耐量子計算機暗号の研究動向調査報告書

CRYPTREC 暗号技術調査 WG (暗号解析評価)

2019 年 3 月

目次

第 1 章

はじめに

暗号技術はインターネットをはじめ, 現代の情報通信システムのセキュリティを支える基盤技術であり, ネットショッピングやインターネットバンキング, 交通系 IC カード, 無線 LAN などで日常的に利用されている. 現在広く使用されている公開鍵暗号方式として RSA 暗号と楕円曲線暗号が挙げられる. RSA 暗号の安全性は整数を素因数分解する計算の困難性を根拠としており, 楕円曲線暗号については離散対数問題を解く計算の困難性が利用されている. 量子計算機の開発が十分に進むと Shor のアルゴリズム [14, 15] により整数の素因数分解や離散対数を高速に計算できるため, それらの暗号方式の安全性が大きく低下する. そのため, 量子計算機に対しても, また現在使用されている計算機に対しても, 安全性を確保できる暗号方式が必要とされており, そのような暗号方式は耐量子計算機暗号 (Post-Quantum Cryptography: PQC) とよばれている.

近年, 量子計算機の開発が世界的に進められている. それにともなって, PQC に関する研究及びその標準化に向けた活動も世界各国の組織で実施されており, 国内でも PQC の研究動向を把握する必要性が高まっている. 2017 年度の暗号技術検討会において, 暗号技術評価委員会の活動計画として 2 年をかけて PQC の研究動向を調査することが決定された. 暗号技術評価委員会は暗号技術調査ワーキンググループ (暗号解析評価) を設置し, 2017 年度及び 2018 年度において本調査を実施した (図 1.1).

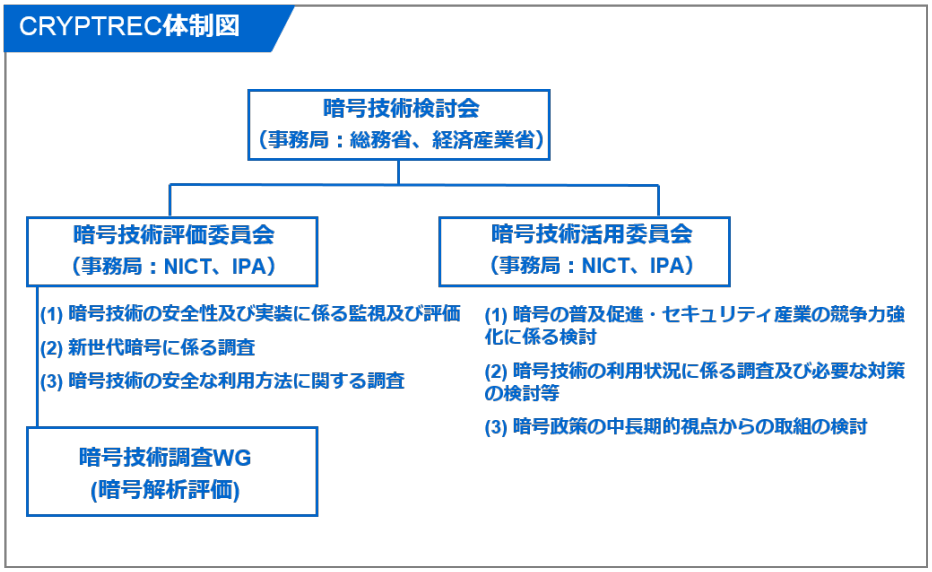

図 1.1　CRYPTREC の体制図 (https://www.cryptrec.go.jp/system.html)

本調査では PQC の代表的な候補である四つの分類 (格子に基づく暗号技術, 符号に基づく暗号技術, 多変数多項式に基づく暗号技術, 同種写像に基づく暗号技術) について調査し, 主に 2018 年 10 月 12 日までの調査結果を本報告書にまとめた. また, 本報告書の 1 章は一般的な読者にむけた内容としてまとめ, それ以外の章については暗号技術に携わる研究者及び技術者を読者として想定した内容とした. ハッシュ関数に基づく署名技術も PQC の主要な候補の一つであり, それについては [1, 9] を参照されたい.

1.1 耐量子計算機暗号 (PQC) の必要性について

量子計算機には, 量子ゲート型の量子計算機と量子アニーリング計算専用のもの等があり, Shor のアルゴリズムが直接適用できるのは量子ゲート型の量子計算機である. 量子ゲート型の場合のプロセッサに関する近年の開発については, 2017 年 11 月に IBM が 50 量子ビットのプロセッサを [17], 2018 年 5 月に Google が 72 量子ビットのプロセッサを開発したと発表しており [11], 量子ゲート型の量子計算機の開発が進んでいることを示す報告がいくつかある. また, NIST の Interim Report 8105 において, 2000 bit の合成数を利用する RSA 暗号を解読可能な量子計算機が 2030 年までに実現される可能性があると, 量子計算機の研究者らが見積もっているとの記述がある [3].

しかし, 文献 [10, 12, 16] で述べられているように, 量子計算機と Shor のアルゴリズムを利用した数値実験において, 大きな整数の素因数分解はまだ成功していない (表 1.1). 例えば, 量子計算機と Shor のアルゴリズムを使用して素因数分解できた現時点での最大の合成数は 21 とされており, 2012 年の Martín-López らによる成果が挙げられる [13]. さらに, 下記の理由により, この数値実験における実装は Shor のアルゴリズムの実装として不適当である. Shor のアルゴリズムの特徴は剰余類環の元の乗法位数を効率よく計算できることであり, この性質が整数の素因数分解や離散対数問題を解く計算に利用される. しかし, この数値実験ではこの位数の計算を行わず, 回路設計の段階で 21 を法としたときの 4 の乗法位数が 3 であることが利用されている. 位数が未知の場合の成果として, 2001 年に Vandersypen らによる 15 の素因数分解の成果 [18] が挙げられるが, この実験でも, ターゲットとなる合成数の特徴を回路設計の段階で過剰に組み込まれており, 汎用的な回路構成となっていない. 従って, 現時点では Shor のアルゴリズムを汎用的な回路構成により量子計算機上で実装して, 素因数分解に成功した成果は知られていない.

分解法	計算手段	合成数	ビット長	発表年	ジャーナル名	備考
Shor	NMR	15	4	2001 年	Nature	# QB=7
	光子	21	5	2012 年	Nature Photonics	# QB=1+log 3
	光集積回路	15	4	2009 年	Science	# QB=5
	ジョセフソン素子	15	4	2012 年	Nature Physics	# QB=3
	イオントラップ	15	4	2016 年	Science	# QB=5

・本分解法は分解対象の素因数分解の結果を用い，Shor のアルゴリズムの演算の一部を省略している.
・# QB は使用した量子ビット数.
・2 行目の # QB の値は 2 進数表現と 3 進数表現から算出されている.

表 1.1 量子ゲート型計算機による素因数分解の記録 ([10] の表 1 を一部改変)

一方で, 量子アニーリング計算専用の量子計算機については, 2017 年 1 月にカナダの D-Wave 社から 2048 量子ビットのプロセッサを開発したとの発表があり, 発売された [5, 6]. 量子アニーリング計算専用の量子計算機において Shor アルゴリズムを効率よく適用する方法は発見されておらず, 二次多変数多項式を利用したアルゴリズムが使用されてい

る. 現時点で, この計算手法によって素因数分解された最大の合成数は 200099 であり, 2016 年の Dridi と Alghassi の数値実験によるものである [7] (表 1.2).

分解法	計算手段	合成数	ビット長	発表年	ジャーナル名	備考
素朴法	NMR	21	5	2008 年	Psysical Review Letters	# QB=3
筆算法	NMR	143	8	2012 年	Psysical Review Letters	# QB=4
	D-Wave	200099	18	2016 年	Scientific Reports	# QB=897

・# QB は使用した量子ビット数.

表 1.2　量子アニーリング計算専用の量子計算機による素因数分解の記録 ([10] の表 1 を一部改変)

以上のことから, RSA 暗号に対する量子計算機を使用した場合の実際的な脅威が差し迫っているとは現時点では断言できない. しかし, 現在広く使用されている公開鍵暗号の安全性は, 整数の素因数分解や離散対数の計算困難性に基づいているため, 十分な性能を持つ量子計算機の開発が進んだ場合のリスクは極めて大きい. また, 使われている暗号システムを更新するためには, その準備として長い年月が必要である. 従って, RSA 暗号や楕円曲線暗号から PQC への移行が必要となった場合に, すぐに PQC を使用できるように十分な時間をかけて前もって準備をしておくことが望ましい. 以上の理由から, 現在, 世界各国において PQC に関する研究開発や標準化が進められている.

1.2　PQC の研究及び標準化等に関する動向

PQC に関する研究成果は Crypto, Eurocrypt, Asiacrypt 等, 暗号分野の国際会議で長年議論されている. さらに PQC を専門に扱う国際会議として PQCrypto が挙げられ, その第 1 回会議は 2006 年に開催され, 2018 年までに計 9 回開催されている.

PQC の標準化に関する近年の動向については, まず 2015 年 8 月, アメリカ国家安全保障局 (NSA) が PQC への将来的な移行計画を発表している. また, アメリカ国立標準技術研究所 (NIST) は 2016 年から PQC の公募を開始し, その締切である 2017 年 11 月 30 日までに 82 件の暗号方式が提案され, 提出書類が完備であることが確認された 69 件が Round 1 の評価対象となった. また, 2019 年 1 月 30 日には, NIST から Round 2 へ進む方式として 26 件が発表された. 現在は, 提案されたそれらの暗号方式の安全性を評価する段階にあり, 2023 年前後までに標準化を進める計画を立てている [4] (図 1.2). 欧州では EU H2020 において SAFECrypto [*1] や PQCRYPTO [*2] などの研究プロジェクトが実施されたほか, ETSI から Quantum-safe Cryptography に関するガイドラインが出版されている [8]. また, ISO/IEC JTC 1/SC 27 WG2 においても PQC に関する Standing Document の作成が始まるなど, 標準化に向けた議論が始まっている.

暗号技術調査ワーキンググループにおいても 2014 年度に PQC の代表的な候補である格子に基づく暗号技術について調査を行い, 報告書「格子問題等の困難性に関する調査」を公開している [2]. さらに 2017 年度から 2018 年度にかけて, PQC の代表的な候補である前述の四つの分類について調査し, 本報告書にまとめた.

[*1] https://www.safecrypto.eu/
[*2] https://pqcrypto.eu.org/

Feb 24-26, 2016	NIST Presentation at PQCrypto 2016: *Announcement and outline of NIST's Call for Submissions (Fall 2016)*, *Dustin Moody*
April 28, 2016	NIST releases NISTIR 8105, Report on Post-Quantum Cryptography
Dec 20, 2016	Formal Call for Proposals
Nov 30, 2017	Deadline for submissions
Dec 4, 2017	NIST Presentation at AsiaCrypt 2017: *The Ship Has Sailed: The NIST Post-Quantum Crypto "Competition"*, *Dustin Moody*
Dec 21, 2017	Round 1 algorithms announced (69 submissions accepted as "complete and proper")
Apr 11, 2018	NIST Presentation at PQCrypto 2018: *Let's Get Ready to Rumble - The NIST PQC "Competition"*, *Dustin Moody*
April 11-13, 2018	First PQC Standardization Conference - Submitter's Presentations
January 30, 2019	Second Round Candidates announced
August 22-24, 2019 *(tentative)*	Second PQC Standardization Conference
2020/2021	Round 3 begins or select algorithms
2022/2024	Draft Standards Available

図 1.2　NIST PQC 標準化スケジュール (`https://csrc.nist.gov/projects/post-quantum-cryptography/workshops-and-timeline`)

1.3　本報告書で扱う四つの PQC

本節では本調査の対象として格子に基づく暗号技術, 符号に基づく暗号技術, 多変数多項式に基づく暗号, 同種写像に基づく暗号技術を選択した理由及びその説明に必要な事項について述べる.

代表的な公開鍵暗号は, 数学的な計算問題の困難性をその安全性の根拠として利用している. 例えば RSA 暗号では, 二つの同程度の大きさでかつ相異なる素数 p, q が秘密鍵, それらの積 $N = pq$ が公開鍵として使用され, N が素因数分解されると秘密鍵 p, q が計算されてしまい, RSA 暗号は解読されてしまう. 楕円曲線暗号の場合も楕円曲線暗号の公開鍵から楕円曲線上の離散対数問題が定義され, それを解くことでその秘密鍵が計算できてしまう. 本報告書で扱う代表的な四つの PQC (格子に基づく暗号技術, 符号に基づく暗号技術, 多変数多項式に基づく暗号, 同種写像に基づく暗号技術) も RSA 暗号と同様に, それらの安全性はそれぞれで利用される数学的な計算問題の困難性を根拠としている. そして, これらの問題を量子計算機を利用して効率よく解くアルゴリズムはまだ発見されていないことが, それら四つの暗号方式が PQC とされている理由である. 本調査の対象である暗号方式と数学的な計算問題の関係は各章の 1 節で説明する.

本報告書の調査対象である四つの暗号技術の研究の歴史は長く, 格子に基づく暗号技術及び同種写像に基づく暗号技術 20 年以上, 多変数多項式に基づく暗号技術は 30 年以上, 符号に基づく暗号技術は 40 年以上研究が行われている. 従って, 本調査ではこれらの暗号技術を有力な PQC として調査した. 上述の各暗号方式の歴史的な事実については各章の 4 節に記載した.

1.4　2017 〜 2018 年度 暗号技術調査 WG (暗号解析評価) の委員構成

主査	高木 剛	東京大学大学院 情報理工学系研究科 教授
委員	青木 和麻呂	日本電信電話株式会社 NTT セキュアプラットフォーム研究所 グループリーダ
委員	草川 恵太	日本電信電話株式会社 NTT セキュアプラットフォーム研究所 研究主任
委員	國廣 昇	東京大学大学院 新領域創成科学研究科 准教授
委員	下山 武司	株式会社富士通研究所 セキュリティ研究所 データ & IoT セキュリティプロジェクト 主管研究員
委員	高島 克幸	三菱電機株式会社 情報技術総合研究所 主管技師長
委員	安田 貴徳	岡山理科大学 工学部 准教授
委員	安田 雅哉	九州大学　マス・フォア・インダストリ研究所 准教授

1.5　各章の執筆者と内容

各章の執筆担当者及び調査内容は下表の通りである.

章	執筆者名	内容
第 1 章	高木 剛 主査, 黒川 貴司 (事務局), 篠原 直行 (事務局), 盛合 志帆 (事務局)	調査の目的, 概要など
第 2 章	下山 武司 委員, 安田 雅哉 委員, 青野 良範 (事務局)	格子に基づく暗号技術
第 3 章	草川 恵太 委員	符号に基づく暗号技術
第 4 章	安田 貴徳 委員	多変数多項式に基づく暗号技術
第 5 章	高島 克幸 委員	同種写像に基づく暗号技術

第 1 章の参考文献

[1] J. Buchmann, E. Dahmen, M. Szydlo, Hash-based Digital Signature Schemes, *Post-Quantum Cryptography*, pp. 35-93. Springer, Heidelberg, 2009.

[2] CRYPTREC, 格子問題等の困難性に関する調査, *CRYPTREC report 2014*, https://www.cryptrec.go.jp/exreport/cryptrec-ex-2404-2014.pdf, 2015.

[3] L. Chen, S. Jordan, Y.-K. Liu, D. Moody, R. Peralta, R. Perlner, D. Smith-Tone, Report on Post-Quantum Cryptography, *NISTIR 8105*, https://nvlpubs.nist.gov/nistpubs/ir/2016/NIST.IR.8105.pdf, 2016.

[4] L. Chen, D. Moody, Y.-K. Liu, Post-Quantum Cryptography, https://csrc.nist.gov/projects/post-quantum-cryptography/workshops-and-timeline, 2018.

[5] D-Wave Systems Inc., D-Wave Announces D-Wave 2000Q Quantum Computer and First System Order, https://www.dwavesys.com/press-releases/d-wave%C2%A0announces%C2%A0d-wave-2000q-quantum-computer-and-first-system-order, 2017.

[6] D-Wave Systems Inc., The D-Wave 2000QTM Quantum Computer Technology Overview, https://www.dwavesys.com/sites/default/files/D-Wave%202000Q%20Tech%20Collateral_0117F.pdf, 2017.

[7] R. Dridi, H. Alghassi, Prime Factorization Using Quantum Annealing and Computational Algebraic Geometry, https://arxiv.org/abs/1604.057962v2, 2016.

[8] ETSI, Quantum-Safe Cryptography, https://www.etsi.org/technologies-clusters/technologies/quantum-safe-cryptography.

[9] A. Huelsing, D. Butin, S. Gazdag, J. Rijneveld, A. Mohaisen, XMSS: eXtended Merkle Signature Scheme, *RFC8391*, https://tools.ietf.org/html/rfc8391, 2018.

[10] 伊豆 哲也, 國廣 昇, 素因数分解の現状, *Proceedings of the 2018 Symposium on Cryptography and Information Security*, SCIS 2018, 2B2-3, 2018.

[11] J. Kelly, A Preview of Bristlecone, Google's New Quantum Processor, https://ai.googleblog.com/2018/03/a-preview-of-bristlecone-googles-new.html, 2018.

[12] 國廣 昇, Shor のアルゴリズムに基づく素因数分解実験の調査, 電子情報通信学会技術研究報告, volume 76, pp. 58-62, IEICE, 2018.

[13] E. Martín-López, A. Laing, T. Lawson, R. Alvarez, X.-Q. Zhou, J.L. O'Brien, Experimental Realization of Shor's Quantum Factoring Algorithm Using Qubit Recycling, *Nature Photonics*, volume 6, number 11, pp. 773-776, 2012.

[14] P. W. Shor, Polynominal time algorithms for discrete logarithms and factoring on a quantum computer, *Proceedings of Algorithmic Number Theory, First International Symposium* , pp. 289-289, 1994.

[15] P. W. Shor, Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum

Computer, *SIAM J. Comput.*, volume 26, number 5, pp. 1484-1509, 1997.

