hamming metric between vectors. An alternative direction in code-based cryptography is the rank metric code-based schemes. Most of the rank metric code based schemes are adapted from Hamming code based schemes by considering the rank metric in vector spaces over finite extension fields instead. It is believed that decoding random rank-metric codes requires considerably more effort than the Hamming-metric counterparts. As such, the rank metric code-based schemes have the conjectured property that for the same security level, rank metric variants occupy smaller public key size and smaller ciphertext size. However, recent breakthroughs in algebraic attacks on hard problems of rank metric code [BBB⁺20, BBC⁺20] result in a significant increase in the public key size of rank metric code based NIST submissions [MAB⁺20, ABD⁺20], namely, an increase by a factor of around 40% compared to the parameters in earlier submissions [MAB⁺19b, ABD⁺19].

1.1.2 Lattice-based Cryptography

Recall that an n -dimensional lattice is an additive discrete subgroup of $\mathbb{R}^n$. Lattice-based cryptography represents the class of cryptosystems whose security is built on hard lattice problems.

In particular, two main problems and their variants lie at the core of most of the lattice-based designs, namely, the short integer solutions (SIS) and learning with errors (LWE) problems. We briefly review the two problems, their variants and applications in what follows.

In 1996, Ajtai [Ajt96] published his significant breakthrough result, which established a connection between worst-case hard approximation lattice problems and the average-case SIS problems. Ajtai defines the problem *short integer solution(SIS)* and proves that the solver for SIS can be used to address worst-case hard lattice problems. We briefly recall the SIS problem here. Let n, m, q be public parameters. Let $\mathbf{s} \in \mathbb{Z}_q^m$ be the secret vector and $\|\mathbf{s}\| \leq \beta$, where β is a small bound. Given $(A, A\mathbf{s}) \in \mathbb{Z}_q^{n \times m} \times \mathbb{Z}_q^n$, where A is uniformly sampled from the domain, the SIS problem asks to find $\mathbf{s}$ such that $\|\mathbf{s}\| \leq \beta$. In a later work in 1997, Ajtai and Dwork [AD97] designed a public key cryptosystem, whose security employs a reduction from worst-case hard lattice problems.

In 2005, Regev [Reg05] defined the learning with errors(LWE) problem. In the form sense, LWE is an extension of the LPN problem from a binary to a q -ary setting. A significant contribution of this work is a quantum polynomial time reduction from worst-case hard approximation lattice problems to average-case LWE problems. We now recall the LWE problem. Let $\mathbf{s} \in \mathbb{Z}_q^n$ be the secret vector, where n is a positive integer representing the dimension of the associated lattice and q is the modulus. Given polynomially many instances in the form of $(\mathbf{a}, e + \langle \mathbf{a}, \mathbf{s} \rangle \bmod q)$, where $\mathbf{a}$ is uniformly drawn from its domain and e is drawn from a small distribution $\mathcal{D}$, the LWE problem seeks to recover the secret vector $\mathbf{s}$ from the samples. Assume the number of LWE samples is m . Then the LWE problem can also be presented in a matrix form. Given $(A, \mathbf{b} = A\mathbf{s} + \mathbf{e}) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m$, the LWE problem is defined to recover the vector $\mathbf{s}$ from the matrix form instances.

There are two versions lattice based encryption schemes from the LWE problem and the SIS problem. One is the dual of the other. We sketch the two encryption schemes here. Regev [Reg05] proposed the first LWE based public key encryption scheme in his LWE paper.

The public key is the LWE instance. For LWE instance $(A, \mathbf{b} = A\mathbf{s} + \mathbf{e}) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^n$, the public key is the pair $(A, \mathbf{b})$. The secret key is the vector $\mathbf{s}$. The security of the key generation procedure is built from the hardness assumption of the LWE problem. For a message bit $\mu \in \{0, 1\}$ to be encrypted, the encryption algorithm samples a vector $\mathbf{r}$ from $\{0, 1\}^m$, computes $(\mathbf{c}_1 = \mathbf{r}^T A, c_2 = \langle \mathbf{r}, \mathbf{b} \rangle + \mu \cdot \lceil q/2 \rceil) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ and outputs $(\mathbf{c}_1, c_2)$ as ciphertext. Let B be the resulting matrix of A augmenting with the vector $\mathbf{b}$, namely, $B = (A, \mathbf{b})$. Then the ciphertext can be viewed as a SIS instance with B being the public matrix and $\mathbf{r}$ the secret vector, i.e., $\mathbf{c} = \mathbf{r}^T B + (\mathbf{0}, \mu \cdot \lceil q/2 \rceil)$. Thus, $(B, \mathbf{c} - (\mathbf{0}, \mu \cdot \lceil q/2 \rceil))$ is a SIS instance. Given ciphertext $(\mathbf{c}_1, c_2)$ and the secret key $\mathbf{s}$, the decryption algorithm computes $\sigma = c_2 - \langle \mathbf{s}, \mathbf{c}_1 \rangle$ and outputs 0 if $|\sigma|$ is less than $q/4$ otherwise outputs 1. The correctness of the cryptosystem follows the computation

$$\sigma = c_2 - \langle \mathbf{s}, \mathbf{c}_1 \rangle = \mathbf{r}^T (A\mathbf{s} + \mathbf{e}) + \mu \cdot \lceil q/2 \rceil - \mathbf{r}^T A\mathbf{s} = \langle \mathbf{r}, \mathbf{e} \rangle + \mu \lceil q/2 \rceil.$$

The scheme will choose parameters such that $|\langle \mathbf{r}, \mathbf{e} \rangle| \leq q/4$ with overwhelming probability. In 2008, Gentry, Peikert and Vaikuntanathan [GPV08] proposed a dual version public key encryption scheme. In the GPV scheme, the order of key generation and encryption are exchanged. The public key is generated from the SIS instance. For SIS instance $(A, \mathbf{b} = A^T \mathbf{s})$, the public key is the pair $(A, \mathbf{b})$. The secret key is the vector $\mathbf{s}$. The security of public key is from the hardness of the SIS problem. For a message bit μ and public key $(A, \mathbf{b})$, the encryption procedure samples a vector $\mathbf{u} \in \mathbb{F}_q^n$ uniformly at random, $\mathbf{e}$ from χ^m and f from χ , where χ is a predefined Gaussian distribution with small derivation, computes $(\mathbf{c}_1, c_2) = (A\mathbf{u} + \mathbf{e}, \langle \mathbf{b}, \mathbf{u} \rangle + f + \mu \cdot \lceil q/2 \rceil)$ and outputs $(\mathbf{c}_1, c_2)$ as ciphertext. Let B be the resulting matrix of A stacking on $\mathbf{b}^T$. Thus the ciphertext can be regarded as a LWE instance with B being the public matrix, $\mathbf{u}$ the secret vector, and $(\mathbf{e}^T, f)^T$ the error vector. Now the security of ciphertext is from the hardness of the LWE problem. Given ciphertext $(\mathbf{c}_1, c_2)$, the decryption procedure computes

$$\sigma = c_2 - \langle \mathbf{c}_1, \mathbf{s} \rangle = \mathbf{s}^T A\mathbf{u} + f + \mu \cdot \lceil q/2 \rceil - \mathbf{s}^T A\mathbf{u} - \mathbf{s}^T \mathbf{e} = f - \mathbf{s}^T \mathbf{e} + \mu \cdot \lceil q/2 \rceil$$

and outputs 0 if the absolute value of σ is less than $q/4$. The correctness follows the same principle as the Regev encryption scheme.

There are many different variants of LWE, for instance, Ring-LWE, Module-LWE, P-LWE, LWR, Middle-Product LWE, and so on. We briefly introduce the variants here. Ring-LWE [LPR10] is a compact variant of LWE, where the public vector $\mathbf{a}$, the secret vector $\mathbf{s}$ and the error vector $\mathbf{e}$ lie in a predefined ring structure. They prove that there is a quantum reduction from the worst-case approximation ideal lattice problems to the Ring-LWE problem. Similar to Ring-LWE, P-LWE is a variant of LWE in which the secret vector, sample vector, and the error vector are defined over a polynomial ring and the operation is taken modulo a predefined polynomial. Module-LWE [LS15] is an extension of LWE and Ring-LWE, where each entry of LWE is sampled from a ring instead of $\mathbb{Z}_q$. As before, there exists a quantum reduction from the worst-case module lattice problems to Module-LWE problems. In practical applications, Module-LWE problem provides more flexibility for the trade-off between the security and computational cost. Learning with rounding (LWR) [BPR12] is a derandomized variant of LWE. Let $p < q$ be a smaller modulus. The rounding function $\lceil \cdot \rceil_p : \mathbb{Z}_q \rightarrow \mathbb{Z}_p$ is defined as $\lceil x \rceil_p := \lceil (p/q)x \rceil_p$. The LWR problem seeks to recover the secret vector $\mathbf{s}$ from a polynomial number of samples of the form $(\mathbf{a}, \lceil \langle \mathbf{a}, \mathbf{s} \rangle \rceil_p) \in \mathbb{Z}_q^n \times \mathbb{Z}_p$. Finally, [RSSS17] introduced the Middle-Product LWE (MP-LWE) variant. Given two polynomials, the middle product of two polynomials is defined as the coefficients of the middle segment of the product of the polynomials. It is the main trait of the Middle-Product LWE that directly punctures the product of two polynomials instead of taking modulo a predefined polynomial. [RSSS17] proved that MP-LWE is as hard as the P-LWE problem with any polynomial with predefined degree and form. In other words, for any polynomial f belonging to a large class, if P-LWE with respect to f is hard, then the MP-LWE is hard.

Apart from encryption schemes, a few lattice-based digital signature schemes are proposed. Generally speaking, there are two kinds of lattice based signatures, including the hash-and-

sign framework and Fiat-Shamir transformation. The hash-and-sign framework was suggested by Diffie and Hellman in their revolutionary paper [DH76]. Gentry, Peikert and Vaikuntanathan [GPV08] (GPV) adapted this framework to the lattice setting and designed a digital signature scheme according to the trapdoor of lattices and the hardness of the SIS problem. An alternative method to build a signature scheme is to construct a three-move identification scheme and then convert the identification scheme to a digital signature scheme via the Fiat-Shamir transformation [FS86, AABN02]. Lyubashevsky [Lyu09] first showed a construction based on the SIS problem. In later work, Lyubashevsky [Lyu12] introduced the rejection sampling to improve the previous scheme. The rejection rate was improved in [DDLL13] and the BLISS signature scheme was proposed. The state-of-the-art of lattice-based signature scheme is derived from the BLISS scheme.

Theoretically speaking, the aforementioned schemes admit worst-case to average-case reductions. Next we review some lattice-based schemes which do not enjoy rigorous reductions but are interesting in their own right.

In the same year when Ajtai published his breakthrough result, the NTRU [HPS98] encryption scheme was invented although the paper was published in 1998. It was the first time that ideal lattices were used in lattice based constructions. The NTRU scheme is efficient and employs small public key size because it uses the ring element as its public key. Up to now, there is no known polynomial reduction from worst-case hard lattice problems to the NTRU problem. Since its invention, the NTRU scheme remains a promising candidate against large scale quantum computers. Inspired by Ajtai's work and McEliece's code-based cryptosystem, the GGH [GGH97] public key encryption scheme was proposed in 1997. The GGH construction can be viewed as a generalization of the NTRU scheme. The original parameters of the GGH scheme were attacked [Ngu99] but not asymptotically. The idea of GGH inspires the lattice trapdoor construction, particularly the GPV trapdoor [GPV08].

Lattices have led to the construction of many different primitives, for instance, fully homo-

morphic encryption [Gen09,BV11], identity-based encryption [GPV08], attribute-based encryption [GVW13], lossy-trapdoor functions [PW08], pseudorandom functions [BPR12], multi-linear maps [GGH13,GGH15,CVW18] and indistinguishable obfuscations [GGH+13,SW14]. For some of these primitives, lattice-based designs are the only known constructions.

Another problem that is closely related to lattice-based problems is the subset sum problem. Let n and Q be two integers. The subset sum problem $SS(n, Q)$ is defined as: given a vector $\mathbf{a}$ consisting of n numbers and a target value t , search for a subset $S \subseteq \{1, \dots, n\}$ satisfying $\sum_{i \in S} a_i = t \pmod{Q}$, where $\mathbf{a}$ and t are uniformly drawn from their domain. After many years' exploring, the hardness of subset sum problem is associated with the *density* of the subset sum problem, which is defined to be $d = n/\log Q$. If the density d is larger than $n/\log^2 n$ or smaller than $1/n$, there exist probabilistic polynomial algorithms to solve the subset sum instance [LO85,Fri86,FP05,Lyu05,Sha08]. On the other hand, if the density d is a constant $O(1)$, or in the magnitude of $O(1/\log n)$, the subset sum problem is believed to be hard, and there are many different algorithms proposed to improve the time complexity in these density ranges. For any $d = o(1)$ or $d = \omega(\log n/n)$, existing algorithms can solve the subset sum problem with density d in time $2^{(n/2+o(n))/\log(1/d)}$ [KF15]. It has been shown that the problem becomes increasingly difficult as the density d tends to 1 [IN96]. For the subset sum instance with density $d = 1$, the past decade has seen much progress. The best classical algorithm runs with time complexity $\tilde{O}(2^{0.291n})$ and the same space complexity [BCJ11]. Recently, the space complexity is improved to polynomial while the time complexity is $\tilde{O}(2^{0.645n})$ [EM20]. The best quantum algorithm for the density 1 subset sum problem has complexity $\tilde{O}(2^{0.209n})$ [LL19]. In general, the subset sum problem is considered to be resistant to quantum computers.

The earliest public key encryption schemes were based on some special variants of the subset sum problem, namely subset sum instances with some structure, to support the decryption operation [MH78]. All these constructions have subsequently been broken. Readers may refer to [Odl90] for a survey on the early subset sum based public key cryptosystems. In a

different direction, Impagliazzo and Naor [IN96] presented provably-secure constructions i.e., pseudorandom generators, computationally hiding commitment schemes and universal one-way functions, building on the conjectured hardness of *random samples* of subset sum problems. The main difference between public key cryptosystems and provably-secure constructions is that the provably-secure constructions use random instances of subset sum problem, while the public key cryptosystems build on subset sum instances possessing extra structures. However, successful cryptanalytic efforts suggest that the extra structures lead to fatal drawbacks.

In 2010, Lyubashevsky, Palacio and Segev [LPS10] constructed a public key cryptosystem. The security of the public key cryptosystem is directly built on the random subset sum problem. Let $Q = q^m$. Given a random subset sum instance $(\mathbf{a}, t)$, Lyubashevsky et al. have a perceptive observation that the vector $\mathbf{a}$ can be converted to a matrix A when viewed in base q and the target t is converted to a vector $\mathbf{t}$ via its base q representation. Then the instance $\langle \mathbf{a}, \mathbf{s} \rangle = t$ is transformed to $\mathbf{t} = A\mathbf{s} + \mathbf{c} \pmod{q}$, where $\mathbf{c}$ is the carry vector when performing the computation in base q . Now the instance of subset sum $(A, \mathbf{t} = A\mathbf{s} + \mathbf{c} \pmod{q})$ has a resemblance to the Learning With Errors [Reg05] instance. In this way, the public key encryption scheme can be adapted from a LWE-based scheme. In addition, the subset sum distribution $(A, \mathbf{t})$ is pseudorandom follows from the knowledge that distinguishing $(\mathbf{a}, t)$ from uniform distribution is as hard as searching for the vector $\mathbf{s}$ [IN96]. In 2016, Faust et al. [FMV16] constructed a public key cryptosystem achieving IND-CCA with a direct reduction from the subset sum problem by employing the transformation of [LPS10].

For a comprehensive survey on lattice-based cryptography and its developments, we refer the reader to [Pei16].

1.1.3 Hash-based Cryptography

Hash-based cryptographic primitives directly rely on the security of the underlying cryptographic hash functions. Lamport [Lam79] proposed to use cryptographic hash functions to

generate digital signatures. The scheme Lamport proposed is a one-time signature scheme. In 1989, Merkle [Mer87] designed a tree based signature scheme to improve Lamport's signature scheme. It is named as the Merkle Signature Scheme(MSS) or the Merkle tree. The Merkle tree scheme allows for many signatures as the Merkle tree reduces the storage cost. The security of MSS depends only on the security of the underlying cryptographic hash function. The eXtended Merkle Signature Scheme(XMSS) [BDH11,HBG⁺18] greatly improves the speed and signature size of prior hash-based signature constructions. The main problem of hash-based signature scheme is that it is stateful. Thus, the signature needs to keep a record for all output signatures. This issue was solved in the SPHINCS [BHH⁺15] scheme. The general idea is that each time the signer takes a random key to produce the signature, it is assumed that the key is not used before with overwhelming probability. The SPHINCS scheme is later improved to SPHINCS⁺ [BHK⁺19], which is also chosen as one of the alternate candidates of NIST round three submissions [NIS20]. Hash-based signatures are the first post-quantum constructions formally defined in two RFCs [HBG⁺18,MCF19].

1.1.4 Isogeny-based Cryptography

Isogeny-based cryptography is built on the assumption that computing the isogeny between any two isogenous elliptic curves is hard. It appears that the problem of computing the isogeny between any two isogenous elliptic curves is resistant to quantum algorithm. In addition, the isogeny-based cryptography comprises primitives with smaller public key size and smaller ciphertext size. The isogeny-based cryptography started from Couveignes [Cou06], who first constructed key exchange protocols based on the more general framework *hard homogeneous spaces* and *ordinary* elliptic curves. The same idea was rediscovered by Rostovtsev and Stolbunov [RS06,Sto04]. They presented a Diffie-Hellman key exchange protocol and an identification scheme built on the isogeny assumptions. Till recently, the most efficient constructions of isogeny-based encryption scheme and identification scheme were built from *supersingular* el-

liptic curves. The isogeny-based schemes can be divided into two categories according to the underlying curves and isogeny. One is the Supersingular-Isogeny Diffie-Hellman(SIDH) protocol while the other is the *Commutative* SIDH(CSIDH). Jao and De Feo [JF11] proposed the SIDH protocol in 2011. SIDH chooses supersingular elliptic curves principally based on two reasons. First, it is easy to find a supersingular elliptic curve achieving a desired group order. Second, for supersingular elliptic curve, the endomorphism ring is non-commutative. Thus, it avoids Kuperberg's [Kup05] subexponential-time quantum attack. SIDH is the main component of the practical key exchange protocol SIKE [JAC⁺], which was chosen as one of the alternate candidates in NIST round three submissions [NIS20].

On the other hand, Castryck et al. [CLM⁺18] proposed the CSIDH in 2018. CSIDH places restrictions on the isogeny graph and supersingular elliptic curves. The curves and the isogenies are defined over the field $\mathbb{F}_p$ instead of the algebraic closure of $\mathbb{F}_p$. Thus, one can construct key exchange protocols under the hard homogeneous space framework originated from Couveignes [Cou06].

Yoo et al. [YAJ⁺17] proposed an identification scheme via adapting the SIDH [JF11] key exchange protocol. A digital signature scheme can be derived via applying the Fiat-Shamir transformation to an identification scheme. However, the signature size of the scheme by Yoo et al. is about 120KB which is huge. Later, Galbraith, Petit and Silva [GPS17] presented a similar method to construct the identification scheme. The resulting signature size is compressed to about 10KB.

Another typical kind of isogeny-based signature scheme is derived from CSIDH. The scheme was first proposed by Couveignes and Stolbunov [Cou06, Sto04]. They built a general framework to construct such signature schemes. In 2019, Feo and Galbraith [FG19] suggested a signature scheme with the name SeaSign. Beulens, Kleinjung and Vercauteren [BKV19] improved the SeaSign scheme to a scheme named CSI-Fish. The parameters of CSI-Fish are practical in the sense that it achieves the smaller signature size and public key size as compared to other NIST

quantum-safe candidates [NIS17b], but is much slower.

1.1.5 Multivariate-quadratic Cryptography

The multivariate-quadratic equation problem has been proven to be NP complete [GJ79]. Matsumoto and Imai [IM85] first considered constructing public key encryption schemes from multivariate-quadratic equations over finite fields. Shamir considered the same public key constructions defined over finite rings instead of finite fields. Encryption schemes and signature schemes are proposed based on the assumption that multivariate-quadratic equation is hard. So far, only the signature schemes survive after many years of cryptanalysis. There are two main classes of multivariate-quadratic equations based signature schemes, specifically, the small-field schemes and the big-field schemes. The former class includes the UOV, Rainbow while the latter class includes the Hidden Field Equations(HFE), Gui and MQDSS. The UOV [KPG99] signature scheme has withstood almost two decades of attacks and is therefore seen as a promising candidate. The Rainbow [DS05] scheme can be viewed as a multilayer version of UOV, in which the key size and signature size are both reduced. The HFE [Pat96] scheme was originally introduced as a public key encryption scheme. After the scheme was attacked, several HFE variants [PCG01] for signature schemes were proposed. For instance, Petzoldt et al. [PCY⁺15] presented the Gui scheme as an extension of the HFE signature scheme. The signature size is very small. As an example, for 80-bits security, the signature size is around 120 bits. The MQDSS scheme [CHR⁺16] employs provable security, thus the security of MQDSS only relies on the hardness of the multivariate quadratic equations. As a consequence, MQDSS cannot provide comparable performance with the Rainbow scheme or HFE variants.

1.2 Security of Cryptosystems

In modern cryptography, an adversary is unable to learn any information from seeing a polynomial number of valid ciphertexts. Concretely, the security is built on the indistinguishability of

ciphertexts from distinct messages. A cryptosystem achieves indistinguishable security if given a ciphertext of a message randomly chosen from two messages $\{m_a, m_b\}$, a probabilistic polynomial time(PPT) adversary is unable to determine the corresponding index of the plaintext. There are several indistinguishable security models, *indistinguishability under chosen plaintext attack(IND-CPA)*, *indistinguishability under chosen ciphertext attack(IND-CCA)*, and *indistinguishability under adaptive chosen ciphertext attack(IND-CCA2)*.

An encryption scheme achieves IND-CPA if a PPT adversary, given a ciphertext of a message randomly chosen from two messages and access to an encryption oracle, has negligible advantage over random choice. The IND-CPA security is formally defined in Section 2.6. For IND-CCA and IND-CCA2 security, the definition is similar to IND-CPA security. For IND-CCA(2) security, the adversary is also given access to a decryption oracle. In IND-CCA, the adversary is limited to only allowed to query the decryption oracle before receiving the challenge ciphertext. In IND-CCA2, the adversary is allowed to query the decryption oracle of its choice except the challenge ciphertext. In IND-CCA model, the adversary is more powerful than in the IND-CPA model.

For NIST submissions [NIS17a], the common approach to convert a IND-CPA encryption scheme or a deterministic encryption scheme is the Fujisaki-Okamoto(FO) transformation [FO99, HHK17] with the assumption of the random oracle model(ROM) [BR93] for involved cryptographic hash functions. Against class adversaries, the FO transformation employs tight security proofs. However, for quantum adversaries, the security proof does not hold. For quantum random oracle model(QROM) [BDF⁺11], the security proofs of FO transformation for deterministic encryption schemes and IND-CPA encryption schemes are completely different. For IND-CPA schemes, [KSS⁺20] provides a tight security proof with the measure-rewind-measure technique. For deterministic schemes, Bindel et al. [BHH⁺19] gives tighter security proofs for several variants of FO transformation.

1.3 Contributions of the Thesis

As we reviewed in Section 1.1, many encryption and signature schemes built from different hard mathematical problems have been proposed. There remains a large margin for improvement, both theoretically and practically. Some of the open problems and challenges include making the existing schemes more efficient, reducing an existing construction to different assumptions to achieve better parameters, adapting the constructions from one realm to another (such as between code-based and lattice-based realms), investigating the concrete security levels of existing schemes, providing an efficient implementation while withstanding side-channel attacks, etc.

In this thesis, we attempt to address some of the above problems. Concretely, we present three quantum-safe constructions, namely,

- A code-based construction to further reduce the public key size of McEliece’s original public key cryptosystem based on Goppa codes. It is reproduced from the paper [LXY19].
- A lattice-based cryptosystem constructed from polynomials that is inspired by the McEliece code-based cryptosystem. It is reproduced from the paper [LLXY20].
- A variant of subset sum problem and its applications. It is reproduced from the paper [HLY].

We next outline the motivation, techniques and results of our three contributions.

1.3.1 Contribution 1

The HQC [MAB⁺19a] construction which has reached NIST round three is an adaptation of the Ring-LWE based schemes. Recent research [BS16, CDPR16, CDW17, BBdV⁺17, DPW19] indicates that for some special ring structure, the Ideal-SVP is solvable in polynomial time by a quantum computer. Therefore, the corresponding Ring-LWE is no longer hard. The case for

the Ring-LPN is still ambiguous, namely, we are not sure whether there are weak polynomial moduli rendering the associated Ring-LPN less hard than conjectured. Therefore, we turn our attention back to the oldest McEliece scheme based on Goppa codes.

In terms of security, Goppa codes seem to be the strongest as they have withstood structural attacks since they were first proposed by McEliece. However, they suffer from the disadvantage of having large public key size.

In this thesis, we propose a new general idea to reduce the public key size of the classical McEliece cryptosystem based on Goppa codes. Our construction is a generalization of prevailing automorphism-induced Goppa codes. Different from known automorphism-induced Goppa codes, proper subsets of automorphism subgroups of the Goppa codes are used to derive a full basis of the public key from smaller subcodes via permutations. In particular, we consider additive and multiplicative automorphism subgroups to instantiate our constructions. We show that our technique can be combined with existing quasi-cyclic or quasi-dyadic constructions to further reduce the public key size. Our construction is resistant to existing structural attacks. Applying our construction to the existing scheme BigQuake [BBB⁺18b], the public key size is reduced by around half.

We generalize the approach of automorphism-induced code constructions to seek for Goppa codes with compact public key sizes. Instead of finding codes with nontrivial automorphism groups, we will construct Goppa codes that are a union of different subcodes and their permutations.

Before presenting our idea, we first review previous efforts and the corresponding attacks. One common approach adopted is to employ quasi-cyclic or quasi-dyadic Goppa codes [MB09, BCGO09] or more generally, codes with a nontrivial automorphism group [Ber00]. Indeed, in [FOP⁺16a], the authors showed that existing quasi-cyclic and quasi-dyadic constructions are induced from nontrivial automorphism subgroups of the rational function field. The main advantage to use a code with a nontrivial automorphism group $\mathcal{G}$ is that there exists a subset of

basis codewords such that this subset together with their permutations induced by $\mathcal{G}$ form the whole basis of the code. As such, this allows one to use the subset of codewords as the public key instead of the entire basis, thereby reducing the size of the public key. Nonetheless, the size reduction leads to a trade-off with respect to its resistance to structural attacks. In particular, it was shown in [FOP⁺16b, FOP⁺16a, Bar18] that the public/private key pair of such codes is equivalent to a public/private key pair of another code with smaller parameters. Algebraic cryptanalysis can then be performed on the corresponding codes with smaller parameters and successful attacks following this approach were carried out in [FOPT10, FPdP14]. The main reason that prior constructions are attacked is that prior constructions take full use of an automorphism subgroup $\mathcal{G}$. Thus, given a Goppa code employing an automorphism subgroup $\mathcal{G}$, $\mathcal{G}$ leads to a subcode of the invariant code of the Goppa code via the folding operation. Then the security of the scheme is reduced from a original Goppa code with larger parameters to a sub Goppa code with smaller parameters, thus compromising the security.

We briefly explain our constructions here. See Section 2.4 for the relevant definitions and results.

In our construction, we first fix an automorphism subgroup $\mathcal{G}$ of the affine general linear group $\text{AGL}_2(q)$. The support L of the Goppa code is the union of $\mathcal{G}$ -orbits under the action of the group $\mathcal{G}$ on $\mathbb{F}_q$. Suppose $g(x)$ is the irreducible Goppa polynomial of the desired Goppa code. The desired Goppa code is $\Gamma(L, g)$. Define the group $\mathcal{G}$ induced polynomial as

$$h(x) = \prod_{\sigma \in \mathcal{G}} g(\sigma(x)).$$

By carefully choosing the parameters of the code length, the degree of the Goppa polynomial, and the extension field degree, we can reasonably assume that h induced Goppa code $\Gamma(L, h)$ is trivial, namely, containing only the zero codeword. Let $h_\sigma(x) = h(x)/g(\sigma(x))$. Then for any $\sigma \neq id$, $\Gamma(L, h_\sigma)$ is a subcode of the desired Goppa code $\Gamma(L, g)$ as g is a factor of h_σ . Besides, for any two polynomials $h_{\sigma_i}(x)$ and $h_{\sigma_j}(x)$ with $\sigma_i \neq \sigma_j$, we prove that the induced

Goppa codes $\Gamma(L, h_{\sigma_i})$ and $\Gamma(L, h_{\sigma_j})$ are disjoint. Then, $\Gamma(L, g)$ is exactly the union of all the $\Gamma(L, h_{\sigma})$ for $\sigma \neq id$. In addition, every codeword of $\Gamma(L, h_{\sigma_i})$ can be viewed as a permutation of a codeword of $\Gamma(L, h_{s_j})$ from the property of the support set L and the group structure of $\mathcal{G}$. Thus, a systematic form of the basis of any $\Gamma(L, h_{\sigma})$ can be given as a public key of the Goppa code $\Gamma(L, g)$. Other basis rows of $\Gamma(L, g)$ can be obtained from the row permutations. Hence, the generator matrix of $\Gamma(L, g)$ can be obtained from $\Gamma(L, h_{\sigma})$ via permutations. Thus, the public key size is roughly reduced by a $|\mathcal{G}|$ factor.

In our construction, an attacker is unable to apply the folding operation to get a subcode of the invariant code for two reasons. First, we use a proper subset of an automorphism subgroup acting on the code instead of the automorphism subgroup itself. Thus the folding operation will not lead to an invariant code as the group elements are not fully used in the construction. Second, in our construction, the automorphism subgroup induced Goppa code is trivial, namely, only containing the zero codeword. Thus, even if an adversary can obtain the invariant subcode, the invariant subcode only contains the zero codeword. Thus, it will not lead to any meaningful information for the adversary.

In our results, we first provide conditions for permutations of codewords to lie in the code. We then consider two instantiations, the partial quasi-cyclic construction and the partial quasi-dyadic construction. Further, our construction can be combined with existing techniques to further reduce the public key size. Concretely, we show that our construction can reduce the public key sizes in BigQuake [BBB⁺18b] by around half.

1.3.2 Contribution 2

There have been various attempts to design lattice-based schemes inspired from code-based constructions and vice versa. For instance, the LWE problem is a q -ary analogue of the LPN problem. The earliest cyclic-SIS [Mic02] problem is also partly inspired by the cyclic code structure. Thus, we may expect to design new lattice-based schemes from code related con-

structions.

The GGH encryption scheme [GGH97] is an analog of the famous code-based encryption scheme—McEliece encryption. However, it was shown in [Ngu99] that the structure of the error provided an inherent weakness and together with the embedding technique, this weakness can be exploited to attack the GGH instances. Even though suggestions were put forth in [Ngu99] to mend the scheme, the corresponding parameters will make the scheme impractical for use.

Nonetheless, the ideas underpinning the GGH scheme remained interesting and motivated other lattice-based trapdoor functions including those based on LWE/SIS problems [GPV08]. Besides, it is interesting that the code-based analog McEliece cryptosystem is still secure against quantum computers. As such, one might hope to improve the lattice-based GGH scheme such that it provides the same level of security but is more efficient than the McEliece scheme.

In this thesis, we design a new class of trapdoor functions to solve the bounded distance decoding problems in lattices with specific structures. The lattices are constructed from polynomials. We prove that the lattices we construct exhibit some good properties, for example, the determinant of the lattice and the length of the shortest nonzero vector are related to the degrees of the polynomials used to construct the lattice. Without knowledge of the polynomials underlying the lattices, the bounded distance decoding problem is hard to solve, while the bounded distance decoding problem can be efficiently solved once the polynomials are known. Thus, in the construction of our trapdoor functions, the underlying polynomial serves as a trapdoor. We then estimate the security of the trapdoor function based on combinatorial attacks and state-of-the-art lattice attacks. The trapdoor function we proposed is used to construct efficient quantum-secure public key encryption schemes with acceptable public key size.

In our construction, we seek to design new trapdoor functions in which the function inversion involves solving the Bounded Distance Decoding problem (BDD), one of the well-known hard lattice problems. Our construction is primarily inspired by the GGH construction [GGH97] and the McEliece code-based cryptosystem [McE78]. In particular, our scheme can be viewed

as the lattice analog of the Goppa-code based McEliece cryptosystem. Like these schemes, our function involves constructing a point that is sufficiently close to a certain point in a lattice determined by the input. Hence, inverting this function will require one to solve BDD for a related lattice.

Here we briefly sketch the construction of the lattice we used. More precisely, we choose n distinct elements $\alpha_1, \alpha_2, \dots, \alpha_n$ of $\mathbb{F}_q$ and t distinct monic irreducible polynomials $c_1(x), c_2(x), \dots, c_t(x)$ of degree d_0 such that $\gcd(\prod_{i=1}^n (x - \alpha_i), \prod_{i=1}^t c_i(x)) = 1$. An integer point $(a_1, a_2, \dots, a_n) \in \mathbb{Z}^n$ is a lattice point if and only if $\prod_{i=1}^n (x - \alpha_i)^{a_i} \equiv 1 \pmod{c(x)}$, where $c(x) = \prod_{i=1}^t c_i(x)$. Then a basis of this lattice can be computed efficiently. We prove that the determinant and the length of shortest nonzero vector of the lattice are functions of the degrees of the polynomials used to construct the lattice. We show that, given q, n and certain range of $d = td_0$, and the basis of this lattice, the embedding technique does not work well to tackle the BDD problem of this lattice. On the other hand, with information on the point set $\{\alpha_1, \alpha_2, \dots, \alpha_n\}$ and the polynomial set $\{c_1(x), c_2(x), \dots, c_t(x)\}$, we are able to efficiently solve the BDD problem of this lattice. With the structure of the trapdoor function, we can design an encryption scheme that is similar to the one in the GGH encryption scheme.

1.3.3 Contribution 3

The security of Ajtai-Dwork [AD97] cryptosystem is established under the assumption that the worst-case gap unique shortest vector problem for lattices is hard. The hardness of the famous LWE [Reg05] problem is built from the gap unique shortest vector problem as well. There is an algorithm that reduces the random subset sum problem to the gap unique shortest vector problem. It follows that the security of the cryptosystems based on the gap unique shortest vector problem problems can be derived from the random subset sum problems in an indirect way and thus, the connection may not be tight.

In this thesis, we propose a variant of the subset sum (SS) problem, called the *rounded subset*

sum (RSS) problem. First, we prove the rounded subset sum distribution is pseudorandom if the underlying subset sum problem is hard. Then we construct a public key cryptosystem. The semantic security of the cryptosystem is built on the rounded subset sum problem. Next we show the robustness of the rounded subset sum problem in the presence of auxiliary input of the secret. According to the robustness results of RSS, we demonstrate a public key cryptosystem that is resilient to adaptive leakage of the secret key, which addresses a problem explicitly left by Lyubashevsky, Palacio, and Segev [LPS10]. The assumption we used to construct adaptive leakage-resilient scheme is weaker than the assumption underlying existing lattice-based leakage-resilient schemes such as Dodis et al. [DGK⁺10]. We present other applications of the rounded subset sum problem as well.

Note that the rounded subset sum problem and its applications are directly built on the random subset sum problem instances.

Rounded Subset Sum

For positive integers n, Q , the subset sum problem $SS(n, Q)$ is defined as, given a public vector $\mathbf{a} \in \mathbb{Z}_Q^n$ and a target $t \in \mathbb{Z}_Q$, find a secret vector $\mathbf{s} \in \{0, 1\}^n$ such that $\langle \mathbf{a}, \mathbf{s} \rangle = t$. To ensure that the problem is hard, the modulus Q is always set to be an exponential of n . We next consider more subset sum instances sharing the same secret. It is easy to see that the subset sum problem is no longer hard once the number of instances becomes n . But what if the number of instances is smaller than n ? At the first glance, this problem is similar to the SIS problem in the form sense. The SIS problem is that for a matrix uniformly sampled matrix $A \in \mathbb{Z}_q^{m \times n}$ and a short vector $\mathbf{s}$, given $(A, A\mathbf{s})$, it is hard to seek out the secret vector $\mathbf{s}$. The SIS problem is proven to be as hard as n -dimensional worst-case hard lattice problems. From the perspective of SIS problem, the subset sum problem becomes as hard as 1-dimensional worst-case hard lattice problems, which cannot provide anything useful to the hardness of subset sum problem as 1-dimensional lattice problems are easy. Thus, our consideration to directly relate to subset

sum problem instead of lattice problems is interesting in its own right.