[16] 清水 俊也, 伊豆 哲也, 篠原 直行, 盛合 志帆, 國廣 昇, アニーリング計算による素因数分解について, *Proceedings of the 2019 Symposium on Cryptography and Information Security*, SCIS 2019, 2B4-3, 2019.

[17] C. Vu, Client systems with 20 qubits ready for use; next-generation IBM Q system in development with first working 50 qubit processor. - -IBM expands its open-source quantum software package QISKit; offers the world's most advanced ecosystem for quantum computing, `https://www-03.ibm.com/press/us/en/pressrelease/53374.wss`, 2017.

[18] L. M. K. Vandersypen, M. Steffen, G. Breyta, C. S. Yannoni, M. H. Sherwood, I. L. Chuang, Experimental Realization of Shor's Quantum Factoring Algorithm Using Nucler Magnetic Resonance, *Nature*, volume 414, pp. 883-887, 2001.

第 2 章

格子に基づく暗号技術

本章では格子に基づく暗号技術についてまとめる. 格子に基づく暗号技術の安全性は, LWE (Learning With Errors) 問題, NTRU 問題, およびそれらの変種等を含む, 格子理論に関係する問題を解く計算の困難性に依存している.

2.1 格子に基づく暗号技術の安全性の根拠となる問題

2.1.1 LWE 問題と代表的な求解法

本節では, 2005 年 Regev が提案した LWE 問題 [40] を紹介すると共に, 格子を利用した LWE 問題に対する求解法を紹介する. また, LWE 問題のいくつかの変種についても言及する.

2.1.1.1 LWE 問題の紹介

LWE 問題は機械学習理論から派生した求解困難な問題で, 整数剰余類環 $\mathbb{Z}_q$ 上の秘密ベクトル $\mathbf{s} \in \mathbb{Z}_q^n$ に関するランダムな連立線形「近似」方程式が与えられたとき, その秘密ベクトルを復元する問題である. 数値例として $n = 4,\ q = 17$ とし, 秘密ベクトル $\mathbf{s} = (s_1, s_2, s_3, s_4) \in \mathbb{Z}_{17}^4$ に関する連立線形近似方程式

$$\begin{cases} 14s_1 + 15s_2 + 5s_3 + 2s_4 \approx 8 & (\mathrm{mod}\ 17) \\ 13s_1 + 14s_2 + 14s_3 + 6s_4 \approx 16 & (\mathrm{mod}\ 17) \\ 6s_1 + 10s_2 + 13s_3 + s_4 \approx 12 & (\mathrm{mod}\ 17) \\ \vdots & \\ 6s_1 + 7s_2 + 16s_3 + 2s_4 \approx 3 & (\mathrm{mod}\ 17) \end{cases}$$

が与えられたとする. (この数値例は [43] から引用した.) ただし, 各線形方程式の値は近似値であり, その誤差はこの例では ± 1 以内と仮定する. このとき, この連立線形近似方程式の解 $\mathbf{s}$ を求めるのが LWE 問題である. (この数値例では $\mathbf{s} = (0, 13, 9, 11) \in \mathbb{Z}_{17}^4$ が解となる.) LWE 問題で注意すべきことは, 連立線形近似方程式に誤差がない場合は, Gauss の消去法により効率的に解を求めることができる点である. 逆に言うと, 連立線形近似方程式で与えられる誤差の大きさが LWE 問題の求解を困難にする. (実際, 誤差が大きくなるほど LWE 問題の求解はより困難になる.)

■離散 Gauss 分布 一般に, LWE 問題における連立線形近似方程式の誤差は, 平均 0, パラメータ $\sigma > 0$ の $\mathbb{Z}$ 上の離散 Gauss 分布 $\chi = D_{\mathbb{Z},\sigma}$ から生成される[*1]. より正確には, χ は各整数 x がサンプルされる確率が $\exp\left(-\frac{\pi x^2}{\sigma^2}\right)$ に比例

[*1] 本章では記号 σ をガウス分布のパラメータ (標準偏差とは異なる) の意味で使い, 署名を表すときには sig を用いる.

する $\mathbb{Z}$ 上の離散確率分布である. この分布は, 数学的な正規分布[*2] とは異なるが, 絶対値の大きな値が生成される確率が非常に小さいという性質は共通している. 例えば, 絶対値が 3σ より大きな整数がサンプルされる確率は非常に小さい. 離散 Gauss 分布の詳細については [33] などを参照.

離散 Gauss 分布を正確に生成するアルゴリズムは実装が容易ではなく, timing attack などの脆弱性 [11] が生まれてしまうため, 現実の方式 (2.3 節参照) においては, 誤差 (ノイズ) として離散 Gauss 分布との統計距離が小さい分布を用いている. それらを区別するため, 方式 Scheme 内で用いられるノイズの分布を $D_{\mathbb{Z},s}^{\mathsf{Scheme}}$ と表現する. ここで, s はパラメータである, また, 記号 $D_{\mathbb{Z}^n,s}^{\mathsf{Scheme}}$, $D_{\mathbb{Z}^{n\times m},s}^{\mathsf{Scheme}}$ によってそれぞれ, 成分を $D_{\mathbb{Z},s}^{\mathsf{Scheme}}$ から独立に生成した n 次元ベクトル, $n \times m$ 行列とする.

■LWE 問題の定式化

定義 2.1 (LWE 問題 [40]) n を正の整数とし, q を奇素数とする. 平均 0, 標準偏差 σ の $\mathbb{Z}$ 上の離散 Gauss 分布を $\chi = D_{\mathbb{Z},\sigma}$ とする. 秘密ベクトル $\mathbf{s} \in \mathbb{Z}_q^n$ を固定する. 一様ランダムに選ばれた $\mathbf{a} \in \mathbb{Z}_q^n$ と離散 Gauss 分布 χ からサンプルされた $e \in \mathbb{Z}$ に対して, $(\mathbf{a}, b) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ の組を出力する確率分布を $L_{\mathbf{s},\chi}$ とする. ただし, $b \equiv \langle \mathbf{a}, \mathbf{s} \rangle + e \pmod q$ とする. (2 つのベクトル $\mathbf{v}$ と $\mathbf{w}$ の内積を $\langle \mathbf{v}, \mathbf{w} \rangle$ で表す.) このとき, 次の 2 つの問題を考える :

1. 判定 LWE (Decision-LWE)　与えられた組 $(\mathbf{a}, b) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ が, 確率分布 $L_{\mathbf{s},\chi}$ からサンプルされた元か, $\mathbb{Z}_q^n \times \mathbb{Z}_q$ 上一様ランダムに生成された元かを決定する問題.
2. 探索 LWE (Search-LWE)　確率分布 $L_{\mathbf{s},\chi}$ からサンプルされた組 $(\mathbf{a}, b)$ から秘密ベクトル $\mathbf{s}$ を復元する問題.

一般に, 上記の LWE 問題において確率分布 $L_{\mathbf{s},\chi}$ は任意個の組 $(\mathbf{a}, b)$ をサンプルするオラクルとしてみなす. 具体的には, ある固定したサンプル数 $m > 0$ に対して, 確率分布 $L_{\mathbf{s},\chi}$ からサンプルされた異なる m 個の組

$$\begin{cases} (\mathbf{a}_1, b_1), & b_1 \equiv \langle \mathbf{a}_1, \mathbf{s} \rangle + e_1 \pmod q \\ (\mathbf{a}_2, b_2), & b_2 \equiv \langle \mathbf{a}_2, \mathbf{s} \rangle + e_2 \pmod q \\ \quad \vdots & \\ (\mathbf{a}_m, b_m), & b_m \equiv \langle \mathbf{a}_m, \mathbf{s} \rangle + e_m \pmod q \end{cases}$$

から LWE 問題を解くことを考える. (解読に要する計算時間が最も短くなるような m を攻撃者が選べることを想定する.) 第 i 行ベクトルを $\mathbf{a}_i$ とする $m \times n$ 行列を $\mathbf{A}$ とし, $\mathbf{b} = (b_1, b_2, \ldots, b_m)$ とおく. このとき, 上記の m 個の LWE サンプルの組は $(\mathbf{A}, \mathbf{b}) \in \mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^n$ と簡潔に表せて, 関係式 $\mathbf{b} \equiv \mathbf{s}\mathbf{A}^\top + \mathbf{e} \pmod q$ を満たす. ただし, $\mathbf{e} = (e_1, e_2, \ldots, e_m) \in \mathbb{Z}^m$ をノイズベクトルとする. (各 e_i は χ からサンプルされた元であることに注意する.)

■LWE 問題の変種　LWE 問題の変種として多項式環 $R_q = \mathbb{Z}_q[x]/(\phi)$ 上の LWE である Ring-LWE [48, 34] [*3] や Module-LWE [36] がある. Ring-LWE では, 3 つの多項式 $s, a_i, e_i \in R_q$ に対する Ring-LWE サンプルとして $\{(a_i, a_i \cdot s + e_i)\}_{i=1}^m$ を考える. (特に, 通常の LWE 問題と同じように, ランダムな s と, 係数が小さい多項式の集合からサンプリングされた e_i が用いられる.) Ring-LWE の基礎環 R_q を定める多項式として $\phi = x^n + 1$ がよく用いられる. また, Module-LWE では, 多項式ベクトル $\mathbf{s}, \mathbf{a}_i \in R_q^k$ と多項式 $e_i \in R_q$ に対する Module-LWE サンプルとして $\{(\mathbf{a}_i, \langle \mathbf{a}_i, \mathbf{s} \rangle + e_i)\}_{i=1}^m$ を考える. Module-LWE の基礎環 R_q を定める多項式としては $\phi = x^{n/k} + 1$ がよく用いられ

[*2] 分散 t^2 に対して数学的な正規分布 $N(0, t^2)$ は, 確率密度関数が $\frac{1}{\sqrt{2\pi}t} e^{-z^2/(2t^2)}$ により定義されるため, $\sqrt{2\pi}$ 倍のずれがある. 格子暗号の安全性を議論する際に格子上の Fourier 変換が用いられることが多く [40], 本文中の定義を用いることで, 数式の表現が簡潔となる.

[*3] 文献 [34] ではより一般的に整数環とイデアルを用いて定義されているが, 後の文献 [15] ではそれの簡略化として, 多項式環 R_q を用いた表現である “polynomial-LWE assumption” が提案された. 2019 年現在では後者の表現の方が Ring-LWE と呼ばれている.

る．さらに, 環上の LWE 以外の LWE 問題の変種として, 丸め込み写像でノイズベクトルを生成する LWR (Learning with Rounding) [14] や middle-product と呼ばれる多項式演算を用いる Middle-product LWE [44] など数多く提案されている．

2.1.1.2 格子の基本事項と q-ary 格子の紹介

■格子の基本事項 m 次元実ベクトル空間 $\mathbb{R}^m$ の一次独立な m 個のベクトル $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m$ の整数係数の線形結合全体 $L = \{\sum_{i=1}^{m} x_i \mathbf{b}_i : x_i \in \mathbb{Z}, 1 \leq i \leq m\}$ を (完全階数の) m 次元**格子**と呼ぶ．特に, 格子 L はベクトル空間 $\mathbb{R}^m$ の (離散) 加法部分群である．また, 格子 L を生成する一次独立な m 個のベクトルの組 $\{\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m\}$ を**基底**と呼び, 各 $\mathbf{b}_i$ を**基底ベクトル**と呼ぶ．さらに, 行ベクトルで表した基底ベクトル $\mathbf{b}_i \in \mathbb{R}^m$ を行として持つ $m \times m$ 行列 $\mathbf{B} = (\mathbf{b}_i)_{i=1}^{m}$ を格子 L の**基底行列**と呼ぶ．2 次元以上の格子を生成する異なる基底は無限に存在し, 同じ格子を生成する 2 つの基底行列 $\mathbf{B}_1$ と $\mathbf{B}_2$ に対し $\mathbf{B}_2 = \mathbf{V}\mathbf{B}_1$ を満たす $m \times m$ のユニモジュラ行列 $\mathbf{V}$ が存在する．また, 基底行列 $\mathbf{B}$ を用いて, 格子 L の**体積**を $\mathrm{vol}(L) = |\det(\mathbf{B})|$ と定める．(体積は基底の取り方に依存しない.) 格子 L の**第 1 逐次最小**は L 上の最短な非零ベクトルの Euclid ノルムを指し, $\lambda_1(L)$ と表す．ベクトル空間 $\mathbb{R}^m$ の完全階数の格子 L に対し, 集合 $\widehat{L} = \{\mathbf{x} \in \mathbb{R}^m : \langle \mathbf{x}, \mathbf{y} \rangle \in \mathbb{Z} \quad (\forall \mathbf{y} \in L)\}$ を格子 L の**双対格子**と呼ぶ．また, 格子 L の基底行列 $\mathbf{B}$ に対して, $\widehat{\mathbf{B}} = \left(\mathbf{B}^{-1}\right)^{\top}$ は双対格子 $\widehat{L}$ の基底行列となり, この $\widehat{\mathbf{B}}$ を**双対基底行列**と呼ぶ．単位行列 $\mathbf{I}_m$ に対し $\mathbf{B}\widehat{\mathbf{B}}^{\top} = \mathbf{I}_m$ を満たすので, $\mathrm{vol}(L) \times \mathrm{vol}(\widehat{L}) = 1$ が成り立つ．

■q-ary 格子 LWE 問題の求解で利用する特殊な格子を紹介する．正の整数 q に対して, $q\mathbb{Z}^m \subseteq L \subseteq \mathbb{Z}^m$ を満たす完全階数の m 次元格子 L を **q-ary 格子**と呼ぶ．2 つの自然数 $m > n$ に対し, 任意の正の整数 q と $n \times m$ 整数行列 $\mathbf{X}$ に対する 2 つの m 次元 q-ary 格子を

$$\Lambda_q(\mathbf{X}) = \{\mathbf{y} \in \mathbb{Z}^m : \exists \mathbf{s} \in \mathbb{Z}^n \text{ s.t. } \mathbf{y} \equiv \mathbf{s}\mathbf{X} \pmod{q}\}, \quad \Lambda_q^{\perp}(\mathbf{X}) = \left\{\mathbf{y} \in \mathbb{Z}^m : \mathbf{y}\mathbf{X}^{\top} \equiv \mathbf{0} \pmod{q}\right\}$$

と定義する．(これら 2 つの集合は $\mathbb{R}^m$ の離散加法部分群なので格子である.) 正規化の差を除き, これら 2 つの q-ary 格子は互いに双対の関係にある．正確には $\Lambda_q^{\perp}(\mathbf{X}) = q\widehat{\Lambda_q(\mathbf{X})}$ と $\Lambda_q(\mathbf{X}) = q\widehat{\Lambda_q^{\perp}(\mathbf{X})}$ が成り立つ．また, 群準同型写像 $f : \mathbb{Z}^m \longrightarrow (\mathbb{Z}/q\mathbb{Z})^n, \quad \mathbf{y} \longmapsto \mathbf{y}\mathbf{X}^{\top} \pmod{q}$ の核は q-ary 格子 $\Lambda_q^{\perp}(\mathbf{X})$ なので, 群の準同型定理から $\mathrm{vol}(\Lambda_q^{\perp}(\mathbf{X})) = [\mathbb{Z}^m : \Lambda_q^{\perp}(\mathbf{X})] = \#\mathrm{Im}(f)$ が成り立つ．(群の指数 $[\mathbb{Z}^m : \Lambda_q^{\perp}(\mathbf{X})]$ は格子の体積の比 $\frac{\mathrm{vol}(\Lambda_q^{\perp}(\mathbf{X}))}{\mathrm{vol}(\mathbb{Z}^m)}$ に一致することに注意する.) これより, 体積 $\mathrm{vol}\left(\Lambda_q^{\perp}(\mathbf{X})\right)$ は q^n を割る．さらに, 元の格子と双対格子の体積の関係から, q^{m-n} は体積 $\mathrm{vol}\left(\Lambda_q(\mathbf{X})\right)$ を割ることが分かる．(ただし, ほとんどの行列 $\mathbf{X}$ に対して写像 f は全射で, その時 $\mathrm{vol}\left(\Lambda_q^{\perp}(\mathbf{X})\right) = q^n$ と $\mathrm{vol}\left(\Lambda_q(\mathbf{X})\right) = q^{m-n}$ が成り立つ.) q-ary 格子 $\Lambda_q(\mathbf{X})$ 上のベクトルは $\mathbf{y} = \mathbf{s}\mathbf{X} + q\mathbf{z}$ $(\mathbf{s} \in \mathbb{Z}^n, \mathbf{z} \in \mathbb{Z}^m)$ とかけるので, その格子は $(n+m) \times m$ 整数行列 $\begin{pmatrix} \mathbf{X} \\ q\mathbf{I}_m \end{pmatrix}$ の一次従属な $(n+m)$ 個の行ベクトルで生成される．この生成行列の Hermite Normal Form を計算することで, m 次元 q-ary 格子 $\Lambda_q(\mathbf{X})$ の基底行列 $\mathbf{B} \in \mathbb{Z}^{m \times m}$ が得られる．また, 双対基底の性質から, もう片方の q-ary 格子 $\Lambda_q^{\perp}(\mathbf{X})$ の基底行列は $(q\mathbf{B}^{-1})^{\top} \in \mathbb{Z}^{m \times m}$ で得られる．

2.1.1.3 LWE 問題の代表的な求解法

■判定 LWE 問題に対する求解 判定 LWE 問題を SIS (Short Integer Solution) 問題に帰着して解く方法を紹介する：正の整数 q と, $0 < \beta < q$ を満たす実数 β を固定する．各成分が剰余環 $\mathbb{Z}/q\mathbb{Z}$ 上一様ランダムに選ばれた $n \times m$ 整数行列 $\mathbf{X}$ に対して, $\|\mathbf{v}\| \leq \beta$ かつ $\mathbf{v}\mathbf{X}^{\top} \equiv \mathbf{0} \pmod{q}$ を満たす非零ベクトル $\mathbf{v} \in \mathbb{Z}^m$ を見つける問題を **SIS 問題**と呼ぶ．つまり, これは q-ary 格子 $\Lambda_q^{\perp}(\mathbf{X})$ 上の短い非零ベクトルを見つける問題である．剰余パラメータ q における LWE 問題のサンプル数を m とし, m 個の LWE サンプルの組を $(\mathbf{A}, \mathbf{b}) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m$ とする．ここで, $n \times m$ の転置行列 $\mathbf{A}^{\top}$ に対する SIS 問題の短い解ベクトル $\mathbf{v} \in \Lambda_q^{\perp}(\mathbf{A}^{\top})$ が得られたとする $(0 < \|\mathbf{v}\| \leq \beta$ と仮定). このとき, LWE サ

ンプルの組 $(\mathbf{A}, \mathbf{b})$ は関係式 $\mathbf{b} \equiv \mathbf{s}\mathbf{A}^\top + \mathbf{e} \pmod q$ を満たすので, $\langle \mathbf{v}, \mathbf{b} \rangle \equiv \langle \mathbf{v}, \mathbf{s}\mathbf{A}^\top + \mathbf{e} \rangle \equiv \langle \mathbf{v}\mathbf{A}, \mathbf{s} \rangle + \langle \mathbf{v}, \mathbf{e} \rangle \equiv \langle \mathbf{v}, \mathbf{e} \rangle \pmod q$ が成り立つ. ($\mathbf{v}\mathbf{A} \equiv \mathbf{0} \pmod q$ に注意する.) さらに, ノイズベクトル $\mathbf{e}$ のすべての成分 e_i は離散 Gauss 分布 χ からサンプルされた元なので, $|\langle \mathbf{v}, \mathbf{e} \rangle| \approx \sigma \|\mathbf{v}\| \leq \sigma\beta$ が期待できる. (離散 Gauss 分布 $\chi = D_{\mathbb{Z},\sigma}$ のサンプル元 e_i の絶対値はおおよそ σ 未満で, 多めに見積もって $\|\mathbf{e}\| = \sigma$ とした.) ゆえに, $\sigma\beta \ll q$ ならば, $|\langle \mathbf{v}, \mathbf{b} \rangle| \pmod q$ の値の大きさから LWE サンプルの組 $(\mathbf{A}, \mathbf{b})$ は確率分布 $L_{\mathbf{s},\chi}$ からサンプルされたものか判定できる.