We consider a variant of this problem. In particular, we consider a rounded form of each instance with respect to a much smaller modulus. Intuitively, given n rounded targets, recovering the secret vector does not appear to be as easy as the original subset sum problem. We state the rounded subset sum problem as follows. Given appropriate parameters (n, m, q, p) , $(\mathbf{a}_1, \dots, \mathbf{a}_m) \in (\mathbb{Z}_q^n)^m$ and $(t_1 = \lceil \langle \mathbf{a}_1, \mathbf{s} \rangle \rceil_p, \dots, t_m = \lceil \langle \mathbf{a}_m, \mathbf{s} \rangle \rceil_p)$, the rounded subset sum problem seeks to find $\mathbf{s}$. Here, $\lceil x \rceil_p$ denotes the integer nearest to $\frac{px}{q}$. Let $\mathbf{a}_i^T$ be the i -th row of a matrix $A \in \mathbb{Z}_q^{m \times n}$. We often write the rounded subset sum instance in the matrix form $(A, \mathbf{t} = \lceil A\mathbf{s} \rceil_p) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_p^m$.

We prove that there is a probabilistic polynomial reduction from the subset sum problem to the rounded subset sum problem. Concretely, we prove that if the subset sum problem $\text{SS}(n, q^m)$ is hard, the distribution $(A, \mathbf{t}) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_p^m$ of the rounded subset sum problem $\text{RSS}(n, m, q, p)$ is pseudorandom. Note that the parameters in our reduction are preserved, namely, the problem with m rounded inner product instances from a modulus q to p is as hard as the subset sum problem with modulus q^m . According to the hardness result of the subset sum problem, m can be taken to be larger than n . According to our reduction, the modulus q and p in the rounded subset sum problem can be $\text{poly}(n)$. With the reduction, we construct a public key encryption scheme which achieves semantic security.

We next outline our reduction proof. The reduction consists of two steps. In step one, we prove that if the subset sum problem $\text{SS}(n, q^m)$ is hard, then the rounded subset sum problem $\text{RSS}(n, m, q, p)$ is hard. In step two, we prove that if the rounded subset sum instance $\text{RSS}(n, m, q, p)$ is hard to invert, then the distribution $(A, \mathbf{t}) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_p^m$ of $\text{RSS}(n, m, q, p)$ is pseudorandom. Combining the two steps leads to a reduction from the subset sum problem to the pseudorandomness of the rounded subset sum distribution. Concretely, in step one, we prove that if there exists an algorithm $\mathcal{A}$ inverting the rounded subset sum $\text{RSS}(n, m, q, p)$, there is an algorithm to invert the subset sum $\text{SS}(n, q^m)$ via calling $\mathcal{A}$. Given an instance $(\mathbf{a}, t) \in \mathbb{Z}_{q^m}^n \times \mathbb{Z}_{q^m}$

of the subset sum problem $\text{SS}(n, q^m)$, we transform the subset sum instance to a matrix-vector representation of subset sum $(A, \mathbf{t} = A\mathbf{s} + \mathbf{f}) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m$, where the vector $\mathbf{f}$ is determined by A and $\mathbf{s}$ and bounded. Next we use the noise flooding technique from lattice-based cryptography to remove the dependence. The noise flooding technique says that for a bounded vector $\mathbf{f}$ and a wide enough Gaussian distributed vector $\mathbf{e}$, the value $\mathbf{f} + \mathbf{e}$ is essentially independent of $\mathbf{f}$, namely, the vector $\mathbf{e}$ floods the vector $\mathbf{f}$. Then the instance becomes $(A, A\mathbf{s} + \mathbf{f} + \mathbf{e})$. The vector $\mathbf{f} + \mathbf{e}$ is close to a Gaussian distribution and it can be considered to be independent from A and $\mathbf{s}$. Next we use a result from [BGM⁺16] to prove that when feeding $(A, \lceil A\mathbf{s} + \mathbf{f} + \mathbf{e} \rceil_p)$ to the algorithm $\mathcal{A}$, it outputs the desired secret vector $\mathbf{s}$. This establishes the reduction of the first step. In step two, we prove that if there is a distinguisher $\mathcal{D}$ that is able to distinguish a rounded subset sum distribution from a uniform distribution, then there is an algorithm to find the secret vector of the rounded subset sum problem. Building a series of hybrid games, we convert the distinguisher $\mathcal{D}$ to a predictor for $\langle \mathbf{a}, \mathbf{s} \rangle$, where $\mathbf{a}$ is drawn from the uniform distribution. Next we convert such a predictor to a distinguisher $\mathcal{D}'$ for the distribution $(\mathbf{a}, \langle \mathbf{a}, \mathbf{s} \rangle)$. We recall a Goldreich-Levin theorem defined over a large field, which says that for any function f , given $f(\mathbf{s})$, and a vector $\mathbf{a}$ drawn from the uniform distribution, if there exists a distinguisher $\mathcal{B}$ that distinguishes $(\mathbf{a}, \langle \mathbf{a}, \mathbf{s} \rangle)$ from the uniform distribution, then there exists an inverter to invert the function f . Combining $\mathcal{D}'$ with the variant of the Goldreich-Levin theorem over a large field leads to an algorithm to produce the vector $\mathbf{s}$ with non-negligible probability. This finishes step two.

Rounded Subset Sum Problem with Weak and Leaky Secrets

We show that our newly proposed rounded subset sum problem is still hard with weak and leaky secrets. The critical point is to prove that given auxiliary inputs of the secrets and the rounded subset sum instances, the secret vector still has enough entropy. It means that we need to compute the conditional entropy of $\mathbf{s}$ given A , rounded subset sum instances $\lceil A\mathbf{s} \rceil_p$,

and an auxiliary input $\mathbf{aux}$, namely, $H_\infty(\mathbf{s}|A, \lceil A\mathbf{s} \rceil_p, \mathbf{aux})$. Note that if A is *fully uniformly and randomly* sampled from $\mathbb{Z}_q^{m \times n}$, the secrets will not have enough entropy since $\mathbf{s}$ is almost determined given the rounded subset sum instances. Thus, we need to change A to a different form such that A is still indistinguishable from the uniform distribution and the secret vector $\mathbf{s}$ still has enough entropy given $A, \lceil A\mathbf{s} \rceil_p$ and $\mathbf{aux}$.

Generally speaking, [GKPV10, AKPW13] provide a method to write A in the form $BC + F \in \mathbb{Z}_q^{m \times n}$ under the LWE assumption. We follow such an idea to replace A to compute $H_\infty(\mathbf{s}|A, \lceil A\mathbf{s} \rceil_p, \mathbf{aux})$. If we directly set $A = BC + F$, where $B \in \mathbb{Z}_q^{m \times \ell}, C \in \{0, 1\}^{\ell \times n}, F \in \mathbb{Z}_q^{m \times n}$ and $\ell < \min(m, n)$, then $A\mathbf{s} = BC\mathbf{s} + F\mathbf{s}$. As $\ell < \min(m, n)$, $C\mathbf{s}$ is a compression of $\mathbf{s}$. $F\mathbf{s}$ is small and it will not influence much of the entropy. With such a replacement of A , $\mathbf{s}$ still has enough entropy given the conditions $B, C, C\mathbf{s}$ and $F\mathbf{s}$.

For the case of subset sum problems, the matrix A can be changed to the form $A := B \odot C = BC + F \in \mathbb{Z}_q^{m \times n}$ under the assumption that n instances of subset sum $\text{SS}(\ell, q^m)$ are hard, where $\odot$ denotes a matrix-vector representation of the subset sum problem (See Section 2.5 for relevant definitions). It seems that based on the $\text{SS}(\ell, q^m)$ problem, we can naturally replace A with $B \odot C = BC + F$. However, every entry f_{ij} of F is determined by the previous entry $\langle \mathbf{b}_{i-1}, \mathbf{c}_j \rangle$ of BC , i.e., F is dependent on BC , where $\mathbf{b}_i$ is the i -th row of B and $\mathbf{c}_j$ is the j -th column of C . In the proof, we need to decouple the F reliance on B and C . We use $(B \odot C + Z) \in \mathbb{Z}_q^{m \times n}$ instead of $B \odot C$, where Z is independently drawn from a discrete Gaussian distribution. Then by the technique of noise flooding, the distributions of $B \odot C + Z = BC + F + Z$ and $BC + Z$ are statistically close. So far, we removed the influence of F and can use $BC + Z$ to analyze the conditional entropy. Besides, we also need to prove $B \odot C + Z$ is indistinguishable from the random A . Since $B \odot C$ is indistinguishable from random A based on the $\text{SS}(\ell, q^m)$ assumption and Z is independent of $B \odot C$, $B \odot C + Z$ is also indistinguishable from random A .

Based on the above chosen structure of A , we can compute the conditional entropy as follows. Define $\mathbf{T} = \{(i, \lceil ((BC + Z)\mathbf{s})_i \rceil_p) : \lceil ((BC + Z)\mathbf{s})_i \rceil_p \neq \lceil (BC\mathbf{s})_i \rceil_p\}$, where $(\cdot)_i$ represents the i -

th entry of the vector. Given B, C, Z, Cs, T , it is easy to obtain the whole vector $\lceil (BC + Z)s \rceil_p$. Then we have

$$H_\infty(s|aux, A, \lceil As \rceil_p) \geq H_\infty(s|aux, B, C, Z, Cs, T).$$

In the entropy expression of the above inequality, B, C, Z are independent of s and Cs is a compression of s . Thus, next we need to bound the entropy loss caused by T . The events $\lceil (BC + Z)s \rceil_p \neq \lceil BCs \rceil_p$ for $i \in [m]$ represent that $(Zs)_i$ will leak some bits of s , but the remaining entries of Zs will not influence the entropy.

The distance between $(BCs)_i$ and $\frac{q}{p} \lceil (BCs)_i \rceil_p$ is within $\frac{q}{2p}$. After adding $(Zs)_i$, if the distance between $(BCs)_i + (Zs)_i$ and $\frac{q}{p} \lceil (BCs)_i \rceil_p$ is larger than $\frac{q}{2p}$, $\lceil (BC + Z)s \rceil_p$ will not equal to $\lceil (BCs)_i \rceil_p$. Since Z is from the discrete Gaussian distribution $\mathcal{D}_{\mathbb{Z}, r}^{m \times n}$, every entry of Zs is bounded by $rn\sqrt{n}$. Hence, for every $i \in [m]$, the probability of $\lceil (BC + Z)s \rceil_p \neq \lceil (BCs)_i \rceil_p$ is at most $\frac{\frac{q}{2p} - (\frac{q}{2p} - |(Zs)_i|)}{\frac{q}{2p}} = \frac{2p|(Zs)_i|}{q}$ which is smaller than $\frac{2prn\sqrt{n}}{q}$. We can choose q such that $q > 2prmn\sqrt{n}$. The probability is at most $\frac{1}{m}$ and the expectation is smaller than 1. Then applying the Chernoff bound we can show that the number of such entries is at most λ with overwhelming probability. So the total leakage bits of those $(Zs)_i$ is at most $\lambda(\log m + \log p) \leq \lambda \log q$. In addition, the leakage bits caused by Cs is at most $\ell \log q$ as C is sampled from $\{0, 1\}^{\ell \times n}$. Therefore,

$$H_\infty(s|aux, A, \lceil As \rceil_p) \geq H_\infty(s|aux, Cs) - \lambda \log q \geq H_\infty(s|aux) - (\lambda + \ell) \log q,$$

and $H_\infty(s|A, \lceil As \rceil_p) \geq n - (\lambda + \ell) \log q$. By setting n such that $n \geq (\lambda + \ell) \log q + 2\lambda$, the secret vector s still has enough conditional entropy.

After achieving the high entropy with leaky secrets, we can prove the rounded subset sum problem with weak and leaky secrets is still hard based on $SS(\ell, q^m)$ assumption via a sequence of hybrid games.

Leakage-Resilient Secure Public Key Encryption

We construct a public key cryptosystem which can achieve leakage-resilient security in adaptive scenarios. We use the security model from [AGV09]. Leakage-resilient security means that when part of the secret is leaked to the adversary, the scheme is still secure. This security notion is meaningful in the real world such as the adversaries can obtain partial information about secrets by side-channel attacks. In [LPS10], the non-adaptive leakage-resilient PKE scheme no longer uses the subset sum problem instance as the public key, but uses an extractor As . In the security proof, to replace the public key to a uniform instance given the auxiliary input $h(s)$ by the leftover hash lemma, $h(s)$ needs to be independent of the public key. Consequently, the scheme can only achieve non-adaptive leakage-resilient security.

Here we outline the idea of our adaptive leakage-resilient proof.

The ciphertext structure of our construction is

$$(\text{pk}, \text{ct}_1, \text{ct}_2) = ((A, \mathbf{b} = \lceil As \rceil_p), \lceil A^T \mathbf{r} \rceil_p, \langle \mathbf{r}, \mathbf{b} \rangle + \mu \lceil p/2 \rceil).$$

Then the adversary's view is $(\text{pk}, \text{ct}_1, \text{ct}_2, h(\text{pk}, \text{sk}))$. In the security proof, we first replace the challenge ciphertexts $(\text{ct}_1, \text{ct}_2)$ which is produced by using the secret key rather than the encryption procedure. Now the adversary's view is

$$((A, \mathbf{b} = \lceil As \rceil_p), \lceil A^T \mathbf{r} \rceil_p, \langle \lceil A^T \mathbf{r} \rceil_p, \mathbf{s} \rangle + \langle \mathbf{r}, \mathbf{e}_1 \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle + \mu \lceil p/2 \rceil, h(\text{pk}, \text{sk}))$$

where $\mathbf{e}_1 = \lceil As \rceil_p - \frac{p}{q} \lceil As \rceil_p$ and $\mathbf{e}_2 = \lceil A^T \mathbf{r} \rceil_p - \frac{p}{q} A^T \mathbf{r}$ are the deterministic errors of rounding. This is identical to the original view of adversaries. Second, the ciphertext ct_1 can be replaced by a uniform vector in $\mathbb{Z}_p^m$ from the weak of and leaky property of RSS. Then the view of adversary becomes

$$((A, \mathbf{b} = \lceil As \rceil_p), \mathbf{u}, \langle \mathbf{u}, \mathbf{s} \rangle + \langle \mathbf{r}, \mathbf{e}_1 \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle + \mu \lceil p/2 \rceil, h(\text{pk}, \text{sk})).$$

From the weak and leaky property, the view of adversary is computational indistinguishable from the previous view. Third, the matrix A can be replaced by $\hat{A} = B \odot C + Z$ from the subset sum assumption. Then the adversary's view turns out to be

$$((\hat{A}, \mathbf{b} = \left\lceil \hat{A}\mathbf{s} \right\rceil_p), \mathbf{u}, \langle \mathbf{u}, \mathbf{s} \rangle + \langle \mathbf{r}, \mathbf{e}_1 \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle + \mu \lceil p/2 \rceil, h(\widehat{\mathbf{pk}}, \mathbf{sk})),$$

where $\widehat{\mathbf{pk}}$ is the matrix A replaced by $\hat{A}$ in $\mathbf{pk}$. Directly from the subset sum assumption, the view is computational indistinguishable from the previous view. Fourth, as the entropy of $\mathbf{sk}$ is enough given $\langle \mathbf{r}, \mathbf{e}_1 \rangle$, $\langle \mathbf{e}_2, \mathbf{s} \rangle$, $h(\widehat{\mathbf{pk}}, \mathbf{sk})$ and $\left\lceil \hat{A}\mathbf{s} \right\rceil_p$ (the secrets of the rounded subset sum problem with weak and leaky secrets still have high entropy), we can use the leftover hash lemma to statistically replace $\mathbf{ct}_2$ to a uniform element in $\mathbb{Z}_p$. Now the ciphertext has no information about the message. Finally, by changing $\hat{A}$ back to the random A , we obtain the leakage-resilient security against adaptive memory attacks.

1.4 Organization of the Thesis

The thesis is organized as follows. Chapter 2 presents the preliminaries of the thesis, including lattices, Goppa codes, the subset sum problem, probability distribution results, security definitions of cryptosystems, GGH and McEliece cryptosystem. In Chapter 3, we present the automorphism-induced Goppa codes constructions via permutations, which leads to a reduced public key size. In Chapter 4, we show a lattice-based public key cryptosystem constructed from polynomials. In Chapter 5, a variant of the subset sum problem and its applications are presented. In Chapter 6, we conclude the thesis with some interesting problems.

2. PRELIMINARIES

2.1 Notations

We use λ to represent our security parameter. Given two probability distributions P and Q , we use $\text{SD}(P; Q)$ to represent the statistical distance of P and Q . If $\text{SD}(P; Q)$ is a negligible function of λ , then we write $P \approx_s Q$. By abuse of notation, we use $P \approx_c Q$ to represent that P and Q are computational indistinguishable based on some assumption. Let $\lceil x \rceil$ be the closest integer to x . We use $\|\mathbf{x}\|_\infty = \max_i |x_i|$. We use $\text{negl}, \text{poly}, \text{PPT}$ to denote negligible function, polynomial function, and probabilistic polynomial time, respectively.

2.2 Probability Distribution Results

2.2.1 Min-entropy

We recall the definitions of min-entropy and conditional min-entropy.

Given a random variable P , the *min-entropy* of P is defined as

$$H_\infty(P) := -\log(\max_p \Pr[P = p]).$$

Given two random variables P and Q , the *conditional min-entropy* of P given Q is defined as

$$H_\infty(P|Q) := -\log(\mathbb{E}_{q \leftarrow Q}[\max_x \Pr[P = x|Q = q]]) = -\log(\mathbb{E}_{q \leftarrow Q}[2^{-H_\infty(P|Q=q)}]).$$

Next we recall the definition of smooth min-entropy, proposed by Renner and Wolf [RW04].

Definition 2.2.1. For a random variable P , the ϵ -smooth min-entropy of P is defined as

$$H_\infty^\epsilon(P) := \max_{X: \text{SD}(P; X) \leq \epsilon} H_\infty(X).$$

Similarly, given two random variables P and Q , the ϵ -smooth conditional min-entropy of P given Q is defined as

$$H_\infty^\epsilon(P|Q) := \max_{(X,Y): \text{SD}((P,Q); (X,Y)) \leq \epsilon} H_\infty(X|Y).$$

Here smooth means a random variable is smoothly replaced by a statistically close random variable.

We recall the leftover hash lemma in the entropy form.

Lemma 2.2.2 ([HILL99, DORS08]). Let P be a probability distribution over $\mathbb{Z}_q^n$ with min-entropy k . Let $\epsilon > 0$ and $\ell \leq (k - 2 \log \frac{1}{\epsilon}) / \log q$. Then

$$\text{SD}((C, C \cdot \mathbf{s}); U(\mathbb{Z}_q^{\ell \times n} \times \mathbb{Z}_q^\ell)) \leq \epsilon,$$

where C is uniformly sampled from $\mathbb{Z}_q^{\ell \times n}$, $\mathbf{s} \in \mathbb{Z}_q^n$ is drawn from the distribution P , and $U(\mathbb{Z}_q^{\ell \times n} \times \mathbb{Z}_q^\ell)$ is the uniform distribution defined over the domain.

2.2.2 Discrete Gaussian Distribution

Given $\mathbf{c} \in \mathbb{R}^n$ and $s > 0$, $\forall \mathbf{x} \in \mathbb{R}^n$, the Gaussian function for $\mathbf{x}$ is defined as $\rho_{s,\mathbf{c}}(\mathbf{x}) := e^{-\pi(|\mathbf{x}-\mathbf{c}|/s)^2}$.

Given $s > 0$, $\mathbf{c} \in \mathbb{R}^n$, the density function of the discrete Normal distribution defined over $\mathbb{Z}^n$ is defined as $\forall \mathbf{x} \in \mathbb{Z}^n$, $\mathcal{D}_{\mathbb{Z},s,\mathbf{c}}^n(\mathbf{x}) = \frac{\rho_{s,\mathbf{c}}(\mathbf{x})}{\rho_{s,\mathbf{c}}(\mathbb{Z}^n)}$, where $\rho_{s,\mathbf{c}}(\mathbb{Z}^n) = \sum_{\mathbf{y} \in \mathbb{Z}^n} \rho_{s,\mathbf{c}}(\mathbf{y})$. The parameter s or $\mathbf{c}$ are omitted if $s = 1$ or $\mathbf{c} = \mathbf{0}$.

2.2.3 Some Results About Bounds

Lemma 2.2.3 (Chernoff bound [MU05]). *Let $X_1, \dots, X_n$ be independent Bernoulli random variables. Define $X = \sum_{i=1}^n X_i$ and $\mu = \mathbb{E}[X]$. For any $R \geq 6\mu$, $\Pr[|X| \geq R] \leq 2^{-R}$.*

Lemma 2.2.4 (Hoeffding bound [Hoe63]). *Let n be a positive integer. Let $X_1, \dots, X_n$ be independent random variables, each ranging over $[a_i, b_i]$ and let $X = \sum_{i=1}^n X_i$. For any $\sigma > 0$,*

$$\Pr[|X - \mathbb{E}[X]| \geq \sigma] \leq 2e^{-\frac{2\sigma^2}{\sum_{i=1}^n (b_i - a_i)^2}}.$$

Next we recall the bound of the discrete Normal distribution with standard deviation σ .

Lemma 2.2.5. [Lyu12, Lemma 4.4(1)] *Let D_σ be the discrete Normal distribution defined over $\mathbb{Z}$ and σ be the standard deviation. For any $k > 0$, $\Pr_{z \leftarrow D_\sigma}[|z| > k\sigma] \leq 2e^{-k^2/2}$.*

Lemma 2.2.6 ([GKPV10]). *Let $\sigma > 0$ and $y \in \mathbb{Z}$ be arbitrary. The statistical distance between the distribution $\mathcal{D}_{\mathbb{Z}, \sigma}$ and $\mathcal{D}_{\mathbb{Z}, \sigma} + y$ is at most $\frac{|y|}{\sigma}$.*

2.2.4 Rényi Divergence

Rényi divergence is used to measure the closeness between two probability distributions.

Definition 2.2.7. *Let P and Q be two probability distributions. The power of their Rényi divergence is $\text{RD}_2(P||Q) = E_{a \leftarrow P} \left[\frac{\Pr[P=a]}{\Pr[Q=a]} \right]$.*

We will use the following property of Rényi divergence in Lemma 5.2.6.

Lemma 2.2.8. *Given any two distributions P and Q , for an event E ,*

$$\Pr[Q \in E] \geq \frac{\Pr[P \in E]}{\text{RD}_2(P||Q)}.$$

It can be directly proved by applying the Cauchy-Schwartz inequality. The RD_2 is sufficient for our reductions. Readers may refer to references [vEH14, BLR⁺18] for more details.

2.3 Lattices

In this section, we briefly review some of the important definitions and notions on lattices. We refer the reader to [MG02, BBD08] for more background materials.

Let n be a positive integer. By usual convention, we will represent vectors in $\mathbb{R}^n$ in the row form. For $\mathbf{x} = (x_1, \dots, x_n) \in \mathbb{R}^n$, denote by $\|\mathbf{x}\|$ the Euclidean norm of $\mathbf{x}$, that is, $\|\mathbf{x}\| = \sqrt{\sum_{i=1}^n x_i^2}$.

Lattice: A lattice $\mathcal{L}$ is a discrete additive subgroup of $\mathbb{R}^n$. Concretely, for $m \leq n$, let $\{\mathbf{b}_1, \dots, \mathbf{b}_m\}$ be m linearly independent vectors in $\mathbb{R}^n$. Then a lattice $\mathcal{L}$ is a set

$$\{a_1 \mathbf{b}_1 + a_2 \mathbf{b}_2 + \dots + a_m \mathbf{b}_m : a_i \in \mathbb{Z}, i = 1, 2, \dots, m\}.$$

m is called the *dimension* or *rank* of the lattice. If $m = n$, then $\mathcal{L}$ is said to have full rank. In this work, we will only focus on full-rank lattices. Further, $B = \{\mathbf{b}_1, \dots, \mathbf{b}_m\}$ is called a *basis* of $\mathcal{L}$. Let M be the m by n matrix with rows $\mathbf{b}_i$, $i = 1, 2, \dots, m$. Then, the *determinant* of $\mathcal{L}$ (or the volume of $\mathcal{L}$) is given by $\det(\mathcal{L}) = \text{vol}(\mathcal{L}) = \sqrt{\|MM^T\|}$.

n -Ball: For $r \in \mathbb{R}$, let $B_n(r) = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\| \leq r\}$ denote the n -dimensional ball centered around the origin with radius r . The volume of $B_n(r)$ is given by $V_n(r) = \frac{\pi^{n/2} r^n}{\Gamma(n/2+1)}$, where $\Gamma(\cdot)$ is the Gamma function.

Short vectors of a lattice: As a lattice is a discrete subgroup of $\mathbb{R}^n$, the set of all their Euclidean norms forms a discrete subset of $\mathbb{R}$. Hence, each lattice $\mathcal{L}$ has a nonzero point such that its norm is the minimum. We denote this minimum norm by $\lambda_1(\mathcal{L})$. More generally, for $i = 1, 2, \dots, n$, $\lambda_i(\mathcal{L})$ denotes the smallest radius r such that the ball $B_n(r)$ contains i linearly independent points in $\mathcal{L}$.

Gaussian heuristic: The Gaussian Heuristic estimates the number of lattice points in certain sets. Let $\mathcal{L}$ and S be a full-rank lattice and a connected n -dimensional subset, respectively. Then the number of lattice points in S is approximated by $\text{vol}(S)/\det(\mathcal{L})$. This leads to

the following Gaussian heuristic estimate on the shortest vector $\lambda_1(\mathcal{L})$ for a random lattice $\mathcal{L}$: $\lambda_1(\mathcal{L}) \approx V_n(1)^{-1/n} \det(\mathcal{L})^{1/n} \approx \sqrt{n/2\pi e} \det(\mathcal{L})^{1/n}$.

Lattice reduction: Given a basis of a lattice $\mathcal{L}$, one can construct a new basis by multiplying the matrix formed by the basis vectors with unimodular integer matrices, that is, integer matrices with determinant ± 1 . In general, one often looks for a basis with short vectors or nearly orthogonal vectors. There are various algorithms to reduce a basis of a lattice into a basis of better quality. Well-known reduction algorithms include the LLL algorithm [LLL82] and the BKZ algorithm [SE94]. In the BKZ algorithm, one essentially tries to find short vectors in the sub-lattice formed by sub-blocks of basis vectors. In fact, the LLL algorithm can be viewed as a special case of the BKZ algorithm where we work with pairs of vectors each time. Evidently, a BKZ algorithm with a bigger block size produces a basis with shorter vectors but this is achieved at the expense of a longer running time.

In our security analysis, we consider the BKZ-2.0 algorithm [Che13] with better practical performance. The BKZ- β algorithm attempts to find the shortest vector by repeatedly calling an SVP oracle in a β dimensional sublattice. It was shown that after a polynomial number of calls to the SVP oracle, the quality of the resulting basis will not be much improved [HPS11]. The basis obtained from BKZ- β reduction is called the BKZ- β reduced basis. In the Hermite-factor model, we assume that the BKZ- β reduced basis contains a vector of length $\|\mathbf{b}_1\| = \delta^n \times \text{Vol}(\mathcal{L})^{1/n}$ with

$$\delta = (((\pi\beta)^{1/\beta})/(2\pi e))^{1/(2(\beta-1))}.$$

We model the lengths of the vectors in the BKZ- β reduced basis using the Geometric Series Assumption as follows.

Assumption 2.3.1 (Geometric Series Assumption [Sch03]). *After lattice reduction, the Gram-Schmidt vectors of the basis satisfy the condition $\|\mathbf{b}_i^*\| = \alpha^{i-1} \times \|\mathbf{b}_1\|$ for some $0 < \alpha < 1$.*

Combining the root-Hermite factor $\|\mathbf{b}_1\| = \delta^n \times \det(\mathcal{L})^{1/n}$ with the GSA assumption, we

get $\alpha = \delta^{-2n/(n-1)}$ and then $\|\mathbf{b}_i^*\| \approx \delta^{n-2i-1} \times \det(\mathcal{L})^{1/n}$.

In this work, the cost of BKZ algorithm with blocksize β in an n dimensional lattice will be taken as [Alb17, AGVW17]

$$8n2^{0.292\beta+16.4}. \quad (2.1)$$

Shortest Vector Problem (SVP): Given a lattice $\mathcal{L}$, the *shortest vector problem* (SVP) seeks a nonzero point $\mathbf{v}$ in $\mathcal{L}$ such that $\|\mathbf{v}\| = \lambda_1(\mathcal{L})$. For low dimensions, some proposed approaches to solve SVP include computing the Voronoi cell of the lattice and sieving (see [HPS11] for details) as well as enumeration methods [Kan83, GNR10]. However, these methods have complexity at best exponential in the lattice dimension n . To date, SVP has been solved in a 155-dimensional lattice (<https://www.latticechallenge.org/svp-challenge/index.php>). Several variants of SVP have been proposed such as the *unique SVP* (uSVP). Specifically, the unique SVP with factor γ seeks for the shortest vector of a lattice given that the length of the second shortest vector $\lambda_2(\mathcal{L})$ is at least γ times of the length of the shortest vector $\lambda_1(\mathcal{L})$.

Closest Vector Problem (CVP)/Bounded Distance Decoding Problem (BDD):

A closely related problem to SVP is the *closest vector problem* (CVP). Given a target vector $\mathbf{t} \in \mathbb{R}^n$, CVP seeks a vector $\mathbf{v} \in \mathcal{L}$ that minimizes $\|\mathbf{t} - \mathbf{v}\|$. Further, when $\|\mathbf{t} - \mathbf{v}\| \leq \gamma\lambda_1(\mathcal{L})$ for some $\gamma < 1$, the problem is more commonly known as the bounded distance decoding problem (BDD).

In [Kan83], an embedding technique was introduced that can transform BDD in an n -dimensional lattice to uSVP in an $(n+1)$ -dimensional lattice. Essentially, suppose that we have a target vector $\mathbf{t}$ that is close to a lattice generated by a basis $B = \{\mathbf{b}_1, \dots, \mathbf{b}_n\}$. We construct

another lattice $\mathcal{L}' \subset \mathbb{Z}^{n+1}$ generated by the following matrix:

$$B' = \begin{pmatrix} \mathbf{b}_1 & 0 \\ \mathbf{b}_2 & 0 \\ \vdots & \vdots \\ \mathbf{b}_n & 0 \\ \mathbf{t} & u \end{pmatrix}.$$

Here, u is called the embedding factor.

We remark that solving BDD is one of the approaches to solve the learning with errors (LWE) problem. As such, various experiments had been performed on LWE instances via the BDD approach [LN13, KMW16]. A good discussion of the various approaches to solve BDD can be found in [APS15].

2.4 Goppa Codes

In this section, we recall Goppa codes and their permutations. We also review automorphism-induced Goppa codes and the associated algebraic attacks. In this section, we always assume that $q = 2^m$ for an integer $m \geq 2$. Let $\mathbb{F}_q$ denote the finite field with q elements.

We first review the definition of Goppa codes [McE78]. In the following, we present a construction that is relevant for this work.

Definition 2.4.1 (Goppa code). *Let t and n be positive integers with $t < n \leq q$. Let $g(x) \in \mathbb{F}_q[x]$ be a polynomial of degree t and let $L = \{\gamma_1, \dots, \gamma_n\}$ be an ordered set containing n distinct elements of $\mathbb{F}_q$ such that $g(\gamma_i) \neq 0$ for $1 \leq i \leq n$. The Goppa code $\Gamma(L, g)$ is defined as*

$$\Gamma(L, g) = \left\{ \mathbf{c} = (c_1, \dots, c_n) \in \mathbb{F}_2^n : \sum_{i=1}^n \frac{c_i}{x - \gamma_i} \equiv 0 \pmod{g(x)} \right\}.$$

The polynomial $g(x)$ is called the Goppa polynomial. When $g(x)$ is irreducible, $\Gamma(L, g)$ is called

an irreducible Goppa code. In this section, we call the ordered set L the Goppa support.

Some of the main properties of Goppa codes are summarized below. Recall m is the extension degree of $\mathbb{F}_q$ over $\mathbb{F}_2$.

Remark 2.4.2. (i) The Goppa code $\Gamma(L, g)$ is a subfield subcode of generalized Reed-Solomon codes. In fact, the class of Goppa codes is a special case of alternant codes.

(ii) The Goppa code $\Gamma(L, g)$ is a binary linear code and has dimension at least $n - mt$, and minimum distance at least $t + 1$.

(iii) If $g(x)$ has no multiple roots, then $\Gamma(L, g)$ has minimum distance at least $2t + 1$. In particular, an irreducible Goppa code $\Gamma(L, g)$ has minimum distance at least $2t + 1$.

Given a Goppa support L and a Goppa polynomial $g(x)$, a parity-check matrix of $\Gamma(L, g)$ is given by $H = VD$, where

$$V = \begin{pmatrix} 1 & 1 & \dots & 1 \\ \gamma_1 & \gamma_2 & \dots & \gamma_n \\ \gamma_1^2 & \gamma_2^2 & \dots & \gamma_n^2 \\ \dots & \dots & \dots & \dots \\ \gamma_1^{t-1} & \gamma_2^{t-1} & \dots & \gamma_n^{t-1} \end{pmatrix} \quad (2.2)$$

and

$$D = \text{diag}(1/g(\gamma_1), 1/g(\gamma_2), \dots, 1/g(\gamma_n)). \quad (2.3)$$

One of the nice properties of Goppa codes lies in its efficient decodability. In particular, there exists a polynomial-time decoding algorithm for Goppa codes [MS78] that can decode up to t errors.

Fix $a \in \mathbb{F}_q^*, b \in \mathbb{F}_q, c \in \mathbb{F}_q^*$. For a Goppa support $L \subset \mathbb{F}_q$, let $L' = \{a^{-1}x - b : x \in L\}$. Then, for any $g \in \mathbb{F}_q[x]$, it is easy to check that the Goppa codes $\Gamma(L, g)$ and $\Gamma(L', cg(ax + b))$

are equal. Consequently, for any Goppa code, one can find at least $q(q-1)$ different Goppa supports and monic Goppa polynomials that define the given code.

The following result follows directly from the definition of Goppa codes [MS78].

Lemma 2.4.3. *Fix $L \subset \mathbb{F}_q$. For any two polynomials $g_1(x)$ and $g_2(x)$, one has:*

$$\Gamma(L, g_1(x)) \cap \Gamma(L, g_2(x)) = \Gamma(L, \text{lcm}(g_1(x), g_2(x))).$$

2.4.1 Permutations of Goppa Codes

In this subsection, we take a look at permutations of Goppa codes.

For a positive integer n , let $\mathcal{S}_n$ denote the symmetric group on n symbols. We let $\mathcal{S}_n$ act on the vector space $\mathbb{F}_2^n$ via

$$\sigma(\mathbf{c}) = \sigma(c_1, \dots, c_n) = (c_{\sigma(1)}, \dots, c_{\sigma(n)})$$

for $\sigma \in \mathcal{S}_n$ and $\mathbf{c} = (c_1, \dots, c_n) \in \mathbb{F}_2^n$. Then, $\sigma(C) = \{\sigma(\mathbf{c}) : \mathbf{c} \in C\}$ is also a code with the same parameters as the code C .

Definition 2.4.4. *The automorphism group of a code $C \subseteq \mathbb{F}_2^n$ is defined as*

$$\text{Aut}(C) = \{\sigma \in \mathcal{S}_n : \sigma(\mathbf{c}) \in C \text{ for all } \mathbf{c} \in C\}.$$

In particular, $\sigma \in \text{Aut}(C)$ if and only if $\sigma(C) = C$.

We next show that there exist permutations $\sigma \in \mathcal{S}_n$ such that $\sigma(C)$ is itself a Goppa code.

Let $F = \mathbb{F}_q(x)$ denote the rational function field over $\mathbb{F}_q$. We denote by $\text{Aut}(F/\mathbb{F}_q)$ the automorphism group of F over $\mathbb{F}_q$, i.e.,

$$\text{Aut}(F/\mathbb{F}_q) = \{\sigma : F \rightarrow F, \text{ where } \sigma \text{ is an } \mathbb{F}_q\text{-automorphism of } F\}. \quad (2.4)$$

It is clear that an automorphism $\sigma \in \text{Aut}(F/\mathbb{F}_q)$ is uniquely determined by $\sigma(x)$. It is well known that every automorphism $\sigma \in \text{Aut}(F/\mathbb{F}_q)$ is given by

$$\sigma(x) = \frac{ax + b}{cx + d} \quad (2.5)$$

for some constants $a, b, c, d \in \mathbb{F}_q$ with $ad - bc \neq 0$ (see [Che51]). Denote by $\text{GL}_2(q)$ the general linear group of 2×2 invertible matrices over $\mathbb{F}_q$. Thus, every matrix $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \text{GL}_2(q)$ induces an automorphism of F given by (2.5). Two matrices of $\text{GL}_2(q)$ induce the same automorphism of F if and only if they belong to the same coset of $Z(\text{GL}_2(q))$, where $Z(\text{GL}_2(q))$ stands for the center $\{aI_2 : a \in \mathbb{F}_q^*\}$ of $\text{GL}_2(q)$. This implies that $\text{Aut}(F/\mathbb{F}_q)$ is isomorphic to the projective linear group $\text{PGL}_2(q) := \text{GL}_2(q)/Z(\text{GL}_2(q))$. Thus, we can identify $\text{Aut}(F/\mathbb{F}_q)$ with $\text{PGL}_2(q)$.

Consider the subgroup of $\text{PGL}_2(q)$,

$$\text{AGL}_2(q) := \left\{ \begin{pmatrix} a & b \\ 0 & 1 \end{pmatrix} : a \in \mathbb{F}_q^*, b \in \mathbb{F}_q \right\}. \quad (2.6)$$

$\text{AGL}_2(q)$ is called the affine linear group. Every element $A = \begin{pmatrix} a & b \\ 0 & 1 \end{pmatrix} \in \text{AGL}_2(q)$ defines an affine automorphism $\sigma_{a,b}(x) = ax + b$.

The following lemma is straightforward to verify.

Lemma 2.4.5. *Let $b \in \mathbb{F}_q$. Then,*

$$\text{ord}(\sigma_{a,b}) = \begin{cases} \text{ord}(a) & \text{if } a \neq 1, \\ 2 & \text{otherwise} \end{cases}$$

We recall the definition of a group action on a set and some of its basic properties.

Definition 2.4.6 ([LN97]). A group $\mathcal{G}$ action on a set S is a mapping from $\mathcal{G} \times S$ to S satisfying the following properties:

$$(i) \quad g_1 \cdot (g_2 \cdot s) = (g_1 g_2) \cdot s \text{ for all } g_1, g_2 \in \mathcal{G}, s \in S,$$

$$(ii) \quad 1 \cdot s = s \text{ for all } s \in S.$$

Definition 2.4.7. Let $\mathcal{G}$ be a group acting on a set S . The equivalence class $\{g \cdot s : g \in \mathcal{G}\}$ is called the orbit of $\mathcal{G}$ containing s .

Remark 2.4.8. (i) The orbits of two different elements in S are either equal or disjoint. In particular, S is partitioned into a disjoint union of orbits.

(ii) For $s \in S$, let $\mathcal{G}_s$ denote the stabilizer subgroup of $\mathcal{G}$ that fixes s under the group action, that is $\mathcal{G}_s = \{g \in \mathcal{G} : g \cdot s = s\}$. Then, the orbit containing s has $|\mathcal{G}|/|\mathcal{G}_s|$ different elements.

For any element $g \in \mathcal{G}$, the action of g induces a permutation on the set S . In fact, g restricted to any orbit $\mathcal{O}$ is a permutation of $\mathcal{O}$. In particular, let $L = \{\gamma_1, \dots, \gamma_n\}$ be a disjoint union of $\mathcal{G}$ -orbits. Then, for any $\sigma \in \mathcal{G}$ and $1 \leq i \leq n$, there exists j such that $\sigma \cdot \gamma_i = \gamma_j$. In this case, we will simply denote the corresponding permutation in $\mathcal{S}_n$ as $\sigma(i) = j$.

Definition 2.4.9. Let L be a subset of $\mathbb{F}_q$ containing n distinct elements. For any $\sigma \in \mathcal{S}_n$, we say that L is invariant under σ if $\sigma(L) = L$. If L is invariant under every σ in a subgroup $\mathcal{G}$ of $\mathcal{S}_n$, we say that L is $\mathcal{G}$ -invariant. In particular, if L is a disjoint union of $\mathcal{G}$ -orbits, L is $\mathcal{G}$ -invariant.

Note that σ is an element of $\mathcal{S}_n$ and it can be easily extended to an element of $\mathcal{S}_q$.

Remark 2.4.10. Consider a group action of $\text{AGL}_2(q)$ on $\mathbb{F}_q$ defined by $\sigma_{a,b} \cdot \gamma = a^{-1}(\gamma - b)$. One easily checks that this is a group action. Then, one has $\sigma_{a,b}(x - \gamma) = a(x - \sigma_{a,b} \cdot \gamma)$.

Proposition 2.4.11. *Let $\mathcal{G}$ be a subgroup of $\text{AGL}_2(q)$ and let $L \subset \mathbb{F}_q$ be a disjoint union of $\mathcal{G}$ -orbits. Let $g \in \mathbb{F}_q[x]$. For any $\sigma \in G$, one has*

$$\sigma(\Gamma(L, g)) = \Gamma(L, g(\sigma^{-1}(x))).$$

From Def 2.4.9, we know L is $\mathcal{G}$ -invariant. The σ on the left hand side is the induced permutation when σ acts on L .

Proof. Write $L = \{\gamma_1, \dots, \gamma_n\}$. Here, it suffices to show that $\sigma(\Gamma(L, g)) \subset \Gamma(L, g(\sigma^{-1}(x)))$. Let $\mathbf{c} = (c_1, c_2, \dots, c_n) \in \Gamma(L, g)$. By definition, we have:

$$\sum_{i=1}^n \frac{c_i}{x - \gamma_i} \equiv 0 \pmod{g(x)}.$$

Applying σ^{-1} to both sides of the equivalence relation Remark 2.4.10 yields

$$\sum_{i=1}^n \frac{c_i}{x - \sigma^{-1} \cdot \gamma_i} \equiv 0 \pmod{g(\sigma^{-1}(x))}.$$

Since L is a union of $\mathcal{G}$ -orbits, for each $i = 1, \dots, n$, there exists some j such that $\gamma_j \in L$ and $\sigma(\gamma_j) = \gamma_i$. Rearranging the equation then gives:

$$\sum_{i=1}^n \frac{c_{\sigma(i)}}{x - \gamma_i} \equiv 0 \pmod{g(\sigma^{-1}(x))},$$

and this proves our desired result. □

2.4.2 Automorphism-induced Goppa Codes

In Proposition 2.4.11, suppose further that there exists some $\alpha \in \mathbb{F}_q^*$ such that $g(\sigma^{-1}) = \alpha g$. Then, it follows that $\sigma(\Gamma(L, g)) = \Gamma(L, g)$, that is, σ is in the automorphism group of $\Gamma(L, g)$. In other words, for any $\mathbf{c} \in \Gamma(L, g)$, the permuted codeword $\sigma^i(\mathbf{c})$ lies in the code as well for

$i = 1, 2, \dots$. This property enables one to construct generator matrices of the code exhibiting a nice structure, thereby reducing the public key size of schemes built from these codes.

As such, a common approach to construct McEliece-based schemes with shorter keys is to employ Goppa codes induced by subgroups of $\text{AGL}_2(q)$ (or more generally, subgroups of projective semi-linear groups of $\text{Aut}(F/\mathbb{F}_q)$). For instance, in [Ber00], the author described alternant codes that can be constructed from prescribed automorphism subgroups of which, Goppa codes form a special case. These codes resulted in quasi-cyclic, quasi-dyadic and monoidic alternant codes which were subsequently employed to construct McEliece-based schemes with compact keys [MB09, BCGO09, BBB⁺18b, BBB⁺18a].

We review the main ideas of this approach below. Here, we are presenting the ideas in a more general manner to facilitate the discussion in the remainder of the section.

Let $\mathcal{G}$ be a subgroup of $\text{AGL}_2(q)$ of size r . Suppose that there are at least s $\mathcal{G}$ -orbits of size r . Let L be a union of s of these orbits and let $n = rs$. Let $t = dr$ for some positive integer d and let $g(x)$ be a polynomial of degree t that is invariant under the $\mathcal{G}$ -action, i.e., for every $\sigma \in \mathcal{G}$, there exists $\alpha \in \mathbb{F}_q^*$ such that $g(\sigma(x)) = \alpha g(x)$. From Proposition 2.4.11, it follows that for every $\sigma \in \mathcal{G}$, $\sigma(\Gamma(L, g)) = \Gamma(L, g)$. Hence, $\Gamma(L, g)$ is a Goppa code with $\mathcal{G}$ as a subgroup of $\text{Aut}(\Gamma(L, g))$.

One common way to construct the $\mathcal{G}$ -invariant polynomial $g(x)$ is as follows. Pick an irreducible polynomial $f(x)$ of degree d . Define

$$g(x) = f\left(\prod_{\sigma \in \mathcal{G}} \sigma(x)\right).$$

Then, $g(x)$ is invariant under any $\sigma \in \mathcal{G}$. In [Ber00, FOP⁺16a], the authors classified polynomials invariant under some particular subgroups of $\text{AGL}_2(q)$.

Denote the orbits in L by $\mathcal{O}_i$, $i = 1, 2, \dots, s$. For each orbit $\mathcal{O}_i$, fix a representative β_i . Fix an order in $\mathcal{G}$, that is $\mathcal{G} = \{\sigma_1, \sigma_2, \dots, \sigma_r\}$. With this order, write $\mathcal{O}_i = \{\sigma_1 \cdot \beta_i, \sigma_2 \cdot \beta_i, \dots, \sigma_r \cdot \beta_i\}$. It is clear that for any $\sigma \in \mathcal{G}$, σ induces the same permutation for each orbit. We denote this

permutation by $\tilde{\sigma}$.

Order the elements in L as

$$L = \mathcal{O}_1 || \mathcal{O}_2 || \dots || \mathcal{O}_s = \{(\sigma_1 \cdot \beta_1 \dots \sigma_r \cdot \beta_1), \dots, (\sigma_1 \cdot \beta_s \dots \sigma_r \cdot \beta_s)\}.$$

Let $\mathbf{c} = (c_1, \dots, c_n) \in \Gamma(L, g)$. We partition $\mathbf{c}$ as $\mathbf{c} = (\mathbf{c}_{\mathcal{O}_1} || \dots || \mathbf{c}_{\mathcal{O}_s})$, where each $\mathbf{c}_{\mathcal{O}_i}$ corresponds to the entries indexed by elements in $\mathcal{O}_i$ with $i \in \{1, \dots, s\}$. Then for each $\sigma \in \mathcal{G}$,

$$\mathbf{c}_\sigma = (\mathbf{c}_{\tilde{\sigma}(\mathcal{O}_1)} || \dots || \mathbf{c}_{\tilde{\sigma}(\mathcal{O}_s)}) \in \Gamma(L, g).$$

Next, we seek to construct a generator matrix for $\Gamma(L, g)$ having a nice form. Let k denote the dimension of $\Gamma(L, g)$. Suppose further that $k = n - mt$. Let $k_0 = k/r = (s - md)$.

Consider a generator matrix M of $\Gamma(L, g)$. Without any loss of generality, suppose that M can be put in systematic form [BS08] (for otherwise, choose other β_i 's), that is, $M = (I, B)$, where B is a $k \times (n - k)$ matrix. Label the rows of M by M_i . Define $P(L, g) := \{M_1, M_{r+1}, \dots, M_{(k_0-1)r+1}\}$. For each $\mathbf{c} \in P(L, g)$, form the matrix

$$\mathbf{c}(\mathcal{G}) = \begin{pmatrix} \sigma_1(\mathbf{c}) \\ \vdots \\ \sigma_r(\mathbf{c}) \end{pmatrix}.$$

Finally, form the generator matrix

$$M(L, g) = \begin{pmatrix} \mathbf{c}_1(\mathcal{G}) \\ \vdots \\ \mathbf{c}_{k_0}(\mathcal{G}) \end{pmatrix},$$

for $\mathbf{c}_i = M_{(i-1)r+1}$. Note that since M is in systematic form, each $\mathbf{c}_i = M_{(i-1)r+1}$ is such that the i -th block is of the form $(1, 0, \dots, 0)$ while the other blocks in the first k_0 blocks are

$(0, 0, \dots, 0)$.

Theorem 2.4.12. *The matrix $M(L, g)$ constructed above is a generator matrix of $\Gamma(L, g)$.*

Proof. First, it is clear that for each $\mathbf{c} \in P(L, g)$ and for each $\sigma \in \mathcal{G}$, $\sigma(\mathbf{c}) \in \Gamma(L, g)$. It thus remains to show that $M(L, g)$ has rank k . Observe that for each $\mathbf{c} \in P(L, g)$, $\mathbf{c}(\mathcal{G})$ is an $r \times n$ matrix. Further, $\mathbf{c}(\mathcal{G})$ can be viewed as a concatenation of s $r \times r$ square matrices where each square matrix is in fact $\mathbf{c}_{\mathcal{O}}(\mathcal{G})$ for an orbit $\mathcal{O}$ contained in L . Consider the first k_0 blocks of $\mathbf{c}_i(\mathcal{G})$. By our choice of $\mathbf{c}_i$, we see that the i -th block of $\mathbf{c}_i(\mathcal{G})$ is a permutation matrix while all other blocks are 0. Hence, the first k columns of $M(L, g)$ have the form

$$\begin{pmatrix} A & 0 & \dots & 0 \\ 0 & A & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & A \end{pmatrix},$$

where $A = \mathbf{x}(\mathcal{G})$ and $\mathbf{x} = (1, 0, \dots, 0)$. Since the rank of A is r , the desired result follows. $\square$

Two particular subgroups of $\text{AGL}_2(q)$ of interest are the multiplicative cyclic subgroup of order $r|(q-1)$ and the additive subgroup of order 2^v for $1 \leq v \leq m$. Concretely, the former comprises automorphisms of the form $\sigma_a(x) = ax$ for $a \in \mathbb{F}_q^*$ of order r and gives rise to quasi-cyclic Goppa codes [BCGO09, BBB⁺18b]. In this case, the orbits in the Goppa support L are cosets of the subgroup of $\mathbb{F}_q^*$ of order r and the Goppa polynomial is of the form $g(x) = f(x^r)$ for some irreducible polynomial $f(x) \in \mathbb{F}_q[x]$.

On the other hand, the additive subgroup construction results in quasi-dyadic Goppa codes [MB09, BBB⁺18a]. Let V be any $\mathbb{F}_2$ -subspace of $\mathbb{F}_q$ with dimension v . The additive automorphism group consists of automorphisms of the form $\sigma_b(x) = x + b$ for all $b \in V$. Thus, the orbits in L are all cosets of V in $\mathbb{F}_q$, each of size 2^v and the Goppa polynomial is of the form $f(L(x))$, where $f(x)$ is an irreducible polynomial and $L(x) = \prod_{b \in V} (x + b)$.

Observe from Theorem 2.4.12 that it suffices to provide the set $P(L, g)$ in order to construct the generator matrix of $\Gamma(L, g)$, that is, these codes are examples of the codes we seek in Problem 1. In other words, when using this code in McEliece-based schemes, the public key size can be reduced. However, matrices with such nice structure also reduce the number of unknowns to mount algebraic attacks, thereby weakening the security with respect to structural attacks. One way to perform algebraic cryptanalysis is to consider the parity-check equations using the parity-check matrix $H = VD$ where V and D are defined in Equations 2.2 and 2.3, respectively, and treating the γ_i 's and $1/g(\gamma_i)$'s as unknowns. One then tries to solve the system of equations using algebraic solving tools such as Gröbner basis algorithms. In the case of automorphism-induced Goppa codes, the unknowns for the Goppa support are reduced to the elements of the subgroup as well as the orbit representatives. At the same time, since $g(a)$ is identical for all a in the same orbit of L , the number of unknowns of the diagonal matrix in Equation 2.3 is reduced to the number of orbits. Essentially, it was shown in [FOP⁺16a, Bar18] that one can construct Goppa codes with much smaller parameters from these automorphism-induced codes, namely, via folded codes and invariant codes defined as follows.

Definition 2.4.13. *Let $C = \Gamma(L, g)$ be a Goppa code induced by an automorphism subgroup $\mathcal{G}$ so that L is a disjoint union of $\mathcal{G}$ -orbits. For $\mathbf{c} \in C$, define $\text{Punc}(\mathbf{c})$ as the codeword obtained from $\mathbf{c}$ by puncturing on a set of representatives for the orbits in L .*

- *The folded code $\phi(C)$ is the subcode of C defined as:*

$$\phi(C) = \{\text{Punc}(\sum_{\sigma \in \mathcal{G}} \sigma(\mathbf{c})) : \mathbf{c} \in C\}.$$

- *The invariant code $C^{\mathcal{G}}$ is the subcode of C defined by:*

$$C^{\mathcal{G}} = \{\text{Punc}(\mathbf{c}) : \mathbf{c} \in C \mid \sigma(\mathbf{c}) = \mathbf{c} \text{ for all } \sigma \in \mathcal{G}\}.$$

Clearly, the folded code is a subcode of the invariant code. In [FOP⁺16a, Bar18], it was shown that for the quasi-cyclic case, these two subcodes are identical and equal to $\Gamma(L', f(x))$, where L' is a set of representatives of orbits in L and $g(x) = f(x^r)$ with r being the order of the cyclic subgroup of $\mathbb{F}_q^*$.

2.5 The Subset Sum Problem

Definition 2.5.1 (Subset Sum Problem). *Let n, m, q be integers. Given a target number $t \in \mathbb{Z}_{q^m}$ and a vector $\mathbf{a} = (a_1 \dots a_n) \in \mathbb{Z}_{q^m}^n$, the subset sum problem is to seek out a subset $S \subseteq \{1 \dots n\}$ satisfying $\sum_{i \in S} a_i = t \pmod{q^m}$.*

The subset sum problem can also be regarded as the problem of finding the preimage of the function $f_{\mathbf{a}} : \{0, 1\}^n \rightarrow \mathbb{Z}_{q^m}$ defined by $f_{\mathbf{a}}(\mathbf{s}) = f_{\mathbf{a}}(s_1 \dots s_n) = \sum_{i=1}^n a_i s_i \pmod{q^m}$.

We denote the subset sum problem as $\text{SS}(n, q^m)$ given parameters n and q^m .

Lyubashevsky, Palacio and Segev [LPS10] observe that a subset sum problem $\text{SS}(n, q^m)$ can be converted to a *matrix-vector* representation. Given an instance $(\mathbf{a}, t) \in \mathbb{Z}_{q^m}^n \times \mathbb{Z}_{q^m}$ of $\text{SS}(n, q^m)$, each entry of $\mathbf{a}$ can be expressed as a vector over $\mathbb{Z}_q^m$ via the q -ary representation. Thus, the vector $\mathbf{a}$ is transformed to a matrix $A \in \mathbb{Z}_q^{m \times n}$ and the target t is converted to a vector $\mathbf{t} \in \mathbb{Z}_q^m$. For convenience, from now onwards, the index of matrix or vector starts from 0 while iterating the entries in the subset sum related equations. Let $A = (A_{ij})_{i \in \{0 \dots m-1\}, j \in \{0 \dots n-1\}}$ and $\mathbf{t} = (t_0, \dots, t_{m-1})$. Then $t = \sum_{i=0}^{m-1} t_i q^i$ and $a_j = \sum_{i=0}^{m-1} A_{ij} q^i$. The instance equation $t = \sum_{i \in \{0 \dots n-1\}} a_i s_i \pmod{q^m}$ is converted to

$$\sum_{i=0}^{m-1} t_i q^i = \sum_{j=0}^{n-1} s_j \sum_{i=0}^{m-1} A_{ij} q^i \pmod{q^m}. \quad (2.7)$$

Given a vector $\mathbf{s} \in \{0, 1\}^n$ and a matrix $A \in \mathbb{Z}_q^{m \times n}$, we denote $\mathbf{t} := A \odot \mathbf{s}$ as vector defined in Equation 2.7. Here $\odot$ denotes a matrix-vector representation of the subset sum problem.

We recall a hardness result of the subset sum problem from [IN96], which is useful in

establishing security of the public key cryptosystem. The result says that the subset sum problem with large modulus is not harder than subset sum problem with smaller modulus.

Lemma 2.5.2. *Let i, j be integers satisfying $i < j$ and $2^n < q^{m+i}$. If $\text{SS}(n, q^{m+j})$ is hard, $\text{SS}(n, q^{m+i})$ is hard. For the case $i > j$ and $2^n > q^{m+i}$, if $\text{SS}(n, q^{m+j})$ is hard, $\text{SS}(n, q^{m+i})$ is hard.*

2.6 Security Definitions

Definition 2.6.1 (Public-key encryption scheme). *A public-key cryptosystem is defined by three efficient PPT algorithms (KeyGen, Enc, Dec)*

- KeyGen, given the security parameter, generates the key pair (sk, pk) .
- Enc, given a message m and the public key pk , produces a ciphertext $C = \text{Enc}(\text{pk}, m)$.
- Dec, given a ciphertext C and the secret key sk , produces a message m' such that

$$\Pr_{(\text{pk}, \text{sk}) \leftarrow \text{KeyGen}}[\text{Dec}(\text{sk}, \text{Enc}(\text{pk}, m)) = m] = 1.$$

A standard notion of security, called *semantic security* or IND-CPA informally ensures that the encryption discloses nothing about the encrypted message to a passive attacker. Formally the definition considers the following experiment. The experiment is parametrized by a bit b .

1. Given the public key encryption scheme, the attacker chooses two messages m_0, m_1 .
2. The attacker receives $\text{Enc}(\text{pk}, m_b)$ for a bit b .
3. The attacker outputs a bit μ and accepts if $\mu = b$.

Definition 2.6.2 (Semantic security [GM84]). *A cryptosystem is semantically secure if the accepting probabilities in both of the two cases $b = 0$ or $b = 1$ differ by a negligible function, namely, the distinguishing advantage between the two cases is negligible.*

Definition 2.6.3 (Adaptive Memory Attacks [AGV09]). *Let $\alpha : \mathbb{Z} \rightarrow \mathbb{Z}$ be a function, and let N be the size of the secret key output by $\text{KeyGen}(1^\lambda)$. Let H_{sk} be an oracle that given a polynomial-size circuit h , produces $h(\text{sk})$. A PPT adversary $\mathcal{A} = (\mathcal{A}_1^{H_{\text{sk}}}, \mathcal{A}_2)$ is called admissible if the total number of bits that $\mathcal{A}$ gets as a result of oracle queries to H_{sk} is at most $\alpha(N)$.*

A public key cryptosystem $\text{PKE} = (\text{KeyGen}, \text{Enc}, \text{Dec})$ achieves semantic security against adaptive $\alpha(N)$ -memory attacks if for any admissible PPT adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, the probability that $\mathcal{A}$ wins in the following experiment differs from $\frac{1}{2}$ by a negligible function in λ .

1. $(\text{pk}, \text{sk}) \leftarrow \text{KeyGen}(1^\lambda)$.
2. $(\mu_0, \mu_1, \text{state}) \leftarrow \mathcal{A}_1^{H_{\text{sk}}}(\text{pk})$ such that $|\mu_0| = |\mu_1|$
3. $c \leftarrow \text{Enc}_{\text{pk}}(\mu_b)$ where $b \in \{0, 1\}$ is a random bit
4. $b' \leftarrow \mathcal{A}_2$ wins the experiment if $b' = b$.

2.7 Cryptosystems

2.7.1 The McEliece Cryptosystem

The McEliece encryption scheme was proposed in [McE78] as a public-key cryptosystem that is based on hard problems in algebraic coding theory instead of the usual integer factoring problem or discrete logarithm problems in groups. More specifically, its construction hinges on the difficulty to decode general linear codes over finite fields. Unlike the widely-used integer factoring problem and discrete logarithm problem which had been proven to be vulnerable to polynomial-time quantum algorithms [Sho94], the decoding problem is touted as one of the potential candidates to be used as a basis for post-quantum cryptography.

Essentially, the McEliece encryption scheme generates two different bases of a linear code, one with an easy decoding strategy while the other is presumably difficult to decode. Concretely, let G be an $[n, k, 2t + 1]$ binary Goppa code which admits an efficient decoding algorithm. Let

U be a $k \times k$ invertible binary matrix and P an $n \times n$ permutation matrix. Let $G' = UGP$. Then G' represents a general linear code with no obvious way to decode. The basic structure of the McEliece encryption scheme can be described as follows:

Public key: The matrix G' and the parameters n, k, t .

Private key: The matrices G, U and P .

Encryption: Let $\mathbf{m}$ be a k -bit message. Randomly pick an n -bit error vector $\mathbf{e}$ with Hamming weight t . The encryption of $\mathbf{m}$ is given by

$$\mathbf{c} = E(\mathbf{m}) = \mathbf{m}G' + \mathbf{e}.$$

Decryption: Let $\mathbf{c}' = \mathbf{c}P^{-1}$. With the secret key G , decode $\mathbf{c}'$ to obtain the message $\mathbf{m}'$. Compute $\mathbf{m} = \mathbf{m}'U^{-1}$.

2.7.2 The GGH Cryptosystem

The GGH cryptosystem was presented in [GGH97] as a lattice analog of the McEliece cryptosystem. While the McEliece scheme exploits the difficulty to decode the received word obtained from a random code whenever errors of small weights are introduced, the GGH scheme relies on a similar phenomenon on general lattices. Indeed, the GGH scheme constructs two different bases of the same lattice, one of which allows CVP to be solved efficiently via Babai's nearest plane algorithm while the other basis is constructed as a random basis of the lattice and hence, has very poor performance with respect to CVP. More precisely, one first constructs a basis B with short highly orthogonal rows (that is a basis with small orthogonality defect) and then multiply B by random unimodular matrices to obtain a basis B' of the same lattice with much higher orthogonality defect. B is then used as the private key while B' will serve as the public key. The basic structure of the GGH encryption scheme is presented next.

Private key: The basis B and a unimodular matrix U ;

Public key: The basis $B' = UB$, and the parameter n and a small positive integer σ ;

Encryption: Let the message $\mathbf{m} \in \mathbb{Z}^n$. Choose an error $\mathbf{e} \in \mathbb{Z}^n$ whose entries are randomly picked to be $\pm\sigma$. The encryption of $\mathbf{m}$ is given by

$$\mathbf{c} = E(\mathbf{m}) = \mathbf{m}B' + \mathbf{e}.$$

Decryption: Using Babai's nearest plane algorithm and the basis B , determine the vector $\mathbf{v}$ closest to $\mathbf{c}$. We have $\mathbf{m} = \mathbf{v}U^{-1}$.

Remark 2.7.1. *We have only outlined the basic McEliece and GGH constructions. These constructions need to be further enhanced to make them semantically secure [GGH97, KI01, HHK17].*

3. GOPPA CODES WITH REDUCED KEY SIZE

3.1 Introduction

In this chapter, we propose a new general construction to reduce the public key size of the classical McEliece cryptosystem based on Goppa codes. Our construction generalize the idea of automorphism-induced Goppa code. In our construction, we consider using nontrivial proper subsets of automorphism subgroups to construct Goppa codes. Choosing proper subsets of automorphism groups makes our construction resistant to existing structural algebraic attacks. In fact, our construction is a general construction and works for any generic automorphism subgroups. In particular, we consider two instantiations including the partial quasi-cyclic construction and the partial quasi-dyadic construction to show the power of our technique. For a classical McEliece cryptosystem with given parameters, we provide a modular method to reduce the public key size. To argue that our construction is not only a theoretical result, but also practical, we apply our construction to the existing scheme to reduce the public key size by around half.

Instead of the typical method of finding codes with nontrivial automorphism groups, we will construct Goppa codes that are a union of different subcodes and their permutations. Specifically, we solve the following problem.

Problem 3.1.1. *Construct Goppa codes C that contain a subset S of linearly independent codewords satisfying the following properties:*

- *For each $\mathbf{c} \in S$, there exists a set of permutations $\mathcal{P}_{\mathbf{c}}$ such that $\sigma(\mathbf{c}) \in C$ for all $\sigma \in \mathcal{P}_{\mathbf{c}}$;*

- $B = \bigcup_{\mathbf{c} \in S} \left(\bigcup_{\sigma \in \mathcal{P}_{\mathbf{c}}} \sigma(\mathbf{c}) \right)$ is a basis of C .

By finding such codes C , one can then use the set S as its public key. Observe that the automorphism-induced codes including the quasi-cyclic and quasi-dyadic codes [MB09, BCGO09] are examples of the codes we seek. In these cases, all the $\mathcal{P}_{\mathbf{c}}$ are identical and equal to the automorphism group of the code. While this construction allows one to use the set S as the public key, the automorphism group can be exploited to reduce the security of the scheme with respect to algebraic attacks.

This chapter seeks to provide other classes of Goppa codes that satisfy the conditions of Problem 3.1.1. In our constructions, the sets $\mathcal{P}_{\mathbf{c}}$ are no longer automorphism groups of the codes. Instead, we consider only subsets of permutations of each codeword. In this way, our codes become resistant to the algebraic attacks of [FOP⁺16b, FOP⁺16a, Bar18] as the folding operation does not lead to an invariant subcode (see Section 2.4.2 for the relevant definitions). Moreover, our construction is generic in the sense that it can be applied to reduce the key size of existing code-based schemes from Goppa codes.

We first prove conditions for permutations of codewords to lie in the code. We then construct codes with a partial quasi-cyclic and quasi-dyadic structure. We demonstrate that the codes we construct are not vulnerable to the algebraic attacks of [FOPT10, FPdP14, FOP⁺16a]. Finally, we apply our technique to automorphism-induced Goppa codes to obtain compact and secure Goppa codes. We show with concrete examples that our technique can reduce the public key sizes in BigQuake [BBB⁺18b] by at least half.

In this chapter, we always assume that $q = 2^m$ for an integer $m \geq 2$. Let $\mathbb{F}_q$ denote the finite field with q elements.

3.1.1 Organization of the Chapter

The subsequent sections are structured as follows. We first present our new construction that exploits subsets of automorphism groups to yield codes with a nice permutation structure.

In Section 3.3, we provide a security discussion of our construction. Finally, we present an algorithm that combines our technique with existing quasi-cyclic constructions to obtain Goppa codes with reduced public key sizes.

3.2 Partial Quasi-cyclic and Partial Quasi-dyadic Goppa Codes

In this section, we present a more general construction for codes satisfying the properties in Problem 1, namely, we do not restrict to nontrivial automorphism subgroups of the code. In other words, our codes may have trivial automorphism groups but contain subsets of codewords and sets of permutations where these codewords and their corresponding permutations generate the whole code. These constructions will prevent an attacker from finding folded and invariant subcodes of the code.

First, we give a sufficient condition for a permuted codeword of a code to remain in the code.

Lemma 3.2.1. *Given a linear code C and a permutation $\sigma \in \mathcal{S}_n$, $C' = C \cap \sigma(C)$ is also a linear code. Suppose that C' is nontrivial. For any codeword $\mathbf{c} \in C'$, one has $\sigma^{-1}(\mathbf{c}) \in C$. Further, if $\sigma^{-1}(\mathbf{c}) \in \sigma(C)$, then $\sigma^i(\mathbf{c}) \in C'$ for $i \in \mathbb{Z}$.*

Proof. Since $\mathbf{c} \in C' \subseteq \sigma(C)$, it follows that $\sigma^{-1}(\mathbf{c}) \in C$. In the case where $\sigma^{-1}(\mathbf{c}) \in \sigma(C)$, then $\sigma^{-1}(\mathbf{c}) \in C'$. Based on the fact that $\mathbf{c}, \sigma^{-1}(\mathbf{c}) \in C'$ and σ is a permutation, one can continue to apply σ^{-1} many times and the result is still a codeword of C' . $\square$

Remark 3.2.2. *Suppose that σ has order 2. Then $\sigma(\mathbf{c}) \in \sigma(C) \cap \sigma(\sigma(C)) = \sigma(C) \cap C$.*

More generally, let σ_0 be the identity permutation and let $\sigma_1, \dots, \sigma_l$ be l distinct permutations in $\mathcal{S}_n$. Suppose that for a linear code C , the code $C' = \bigcap_{i=0}^l \sigma_i(C)$ is nontrivial. Then for $\mathbf{c} \in C'$ and for all $i = 1, 2, \dots, l$, $\sigma_i^{-1}(\mathbf{c}) \in C$.

In terms of Goppa codes, one has the following result.

Lemma 3.2.3. *Let L be a subset of n distinct elements from $\mathbb{F}_q$. Let $\sigma_1, \dots, \sigma_l$ be l distinct automorphisms from $\text{AGL}_2(q)$ such that L is invariant under each σ_i , $i = 1, 2, \dots, l$. Let $g(x) \in \mathbb{F}_q[x]$ be such that $g(x), g(\sigma_1(x)), \dots, g(\sigma_l(x))$ are pairwise co-prime. Define $G(x) = \prod_{i=0}^l g(\sigma_i(x))$. If $\Gamma(L, G(x))$ is nontrivial, then for every codeword $\mathbf{c} \in \Gamma(L, G(x))$, the permuted codeword $\sigma_i(\mathbf{c})$ lies in $\Gamma(L, g(x))$.*

Moreover, if $\sigma_0 = \text{id}, \sigma_1, \dots, \sigma_l$ form a subgroup of $\text{AGL}_2(q)$, then each $\sigma_i(\mathbf{c})$ lies in $\Gamma(L, G(x))$.

Proof. From Lemma 2.4.3 and Proposition 2.4.11, one has

$$\begin{aligned} \Gamma(L, G(x)) &= \bigcap_{i=0}^l \Gamma(L, g(\sigma_i(x))) \\ &= \bigcap_{i=0}^l \sigma_i^{-1}(\Gamma(L, g)). \end{aligned}$$

Hence, it follows from the above argument that for $i \in \{1 \dots l\}$ and $\mathbf{c} \in \Gamma(L, G(x))$ lies in the code $\Gamma(L, g)$ as well. For the latter part, if $\mathcal{G} = \{\sigma_0, \sigma_1, \dots, \sigma_l\}$ is a group, then $\Gamma(L, g(x)g(\sigma_1(x)) \dots g(\sigma_l(x)))$ is a code with $\mathcal{G}$ as its automorphism subgroup. $\square$

There are different ways one can apply Lemma 3.2.3 to construct Goppa codes with more permuted codewords. Here, we provide two different constructions.

In the following, we assume that $n = lmt$ for some positive integer l .