■探索 LWE 問題に対する求解法 探索 LWE 問題を BDD (Bounded Distance Decoding) 問題に帰着して解く方法を紹介する：格子 L と目標ベクトル $\mathbf{w}$ との距離に関して, ある $0 < \mu \leq \frac{1}{2}$ が存在し $\mathrm{dist}(\mathbf{w}, L) = \min_{\mathbf{v} \in L} \|\mathbf{w} - \mathbf{v}\| < \mu\lambda_1(L)$ を満たすと仮定する. 格子 L の基底が与えられたとき, 目標ベクトル $\mathbf{w}$ に最も近い格子ベクトル $\mathbf{v} \in L$ を見つける問題を **BDD 問題**と呼ぶ. m 個の LWE サンプルの組 $(\mathbf{A}, \mathbf{b}) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m$ は関係式 $\mathbf{b} \equiv \mathbf{s}\mathbf{A}^\top + \mathbf{e} \pmod q$ を満たすので, 探索 LWE 問題は $\mathbf{b}$ を目標ベクトルとする q-ary 格子 $\Lambda_q(\mathbf{A}^\top)$ 上の BDD 問題とみなせる. 実際, 目標ベクトル $\mathbf{b} = \mathbf{s}\mathbf{A}^\top + \mathbf{e} + q\mathbf{z}$ $(\exists \mathbf{z} \in \mathbb{Z}^m)$ に対して, 格子ベクトルを $\mathbf{v} = \mathbf{s}\mathbf{A}^\top + q\mathbf{z} \in \Lambda_q(\mathbf{A}^\top)$ とおくと, $\mathbf{b} - \mathbf{v} = \mathbf{e}$ が成り立つ. ノイズベクトル $\mathbf{e}$ のすべての成分 e_i は離散 Gauss 分布 χ からサンプルされた元であるため, 分散と次元が大きい場合にはおおよそスケーリングされたカイ二乗分布に従い, 高い確率で $\|\mathbf{e}\| \approx \frac{\sigma}{\sqrt{2\pi}} \cdot \sqrt{m}$ となる. ゆえに, 目標ベクトル $\mathbf{b}$ との距離が $\sigma\sqrt{m}$ 以下となる q-ary 格子 $\Lambda_q(\mathbf{A}^\top)$ 上の格子ベクトル $\mathbf{v}$ を見つけることで, ノイズベクトル $\mathbf{e}$ を復元することができる.

2.1.2 NTRU 問題と代表的な求解法

ここでは NTRU 問題とその代表的な求解法を紹介する. まず以下で, NTRU 問題について述べる：

定義 2.2 (NTRU 問題 [28]) 2 つの正の整数 n と q に対し, $\phi \in \mathbb{Z}[x]$ を次数 n の多項式とし, $R_q = \mathbb{Z}_q[x]/(\phi)$ とする. 係数が小さい 2 つの多項式 $f \in R_q^\times, g \in R_q$ に対して, $h = g \cdot f^{-1} \in R_q$ とする. (特に f は環 R_q の可逆元.) このとき, 与えられた多項式 h から, f または g の多項式を復元する問題を (探索) NTRU 問題という.

NTRU 問題における多項式 ϕ の選び方として, $\phi = x^n \pm 1$, $x^n - x - 1$, $x^n - x^{n/2} + 1$, $\sum_{i=0}^{n-1} x^i$ などがある [6, Table 1]. (最後の ϕ の次数は $n-1$ である.) また, 多項式 f (または g) の選び方として, $\{-1, 0, 1\}$ などの小さい係数を持つ多項式や, 小さい素数 p と係数が小さい多項式 F に対し $f = pF$ または $f = pF + 1$ と選ぶことが多い.

次に, NTRU 問題の代表的な求解法を紹介する. まず, 与えられた多項式 $h \in R_q$ に対して, h の"rotation"行列を $\mathbf{H} \in \mathbb{Z}^{n \times n}$ とする. ($n \times n$ 整数行列 $\mathbf{H}$ の i 行ベクトルを多項式 $x^{i-1}h \in R_q$ の係数ベクトルとする.) つまり

$$\mathbf{H} \begin{pmatrix} 1 \\ x \\ \vdots \\ x^{n-1} \end{pmatrix} = \begin{pmatrix} h \\ xh \\ \vdots \\ x^{n-1}h \end{pmatrix} \in R_q^n$$

を満たす. ここで, $2n \times 2n$ 行列 $\mathbf{B} = \begin{pmatrix} \mathbf{I}_n & \mathbf{H} \\ \mathbf{0} & q\mathbf{I}_n \end{pmatrix}$ の行ベクトルで生成される **NTRU 格子**を L とする. このとき, $2n$ 次元の NTRU 格子 L は短いベクトル $(\mathbf{f} \mid \mathbf{g}) \in \mathbb{Z}^{2n}$ を含む. (ただし, $\mathbf{f}, \mathbf{g} \in \mathbb{Z}^n$ を多項式 $f, g \in R_q$ の係数ベクトルとする.) 実際, $hf = g \pmod q$ より, $g = hf + qr$ を満たす多項式 $r \in R_q$ が存在する. また, 多項式 r の係数ベクトル

を $\mathbf{r} \in \mathbb{Z}^n$ とすると,

$$\mathbf{g}\begin{pmatrix}1\\x\\\vdots\\x^{n-1}\end{pmatrix} = g = hf + qr = \mathbf{f}\begin{pmatrix}h\\xh\\\vdots\\x^{n-1}h\end{pmatrix} + q\mathbf{r}\begin{pmatrix}1\\x\\\vdots\\x^{n-1}\end{pmatrix} = (\mathbf{fH} + q\mathbf{r})\begin{pmatrix}1\\x\\\vdots\\x^{n-1}\end{pmatrix} \in R_q$$

となるので, $\mathbf{g} = \mathbf{fH} + q\mathbf{r}$ が成り立つ. これより, $(\mathbf{f} \mid \mathbf{g}) = (\mathbf{f} \mid \mathbf{fH} + q\mathbf{r}) = (\mathbf{f} \mid \mathbf{r})\mathbf{B} \in L$ が成り立つ. (つまり, ベクトル $(\mathbf{f} \mid \mathbf{g})$ が NTRU 格子 L に含まれる.) ベクトル $(\mathbf{f} \mid \mathbf{g}) \in \mathbb{Z}^{2n}$ が十分小さく NTRU 格子 L 上の唯一の最短ベクトルと仮定すると, これは NTRU 問題を, unique-SVP に帰着できることを示している.

2.1.3 格子問題を解くアルゴリズムとその計算量について

最短ベクトル問題 (Shortest Vector Problem, SVP) や上記で紹介した LWE 問題・NTRU 問題などの格子問題を解くのに有用な技術として**格子基底簡約**がある. 格子基底簡約は, 与えられた格子 L の基底から, 各ベクトル $\mathbf{b}_i$ が短く・互いのベクトルが直交に近い格子 L の新しい基底 $\{\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m\}$ を見つける操作である. (明確な定義はないが, このような基底を簡約基底または良い基底と呼ぶ.)

2.1.3.1 代表的な格子基底簡約アルゴリズムの紹介

基底簡約アルゴリズムを紹介するために, Gram-Schmidt の直交化 (Gram-Schmidt orthogonalization, GSO) を説明する：基底 $\{\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m\}$ の GSO ベクトル $\mathbf{b}_i^*$ は次のように再帰的に定まる：$\mathbf{b}_1^* = \mathbf{b}_1$, $\mathbf{b}_i^* = \mathbf{b}_i - \sum_{j=1}^{i-1} \mu_{i,j}\mathbf{b}_j^*$, $\mu_{i,j} = \frac{\langle \mathbf{b}_i, \mathbf{b}_j^* \rangle}{\|\mathbf{b}_j^*\|^2}$. また, 各 $2 \leq \ell \leq m$ に対し $\mathbb{R}^m$ から $\mathbb{R}$-ベクトル空間 $\langle \mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_{\ell-1} \rangle_{\mathbb{R}}$ の直交補空間への直交射影を π_ℓ とかく. (便宜上 π_1 を恒等写像とする.) 以下で, 代表的な 2 つの格子基底簡約アルゴリズムを紹介する：

■LLL [31] 簡約パラメータ $\frac{1}{4} < \delta < 1$ に対し, LLL 基底簡約は次の 2 条件を満たす基底 $\{\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m\}$ を見つける (次元 m に関する) 多項式時間アルゴリズムである： (i) 基底 $\{\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m\}$ はサイズ簡約されている. つまり, GSO 係数が $|\mu_{i,j}| \leq \frac{1}{2}$ $(\forall i > \forall j)$ を満たす. (ii) 基底 $\{\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m\}$ は Lovász 条件を満たす. つまり, $\delta\|\mathbf{b}_{k-1}^*\|^2 \leq \|\pi_{k-1}(\mathbf{b}_k)\|^2$ $(2 \leq \forall k \leq m)$ を満たす. 入力基底に対して, Lovász 条件が成り立たないとき LLL 基底簡約アルゴリズム内で隣り合う基底ベクトル $\mathbf{b}_{k-1}$ と $\mathbf{b}_k$ の交換を行い, (i) と (ii) の 2 条件を満たす基底を見つける.

■BKZ [45] BKZ 基底簡約アルゴリズムはブロックサイズ β による LLL 基底簡約アルゴリズムの一般化である. LLL に比べ, BKZ 基底簡約アルゴリズムでより良い簡約基底を見つけることが可能であるが, その計算量は β に関して指数時間である. 特に, BKZ 基底簡約アルゴリズムに入力するブロックサイズ β を増やすごとに, 実行時間が非常に遅くなるが, より短い基底ベクトルを出力する. 具体的には, ブロックサイズ $2 \leq \beta \leq m$ に対して, BKZ 基底簡約アルゴリズムは次の 2 つの条件を満たす格子 L の基底 $\{\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m\}$ を見つける： (i) 基底はサイズ簡約されている. (ii) すべての $1 \leq j \leq m$ に対し $\|\mathbf{b}_j^*\| = \lambda_1(L_{[j,k]})$ を満たす. ただし, $k = \min(j + \beta - 1, m)$ とし, ベクトル $\pi_j(\mathbf{b}_j), \ldots, \pi_j(\mathbf{b}_k)$ で生成されるブロック射影格子を $L_{[j,k]}$ とする. 入力基底に対して, BKZ 基底簡約アルゴリズム内ではブロック射影格子 $L_{[j,k]}$ 上の SVP オラクルを繰り返し呼びだし, (i) と (ii) の 2 条件を満たす基底を見つける.

2.1.3.2 BKZ 基底簡約アルゴリズムの出力基底と計算量

これまで BKZ2.0 [20] などの効率的な BKZ の改良アルゴリズムが提案され, 格子に基づく暗号技術の安全性評価でよく利用されている. 以下で BKZ の出力基底と計算量評価の見積もりについて紹介する (詳細は [6] を参照) ：

■BKZ の出力基底の見積もり　格子基底簡約アルゴリズムが出力する簡約基底の「良さ」を測る指標として Hermite 因子がある．m 次元格子 L の基底が与えられたとき, アルゴリズムが出力する最短な基底ベクトルを $\mathbf{b} \in L$ とする．このとき, その基底簡約アルゴリズムの **Hermite 因子**は $\gamma = \frac{\|\mathbf{b}\|}{\mathrm{vol}(L)^{1/m}}$ で定義される．(つまり, Hermite 因子が小さいほど, より短い基底ベクトルの出力を意味する.) 100 以上の高次元のランダム格子に対し, LLL や BKZ などの基底簡約アルゴリズムの Hermite 因子の m 乗根 $\gamma^{1/m}$ は定数に収束することが実験的に知られている．高い次元 m のランダム格子において, ブロックサイズ $\beta \geq 50$ に対する BKZ 基底簡約アルゴリズムの root Hermite 因子はおおよそ $\gamma^{\frac{1}{m}} \approx \left(\nu_\beta^{-\frac{1}{\beta}}\right)^{\frac{1}{\beta-1}} \approx \left(\frac{\beta}{2\pi e}(\pi\beta)^{\frac{1}{\beta}}\right)^{\frac{1}{2(\beta-1)}}$ に従うことが実験的に知られている．ただし, ν_β は β-次元の単位超球の体積とする．(例えば $\beta = 85$ で $\gamma^{1/m} \approx 1.01$ となる.) この root Hermite 因子の見積もりを用いて, 格子に基づく暗号技術の安全性評価対象の格子問題の求解で必要となる BKZ のブロックサイズ β を求めることができる.

■BKZ の計算量の見積もり　BKZ 基底簡約アルゴリズムの計算量は, β 次元格子上の「SVP オラクルの計算量」と「呼び出し回数」の積で見積もることができる．β 次元格子上の SVP オラクルに適したアルゴリズムとして篩 (sieving) と数え上げ (enumeration) があり, 篩の方が漸近計算量が小さい．(ただし, 篩の空間計算量は β の指数関数で, 数え上げに比べて非常に大きい.) β 次元格子上の篩の計算量は $2^{c\beta+o(\beta)}$ で, 古典計算機上では $c = 0.292$ で, Grover アルゴリズムによって量子計算機上で $c = 0.265$ と見積もられている．一方, 数え上げの計算量は古典計算機上で $2^{c_1\beta\log\beta+c_2\beta+c_3}$ または $2^{c_1\beta^2+c_2\beta+c_3}$ で, Grover アルゴリズムにより量子計算機上ではその指数部分が半分になると見積もられている．(定数 c_1, c_2, c_3 に関しては様々な評価があり, 具体的な値については [6, Table 4] を参照.) また, BKZ 内の SVP オラクルの呼び出し回数については, β または $8m$ と見積もることが多い．(β は BKZ のブロックサイズで, m は格子の次元とする.)

2.2　代表的な格子に基づく暗号方式の説明

本節では, 格子に基づく代表的な暗号方式として, LWE 問題に基づく Regev による暗号化方式 [40], ならびに Ring-LWE 問題に基づく Brakerski らによる暗号化方式 [15], 更に NTRU 問題に基づく Hoffstein らによる暗号化方式 [28] について述べる.

2.2.1　LWE に基づく暗号化

Regev による暗号化方式 [40] の構成には, 以下の 4 つのパラメータが必要である.

- n: 安全性パラメータ
- m: LWE サンプルの個数 ($m \geq 1.1 \cdot n\log q$ となる最小の整数を選ぶ)
- q: 剰余パラメータ (q として $n^2 \leq q \leq 2n^2$ を満たす素数を選ぶ)
- $\alpha > 0$: ノイズパラメータ ($\alpha = 1/(\sqrt{n} \cdot \log^2 n)$)

以下に具体的な暗号方式の構成を示す.

秘密鍵の生成　一様ランダムに $\mathbf{s} \leftarrow \mathbb{Z}_q^n$ を選ぶ

公開鍵の生成　秘密鍵 $\mathbf{s}$, 剰余パラメータ q ノイズパラメータ α を持つ LWE 分布から生成した m 個のサンプル $(\mathbf{a}_i, b_i)_{i=1}^m \leftarrow A_{\mathbf{s},\chi}^m$ を公開鍵とする．ただし各 i について, $a \leftarrow \mathbb{Z}_q^n$, $e_i \leftarrow \chi = D_{\mathbb{Z},\alpha q}$ とした時, $b_i = \langle \mathbf{a}_i, s\rangle + e_i \in \mathbb{Z}_q$ とする.