Proposition 3.2.4 (Partial quasi-dyadic construction). *Let $l = 2^v$, i.e., $n = 2^v mt$ for some positive integer $v \leq m/2$. Let V be a v -dimensional subspace of $\mathbb{F}_q$. Write V as $V = \{b_0, b_1, b_2, \dots, b_{2^v-1}\}$ with $b_0 = 0$. Let L contain mt different cosets of V under some ordering. Let $g(x)$ be an irreducible polynomial of degree t such that the Goppa code $\Gamma(L, g)$ has dimension $k = n - mt = (2^v - 1)mt$. For $i = 0, 1, \dots, 2^v - 1$, define σ_i as the automorphism where $\sigma_i(x) = x + b_i$. Define $h(x) = \prod_{i=0}^{2^v-1} g(\sigma_i(x))$. Suppose that the code $\Gamma(L, h)$ is the trivial code. Then, one can construct a subset $P(L, g)$ of $\Gamma(L, g)$ satisfying the conditions of Problem 3.1.1.*

Proof. Fix a σ_i with $i \neq 0$ and let $h_i(x) = \frac{h(x)}{g(\sigma_i(x))}$. Let $C = \Gamma(L, h_i(x))$. Clearly, C is a subcode of $\Gamma(L, g(x))$. For each codeword $\mathbf{c} \in C$, one has $\sigma_j(\mathbf{c}) \in \Gamma(L, g(x))$ for $j \neq i$. In particular, $\sigma_j(C)$ is a subcode of $\Gamma(L, g(x))$ for all $j \neq i$ (since the order of σ_j is 2). We claim that for j_1, j_2 such that i, j_1, j_2 are all distinct, one has $\sigma_{j_1}(C) \cap \sigma_{j_2}(C)$ is the trivial code. Indeed,

$$\sigma_j(\Gamma(L, h_i)) = \Gamma(L, \prod_{e \neq i} g(\sigma_e \sigma_j(x))).$$

Thus, by the assumption that the $g(\sigma_i(x))$'s are coprime and Lemma 2.4.3,

$$\sigma_{j_1}(\Gamma(L, h_i(x))) \cap \sigma_{j_2}(\Gamma(L, h_i(x))) = \Gamma(L, h(x))$$

which is trivial by our assumption. In addition, since $\dim(\sigma_j(C)) \geq n - (2^v - 1)mt = mt$, it follows that $\dim(\sigma_j(C)) = mt$ for all $j \neq i$. Let $P(L, g)$ be a basis of C . Then, $P(L, g)$ satisfies the conditions of Problem 1 by letting the set of permutations to be $\{\sigma_j : j \neq i\}$. $\square$

Proposition 3.2.5 (Partial quasi-cyclic construction). *Let r be a positive integer with $r|(q-1)$. Let $n = rmt$. Let H be a subgroup of $\mathbb{F}_q^*$ of order r . Let L be a disjoint union of mt cosets of H in some order. Fix an irreducible polynomial $g(x)$ of degree t such that the Goppa code $\Gamma(L, g)$ has dimension $(r-1)mt$. For each $a \in H$, let σ_a be the automorphism such that $\sigma_a(x) = ax$. Define $h(x) = \prod_{a \in H} g(ax)$. Suppose that the code $\Gamma(L, h)$ is trivial. Then, one can find a set $P(L, g)$ of mt linearly independent vectors that satisfies the conditions of Problem 1.*

Proof. Fix a generator a of H . Define $g_a(x) = \prod_{i=0}^{r-2} g(a^i x)$. Let C be the code $C = \Gamma(L, g_a)$. By Proposition 3.2.3, for all $i = 1, \dots, r-2$, we have $\sigma_{a^i}(\mathbf{c}) \in \Gamma(L, g)$ for all $\mathbf{c} \in C$, that is, $\sigma_{a^i}(C) \subset \Gamma(L, g)$ for $i = 2, 3, \dots, r$. We claim that $\Gamma(L, g)$ is a disjoint union of $C, \sigma_{a^2}(C), \dots, \sigma_{a^{r-1}}(C)$. First, we show that for $i \neq j$, $\sigma_{a^i}(C) \cap \sigma_{a^j}(C)$ is trivial. Now, we have

$$\sigma_{a^i}(C) = \Gamma(L, h_a(\sigma_{a^{-i}}(x))) = \Gamma(L, \prod_{w \neq r-1-i} g(\sigma_{a^w}(x))).$$

Similarly, $\sigma_{aj}(C) = \Gamma(L, \prod_{w \neq r-1-j} g(\sigma_{aw}(x)))$. From Lemma 2.4.3, $\sigma_{ai}(C) \cap \sigma_{aj}(C) = \Gamma(L, h)$ which is trivial. Since $\dim(C) \geq n - (r-1)mt = mt$, the claim follows. Consequently, one may choose $P(L, g)$ as a basis of C . $\square$

Remark 3.2.6. • In both Propositions 3.2.4 and 3.2.5, we have $\dim(\Gamma(L, h)) \geq n - \deg(h)m = 0$. In practice, this code is trivial for most cases.

- In fact, one can generalize the results to non-irreducible polynomials g . In this case, one needs to check that the polynomials $g(\sigma(x))$'s are all pairwise coprime for all the automorphisms σ involved.

Remark 3.2.7. Just as in Theorem 2.4.12, one may transform $P(L, g)$ such that $P(L, g)$ takes the following form. Let

$$P(L, g) = (S_1 || S_2 || \dots || S_{mt-1} || S_{mt}),$$

where each S_i is an $mt \times r$ matrix and for $i = 1, 2, \dots, mt-1$, S_i is 0 everywhere except at the $(i, 1)$ -position. Thus, one may use S_{mt} as the public key.

From Propositions 3.2.4 and 3.2.5, we see that one can generalize the construction to any subgroup of $\text{AGL}_2(q)$. More precisely, we give the general construction in the next theorem and the proof is similar to the proofs of Propositions 3.2.4 and 3.2.5.

Theorem 3.2.8. Let t be a positive integer. Let $\mathcal{G}$ be a subgroup of $\text{AGL}_2(q)$ and let L be a disjoint union of mt $\mathcal{G}$ -orbits under the action of $\mathcal{G}$ on $\mathbb{F}_q$. Let $n = |L| = |\mathcal{G}|mt$. Let $g(x)$ be an irreducible polynomial of degree t such that the code $\Gamma(L, g(x))$ has dimension $n - mt$. Define $h(x) = \prod_{\sigma \in \mathcal{G}} g(\sigma(x))$. Assume that $\Gamma(L, h)$ is trivial. Fix a $\sigma \in \mathcal{G}$ where σ is not the identity automorphism and let $\mathcal{G}' = \mathcal{G} \setminus \{\sigma\}$. Construct $C = \Gamma(L, \prod_{\sigma' \in \mathcal{G}'} g(\sigma'(x)))$. Then, we may take a basis of C as the set $P(L, g)$ in Problem 1 with the set of permutations to be the permutations in $\mathcal{G}'$.

3.3 Security Discussion

In general, there are two classes of attacks on McEliece scheme, namely, information set decoding (ISD) attacks and structural attacks. The former attacks were proposed in [Pra62] as a generic decoding attack. Essentially, such attacks seek to perform the decoding directly from the public matrix given. In its most basic form, an adversary tries to guess k positions such that the submatrix restricted to these positions is invertible and the error vector, restricted to these positions, of very small Hamming weight. Such properties will enable the adversary to brute force the error entries at these positions and subsequently, discover the message. Other improvements and variants were later proposed to improve the attack [LB88, Ste88, Dum91, CC98, FS09, BLP11, MMT11, BJMM12, MO15]. So far, the best attack has complexity $2^{0.0967n}$ [MO15], where n is the code length.

On the other hand, the structural attacks attempt to take advantage of the structure of the specific code used. For random Goppa codes, one can perform a brute force search on the Goppa polynomial and/or the elements in the Goppa support. In [Sen00], an algorithm known as the support splitting algorithm was proposed which can correct permutation of the support elements once the Goppa polynomial and the set of support elements are known.

In addition, one can launch an algebraic attack on a Goppa code C with Goppa polynomial g as follows. Consider the equations:

$$\sum_{i=1}^n c_i x_i^j y_i^w = 0 \quad (3.1)$$

for $\mathbf{c} = (c_1, \dots, c_n) \in C$, $0 \leq j \leq t-1$ when $w = 1$ and $0 \leq j \leq 2t-1$ when $w = 2$, $x_1, x_2, \dots, x_n$ are the unknowns for the Goppa support and $y_i = 1/g(x_i)$. Recall that these equations come from the parity-check equations (refer to Equations 2.2 and 2.3). Hence, solving for the unknowns will give the Goppa support and together with the knowledge of the y_i 's will enable one to recover $g(x)$. Thus the challenge remains to solve the above system.

Observe that when $j = 0$ yields k linear equations in the unknowns y_i 's. Thus, one way to solve the system is to guess the remaining mt y_i 's which results in a system in the x_i 's. By considering the $k \lfloor 2 \log_2 t \rfloor$ equations which are powers of 2, one can then solve for the unknowns x_i 's. The complexity of the approach is asymptotically $O(2^{m^2 t})$. Other approaches exist to solve the system [FOPT10, FOP⁺16b] but the complexities of these approaches are more difficult to estimate.

As mentioned in Section 2.4.2, the automorphism-induced Goppa codes are more vulnerable to algebraic attacks as compared to random Goppa codes. This is due to the existence of invariant subcodes which are themselves Goppa codes with smaller supports and lower degree Goppa polynomials. In particular, for an automorphism-induced Goppa code, particularly quasi-cyclic code, such that it has length n , degree t Goppa polynomial and automorphism group of order l , the invariant code has length n/l with degree t/l Goppa polynomial. A thorough security analysis of quasi-cyclic Goppa codes can be found in [BBB⁺18b].

For our construction, only subsets of the automorphism subgroup of $\text{AGL}_2(q)$ are used. In particular, we typically remove an element σ from a subgroup $\mathcal{G}$ of $\text{AGL}_2(q)$ (see Theorem 3.2.8).

Thus, a similar folding operation will be of the form

$$\sum_{\sigma' \in \mathcal{G}, \sigma' \neq \sigma} \sigma'(\mathbf{c})$$

for any $\mathbf{c}$ in the code. Since the coordinates of $\mathbf{c}$ on each orbit are unlikely to be identical, one cannot use the same trick to puncture the code or to construct an invariant code.

However, one may attempt to mount an algebraic attack on our constructions.

In Propositions 3.2.4 and 3.2.5, suppose that V and H are known, respectively (this assumption is valid as one can exhaustively search for them). Referring to the system of equations in 3.1 and the structure of the Goppa support as a disjoint union of cosets, the unknowns for the Goppa support are reduced to finding mt representatives in the orbits. However, since our

choice of $g(x)$ is such that it is not invariant under any automorphism used in the construction, there is no clear relationship between the y_i 's that we can exploit. Consequently, one has to solve a system of equations in mt unknowns in the x_i 's and n unknowns in the y_i 's. Once again, by exploiting the k linear relationships among the y_i 's, one way to solve the system is to perform a brute force search on the remaining free mt y_i 's. Alternatively, one can search through all possible mt x_i 's and solve the resulting linear equations in the y_i 's. In either case, the time complexity is $O(2^{m^2t})$. Hence, unlike the quasi-cyclic or quasi-dyadic constructions, we see that our constructions do not weaken the security of the codes with respect to algebraic attacks.

3.4 Practical Implementation Considerations

3.4.1 Practical Implementation Scheme

Observe that our constructions place some restrictions on the choice of the parameters. For instance, we require n to be an integral multiple of mt . Moreover, the reduction factor is at most $(n-mt)/mt = k/(n-k)$. Thus, to achieve a large reduction size, we need t to be small. However, t needs to be sufficiently large to prevent an exhaustive search on the error vector, or more generally, to guard against the state-of-the-art information set-decoding attacks. In addition, for fixed m and t , n cannot be too large to prevent the distinguishing attack [FGO⁺13]. On the other hand, even though the automorphism-induced codes are more vulnerable to algebraic attacks, the relatively large gap between the complexity of these attacks and the information set-decoding attacks give some room to choose the parameters appropriately for any security level. As such, we propose combining the two constructions to achieve a greater key reduction for a desired security level. The following algorithm gives a possible construction meeting a given security level. Here, we only consider the quasi-cyclic construction as the dyadic construction is similar.

1. Fix a security parameter λ .

2. Fix m such that $q - 1 = 2^m - 1$ has small factors.
3. For any two distinct integers l_1 and l_2 with $l_1 l_2 | (q - 1)$ and $\gcd(l_1, l_2) = 1$, do the following:
 - Pick a t_0 satisfying the following two conditions:
 - $\binom{q}{t_0} \geq 2^\lambda$;
 - A $[l_1 l_2 m t_0, (l_1 l_2 - l_1) m t_0, 2 l_1 t_0]$ Goppa code achieves the security level with respect to the information set-decoding attacks.
4. Let $t = l_1 t_0$ and $n = l_1 l_2 m t_0$.
5. Let H be a subgroup of $\mathbb{F}_q^*$ of order $l_1 l_2$ and pick a generator a of H .
6. Let L be a disjoint union of $m t_0$ different cosets of H and randomly choose an ordering of the cosets.
7. Randomly pick a polynomial $g_0(x)$ of degree t_0 . Let $g(x) = g_0(x^{l_1})$.
8. Let $h(x) = \prod_{i=0}^{l_2-2} g(a^{i l_1} x)$ and $f(x) = \prod_{i=0}^{l_2-1} g(a^{i l_1} x)$.
9. Construct the code $\Gamma(L, g)$, $\Gamma(L, h)$ and $\Gamma(L, f)$.
10. If $\dim(\Gamma(L, f)) \neq 0$ or $\dim(\Gamma(L, h)) \neq m t$ or $\dim(\Gamma(L, g)) \neq (l_2 - 1) m t$, choose another $g_0(x)$.
11. Otherwise, return $\Gamma(L, g)$.

By Proposition 3.2.5, for all $\mathbf{c} \in \Gamma(L, h)$ and for $i = 0, 1, \dots, l_2 - 2$, we have $\sigma_{a^{l_1 i}}(\mathbf{c}) \in \Gamma(L, g)$. Moreover, $\Gamma(L, h)$ is a quasi-cyclic code induced by a subgroup of H of order l_1 . Consequently, one can find a set $P(L, g)$ of cardinality $m t_0$ such that for all $i = 0, 1, \dots, l_2 - 2$, $j = 0, 1, \dots, l_1 - 1$, $\sigma_{a^{l_1 i + l_2 j}}(\mathbf{c}) \in \Gamma(L, g)$ whenever $\mathbf{c} \in P(L, g)$. Similar to Theorem 2.4.12 and Remark 3.2.7, one can construct an $m t_0 \times m t$ matrix that can be extended to a generator matrix of $\Gamma(L, g)$. Consequently, the public key size of such a code is $l_1 m^2 t_0^2$.

In our construction, by our choice of t_0 , we ensure that it is secure against a brute force search on all possible $g_0(x)$ and in fact, it is resistant to the existing known structural attacks (see Section 3.3). In addition, we have ensured that the code $\Gamma(L, g)$ is secure against message recovery attacks.

Remark 3.4.1. *By employing only the quasi-cyclic construction, one gets a public key size of $(l_2 - 1)l_1 m^2 t_0^2$. Thus, we see that our technique can be further applied to quasi-cyclic constructions to yield more compact public keys.*

3.4.2 Concrete Parameters

We apply our constructions to the parameters in BigQuake [BBB⁺18b] to reduce the public key size and to keep the desired security level. We give 3 groups of parameters corresponding to 3 different security levels and each group comprises two code parameters in Table 3.1. The first is the code parameters from BigQuake [BBB⁺18b] and the second is the reduced code parameters we get after applying our scheme to the first code. Now we explain the notations of each column of Table 3.1 as follows.

- λ : security level of the parameters.
- m : extension degree of the finite field.
- $[n, k, t]$ denotes the resulting Goppa code parameters, n for the code length, k for the code dimension and t for the error correcting ability.
- l_1 : the size of the quasi-cyclic group which acts on x .
- l_2 : the size of the partial quasi-cyclic group which acts on the Goppa polynomial $g(x)$.
This column is not applicable to the BigQuake [BBB⁺18b] code parameters.
- ω_{msg} : the logarithm of the workload of ISD, which is computed using the **CaWoF** [Tor17] library.

- $Size(bytes)$: the resulting public key size.

From Table 3.1, we can see that our scheme reduces the public key size of BigQuake by at least half. We remark that the parameters given in Table 3.1 may be vulnerable to other attacks. Here we show the parameters solely to illustrate the power of our construction which, to our best knowledge, are secure against current known attacks. In fact, our construction can be applied to most of the existing Goppa codes based constructions to reduce the public key size without compromising the security level.

Tab. 3.1: More compact public key parameters by applying our construction to BigQuake

λ	m	$[n, k, t]$	l_1	l_2	ω_{msg}	Size (bytes)
AES128	12	[3510, 2418, 91]	13	NA	132	25389
	16	[12240, 11520, 45]	5	17	143	12960
AES192	18	[7410, 4674, 152]	19	NA	195	84132
	18	[8208, 5472, 152]	19	3	213	49248
AES256	18	[10070, 6650, 190]	19	NA	263	149625
	18	[10260, 6840, 190]	19	3	267	76950

Remark 3.4.2. *The public key size of the foregoing scheme does not involve the parameter l_2 . Therefore one could select a bigger l_2 to keep the public key size and ensure that the increased code parameters n and k will still make the code resistant to, even by a large margin improved and powerful ISD attacks in the future. Alternatively, one is able to exploit this property of our scheme to minimize the public key size by selecting very small l_1 and very big l_2 , and increasing t_0 to some reasonable value. Note that this only applies to big enough m , for otherwise it will suffer from some algebraic attacks. In conclusion, one can vary the parameters l_1, l_2, t_0, m, n appropriately to get a compact public key achieving the desired security level with respect to all known attacks.*

4. POLYNOMIAL LATTICES

4.1 Introduction

In this chapter, we build a lattice from polynomial functions which admits some good properties. In particular, we can prove that the determinant and the length of the shortest nonzero vector of the lattice are only related to the degrees of the polynomials used to construct the lattice. Based on the properties of the lattice we constructed, we design a trapdoor function where inverting the trapdoor function is almost equivalent to solving the bounded distance decoding problem in the lattice. The polynomial used to generate the lattice is taken as the trapdoor. With the trapdoor information, one can solve the bounded distance decoding problem for the lattice. We conduct a thorough security analysis on our trapdoor function via the state-of-the-art bounded distance decoding attacks as well as combinatorial attacks. Furthermore, our trapdoor function can be used as a core module to design a quantum resistant public key encryption scheme with acceptable key size.

4.1.1 Comparison

The main contribution of this chapter is to design a new lattice-based trapdoor function whose security relies on the hardness of the BDD problem on lattices. Our construction is analogous to the construction of Goppa codes that admit an efficient decoding based on polynomial operations. Unlike binary Goppa codes, our construction works over finite rings and its error decodability does not depend on the norm of the constructed lattices or the Hamming weight of the associated code.

Similar to the GGH construction which is based on the hardness of the closest vector problem (CVP), our function relies on the infeasibility of the related BDD problem. Nonetheless, we believe that our function provides the following advantages over the GGH construction:

- We do not restrict to a basis that admits a solution via CVP solving algorithms as a trapdoor. Instead, by using lattices constructed from polynomial functions, one can invert the function efficiently as long as one has access to the polynomials and points involved. As such, the vector norms of our basis can be better controlled, one of the main limitations of the GGH scheme.
- For our construction, we do not need to multiply the basis by unimodular matrices to hide the underlying structure.
- Our construction allows us to consider part of the generator matrix as the public key instead of using the entire square matrix, thereby reducing the public key size of our scheme.
- The security of our scheme against embedding attacks can be analyzed concretely.

It is worth to mention that the Chor-Rivest scheme proposed in [CR84] can be regarded as a special case of our scheme by considering an irreducible polynomial in our construction. For our scheme in Section 4.5, we propose to use a product of linear polynomials instead. This has the advantage of making key generation more efficient. Moreover, as discussed in Section 4.4, such a scheme is more secure.

4.1.2 Organization of the Chapter

The remaining of this chapter is organized as follows. In Section 4.2, we describe a family of lattices constructed from polynomials. We then present our new trapdoor functions based on these lattices in Section 4.3. This is followed by a security analysis of these trapdoor functions

in Section 4.4. In Section 4.5, we give details of a semantically secure encryption scheme based on our trapdoor functions. In addition, we propose some possible parameters for our scheme.

4.2 Polynomial Lattices

In this section, we give a new construction of lattices via polynomials over a finite field. Let q be a prime power. We denote by $\mathbb{F}_q$ the finite field with q elements. Let $\mathfrak{R}$ denote the polynomial ring $\mathbb{F}_q[x]$. Fix a monic polynomial $c(x) \in \mathfrak{R}$ of degree d and let $\mathfrak{Q}_{c(x)}$ denote the quotient ring $\mathfrak{R}/c(x)$. Let $\mathfrak{Q}_{c(x)}^*$ denote the unit group of $\mathfrak{Q}_{c(x)}$, i.e., let

$$\mathfrak{Q}_{c(x)}^* = \{\overline{f(x)} \in \mathfrak{Q}_{c(x)} : \gcd(f(x), c(x)) = 1\},$$

where $\overline{f(x)}$ denotes the equivalence class of $f(x)$ in $\mathfrak{Q}_{c(x)}$. It is easy to verify that $\mathfrak{Q}_{c(x)}^*$ forms a multiplicative group. Furthermore, the cardinality of $\mathfrak{Q}_{c(x)}^*$, denoted by $\Phi(c(x))$, is given by the following formula.

Lemma 4.2.1. *[LN97, Lemma 3.69] Let $c(x) \in \mathfrak{R}$. Recall that $c(x)$ has the canonical factorization*

$$c(x) = \prod_{i=1}^t c_i(x)^{e_i},$$

where $c_i(x)$'s are pairwise distinct monic irreducible polynomials over $\mathbb{F}_q$ and d_i 's are the degrees of $c_i(x)$ and $e_i \geq 1$. We have

$$\Phi(c(x)) = \prod_{i=1}^t (q^{e_i d_i} - q^{(e_i-1)d_i}).$$

Let $\alpha_1, \dots, \alpha_n$ be n distinct elements in $\mathbb{F}_q$ such that $c(\alpha_i) \neq 0$ for $i = 1, \dots, n$. Denote by $\mathbf{a}$ the vector $(\alpha_1, \dots, \alpha_n) \in \mathbb{F}_q^n$.

Define the map

$$\begin{aligned} \phi_{\mathbf{a}} : \quad \mathbb{Z}^n &\longrightarrow \mathbb{F}_q(x) \longrightarrow \mathfrak{Q}_{c(x)}^* \\ (u_1, \dots, u_n) &\longmapsto f = \prod_{i=1}^n (x - \alpha_i)^{u_i} \longmapsto f(x) \pmod{c(x)}. \end{aligned}$$

Observe that $\phi_{\mathbf{a}}$ is a group homomorphism from $\mathbb{Z}^n$ to $\mathfrak{Q}_{c(x)}^*$. Let $\mathcal{L}_{\mathbf{a},c(x)}$ denote the kernel of $\phi_{\mathbf{a}}$. As $\mathcal{L}_{\mathbf{a},c(x)}$ is a subgroup of $\mathbb{Z}^n$, $\mathcal{L}_{\mathbf{a},c(x)}$ is a lattice. The following lemma provides some important properties of $\mathcal{L}_{\mathbf{a},c(x)}$.

Lemma 4.2.2. *The lattice $\mathcal{L}_{\mathbf{a},c(x)}$ defined above satisfies the following properties:*

- (i) $\mathcal{L}_{\mathbf{a},c(x)}$ has rank n .
- (ii) The determinant $\det(\mathcal{L}_{\mathbf{a},c(x)})$ is upper bounded by $\Phi(c(x))$. Furthermore, $\det(\mathcal{L}_{\mathbf{a},c(x)}) = \Phi(c(x))$ if $\phi_{\mathbf{a}}$ is surjective.
- (iii) $\lambda_1(\mathcal{L}_{\mathbf{a},c(x)}) \geq \sqrt{d}$. Moreover, $\lambda_1(\mathcal{L}_{\mathbf{a},c(x)}) = \sqrt{d}$ if and only if there exists a lattice point $\mathbf{v}$ of the form $\mathbf{v} = \pm \mathbf{v}_0$, where $\mathbf{v}_0$ has d nonzero entries which are all equal to 1.

Proof. (i) Observe that for each $i = 1, 2, \dots, n$, we have

$$(0, 0, \Phi(c(x)), \dots, 0) \mapsto (x - \alpha_i)^{\Phi(c(x))} \mapsto 1$$

under $\phi_{\mathbf{a}}$. Hence, each of these points is in $\mathcal{L}_{\mathbf{a},c(x)}$. As these n points are clearly linearly independent, they form a sub-lattice of $\mathcal{L}_{\mathbf{a},c(x)}$ of rank n . Consequently, $\mathcal{L}_{\mathbf{a},c(x)}$ has rank n .

- (ii) As $\mathbb{Z}^n / \mathcal{L}_{\mathbf{a},c(x)} \simeq \text{Im}(\phi_{\mathbf{a}}) \leq \mathfrak{Q}_{c(x)}^*$ and $\det(\mathcal{L}_{\mathbf{a},c(x)}) = [\mathbb{Z}^n : \mathcal{L}_{\mathbf{a},c(x)}] \det(\mathbb{Z}^n) = [\mathbb{Z}^n : \mathcal{L}_{\mathbf{a},c(x)}] = |\text{Im}(\phi_{\mathbf{a}})|$, we obtain the desired inequality. In addition, if $\phi_{\mathbf{a}}$ is surjective, then $\mathbb{Z}^n / \mathcal{L}_{\mathbf{a},c(x)} \simeq \mathfrak{Q}_{c(x)}^*$. Hence the equality follows.

(iii) Let $\mathbf{v} = (v_1, v_2, \dots, v_n)$ be a nonzero point in $\mathcal{L}_{\mathbf{a}, c(x)}$. Denote by I and J the sets $\{1 \leq i \leq n : v_i > 0\}$ and $\{1 \leq j \leq n : v_j < 0\}$, respectively. By definition of $\mathcal{L}_{\mathbf{a}, c(x)}$, we have

$$\prod_{i \in I} (x - \alpha_i)^{v_i} \prod_{j \in J} (x - \alpha_j)^{v_j} - 1 \equiv 0 \pmod{c(x)},$$

i.e., the nonzero polynomial $\prod_{i \in I} (x - \alpha_i)^{v_i} - \prod_{j \in J} (x - \alpha_j)^{-v_j}$ is divisible by $c(x)$. Hence,

$$\deg\left(\prod_{i \in I} (x - \alpha_i)^{v_i} - \prod_{j \in J} (x - \alpha_j)^{-v_j}\right) \geq d,$$

i.e., $\sum_{i \in I} v_i \geq d$ or $\sum_{j \in J} -v_j \geq d$. This gives $\sum_{i=1}^n |v_i| \geq d$. Therefore, $\|\mathbf{v}\| = \sqrt{\sum_{i=1}^n v_i^2} \geq \sqrt{\sum_{i=1}^n |v_i|} \geq \sqrt{d}$ (note that each v_i is an integer). If there exists a lattice point with d nonzero entries which are either all 1 or -1 , then it is clear that $\lambda_1(\mathcal{L}_{\mathbf{a}, c(x)}) = \sqrt{d}$. Conversely, assume $\lambda_1(\mathcal{L}_{\mathbf{a}, c(x)}) = \sqrt{d}$. Then there exists a nonzero lattice point $\mathbf{v} = (v_1, v_2, \dots, v_n)$ in $\mathcal{L}_{\mathbf{a}, c(x)}$ such that $\|\mathbf{v}\| = \sqrt{\sum_{i=1}^n v_i^2} = \sqrt{d}$. Since

$$\deg\left(\prod_{i \in I} (x - \alpha_i)^{v_i} - \prod_{j \in J} (x - \alpha_j)^{-v_j}\right) \geq d,$$

we must have that either

$$I = \emptyset \text{ and } \sum_{j \in J} -v_j = d$$

or

$$J = \emptyset \text{ and } \sum_{i \in I} v_i = d.$$

This forces that either $v_i = 1$ for all $i \in I$ or $v_j = -1$ for all $j \in J$.

□

According to Lemma 4.2.2 (iii), we see that $\mathcal{L}_{\mathbf{a}, c(x)}$ has minimum norm $\lambda_1(\mathcal{L}_{\mathbf{a}, c(x)}) = \sqrt{d}$

when there exist $i_1, \dots, i_d \in [n]$ such that

$$\prod_{j=1}^d (x - \alpha_{i_j}) = 1 + c(x) \text{ or } \prod_{j=1}^d (x - \alpha_{i_j})^{-1} = 1 + c(x).$$

It follows that there are at most $2\binom{n}{d}$ different $c(x) \in \mathfrak{R}$ of degree d out of a total of q^d such polynomials such that $\mathcal{L}_{\mathbf{a}, c(x)}$ satisfies $\lambda_1(\mathcal{L}_{\mathbf{a}, c(x)}) = \sqrt{d}$. In other words, given a polynomial $c(x)$ of degree d and an $(\alpha_1, \dots, \alpha_n) \in \mathbb{F}_q^n$, the probability that the lattice $\mathcal{L}_{\mathbf{a}, c(x)}$ has minimum norm $\sqrt{d}$ is less than $2\binom{n}{d}/q^d < 1/d!$ and we can expect the minimum norm of the lattice $\mathcal{L}_{\mathbf{a}, c(x)}$ to be bigger. In particular, we will use the Gaussian heuristic to estimate the minimum norm of the lattices. Assume that the map $\phi_{\mathbf{a}}$ is surjective. By Lemma 4.2.2 (ii) and Lemma 4.2.1, the determinant of $\mathcal{L}_{\mathbf{a}, c(x)}$ is approximately q^d . The Gaussian heuristic suggests that a random lattice of dimension n and determinant q^d has minimum norm approximately $\sqrt{n/2\pi e} q^{d/n}$.

Next, we describe how to construct the set $\mathbf{a} = (\alpha_1, \dots, \alpha_n) \subset \mathbb{F}_q^n$ for which $\mathcal{L}_{\mathbf{a}, c(x)}$ admits a nice basis for a class of $c(x)$. In the following, randomness will always refer to picking an element randomly from a uniform distribution. Further, we assume that $c(x)$ is of the form $c(x) = c_1(x) \dots c_t(x)$, where $c_i(x)$'s are distinct irreducible polynomials over $\mathbb{F}_q$, each having degree d_0 . Hence, $\mathfrak{Q}_{c(x)} \cong \oplus_{i=1}^t \mathbb{F}_{q^{d_0}}$. Let β denote a generator of $(\mathbb{F}_{q^{d_0}})^*$.

Let $\alpha_{n-t+1}, \dots, \alpha_n$ be t distinct elements in $\mathbb{F}_q$. For $i = 1, \dots, t$ and $j = 1, \dots, t$, let $\gamma_{ij} = x - \alpha_{n-t+i} \pmod{c_j(x)}$. Let $m_{ij} = \log_{\beta} \gamma_{ij}$ and $M = (m_{ij})_{i=1, \dots, t, j=1, \dots, t}$.

Suppose that M is invertible over the ring $\mathbb{Z}_{q^{d_0}-1}$. For each $\alpha \in \mathbb{F}_q$ with $\alpha \neq \alpha_{n-t+1}, \dots, \alpha_n$, let $\mathbf{y} = (y_1, \dots, y_t)$, where $y_j = \log_{\beta}((x - \alpha) \pmod{c_j(x)})$, $j = 1, \dots, t$. Let $\mathbf{g}_{\alpha} = \mathbf{y}M^{-1} \pmod{q^{d_0}-1}$. Write $\mathbf{g}_{\alpha} = (g_{\alpha, n-t+1}, \dots, g_{\alpha, n})$. Note that for each $j = 1, \dots, t$,

$$y_j = \sum_{i=1}^t g_{\alpha, n-t+i} m_{ij} \pmod{q^{d_0}-1}. \quad (4.1)$$

For $j = 1, \dots, t$, we have

$$\begin{aligned}
\prod_{i=1}^t (x - \alpha_{n-t+i})^{g_{\alpha, n-t+i}} \mod c_j(x) &\equiv \prod_{i=1}^t (\gamma_{ij}^{m_{ij}})^{g_{\alpha, n-t+i}} \mod c_j(x) \\
&\equiv \prod_{i=1}^t (\beta^{m_{ij}})^{g_{\alpha, n-t+i}} \mod c_j(x) \\
&\equiv \beta^{\sum_{i=1}^t g_{\alpha, n-t+i} m_{ij}} \mod c_j(x) \\
&\equiv \beta^{y_j} \mod c_j(x) \equiv x - \alpha \mod c_j(x).
\end{aligned}$$

Note that the first equality follows from the definition of γ_{ij} , the second follows from the definition of the entry m_{ij} of the matrix M , the fourth is derived from the relation of $\mathbf{y}$ and $\mathbf{g}_\alpha$ as shown in 4.1, and the last follows from the definition of y_j . Since it holds for any $c_j(x)$, it follows that $x - \alpha \equiv \prod_{i=1}^t (x - \alpha_{n-t+i})^{g_{\alpha, n-t+i}} \mod c(x)$. Consequently, the point $(0, \dots, 1, 0, \dots, -g_{\alpha, n-t+1}, \dots, -g_{\alpha, n})$, where 1 is in the entry indexed by α is a point in $\mathcal{L}_{\mathbf{a}, c(x)}$ for any $\alpha \in (\alpha_1, \dots, \alpha_{n-t})$, which follows the definition of $\mathcal{L}_{\mathbf{a}, c(x)}$ and the map $\phi_{\mathbf{a}}$ in Lemma 4.2.1.

Proposition 4.2.3. *Let $\alpha_{n-t+1}, \dots, \alpha_n$ be t distinct elements in $\mathbb{F}_q$ with the matrix M as above. Suppose that M is invertible over $\mathbb{Z}_{q^{d_0}-1}$. Pick $\alpha_1, \dots, \alpha_{n-t}$ randomly from $\mathbb{F}_q$ such that $\mathbf{a} = (\alpha_1, \dots, \alpha_n)$ contains n distinct elements. Define G as the $(n-t) \times t$ matrix with rows given by $\mathbf{g}_{\alpha_i}$, for $i = 1, \dots, n-t$. A basis of the lattice $\mathcal{L}_{\mathbf{a}, c(x)}$ is given by:*

$$B_{\mathbf{a}, c(x)} = \begin{pmatrix} I_{n-t} & -G \\ 0_{t \times (n-t)} & (q^{d_0} - 1)I_t \end{pmatrix},$$

where I_r denotes the identity matrix of rank r .

Proof. According to the preceding arguments, the first $n-t$ rows of $B_{\mathbf{a}, c(x)}$ are points in $\mathcal{L}_{\mathbf{a}, c(x)}$. Since $(x - \alpha_i)^{q^{d_0}-1} \equiv 1 \mod c(x)$ for $i = n-t+1, \dots, n$, the last t rows of $B_{\mathbf{a}, c(x)}$ are also in $\mathcal{L}_{\mathbf{a}, c(x)}$. Clearly, the rows of the matrix are linearly independent. It remains to show that

the rows span $\mathcal{L}_{\mathbf{a},c(x)}$. Let $\mathbf{u} = (u_1, \dots, u_n)$ be a point in $\mathcal{L}_{\mathbf{a},c(x)}$ so that $\prod_{i=1}^n (x - \alpha_i)^{u_i} \equiv 1 \pmod{c(x)}$. Consider the point $\mathbf{v} = \mathbf{u} - \sum_{i=1}^{n-t} u_i B_i$, where B_i denotes the i -th row of $B_{\mathbf{a},c(x)}$. Hence, $\mathbf{v} \in \mathcal{L}_{\mathbf{a},c(x)}$ and we can write $\mathbf{v} = (0, \dots, 0, v_{n-t+1}, \dots, v_n)$. It is sufficient to show that $v_{n-t+i} \equiv 0 \pmod{q^{d_0} - 1}$ for $i = 1, \dots, t$. In other words, $\prod_{i=1}^t (x - \alpha_{n-t+i})^{v_{n-t+i}} \equiv 1 \pmod{c(x)}$, equivalently, $\prod_{i=1}^t (x - \alpha_{n-t+i})^{v_{n-t+i}} \equiv 1 \pmod{c_j(x)}$ for $j = 1, \dots, t$. Now,

$$\prod_{i=1}^t (x - \alpha_{n-t+i})^{v_{n-t+i}} \equiv \prod_{i=1}^t (\beta^{m_{ij}})^{v_{n-t+i}} \equiv \beta^{\sum_{i=1}^t v_{n-t+i} m_{ij}} \equiv 1 \pmod{c_j(x)}$$

which gives $(v_{n-t+1}, \dots, v_n)M = 0 \pmod{q^{d_0} - 1}$. Since M is invertible, we conclude that $v_{n-t+i} \equiv 0 \pmod{q^{d_0} - 1}$ for $i = 1, \dots, t$. $\square$

Remark 4.2.4. Note that the lattices $\mathcal{L}_{\mathbf{a},c(x)}$ for different pairs of $\mathbf{a}$ and $c(x)$ are not all distinct. For instance, let $\mathbf{a} = (\alpha_1, \dots, \alpha_n)$ and let $\gamma \neq 0 \in \mathbb{F}_q$. Let $\mathbf{a}' = (\alpha_1 + \gamma, \dots, \alpha_n + \gamma)$ and $c'(x) = c(x - \gamma)$. Then, it is easy to check that $\mathcal{L}_{\mathbf{a},c(x)} = \mathcal{L}_{\mathbf{a}',c'(x)}$.

Next, we analyze the complexity of constructing $B_{\mathbf{a},c(x)}$. First, one needs to compute about tn discrete logs in the field $\mathbb{F}_{q^{d_0}}$. The discrete logarithm problem over finite fields is one of the fundamental hard problems widely used in cryptography. Extensive studies have been done in this area and various methods have been proposed to solve the discrete logarithm problem over finite fields. In particular, it is adequate for us to employ Pollard's rho method to compute discrete logarithm with time complexity $O(\sqrt{q^{d_0}})$. Please refer to the survey paper [JP16] for the state-of-the-art results on the discrete logarithm problem. For our construction, we have $r = q^{d_0}$. For $q = O(n)$ and $d_0 = O(1)$, it follows that solving the discrete log is efficient.

Remark 4.2.5. To achieve a more practical scheme, we suggest that the user chooses d_0 as a small constant. Otherwise, computing the discrete logarithm in the field $\mathbb{F}_{q^{d_0}}$ may not be so efficient. For our concrete construction in Section 4.5, we choose $d_0 = 1$.

Second, one needs to pick $\alpha_{n-t+1}, \dots, \alpha_n$ so that the matrix M is invertible. Now, each

entry m_{ij} is the discrete log of $x - \alpha_{n-t+i} \bmod c_j(x)$. Since α_{n-t+i} is random, we may assume that the matrix M is a random matrix in the ring $\mathbb{Z}_{q^{d_0}-1}$.

Lemma 4.2.6. *[Ngu99, Theorem 2] Let $s = q^{d_0} - 1$ be a positive integer. Let $p_1, \dots, p_m$ be the distinct prime divisors of s . The probability that a random $t \times t$ matrix in $\mathbb{Z}_s$ is invertible is*

$$P_s = \prod_{i=1}^m \prod_{j=1}^t (1 - p_i^{-j}).$$

It can be seen from the formula that the probability of a random $t \times t$ matrix being invertible is non-negligible.

Remark 4.2.7. *One sees that our construction bears some similarities to the Goppa code construction. Recall that the binary Goppa code with Goppa support $\mathbf{a}$ and Goppa polynomial $c(x)$ consists of $(c_1, \dots, c_n) \in \mathbb{F}_2^n$ such that*

$$\sum_{i=1}^n \frac{c_i}{x - \mathbf{a}_i} \equiv 0 \pmod{c(x)}.$$

Our construction in this chapter is defined over the finite ring $\mathbb{Z}_{q^{d_0}-1}$ instead of over finite fields. As seen in Theorem 4.3.2, the error decodability of our construction is up to $\deg(c(x)) - 1$, independent of the shortest length of the lattice or the Hamming weight of the corresponding code.

Remark 4.2.8. *Note that distinct degrees of $c_i(x)$'s are still possible. If so, the discrete logarithm should be computed in different fields when computing the matrix M , which also makes the notation too complex there. For simplicity, we just assume that all the $c_i(x)$'s are of the same degree. In particular, in our concrete construction, we let the degree of every $c_i(x)$ be 1.*

4.3 Construction of Our Trapdoor Functions

In this section, we describe new trapdoor functions where inverting the function amounts to solving the BDD for the associated lattices. Unlike the GGH construction, we do not generate two different bases of a lattice. Instead, we require only one basis of our polynomial lattice as the trapdoor involves information to construct the polynomial lattice. Recall that a trapdoor function encompasses four different sub-algorithms, namely, *generate*, *sample*, *evaluate* and *invert*. We will now present each of these in detail.

Generate: Set the public parameters q, n, d according to the desired security level (see the next section for details) such that $\sqrt{n/2\pi e}q^{d/n} > 2\sqrt{d-1}$. Let $d = d_0 t$. Choose t irreducible polynomials $c_i(x)$ of degree d_0 and let $c(x) = c_1(x) \dots c_t(x)$. Choose a set $\mathbf{a} = (\alpha_1, \dots, \alpha_n) \in \mathbb{F}_q^n$ such that $c(\alpha_i) \neq 0$ for $i = 1, 2, \dots, n$ and the elements $\alpha_{n-t+1}, \dots, \alpha_n$ satisfy the conditions in Proposition 4.2.3. Construct the basis $B_{\mathbf{a}, c(x)}$ of the lattice $\mathcal{L}_{\mathbf{a}, c(x)}$ as described in Proposition

4.2.3. Write $B_{\mathbf{a}, c(x)} = \begin{pmatrix} I_{n-t} & -G \\ 0_{t \times (n-t)} & (q^{d_0} - 1)I_t \end{pmatrix}$.

Let $H = B'_{\mathbf{a}, c(x)} = \begin{pmatrix} I_{n-t} & -G \end{pmatrix}$. The trapdoor for our function includes the polynomial $c(x)$ and the set $\mathbf{a} = (\alpha_1, \dots, \alpha_n)$.

Sample: Randomly sample $\mathbf{m} \in \mathbb{Z}_{q^{d_0}-1}^{n-t}$ and the error $\mathbf{e} = (e_1, \dots, e_n) \in \{0, 1\}^n$ satisfying: $\sum_{i=1}^n e_i = d - 1$.

Evaluate: For each input $\mathbf{m} \in \mathbb{Z}_{q^{d_0}-1}^{n-t}$, the function f is evaluated on $\mathbf{m}$ as

$$\mathbf{c} = f(\mathbf{m}, \mathbf{e}) = \mathbf{m}H + \mathbf{e} \pmod{q^{d_0} - 1}.$$

Invert: Suppose that we are given a valid output $\mathbf{c} = (c_1, \dots, c_n)$ of the function f . The inversion process is as follows.

Step 1: Compute

$$r(x) = \prod_{i=1}^n (x - \alpha_i)^{c_i} \mod c(x).$$

Step 2: Factorize $r(x)$ over the ring $\mathbb{F}_q[x]$ as $r(x) = \prod_{i=1}^n (x - \alpha_i)^{u_i}$. Let $\mathbf{u} = (u_1, u_2, \dots, u_n)$.

Step 3: Compute $\mathbf{v}' = \mathbf{c} - \mathbf{u}$. Write $\mathbf{v}' = (v'_1, \dots, v'_n)$.

Step 4: Let $\mathbf{m}' = (v'_1, \dots, v'_{n-t})$.

Without knowledge of the trapdoor, observe that inverting the function will require us to find the error $\mathbf{e}$ or equivalently, a point $\mathbf{t}$ in $\mathcal{L}_{\mathbf{a}, c(x)}$ such that $\|\mathbf{c} - \mathbf{t}\| = \sqrt{d-1}$. Concretely, one will use the basis formed by the rows of the matrix $\begin{pmatrix} H \\ 0 \quad (q^{d_0} - 1)I_t \end{pmatrix}$

Thus, one needs to be able to solve BDD with respect to this basis. We will discuss more about this in the next section.

Remark 4.3.1. Here, we have chosen the parameters such that the expected shortest vector in the lattice $\mathcal{L}_{\mathbf{a}, c(x)}$ is greater than $2\sqrt{d-1}$ so we are in fact solving BDD with $\gamma < 1/2$. Thus, a solution of the BDD will yield the error vector.

The following theorem shows that the inversion process indeed recovers $\mathbf{m}$.

Theorem 4.3.2. Let $\mathbf{m}$ be an element in $\mathbb{Z}_{q^{d_0}-1}^n$ and let $\mathbf{c}$ be the output produced by the **Evaluate** algorithm. Let $\mathbf{m}'$ be the output of the **Invert** algorithm. Then $\mathbf{m}' = \mathbf{m}$.

Proof. First, we have $\mathbf{c} = \mathbf{m}B'_{\mathbf{a}, c(x)} + \mathbf{e}$. We claim that $\mathbf{v} = \mathbf{v}'$, where $\mathbf{v} = \mathbf{m}B'_{\mathbf{a}, c(x)}$. To see this, note that

$$\begin{aligned} \prod_{i=1}^n (x - \alpha_i)^{c_i} &= \prod_{i=1}^n (x - \alpha_i)^{v_i + e_i} = \prod_{i=1}^n (x - \alpha_i)^{v_i} \prod_{i=1}^n (x - \alpha_i)^{e_i} \\ &\equiv 1 \cdot \prod_{i=1}^n (x - \alpha_i)^{e_i} \mod c(x) \equiv \prod_{i=1}^n (x - \alpha_i)^{e_i} \mod c(x) \end{aligned}$$

Since $\sum_{i=1}^n e_i = d - 1 < d$, we must have $r(x) = \prod_{i=1}^n (x - \alpha_i)^{e_i}$, so $u_i = e_i$. Therefore, $\mathbf{v}' = \mathbf{c} - \mathbf{u} = \mathbf{m}B'_{\mathbf{a},c(x)} + \mathbf{e} - \mathbf{u} = \mathbf{v}$ and the claim is proved.

Note that we have $\mathbf{v} = \mathbf{m}B'_{\mathbf{a},c(x)} = (\mathbf{m}, -\mathbf{m}G)$. Therefore, we have $(v_1, \dots, v_{n-t}) = \mathbf{m} \bmod q^{d_0} - 1$. $\square$

Remark 4.3.3. • *In general, we like to have as many nonzero entries of the error as possible. Hence, we choose e_i to take small values. In particular, we typically let $e_i = 1$.*

- *Instead of letting all the error entries be positive, we can equivalently let them be all negative. In this case, in the inversion process, one needs to check if $r(x)$ or $1/r(x) \bmod c(x)$ can be factorized. In the former case, we have the usual case where $e_i \geq 0$. In the latter case, it is easy to verify that we have $e_i \leq 0$, that is all the nonzero entries of the error are -1 .*
- *For the inversion process, one can simply check if $r(\alpha_i) = 0$ to check if $u_i = 0$ or 1 .*

Remark 4.3.4. • *Here, only the right part $-G$ of the matrix $H = \begin{pmatrix} I_{n-t} & -G \end{pmatrix}$, which is used to evaluate the function, is undetermined. It is an $(n-t) \times t$ matrix over $\mathbb{Z}_{q^{d_0}-1}$ and thus, has size $(n-t)td_0 \log_2 q$ bits.*

- *Unlike the GGH scheme, inversion does not require solving the CVP. Instead, inversion is carried out using properties of polynomials and remainders.*
- *In the above scheme, the first $n-t$ positions of $\mathbf{c}$ may contain some information about $\mathbf{m}$. This is because we have only introduced error to $d-1$ positions, and thus, at least $n-t-d+1$ positions will be in the clear. In Section 4.5, we present a practical encoding scheme to mask the original message m .*
- *Apart from the above scheme, other modifications are possible. Randomly pick an $(n-t) \times (n-t)$ unimodular matrix T with small entries and an $n \times n$ permutation matrix P . Construct $H = TB'_{\mathbf{a},c(x)}P \bmod q^{d_0} - 1$. The left part of the new matrix H will hide all*

the information of message m . This is analogous to the classic McEliece scheme where two secret matrices are used to hide the structure of the underlying code used. However, in situations where the inputs are completely random, the roles of T and P will not be so critical. Indeed, multiplying by a permutation matrix P serves to permute the elements in the set $\mathbf{a}$, which can be achieved if the elements in $\mathbf{a}$ are chosen randomly. Similarly, the unimodular matrix T will not improve the security of the scheme as given a matrix H , one can efficiently transform it such that the left portion is a scalar matrix.

4.4 Security Analysis of Our Trapdoor Functions

In deciding the parameters for our scheme, we will like to achieve the following:

- The public key size should be reasonably small;
- Key generation, encryption and decryption should be efficient;
- The scheme is resistant against all existing attacks.

We now discuss some possible attacks on our scheme to help us decide the appropriate parameters. First, suppose that $d \geq n/2 + 1$. Let $\mathbf{c}$ be a valid output with error $\mathbf{e}$. Then, $\mathbf{e}$ has $d - 1$ entries = 1. Consider $\mathbf{c}' = \mathbf{c} - (1, \dots, 1)$. It is $\mathbf{c}' = \mathbf{m}H + \mathbf{e} - (1, \dots, 1) = \mathbf{m}H + \mathbf{e}'$, where $\mathbf{e}'$ has $< n/2$ entries = -1. Hence, one may decrypt using $\mathbf{c}'$ instead. (refer to Remark 4.3.3 on how one can decrypt when the entries are negative.) It follows that we may assume that $d \leq n/2$. Therefore, we have $t \leq d \leq n/2$ and $n \leq q$.

4.4.1 Error Search

At first glance, it appears that one needs to search through all $\binom{n}{d-1}$ entries to find the error. However, one can in fact reduce the search in the following way. It is obvious that the first $n - t$ columns of H are linearly independent. Let $I = \{1, \dots, n - t\}$. Search through all possible error positions in I . Recall that there are at most $d - 1$ such positions. Assuming that the error

bits are uniformly distributed, the number of nonzero error bits in these positions is roughly $l = \frac{(n-t)(d-1)}{t}$. Consequently, the number of tries is around $\binom{n-t}{l}$.

Let $\mathbf{c}_I$ denote the vector formed by the entries in $\mathbf{c}$ indexed by I . For each guess $\mathbf{e}_I$, define $\mathbf{x} = \mathbf{c}_I - \mathbf{e}_I \pmod{q^{d_0} - 1}$. To verify if our guess is correct, check if $\mathbf{c} - \mathbf{x}H$ is of the correct error form, that is, contains exactly $d - 1$ 1's and all other entries are 0.

Consequently, the complexity of this attack is $O(\binom{n-t}{l}(n-t)t)$.

4.4.2 Search for the Trapdoor

One obvious way to attack the function is to find the trapdoor information. We will need to search for $c(x)$ and $\mathbf{a} = (\alpha_1, \dots, \alpha_n)$. One way to do this is as follows:

- Exhaustively search for the polynomial $c(x)$. There are $O(q^d)$ different $c(x)$ of degree d of the form $c(x) = c_1(x) \dots c_t(x)$.
- For each $c(x)$, guess the ordered set $(\alpha_{n-t+1}, \dots, \alpha_n)$. For each such set, determine if there exist $\alpha_1, \dots, \alpha_{n-t}$ that satisfy the matrix $B'_{\mathbf{a}, c(x)} = \begin{pmatrix} I_{n-t} & -G \end{pmatrix}$. Let $-G = (b_{i, n-t+j})_{i=1, \dots, t, j=1, \dots, t}$. Specifically, from the definition of $\mathcal{L}_{\mathbf{a}, c(x)}$ and $B'_{\mathbf{a}, c(x)}$, we can construct α_i by checking for α_i such that $(x - \alpha_i) \times \prod_{j=1}^t (x - \alpha_{n-t+j})^{b_{i, n-t+j}} \equiv 1 \pmod{c(x)}$ for each $i \in \{1, \dots, n-t\}$. Our guesses of $c(x)$ and α_i 's are correct if we can reconstruct $(\alpha_1, \dots, \alpha_{n-t})$ by the preceding procedure. There are ${}^{n-t}P_n \approx n^{n-t}$ possible ordered sets $(\alpha_{n-t+1}, \dots, \alpha_n)$.

The overall complexity of this attack is $O(q^d n^{n-t} (n-t) d^2 (\log q)^2)$ if we assume that the complexity of polynomial multiplication in the ring $\mathbb{F}_q[x]/c(x)$ is $O(d^2 (\log q)^2)$.

Remark 4.4.1. In Remark 4.2.4, we have discussed that for each $c(x)$ and ordered set $\mathbf{a}$, there are at least $q - 1$ different choices of pairs of polynomials and ordered sets that give rise to the same lattice. Hence, for a more precise complexity, one will need to take into account these

equivalent lattices. However, as this number is relatively small and has a negligible impact on the complexity, we will omit these details in our complexity analysis.

4.4.3 Embedding Attack to Solve BDD

As mentioned in the preceding section, one way to recover the message is to solve the BDD problem on the lattice given the output. A common way to solve BDD is via the embedding technique (see Section 2.3). We now investigate the cost of this attack.

There are two typical estimates for the costs of the embedding attack to recover the error vector when employing BKZ to solve the uSVP. The first was proposed in [GN08] and the second was sketched in [ADPS16]. The predicted costs of the two estimates are quite different. [AGVW17] investigated these two estimates and suggested that the latter estimate is more indicative of the cost of the attack. As such, we will use the second estimate in our analysis of the embedding attack.

Briefly, this estimate forecasts that the attack outputs the correct error $\mathbf{e}$ if

$$\sqrt{\beta/n}\|(\mathbf{e}, 1)\| \leq \delta^{2\beta-n} \det(\mathcal{L})^{1/n} \quad (4.2)$$

under the assumption that GSA holds, where n is the dimension of the full rank lattice $\mathcal{L}$, β is the block size when running the BKZ reduction. A concise explanation of the condition is that the projection of the error vector $\mathbf{e}$ onto the last β vectors is shorter than the $b_{n-\beta+1}^*$ under the GSA assumption. Then an SVP oracle call on the last block of size β can find the error vector $\mathbf{e}$.

It follows that in our situation, we will require β to satisfy

$$\sqrt{\beta/n}\|(\mathbf{e}, 1)\| \geq \delta^{2\beta-n} (q-1)^{d/n}.$$

To achieve a desired level λ , one will need to ensure that the cost of running BKZ- β given

by Equation 2.1 to be at least 2^λ .

The embedding attack can be enhanced by combining with partial search of the error bits [APS15]. Specifically, if k nonzero error bits are guessed correctly, the remaining error norm will be reduced to $\sqrt{d-k}$, thereby making the gap from $\lambda_1(\mathcal{L}_{\mathbf{a},c(x)})$ bigger. In fact, the dimension of the embedding lattice is now reduced from n to $n-k$. Hence, we need to ensure that $\binom{n}{k}$ times of each single BKZ execution will be infeasible to carry out.

4.4.4 Attack When $c(x)$ Is Irreducible

Note that with knowledge of the public information G , one can easily construct the matrix $B_{\mathbf{a},c(x)}$. The question is whether this matrix will leak information about the polynomial $c(x)$ as well as $\mathbf{a}$. Here, we discuss a possible attack when $c(x)$ is irreducible over $\mathbb{F}_q$, that is, $t = 1$.

In this case, the matrix $B_{\mathbf{a},c(x)}$ takes a very simple form, namely,

$$B_{\mathbf{a},c(x)} = \begin{pmatrix} I_{n-1} & -G \\ 0_{1 \times (n-1)} & q^d - 1 \end{pmatrix},$$

where G is a $(n-1) \times 1$ column. Write G as $G = \begin{pmatrix} g_1 \\ g_2 \\ \vdots \\ g_{n-1} \end{pmatrix}$, where each $g_i \in \mathbb{Z}_{q^d-1}$. Note that

g_i satisfies $x - \alpha_i \equiv (x - \alpha_n)^{g_i} \pmod{c(x)}$. Without any loss of generality, we may assume that $\alpha_n = 0$ (by substituting x by $x - \alpha_n$ in the whole system). It follows that $c(x)$ is a common factor of the polynomials $x - \alpha_i - x^{g_i}$, $i = 1, 2, \dots, n-1$. Since $c(x)$ is irreducible over $\mathbb{F}_q$ of degree d , it is a factor of $x^{q^d} - x$.

We can now perform the following steps to recover $c(x)$ and the α_i 's.

- Randomly guess α_1 . For each α_1 , compute the greatest common divisor $h(x) = \gcd(x - \alpha_1 - x^{g_1}, x^{q^d} - x)$. Find all pairs $\alpha_1, d(x)$ such that $d(x)$ is irreducible of degree d and

divides the polynomial $h(x)$.

- For each pair $\alpha_1, d(x)$ found above, test for α_2 such that $d(x)$ is also a factor of $x - \alpha_2 - x^{g^2}$.
- Continue the process until one $d(x)$ is left. Let $c(x)$ to be this $d(x)$.
- Find the remaining α_i 's by direct computation of $\alpha_i = x - x^{g^i} \bmod d(x)$.

We remark that with high probability, the set of possible $d(x)$ after the first step will be very small. It follows that the main complexity of the above attack comes from performing the gcd computations to find gcd of polynomials of the form $x - \alpha - x^g$ and $x^{q^d} - x$. In general, such a gcd computation has complexity polynomial in g . Furthermore, with high probability, g is of the order of q^d . Consequently, in general, the above attack has complexity polynomial in q^d . However, the above attack works if g is small or is of a special form that makes the gcd computation easy.

The above attack easily generalizes to the case when $t > 1$ but the complexity increases as well. Specifically, we will need to guess t different values of α in the first step. This has complexity $n!/(n-t)! \approx n^t$. In view of these considerations, we will choose t to be as big as possible, say $c(x)$ is a product of linear or quadratic polynomials.

4.4.5 Attack Based on Chor-Rivest Scheme

When $c(x)$ is irreducible, our scheme can be regarded as a variant of the Chor-Rivest scheme [CR84], which is a knapsack based cryptosystem. The public key in Chor-Rivest scheme consists of q integers, which are the discrete logarithms of q field elements in the extension field $\mathbb{F}_q[x]/(c(x))$, shifted by a random number v and then permuted by a random permutation. When $v = 1$, these integers in fact constitute the matrix G described in the preceding attack. In the final version of Chor-Rivest scheme [CR88, 7.1.e], they showed a sub-exponential attack attributed to Ernest Brickell by Chor and Rivest. The attack consists of two major steps. The first step computes a very small weight and small Euclidean norm vector in the dual space of

the public key such that the sum of the entries in the vector is also zero by employing a tweaked Laarias-Odlyzko lattice. The second step uses the newfound small vector to facilitate guessing the private permutation and the private shift number to proceed a successful attack.

We next consider the situation when $c(x)$ is a product of t small degree polynomials, say $c(x)$ is a product of d linear polynomials. In this case, the public key will no longer be one column (as in the case of the Chor-Rivest scheme) but comprises d columns. In the following, a *short vector* refers to a vector of small Hamming weight and small Euclidean norm. Following the attack amounts to find a short coefficient vector for the public key such that the linear combination of the public key is a zero vector. By considering the lattice formed by the rows of the public key, this problem is equivalent to finding short vectors in the dual lattice. It is not clear that very short vectors exist and our experiments for different choices of parameters have not produced such vectors.

Even if short vectors exist, finding any one of them is believed to be a difficult problem. As such, we believe that in the case when $c(x)$ is a product of linear polynomials, the scheme will not suffer from the attack on the Chor-Rivest scheme.

4.5 A Practical Encryption Scheme

4.5.1 Description

Similar to the GGH encryption scheme and the McEliece encryption scheme, in order to transit from the one-way trapdoor function to an encryption scheme, one needs a method to encode the message before passing to the trapdoor function. In particular, the chosen encoding scheme should ensure that the encryption scheme is semantically secure.

Recall that a cryptosystem is said to achieve CCA2 security if an attacker has no advantage to decipher a given ciphertext given the plaintexts of other (possibly selected) ciphertexts. It is indistinguishable in the CCA2-model if the attacker, when provided with two plaintexts and a ciphertext, has no advantage to determine which plaintext correctly corresponds to the

ciphertext. Different proposals were presented in [GGH97, KI01, HHK17] to achieve semantic (or CCA2) security for the GGH scheme and the McEliece scheme. We first show why the scheme employed in [GGH97] will not work for our construction.

Recall that for the GGH scheme, it was suggested to encode the plaintext bits as the least significant bits of the input message to the trapdoor function and the other bits are allowed to be picked randomly. We now show how this will make our scheme vulnerable to the related message attack.

To this end, let $\mathbf{p}$ be an $(n - t)$ -bit plaintext to be encrypted. Suppose that $\mathbf{p}$ is encrypted twice, that is, encoded into $\mathbf{m}_1$ and $\mathbf{m}_2$ with $\mathbf{p}$ occupying the least significant bits of $\mathbf{m}_1$ and $\mathbf{m}_2$. Thus, $\mathbf{m}_1 + \mathbf{m}_2 \bmod 2 = \mathbf{0}$. This gives $\mathbf{c}_1 = \mathbf{m}_1 H + \mathbf{e}_1 \bmod q^{d_0} - 1$ and $\mathbf{c}_2 = \mathbf{m}_2 H + \mathbf{e}_2 \bmod q^{d_0} - 1$. Summing up, this yields $(\mathbf{m}_1 + \mathbf{m}_2)H + \mathbf{e}_1 + \mathbf{e}_2 \bmod q^{d_0} - 1 = \mathbf{c}_1 + \mathbf{c}_2$. If q is odd, we can consider the equation modulo 2 to get $0 \cdot H + \mathbf{e}_1 + \mathbf{e}_2 = \mathbf{c}_1 + \mathbf{c}_2 \bmod 2$ or $\mathbf{e}_1 + \mathbf{e}_2 \bmod 2 = \mathbf{c}_1 + \mathbf{c}_2$. If d is small relative to n , the number of entries which are 1 in both $\mathbf{e}_1$ and $\mathbf{e}_2$ will be very small. Hence, we can guess the positions in which $\mathbf{e}_1$ or $\mathbf{e}_2$ is 1 from the non-zero entries in $\mathbf{c}_1 + \mathbf{c}_2 \bmod 2$ and use the attacks in Section 4.4 to recover $\mathbf{m}$.

In view of the above, we modify the encoding scheme to work for our trapdoor function. Suppose that the parameters q, n, d, t are fixed. Our input to our trapdoor function is a vector in $\mathbb{Z}_{q^{d_0}-1}^{n-t}$. Thus, each entry is an s -bit string, where $s = 1 + \lfloor \log_2(q^{d_0} - 1) \rfloor$. We will encode an $(n - t)$ -bit plaintext message $\mathcal{P}$ into the input $\mathbf{m}$ for the trapdoor function f . The ciphertext will be the output of f in $\mathbb{Z}_{q^{d_0}-1}^n$. The entire encryption and decryption processes are described as follows.

Let $\mathbf{m} = (m_1, \dots, m_{n-t})$ be in $\mathbb{Z}_{q^{d_0}-1}^{n-t}$ and let $m_i^{(j)}$ denote the j -th least significant bit of m_i , $j = 0, 1, \dots, s - 1$. Further, let $\mathbf{m}^{(j)} = (m_1^{(j)}, \dots, m_{n-t}^{(j)})$. Suppose the plaintext message is $\mathcal{P} = (\mathcal{P}_1, \dots, \mathcal{P}_{n-t})$. In the following, let Hash denote a cryptographic hash function from $\{0, 1\}^*$ to $\{0, 1\}^{n-t}$. Let f be the trapdoor function with all the notations in Section 4.3.

Private key: The degree d polynomial $c(x)$, the n elements $\alpha_1, \dots, \alpha_n$, the unimodular

matrix T and the permutation matrix P .

Public key: The parameters n, q, d, t and the matrix $-G$.

Encryption:

- Randomly select $n - t$ bits $\mathbf{z} = (z_1, \dots, z_{n-t})$.
- Randomly select the error string $\mathbf{e} = (e_1, \dots, e_n)$ satisfying the desired properties.
- Set $\mathbf{m}^{(0)} = \mathcal{P} \oplus \mathbf{z}$.
- Set $\mathbf{m}^{(1)} = \mathbf{z}$.
- Set $\mathbf{m}^{(2)} = \text{Hash}(\mathcal{P} || \mathbf{z} || \mathbf{e})$.
- For $j = 3, \dots, s - 1$, set $\mathbf{m}^{(j)}$ randomly.
- Let $H = \begin{pmatrix} I_{n-t} & -G \end{pmatrix}$. Then, the ciphertext $\mathbf{c}$ is $\mathbf{c} = f(\mathbf{m}, \mathbf{e}) = \mathbf{m}H + \mathbf{e}$.

Decryption: Given a ciphertext $\mathbf{c}$, the decryption proceeds as follows:

- Compute $\mathbf{m} = f^{-1}(\mathbf{c})$ using the private key. Let $\mathbf{e}$ be the corresponding error. If $\mathbf{e}$ contains only 0 or 1 entries with exactly $d - 1$ 1's, then continue. Otherwise, decryption fails.
- Write $\mathbf{m} = (m_i^{(j)})_{i=1, \dots, n-t, j=0, 1, \dots, s-1}$.
- Set $\mathcal{P}' = \mathbf{m}^{(0)} \oplus \mathbf{m}^{(1)}$.
- If $\text{Hash}(\mathcal{P}' || \mathbf{m}^{(1)} || \mathbf{e}) = \mathbf{m}^{(2)}$, then $\mathcal{P} = \mathcal{P}'$ and the decryption is successful. Otherwise, decryption fails.

Theorem 4.5.1. *Assume the one-wayness of the trapdoor function, the above PKE scheme achieves IND-CPA security.*

Proof. The proof follows the security proof of conversion α of [KI01]. □

Remark 4.5.2. • *Like the GGH scheme [GGH97], we encode our plaintext bits in the least significant bits of the input to our trapdoor function.*

- *In our scheme, the input includes not only the plaintext bits but the error bits and random bits as well. By including the error bits to the input, changing bits of the ciphertext will likely make the decryption process fail. This helps to prevent reactive attacks where attackers try to guess the error bits by sending modified ciphertexts.*
- *Similar to the conversion schemes suggested in [KI01] for the McEliece encryption scheme, random bits and the hash of the plaintext bits are added to the input to ensure semantic security and to prevent other attacks such as related message attacks.*
- *As mentioned in Section 1.2, our encryption scheme can be transformed to achieve IND-CCA2 security in the QROM model.*

4.5.2 Choosing the Parameters

In view of the attacks presented in Section 4.4, we will choose the parameters q, n, d, t to resist all the possible attacks. Concretely, the following choices will be made.

- We let $t = d$, that is, $c(x)$ is a product of linear polynomials.
- We set q to be the smallest prime bigger than $n + d$.
- We set d to satisfy $20 \leq d \leq n/2$.
- For a security level λ , we set n and d so that $\binom{n-d}{l} \geq 2^\lambda$, where $l = \frac{(n-d)(d-1)}{n}$.

With $d = t$, we have $d_0 = 1$ so all our operations are done modulo $q - 1$. Our public key size is $(n - t)t(1 + \lceil \log_2(q - 2) \rceil)$ bits. Since encryption only involves matrix multiplication modulo $q - 1$, encryption is efficient with complexity $O(n^2)$.

We now provide possible sets of values of n and d for our encryption scheme. For each pair of n and d , we compute the smallest β such that Equation 4.2 holds for specified parameters, which gives the corresponds BKZ cost by Equation 2.1.

Tab. 4.1: Possible n and d for Practical Encryption Scheme

n	d	q	$\log_2 \binom{n-d}{l}$	β	$\log_2(BKZ_{cost})$	public key size	P_s
285	41	2819	138	180	80.1	120048bits= 14.66KB	0.289
500	43	29599	184	342	128.2	314416bits= 38.39KB	0.162
729	42	152003	208	518	180.1	519372bits= 63.40KB	0.289

Remark 4.5.3. *Note that for the parameters in Table 4.1, the discrete logarithm is computed over the field $\mathbb{F}_q$ rather than $\mathbb{F}_{q^d}$. Thus, Pollard rho method suffices to accomplish this task. As commented in [DP19], our construction can be viewed as an improvement of the Chor-Rivest scheme in terms of efficiency. The improvement mainly lies on the fact that the discrete logarithm computation is done in the small field $\mathbb{F}_q/(c_i(x))$ rather than in the big field $\mathbb{F}_q/(c(x))$. In particular, for the parameters in Table 4.1, we choose $\deg(c_i(x)) = 1$.*

Remark 4.5.4. *As mentioned in Remark 4.2.7, our construction is analogous to the Goppa code construction. One may use techniques to reduce the key size of Goppa codes in our construction as well. In particular, the techniques in [BBB⁺18b, LXY19] and in Chapter 3 based on automorphism groups can be adapted to reduce the public key size of our construction.*

5. ROUNDED SUBSET SUM

5.1 Introduction

In this chapter, we propose a new problem which we call the rounded subset sum (RSS) problem and present some of its applications. The hardness of the (RSS) problem hinges on the hardness of the subset sum (SS) problem. In particular, we prove the pseudorandomness of the rounded subset sum distribution under the assumption that the SS is hard. Based on the pseudorandomness of the RSS distribution, we present a public key cryptosystem that achieves IND-CPA security. In addition, we prove that the RSS problem remains hard in the presence of weak and leaky secrets. This enables us construct an adaptive leakage-resilient public key cryptosystem, thereby addressing an open problem proposed by Lyubashevsky, Palacio, and Segev [LPS10]. The construction is based on a weaker assumption subset sum as compared to the lattice-based adaptive leakage-resilient public key encryption scheme [DGK⁺10]. Finally, we present other applications of the RSS problem, including reusable extractors.

5.1.1 Comparisons With Lattice-based Schemes and Related Work

The LWE assumption is that for a secret vector $\mathbf{s} \in \mathbb{Z}_q^n$, a uniformly randomly sampled matrix $A \in \mathbb{Z}_q^{m \times n}$, and a short noise vector $\mathbf{e} \in \mathbb{Z}_q^m$, the distribution $(A, \mathbf{e} + A\mathbf{s})$ is pseudorandom. Lyubashevsky et al. observed that a subset sum instance can be converted to matrix-vector form, which is similar to the LWE problem. Following the idea of lattice-based public key encryption schemes, they constructed a public key cryptosystem directly built on subset sum. [LPS10] also constructed an oblivious transfer protocol based on the public key cryptosystem achieving

security against an semi-honest receiver and a malicious sender. Based on the observation of [LPS10], Faust et al. [FMV16] constructed a public key encryption which achieves IND-CCA security. Banerjee et al. [BPR12] proposed the Learning With Rounding(LWR) problem to construct pseudorandom functions from lattices. The LWR assumption is that for a vector $\mathbf{s} \in \mathbb{Z}_q^n$ and a uniformly randomly sampled matrix $A \in \mathbb{Z}_q^{m \times n}$, the distribution of $(A, \lceil A\mathbf{s} \rceil_p)$ is pseudorandom. The rounding technique in LWR is used to remove the random error in LWE. It is also used in fully homomorphic encryption to do the modulus switching [BV11] to reduce the decryption complexity.

Our rounded subset sum problem employs a similar form with the LWR problem. There are several differences of our rounded subset sum problem and LWR problem. The first difference is that the hardness of LWR problem is based on the hard lattice problem whereas the hardness of rounded subset sum is based on a weaker assumption i.e., the subset sum problem. The second difference is the motivation. For LWR problem, the rounding operation is introduced to derandomize the error terms in LWE problem, and thus generate deterministic error terms. For subset sum problem, when viewed as matrix-vector representation, the error term is inherently controlled by the secret vector $\mathbf{s}$ and the public matrix A . The rounding operation introduced to subset sum problem is to infuse more flexibility to the subset sum problem, therefore, leads to more interesting applications directly from SS/RSS.

5.1.2 Organization

The subsequent sections are structured as follows. In Section 5.2, we present the rounded subset sum problem and show that the rounded subset sum distribution is pseudorandom built from the hardness of subset sum problem. A public key cryptosystem is presented in Section 5.3. The semantic security of the cryptosystem is built on the rounded subset sum problem. In Section 5.4 and 5.5, we prove that the Rounded Subset Sum problem with weak and leaky secrets is still hard, and show that the scheme in Section 5.3(with different parameters) is leakage-resilient

secure in adaptive scenario. At last, in Section 5.6, we construct a reusable extractor from the rounded subset sum problem.

5.2 Rounded Subset Sum

In this section, we define the *rounded subset sum* problem and show that the rounded subset sum distribution is pseudorandom under the assumption that the corresponding subset sum problem is hard.

Before defining the rounded subset sum problem, we first define the *rounding* function used in the rounded subset sum. Let p, q be integers satisfying $p < q$. The rounding function $\lceil \cdot \rceil_p : \mathbb{Z}_q \rightarrow \mathbb{Z}_p$ is defined as $\lceil x \rceil_p = \lceil \bar{x} \cdot p/q \rceil \pmod p$, where $\bar{x} \in \mathbb{Z}$ equals to x modulo q . We generalize the rounding function component-wise to vectors.

The *rounded subset sum* problem is characterized by positive integers m, n, p, q .

Definition 5.2.1 (Rounded Subset Sum Problem). *Define n as the main security parameter. Let $m = \text{poly}(n)$, $p = \text{poly}(n)$, $q = \text{poly}(n)$, $q > 2pmn^2\sqrt{2\pi m}$. Let q be prime. Given a target vector $\mathbf{b} \in \mathbb{Z}_p^m$ and a matrix $A = (\mathbf{a}_1, \dots, \mathbf{a}_n) \in \mathbb{Z}_q^{m \times n}$, the rounded subset sum problem is to seek out a subset $S \subseteq \{1, \dots, n\}$ satisfying $\lceil \sum_{i \in S} \mathbf{a}_i \pmod q \rceil_p = \mathbf{b}$.*

The rounded subset sum problem can be considered as the problem of finding the preimage of the function $g_A : \{0, 1\}^n \rightarrow \mathbb{Z}_p^m$ defined as $g_A(\mathbf{s}) = \lceil A\mathbf{s} \pmod q \rceil_p = \lceil \sum_{i=1}^n s_i \mathbf{a}_i \pmod q \rceil_p$, where $\mathbf{a}_i$ is defined as the i -th column of A . For a vector $\mathbf{s} \in \{0, 1\}^n$, the *rounded subset sum distribution* $A_{\mathbf{s}, n, m, q, p}$ over $\mathbb{Z}_q^{m \times n} \times \mathbb{Z}_p^m$ is defined as: draw $A \in \mathbb{Z}_q^{m \times n}$ uniformly at random, and output $(A, \lceil A\mathbf{s} \pmod q \rceil_p)$.

There are two versions of the rounded subset sum problem: the *search* version(search-RSS), which is to search for the vector $\mathbf{s}$ given $(A, \lceil A\mathbf{s} \pmod q \rceil_p)$ and the *decision* version(decision-RSS), which is to tell apart the rounded subset sum distribution from the uniform distribution over the same domain. We denote the rounded subset sum problem characterized by parameters (n, m, q, p) as $\text{RSS}(n, m, q, p)$.