暗号化　集合 S を $\{1,2,\ldots,m\}$ の中から一様ランダムに選んだ部分集合とする．このとき, 平文ビットが 0 の暗号文を $(\sum_{i\in S}\mathbf{a}_i, \sum_{i\in S} b_i)$ とし, 平文ビットが 1 の暗号文を $(\sum_{i\in S}\mathbf{a}_i, \lfloor\frac{q}{2}\rfloor + \sum_{i\in S} b_i)$ とする．

復号　暗号文 $(\mathbf{a}, b)$ に対し, $b - \langle\mathbf{a},\mathbf{s}\rangle \in \mathbb{Z}_q$ が $\lfloor\frac{q}{2}\rfloor$ より 0 に近い場合, 復号結果として 0 を出力し, それ以外の場合は 1 を出力する．

復号の正当性について．平文の 0 に対応する暗号文 $(\mathbf{a}, b) = (\sum_{i\in S}\mathbf{a}, \sum_{i\in S} b_i)$ の場合, $b - \langle\mathbf{a},\mathbf{s}\rangle \in \mathbb{Z}_q = \sum_{i\in S}(b_i - \langle\mathbf{a}_i,\mathbf{s}\rangle) = \sum_{i\in S} e_i$ なので, $-\frac{q}{4} < \sum_{i\in S} e_i < \frac{q}{4}$ であれば復号に成功し, 0 が出力される．

各ノイズ e_i は Gauss 分布 $\chi = D_{\mathbb{Z},\alpha q}$ から選ばれているので, $\sum_{i\in S} e_i$ の標準偏差は高々 $\sqrt{m}\alpha q$ となる．

ここで, 各パラメータの選択方法から $\sqrt{m}\alpha q < q/\log n$ であり, 非常に高い確率で復号に成功することが分かる．また平文ビットが 1 の暗号文に対しても同様の議論が成り立つ．この暗号方式の安全性については, LWE 仮定の下で CPA 安全であることが証明されている [46].

ここで紹介した [40] による暗号方式は, 公開鍵のサイズが $(mn\log q) = \widetilde{O}(n^2)$ で, 暗号文サイズも平文サイズの $O(n\log q) = \widetilde{O}(n)$ 倍に増加するため, 決して効率的ではない．より効率的な方式としては [39] などを参照.

2.2.2　Ring-LWE に基づく暗号化

Brakerski らによる Ring-LWE 問題にもとづく暗号化方式は, 暗号化したまま限定回の加算と乗算が可能な somewhat 準同型暗号として提案されているものである．この暗号方式には, 以下の 4 つのパラメータが必要である．

- n: 2 べき整数で, 暗号方式を構成する基礎的な環 $R = \mathbb{Z}[x]/(x^n+1)$ を定義する (n が 2 べき整数の場合のみ, 多項式 x^n+1 は $\mathbb{Z}$ 上既約となることに注意).
- q: $q \equiv 1 \pmod{2n}$ を満たす素数で, 暗号文空間の基礎環 $R_q = \mathbb{Z}_q[x]/(x^n+1)$ を定義する．
- t: 条件 $t < q$ を満たす整数で, 暗号方式の平文空間 $R_t = \mathbb{Z}_t[x]/(x^n+1)$ を定義する．
- $\sigma > 0$: ノイズを与えるための離散ガウス分布のパラメータ.

以下に具体的な暗号方式の構成を示す．なお, $a_0 + a_1x + \cdots + a_{n-1}x^{n-1} \to (a_0, a_1, \ldots, a_{n-1})$ によって, 環 R を $\mathbb{Z}^n$ と同一視する．また同様に R_q と $\mathbb{Z}_q^n$ を同一視する．

鍵生成　$s \in R \leftarrow D_{\mathbb{Z}^n,\sigma}$ を選び, 一様ランダムに $p_i \in R_q$ を取り, 小さなエラー $e \leftarrow \chi$ を固定する．([15] では $s \leftarrow \chi$ を一様ランダムに選択するのに対し, [32] では一様ランダムには選択しない点だけが異なる)．そこで, 公開鍵を $\mathsf{pk} = (p_0, p_1)$ とし (ただし, $p_0 = -(p_1s + te)$ とする), 秘密鍵を $\mathsf{sk} = s$ とする.

暗号化　平文情報 $m \in R_t$ と公開鍵 $\mathsf{pk} = (p_0, p_1)$ に対し, まず $R \ni u, f, g \leftarrow \chi$ を選び, 暗号文を

$$\mathsf{Enc}(m, \mathsf{pk}) = (c_0, c_1) = (p_0u + tg + m, p_1u + tf),$$

と定義する．ただし, 条件 $t < q$ より, 上記の数式では元 $m \in R_t$ を環 R_q の元として見なして計算する．つまり, 上記の暗号文は $(R_q)^2$ の元として表現される.

復号　任意の長さの暗号文 $\mathsf{ct} = (c_0, c_1, \ldots, c_\xi)$ に対して, 復号は

$$\mathsf{Dec}(\mathsf{ct}, \mathsf{sk}) = [\tilde{m}]_q \bmod t \in R_t,$$

で計算できる．ただし, $\tilde{m} = \sum_{i=0}^{\xi} c_i s^i \in R_q$ であり, $[\tilde{m}]_q$ は元 $\tilde{m}$ の各係数の $[-q/2, q/2)$ への剰余とする．また, $\mathbf{s} = (1, s, s^2, \ldots, s^\xi)$ としたとき, この復号処理を $\mathsf{Dec}(\mathsf{ct}, \mathsf{sk}) = [\langle\mathsf{ct}, \mathbf{s}\rangle]_q \bmod t$ と書き直すこともできる.

復号の正当性については, 上記の暗号アルゴリズムで得られる暗号文 $\mathsf{ct} = (c_0, c_1)$ に対し, 関係式 $p_0 + p_1 s = -te$ が成り立つことから, $\langle \mathsf{ct}, \mathbf{s} \rangle = (p_0 u + tg + m) + s \cdot (p_1 u + tf) = m + t \cdot (g + sf - ue)$ が環 R_q 上で成り立つ. ここで, 元 $m + t \cdot (g + sf - ue)$ を環 R の元と見なしたとき, その各係数が $[-q/2, q/2)$ 内に収まっている限り, $[\langle \mathsf{ct}, \mathbf{s} \rangle]_q = m + t \cdot (g + sf - ue)$ が環 R 上で成立する (元 $e, f, g, u \leftarrow \chi$ が十分小さなノイズとして選択されていることに注意). この場合, 剰余 mod t の操作で正しい復号結果 $m \in R_t$ が得られる.

また, この暗号方式の安全性については, Ring-LWE 問題の計算量困難性仮定の下で KDM 安全 (key dependent message security) であることが証明されている [15].

2.2.3 NTRU 問題に基づく暗号化

Hoffstein らによる NTRU 問題に基づく暗号化方式 NTRUEncrypt [28] の構成には次のパラメータが必要である.

- n : 正の整数 (セキュリティパラメータ)
- q : 正の整数 (素数である必要はない)
- p : q と互に素で $p \ll q$ である正の整数
- ϕ: 次数 n の多項式であり環 $R_p = \mathbb{Z}_p[x]/(\phi)$, $R_q = \mathbb{Z}_q[x]/(\phi)$ を定義する (ϕ としては例えば $x^n \pm 1$, $x^n - x - 1$ 等)

以下に具体的な暗号方式の構成を示す.

鍵生成 すべての係数の絶対値が小さい二つの多項式 $f \in R_q, g \in R_q$ を選ぶ. ただし, f は R_p, R_q の両方において可逆な要素とする. すなわち, ある f_p, f_q が存在し, 以下を満たす.

$$f_p \cdot f = 1 \in R_p, f_q \cdot f = 1 \in R_q$$

ここで f, f_p を秘密鍵とし, $h = p f_q \cdot g \in R_q$ を公開鍵とする. なお f_p, f_q ならびに g は f と h を用いて復元可能であることに注意する.

暗号化 平文情報として, すべての係数の絶対値が p より小さい (例えば $-1, 0, 1$ のいずれかである) 要素 $m \in R_q$ とし, 公開鍵 $\mathsf{pk} = h$ に対し, $r \in R_q$ を係数が小さい多項式からランダムに選び, 暗号文を

$$\mathsf{Enc}(m, \mathsf{pk}) = r \cdot h + m \in R_q$$

と定義する.

復号 暗号文 $c \in R_q$ に対し, 復号は

$$\mathsf{Dnc}(m, \mathsf{sk}) = [f_p \cdot [f \cdot c]_q]_p$$

で求められる. ただし $[a]_q$, $[a]_p$ は元 $a \in R_q$ の各係数をそれぞれ $[-q/2, q/2)$, $[-p/2, p/2)$ に収めたものとする.

復号の正当性については, 次のように示される. $[f \cdot c]_q$ は, $f \cdot c = f \cdot (r \cdot h + m) = f(r \cdot p f_q \cdot g + m) = pr \cdot g + f \cdot m \in R_q$ と変形されるが, r, g, f, m 共に, 係数が小さいものから抽出しており, また $p \ll q$ であること, 更に係数が $[-q/2, q/2)$ に収められていることから適切なパラメータ選択により, 上記式は q による剰余を伴わない等式, すなわち $f \cdot c = pr \cdot g + f \cdot m \in Z[x]/(\phi)$ が満たされる. また右辺第一項は p 倍項であることから, 続く p による剰余で消去され, $f_p \cdot (pr \cdot g + f \cdot m) = f_p \cdot f \cdot m = m \in R_p$ となり正しい復号結果 m が得られる.

この NTRU Encrypt 暗号の安全性についてはアルゴリズム提案当初格子問題への安全性帰着がついていなかったが, Stehlé ら [47] により, standard model の CPA 仮定に基づくイデアル格子上の Ring-SIS (Small Integer Solution) 問題、ならびに Ring-LWE 問題に帰着されることが示されている.

2.3 具体的な暗号方式

本節では, 格子問題の困難性を安全性の根拠とした暗号方式のうち, 主要なものを紹介する. 根拠となる問題の多様性と構成の単純さを基準として選ぶ. 取り上げた方式は全て, NIST が主催する耐量子計算機暗号選定プロジェクトの Round 1 候補であったもので, 詳細な仕様書が公開されているものである. また, いくつかは効率的な実装のための補助サブルーチンを用いているが, 暗号方式の原理にフォーカスするために極力省いてある. 例えば, NewHope [2] 等で用いられている乱数を入力として公開鍵用の多項式 a を出力する関数は省き, 直接に a を渡す形での表現をしている. 各暗号方式はその構成に応じて様々なノイズを用いている

表 2.1　本節で扱う主要な格子に基づく暗号技術

文献	暗号化	鍵交換	署名
LOTUS [38]	○	○	
NewHope [2, 3, 9, 10]		○	
Lizard [21, 18, 19]	○	○	
CRYSTALS-Dilithium [23, 22, 24]			○
pqNTRUSign [17, 29, 30]			○

2.3.1 LOTUS

LOTUS は, Phong ら [38] により NIST に提案する耐量子暗号として開発された. LWE 問題を安全性の根拠と置いた暗号化方式は, Regev[46] による 1 ビットの暗号化を行うものが最初であり, 同様の原理で多ビットの暗号化を可能とした Lindner-Peikert 方式 [33] を発展させた暗号方式が多く提案されている. LOTUS [*4] はそのような暗号方式の中でも最も単純と考えられる. 方式の構成は, Lindner-Peikert 方式 (表 2.2) に藤崎-岡本変換 [26, 27] を適用し, 微修正を行ったもので, 原型は青野らの文献 [5, Sect. 4] に見られる.

表 2.3 に概略を示す. LWE ノイズ $D_{\mathbb{Z},\sigma}^{\mathsf{LOTUS}}$ は, $\sigma = 8.0$ の離散ガウス分布 $D_{\mathbb{Z},8.0}$ である. Knuth-Yao アルゴリズムを用いて, 理想的な出力との統計距離が 2^{-256} 以下となる実装を行っている. 実装時の共通鍵暗号は AES-CTR で, 対応するセキュリティレベルごとにブロック長を定めている. ハッシュ関数は SHA-512 を用い, $G(x) = \mathsf{SHA\text{-}512}(x||\mathtt{0x01})$, $H(x) = \mathsf{SHA\text{-}512}(x||\mathtt{0x02})$ としている. 全ての演算は $\mathbb{Z}_q$ 上で行われ, 自動的に区間 $(-q/2, q/2]$ に入るように計算される.

[*4] **L**earning with err**O**rs based encryption with chosen ciphertex**T** sec**U**rity for po**S**t quantum era の略称

表 2.2　Lindner-Peikert PKE

パラメータ	λ (セキュリティ), n, ℓ (次元), q (法), σ (離散ガウス分布のパラメータ)
鍵生成 $\mathsf{KeyGen}(1^\lambda) \to (\mathsf{sk}, \mathsf{pk})$	$A \xleftarrow{\$} \mathbb{Z}_q^{n\times n}$, $R, S \leftarrow D^{\mathsf{LOTUS}}_{\mathbb{Z}^{n\times\ell},\sigma}$ $\mathsf{sk} = S$, $\mathsf{pk} = (A, P = R - AS \pmod q)$
暗号化 $\mathsf{Enc}(\mathsf{pk}, \mathsf{M} \in \{0,1\}^{1\times\ell}) \to \mathsf{ct}$	$\mathbf{e}_1, \mathbf{e}_2 \leftarrow D^{\mathsf{LOTUS}}_{\mathbb{Z}^{1\times n},\sigma}$, $\mathbf{e}_3 \xleftarrow{\$} D^{\mathsf{LOTUS}}_{\mathbb{Z}^{1\times\ell},\sigma}$ $\mathbf{c}_1 = \mathbf{e}_1 A + \mathbf{e}_2 \pmod q$, $\mathbf{c}_2 = \mathbf{e}_1 P + \mathbf{e}_3 + \mathsf{M} \cdot \lfloor q/2 \rceil \pmod q$ $\mathsf{ct} = (\mathbf{c}_1, \mathbf{c}_2)$
復号 $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) \to M'$	$\overline{M} = \mathbf{c}_1 S + \mathbf{c}_2 \pmod q := (\overline{M_1}, \overline{M_2}, \ldots, \overline{M_\ell})$ // mod q の結果は区間 $[-q/2, q/2]$ に入るように計算される $M'_i :=$ (復元メッセージ M' の i ビット目) $= \begin{cases} 0 & (\text{if } \lvert\overline{M_i}\rvert < q/4) \\ 1 & (\text{if otherwise}) \end{cases}$

表 2.3　LOTUS-PKE の概略 [38, Section 3.3]

パラメータ	n, ℓ (次元), q (法), $\sigma = 8.0$ (離散ガウス分布のパラメータ)
共通鍵暗号	暗号化 $\mathsf{SE}_K(m)$, 復号 $\mathsf{SD}_K(c)$
ハッシュ関数	G, H
鍵生成 $\mathsf{KeyGen}(1^\lambda) \to (\mathsf{sk}, \mathsf{pk})$	1: $A \xleftarrow{\$} \mathbb{Z}_q^{n\times n}$, $R, S \leftarrow D^{\mathsf{LOTUS}}_{\mathbb{Z}^{n\times\ell},\sigma}$ 2: $\mathsf{sk} = S$, $\mathsf{pk} = (A, P = R - AS \pmod q)$
暗号化 $\mathsf{Enc}(\mathsf{pk}, M \in \{0,1\}^{1\times\ell}) \to \mathsf{ct}$	1: $\mathsf{str} \xleftarrow{\$} \{0,1\}^\ell$, $c_{sym} = \mathsf{SE}_{G(\mathsf{str})}(M)$, $h = H(\mathsf{str} \| c_{sym})$ 2: $\mathbf{e}_1, \mathbf{e}_2 \xleftarrow{\$} D^{\mathsf{LOTUS}}_{\mathbb{Z}^{1\times n},\sigma}$, $\mathbf{e}_3 \xleftarrow{\$} D^{\mathsf{LOTUS}}_{\mathbb{Z}^{1\times\ell},\sigma}$ // h をシードとして, 離散ガウス分布を生成 3: $\mathbf{c}_1 = \mathbf{e}_1 A + \mathbf{e}_2 \pmod q$, $\mathbf{c}_2 = \mathbf{e}_1 P + \mathbf{e}_3 + \mathsf{str} \cdot \lfloor q/2 \rceil \pmod q$ // 乱数列 str を暗号化 4: $\mathsf{ct} = (\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_{sym})$
復号 $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) \to M'$	1: $\overline{\mathsf{str}} = \mathbf{c}_1 S + \mathbf{c}_2 \pmod q := (\overline{s_1}, \overline{s_2}, \ldots, \overline{s_\ell})$ 2: $\mathsf{str}' = (s'_1, s'_2, \ldots, s'_\ell)$ where $s'_i = \begin{cases} 0 & (\text{if } \lvert\overline{s_i}\rvert < q/4) \\ 1 & (\text{if otherwise}) \end{cases}$ 3: $h' = H(\mathsf{str}' \| c_{sym})$ 4: (整合性チェック) $\mathbf{e}'_1, \mathbf{e}'_2 \xleftarrow{\$} D^{\mathsf{LOTUS}}_{\mathbb{Z}^{1\times n},\sigma}$, $\mathbf{e}'_3 \xleftarrow{\$} D^{\mathsf{LOTUS}}_{\mathbb{Z}^{1\times\ell},\sigma}$ //h' をシードとして, 離散ガウス分布を生成 5: $\mathbf{c}'_1 = \mathbf{e}_1 A + \mathbf{e}_2 \pmod q$; $\mathbf{c}'_2 = \mathbf{e}_1 P + \mathbf{e}_3 + \mathsf{str}' \cdot \lfloor q/2 \rceil \pmod q$ 6: $(\mathbf{c}'_1, \mathbf{c}'_2) = (\mathbf{c}_1, \mathbf{c}_2)$ であれば, 復号成功として $M' = \mathsf{SD}_{G(\sigma')}(\mathbf{c}_{sym})$ を出力 7: $(\mathbf{c}'_1, \mathbf{c}'_2) \neq (\mathbf{c}_1, \mathbf{c}_2)$ であれば, 復号失敗としてエラーを出力

表 2.4　LOTUS-KEM のパラメータ [38, Table 2.5]

(n, q, s)	安全性レベル	公開鍵サイズ	秘密鍵サイズ	平文サイズ	暗号文サイズ (鍵カプセル化後)
$(576, 8192, 3.0)$	128 bits	658.95KB	700.42KB	16B	1.144KB
$(704, 8192, 3.0)$	192 bits	1025.0KB	1101.0KB	24B	1.456KB
$(832, 8192, 3.0)$	256 bits	1471.0KB	1590.8KB	32B	1.768KB

暗号文が正しい場合, 復号アルゴリズムのステップ 1 で計算された $\overline{\mathsf{str}}$ は

$$\overline{\mathsf{str}} = \mathbf{e}_1 R + \mathbf{e}_2 S + \mathbf{e}_3 + \mathsf{str} \cdot \lfloor q/2 \rceil \pmod q$$

を満たす. 右辺の $\mathbf{e}_1 R + \mathbf{e}_2 S + \mathbf{e}_3$ の要素は全て $D_{\mathbb{Z},\sigma}^{\mathsf{LOTUS}}$ から生成された整数の積・和であるため, $q/2$ と比べて小さく, ステップ 2 の操作により元の乱数列 str が高確率で正しく復元できることがわかる. よって, ステップ 3 の h' が暗号化時に用いられた h と同一のビット列となるため, 以降の計算は暗号化時のものと等しくなり, 整合性のチェックが可能となる.