Note that in concrete applications, we choose the parameter such that $\log^2 n < m \log q < n^2$ because only under this parameter setting, the subset sum problem is believed to be hard. In addition, under this condition, the pseudorandomness of RSS distribution makes sense. Without the rounding operation, given $m > n$, the problem is easy to solve for a uniformly randomly sampled A with non-negligible probability via Gaussian elimination. With the rounding operation and the subset sum assumption, the rounded subset sum problem is hard, and the rounded subset sum distribution is pseudorandom as well. Intuitively, this implies that the rounding operation provides more choices for the value m and introduces more flexibility to the subset sum problem.

We prove that for appropriate parameters, the *decision*-RSS problem is hard if the subset sum problem is hard for consistent parameters. In other words, if the subset sum function is one-way, the rounded subset sum distribution is pseudorandom.

Theorem 5.2.2. *Define n as the main security parameter and $m = \text{poly}(n)$, $p = \text{poly}(n)$, $q = \text{poly}(n)$ and $p < q$. If the subset-sum function is one-way, the rounded subset sum distribution is pseudorandom.*

In particular, for any secret $s \in \{0, 1\}^n$, solving $\text{decision-RSS}_{n,m,q,p}$ is as hard as inverting $\text{SS}(n, q^m)$.

According to our main theorem, if the parameters (n, m, q) are appropriately chosen, namely, choosing parameters such that $\text{SS}(n, q^m)$ is hard and the condition in Theorem 5.2.2 is satisfied, the rounded subset sum distribution $A_{s,n,m,q,p}$ is pseudorandom. Note that in Theorem 5.2.2, our reduction does not cause any parameter loss, namely, the subset sum problem and the rounded subset sum problem share the same parameters (n, m, q) .

Building the reduction in Theorem 5.2.2 is complicated. The proof of Theorem 5.2.2 consists of two steps. In step one, we show that the one-wayness of the subset sum function implies the one-wayness of the rounded subset sum function. This step uses carries of subset sum problem to convert subset sum instance to the matrix-vector representation, uses noise flooding technique

to decouple the dependence of noise vector on the matrix and the secret vector in matrix-vector representation, and uses Renyi-Divergence to argue that the algorithm used to solve RSS can be used to solve the flooded matrix-vector representation of subset sum instance. In step two, we prove that the one-wayness of the rounded subset sum function implies the pseudorandomness of the rounded subset sum distribution. This step follows a sequence of standard hybrid games stemmed from Yao [Yao82] and a variant of Goldreich-Levin theorem over a large field from Dodis et al. [DGK⁺10]. Theorem 5.2.3 and Theorem 5.2.9 are devoted to each step of the reduction.

Proof of Theorem 5.2.2. The parameter conditions of Theorem 5.2.3 and Theorem 5.2.9 are both satisfied. Thus, a direct combination of the two theorems yields a proof of the result without parameter loss. $\square$

5.2.1 One-wayness of Rounded Subset Sum

In this section, we prove the one-wayness of the rounded subset sum function assuming the one-wayness of the subset sum function.

Theorem 5.2.3. *Define n, m, q, p as integers satisfying $q > 2pmn^2\sqrt{2\pi m}$. If the subset sum function is one-way, the rounded subset sum function is one-way.*

In particular, if there is a PPT algorithm $\mathcal{A}$ inverting $\text{RSS}(n, m, q, p)$ with running time t , there is a PPT algorithm inverting $\text{SS}(n, q^m)$ with running time $\text{poly}(t, n, m, p, q)$.

To prove Theorem 5.2.3, we first convert an instance $(\mathbf{a}, b = \langle \mathbf{a}, \mathbf{s} \rangle)$ of the subset sum problem to an instance of the matrix-vector form $(A, \mathbf{b} = A \odot \mathbf{s} = A\mathbf{s} + \mathbf{f} \pmod{q})$ with the vector $\mathbf{f}$ being bounded, not independent over all of the coordinates and relying on A and $\mathbf{s}$. Next, we use the *noise flooding* technique to decouple the dependence to get the instance $(A, \mathbf{c} = \mathbf{b} + \mathbf{e} = A\mathbf{s} + \mathbf{f} + \mathbf{e})$, where $\mathbf{f} + \mathbf{e}$ is close to a distribution independent from A and $\mathbf{s}$. Then combining with the algorithm that solves the rounded subset sum problem, we get the desired solution of the subset sum instance.

We recall some lemmas necessary to prove Theorem 5.2.3. The first lemma is adapted from [LPS10, Lemma 3.3]. They presented a similar result highlighting that each entry of $A \odot \mathbf{s} - A\mathbf{s}$ is bounded. Here we provide a version of their result with better parameter choices and stress that each entry of $A \odot \mathbf{s} - A\mathbf{s}$ relies on A and $\mathbf{s}$.

Lemma 5.2.4 (Adapted from [LPS10]). *Given n, m, q being integers satisfying $2n + 1 < q$ and $\mathbf{s} \in \{0, 1\}^n$ being a nonzero vector. Then,*

$$\Pr_{A \leftarrow \mathbb{Z}_q^{m \times n}} [\|A \odot \mathbf{s} - A\mathbf{s}\|_\infty \leq n] \geq 1 - 2^{-\Omega(n)}.$$

Define $\mathbf{f} := A \odot \mathbf{s} - A\mathbf{s}$. Let $\mathbf{a}_i^T$ be the i -th row of A . Moreover, the vector $\mathbf{f}$ can be considered as the sum of two vectors $\mathbf{u}, \mathbf{v} \in \mathbb{Z}_q^m$ where each entry u_i of $\mathbf{u}$ is independently distributed with mean 0 and depends on $\langle \mathbf{a}_{i-1}, \mathbf{s} \rangle$, and each entry v_i of $\mathbf{v}$ has absolute value 1.

The proof follows the proof of Lemma 3.3 in Lyubashevsky et al. [LPS10]. In the proof of this lemma, we do the computation of $A \odot \mathbf{s}$ in the entry by entry method in q -ary, which makes it possible for us to split $A \odot \mathbf{s} - A\mathbf{s}$ to the carry vector and the real noise vector. For completeness, we include the proof here.

Proof of Lemma 5.2.4. From Equation 2.7, the matrix-vector equation of subset sum can be converted to the following form

$$\sum_{i=0}^{m-1} b_i q^i = \sum_{i=0}^{m-1} q^i \sum_{j=0}^{n-1} A_{ij} s_j \pmod{q^m}.$$

Let $\mathbf{a}_i^T$ be the i -th row vector of A . Thus, we can iteratively compute each b_i . In each iteration, the sum is computed according to the inner product of the current iteration and the carry from the last iteration. We now show an algorithm to compute vector $\mathbf{b}$.

1. Let $d_0 = 0$

2. For i from 0 to $m - 1$
3. $d_{i+1} = \lceil (d_i + \langle \mathbf{a}_i, \mathbf{s} \rangle) / q \rceil$
4. $b_i = d_i + \langle \mathbf{a}_i, \mathbf{s} \rangle \pmod{q}$
5. Output the vector $\mathbf{b} = (b_0 \dots b_{m-1})$

Note that in the above computation procedure, each carry d_{i+1} relies on the previous inner product $\langle \mathbf{a}_i, \mathbf{s} \rangle$ and the previous carry d_i . We can rewrite d_i as $u_i + v_i$ such that all the u_i are independent and bounded with high probability, and the absolute value of each v_i is bounded by 1.

1. Let $u_0 = v_0 = 0$
2. For i from 0 to $m - 1$
3. $b_i = (u_i + v_i + \langle \mathbf{a}_i, \mathbf{s} \rangle) \pmod{q}$
4. $u_{i+1} = \lceil \langle \mathbf{a}_i, \mathbf{s} \rangle \rceil$
5. $v_{i+1} = \lceil (x_i + y_i + \langle \mathbf{a}_i, \mathbf{s} \rangle) / q \rceil - \lceil \langle \mathbf{a}_i, \mathbf{s} \rangle / q \rceil$
6. Output $\mathbf{b} = (b_0, \dots, b_{m-1})$

Note that in this kind of computation, u_i does not rely on previous computed values of vector $\mathbf{u}$ and only relies on the previous inner product. We now bound the value of u_i . Every row of A is drawn from a uniform distribution defined over $\mathbb{Z}_q$. By applying the Hoeffding bound to $\langle \mathbf{a}_i, \mathbf{s} \rangle$, we get that

$$Pr_{\mathbf{a}_i} [|\langle \mathbf{a}_i, \mathbf{s} \rangle| \geq qn] \leq 2e^{-2n}.$$

Thus, with probability at least $1 - 2e^{-2n}$, the absolute value of each entry $|u_i|$ is bounded by n . We prove that $|v_i| \leq 1$ by induction. It is true for $v_0 = 0$. Assume that $|v_i| \leq 1$. Then,

$$|v_{i+1}| = |\lceil (x_i + y_i + \langle \mathbf{a}_i, \mathbf{s} \rangle) / q \rceil - \lceil \langle \mathbf{a}_i, \mathbf{s} \rangle / q \rceil| \leq |\lceil (x_i + y_i) / q \rceil + 1| \leq 1.$$

From $n < q/2 - 1$ and $|y_i| \leq 1$, then $|x_i + y_i| \leq q/2$. The last inequality follows. Therefore, $|x_i + y_i| \leq n$ with high probability. $\square$

Given the Lemma 5.2.4, in the $A \odot \mathbf{s} = A\mathbf{s} + \mathbf{f}$ form representation of the subset sum instance, each entry of f_i is bounded but relies on the previous inner product $\langle \mathbf{a}_{i-1}, \mathbf{s} \rangle$. We would like to decouple the dependency relations. To make every entry f_i statistically independent from A and $\mathbf{s}$, we borrow the *noise flooding* [GKPV10] technique from lattice-based cryptography to remove the dependence. The approach of noise flooding says that for a bounded vector $\mathbf{f}$ and a wide enough Gaussian distributed vector $\mathbf{e}$, the value $\mathbf{f} + \mathbf{e}$ is essentially independent of $\mathbf{f}$, namely, the vector $\mathbf{e}$ *floods* the bounded value of $\mathbf{f}$.

The second lemma is the noise flooding result from Bogdanov et al. [BGM⁺16, Corollar 16].

Lemma 5.2.5. *Let n, m, q be parameters. Let a standard deviation σ and a bound B satisfy $B < \sigma < q$. Denote D_σ as a discrete Normal distribution over $\mathbb{Z}_q$ and σ as the standard deviation. Given $e \in \mathbb{Z}_q$ and $|e| \leq B$, we obtain*

$$\text{RD}_2((e + D_\sigma)^m || D_\sigma^m) \leq \exp\left(2\pi m \left(\frac{B}{\sigma}\right)^2\right).$$

The lemma says if the standard deviation of the Gaussian distribution satisfies the lower bound condition, then the flooded distribution is close to the Gaussian distribution. Note that the standard deviation σ and q can both be $\text{poly}(n)$ in the above lemma. If $\sigma = \Omega(B\sqrt{m/\log n})$, then the RD_2 is $\text{poly}(n)$.

After noise flooding, the subset sum instance $(\mathbf{a}, f_{\mathbf{a}}(\mathbf{s}))$ is converted to an instance of the form $(A, \mathbf{b} = A\mathbf{s} + \mathbf{f} + \mathbf{e})$. Next we prove a lemma which states the one-wayness of $(A, A\mathbf{s} + \mathbf{f} + \mathbf{e})$

implies the one-wayness of $(A, As + \mathbf{e})$.

Lemma 5.2.6. *Let n, q, m be integers satisfying $q > n\sqrt{2\pi m}$. If there is a PPT algorithm inverts $(A, As + \mathbf{e})$ with advantage ϵ , then there is a PPT algorithm inverts $(A, As + \mathbf{f} + \mathbf{e})$ with advantage $\epsilon^2/e - 2^{-\Omega(n)}$, where $A \leftarrow \mathbb{Z}_q^{m \times n}$, $\mathbf{s} \leftarrow \{0, 1\}^n$.*

Proof. From Lemma 5.2.5, we have $\text{RD}_2((A, As + \mathbf{f} + \mathbf{e}) || (A, As + \mathbf{e})) \leq \exp\left(2\pi m \left(\frac{B}{\sigma}\right)^2\right)$. Let $\mathcal{A}$ be a PPT algorithm. Let E be the event $\{\mathcal{A}(A, \mathbf{z}) = \mathbf{s} : (A, \mathbf{z})\}$. From Lemma 2.2.8, we have

$$\Pr_{A, \mathbf{s}, \mathbf{e}}[\mathcal{A}(A, As + \mathbf{e})] \geq \frac{\Pr_{A, \mathbf{s}, \mathbf{e}}[\mathcal{A}(A, As + \mathbf{e})]}{\text{RD}_2((A, As + \mathbf{f} + \mathbf{e}) || (A, As + \mathbf{e}))} \geq \frac{\Pr_{A, \mathbf{s}, \mathbf{e}}[\mathcal{A}(A, As + \mathbf{e})]}{\exp\left(2\pi m \left(\frac{B}{\sigma}\right)^2\right)}.$$

From Lemma 5.2.4, each entry of $\mathbf{f}$ is bounded by n with overwhelming probability $1 - 2^{-\Omega(n)}$. Taking $\sigma \geq n\sqrt{2\pi m}$, then the PPT algorithm $\mathcal{A}$ inverts $(A, As + \mathbf{f} + \mathbf{e})$ with advantage at least $\epsilon^2/e - 2^{-\Omega(n)}$. $\square$

The instance $(A, As + \mathbf{e})$ is similar to the matrix version of the LWE problem. Thus we name the instance as *subset sum with independent noise* (SSWIN). SSWIN and LWE are different in several aspects. The main difference of SSWIN and LWE is that LWE is based on the worst-case hard problem in n -dimensional lattices while SSWIN relies on subset sum problems. The subset sum problem is a weaker assumption.

Given the third lemma, the vector $\mathbf{f} + \mathbf{e}$ can be viewed as being independently sampled from a discrete Normal distribution over all coordinates. Bogdanov et al. [BGM⁺16] proved that given an algorithm solving the LWR problem, the algorithm can also be employed to work out the secret vector of the LWE problem. As our rounded subset sum problem is similar to the learning with rounding problem and the SSWIN problem is similar to the learning with error problem, the algorithm for the RSS problem can be employed to deal with the SSWIN problem.

We recall the hardness result for LWR from [BGM⁺16]. It says that the one-wayness of LWE function implies the one-wayness of LWR function. This result can be used in our result

as well, namely that the one-wayness of subset sum problem with independent noise implies the one-wayness of rounded subset sum function.

Lemma 5.2.7. *[BGM⁺16, Theorem 1] Define p, q, m, B as integers satisfying $q > 2Bpm$. If there exists a PPT algorithm solving the rounded subset sum function, there exists a PPT algorithm solving the subset sum with independent noise.*

Moreover, if there is a PPT algorithm $\mathcal{A}$, given input $(A, \lceil As \rceil_p)$, outputs $\mathbf{s}$ with advantage ϵ , then the algorithm $\mathcal{A}$, given input $(A, \lceil As + \mathbf{e} \rceil_p)$ can be used to output $\mathbf{s}$ with advantage $\frac{\epsilon^2}{(1+2pB/q)^m}$, i.e.,

$$Pr_{A,s,e} [\mathcal{A}(A, \lceil As + \mathbf{e} \rceil_p) = \mathbf{s}] \geq \frac{Pr_{A,s} [\mathcal{A}(A, \lceil As \rceil_p) = \mathbf{s}]^2}{(1 + 2pB/q)^m} \geq Pr_{A,s} [\mathcal{A}(A, \lceil As \rceil_p) = \mathbf{s}]^2 / e$$

where $A \leftarrow \mathbb{Z}_q^{m \times n}$, $\mathbf{s}$ is drawn from any distribution defined over $\mathbb{Z}_q^{n}$, and every coordinate of the $\mathbf{e}$ is independent from others and less than a bound B .*

In the above lemma, the success probability does not deteriorate too much when the algorithm for LWR is used to solve LWE. In our case, the secret vector $\mathbf{s}$ is from $\{0, 1\}^n$ so it satisfies the condition $\mathbf{s} \in \mathbb{Z}_q^{n*}$. Thus, in our case, the probability of $\mathcal{A}$ succeeding does not weaken too much when the RSS instance is replaced by the SSWIN instance. According to the lemma, if the SSWIN problem is hard, then the RSS problem is hard.

We now combine the previous lemmas with other results to state it as a reduction proof from Subset Sum problem to Rounded Subset Sum problem. Specifically, we demonstrate the proof for Theorem 5.2.3.

Proof of Theorem 5.2.3. Given subset sum $\text{SS}(n, q^m)$ instance $(\mathbf{a}, b)$ and an algorithm $\mathcal{A}$ solving the rounded subset sum instance $\text{RSS}(n, m, q, p)$, the reduction can be obtained in the following steps.

- Apply the matrix-vector representation to obtain subset sum with bounded noise, i.e.,

$$\mathbf{b} = A\mathbf{s} + \mathbf{f}.$$

- Apply noise-flooding to rerandomize $\mathbf{f}$ to obtain an instance of the subset sum with independent noise, i.e., $\mathbf{c} = \mathbf{b} + \mathbf{e} = A\mathbf{s} + \mathbf{f} + \mathbf{e}$.
- Feed the rounded subset sum with independent noise instance to the algorithm $\mathcal{A}$, i.e., $\mathcal{A}(A, \lceil \mathbf{c} \rceil_p)$, to get a solution to the subset sum instance $(\mathbf{a}, b)$.

According to Lemma 5.2.6, in the second step, we set the standard deviation σ of vector $\mathbf{e}$ to be $\sigma \geq n\sqrt{2\pi m}$. From Lemma 2.2.5, each entry of $\mathbf{e}$ is less than $n\sigma = n^2\sqrt{2\pi m}$ with overwhelming probability $1 - 2^{-\Omega(n)}$.

From the second step, we have

$$Pr_{A,s}[\mathcal{A}(A, A\mathbf{s} + \mathbf{f}) = \mathbf{s}] \geq Pr_{A,s,e}[\mathcal{A}(A, A\mathbf{s} + \mathbf{f} + \mathbf{e}) = \mathbf{s}].$$

From Lemma 5.2.6, we have

$$Pr_{A,s,e}[\mathcal{A}(A, A\mathbf{s} + \mathbf{f} + \mathbf{e}) = \mathbf{s}] \geq \frac{Pr_{A,s,e}[\mathcal{A}(A, A\mathbf{s} + \mathbf{e}) = \mathbf{s}]^2}{e}.$$

The condition $q > 2pmB = 2pmn^2\sqrt{2\pi m}$ in Lemma 5.2.7 is satisfied. From Lemma 5.2.7, we have

$$Pr_{A,s,e}[\mathcal{A}(A, \lceil A\mathbf{s} + \mathbf{e} \rceil_p) = \mathbf{s}] \geq \frac{Pr_{A,s}[\mathcal{A}(A, \lceil A\mathbf{s} \rceil) = \mathbf{s}]^2}{(1 + 2pn^2\sqrt{2\pi m}/q)^m} \geq Pr_{A,s}[\mathcal{A}(A, \lceil A\mathbf{s} \rceil) = \mathbf{s}]^2 / e.$$

Thus, we have $Pr_{A,s}[\mathcal{A}(A, A\mathbf{s} + \mathbf{f}) = \mathbf{s}] \geq Pr_{A,s}[\mathcal{A}(A, \lceil A\mathbf{s} \rceil) = \mathbf{s}]^4 / e^3$. $\square$

5.2.2 Pseudorandomness of Rounded Subset Sum

In this section, we prove the pseudorandomness of rounded subset sum distribution. Bogdanov et al. [BGM⁺16] provides a similar results in the case of the learning with rounding problem.

However, their proof deeply relies on the learning heavy Fourier coefficients of a function, so the proof is quite involved. Here we show a simple proof building on hybrid games stemmed from Yao [Yao82], Goldreich and Levin [GL89]. Our proof uses a Goldreich-Levin type theorem over a prime field $\mathbb{F}_q$. The original Goldreich-Levin theorem says that for arbitrary one-way function f , given $f(\mathbf{s})$, and $\mathbf{c}$ uniformly sampled from $\mathbb{F}_2^n$, the inner product $\langle \mathbf{s}, \mathbf{c} \rangle$ distribution is pseudorandom. Dodis et al. [DGK⁺10] extend the Goldreich-Levin theorem and show the following variant of the Goldreich-Levin theorem over a large field $\mathbb{F}_q$ instead of $\mathbb{F}_2$. Thus, all the results in this section build under the condition that q is prime.

Suppose that the rounded subset sum problem is hard. We show that rounded subset sum distribution is pseudorandom. Namely, we present a reduction from search-RSS to decision-RSS.

Lemma 5.2.8. *[LPS10, Theorem 4.1] Denote q as a prime number, and denote H as any subset of $\mathbb{F}_q$. Denotes $f : H^n \rightarrow \{0, 1\}^*$ as (randomized) function. Let $\mathbf{c}$ be uniformly at random sampled over $\mathbb{F}_q^n$ and $\mathbf{s}$ be sampled from a distribution $\mathcal{D}$ over H^n . If f is one-way, then $(\mathbf{c}, \langle \mathbf{c}, \mathbf{s} \rangle)$ is pseudorandom.*

In particular, if there is a PPT algorithm $\mathcal{A}$, given $f(\mathbf{s})$, that distinguishes $(\mathbf{c}, \langle \mathbf{c}, \mathbf{s} \rangle)$ from uniform distribution over $\mathbb{F}_q^{n+1}$ with running time t and advantage ϵ , there is a PPT algorithm $\mathcal{B}$ that finds the preimage of $f(\mathbf{s})$ with running time $t' = t \cdot \text{poly}(n, |H|, 1/\epsilon)$ and probability $\frac{\epsilon^3}{512nq^2}$.

Based on the previous lemma and other results, we show the pseudorandomness of rounded subset sum distribution.

Theorem 5.2.9. *Let n be integer, $m = \text{poly}(n)$, $p = \text{poly}(n)$, $q = \text{poly}(n)$, and q is prime. If the function of subset sum with independent error vector is one-way, then the rounded subset sum distribution is pseudorandom.*

Moreover, if there exists a PPT time distinguisher $\mathcal{D}$ that tells apart $(A, [A\mathbf{s}]_p)$ from uniform distribution $(A, [\mathbf{u}]_p)$ with advantage ϵ and running time t , where A is uniformly at random sampled from $\mathbb{Z}_q^{m \times n}$ and $\mathbf{u}$ is uniformly at random sampled from $\mathbb{Z}_q^m$, then there is an algorithm

$\mathcal{A}$ that inverts the function $f(\mathbf{s}) = \lceil A\mathbf{s} \rceil_p$ with advantage at least $\frac{\epsilon^3}{512nm^3q^5}$ and running time $t' = \text{poly}(n, m, q, t, 1/\epsilon)$.

In the proof of theorem Theorem 5.2.9, we convert the distinguisher $\mathcal{D}$ to a predictor $\mathcal{P}$ for $\langle \mathbf{s}, \mathbf{r} \rangle$ with $\mathbf{r}$ drawn from uniform distribution. Next we convert the predictor $\mathcal{P}$ to a distinguisher $\mathcal{D}'$ for $\langle \mathbf{s}, \mathbf{r} \rangle$. In the last, we use Lemma 5.2.8 to invert the function f . The construction of $\mathcal{P}$ and $\mathcal{D}'$ follow a standard hybrid game from [Yao82, GL89]. Details follow.

Proof. We first present a prediction algorithm $\mathcal{P}$.

Given input $(A, \mathbf{b}) = ((\mathbf{a}_1, b_1), \dots, (\mathbf{a}_m, b_m)) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_p^m$ and $\mathbf{a}$ uniformly drawn from $\mathbb{Z}_q^m$,

1. Select i uniformly in $[m]$ and a guess g uniformly in $\mathbb{Z}_q$ for $\langle \mathbf{s}, \mathbf{a} \rangle$.
2. Select $u_{i+1}, \dots, u_m$ uniformly in $\mathbb{Z}_p$.
3. Obtain $(\bar{A}, \bar{\mathbf{b}})$ from $(A, \mathbf{b})$ by assigning $(\bar{\mathbf{a}}_i = \mathbf{a}, \bar{b}_i = \lceil g \rceil_p)$ and $\bar{b}_j = \lceil u_j \rceil_p$ for $i+1 \leq j \leq m$.
4. Output g if $\mathcal{D}(\bar{A}, \bar{\mathbf{b}}) = 1$. Sample an element uniformly at random in $\mathbb{F}_q$ and output it otherwise.

Define $H_i := (\lceil \langle \mathbf{a}_1, \mathbf{s} \rangle \rceil, \dots, \lceil \langle \mathbf{a}_i, \mathbf{s} \rangle \rceil, \lceil u_{i+1} \rceil, \dots, \lceil u_m \rceil) \in \mathbb{Z}_p^m$. Thus, $H_0 = \lceil A\mathbf{s} \rceil_p$ and $H_m = \lceil \mathbf{u} \rceil_p$. In the algorithm $\mathcal{P}$, if the guess g is correct, then the input fed to $\mathcal{D}$ is H_i . Otherwise, the input to $\mathcal{D}$ is H_{i-1} . From the assumption on $\mathcal{D}$, we have that

$$\Pr \left[\mathcal{D} \left(A, \lceil A\mathbf{s} \rceil_p \right) = 1 \right] - \Pr \left[\mathcal{D} \left(A, \lceil \mathbf{u} \rceil_p \right) = 1 \right] \geq \epsilon.$$

Averaging over i , we have

$$\Pr \left[\mathcal{D} \left(\bar{A}, \lceil H_i \rceil \right) = 1 \right] - \Pr \left[\mathcal{D} \left(\bar{A}, \lceil H_{i-1} \rceil \right) = 1 \right] \geq \epsilon/m.$$

If the guess g is correct, then $\mathcal{P}$ outputs the correct guess with probability ϵ/m . If $\mathcal{D}$ outputs 0, the output is correct with probability $1/q$. Thus, the algorithm $\mathcal{P}$ correctly guesses the inner

product $\langle \mathbf{a}, \mathbf{s} \rangle$ with probability

$$\begin{aligned} \Pr[\mathcal{P}(A, \mathbf{b}, \mathbf{a}) = \langle \mathbf{a}, \mathbf{s} \rangle] &\geq \Pr[\mathcal{D}(\bar{A}, \bar{\mathbf{b}}) = 1 \wedge g = \langle \mathbf{a}, \mathbf{s} \rangle] + \frac{1}{q} \Pr[\mathcal{D}(\bar{A}, \bar{\mathbf{b}}) \neq 1] \\ &= \frac{1}{q} + \frac{1}{q} (\Pr[\mathcal{D}(\bar{A}, H_i) = 1] - \Pr[\mathcal{D}(\bar{A}, H_{i-1})]) \geq \frac{1}{q} + \frac{\epsilon}{qm}. \end{aligned}$$

The above computation shows that $\mathcal{P}$ predicts $\langle \mathbf{a}, \mathbf{s} \rangle$ with advantage $\frac{1}{q} + \frac{\epsilon}{qm}$ and running time $\text{poly}(n, m, q, t)$.

Next we convert the predictor $\mathcal{P}$ of $\langle \mathbf{a}, \mathbf{s} \rangle$ to a distinguisher $\mathcal{D}'$ for $(\mathbf{a}, \langle \mathbf{a}, \mathbf{s} \rangle)$. Given the input $(A, \mathbf{b}, \mathbf{a}, b)$, $\mathcal{D}'$ runs $\mathcal{P}(A, \mathbf{b}, \mathbf{a})$ and outputs 1 if the output of $\mathcal{P}$ equals b . Otherwise outputs 0. If b is equal to $\langle \mathbf{a}, \mathbf{s} \rangle$, then $\mathcal{D}'$ outputs 1 with probability $\frac{1}{q} + \frac{\epsilon}{qm}$ because it only happens when $\mathcal{P}$ outputs b . If b is sampled from uniform distribution, then $\mathcal{D}'$ outputs 1 only when b exactly equals $\langle \mathbf{a}, \mathbf{s} \rangle$ and $\mathcal{P}$ outputs b . Then for uniformly sampled b , $\mathcal{D}'$ produces 1 with probability $\leq 1/q$. Then the advantage of $\mathcal{D}'$ distinguishes $(\mathbf{a}, \langle \mathbf{a}, \mathbf{s} \rangle)$ from uniform distribution is at least $\frac{\epsilon}{qm}$. The running time of $\mathcal{D}'$ is also $\text{poly}(n, m, q, t)$.

Next, with the distinguisher $\mathcal{D}'$, we can call the Lemma 5.2.8 to get an inverter for the rounded subset sum problem with advantage $\frac{\epsilon^3}{512nm^3q^5}$ and running time $\text{poly}(n, m, q, t, 1/\epsilon)$. $\square$

5.3 Public Key Cryptosystem Based on Rounded Subset Sum

In this section, we demonstrate a public key cryptosystem which is directly built from the hardness of our rounded subset sum problem. The cryptosystem is parameterized by $n, q = \text{poly}(n), p = \text{poly}(n)$ satisfying $q > 2pn^3\sqrt{2\pi n}$ and q is prime.

- **KeyGen**: Sample a matrix $A \in \mathbb{Z}_q^{n \times n}$ and the secret vector $\mathbf{s} \in \{0, 1\}^n$ uniformly at random. Let $\mathbf{b} = \lceil A\mathbf{s} \rceil_p$. The public key is $(A, \mathbf{b})$.
- **Enc** $(A, \mathbf{b}, \mu)$: To encrypt one bit $\mu \in \{0, 1\}$. Draw $\mathbf{r} \in \{0, 1\}^n$ uniformly and then let $\mathbf{c}_1 = \lceil A^T \mathbf{r} \rceil_p, c_2 = \langle \mathbf{b}, \mathbf{r} \rangle + \mu \cdot \lceil p/2 \rceil$. Output the ciphertext pair $(\mathbf{c}_1, c_2) \in \mathbb{Z}_p^m \times \mathbb{Z}_p$.
- **Dec** $(\mathbf{s}, (\mathbf{c}_1, c_2))$: Compute $y = c_2 - \langle \mathbf{c}_1, \mathbf{s} \rangle$. Output 0 if $|y| \leq q/4$ and otherwise output 0.

The above public key cryptosystem is semantically secure against passive eavesdroppers, assuming the hardness of subset sum problems. Next we prove the security and correctness of the above cryptosystem.

Theorem 5.3.1 (Correctness). *Let n, p, q satisfy the conditions that $q > 2pn^3\sqrt{2\pi n}$ and q is prime. Then the Dec outputs the correct message.*

Proof. Following the definition of the $\lceil \cdot \rceil_p$ operation, the key generation operation can be written as $\mathbf{b} = \frac{p}{q}A\mathbf{s} + \mathbf{e}_1$ with $\|\mathbf{e}_1\|_\infty < 1/2$. The ciphertext $\mathbf{c}_1$ can be written as $\mathbf{c}_1 = \frac{p}{q}A^T\mathbf{r} + \mathbf{e}_2$ with $\|\mathbf{e}_2\|_\infty < 1/2$. The decryption algorithm computes $c_2 - \langle \mathbf{c}_1, \mathbf{s} \rangle = \langle \mathbf{e}_1, \mathbf{r} \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle + \mu \lceil p/2 \rceil$. By Cauchy-Schwarz inequality, $\langle \mathbf{e}_1, \mathbf{r} \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle$ is bounded by $(m+n)/2$, which is less than $p/4$. So if $\mu = 0$, we will have

$$|c_2 - \langle \mathbf{c}_1, \mathbf{s} \rangle| = |\langle \mathbf{e}_1, \mathbf{r} \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle| < p/4,$$

and we will decrypt 0. If $\mu = 1$, we will have

$$|c_2 - \langle \mathbf{c}_1, \mathbf{s} \rangle| = p/2 - |\langle \mathbf{e}_1, \mathbf{r} \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle| > p/4,$$

and we will decrypt to 1. □

Theorem 5.3.2 (Security). *Let n, p, q satisfy the conditions that $q > 2pn^3\sqrt{2\pi n}$ and q is prime. Assume that $\text{SS}(n, q^{n+1})$ is hard. The above public key cryptosystem achieves semantic security.*

Proof. According to Lemma 2.5.2 and Theorem 5.2.9, if $\text{SS}(n, q^{n+1})$ is hard, then $\text{RSS}(n, q^{n+1})$ is pseudorandom and $\text{RSS}(n, q^n)$ is pseudorandom. We will prove that for a ciphertext $(\mathbf{c}_1, c_2)$ of either $\mu = 0$ or 1, the whole view

$$(A, \mathbf{b}, \mathbf{c}_1, c_2) = \left(A, \mathbf{b}, \lceil A^T \mathbf{r} \rceil_p, \langle \mathbf{b}, \mathbf{r} \rangle + \mu \cdot \lceil p/2 \rceil \right)$$

of the adversary is computational indistinguishable from each other. As a matter of fact, we provide even a stronger result, namely, the view of the adversary is computational indistinguish-

able from uniform distribution over the respective domain. The hardness of $\text{RSS}(n, q^n)$ implies that $(A, \mathbf{b})$ is computational indistinguishable from uniform distribution. Thus we can replace $\mathbf{b}$ in the adversary by a uniform vector $\mathbf{u}$. The view of the adversary turns out to be

$$\left(A, \mathbf{u}, \lceil A^T \mathbf{r} \rceil_p, \langle \mathbf{u}, \mathbf{r} \rangle + \mu \cdot \lceil p/2 \rceil \right).$$

From the property of the rounding function $\lceil \cdot \rceil_p$, the vector $\mathbf{u}$ can be replaced by $\lceil \mathbf{v} \rceil_p$, where $\mathbf{v}$ is uniformly distributed over $\mathbb{Z}_q^n$. Then the view of the adversary becomes

$$\left(A, \lceil \mathbf{v} \rceil_p, \lceil A^T \mathbf{r} \rceil_p, \lceil \langle \mathbf{v}, \mathbf{r} \rangle \rceil_p + \mu \cdot \lceil p/2 \rceil \right).$$

The hardness of $\text{RSS}(n, q^{n+1})$ implies that the view of the adversary is computational indistinguishable from the uniform distribution. This finishes the semantic security proof of the scheme. $\square$

5.4 Rounded Subset Sum Problem With Weak and Leaky Secrets

In this section, we show that the secret of rounded subset sum problem owns high entropy under some lossy mode. Then, based on the subset sum problem and the high entropy result, we prove that the rounded subset sum problem is still pseudorandom even though partial information about the secret is learned.

Firstly, we prove that the secrets of rounded subset sum problem in the lossy mode has high entropy. Here lossy mode means the matrix is sampled such that the matrix is *computational* indistinguishable from uniform distribution if the underlying subset sum problem is hard, rather than directly sampling the matrix from uniform distribution.

Lemma 5.4.1. *Let m, l, n, q, p, r be parameters. Let $q \geq 2prmn\sqrt{n}$ be prime and $r = n^{\omega(1)}$. The distribution Lossy is defined as the output of the following procedure.*

- Uniformly randomly sample the matrix $B \in \mathbb{Z}_q^{m \times \ell}$ and $C \in \{0, 1\}^{\ell \times n}$. Sample Z from the discrete Gaussian distribution $\mathcal{D}_{\mathbb{Z}, r}^{m \times n}$.
- Let $\hat{A} = B \odot C + Z$.
- Output $\hat{A}$.

Assume that $\text{SS}(\ell, q^m)$ is hard. Let ϵ be a negligible function of λ . For $\mathbf{s}$ uniformly sampled from $\{0, 1\}^n$ and $\mathbf{aux}$ being a correlated random variable of $\mathbf{s}$, then we have

1. The Lossy distribution is computational indistinguishable from $U(\mathbb{Z}_q^{m \times n})$.
2. $H_\infty^\epsilon(\mathbf{s} | \hat{A}, \left\lceil \hat{A} \mathbf{s} \right\rceil_p) \geq n - (\ell + \lambda) \log q$.
3. $H_\infty^\epsilon(\mathbf{s} | \mathbf{aux}, \hat{A}, \left\lceil \hat{A} \mathbf{s} \right\rceil_p) \geq H_\infty(\mathbf{s} | \mathbf{aux}) - (\ell + \lambda) \log q$.

Proof. 1. From the definition of the Lossy distribution, we have

$$\hat{A} = B \odot C + Z = BC + F + Z,$$

where $B \odot C = BC + F$ is from Lemma 5.2.4. From the assumption that $\text{SS}(\ell, q^m)$ is hard, then $B \odot C$ is pseudorandom. Thus, $B \odot C + Z = \hat{A}$ is pseudorandom.