LWE 問題の困難性を仮定すると, LOTUS-PKE が IND-CCA2 安全であることが証明できる [38, Theorem 3].

2017 年 12 月末に LOTUS-KEM の実装に対して, 選択暗号文攻撃が可能となるとの指摘があった. 翌年 1 月, 指摘部分のコードを修正することで解決している.

2.3.2　NewHope

NewHope は, Alkim ら [10] により国際会議 Usenix Security で提案された. 最初の提案プロトコルでは Reconciliation を行っていたが, 後に提案された NewHope-Simple [9] ではその部分が削られ, より簡素なプロトコルとなっている.

NIST の公募には NewHope [2] が, 後に微修正を施した NewHope v1.01 [3] が著者らの Web ページで公開されている. 表 2.5 に基礎となる NewHope-CPA-PKE の概要を述べる. 実際のプロトコルにはビット列へのエンコード・デコードの関数が含まれているが, ここでは省略してある. 全ての演算は環 $R_q := \mathbb{Z}_q[x]/(x^n + 1)$ 上の多項式の操作として表現され, 係数は自動的に区間 $[-q/2, q/2)$ 内に収められるものとする.

方式の特徴として, 環での演算を高速化するため, 数論変換 (NTT, Number Theoretic Transform)[3, p. 7-8]

$$a = \sum_{i=0}^{n-1} a_i x^i \in R_q \Leftrightarrow \mathsf{NTT}(a) := \sum_{i=0}^{n-1} \hat{a} x^i,\ \hat{a}_i := \sum_{j=0}^{n-1} \gamma^j a_j \omega^{ij} \pmod q$$

を行い, $a, b \in R_q$ の積が $a * b = \mathsf{NTT}^{-1}(\mathsf{NTT}(a) \circ \mathsf{NTT}(b))$ を満たすことを利用している. ここで, ω は 1 の $\mathbb{Z}_q$ での n 乗根, $\gamma := \sqrt{w} \pmod q$ とし, 実装時にはあらかじめ定数として組み込んでおくものとする. また, 記号 $\circ$ は多項式の成分同士の積を取ることを表す. 実装では上の定義式をそのまま計算せず, 高速数論変換を用いている.

鍵生成のステップ 1 では, 32 バイト (=256 ビット) の乱数を $\mathsf{SHAKE\text{-}256}$ ハッシュ関数を用いて 64 バイトに伸長し, その前半を公開鍵 $\hat{a} \in R_q$ の生成に, 後半を秘密鍵 $\hat{s}$ およびノイズ $\hat{e}$ の生成に使っている. このとき, $\hat{a}, \hat{s}, \hat{e}$ は数論変換後の形式で格納される. ステップ 2 の GenA 関数は, ハッシュ関数の出力の前半 $z[0:31]$ をシードとしてランダムな R_q の元を生成するものである. $\hat{a}$ がランダムな多項式の数論変換であることは, ランダム多項式の数論変換がまたランダム多項式となることから従う.

表 2.5　NewHope-CPA-PKE の概要 ([3, Algorithm 1-3] より一部改変)

パラメータ	n (次元), q (法), k (ノイズエラー), $\gamma \in \mathbb{Z}_q$ (NTT) $R_q := \mathbb{Z}_q[x]/(x^n+1)$
鍵生成 $\mathsf{KeyGen}(1^\lambda) \to (\mathsf{sk}, \mathsf{pk})$	1: $\mathsf{seed} \xleftarrow{\$} \{0,1,\ldots,255\}^{32}$, $z = \mathsf{SHAKE256}(64, \mathsf{seed})$ //32 バイトの seed を 64 バイトに伸長 2: $\hat{a} = \mathsf{GenA}(z[0:31]) \in R_q$ 3: $s = \mathsf{PolyBitRev}(\mathsf{Sample}(z[32:63], 0)) \in R_d$; $\hat{s} = \mathsf{NTT}(s)$ 4: $e = \mathsf{PolyBitRev}(\mathsf{Sample}(z[32:63], 1)) \in R_d$; $\hat{e} = \mathsf{NTT}(e)$ 5: $\hat{b} = \hat{a} \circ \hat{s} + \hat{e}$ 6: $\mathsf{sk} = \hat{s}$, $\mathsf{pk} = (\hat{a}, \hat{b})$
暗号化 $\mathsf{Enc}(\mathsf{pk}, M \in \{0,1,\ldots,255\}^{32}) \to \mathsf{ct}$	1: $\mathsf{coin} \xleftarrow{\$} \{0,1,\ldots,255\}^{32}$ // ランダムシード 2: $s' = \mathsf{PolyBitRev}(\mathsf{Sample}(\mathsf{coin}, 0)) \in R_d$; $\hat{t} = \mathsf{NTT}(s')$ 3: $e' = \mathsf{PolyBitRev}(\mathsf{Sample}(\mathsf{coin}, 1)) \in R_d$ 4: $e'' \leftarrow \mathsf{Sample}(\mathsf{coin}, 2) \in R_d$ 5: $\hat{u} = \hat{a} \circ \hat{t} + \mathsf{NTT}(e')$ 6: $v = \mathsf{NTT}^{-1}(\hat{b} \circ \hat{t}) + e'' + \mathsf{Encode}(M)$ 7: $\mathsf{ct} = (\hat{u}, v)$
復号 $Dec(sk, c = (\hat{u}, v))$	1: $M' = \mathsf{Decode}(v - \mathsf{NTT}^{-1}(\hat{u} \circ \hat{s}))$

表 2.6　NewHope CPA-KEM のパラメータ [2, Table 3].

(n, q, k, γ)	安全性レベル	公開鍵サイズ	秘密鍵サイズ	平文サイズ	暗号文サイズ (鍵カプセル化後)
$(512, 12289, 8, 10968)$	128 bits	928B	869B	32B	1088B
$(1024, 12289, 8, 7)$	256 bits	1824B	1792B	32B	2176B

2 番目のパラメータは Category 5 に入ると主張されているが、表 [2, Table 3] では 233 bit-security と書かれている.

ステップ 3 および 4 の, Sample 関数は R_q の各係数を独立に ψ_8 からサンプリングしたものであり, 自然数 k に対して, ψ_k の出力は $b_i, b'_i \in \{0,1\}$, $i = 1, 2, \ldots, k$ を独立に取ったときの $\sum_{i=1}^{k}(b_i - b'_i)$ として定義される. PolyBitRev 関数は多項式 $a = \sum_{i=0}^{n-1} a_i x^i$ に対して, 指数部分をビット反転した多項式 $a^{\mathrm{rev}} := \sum_{i=0}^{n-1} a_i x^{\mathsf{BitRev}(i)}$ を出力する. より具体的に書くと, $h = \log_2(n)$, $i = \sum_{j=0}^{h-1} b_j 2^j$ と 2 進数展開したときに, $\mathsf{BitRev}(i) := \sum_{j=0}^{h-1} b_j 2^{h-j-1}$ である. 実装時には BitRev をあらかじめ計算した配列を用意し, 多項式の係数が格納されている配列の要素の入れ替えを行うことで高速処理が可能である. 高速数論変換を行うときの実装上の要請からあらかじめ指数部分のビット反転を行う必要があり, 鍵生成および復号アルゴリズム内で NTT 関数の直前にこれを行っている[*5].

復号アルゴリズムの Decode 関数の中身を計算すると,

$$v - \mathsf{NTT}^{-1}(\hat{u} \circ \hat{s}) = te + se' + e'' + \mathsf{Encode}(M) \in R_q$$

となり, 右辺が $\mathsf{Encode}(M) + (\text{ノイズ項})$ となることがわかる. ノイズの影響を受けずに M を正しくデコード可能とするため, Encode 関数は 256 ビットの M の情報を $n = 1024$ 個の係数に分散させている. 具体的には, 1 つのビットの情

[*5] 従って, 例えば鍵生成アルゴリズムのステップ 3 は $s \leftarrow \mathsf{Sample}(z[32:63], 0) \in R_d$; $\hat{s} = \mathsf{NTT}(\mathsf{PolyBitRev}(s))$ と書いた方が自然かもしれないが, 原著の表現に従った.

表 2.7　Lizard PKE [21, Section 3.2.1]

パラメータ	λ (セキュリティ), n, m, ℓ (次元), p, q (法), α, ρ, h_r (ノイズ分布)
鍵生成 $\mathsf{KeyGen}(1^\lambda) \to (\mathsf{sk}, \mathsf{pk})$	1: $A \xleftarrow{\$} Z_q^{m\times n}$, $\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_\ell \leftarrow \mathcal{ZO}_{\mathbf{n}}(\rho)$, $S = (\mathbf{s}_1 \| \mathbf{s}_2 \| \cdots \| \mathbf{s}_\ell) \in Z_q^{n\times \ell}$, $E \leftarrow D^{\mathsf{Lizard}}_{\mathbb{Z}^{m\times\ell}, \alpha q}$ 2: $B = -AS + E \pmod q$ 3: $\mathsf{sk} = S$, $\mathsf{pk} = (A \| B)$
暗号化 $\mathsf{Enc}(\mathsf{pk}, M \in \{0,1\}^{\ell\times 1}) \to \mathsf{ct}$	1: $\mathbf{r} \leftarrow B_{m,h_r}$ 2: $\mathbf{a} = \lfloor (p/q) \cdot A^T\mathbf{r} \rceil$, $\mathbf{b} = \lfloor (p/q) \cdot ((q/2) \cdot m + B^T\mathbf{r}) \rceil$ // ベクトルに対する Rounding $\lfloor \cdot \rceil$ は成分ごとに行う 3: $\mathsf{ct} = (\mathbf{a}, \mathbf{b}) \in \mathbb{Z}_p^{n\times 1} \times \mathbb{Z}_p^{\ell\times 1}$
復号 $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) \to M'$	1: $M' \leftarrow \left\lfloor \frac{2}{p}(\mathbf{b} + S^T\mathbf{a}) \right\rceil \pmod 2$

報を 4 個の係数に割り当ててエンコードし, Decode 関数内では 4 つ分の係数の和を取ることでエラーを打ち消すことで, 復号の成功確率を高めている.

2.3.3　Lizard

Cheon ら [18] によって 2016 年に提案された Lizard は LWE 問題と LWR 問題の困難性を同時に仮定することで, IND-CPA 安全であることが証明される公開鍵暗号方式である. NIST に提案された [21] 後, 国際会議 SCN2018 においても発表されている [19]. ここでは, 基本的な公開鍵暗号方式 [21, Section 3.2.1] の概要を紹介する. 上記 2 つの方式とほぼ同様に, 藤崎-岡本変換 [26, 27] を用いて IND-CCA 安全な方式とすることができる.

表 2.7 に方式の概要を示す. 内部で用いられるサブルーチンとして, 3 種類の乱数生成ルーチンが用いられる. 鍵生成のステップ 1 における $\mathcal{ZO}_n(\rho)$ は, 各成分を $\Pr[x_i = 1] = \Pr[x_i = -1] = \rho/2$, $\Pr[x_i = 0] = 1 - \rho$ を満たす確率変数から独立に取った n 次元列ベクトルを出力する関数である. 行列 E を生成するときに用いるノイズ分布 $D^{\mathsf{Lizard}}_{\mathbb{Z},\alpha q}$ は, スケール αq の離散ガウス分布 $D_{\mathbb{Z},\alpha q}$ と定義されている. また, 暗号化のステップ 1 で用いる B_{m,h_r} は, m 次元の列ベクトルで, 成分のうち h_r 個が $+1$ もしくは -1, 残りが 0 となるものを一様ランダムにサンプリングしたものとする.

復号アルゴリズムのステップ 1 において, $q \gg p$ の場合には

$$\mathbf{b} + S^T\mathbf{a} \approx \frac{p}{q}\left(A^T\mathbf{r} + \frac{q}{2}\cdot m + B^T\mathbf{r}\right) = \frac{p}{2}\cdot m + \frac{p}{q}E^T\mathbf{r} \pmod p$$

がなりたち, $E^T\mathbf{r}$ が小さいために復号が高確率で成功することが直感的にわかる.

安全性の根拠となる問題は, sparse small secret LWE と呼ばれる, LWE 問題の秘密ベクトルを一様分布ではなく $\mathcal{ZO}_n(\rho)$ から取るものであるが, この問題の困難性が (適当なパラメータの変換により) 通常の LWE 問題と等しいことが証明されている [16].

2.3.4　CRYSTALS-Dilithium

CRYSTALS-Dilithium は, Ducas らにより提案された署名方式で, Module-LWE 問題と Module-SIS 問題の困難性を安全性の根拠としている. 同時に提案された KEM として, CRYSTALS-KYBER が存在する. NIST への提出 [23] と前後して, Cryptology ePrint Archive [22] および国際会議 CHES 2018 において公開された版 [24] が存在するが, ここでは NIST 版の概略を紹介する.

表 2.9 に概略を示す. 演算は全て環 $R_q := \mathbb{Z}_q[x]/(x^n + 1)$ の上で行われ, 係数は自動的に $(-\frac{q}{2}, \frac{q}{2}]$ の範囲に収められ

表 2.8　Lizard.CCA のパラメータ [21, p.22-24,31].

(n, m, ℓ, p, q, ρ)	安全性レベル	公開鍵サイズ [B]	秘密鍵サイズ [B]	平文サイズ [B]	暗号文サイズ [B]
$(536, 1024, 256, 512, 2048, 1/2)$	128 bits	1130496B	8608B	32B	17696B
$(663, 1024, 256, 256, 1024, 1/4)$	128 bits	1390592B	10640B	32B	10896B
$(816, 1024, 384, 512, 1024, 1/2)$	192 bits	1720320B	19632B	48B	26928B
$(952, 1024, 384, 512, 2048, 1/4)$	192 bits	1998848B	22896B	48B	31280B
$(1088, 2048, 512, 1024, 4096, 1/2)$	256 bits	4584520B	34880B	64B	35904B
$(1300, 2048, 512, 512, 2048, 1/4)$	256 bits	2727936B	41664B	64B	42688B

離散 Gauss 分布を表現するパラメータ α に関して, 実装時にはその累積分布関数を近似した数表として与えている.

るものとする. ノイズ生成関数 $D^{\mathsf{Dilithium}}_{R_q,\eta}$ は, R_q の元の中で係数の絶対値が η 以下のものを一様ランダムに出力する関数として定義される. 整数 $a \in \mathbb{Z}_q$ と γ に対して, $\mathsf{HighBits}(a, \gamma)$, $\mathsf{LowBits}(a, \gamma)$ 関数はそれぞれ

$$a = \mathsf{HighBits}(a, \gamma) \cdot 2\gamma + \mathsf{LowBits}(a, \gamma), \text{ ただし} \quad -\gamma < \mathsf{LowBits}(a, \gamma) \leq \gamma$$

をみたす, つまり a を上位ビットと下位ビットに切り分ける関数である[*6].

署名のステップ 4 において公開鍵から分解された $w_1 \in R_q^{k\times 1}$ を, 次のステップにおいてハッシュ関数 H を用いて別の多項式 $c \in \{\sum_{i=0}^{n-1} a_i x^i\}$ に写す. ただし, c の係数 a_i は $|a_i| \leq 1$ かつ $|a_i| = 1$ となる i が 60 個となるものの集合から取られる.

署名が正しい場合, 検証アルゴリズムのステップ 1 において, $Az - ct = Ay - cs_2$ となる. cs_2 は作り方から係数が小さいため, $\mathsf{HighBits}(Az - ct, 2\gamma_2) = \mathsf{HighBits}(Ay, 2\gamma_2)$ つまり $w_1 = w_1'$ がなりたち署名が検証される.

安全性の強さと根拠となる問題の対応がいくつか議論されている [24, Section 4]. 例えば, 鍵の安全性のみを議論する際には Module-LWE 問題の困難性のみを仮定すれば良いことが示されているが, 署名の偽造に関しては Module-SIS 問題およびその変種である SelfTargetModule-SIS 問題の困難性を用いる.

2.3.5　pqNTRUSign

pqNTRUSign は, Chen らによる NIST への提案 [17] であり, 先行する 2 つの論文 [29, 30] をベースとしている. 最初に提案された 2014 年版 [29] においては, 安全性証明が明記されていなかったが, 2017 年版 [30] では格子問題の一種である LWT (Learning with Truncation) 問題の求解困難性を署名の偽造不可能性の根拠としており, その問題が LWR 問題よりも難しいことが示されている [30, Section 3.5]. 同様に, 秘密鍵の復元に関しては SIS 問題への還元を行っている. 実際のパラメータ設定は, 問題を uSVP のインスタンスに変換した後に BKZ 2.0 による評価 [20, 2] を行っている.

また, 2014 年版と 2017 年版の大きな違いとして, 署名生成時に用いる乱数を一様分布から離散ガウス分布へと変更することで署名サイズを削減している点が挙げられる.

アルゴリズムの概要を表 2.11 に示す. より直感的な説明として, 公開鍵から作られる格子 L を用いて, メッセージ $\mu \in \mathbb{Z}_p^n$ に対する署名 w は, $v \equiv \mu \pmod p$ を満たす格子ベクトルとみることができる. この関係を NTRU 暗号をベースに構成するため, 計算を多項式環 $\mathcal{R} = \mathbb{Z}[x]/(x^n \pm 1)$ の元[*7] として行っている.

[*6] 実際には, 安全性のために境界値における挙動が調整されている [23, Figure 3]

[*7] 効率的な実装のため, 剰余多項式は $x^n + 1$ もしくは $x^n - 1$ のどちらかであるとしているが, アルゴリズムの定義中では $x^n + 1$ が用いられている.

表 2.9　CRYSTALS-Dilithium 署名方式 [23, Figure 1] より改変

パラメータ	n, k, ℓ (次元), q (法), η (ノイズ) $R_q := \mathbf{Z}_q[x]/(x^n+1)$
鍵生成 $\mathsf{KeyGen}(1^\lambda) \to (\mathsf{sk}, \mathsf{pk})$	1: $s_1 \leftarrow D^{\mathsf{Dilithium}}_{R_q^{\ell\times 1},\eta}$, $s_2 \leftarrow D^{\mathsf{Dilithium}}_{R_q^{k\times 1},\eta}$ 2: $A \xleftarrow{\$} R_q^{k\times\ell}$ 3: $t = As_1 + s_2$ 4: $\mathsf{sk} = (A, t, s_1, s_2)$, $\mathsf{pk} = (A, t)$
署名 $\mathsf{Sign}(\mathsf{sk}, M) \to \mathsf{sig}$	1: $z \leftarrow \perp$ 2: while $z = \perp$ do 3: $\quad y \leftarrow D^{\mathsf{Dilithium}}_{R_q^{\ell\times 1},\gamma_1-1}$ 4: $\quad w_1 \leftarrow \mathsf{HighBits}(Ay, 2\gamma_2)$ 5: $\quad c = H(M\|w_1)$; $z \leftarrow y + cs_1$ 6: $\quad$ if $(\|z\|_\infty \geq \gamma_1 - \beta)$ OR $(\|\mathsf{LowBits}(Ay - cs_2, 2\gamma_2)\|_\infty \geq \gamma_2 - \beta)$ then: $z = \perp$ 7: $\mathsf{sig} \leftarrow (z, c)$
検証 $\mathsf{Verify}(\mathsf{pk}, M, \mathsf{sig})$	1: $w_1' = \mathsf{HighBits}(Az - ct, 2\gamma_2)$ 2: if $(\|z\|_\infty < \gamma_1 - \beta)$ AND $(c = H(M\|w_1'))$ then accept else reject

表 2.10　CRYSTALS-Dilithium 署名方式のパラメータ [23]

(n, k, ℓ, q, η)	安全性レベル	公開鍵サイズ	秘密鍵サイズ	署名サイズ
$(256, 4, 3, 8380417, 6)$	128 bits	1184B	-	2044B
$(256, 5, 4, 8380417, 5)$	192 bits	1472B	-	2701B
$(256, 6, 5, 8380417, 3)$	256 bits	1760B	-	3366B

鍵生成アルゴリズムのステップ 1 および 2 では, 棄却サンプリングを用いて NORMF$(\cdot)$ の大きくない多項式を生成する. ここで, 多項式 $f(x) \in \mathsf{R}$ に対して NORMF(f) は多項式 $t = \sum_{i=0}^{n-1} t_i x^i := \sum_{i=0}^{n-1} x^i f(x) \pmod{x^n+1}$ に対する $\max_i |t_i|$ として定義される.

署名アルゴリズムにおいて, 入力メッセージ M の形式およびハッシュ関数は特に指定されておらず[*8] , その出力を適切に 2 つの多項式 $u_p, v_p \in R_q$ に埋め込むことができれば良い構造となっている.

ステップ 2 の $D^{\mathsf{pqNTRUSign}}_{R_q,\sigma}$ 関数は, 各係数を分散が σ^2 となる離散ガウス分布から独立に取った多項式 r を出力する. ステップ 5, 7 および 8 において様々な形でスクリーニングされた後に, 署名 sig が出力されるが, これらの条件は検証時のエラー率を下げるために行われている.

2.4　まとめ

格子に基づく暗号技術は, LWE 問題, Ring-LWE 問題, NTRU 問題を安全性の根拠とする方式がこれまで数多く提案され, NIST PQC プロジェクトの候補暗号では最も多くの暗号がこのカテゴリーに分類されている.

格子に基づく暗号技術の安全性の根拠となる問題としては、上記以外にも Compact LWE 問題, Module-LWE 問題, LWR (Learning With Rounding) 問題, BDD (Bounded Distance Decoding) 問題, SIS (Small Integer Solution) 問題他, 多くのバリエーションが存在している. 一般的な格子問題を解く手法としては, LLL アルゴリズム, BKZ アルゴ

[*8] Reference Implementation では, セキュリティレベルに応じた長さの SHA アルゴリズムを用いている.

表 2.11　pqNTRUSign 署名方式 (離散 Gauss 分布をノイズとして用いたバージョン) [17, Algorithm 1-3]

パラメータ	N (次元) p, q (法) d, s (ノイズ) B_k (鍵多項式のノルムの上界) $T(d_1, d_2) = \left\{ \sum_{i=0}^{N} a_i x^i : \begin{array}{r} a_i \in \{-1, 0, 1\} \\ \text{number of } +1 \text{ is } d_1 \\ \text{number of } -1 \text{ is } d_2 \end{array} \right\}$
鍵生成 $KeyGen(1^\lambda) \to (\mathsf{sk}, \mathsf{pk})$	1: mod q で可逆かつ NORMF$(f) < B_k$ をみたす元 f を $T(d+1, d)$ から一様にサンプリング 2: mod q で可逆かつ NORMF$(g) < B_k$ をみたす元 f を $T(d+1, d)$ から一様にサンプリング $h = h/(p \cdot f) \pmod q$ $\mathsf{sk} = (p \cdot f, g)$, $\mathsf{pk} = h$
署名 $\mathsf{Sign}(\mathsf{sk}, M) \to \mathsf{sig}$	1: $(u_p, v_p) = H(M \| h)$ 2: $r \leftarrow D^{\mathsf{pqNTRUSign}}_{R_q, \sigma}$, $b \xleftarrow{\$} \{0, 1\}$ 3: $u_1 = p \cdot r + u_p$; $v_1 = u_1 h \pmod q$ 4: $a \leftarrow (v_p - v_1) \cdot g^{-1} \pmod p$ 5: if $(\|a \cdot f\|_2 > B_s)$ OR $(\|a \cdot g\|_\infty > B_t)$ goto step 2 6: $v \leftarrow v_1 + (-1)^b a g$ 7: if $(\|v\|_\infty > q/2 - B_t)$ goto step 2 8: return $\mathsf{sig} = r + (-1)^b a \cdot f$ with probability $1/(M_s \exp(-\|a \cdot f\|/2\sigma^2) \cosh(\langle b, a \cdot f \rangle / \sigma^2)$ 9: goto step 2
検証 $\mathsf{Verify}(\mathsf{pk}, M, \mathsf{sig})$	1: $(u_p, v_p) = H(M \| h)$ 2: $u = pB + u_p$ 3: if $(\|u\|^2 > p^2 s^2 N)$ then reject 4: $v = uh \pmod q$ 5: if $(v \not\equiv v_p \pmod p)$ OR $(\|v\|_\infty > q/2 - B_t)$ then reject 6: accept

表 2.12　pqNTRUSign 署名方式のパラメータ [17].

(N, q, d, σ)	安全性レベル	公開鍵サイズ	秘密鍵サイズ	署名サイズ
$(1024, 2^{16}+1, 205, 250)$	256 bits	2048B	-	1408B

文献 [17] には “For all 5 NIST’s required security levels, we suggest the use of Gaussian-1024 or Uniform-1024 parameter sets.” とあり, 1 つのパラメータセットで全てのカテゴリを含む形で与えられている.

リズムがよく知られており, LWE 問題については更に SIS 問題や BDD 問題に還元する解析手法が知られている。

格子問題の困難性をベースとした暗号方式で最初のものは, Ajtai[1] により 1996 年に行われた, SIS 問題が格子問題の最悪時よりも困難であることの証明およびそれを用いた暗号学的ハッシュ関数の構成である. また, 1997 年には Ajtai と Dwork[8] により, unique SVP の最悪困難性を安全性の根拠とした公開鍵暗号が提案されている. この公開鍵暗号方式は翌年, Nguyen らによる解読実験 [37] により必要なパラメータが長大となり実用的でないことが明らかにされたものの, その後の格子暗号の構成の基礎となっている.

1996 年に Hoffstein らによって提案された NTRU 暗号 [28][*9] は, 発表当初安全性証明が付けられておらず, 攻撃と修正が繰り返されていたが, 2011 年 Stehlé ら [46] により方式を修正することでイデアル格子上の問題の困難性に還元可能なことが示されている. 一方で, 2016 年には subfield attack[4] のような体の構造を使って格子の次元を圧縮する

[*9] 文献上は 1998 年の国際会議 ANTS だが, 初出は CRYPTO1996 の Rump Session である.

攻撃も提案されており, 暗号の構成のためには次元や法の大きさだけでなく, 環・体の構造にも注意を払う必要がある.

2005 年に Regev[40] により提案された LWE 問題は, 論文発表と同時にそれを暗号の安全性根拠として保障する重要な三つの性質が示された. ひとつは問題の average-case to worst case reduction, つまりパラメータを固定した際, 問題の (秘密ベクトル $\mathbf{s}$ に関する) 平均的な計算量が, 最悪計算量 (難しいインスタンスを生成するような $\mathbf{s}$ の集合に対する計算量) と高々多項式倍の違いしか無いことであり, 残りの二つは判定 LWE と探索 LWE の等価性, および量子アルゴリズムによる困難な格子問題への還元である. これらの定理を組み合わせることにより, Regev 自身により提案された公開鍵暗号を解読することが平均的に難しいことが示され, その後の様々な LWE ベース暗号の構成の基礎なった. LWE 格子問題への還元に関して, 2013 年には古典計算機による還元も示されている [13].

LWE 問題の欠点である鍵サイズの大きさを改善するため, 2010 年には Lyubashevsky ら [34, 35] により Ring-LWE 問題が, 2015 年には Langlois ら [36] により Module-LWE 問題が暗号化方式と同時に提案され, LWE 問題における関係と類似の, 解読の平均的な困難さが証明されている. 一方で, これらの変種とオリジナルの LWE 問題との関係性は自明ではなく, 同程度の難しさを持つかどうかは未解決問題である. 一般的に Ring(Module)-LWE 問題のインスタンスは LWE 問題のインスタンスとして書きなおすことができるため, LWE 問題は Ring(Module)-LWE 問題よりも困難であるという関係は自明であるが, 逆の関係は知られていない. 法 q が大きい場合には, Ring-LWE は Module-LWE よりも困難であることが知られている [7].

実装時の問題として, 離散 Gauss 分布を正確に生成することは難しい. ノイズをある整数区間から一様分布として取った場合でも, 格子問題へと量子帰着が可能であることが 2013 年に Döttling ら [25] により示された. この方向性の研究として, Bai ら [12] により提案された, Rényi エントロピーを用いた, 理想的な Gauss 分布を用いた暗号方式とそれを近似的な分布に置き換えた方式の間での安全性の低下を議論するものがある.

また, パラメータ設定手法に関して 2016 年より暗号解読コンテスト LWE Challenge[50] が開催されている.

これらの格子に基づく暗号技術の安全性の根拠となる問題は, 古典計算機・量子計算機のいずれにおいても現時点で効率的な解読手法は見つかっていない.

第 2 章の参考文献

[1] M. Ajtai, Generating hard instances of lattice problems, *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing*, pp. 99-108, 1996.

[2] E. Alkim, R. Avanzi, J. Bos, L. Ducas, A. de la Piedra, T. Pöppelmann, P. Schwabe, D. Stebila, NewHope Algorithm Specifcations, and Supporting Documentation, `https://newhopecrypto.org/data/NewHope_2017_12_21.pdf`

[3] E. Alkim, R. Avanzi, J. Bos, L. Ducas, A. de la Piedra, T. Pöppelmann, P. Schwabe, D. Stebila, NewHope Algorithm Specifications, and Supporting Documentation (Version 1.01), Updated December 2, 2018, `https://newhopecrypto.org/data/NewHope_2018_12_02.pdf`

[4] M. Albrecht, S. Bai, L. Ducas, A subfield lattice attack on overstretched NTRU assumptions, *Advances in Cryptology – CRYPTO 2016 – 36th Annual International Cryptology Conference*, Part I, Springer LNCS vol. 9814, pp. 153-178, 2016.

[5] Y. Aono, X. Boyen, L. T. Phong, L Wang, Key-private proxy re-encryption under LWE, *Progress in Cryptology – INDOCRYPT 2013 – 14th International Conference on Cryptology in India*, Springer LNCS vol. 8250, pp. 1-18, 2013.

[6] M. R. Albrecht, B. R. Curtis, A. Deo, A. Davidson, R. Player, E. W. Postlethwaite, F. Virdia, T. Wunderer, Estimate all the {LWE, NTRU} schemes!(2018), *Security and Cryptography for Networks – 11th International Conference*, Springer LNCS vol. 11035, pp. 351-367, 2018.

[7] M. Albrecht, A. Deo, Large modulus Ring-LWE $\geq$ Module-LWE, *Advances in Cryptology – ASIACRYPT 2017 – 23rd International Conference on the Theory and Applications of Cryptology and Information Security*, Part I, Springer LNCS vol. 10624, pp. 267-296, 2017.

[8] M. Ajtai, C. Dwork, A public-key cryptosystem with worst-case/average-case equivalence, *Proceedings of the Twenty-Ninth Annual ACM Symposium on the Theory of Computing*, pp. 284-293, 1997.

[9] E. Alkim, L. Ducas, T. Pöppelmann, P. Schwabe, NewHope without reconciliation, *Cryptology ePrint Archive*, Report 2016/1157.

[10] E. Alkim, L. Ducas, T. Pöppelmann, P. Schwabe, Post-quantum key exchange – A New Hope, *the 25th USENIX Security Symposium*, pp. 327-343, 2016.

[11] L. G. Bruinderink, A. Hülsing, T. Lange, Y. Yarom, Flush, gauss, and reload – A cache attack on the BLISS lattice-based signature scheme, *Cryptographic Hardware and Embedded Systems – CHES 2014 – 16th International Workshop*, Springer LNCS vol. 9813, pp. 323-345, 2016.

[12] S. Bai, T. Lepoint, A. R.-Langlois, A. Sakzad, D. Stehlé, R. Steinfeld, Improved security proofs in lattice-based cryptography: Using the Rényi divergence rather than the statistical distance, *Journal of Cryptology*,

vol. 31, Iss. 2, pp. 610-640, 2018.

[13] Z. Brakerski, A. Langlois, C. Peikert, O. Regev, D. Stehlé, Classical hardness of learning with errors, *Symposium on Theory of Computing Conference, STOC'13*, pp. 575-584, 2013.

[14] A. Banerjee, C. Peikert, A. Rosen, Pseudorandom functions and lattices, *Advances in Cryptology – EUROCRYPT 2012 – 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer LNCS vol. 7237, pp. 719-737, 2012.

[15] Z. Brakerski, V. Vaikuntanathan, Fully homomorphic encryption from ring-LWE and security for key dependent messages, *Advances in Cryptology – CRYPTO 2011 – 31st Annual Cryptology Conference*, Springer LNCS vol. 6841, pp. 505-524, 2011.

[16] J. H. Cheon, K. Han, J. Kim, C. Lee, Y. Son, A practical post-quantum public-key cryptosystem based on spLWE, *Information Security and Cryptology – ICISC 2016 – 19th International Conference*, Springer LNCS vol. 10157, pp. 51-74.

[17] C. Chen, J. Hoffstein, W. Whyte, Z. Zhang, NIST PQ Submission: pqNTRUSign A modular lattice signature scheme, `https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/pqNTRUsign.zip`

[18] J. H. Cheon, D. Kim, J. Lee, Y. Song, Lizard: Cut off the Tail! A Practical Post-Quantum Public-Key Encryption from LWE and LWR, *Cryptology ePrint Archive*, Report 2016/1126.

[19] J. H. Cheon, D. Kim, J. Lee, Y. Song, Lizard: Cut off the Tail! A Practical Post-Quantum Public-Key Encryption from LWE and LWR, *Security and Cryptography for Networks – 11th International Conference*, Springer LNCS vol. 11035, pp. 160-177, 2018.

[20] Y. Chen, P. Q. Nguyen, BKZ 2.0: Better lattice security estimates, *Advances in Cryptology – ASIACRYPT 2011 – 17th International Conference on the Theory and Application of Cryptology and Information Security*, Springer LNCS vol. 7073, pp. 1-20.

[21] J. H. Cheon, S. Park, J. Lee, D. Kim, Y. Song, S. Hong, D. Kim, J. Kim, S.-M. Hong, A. Yun, J. Kim, H. Park, E. Choi, K. kim, J.-S. Kim, J. Lee Lizard Public Key Encryption Submission to NIST proposal, Algorithm specification, `https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/Lizard.zip`

[22] L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, D. Stehlé, CRYSTALS – Dilithium: Digital Signatures from Module Lattices, *Cryptology ePrint Archive*, Report 2017/633.

[23] L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, D. Stehlé, CRYSTALS-Dilithium Algorithm Specifications and Supporting Documentation, `https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/CRYSTALS_Dilithium.zip`

[24] L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, D. Stehlé, CRYSTALS – Dilithium: Digital Signatures from Module Lattices, *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2018(1), pp. 238-268, 2018.

[25] N. Döttling, J. M.-Quade, Lossy codes and a new variant of the learning-with-errors problem, *Advances in Cryptology – EUROCRYPT 2013 – 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer LNCS vol. 7881, pp. 18-34, 2013.

[26] E. Fujisaki, T. Okamoto, Secure integration of asymmetric and symmetric encryption schemes, *Advances in Cryptology – CRYPTO '99, 19th Annual International Cryptology Conference*, Springer LNCS vol. 1666, pp.