2. In our proof, we use $\epsilon = \epsilon_1 + \epsilon_2$, where ϵ, ϵ_1 and ϵ_3 are all negligible functions of λ . ϵ_1 and ϵ_2 are defined in the following. First, we prove that the Lossy distribution is statistically close to a distribution with independent noise vector, namely,

$$\hat{A} = BC + F + Z \approx_s BC + Z. \quad (5.1)$$

From Lemma 5.2.4, we have $|F|_\infty \leq n$ with $1 - 2^{-\Omega(n)}$ probability. Since Z is sampled from the discrete Normal distribution $\mathcal{D}_{\mathbb{Z}, r}^{m \times n}$ where $r = n^{\omega(1)}$, then

$$\text{SD}(F + Z, Z) \leq \frac{|F|_\infty}{r} \leq \frac{n}{r} = \text{negl}(\lambda)$$

by the noise flooding Lemma 2.2.6. Therefore the statistical distance between the distribution of $BC + F + Z$ and the distribution of $BC + Z$ is $\epsilon_1 = \text{negl}(\lambda)$, i.e. the approximate formula in Equation 5.1 holds.

Replacing $\hat{A}$ by $BC + Z$ in the min-entropy form, we have

$$H_{\infty}^{\epsilon} \left(\mathbf{s} | \hat{A}, \left\lceil \hat{A} \mathbf{s} \right\rceil_p \right) \geq H_{\infty}^{\epsilon - \epsilon_1} \left(\mathbf{s} | BC + Z, \left\lceil (BC + Z) \mathbf{s} \right\rceil_p \right).$$

Next we consider the entropy loss cause by the vector entry by entry. The entry by entry idea is firstly proposed by [AKPW13].

Define the set

$$\text{BAD} = \{i \in [m] : \left\lceil ((BC + Z) \mathbf{s})_i \right\rceil_p \neq \left\lceil (BC \mathbf{s})_i \right\rceil_p\}$$

where $(\cdot)_i$ represents the i -th entry of the vector. The set **BAD** represents those entries in which $(Z \mathbf{s})_i$ will change the rounding results of $(BC \mathbf{s})_i$. We will prove the size of this set is less than λ with overwhelming probability. If $\mathbf{s} = 0$, $|\text{BAD}| = 0$. Then for any fixed $\mathbf{s} \neq 0$,

$$\Pr[|\text{BAD}| \geq \lambda] = \Pr[C \mathbf{s} = 0] \Pr[|\text{BAD}| \geq \lambda | C \mathbf{s} = 0] + \Pr[C \mathbf{s} \neq 0] \Pr[|\text{BAD}| \geq \lambda | C \mathbf{s} \neq 0].$$

For the case $C \mathbf{s} = 0$, from $\Pr_C[C \mathbf{s} = 0] \leq \frac{1}{q^{\ell}}$, we obtain

$$\Pr[C \mathbf{s} = 0] \Pr[|\text{BAD}| \geq \lambda | C \mathbf{s} = 0] \leq \frac{1}{q^{\ell}} \cdot 1.$$

For the case $C \mathbf{s} \neq 0$, fix the value of $\mathbf{s}$ and C satisfying $C \mathbf{s} \neq 0$. We consider $\Pr[i \in \text{BAD}]$ for $i \in [m]$. Then we have,

$$\begin{aligned}
\Pr[i \in \text{BAD}] &= \Pr_{\mathbf{b}_i, \mathbf{z}_i} \left[\lceil \mathbf{b}_i \cdot C\mathbf{s} \rceil_p \neq \lceil \mathbf{b}_i \cdot C\mathbf{s} + \mathbf{z}_i\mathbf{s} \rceil_p \right] \\
&= \Pr_{\mathbf{b}_i, \mathbf{z}_i} \left[|\langle \mathbf{z}_i, \mathbf{s} \rangle| > \frac{q}{2p} - \left| \langle \mathbf{b}_i, C\mathbf{s} \rangle - \frac{q}{p} \lceil \langle \mathbf{b}_i, C\mathbf{s} \rangle \rceil_p \right| \right] \\
&= \sum_j \Pr_{\mathbf{z}_i} [|\langle \mathbf{z}_i, \mathbf{s} \rangle| = j] \Pr_{\mathbf{b}_i} \left[j > \frac{q}{2p} - \left| \mathbf{b}_i \cdot C\mathbf{s} - \frac{q}{p} \lceil \mathbf{b}_i \cdot C\mathbf{s} \rceil_p \right| \right] \\
&= \sum_j \Pr_{\mathbf{z}_i} [|\langle \mathbf{z}_i, \mathbf{s} \rangle| = j] \frac{\frac{q}{2p} - (\frac{q}{2p} - j)}{\frac{q}{2p}} \tag{5.2}
\end{aligned}$$

$$\begin{aligned}
&= \sum_j \Pr_{\mathbf{z}_i} [|\langle \mathbf{z}_i, \mathbf{s} \rangle| = j] \frac{2pj}{q} = \frac{2p}{q} \mathbb{E} [|\langle \mathbf{z}_i, \mathbf{s} \rangle|] \\
&\leq \frac{2prn\sqrt{n}}{q} \leq \frac{1}{m}, \tag{5.3}
\end{aligned}$$

where $\mathbf{b}_i$ and $\mathbf{z}_i$ are the i -th row vectors of B and Z . The value $\left| \langle \mathbf{b}_i, C\mathbf{s} \rangle - \frac{q}{p} \lceil \langle \mathbf{b}_i, C\mathbf{s} \rangle \rceil_p \right|$ is uniformly distributed over $[0, \dots, \frac{q}{2p}]$ since $\langle \mathbf{b}_i, C\mathbf{s} \rangle$ is uniformly distributed over $\mathbb{Z}_q$. So

$$\Pr_{\mathbf{b}_i} \left[\left| \langle \mathbf{b}_i, C\mathbf{s} \rangle - \frac{q}{p} \lceil \langle \mathbf{b}_i, C\mathbf{s} \rangle \rceil_p \right| > \frac{q}{2p} - j \right] = \frac{\frac{q}{2p} - (\frac{q}{2p} - j)}{\frac{q}{2p}}.$$

Thus, Equation 5.2 holds. Because $|\mathbf{z}_i|_\infty \leq r\sqrt{n}$ with overwhelming probability and $\mathbf{s}$ is drawn from the uniform distribution over $\{0, 1\}^n$, we have $|\langle \mathbf{z}_i, \mathbf{s} \rangle| \leq n \cdot r\sqrt{n}$ with overwhelming probability. Then Equation 5.3 holds.

By the above result of $\Pr[i \in \text{BAD}] \leq \frac{1}{m}$ for $i \in [m]$, and events $i \in \text{BAD}$ are independent of each other based on independently chosen $\mathbf{b}_i, \mathbf{z}_i$, we obtain $\mathbb{E}(|\text{BAD}|) \leq 1$. Thus by the Chernoff bound Lemma 2.2.3, $\Pr[C\mathbf{s} \neq 0] \Pr[|\text{BAD}| \geq \lambda | C\mathbf{s} \neq 0] \leq 1 \cdot \frac{1}{2^\lambda}$.

Therefore, $\Pr[|\text{BAD}| \geq \lambda] \leq \frac{1}{q^\lambda} + \frac{1}{2^\lambda} = \text{negl}(\lambda) := \epsilon_2$, i.e. the size of BAD is less than λ with overwhelming probability.

Define $\mathsf{T} := \{(i, \lceil (BC + Z)\mathbf{s} \rceil_p) : i \in \text{BAD}\}$. Then for $\epsilon = \epsilon_1 + \epsilon_2$,

$$\begin{aligned}
H_\infty^\epsilon \left(\mathbf{s} | \hat{A}, \left\lceil \hat{A} \mathbf{s} \right\rceil_p \right) &\geq H_\infty^{\epsilon - \epsilon_1} \left(\mathbf{s} | BC + Z, \lceil (BC + Z) \mathbf{s} \rceil_p \right) \\
&\geq H_\infty^{\epsilon_2} (\mathbf{s} | B, C, Z, C\mathbf{s}, \mathbf{T}) \\
&\geq H_\infty (\mathbf{s} | B, C, Z, C\mathbf{s}) - \lambda \log m - \lambda \log p \tag{5.4}
\end{aligned}$$

$$\begin{aligned}
&\geq H_\infty (\mathbf{s} | B, C, Z) - \ell \log q - \lambda \log m - \lambda \log p \\
&\geq n - (\ell + \lambda) \log q. \tag{5.5}
\end{aligned}$$

Considered $\mathbf{T}$ as a binary string, the length of $\mathbf{T}$ is $(\log p + \log m) |\mathbf{BAD}| \leq \lambda(\log p + \log m)$, the Equation 5.4 holds. Since $q \geq 2pmrn\sqrt{n}$ and B, C, Z are independent with $\mathbf{s}$, the Equation 5.5 holds. Thu, the entropy of s is still big enough via appropriate choosing the parameters.

3. By considering $(\mathbf{s}, \mathbf{aux})$ as correlated random variables, we can apply the proof technique of 2 to get

$$\begin{aligned}
H_\infty^\epsilon \left(\mathbf{s} | \mathbf{aux}, \hat{A}, \left\lceil \hat{A} \mathbf{s} \right\rceil_p \right) &\geq H_\infty^{\epsilon - \epsilon_1} \left(\mathbf{s} | \mathbf{aux}, BC + Z, \lceil (BC + Z) \mathbf{s} \rceil_p \right) \\
&\geq H_\infty^{\epsilon_2} (\mathbf{s} | \mathbf{aux}, B, C, Z, C\mathbf{s}, \mathbf{T}) \\
&\geq H_\infty (\mathbf{s} | \mathbf{aux}, B, C, Z, C\mathbf{s}) - \lambda \log m - \lambda \log p \\
&\geq H_\infty (\mathbf{s} | \mathbf{aux}, B, C, Z) - \ell \log q - \lambda \log m - \lambda \log p \\
&\geq H_\infty (\mathbf{s} | \mathbf{aux}) - (\lambda + \ell) \log q.
\end{aligned}$$

□

Then we show that the rounded subset sum distribution with weak and leaky secrets is still pseudorandom. Since we use the above entropy results, the parameter q need to be super-polynomial of n . And the number of samples m need to be given out in advance. We use a

series of hybrid games to prove the pseudorandomness of the rounded subset sum distribution with weak and leaky secrets.

Theorem 5.4.2. *Let $q \geq 2pmn^{\omega(1)}$ be a prime. Assuming that $\text{SS}(\ell, q^m)$ is hard, then*

1. *If $(\lambda + \ell + 1) \log q + 2\lambda \leq n$, the $\text{RSS}(n, m, q, p)$ distribution is pseudorandom.*
2. *If $(\lambda + \ell + 1) \log q + 2\lambda \leq k$, the weak and leaky $\text{RSS}(n, m, q, p)$ distribution is pseudorandom, which means that $(\text{aux}, A, \lceil A\mathbf{s} \rceil_p) \approx_c (\text{aux}, A, \lceil \mathbf{u} \rceil_p)$ where $H_\infty^\epsilon(\mathbf{s}|\text{aux}) \geq k$ and $\mathbf{u}$ is uniformly sampled from $\mathbb{Z}_q^m$.*

Proof. We first prove the second part.

Let $\mathcal{H}_i = (\text{aux}, A, \mathbf{b}^{(i)})$ denotes the hybrid distribution for $i \in [0, \dots, m]$, in which, the first i entries of $\mathbf{b}^{(i)} \in \mathbb{Z}_q^i$ are the first i entries of $A\mathbf{s} \in \mathbb{Z}_q^n$, and the rest $m - i$ entries are uniform variables $\mathbf{u}$ sampled from $\mathbb{Z}_q^{m-i}$.

Thus, $\mathcal{H}_m$ is $(\text{aux}, A, \lceil A\mathbf{s} \rceil_p)$, and $\mathcal{H}_0$ is $(\text{aux}, A, \lceil \mathbf{u} \rceil_p)$. For any $0 \leq i \leq m$,

$$\begin{aligned} \mathcal{H}_{i+1} &= \left(\text{aux}, \begin{bmatrix} A_1 \\ \mathbf{a}^T \\ A_2 \end{bmatrix}, \begin{bmatrix} \lceil A_1 \mathbf{s} \rceil_p \\ \lceil \langle \mathbf{a}, \mathbf{s} \rangle \rceil_p \\ \lceil \mathbf{u} \rceil_p \end{bmatrix} \right) \\ &\approx_c \left(\text{aux}, \begin{bmatrix} \hat{A}_1 \\ \mathbf{a}^T \\ A_2 \end{bmatrix}, \begin{bmatrix} \lceil \hat{A}_1 \mathbf{s} \rceil_p \\ \lceil \mathbf{a}^T \mathbf{s} \rceil_p \\ \lceil \mathbf{u} \rceil_p \end{bmatrix} \right) \end{aligned} \quad (5.6)$$

$$\approx_s \left(\text{aux}, \begin{bmatrix} \hat{A}_1 \\ \mathbf{a}^T \\ A_2 \end{bmatrix}, \begin{bmatrix} \lceil \hat{A}_1 \mathbf{s} \rceil_p \\ \lceil u_1 \rceil_p \\ \lceil \mathbf{u} \rceil_p \end{bmatrix} \right) \quad (5.7)$$

$$\approx_c \left(\text{aux}, \begin{bmatrix} A_1 \\ \mathbf{a}^T \\ A_2 \end{bmatrix}, \begin{bmatrix} \lceil A_1 \mathbf{s} \rceil_p \\ \lceil u_1 \rceil_p \\ \lceil \mathbf{u} \rceil_p \end{bmatrix} \right) = \mathcal{H}_i \quad (5.8)$$

where $A_1 \in \mathbb{Z}_q^{i \times n}$, $A_2 \in \mathbb{Z}_q^{(m-i-1) \times n}$, $\mathbf{a} \in \mathbb{Z}_q^n$, $u_1 \in \mathbb{Z}_q$, $\mathbf{u} \in \mathbb{Z}_q^{m-i-1}$, and $\mathbf{s} \in \{0, 1\}^n$, are all uniformly sampled from their domain, and $\hat{A}_1$ is sampled from the **Lossy** distribution. By Lemma 5.4.1 (i), $\hat{A}$ is computational indistinguishable from the uniform distribution from the hardness of $\text{SS}(\ell, q^{i+1})$, so the approximate Equation 5.6 holds. By Lemma 5.4.1 (iii), $H_\infty^\epsilon(\mathbf{s}|\text{aux}, \hat{A}, \left\lceil \hat{A}\mathbf{s} \right\rceil_p) \geq H_\infty(\mathbf{s}|\text{aux}) - (l + \lambda) \log q \geq 2\lambda + \log q$, $\mathbf{s}$ still has enough entropy. From the leftover hash lemma, the approximate Equation 5.7 holds. Using the pseudorandomness $\hat{A}$ from Lemma 5.4.1 (i), the approximate Equation 5.8 holds.

The first part follows by replacing aux to an independent random variable and Lemma 5.4.1 (ii). $\square$

5.5 Adaptive Leakage-Resilient Secure PKE

In this section, we prove that the PKE in Section 5.3 with new parameters is leakage-resilient secure against $(n - o(n))$ -adaptive memory attacks. We follow the security model defined in [AGV09].

Construction. The structure is same as the PKE scheme in Section 5.3.

Parameters. $m \geq (\ell + \lambda + 1) \log q + 2 \log p + 2\lambda$, $q \geq 2pmn^{\omega(1)}$, $n \geq (\ell + \lambda + 3) \log q + 2\lambda$ and $m + n < \frac{p}{8}$.

Correctness. For any key pair (sk, pk) generated by $\text{KeyGen}(1^\lambda)$ and any message μ , $\text{Dec}(\text{sk}, \text{Enc}(\mu, \text{pk})) = \mu$ with overwhelming probability. The proof is same as Theorem 5.3.1. Next we prove the security of this scheme.

Theorem 5.5.1 (Security). *Assuming that $\text{SS}(n, q^{n+1})$ is hard, the above public key cryptosystem achieves semantic security against adaptive $\alpha(n)$ -memory attacks, where $\alpha(n) = n - (\ell + \lambda) \log q - 3 \log p - 2\lambda$.*

Proof. As stated in [AGV09], without loss of generality, we consider the adversary that has access to a single function h with $\alpha(n)$ bounded output length and the adversary gets $(\mathbf{pk}, h(\mathbf{pk}, \mathbf{sk}))$ as a result.

For a message $\mu \in \{0, 1\}$, the adversary will get

$$(\mathbf{pk}, \text{Enc}(\mathbf{pk}, \mu), h(\mathbf{pk}, \mathbf{sk})) = (A, \lceil A\mathbf{s} \rceil_p, \lceil A^T \mathbf{r} \rceil_p, \mathbf{r}^T \lceil A\mathbf{s} \rceil_p + \mu \lceil \frac{p}{2} \rceil, h(\mathbf{pk}, \mathbf{sk})).$$

By the definition of rounded subset sum problem, $\lceil A\mathbf{s} \rceil_p = \frac{p}{q}A\mathbf{s} + \mathbf{e}_1$ and $\lceil A^T \mathbf{r} \rceil_p = \frac{p}{q}A^T \mathbf{r} + \mathbf{e}_2$ for certain $\|\mathbf{e}_1\|_\infty < \frac{1}{2}$ and certain $\|\mathbf{e}_2\|_\infty < \frac{1}{2}$. Then

$$\begin{aligned} & \left(A, \lceil A\mathbf{s} \rceil_p, \lceil A^T \mathbf{r} \rceil_p, \langle \mathbf{r}, \lceil A\mathbf{s} \rceil_p \rangle + \mu \lceil \frac{p}{2} \rceil, h(\mathbf{pk}, \mathbf{sk}) \right) \\ &= \left(A, \lceil A\mathbf{s} \rceil_p, \lceil A^T \mathbf{r} \rceil_p, \frac{p}{q} \langle \mathbf{r}, A\mathbf{s} \rangle + \langle \mathbf{r}, \mathbf{e}_1 \rangle + \mu \lceil \frac{p}{2} \rceil, h(\mathbf{pk}, \mathbf{sk}) \right) \\ &= \left(A, \lceil A\mathbf{s} \rceil_p, \lceil A^T \mathbf{r} \rceil_p, \frac{p}{q} \langle A^T \mathbf{r}, \mathbf{s} \rangle + \langle \mathbf{e}_2, \mathbf{s} \rangle + \langle \mathbf{r}, \mathbf{e}_1 \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle + \mu \lceil \frac{p}{2} \rceil, h(\mathbf{pk}, \mathbf{sk}) \right) \\ &= \left(A, \lceil A\mathbf{s} \rceil_p, \lceil A^T \mathbf{r} \rceil_p, \langle \lceil A^T \mathbf{r} \rceil_p, \mathbf{s} \rangle + \langle \mathbf{r}, \mathbf{e}_1 \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle + \mu \lceil \frac{p}{2} \rceil, h(\mathbf{pk}, \mathbf{sk}) \right). \end{aligned} \quad (5.9)$$

Let $\text{aux}(\mathbf{r}, \mathbf{e}_2) = \langle \mathbf{r}, \mathbf{e}_1 \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle$. Then

$$H_\infty(\mathbf{r} | \text{aux}) \geq H_\infty(\mathbf{r} | \langle \mathbf{r}, \mathbf{e}_1 \rangle, \langle \mathbf{e}_2, \mathbf{s} \rangle) \geq m - 2 \log p \geq (\ell + \lambda + 1) \log q + 2\lambda,$$

where $\mathbf{e}_2 = \lceil A^T \mathbf{r} \rceil_p - \frac{p}{q}A^T \mathbf{r} \pmod{p}$ is correlated with $\mathbf{r}$.

From the above entropy result and Theorem 5.4.2(ii), we can replace $\lceil A^T \mathbf{r} \rceil_p$ by $\mathbf{u}$ namely,

$$\left(A, \lceil A^T \mathbf{r} \rceil_p, \langle \mathbf{r}, \mathbf{e}_1 \rangle, \langle \mathbf{e}_2, \mathbf{s} \rangle \right) \approx_s \left(A, \mathbf{u}, \langle \mathbf{r}, \mathbf{e}_1 \rangle, \langle \mathbf{e}_2, \mathbf{s} \rangle \right),$$

where $\mathbf{u}$ represents a random variable uniformly sampled from $\mathbb{Z}_p^n$.

Hence, replacing $\lceil A^T \mathbf{r} \rceil_p$ by $\mathbf{u}$ in Equation 5.9, we have

$$\begin{aligned} & \left(A, \lceil A\mathbf{s} \rceil_p, \lceil A^T \mathbf{r} \rceil_p, \langle \lceil A^T \mathbf{r} \rceil_p, \mathbf{s} \rangle + \langle \mathbf{r}, \mathbf{e}_1 \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle + \mu \left\lceil \frac{p}{2} \right\rceil, h(\mathbf{pk}, \mathbf{sk}) \right) \\ & \approx_s \left(A, \lceil A\mathbf{s} \rceil_p, \mathbf{u}, \langle \mathbf{u}, \mathbf{s} \rangle + \langle \mathbf{r}, \mathbf{e}_1 \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle + \mu \left\lceil \frac{p}{2} \right\rceil, h(\mathbf{pk}, \mathbf{sk}) \right). \end{aligned} \quad (5.10)$$

Let $\hat{A} = B \odot C + Z$, where $B \in \mathbb{Z}_q^{m \times \ell}$ and $C \in \{0, 1\}^{\ell \times n}$ are uniformly sampled from their domain, and Z is sampled from the distribution $\mathcal{D}_{\mathbb{Z}, r}^{m \times n}$. From Lemma 5.4.1(i) and the assumption $\text{SS}(\ell, q^m)$ is hard, we have $A \approx_c \hat{A}$.

Then replacing A by $\hat{A}$, we have

$$\begin{aligned} & \left(A, \lceil A\mathbf{s} \rceil_p, \mathbf{u}, \langle \mathbf{u}, \mathbf{s} \rangle + \langle \mathbf{r}, \mathbf{e}_1 \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle + \mu \left\lceil \frac{p}{2} \right\rceil, h(\mathbf{pk}, \mathbf{sk}) \right) \\ & \approx_c \left(\hat{A}, \lceil \hat{A}\mathbf{s} \rceil_p, \mathbf{u}, \langle \mathbf{u}, \mathbf{s} \rangle + \langle \mathbf{r}, \mathbf{e}_1 \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle + \mu \left\lceil \frac{p}{2} \right\rceil, h(\widehat{\mathbf{pk}}, \mathbf{sk}) \right), \end{aligned} \quad (5.11)$$

where $\widehat{\mathbf{pk}} := \left(\hat{A}, \lceil \hat{A}\mathbf{s} \rceil_p \right)$.

Next from Lemma 5.4.1 (iii), we have

$$H_\infty^e \left(\mathbf{s} | \hat{A}, \lceil \hat{A}\mathbf{s} \rceil_p, \langle \mathbf{r}, \mathbf{e}_1 \rangle, \langle \mathbf{e}_2, \mathbf{s} \rangle, h(\widehat{\mathbf{pk}}, \mathbf{sk}) \right) \geq n - (\ell + \lambda) \log q - 2 \log p - \alpha(n) \geq \log p + 2\lambda.$$

From the above entropy result and the leftover hash lemma 2.2.2, we can replace $\langle \mathbf{u}, \mathbf{s} \rangle$ by u in Equation 5.11, namely,

$$\begin{aligned} & \left(\hat{A}, \lceil \hat{A}\mathbf{s} \rceil_p, \mathbf{u}, \langle \mathbf{u}, \mathbf{s} \rangle + \langle \mathbf{r}, \mathbf{e}_1 \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle + \mu \left\lceil \frac{p}{2} \right\rceil, h(\widehat{\mathbf{pk}}, \mathbf{sk}) \right) \\ & \approx_s \left(\hat{A}, \lceil \hat{A}\mathbf{s} \rceil_p, \mathbf{u}, u + \langle \mathbf{r}, \mathbf{e}_1 \rangle - \langle \mathbf{e}_2, \mathbf{s} \rangle + \mu \left\lceil \frac{p}{2} \right\rceil, h(\widehat{\mathbf{pk}}, \mathbf{sk}) \right) \\ & \approx_s \left(\hat{A}, \lceil \hat{A}\mathbf{s} \rceil_p, \mathbf{u}, u' + \mu \left\lceil \frac{p}{2} \right\rceil, h(\widehat{\mathbf{pk}}, \mathbf{sk}) \right). \end{aligned} \quad (5.12)$$

where u and u' are drawn from $\mathbb{Z}_p$ randomly and independently.

Using Lemma 5.4.1(i) again to replace $\hat{A}$ back to A ,

$$\left(\hat{A}, \left\lceil \hat{A}\mathbf{s} \right\rceil_p, \mathbf{u}, u' + \mu \left\lceil \frac{p}{2} \right\rceil, h(\widehat{\mathbf{pk}}, \mathbf{sk}) \right) \approx_c \left(A, \left\lceil A\mathbf{s} \right\rceil_p, \mathbf{u}, u' + \mu \left\lceil \frac{p}{2} \right\rceil, h(\mathbf{pk}, \mathbf{sk}) \right).$$

This step also relies on the $\text{SS}(\ell, q^m)$ assumption.

Therefore, $(\mathbf{pk}, \text{Enc}(\mathbf{pk}, \mu), h(\mathbf{pk}, \mathbf{sk})) \approx_c (A, \left\lceil A\mathbf{s} \right\rceil_p, \mathbf{u}, u' + \mu \left\lceil \frac{p}{2} \right\rceil, h(\mathbf{pk}, \mathbf{sk}))$. $\square$

In our security proof, we use the weak and leaky secrets property of rounded subset sum problem to establish the security against adaptive leakage of the secret key.

5.6 Reusable Extractors

Dodis et al. [DKL09] proposed the notion *computational reusable extractor*, which allows one to extract many random outputs from the *same* source $\mathbf{s}$ with different public seeds as long as the secret source has sufficient entropy. [DKL09] constructed a computational reusable extractor under the auxiliary-input learning parity with noise assumption. Alwen et al. [AKPW13] proposed a reusable extractor construction, whose security relies on the LWE problem. Here we construct reusable extractors from the rounded subset sum problem.

Before presenting our constructions, we show the definition of reusable extractors.

Definition 5.6.1 (Reusable Extractor). *Let D be the domain of the public seed, S be the domain of the source and U be the range of the extractor.*

For any polynomially many samplable correlated random variable pair $(\mathbf{s}, \mathbf{aux})$ satisfying the domain of $\mathbf{s}$ being S , $H_\infty(\mathbf{s}|\mathbf{aux}) \geq r$, and the two distributions being computational indistinguishable,

$$(\mathbf{aux}, \mathbf{a}_1, \dots, \mathbf{a}_t, \text{Ext}(\mathbf{s}, \mathbf{a}_1), \dots, \text{Ext}(\mathbf{s}, \mathbf{a}_t)) \approx (\mathbf{aux}, \mathbf{a}_1, \dots, \mathbf{a}_t, u_1, \dots, u_t),$$

where the $\mathbf{a}_i$ is uniformly drawn from D and u_i is uniformly drawn from U , $\text{Ext} : S \times D \rightarrow U$

is defined as a (r, t) -reusable extractor.

Theorem 5.6.2. *Let n, t, p, q be integers. Assume that $\text{SS}(l, q^t)$ is hard. Given $(l + \lambda + 1) \log q + 2\lambda \leq n$, the function*

$$\text{Ext} : \{0, 1\}^n \times \mathbb{Z}_q^n \rightarrow \mathbb{Z}_p \text{ where } \text{Ext}(\mathbf{s}, \mathbf{a}) = \lceil \langle \mathbf{s}, \mathbf{a} \rangle \rceil_p$$

becomes a (r, t) -reusable extractor.

Proof. According to the definition of reusable extractor, we would like to show that

$$(\mathbf{aux}, \mathbf{a}_1, \dots, \mathbf{a}_t, \text{Ext}(\mathbf{s}, \mathbf{a}_1), \dots, \text{Ext}(\mathbf{s}, \mathbf{a}_t)) \approx (\mathbf{aux}, \mathbf{a}_1, \dots, \mathbf{a}_t, u_1, \dots, u_t). \quad (5.13)$$

The proof directly follows Theorem 5.4.2. Let $A \in \mathbb{Z}_q^{t \times n}$ be a matrix with $A^T = (\mathbf{a}_1, \dots, \mathbf{a}_t)$. Then we have $(\mathbf{aux}, A, \lceil A\mathbf{s} \rceil_p) \approx (\mathbf{aux}, A, \lceil \mathbf{u} \rceil_p)$, where $\mathbf{u}$ is the uniform distribution over $\mathbb{Z}_q^t$. This is indeed the matrix form of Equation 5.13. $\square$

Note that in our construction, t cannot be arbitrary polynomial of the security parameter. The security of the reusable extractor construction is built on the subset sum problem $\text{SS}(l, q^t)$. For very large t , the subset sum problem becomes no longer hard.

6. CONCLUSIONS

In this thesis, we present three quantum-safe constructions, including code-based constructions, lattice-based constructions, and the rounded subset sum problem and its applications. In this chapter, we summarize our contributions with some interesting problems.

In Chapter 3, we presented a general construction to reduce the public key size of McEliece cryptosystems constructed from automorphism-induced Goppa codes. We generalized the ideas of automorphism-induced Goppa codes by considering proper subsets of automorphism subgroups to construct Goppa codes. Our construction is resistant to the invariant subcode attack, because in our construction, the automorphism subgroup induce Goppa codes is trivial. Thus, applying the folding operation to our construction does not yield any meaningful subcode. In our construction, we considered two instantiations, namely, the partial quasi-cyclic construction and the partial quasi-dyadic construction. Our construction can also be combined with existing automorphism-induced Goppa codes to further reduce the public key size.

In Chapter 4, we proposed a new lattice based trapdoor functions to solve the bounded distance decoding problem that inherits the nice properties of the McEliece encryption scheme. Based on the trapdoor function, we designed an efficient public key encryption scheme that can be converted to achieve CCA-security. We estimated the security of the trapdoor functions and the public key encryption scheme.

In Chapter 5, we proposed the rounded subset sum problem as a variant of the subset sum problem. We proved that the rounded subset sum problem is as hard as the subset sum problems. We presented a public key cryptosystem achieving IND-CPA based on the rounded subset sum problem. Then we proved that the rounded subset sum problem is still hard with

weak and leaky secrets. Based on the weak and leaky results, we constructed an adaptive leakage resilient public key encryption scheme, which solves an open problem proposed in the work of Lyubashevsky, Palacio, and Segev (TCC 2010). Finally, we presented other applications of the rounded subset sum problem.

6.1 Future Work

Several interesting open problems arise from our research. In this section, we highlight a couple of the main related problems.

For the Goppa code-based construction, the main problem is to significantly reduce the public key size of the scheme that remains resistant to algebraic attacks and other attacks. While we have provided a generic construction, we have only considered two explicit forms of the constructions, the additive automorphism subgroups and the multiplicative automorphism subgroups. In fact, one can directly combine the multiplicative subgroups and additive subgroups to consider a general subgroup of $\text{AGL}_2(q)$. It is interesting to investigate if more general subgroups can lead to even smaller key sizes for some sets of parameters.

Let the automorphism group be denoted by $\mathcal{G}$. In our constructions, we only consider the case that the desired Goppa code is a union of sub-Goppa codes, where every sub-Goppa code is induced by h_σ and h_σ is the product of $|\mathcal{G}| - 1$ polynomials. We expect future work to consider smaller number of polynomials, for example, $|\mathcal{G}| - 2, |\mathcal{G}| - 3, \dots$. In this case, the induced Goppa codes are no longer disjoint. One can attempt to adapt our approach to reduce the public key size without compromising the security.

While our lattice-based scheme is an improvement over the GGH scheme, the public key sizes remain large. A potential direction to reduce the key size is to employ the approach of the induced automorphism groups as in our code-based construction. To this end, one may generalize our construction from polynomials to other function fields with a large automorphism group, such as the Hermitian function fields. Likewise, the subfield subcode of the Hermitian

codes can be employed to reduce the public key size of McEliece schemes.

Last but not least, we expect that the techniques used in the rounded subset sum-based schemes can be adapted to yield more applications directly constructed from subset sum problem or the rounded subset sum problem. This may potentially lead to better parameters as the subset sum problem is a weaker assumption than existing lattice assumptions. For instance, an interesting problem is to construct pseudorandom functions directly from the subset sum problem or the rounded subset sum problem. Note that the subset sum function is a pseudorandom generator. Following the framework of [GGM84] to construct pseudorandom functions is then straightforward. Thus, we expect that one can construct more efficient pseudorandom functions from the subset sum problem, for example, constructing pseudorandom synthesizers [NR95] as tools to construct pseudorandom functions.

BIBLIOGRAPHY

- [AAB⁺19] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019. [1]
- [AABN02] Michel Abdalla, Jee Hea An, Mihir Bellare, and Chanathip Namprempre. From identification to signatures via the fiat-shamir transform: Minimizing assumptions for security and forward-security. In *EUROCRYPT*, volume 2332, pages 418–433. Springer, 2002. [9]
- [ABB⁺18] Nicolas Aragon, Paulo Barreto, Slim Bettaieb, Loic Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Phillipe Gaborit, Shay Gueron, Tim Guneyasu, Carlos Aguilar Melchor, Rafael Misoczki, Edoardo Persichetti, Nicolas Sendrier, Jean-Pierre Tillich, and Gilles Zémor. BIKE: Bit Flipping Key Encapsulation, Oct 2018. <https://bikesuite.org/files/BIKE.pdf>. [3]
- [ABD⁺19] N Aragon, O Blazy, JC Deneuville, P Gaborit, A Hauteville, O Ruatta, JP Tillich, G Zémor, C Aguilar Melchor, S Bettaieb, et al. **ROLLO - Rank-Ouroboros, LAKE & LOCKER**, Aug 2019. https://pqc-rollo.org/doc/rollo-specification_2019-08-24.pdf. [5]
- [ABD⁺20] N Aragon, O Blazy, JC Deneuville, P Gaborit, A Hauteville, O Ruatta, JP Tillich, G Zémor, C Aguilar Melchor, S Bettaieb, et al. **ROLLO - Rank-Ouroboros**,

- LAKE & LOCKER, Apr 2020. https://pqc-rqc.org/doc/rqc-specification_2020-04-21.pdf. [5]
- [AD97] Miklós Ajtai and Cynthia Dwork. A public-key cryptosystem with worst-case/average-case equivalence. In *STOC 1997*, pages 284–293, 1997. [4, 6, 21]
- [ADPS16] Erdem Alkim, Léo Ducas, Thomas Pöppelmann, and Peter Schwabe. Post-quantum key exchange - A new hope. In *25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016.*, pages 327–343, 2016. [77]
- [AGV09] Adi Akavia, Shafi Goldwasser, and Vinod Vaikuntanathan. Simultaneous hardcore bits and cryptography against memory attacks. In *TCC 2009*, volume 5444, pages 474–495. Springer, 2009. [27, 47, 106, 107]
- [AGVW17] Martin R. Albrecht, Florian Göpfert, Fernando Virdia, and Thomas Wunderer. Revisiting the expected cost of solving usvp and applications to LWE. In *Advances in Cryptology - ASIACRYPT 2017 - 23rd International Conference on the Theory and Applications of Cryptology and Information Security, Hong Kong, China, December 3-7, 2017, Proceedings, Part I*, pages 297–322, 2017. [34, 77]
- [Ajt96] Miklós Ajtai. Generating hard instances of lattice problems (extended abstract). In *STOC 1996*, pages 99–108. ACM, 1996. [6]
- [AKPW13] Joël Alwen, Stephan Krenn, Krzysztof Pietrzak, and Daniel Wichs. Learning with rounding, revisited - new reduction, properties and applications. In *CRYPTO 2013*, pages 57–74, 2013. [25, 102, 109]
- [Alb17] Martin R. Albrecht. On dual lattice attacks against small-secret LWE and parameter choices in helib and SEAL. In *Advances in Cryptology - EUROCRYPT 2017*

- *36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part II*, pages 103–129, 2017. [34]
- [Ale03] Michael Alekhnovich. More on average case vs approximation complexity. In *FOCS*, pages 298–307, 2003. [4]
- [APS15] Martin R. Albrecht, Rachel Player, and Sam Scott. On the concrete hardness of learning with errors. *J. Mathematical Cryptology*, 9(3):169–203, 2015. [35, 78]
- [Bar18] Elise Barelli. On the security of some compact keys for McEliece scheme. *CoRR*, abs/1803.05289, 2018. [18, 44, 45, 52]
- [BBB⁺18a] Gustavo Banegas, Paulo S. L. M. Barreto, Brice Odilon Boidje, Pierre-Louis Cayrel, Gilbert Ndollane Dione, Kris Gaj, Cheikh Thiécoumba Gueye, Richard Haeussler, Jean Belo Klamti, Ousmane Ndiaye, Duc Tri Nguyen, Edoardo Persichetti, and Jefferson E. Ricardini. DAGS: Key encapsulation from dyadic GS codes, Jun 2018. https://www.dags-project.org/pdf/DAGS_spec_v2.pdf. [3, 41, 43]
- [BBB⁺18b] Magali Bardet, Elise Barelli, Olivier Blazy, Rodolfo Canto-Torres, Alain Couvreur, Philippe Gaborit, Ayoub Otmani, Nicolas Sendrier, and Jean-Pierre Tillich. BIG QUAKE: BInary Goppa QUAsi-cyclic Key Encapsulation, Apr 2018. https://bigquake.inria.fr/files/2018/04/corrected_proposal.pdf. [3, 17, 19, 41, 43, 52, 58, 61, 84]
- [BBB⁺20] Magali Bardet, Pierre Briaud, Maxime Bros, Philippe Gaborit, Vincent Neiger, Olivier Ruatta, and Jean-Pierre Tillich. An algebraic attack on rank metric code-based cryptosystems. In *EUROCRYPT*, volume 12107, pages 64–93. Springer, 2020. [5]

- [BBC⁺20] Magali Bardet, Maxime Bros, Daniel Cabarcas, Philippe Gaborit, Ray A. Perlner, Daniel Smith-Tone, Jean-Pierre Tillich, and Javier A. Verbel. Algebraic attacks for solving the rank decoding and minrank problems without gröbner basis. *CoRR*, abs/2002.08322, 2020. [5]
- [BBD08] Daniel J. Bernstein, Johannes Buchmann, and Erik Dahmen. *Post Quantum Cryptography*. Springer Publishing Company, Incorporated, 1st edition, 2008. [32]
- [BBdV⁺17] Jens Bauch, Daniel J. Bernstein, Henry de Valence, Tanja Lange, and Christine van Vredendaal. Short generators without quantum computers: The case of multiquadratics. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT*, volume 10210 of *Lecture Notes in Computer Science*, pages 27–59, 2017. [16]
- [BCGO09] Thierry P. Berger, Pierre-Louis Cayrel, Philippe Gaborit, and Ayoub Otmani. Reducing key length of the McEliece cryptosystem. In *AFRICACRYPT*, volume 5580, pages 77–97. Springer, 2009. [17, 41, 43, 52]
- [BCJ11] Anja Becker, Jean-Sébastien Coron, and Antoine Joux. Improved generic algorithms for hard knapsacks. In *EUROCRYPT 2011*, volume 6632, pages 364–385. Springer, 2011. [10]
- [BCL⁺17] Daniel J. Bernstein, Tung Chou, Tanja Lange, Ingo von Maurich, Rafael Misoczki, Ruben Niederhagen, Edoardo Persichetti, Christiane Peters, Peter Schwabe, Nicolas Sendrier, Jakub Szefer, and Wen Wang. Classic McEliece: conservative code-based cryptography, Nov 2017. <https://classic.mceliece.org/nist/mceliece-20171129.pdf>. [3]
- [BDF⁺11] Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random oracles in a quantum world. In *Advances in Cryptology*

- *ASIACRYPT 2011 - 17th International Conference on the Theory and Application of Cryptology and Information Security, Seoul, South Korea, December 4-8, 2011. Proceedings*, pages 41–69, 2011. [15]
- [BDH11] Johannes A. Buchmann, Erik Dahmen, and Andreas Hülsing. XMSS - A practical forward secure signature scheme based on minimal security assumptions. In *PQCrypto*, pages 117–129, 2011. [12]
- [Ber00] Thierry P. Berger. Goppa and related codes invariant under a prescribed permutation. *IEEE Trans. Information Theory*, 46(7):2628–2633, 2000. [17, 41]
- [BFKL93] Avrim Blum, Merrick L. Furst, Michael J. Kearns, and Richard J. Lipton. Cryptographic primitives based on hard learning problems. In *CRYPTO '93*, pages 278–291, 1993. [4]
- [BGM⁺16] Andrej Bogdanov, Siyao Guo, Daniel Masny, Silas Richelson, and Alon Rosen. On the hardness of learning with rounding over small modulus. In *TCC 2016-A*, pages 209–224, 2016. [24, 92, 93, 94, 95]
- [BHH⁺15] Daniel J. Bernstein, Daira Hopwood, Andreas Hülsing, Tanja Lange, Ruben Niederhagen, Louiza Papachristodoulou, Michael Schneider, Peter Schwabe, and Zooko Wilcox-O’Hearn. SPHINCS: practical stateless hash-based signatures. In *EUROCRYPT*, pages 368–397, 2015. [12]
- [BHH⁺19] Nina Bindel, Mike Hamburg, Kathrin Hövelmanns, Andreas Hülsing, and Edoardo Persichetti. Tighter proofs of CCA security in the quantum random oracle model. In *Theory of Cryptography - 17th International Conference, TCC 2019, Nuremberg, Germany, December 1-5, 2019, Proceedings, Part II*, pages 61–90, 2019. [15]
- [BHK⁺19] Daniel J. Bernstein, Andreas Hülsing, Stefan Kölbl, Ruben Niederhagen, Joost

- Rijneveld, and Peter Schwabe. The sphincs⁺ signature framework. In *CCS*, pages 2129–2146, 2019. [12]
- [BJMM12] Anja Becker, Antoine Joux, Alexander May, and Alexander Meurer. Decoding random binary linear codes in $2^{n/20}$: How $1 + 1 = 0$ improves information set decoding. In *EUROCRYPT*, volume 7237, pages 520–536. Springer, 2012. [57]
- [BKV19] Ward Beullens, Thorsten Kleinjung, and Frederik Vercauteren. Csi-fish: Efficient isogeny based signatures through class group computations. In *ASIACRYPT*, pages 227–247, 2019. [13]
- [BL17] Daniel J. Bernstein and Tanja Lange. Post-quantum cryptography. *Nat.*, 549(7671):188–194, 2017. [2]
- [BLP11] Daniel J. Bernstein, Tanja Lange, and Christiane Peters. Smaller decoding exponents: Ball-collision decoding. In *CRYPTO*, volume 6841, pages 743–760. Springer, 2011. [57]
- [BLR⁺18] Shi Bai, Tancrede Lepoint, Adeline Roux-Langlois, Amin Sakzad, Damien Stehlé, and Ron Steinfeld. Improved security proofs in lattice-based cryptography: Using the rényi divergence rather than the statistical distance. *J. Cryptology*, 31(2):610–640, 2018. [31]
- [BPR12] Abhishek Banerjee, Chris Peikert, and Alon Rosen. Pseudorandom functions and lattices. In *EUROCRYPT 2012*, volume 7237, pages 719–737. Springer, 2012. [8, 10, 86]
- [BR93] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In Dorothy E. Denning, Raymond Pyle, Ravi Ganesan, Ravi S. Sandhu, and Victoria Ashby, editors, *CCS '93, Proceedings of*

- the 1st ACM Conference on Computer and Communications Security, Fairfax, Virginia, USA, November 3-5, 1993*, pages 62–73. ACM, 1993. [15]
- [BS08] Bhaskar Biswas and Nicolas Sendrier. McEliece cryptosystem implementation: Theory and practice. In *Post-Quantum Cryptography, Second International Workshop, PQCrypto 2008, Cincinnati, OH, USA, October 17-19, 2008, Proceedings*, pages 47–62, 2008. [42]
- [BS16] Jean-François Biasse and Fang Song. Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields. In *SODA*, pages 893–902, 2016. [16]
- [BV11] Zvika Brakerski and Vinod Vaikuntanathan. Efficient fully homomorphic encryption from (standard) LWE. In *FOCS*, pages 97–106, 2011. [10, 86]
- [CC98] Anne Canteaut and Florent Chabaud. A new algorithm for finding minimum-weight words in a linear code: Application to McEliece’s cryptosystem and to Narrow-Sense BCH codes of length 511. *IEEE Trans. Information Theory*, 44(1):367–378, 1998. [57]
- [CCJ⁺16] Lily Chen, Lily Chen, Stephen Jordan, Yi-Kai Liu, Dustin Moody, Rene Peralta, Ray Perlner, and Daniel Smith-Tone. *Report on post-quantum cryptography*, volume 12. US Department of Commerce, National Institute of Standards and Technology, 2016. [1]
- [CDPR16] Ronald Cramer, Léo Ducas, Chris Peikert, and Oded Regev. Recovering short generators of principal ideals in cyclotomic rings. In *EUROCRYPT*, pages 559–585, 2016. [16]
- [CDW17] Ronald Cramer, Léo Ducas, and Benjamin Wesolowski. Short stickelberger class relations and application to ideal-svp. In *EUROCRYPT*, pages 324–348, 2017. [16]

- [Che51] Claude Chevalley. *Introduction to the theory of algebraic functions of one variable*. Number 6. American Mathematical Soc., 1951. [38]
- [Che13] Yuanmi Chen. *Réduction de réseau et sécurité concrète du chiffrement complètement homomorphe*. PhD thesis, Paris 7, 2013. [33]
- [CHR⁺16] Ming-Shing Chen, Andreas Hülsing, Joost Rijneveld, Simona Samardjiska, and Peter Schwabe. From 5-pass MQ -based identification to MQ -based signatures. In *ASIACRYPT*, pages 135–165, 2016. [14]
- [CLM⁺18] Wouter Castryck, Tanja Lange, Chloe Martindale, Lorenz Panny, and Joost Renes. CSIDH: an efficient post-quantum commutative group action. In *ASIACRYPT*, pages 395–427, 2018. [13]
- [Cou06] Jean Marc Couveignes. Hard homogeneous spaces. *IACR Cryptol. ePrint Arch.*, 2006:291, 2006. [12, 13]
- [CR84] Benny Chor and Ronald L. Rivest. A knapsack type public key cryptosystem based on arithmetic in finite fields. In *Advances in Cryptology, Proceedings of CRYPTO '84, Santa Barbara, California, USA, August 19-22, 1984, Proceedings*, pages 54–65, 1984. [64, 79]
- [CR88] Benny Chor and Ronald L. Rivest. A knapsack-type public key cryptosystem based on arithmetic in finite fields. *IEEE Trans. Information Theory*, 34(5):901–909, 1988. [79]
- [CVW18] Yilei Chen, Vinod Vaikuntanathan, and Hoeteck Wee. GGH15 beyond permutation branching programs: Proofs, attacks, and candidates. In *CRYPTO*, pages 577–607, 2018. [10]
- [DDLL13] Léo Ducas, Alain Durmus, Tancrede Lepoint, and Vadim Lyubashevsky. Lat-

- tice signatures and bimodal gaussians. In *CRYPTO*, volume 8042, pages 40–56. Springer, 2013. [9]
- [DGK⁺10] Yevgeniy Dodis, Shafi Goldwasser, Yael Tauman Kalai, Chris Peikert, and Vinod Vaikuntanathan. Public-key encryption schemes with auxiliary inputs. In *TCC 2010*, pages 361–381, 2010. [22, 85, 89, 96]
- [DH76] Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE Trans. Information Theory*, 22(6):644–654, 1976. [9]
- [DKL09] Yevgeniy Dodis, Yael Tauman Kalai, and Shachar Lovett. On cryptography with auxiliary input. In *STOC 2009*, pages 621–630. ACM, 2009. [109]
- [DORS08] Yevgeniy Dodis, Rafail Ostrovsky, Leonid Reyzin, and Adam D. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM J. Comput.*, 38(1):97–139, 2008. [30]
- [DP12] Ivan Damgård and Sunoo Park. Is public-key encryption based on LPN practical? *IACR Cryptology ePrint Archive*, 2012:699, 2012. [5]
- [DP19] Léo Ducas and Cécile Pierrot. Polynomial time bounded distance decoding near minkowski’s bound in discrete logarithm lattices. *Des. Codes Cryptogr.*, 87(8):1737–1748, 2019. [84]
- [DPW19] Léo Ducas, Maxime Plançon, and Benjamin Wesolowski. On the shortness of vectors to be found by the ideal-svp quantum algorithm. In *CRYPTO*, pages 322–351, 2019. [16]
- [DS05] Jintai Ding and Dieter Schmidt. Rainbow, a new multivariable polynomial signature scheme. In *ACNS*, pages 164–175, 2005. [14]

- [Dum91] Ilya Dumer. On minimum distance decoding of linear codes. In *Proc. 5th Joint Soviet-Swedish Int. Workshop Inform. Theory*, pages 50–52, 1991. [57]
- [EM20] Andre Esser and Alexander May. Low weight discrete logarithm and subset sum in $2^{0.65n}$ with polynomial memory. In *EUROCRYPT 2020*, volume 12107, pages 94–122. Springer, 2020. [10]
- [FG19] Luca De Feo and Steven D. Galbraith. Seasign: Compact isogeny signatures from class group actions. In *EUROCRYPT*, pages 759–789, 2019. [13]
- [FGO⁺13] Jean-Charles Faugère, Valérie Gauthier-Umaña, Ayoub Otmani, Ludovic Perret, and Jean-Pierre Tillich. A distinguisher for high-rate McEliece cryptosystems. *IEEE Trans. Information Theory*, 59(10):6830–6844, 2013. [59]
- [FMV16] Sebastian Faust, Daniel Masny, and Daniele Venturi. Chosen-ciphertext security from subset sum. In *PKC 2016*, pages 35–46, 2016. [11, 86]
- [FO99] Eiichiro Fujisaki and Tatsuaki Okamoto. How to enhance the security of public-key encryption at minimum cost. In *Public Key Cryptography, Second International Workshop on Practice and Theory in Public Key Cryptography, PKC '99, Kamakura, Japan, March 1-3, 1999, Proceedings*, volume 1560 of *Lecture Notes in Computer Science*, pages 53–68. Springer, 1999. [15]
- [FOP⁺16a] Jean-Charles Faugère, Ayoub Otmani, Ludovic Perret, Frédéric de Portzamparc, and Jean-Pierre Tillich. Folding alternant and Goppa codes with non-trivial automorphism groups. *IEEE Trans. Information Theory*, 62(1):184–198, 2016. [17, 18, 41, 44, 45, 52]
- [FOP⁺16b] Jean-Charles Faugère, Ayoub Otmani, Ludovic Perret, Frédéric de Portzamparc, and Jean-Pierre Tillich. Structural cryptanalysis of McEliece schemes with compact keys. *Des. Codes Cryptography*, 79(1):87–112, 2016. [18, 52, 58]

- [FOPT10] Jean-Charles Faugère, Ayoub Otmani, Ludovic Perret, and Jean-Pierre Tillich. Algebraic cryptanalysis of McEliece variants with compact keys. In *EUROCRYPT*, pages 279–298, 2010. [18, 52, 58]
- [FP05] Abraham Flaxman and Bartosz Przydatek. Solving medium-density subset sum problems in expected polynomial time. In *STACS 2005*, pages 305–314, 2005. [10]
- [FPdP14] Jean-Charles Faugère, Ludovic Perret, and Frédéric de Portzamparc. Algebraic attack against variants of McEliece with Goppa polynomial of a special form. In *ASIACRYPT (1)*, pages 21–41. Springer, 2014. [18, 52]
- [Fri86] Alan M. Frieze. On the lagarias-odlyzko algorithm for the subset sum problem. *SIAM J. Comput.*, 15(2):536–539, 1986. [10]
- [FS86] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *CRYPTO '86*, volume 263, pages 186–194. Springer, 1986. [9]
- [FS09] Matthieu Finiasz and Nicolas Sendrier. Security bounds for the design of code-based cryptosystems. In *ASIACRYPT*, volume 5912, pages 88–105. Springer, 2009. [57]
- [Gam84] Taher El Gamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In *CRYPTO '84*, pages 10–18, 1984. [1]
- [Gen09] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *STOC*, pages 169–178, 2009. [10]
- [GGH+13] Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. *SIAM J. Comput.*, 45(3):882–929, 2016. Preliminary version in FOCS 2013. [10]

- [GGH97] Oded Goldreich, Shafi Goldwasser, and Shai Halevi. Public-key cryptosystems from lattice reduction problems. In *CRYPTO '97*, pages 112–131, 1997. [9, 20, 48, 49, 81, 83]
- [GGH13] Sanjam Garg, Craig Gentry, and Shai Halevi. Candidate multilinear maps from ideal lattices. In *EUROCRYPT*, pages 1–17, 2013. [10]
- [GGH15] Craig Gentry, Sergey Gorbunov, and Shai Halevi. Graph-induced multilinear maps from lattices. In *TCC*, volume 9015, pages 498–527. Springer, 2015. [10]
- [GGM84] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions (extended abstract). In *FOCS*, pages 464–479. IEEE Computer Society, 1984. [113]
- [GJ79] M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979. [14]
- [GJS16] Qian Guo, Thomas Johansson, and Paul Stankovski. A key recovery attack on MDPC with CCA security using decoding errors. In *ASIACRYPT*, volume 10031, pages 789–815, 2016. [4]
- [GKPV10] Shafi Goldwasser, Yael Tauman Kalai, Chris Peikert, and Vinod Vaikuntanathan. Robustness of the learning with errors assumption. In *ICS*, pages 230–240, 2010. [25, 31, 92]
- [GL89] Oded Goldreich and Leonid A. Levin. A hard-core predicate for all one-way functions. In *STOC 1989*, pages 25–32, 1989. [96, 97]
- [GM84] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *J. Comput. Syst. Sci.*, 28(2):270–299, 1984. [46]

- [GN08] Nicolas Gama and Phong Q. Nguyen. Predicting lattice reduction. In *Advances in Cryptology - EUROCRYPT 2008, 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Istanbul, Turkey, April 13-17, 2008. Proceedings*, pages 31–51, 2008. [77]
- [GNR10] Nicolas Gama, Phong Q. Nguyen, and Oded Regev. Lattice enumeration using extreme pruning. In *Advances in Cryptology - EUROCRYPT 2010, 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, French Riviera, May 30 - June 3, 2010. Proceedings*, pages 257–278, 2010. [34]
- [GPS17] Steven D. Galbraith, Christophe Petit, and Javier Silva. Identification protocols and signature schemes based on supersingular isogeny problems. In *ASIACRYPT*, pages 3–33, 2017. [13]
- [GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *STOC*, pages 197–206, 2008. [7, 9, 10, 20]
- [GVW13] Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. Attribute-based encryption for circuits. *J. ACM*, 62(6):45:1–45:33, 2015. Preliminary version in STOC 2013. [10]
- [HBG⁺18] Andreas Hülsing, Denis Butin, Stefan-Lukas Gazdag, Joost Rijneveld, and Aziz Mohaisen. XMSS: extended merkle signature scheme. *RFC*, 8391:1–74, 2018. [12]
- [HHK17] Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz. A modular analysis of the fujisaki-okamoto transformation. In *Theory of Cryptography - 15th International Conference, TCC 2017, Baltimore, MD, USA, November 12-15, 2017, Proceedings, Part I*, pages 341–371, 2017. [3, 15, 49, 81]

- [HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudo-random generator from any one-way function. *SIAM J. Comput.*, 28(4):1364–1396, 1999. [30]
- [HKL⁺12] Stefan Heyse, Eike Kiltz, Vadim Lyubashevsky, Christof Paar, and Krzysztof Pietrzak. Lapin: An efficient authentication protocol based on ring-lpn. In *FSE*, volume 7549, pages 346–365. Springer, 2012. [5]
- [HLY] Jingnan He, Zhe Li, and Sze Ling Yeo. A variant of subset sum problem with provably secure cryptographic applications. In submission. [16]
- [Hoe63] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963. [31]
- [HPS98] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. NTRU: A ring-based public key cryptosystem. In *ANTS-III*, pages 267–288, 1998. [9]
- [HPS11] Guillaume Hanrot, Xavier Pujol, and Damien Stehlé. Algorithms for the shortest and closest lattice vector problems. In *Coding and Cryptology - Third International Workshop, IWCC 2011, Qingdao, China, May 30-June 3, 2011. Proceedings*, pages 159–190, 2011. [33, 34]
- [IM85] Hideki Imai and Tsutomu Matsumoto. Algebraic methods for constructing asymmetric cryptosystems. In *Algebraic Algorithms and Error-Correcting Codes, 3rd International Conference, AAECC-3, Grenoble, France, July 15-19, 1985, Proceedings*, volume 229, pages 108–119. Springer, 1985. [14]
- [IN96] Russell Impagliazzo and Moni Naor. Efficient cryptographic schemes provably as secure as subset sum. *J. Cryptology*, 9(4):199–216, 1996. [10, 11, 45]
- [JAC⁺] David Jao, Reza Azarderakhsh, Matthew Campagna, Craig Costello, Luca De Feo, Basil Hess, Amir Jalali, Brian Koziel, Brian LaMacchia, Patrick Longa,

- Michael Naehrig, Joost Renes, Vladimir Soukharev, David Urbanik, and Geovandro Pereira. SIKE. <https://sike.org/>. [13]
- [JF11] David Jao and Luca De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In *PQCrypto*, pages 19–34, 2011. [13]
- [JM96] Heeralal Janwa and Oscar Moreno. McEliece public key cryptosystems using algebraic-geometric codes. *Des. Codes Cryptography*, 8(3):293–307, 1996. [4]
- [JP16] Antoine Joux and Cécile Pierrot. Technical history of discrete logarithms in small characteristic finite fields - the road from subexponential to quasi-polynomial complexity. *Des. Codes Cryptography*, 78(1):73–85, 2016. [70]
- [Kan83] Ravi Kannan. Improved algorithms for integer programming and related lattice problems. In *Proceedings of the 15th Annual ACM Symposium on Theory of Computing, 25-27 April, 1983, Boston, Massachusetts, USA*, pages 193–206, 1983. [34]
- [KF15] Paul Kirchner and Pierre-Alain Fouque. An improved BKW algorithm for LWE with applications to cryptography and lattices. In *CRYPTO*, pages 43–62, 2015. [10]
- [KI01] Kazukuni Kobara and Hideki Imai. Semantically secure McEliece public-key cryptosystems-conversions for McEliece PKC. In *PKC*, pages 19–35, 2001. [3, 49, 81, 82, 83]
- [KMW16] Elena Kirshanova, Alexander May, and Friedrich Wiemer. Parallel implementation of BDD enumeration for LWE. In *Applied Cryptography and Network Security - 14th International Conference, ACNS 2016, Guildford, UK, June 19-22, 2016. Proceedings*, pages 580–591, 2016. [35]
- [KPG99] Aviad Kipnis, Jacques Patarin, and Louis Goubin. Unbalanced oil and vinegar

- signature schemes. In *EUROCRYPT '99*, volume 1592, pages 206–222. Springer, 1999. [14]
- [KSS⁺20] Veronika Kuchta, Amin Sakzad, Damien Stehlé, Ron Steinfeld, and Shifeng Sun. Measure-rewind-measure: Tighter quantum random oracle model proofs for one-way to hiding and CCA security. In *Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part III*, pages 703–728, 2020. [15]
- [Kup05] Greg Kuperberg. A subexponential-time quantum algorithm for the dihedral hidden subgroup problem. *SIAM J. Comput.*, 35(1):170–188, 2005. [13]
- [Lam79] Leslie Lamport. Constructing digital signatures from a one-way function. Technical report, 1979. [11]
- [LB88] Pil Joong Lee and Ernest F. Brickell. An observation on the security of McEliece’s public-key cryptosystem. In *Advances in Cryptology - EUROCRYPT '88, Workshop on the Theory and Application of Cryptographic Techniques, Davos, Switzerland, May 25-27, 1988, Proceedings*, pages 275–280, 1988. [57]
- [LL19] Yang Li and Hongbo Li. Improved quantum algorithm for the random subset sum problem. *CoRR*, abs/1912.09264, 2019. [10]
- [LLL82] A. K. Lenstra, H. W. Lenstra, Jr., and L. Lovász. Factoring polynomials with rational coefficients. *Math. Ann.*, 261(4):515–534, 1982. [33]
- [LLXY20] Zhe Li, San Ling, Chaoping Xing, and Sze Ling Yeo. On the bounded distance decoding problem for lattices constructed and their cryptographic applications. *IEEE Trans. Inf. Theory*, 66(4):2588–2598, 2020. [16]

- [LN97] Rudolf Lidl and Harald Niederreiter. Finite fields: Encyclopedia of mathematics and its applications. *Computers & Mathematics with Applications*, 33(7):136–136, 1997. [39, 65]
- [LN13] Mingjie Liu and Phong Q. Nguyen. Solving BDD by enumeration: An update. In *Topics in Cryptology - CT-RSA 2013 - The Cryptographers' Track at the RSA Conference 2013, San Francisco, CA, USA, February 25-March 1, 2013. Proceedings*, pages 293–309, 2013. [35]
- [LO85] J. C. Lagarias and Andrew M. Odlyzko. Solving low-density subset sum problems. *J. ACM*, 32(1):229–246, 1985. [10]
- [LPR10] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. *J. ACM*, 60(6):43:1–43:35, 2013. Preliminary version in Eurocrypt 2010. [8]
- [LPS10] Vadim Lyubashevsky, Adriana Palacio, and Gil Segev. Public-key cryptographic primitives provably as secure as subset sum. In *TCC 2010*, pages 382–400, 2010. [11, 22, 27, 45, 85, 86, 90, 96]
- [LS15] Adeline Langlois and Damien Stehlé. Worst-case to average-case reductions for module lattices. *Des. Codes Cryptogr.*, 75(3):565–599, 2015. [8]
- [LXY19] Zhe Li, Chaoping Xing, and Sze Ling Yeo. Reducing the key size of mceliece cryptosystem from automorphism-induced goppa codes via permutations. In *Public-Key Cryptography - PKC 2019 - 22nd IACR International Conference on Practice and Theory of Public-Key Cryptography, Beijing, China, April 14-17, 2019, Proceedings, Part II*, volume 11443, pages 599–617. Springer, 2019. [16, 84]
- [Lyu05] Vadim Lyubashevsky. The parity problem in the presence of noise, decoding ran-

- dom linear codes, and the subset sum problem. In *APPROX 2005*, pages 378–389, 2005. [10]
- [Lyu09] Vadim Lyubashevsky. Fiat-shamir with aborts: Applications to lattice and factoring-based signatures. In *ASIACRYPT*, volume 5912, pages 598–616. Springer, 2009. [9]
- [Lyu12] Vadim Lyubashevsky. Lattice signatures without trapdoors. In *EUROCRYPT*, volume 7237, pages 738–755. Springer, 2012. [9, 31]
- [MAB⁺19a] Carlos Aguilar Melchor, Nicolas Aragon, Slim Bettaieb, Loic Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Edoardo Persichetti, and Gilles Zémor. Hamming Quasi-Cyclic (HQC), May 2019. https://pqc-hqc.org/doc/hqc-specification_2019-04-10.pdf. [3, 5, 16]
- [MAB⁺19b] Carlos Aguilar Melchor, Nicolas Aragon, Slim Bettaieb, Loic Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Phillippe Gaborit, and Gilles Zémor. Rank Quasi-Cyclic (RQC) Second round version, Aug 2019. https://pqc-rqc.org/doc/rqc-specification_2019-08-24.pdf. [5]
- [MAB⁺20] Carlos Aguilar Melchor, Nicolas Aragon, Slim Bettaieb, Loic Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Phillippe Gaborit, and Gilles Zémor. Rank Quasi-Cyclic (RQC) Second round version, Apr 2020. https://pqc-rqc.org/doc/rqc-specification_2020-04-21.pdf. [5]
- [MB09] Rafael Misoczki and Paulo S. L. M. Barreto. Compact McEliece keys from Goppa codes. In *Selected Areas in Cryptography*, volume 5867, pages 376–392. Springer, 2009. [17, 41, 43, 52]
- [MBD⁺18] Carlos Aguilar Melchor, Olivier Blazy, Jean-Christophe Deneuville, Philippe Ga-

- borit, and Gilles Zémor. Efficient encryption from random quasi-cyclic codes. *IEEE Trans. Information Theory*, 64(5):3927–3943, 2018. [5]
- [McE78] Robert J McEliece. A Public-Key Cryptosystem Based On Algebraic Coding Theory. *Deep Space Network Progress Report*, 44:114–116, Jan 1978. [2, 4, 20, 35, 47]
- [MCF19] David A. McGrew, Michael Curcio, and Scott R. Fluhrer. Leighton-micali hash-based signatures. *RFC*, 8554:1–61, 2019. [12]
- [Mer87] Ralph C. Merkle. A digital signature based on a conventional encryption function. In *CRYPTO '87*, volume 293, pages 369–378. Springer, 1987. [12]
- [MG02] Daniele Micciancio and Shafi Goldwasser. *Complexity of Lattice Problems: a cryptographic perspective*, volume 671 of *The Kluwer International Series in Engineering and Computer Science*. Kluwer Academic Publishers, Boston, Massachusetts, March 2002. [32]
- [MH78] Ralph C. Merkle and Martin E. Hellman. Hiding information and signatures in trapdoor knapsacks. *IEEE Trans. Inf. Theory*, 24(5):525–530, 1978. [10]
- [Mic02] Daniele Micciancio. Generalized compact knapsacks, cyclic lattices, and efficient one-way functions from worst-case complexity assumptions. In *FOCS*, pages 356–365. IEEE Computer Society, 2002. [19]
- [MMT11] Alexander May, Alexander Meurer, and Enrico Thomae. Decoding random linear codes in $\tilde{O}(2^{0.054n})$. In *ASIACRYPT*, volume 7073, pages 107–124. Springer, 2011. [57]
- [MO15] Alexander May and Ilya Ozerov. On computing nearest neighbors with applications to decoding of binary linear codes. In *EUROCRYPT (1)*, volume 9056, pages 203–228. Springer, 2015. [57]

- [MS78] F.J. MacWilliams and N.J.A. Sloane. *The Theory of Error Correcting Codes*. Number v. 2, pt. 2 in Mathematical Studies. North-Holland Publishing Company, 1978. [36, 37]
- [MS07] Lorenz Minder and Amin Shokrollahi. Cryptanalysis of the sidelnikov cryptosystem. In *EUROCRYPT*, volume 4515, pages 347–360. Springer, 2007. [4]
- [MSOS92] V M. Sidelnikov and S O. Shestakov. On insecurity of cryptosystems based on generalized reed-solomon codes. 2:439–444, 01 1992. [4]
- [MTSB13] Rafael Misoczki, Jean-Pierre Tillich, Nicolas Sendrier, and Paulo S. L. M. Barreto. Mdpcc-mceliece: New McEliece variants from moderate density parity-check codes. In *ISIT*, pages 2069–2073. IEEE, 2013. [4]
- [MU05] Michael Mitzenmacher and Eli Upfal. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005. [31]
- [NDR⁺19] Hamid Nejatollahi, Nikil D. Dutt, Sandip Ray, Francesco Regazzoni, Indranil Banerjee, and Rosario Cammarota. Post-quantum lattice-based cryptography implementations: A survey. *ACM Comput. Surv.*, 51(6):129:1–129:41, 2019. [2]
- [Ngu99] Phong Q. Nguyen. Cryptanalysis of the goldreich-goldwasser-halevi cryptosystem from crypto ’97. In *CRYPTO ’99*, volume 1666, pages 288–304. Springer, 1999. [9, 20, 71]
- [Nie86] H Niederreiter. Knapsack type cryptosystems and algebraic coding theory. 15(2):159–166, 01 1986. [3, 4]
- [NIS17a] National Institute of Standards and Technology (NIST), Post-Quantum Cryptography Standardization, Nov 2017. <https://csrc.nist.gov/Projects/post-quantum-cryptography/Post-Quantum-Cryptography-Standardization>. [1, 15]

- [NIS17b] NIST round 1 submissions, 2017. <https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions>. [3, 14]
- [NIS20] Round 3 Submissions, July 2020. <https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions>. [1, 3, 5, 12, 13]
- [NR95] Moni Naor and Omer Reingold. Synthesizers and their application to the parallel construction of psuedo-random functions. In *FOCS*, pages 170–181. IEEE Computer Society, 1995. [113]
- [Odl90] Andrew M Odlyzko. The rise and fall of knapsack cryptosystems. *Cryptology and computational number theory*, 42:75–88, 1990. [10]
- [Pat96] Jacques Patarin. Hidden fields equations (HFE) and isomorphisms of polynomials (IP): two new families of asymmetric algorithms. In *EUROCRYPT '96*, volume 1070, pages 33–48. Springer, 1996. [14]
- [PCG01] Jacques Patarin, Nicolas T. Courtois, and Louis Goubin. Quartz, 128-bit long digital signatures. In *CT-RSA 2001*, volume 2020, pages 282–297. Springer, 2001. [14]
- [PCY⁺15] Albrecht Petzoldt, Ming-Shing Chen, Bo-Yin Yang, Chengdong Tao, and Jintai Ding. Design principles for hfev- based multivariate signature schemes. In *ASIACRYPT*, volume 9452, pages 311–334. Springer, 2015. [14]
- [Pei16] Chris Peikert. A decade of lattice cryptography. *Foundations and Trends in Theoretical Computer Science*, 10(4):283–424, 2016. [11]
- [Pra62] Eugene Prange. The use of information sets in decoding cyclic codes. *IRE Trans. Information Theory*, 8(5):5–9, 1962. [57]

- [PW08] Chris Peikert and Brent Waters. Lossy trapdoor functions and their applications. volume 40, pages 1803–1844, 2011. Preliminary version in STOC 2008. [10]
- [Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6):34:1–34:40, 2009. Preliminary version in STOC 2005. [6, 11, 21]
- [RS06] Alexander Rostovtsev and Anton Stolbunov. Public-key cryptosystem based on isogenies. *IACR Cryptol. ePrint Arch.*, 2006:145, 2006. [12]
- [RSA78] Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21(2):120–126, 1978. [1]
- [RSSS17] Miruna Rosca, Amin Sakzad, Damien Stehlé, and Ron Steinfeld. Middle-product learning with errors. In *CRYPTO*, volume 10403, pages 283–297. Springer, 2017. [8]
- [RW04] Renato Renner and Stefan Wolf. Smooth renyi entropy and applications. In *ISIT*, page 233. IEEE, 2004. [29]
- [Sch03] Claus-Peter Schnorr. Lattice reduction by random sampling and birthday methods. In *STACS*, volume 2607, pages 145–156. Springer, 2003. [33]
- [SE94] C. P. Schnorr and M. Euchner. Lattice basis reduction: Improved practical algorithms and solving subset sum problems. *Math. Program.*, 66(2):181–199, September 1994. [33]
- [Sen00] Nicolas Sendrier. Finding the permutation between equivalent linear codes: The support splitting algorithm. *IEEE Trans. Information Theory*, 46(4):1193–1203, 2000. [57]

- [Sha08] Andrew Shallue. An improved multi-set algorithm for the dense subset sum problem. In *ANTS-VIII*, pages 416–429, 2008. [10]
- [Sho94] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.*, 26(5):1484–1509, 1997. Preliminary version in FOCS 1994. [1, 47]
- [Sid94] Vladimir Michilovich Sidelnikov. A public-key cryptosystem based on binary Reed-Muller codes. *Discrete Mathematics and Applications*, 4(3):191–208, 1994. [4]
- [Ste88] Jacques Stern. A method for finding codewords of small weight. In *Coding Theory and Applications, 3rd International Colloquium, Toulon, France, November 2-4, 1988, Proceedings*, pages 106–113, 1988. [57]
- [Sto04] Anton Stolbunov. *Public-key encryption based on cycles of isogenous elliptic curves*. PhD thesis, Master’s thesis, Saint-Petersburg State Polytechnical University, 2004., 2004. [12, 13]
- [SW14] Amit Sahai and Brent Waters. How to use indistinguishability obfuscation: deniable encryption, and more. In *STOC*, pages 475–484, 2014. [10]
- [Tor17] Rodolfo Canto Torres. CaWoF, C library for computing asymptotic exponents of generic decoding work factors, Jan 2017. <https://gforge.inria.fr/projects/cawof/>. [61]
- [vEH14] Tim van Erven and Peter Harremoës. Rényi divergence and kullback-leibler divergence. *IEEE Trans. Inf. Theory*, 60(7):3797–3820, 2014. [31]
- [YAJ⁺17] Youngho Yoo, Reza Azarderakhsh, Amir Jalali, David Jao, and Vladimir Soukharev. A post-quantum digital signature scheme based on supersingular isogenies. In *FC*, pages 163–181, 2017. [13]

- [Yao82] Andrew Chi-Chih Yao. Theory and applications of trapdoor functions (extended abstract). In *FOCS 1982*, pages 80–91, 1982. [89, 96, 97]