537-554, 1999.

[27] E. Fujisaki, T. Okamoto, Secure integration of asymmetric and symmetric encryption schemes, *Journal of Cryptology*, vol.26, No.1, pp. 80-101, 2013.

[28] J. Hoffstein, J. Pipher, J. H. Silverman, NTRU: A ring-based public key cryptosystem, *Algorithmic Number Theory, Third International Symposium, ANTS-III*, Springer LNCS vol. 1423, pp. 267-288, 1998.

[29] J. Hoffstein, J. Pipher, J. M. Schanck, J. H. Silverman, W. Whyte, Transcript secure signatures based on modular lattices, *Post-Quantum Cryptography – 6th International Workshop*, Springer LNCS vol. 8772, pp. 142-159, 2014.

[30] J. Hoffstein, J. Pipher, W. Whyte, Z. Zhang, A signature scheme from learning with truncation, *Cryptology ePrint Archive*, Report 2017/995.

[31] A. K. Lenstra, H. W. Lenstra, L. Lovász, Factoring polynomials with rational coefficients, *Mathematische Annalen*, 261(4), pp. 515-534, 1982.

[32] K. Lauter, M. Naehrig, V. Vaikuntanathan, Can homomorphic encryption be practical?, *ACM workshop on Cloud computing security workshop–CCSW 2011*, ACM, pp. 113-124, 2011.

[33] R. Lindner, C. Peikert, Better Key sizes (and attacks) for LWE-Based encryption, *Topics in Cryptology – CT-RSA 2011 – The Cryptographers' Track at the RSA Conference 2011*, Springer LNCS vol. 6558, pp. 319-339.

[34] V. Lyubashevsky, C. Peikert, O. Regev, On ideal lattices and learning with errors over rings, *Advances in Cryptology – EUROCRYPT 2010 – 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer LNCS vol. 6110, pp. 1-23, 2010.

[35] V. Lyubashevsky, C. Peikert, O. Regev, On ideal lattices and learning with errors over rings, *Journal of the ACM*, Vol. 60, Iss. 6, pp. 1-35, 2013.

[36] A. Langlois, D. Stehlé, Worst-case to average-case reductions for module lattices, *Designs, Codes and Cryptography*, Vol. 75, Iss. 3, pp. 565-599, 2015.

[37] P. Q. Nguyen, J. Stern, Cryptanalysis of the Ajtai-Dwork cryptosystem, *Advances in Cryptology – CRYPTO '98, 18th Annual International Cryptology Conference*, Springer LNCS vol. 1462, pp. 223-242, 1998.

[38] L. T. Phong, T. Hayashi, Y. Aono, S. Moriai, LOTUS specification `https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/LOTUS.zip`

[39] C. Peikert, V. Vaikuntanathan, B. Waters, A framework for efficient and composable oblivious transfer, *Advances in Cryptology – CRYPTO 2008, 28th Annual International Cryptology Conference*, Springer LNCS vol. 5157, pp. 554-571, 2008.

[40] O. Regev, On lattices, learning with errors, random linear codes, and cryptography, *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pp. 84-93.

[41] O. Regev, On lattices, learning with errors, random linear codes, and cryptography, *Journal of the ACM*, Vol. 56, Iss. 6, pp. 1-40, 2009.

[42] O. Regev, The Learning with Errors Problem, `https://cims.nyu.edu/~regev/papers/lwesurvey.pdf`

[43] O. Regev, The learning with errors problem (invited survey), *Proceedings of the 25th Annual IEEE Conference on Computational Complexity, CCC 2010*, pp. 191-204, 2010

[44] M. Roşca, A. Sakzad, D. Stehlé, R. Steinfeld, Middle-product learning with errors, *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference*, Part III, Springer LNCS vol. 10403,

pp. 283-297, 2017.

[45] C. P. Schnorr, M. Euchner, Lattice basis reduction: Improved practical algorithms and solving subset sum problmes, *Mathematical programming*, 66, pp. 181-199, 1994.

[46] D. Stehlé, R. Steinfeld, Making NTRU as secure as worst-case problems over ideal lattices, *Advances in Cryptology – EUROCRYPT 2011 – 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer LNCS vol. 6632, pp. 27-47, 2011.

[47] D. Stehlé, R. Steinfeld, Making NTRUEncrypt and NTRUSign as secure as standard worst-case problems over ideal lattices, *Cryptology ePrint Archive*, Report 2013/004.

[48] D. Stehlé, R. Steinfeld, K. Tanaka, K. Xagawa, Efficient public key encryption based on ideal lattices, *Advances in Cryptology – ASIACRYPT 2009 – 15th International Conference on the Theory and Application of Cryptology and Information Security*, Springer LNCS vol. 5912, pp. 617-635, 2009.

[49] N. P. Smart, F. Vercauteren, Fully homomorphic SIMD operations, *Designs, Codes and Cryptography*, Vol, 71, Iss. 1, pp. 57?81, 2014.

[50] TU Darmstadt UC San Diego, LWE Challenge, `https://www.latticechallenge.org/lwe_challenge/challenge.php`

第 3 章

符号に基づく暗号技術

本章では符号に基づく暗号技術についてまとめる. 符号に基づく暗号技術の安全性はシンドローム復号問題を解く計算の困難性に依存している.

■**準備**: 本章で使用する記号・用語を以下にまとめる. 以下では, q を素数 p の冪とする.

有限体: $\mathbb{F}_q$ で位数が q の有限体を表す.

ハミング重みとハミング距離: V_n を有限体 $\mathbb{F}_{q^m}$ 上の n 次元ベクトル空間とする.

- ベクトル $v = (v_1, v_2, \ldots, v_n) \in V_n$ のハミング重みとは, 非ゼロの係数の数である. すなわち, $\mathsf{HW}(v) = \#\{v_i \mid v_i \neq 0\}$.
- ハミング距離を $d_H(x, y) = \mathsf{HW}(x - y)$ で定義する.
- $\mathcal{S}_H(n, t)$ でハミング重みが t の n 次元ベクトル全体の集合を表す.

ランク重みとランク距離: V_n を有限体 $\mathbb{F}_{q^m}$ 上の n 次元ベクトル空間とし, $\beta = (\beta_1, \beta_2, \ldots, \beta_m)$ を $\mathbb{F}_{q^m}$ の $\mathbb{F}_q$ - 基底とする. $f_i \colon \mathbb{F}_{q^m} \to \mathbb{F}_q$ を, $x \in \mathbb{F}_{q^m}$ に対して, $f_i(x)$ を基底 β の下での i 番目の係数とする. $v = (v_1, v_2, \ldots, v_n) \in V_n$ に対して, 行列 $\bar{v} \in \mathcal{M}_{m,n}(\mathbb{F}_q)$ を, $\bar{v}_{ij} = f_i(v_j)$ で定義する.

- ベクトル v のランク重みとは, 対応する行列 $\bar{v}$ のランク $\mathrm{rank}(\bar{v})$ で定義される.
- ランク距離を $d_R(x, y) = \mathrm{rank}(\overline{x - y})$ で定義する.
- $\mathcal{S}_R(n, t)$ でランク重みが t の n 次元ベクトル全体の集合を表す.

3.1 符号に基づく暗号技術の安全性の根拠となる問題

本節では Learning Parity with Noise （LPN） 問題や符号に関連する問題の困難性について調査結果を述べる.

3.1.1 LPN 問題とは

LPN 問題とは誤差付きの線形方程式を解けるかどうかという問題である. 1993 年に, Blum, Furst, Kearns, Lipton [9] が困難と思われる問題として挙げ, 定式化を行った. 2 章において, この問題を一般化した LWE 問題を既に扱っている.

Ber_τ でパラメータ τ のベルヌーイ分布を表すことにする. (確率 τ で 1, 確率 $1 - \tau$ で 0 となる $\mathbb{F}_2$ 上の分布である.) また, 自然数 $k \geq 1$ について, Ber_τ^k で, Ber_τ から独立に k 個サンプルを取ったときの $\mathbb{F}_2^k$ 上の分布を表す.

■ LPN 問題: $\mathbb{F}_2$ 上の分布 χ および $\boldsymbol{s} \in \mathbb{F}_2^k$ について, オラクル $\mathcal{O}_{\boldsymbol{s},\chi}$ を以下で定義する. (1) $\boldsymbol{a}$ を $\mathbb{F}_2^k$ からランダムに選び, (2) e を分布 χ に従い選び, (3) $b = \boldsymbol{s} \cdot \boldsymbol{a}^\top + e$ と計算し, (4) $(\boldsymbol{a}, b)$ を出力する. 定義より, また, オラクル $\mathcal{U}$ を $(\boldsymbol{a}, b) \leftarrow \mathbb{F}_2^{k+1}$ とランダムな組を出力するオラクルとして定義する.

定義 3.1 (探索版 LPN 問題) 探索版 LPN 問題とは, オラクル $\mathcal{O}_{\boldsymbol{s},\chi}$ へのアクセスが可能なときに, $\boldsymbol{s}$ を出力する問題である.

特に $\chi = \mathsf{Ber}_\tau$ のとき, $\mathsf{LPN}_{k,\tau}$ 問題と呼ぶ. また $\mathsf{LPN}_{k,\tau}$ 問題でオラクルからのサンプル数が $n = n(k)$ に制限されるものを, $\mathsf{LPN}_{k,n,\tau}$ 問題と呼ぶ.

定義 3.2 (探索版 LPN 仮定) $\mathbb{F}_2$ 上の確率分布 χ について, 敵 $\mathcal{A}$ の優位性を

$$\mathsf{Adv}_{\mathcal{A}}(k) = \Pr_{\boldsymbol{s} \leftarrow \mathbb{F}_2^k}[\mathcal{A}^{\mathcal{O}_{\boldsymbol{s},\chi}}(1^k) = \boldsymbol{s}]$$

で定義する. 任意の多項式時間の敵 $\mathcal{A}$ について, その優位性が無視できるとき, 探索版 LPN 仮定が成立するという.

暗号プリミティブや暗号プロトコルの安全性証明のために, 判定版 LPN 仮定を用いることも多い. 判定版 LPN 問題と判定版 LPN 仮定は以下で定義される.

定義 3.3 (判定版 LPN 問題) 判定版 LPN 問題とは, オラクル $\mathcal{O}_{\boldsymbol{s},\chi}$ またはオラクル $\mathcal{U}$ へのアクセスが与えられたときに, どちらのオラクルにアクセスしているかを判定する問題である.

定義 3.4 (判定版 LPN 仮定) $\mathbb{F}_2$ 上の確率分布 χ について, 敵 $\mathcal{A}$ の優位性を

$$\mathsf{Adv}_{\mathcal{A}}(k) = \left| \Pr_{\boldsymbol{s} \leftarrow \mathbb{F}_2^k}[\mathcal{A}^{\mathcal{O}_{\boldsymbol{s},\chi}}(1^k) = 1] - \Pr[\mathcal{A}^{\mathcal{U}}(1^k) = 1] \right|$$

で定義する. 任意の多項式時間の敵 $\mathcal{A}$ について, その優位性が無視できる関数であるとき, 判定版 LPN 仮定が成立するという.

探索版 LPN 問題にはランダム自己帰着が存在する [9]. すなわち, ランダムに選ばれた $\boldsymbol{s} \in \mathbb{F}_2^k$ について探索版 LPN 問題を解けるならば, 任意の $\boldsymbol{s} \in \mathbb{F}_2^k$ について探索版 LPN 問題を解くことが出来る.

Katz, Shin, Smith [32] によれば, [9, 46] と同様に判定版 LPN 仮定を探索版 LPN 仮定に帰着することが出来る.

定理 3.5 ([32]) 判定版 $\mathsf{LPN}_{k,\tau}$ 仮定を破る t ステップ, m 回のクエリ, 優位性 δ の敵が存在すると仮定する. このとき, 探索版 $\mathsf{LPN}_{n,\tau}$ 仮定を破る t' ステップ, m' 回のクエリ, 優位性 δ' の敵が存在する. ここで,

$$t' = O(\delta^{-2} tk \log k),\ m' = O(\delta^{-2} m \log k),\ \delta' \geq \delta/4.$$

■変種: 以上に列挙した LPN 問題・仮定では, 基礎となる体として $\mathbb{F}_2$ を用いていた. 体を $\mathbb{F}_q$ に変更した LPN 問題・仮定が用いられることもある. 特に q を素数とした場合には LWE 問題と非常によく似た問題・仮定となるが, 誤差分布 χ の定義が異なることが多い.

LWE 問題では剰余環 $\mathbb{Z}_q$ を用いている. 応用の観点からは, 誤差分布 χ からのサンプル x の絶対値が高い確率で小さいことが重視される. 一方, LPN 問題では有限体 $\mathbb{F}_q$ を用いている. また, 符号からの要求としてハミング重みを考えることが多いため, 誤差分布 χ は 0 を取る確率が大きいことが求められる. たとえば, ベルヌーイ分布の一般化として, 確率 τ で 0 を確率 $1-\tau$ で $\mathbb{F}_q \setminus \{0\}$ のランダムな値を取る分布が用いられる. これは格子問題と符号問題のアナロジーとして考えることができる.

3.1.2 LPN 問題の拡張

3.1.2.1 復号問題

オラクルからのサンプル数を固定し $n = n(k)$ とする．$\mathsf{LPN}_{k,n,\tau}$ 問題での m 個のサンプル $(\boldsymbol{a}_1, b_1), (\boldsymbol{a}_2, b_2), \ldots, (\boldsymbol{a}_n, b_n)$ を行列・ベクトル表示して，

$$\boldsymbol{A} = [\boldsymbol{a}_1^\top \boldsymbol{a}_2^\top \ldots \boldsymbol{a}_n^\top] \in \mathbb{F}_2^{k \times n}, \boldsymbol{b} = \boldsymbol{s} \cdot \boldsymbol{A} + \boldsymbol{e}$$

とする．符号理論の観点からは，ランダム行列 $\boldsymbol{A} \in \mathbb{F}^{k \times n}$ を生成行列[*1] とする $[n, k]_q$ -線形符号の受信語 $\boldsymbol{b}$ から元のメッセージ $\boldsymbol{s}$ を復元する問題と捉えることができる．

3.1.2.2 シンドローム復号問題

先ほど挙げた復号問題の “双対” として，シンドローム復号問題が挙げられる．シンドローム復号問題 $\mathsf{SD}_{k,n,w}$ とは，

$$\boldsymbol{H} = [\boldsymbol{h}_1^\top \boldsymbol{h}_2^\top \ldots \boldsymbol{h}_n^\top] \in \mathbb{F}_2^{(n-k) \times n}, \boldsymbol{u} \in \mathbb{F}_2^k$$

および自然数 w が与えられた時に，$\boldsymbol{e} \cdot \boldsymbol{H}^\top = \boldsymbol{u}$ かつハミング重みが w 以下となる $\boldsymbol{e} \in \mathbb{F}_2^n$ を求める問題である．

$\boldsymbol{A} \in \mathbb{F}^{k \times n}$ で生成される符号のパリティ検査行列[*2] を $\boldsymbol{H} \in \mathbb{F}^{(n-k) \times n}$ とし，$\boldsymbol{b} \cdot \boldsymbol{H}^\top (= \boldsymbol{e} \cdot \boldsymbol{H}^\top)$ を $\boldsymbol{u}$ とすれば，$\mathsf{LPN}_{k,n,\tau}$ 問題や復号問題をシンドローム復号問題 $\mathsf{SD}_{k,n,O(\tau n)}$ に変換可能である．

3.1.2.3 Exact-LPN 問題

誤差分布として，$\boldsymbol{e} \leftarrow \mathsf{Ber}_\tau^n$ ではなく，ハミング重みが丁度 w のものだけを考える．(すなわち $\boldsymbol{e} \leftarrow \mathcal{S}_H(n, w)$.) このように誤差分布を変えた問題を Exact-LPN 問題と呼ぶ．

3.1.2.4 Sparse-LPN 問題

一部の暗号方式では，$\boldsymbol{s}$ のハミング重みが小さい，すなわち，疎 (sparse) であることを要求する．Applebaum ら [3] は $\boldsymbol{s}$ を誤差分布である χ^k から選んだ場合の LPN 問題と $\boldsymbol{s}$ を $\mathbb{F}^k$ からランダムに選んだ場合の問題とが等価であることを示している．

3.1.2.5 Toeplitz-LPN 問題

Gilbert, Robshaw, Seurin [26] が認証プロトコルの効率化のために導入した．

行列 $\boldsymbol{A} = \{a_{i,j}\} \in \mathbb{F}_2^{k \times n}$ が Toeplitz 行列であるとは，任意の i, j について $a_{i-1,j-1} = a_{i,j}$ が成立することである．Toeplitz 行列を表現するには左端の列ベクトルおよび最も上の行ベクトルがあれば良い．そのため $\boldsymbol{A}$ の表現は $k + n - 1$ ビットで可能である．

復号問題の節で，探索版 LPN 問題でのサンプル $(\boldsymbol{a}_1, b_1), (\boldsymbol{a}_2, b_2), \ldots, (\boldsymbol{a}_n, b_n)$ を行列・ベクトル表示して，

$$\boldsymbol{A} = [\boldsymbol{a}_1^\top \boldsymbol{a}_2^\top \ldots \boldsymbol{a}_n^\top] \in \mathbb{F}^{k \times n}, \boldsymbol{b} = \boldsymbol{s} \cdot \boldsymbol{A} + \boldsymbol{e}$$

を考えた．オラクル $\mathcal{O}$ (および $\mathcal{U}$) を変更し，$\boldsymbol{A}$ が必ず Toeplitz 行列になる場合の LPN 問題を考える．これを Toeplitz-LPN 問題と呼ぶ．

[*1] $[n, k]_q$ -線形符号 $\mathcal{C}$ の生成行列とは，符号 $\mathcal{C}$ の基底ベクトルを行とする行列 $\boldsymbol{G} \in \mathbb{F}_q^{k \times n}$ である．

[*2] $[n, k]_q$ -線形符号 $\mathcal{C}$ のパリティ検査行列とは，行列 $\boldsymbol{H} \in \mathbb{F}_q^{r \times n}$ で，$\boldsymbol{c} \in \mathbb{F}_q^n$ に対して，$\boldsymbol{c} \in \mathcal{C}$ ならばかつその時に限り $\boldsymbol{c} \cdot \boldsymbol{H}^\top = \boldsymbol{0}$ となるものである．$\boldsymbol{H}$ の行が線形独立であれば，$r = n - k$ である．

3.1.2.6 Ring-LPN 問題

Heyse, Kiltz, Lyubashevsky, Paar, Pietrzak [28] は, Ring-LPN 問題を定義した. この問題は Ring-LWE 問題 (2.1.1.1 節) と同様に定義される.

定義 3.6 (探索版 Ring-LPN 問題) 適当な k 次の $\mathbb{F}_q$ 係数多項式 $f(x)$ を考え, 環 $R_q = \mathbb{F}_q[x]/(f(x))$ を固定する. R_q 上の確率分布 χ を固定する.

R_q 上の誤差分布 χ および $s \in R_q$ について, オラクル $\mathcal{O}_{\boldsymbol{s},\chi}$ を以下で定義する. (1) a を R_q からランダムに選び, (2) e を分布 χ に従い選び, (3) $b = sa + e$ と計算し, (4) $(a, b) \in R_q^2$ を出力する.

探索版 Ring-LPN 問題とは, オラクル $\mathcal{O}_{s,\chi}$ へのアクセスが可能なときに, $s \in R_q$ を出力する問題である.

3.1.3 LPN 問題に対する評価

サンプル数を固定した場合, $\boldsymbol{A}$ および $\boldsymbol{b}$ の最悪時を考えると NP 困難になることが Berlekamp, McEliece, van Tilborg [16] によって示されている. また, Håstad [27] により近似版 LPN 問題の NP 困難性も示されている.

しかし平均時の困難性についてはよく分かっていない. そのため LPN 問題を解くための提案されたアルゴリズムについて調査を行った.

$\mathsf{LPN}_{k,n,\tau}$ 問題を解くための素朴な方法として, 時間 $\mathrm{poly}(n,k) \cdot O(2^n)$ で動作する総当たり法がある. 閾値 $d \geq 1$ を固定する. $\boldsymbol{s} \in \mathbb{F}_2^k$ の候補ごとに, $\boldsymbol{e} = \boldsymbol{b} - \boldsymbol{s}\boldsymbol{A}$ を計算し, $\boldsymbol{e}$ のハミング重みが $(1 + 1/d)\tau n$ 以下であれば $\boldsymbol{s}$ を解として出力するというものである. Chernoff の補題[*3]から $\boldsymbol{e} \leftarrow \mathsf{Ber}_\tau^n$ としたとき, $d \geq 1$ について $\Pr[\mathsf{HW}(\boldsymbol{e}) \leq (1 + 1/d)\tau n] \leq \exp(-\tau n/3d^2)$ である.

以降では, $O(2^n)$ 以下の時間で解を求めるアルゴリズムについて考察する. 現在, 大別して以下の 3 つのアルゴリズムが知られている.

1. Blum, Kalai, Wasserman [11] の BKW アルゴリズム
2. Arora, Ge [4] の「再線形化」アルゴリズム
3. シンドローム復号問題として解くアルゴリズム

3.1.4 BKW アルゴリズムおよびその改良

Blum, Kalai, Wasserman [11] は BKW アルゴリズムと呼ばれるアルゴリズムを提案した.

基本アイデアは以下である. オラクルからのサンプル $(\boldsymbol{a}, b)$ が常に $\boldsymbol{a} = (1, 0, \ldots, 0)$ という形であれば, $b = s_1 + e$ となる. このようなサンプルを大量に集めれば, s_1 を多数決法で求めることが出来る. 一般に $\boldsymbol{u}_j$ を j 番目の単位ベクトルとして, $(\boldsymbol{u}_j, b)$ という形のサンプルを集めれば s_j を多数決法で求められる. そこでオラクル $\mathcal{O}_{\boldsymbol{s},\tau}$ からのサンプルを用いて, 上記のようなサンプルを生成することを目指す.

■BKW アルゴリズムの概要: $(t-1)\ell < k \leq t\ell$ を満たす適当な自然数 t, ℓ を固定する. 以下では,

$$A_{\boldsymbol{s},\delta,i} = \{\boldsymbol{a} \leftarrow \mathbb{F}_2^{k-i\ell} \times \{0\}^{i\ell}, e \leftarrow \mathsf{Ber}_{(1+\delta)/2} : (\boldsymbol{a}, \boldsymbol{s} \cdot \boldsymbol{a}^\top + e)\}$$

[*3] $X_1, X_2, \ldots, X_n$ を相互に独立な $\{0,1\}$ 確率変数とし, $X = X_1 + X_2 + \cdots + X_n$ の期待値を μ とすると, $\Pr[X \geq (1+\epsilon)\mu] < \left(\exp(\epsilon)/(1+\epsilon)^{(1+\epsilon)}\right)^\mu$. 特に, $0 < \epsilon \leq 1$ の場合, $\Pr[X \geq (1+\epsilon)\mu] < \exp(-\epsilon^2\mu/3)$.

というオラクルを考える．$A_{\boldsymbol{s},\delta,i}$ から得たサンプル $(\boldsymbol{a},b)$ は $\boldsymbol{a}$ の末尾から $i\ell$ 個の要素が必ず 0 である．$i=0$, $\delta=1-2\tau$ とすれば, $A_{\boldsymbol{s},\delta,i}=\mathcal{O}_{\boldsymbol{s},\tau}$ となる．

基本アルゴリズムは以下である．

1. $A_{\boldsymbol{s},\delta_0,0}=\mathcal{O}_{\boldsymbol{s},\tau}$ からのサンプルを L_0 個用意する．
2. $i=0,1,\ldots,t-2$ について, サイズ L_i の $A_{\boldsymbol{s},\delta_i,i}$ からのサンプルを用いて, $O(L_i)$ 時間でサイズ $L_{i+1}=L_i-2^k$ の $A_{\boldsymbol{s},\delta_i^2,i+1}$ からのサンプルを構成する．
   - サンプル $(\boldsymbol{a},b)\in L_i$ について, $\boldsymbol{a}=(a_1,a_2,\ldots,a_{k-i\ell},0,\ldots,0)\in\mathbb{F}_2^k$ の $(a_{k-(i+1)\ell+1},a_{k-(i+1)\ell+2},,\ldots,a_{k-i\ell})\in\mathbb{F}_2^\ell$ に従って分類を行う．
   - 各組で代表を一つとり, それを $(\boldsymbol{a}^*,b^*)$ とする．
   - 各組の代表以外の要素 $(\boldsymbol{a},b)$ を $(\boldsymbol{a}\oplus\boldsymbol{a}^*,b\oplus b^*)$ で置き換える．
   - 全組をまとめてサイズ L_i-2^ℓ の $A_{\boldsymbol{s},\delta_i^2,i+1}$ からのサンプルとする．

   最終的に, サイズ $L_{t-1}=L_0-(t-1)2^\ell$ の $A_{\boldsymbol{s},\delta_0^{2^{t-1}},t-1}$ からのサンプルが得られる．
3. 得られた L_{t-1} 個の $A_{\boldsymbol{s},\delta_0^{2^{t-1}},t-1}$ からのサンプルを用いて, s_j を投票で決める．
   - $j=1,2,\ldots,k-(t-1)\ell$ について, $\boldsymbol{u}_j$ を $\mathbb{F}_2^k$ の標準基底 j 番目の単位ベクトルとする．サンプル $\{(\boldsymbol{a}_i,b_i)\}_{i=1,2,\ldots,m}$ から z 個のベクトルを $\boldsymbol{a}_{i_1}+\boldsymbol{a}_{i_2}+\cdots+\boldsymbol{a}_{i_z}=\boldsymbol{u}_j$ となるようにうまく選ぶ．このとき, $b_{i_1}+b_{i_2}+\cdots+b_{i_z}=s_j+e_{i_1}+\cdots+e_{i_z}$ となり, 誤差が 0 になる確率は $\Pr[e_{i_1}+e_{i_2}+\ldots+e_{i_z}=0]>1/2+(1-2\delta_0^{2^{t-1}})^z/2$ で与えられる．適当な回数この試行を行い, s_j を多数決投票で決めれば良い．

Blum らの見積もりでは, サンプル数および計算ステップ数は $\delta_0=1-2\tau$ として, $\mathrm{poly}\left(\delta_0^{-2^t},2^\ell\right)$ であった．$\tau<1/2$ を定数とし, $t=\frac{1}{2}\log k$, $\ell=2k/\log k$ とすれば, 時間計算量・空間計算量ともに $2^{O(k/\log k)}$ を得る．

■LF アルゴリズム: Levieil と Fouque [36] は BKW アルゴリズムの一部を改良し LF1 アルゴリズムを提案した．

簡単のために $k=t\ell$ を仮定する．BKW アルゴリズムでは基本アルゴリズムのステップ 3 において $\boldsymbol{s}$ の各要素を 1 つずつ決定している．ステップ 3 において得られたサンプルは, $A_{\boldsymbol{s},\delta_0^{2^{t-1}},t-1}$ からのサンプルであるため, $((a_1,a_2,\ldots,a_k,0,\ldots,0),b)$ という形をしている．このとき, $b=\sum_{i=1}^{\ell}a_is_i+e$ となり, サンプルに影響を与えるのは, $\boldsymbol{s}$ の ℓ ビット分である．LF アルゴリズムでは, $s_1,s_2,\ldots,s_\ell$ を総当りで計算する．

Levieil と Fouque は BKW アルゴリズムおよび LF1 アルゴリズムが必要とするサンプル数および計算ステップ数を, 以下のように詳細に解析した．[*4]

定理 3.7 $k=t\ell$ とし, $\delta=1-2\tau$ とする．

- *BKW* アルゴリズムはクエリ数 $n=20\ln(4k)2^\ell\delta^{-2^t}$, ステップ数 $T=O(ktn)$, メモリ量 $M=kn$, 成功確率 $\theta=1/2$ で $\mathsf{LPN}_{k,n,\tau}$ 問題を解く．
- *LF1* アルゴリズムはクエリ数 $n=(8\ell+200)\delta^{-2^t}+(t-1)2^\ell$, ステップ数 $T=O(ktn)$, メモリ量 $M=kn+\ell 2^\ell$. 成功確率 $\theta=1/2$ で $\mathsf{LPN}_{k,n,\tau}$ 問題を解く．

Levieil と Fouque は, LF1 アルゴリズムに一部のヒューリスティクを組み合わせた LF2 アルゴリズムも提案している．報告によれば, $k=99$, $\tau=1/4$, $n=10000$ の LPN 問題を CPU: Pentium 4 (3GHz), RAM: 1GB のマシンで解くことが可能である．Devadas, Ren, Xiao [19] は LF2 アルゴリズムについて詳細な解析を与え, BKW アルゴリズムとの比較を行っている．Devadas らの報告によれば, メモリを $O(\delta^{-2^t})$ 倍多く使うが, 時間計算量が $O(\delta^{-2^t})$ 倍改善され

[*4] 後に, Zhang, Jiao, Wang [48] らにより, この解析にはヒューリスティックが必要との指摘があった．

るとのことである.

■Kirchner の指摘: Kirchner [29] はランダムに選ばれた $\boldsymbol{s}$ よりは Ber_τ に従って選ばれる誤りベクトル $\boldsymbol{e}$ の方が, ハミング重みが小さく, 取りうる値が少ないことに着目した. そこで, LPN 問題を Sparse-LPN 問題に置き換えた上で問題を解くことを提案している.

Kirchner の手法は以下のようにまとめられる.

1. Applebaum ら [3] と同様の手法を用いて, $\mathcal{O}_{\boldsymbol{s},\chi}$ というオラクルを $\boldsymbol{e}' \leftarrow \mathsf{Ber}_\tau^k$ とランダムに選んだ場合の $\mathcal{O}_{\boldsymbol{e}',\chi}$ というオラクルに変換する.
2. BKW アルゴリズムや LF アルゴリズムと同様に基本アルゴリズムのステップ 1, 2 を行い, $A_{\boldsymbol{e}',\delta^{2^{t-1}},t-1}$ からのサンプルを得る.
3. ステップ 3 で, ℓ ビットを決定する際に, $\boldsymbol{e}'$ の該当部分の重みが少ないことを考慮して総当りを行う.
4. ステップ 1 の逆を行い, $\boldsymbol{e}'$ を $\boldsymbol{s}$ に戻す.

一般の $\boldsymbol{s}$ であれば, 総当りに必要な回数は 2^ℓ となる. 一方, $\boldsymbol{e}'$ はスパースであることが期待される. $d \geq 1$ を固定し ℓ が十分に大きいとする. このとき, 圧倒的な確率の下で, ハミング重みは $(1+1/d)\tau\ell$ 以下である. よって, $\boldsymbol{e}'$ の候補数は $\binom{\ell}{(1+1/d)\tau\ell}$ 以下となり, 総当りに必要な回数が削減される.

■Ring- LPN 問題への応用: Bernstein と Lange [12] は Levieil と Fouque の高速化手法および Kirchner のアイデアを用いることにより, Ring-LPN 問題の解法が高速化できることを示している.

■その後の進展: Guo, Johansson, Löndahl [25] は, covering codes と呼ばれる符号を用いて Kirchner の手法の高速化を提案している. Kirchner の手法ではステップ 3 で, $A_{\boldsymbol{e}',\delta^{2^{t-1}},t-1}$ からのサンプル $\{(\boldsymbol{a}_i, b_i)\}$ が得られる. この $\boldsymbol{a}_i$ を covering code の受信語とみなすことで探索空間の圧縮を行い, 高速化を行っている. [*5]

Zhang, Jiao, Wang [48] は別の符号を用いて GJL アルゴリズムを改良している.

Bogos と Vaudenay [17] は GJL アルゴリズムの解析が一部欠けていることを分析し, 最適化を行いつつ詳細な計算量評価を与えた. Bogos らは同時に Gauss アルゴリズムと呼ばれるアルゴリズムについても解析を与えている.

Esser, Kübler, May [22] では BKW および Gauss アルゴリズムにより詳細な解析を行った.

Esser, Heuer, Kübler, May, Sohler [21] は BKW アルゴリズムに対して時間・メモリのトレードオフを提案している.

■サンプル数が少ない場合: これまでに挙げてきた BKW アルゴリズムおよびその改良では, サンプルが $O(2^{k/\log k})$ 個必要であった. Lyubashevsky [38] はサンプル数が $k^{1+\epsilon}$ 個と少ない場合であっても, BKW アルゴリズムを適用できるような指数個のサンプルの構成法を示している. Kirchner [29] も同様の構成法を示している. また, 上中谷と國廣 [31] は BKW アルゴリズムと Lyubashevsky の方法とを補間するようなアルゴリズムを提案している. Esser, Kubler, May [22] はサンプル数が少ない場合の BKW および Gauss アルゴリズムについて, より詳細な解析を行った.

3.1.5 Arora-Ge アルゴリズム

Arora と Ge [4] は多変数多項式問題で古くから用いられている再線形化と呼ばれる手法を用いて, LPN 問題を解くことを考えた. このアルゴリズムを $\mathsf{LPN}_{k,n,\tau}$ に用いた場合, $w = \tau n$ として, $\mathrm{poly}(k^w)$ 時間で解くことができる.

[*5] ただし, 国際会議でのプレゼンテーションではサンプル数が不足していたとの報告があり, 計算量・メモリ・サンプル数の評価は見直されている. 詳しくは, [48] および [17] を参照のこと

表 3.1　確率 1/2 以上で SD 問題を解く場合のパラメータ例 (Full Distance Decoding の場合)

	$\lg(\mathrm{T}ime)/n$	備考
Pra62 (Lee-Brickel)	0,121	[45], [34]
Stern89	0,117	[47]
MMT11	0,112	[40]
BJMM12	0,102	[10]
MO15	0,096 7	[41]
BM17	0,095 3	[14]; MO15 を最適化したもの
BM18	0,088 5	[15]

$\mathrm{poly}(k^w) = 2^{O(\tau n \log k)}$ であるから, $\tau = o(k/(n\log^2 k))$ のようにエラーがスパースであれば, BKW アルゴリズムよりも効率が良い.

3.1.6　SD 問題を経由するアルゴリズム

$\mathsf{LPN}_{k,n,\tau}$ に対応するシンドローム復号問題を考える. 重みを $w \approx \tau n$ とし, $\boldsymbol{H} \in \mathbb{F}_2^{(n-k)\times n}$ および $\boldsymbol{u} \in \mathbb{F}_2^{n-k}$ が与えられ, $\boldsymbol{e}\cdot\boldsymbol{H}^\top = \boldsymbol{u}$ となるような, 重みが w 以下の $\boldsymbol{e} \in \mathbb{F}_2^n$ を探索する問題である.

対応する線形符号の最小距離を d と置く. (2 進符号の場合, Gilvert-Varshamov 限界により, $k/n \approx 1 - H(d/n)$ である[*6].) $w \approx d$ の場合を Full Distance Decoding の場合と呼び, $w \approx d/2$ の場合を Half Distance Decoding と呼ぶ.

この問題を総当りで解く場合には, 重みが w の n 次元ベクトル $\boldsymbol{e}$ を列挙すればよい. そのため, 時間計算量は $O(\binom{n}{w})$ となる.

より効率的な手法として, Prange は “Information set decoding” と呼ばれる手法 [45] を提案した. 基本アイデアは以下である:

1. ランダムに $\boldsymbol{H}$ の列ベクトルを入れ替え, $\tilde{\boldsymbol{H}} = \boldsymbol{H}\cdot\boldsymbol{P}$ とする.
2. $\tilde{\boldsymbol{H}}$ を組織化し, $[\boldsymbol{I}_{n-k} \mid \boldsymbol{Z}] = \boldsymbol{S}\cdot\tilde{\boldsymbol{H}}$ とする.
3. $\boldsymbol{u}' = \boldsymbol{u}\boldsymbol{S}^\top$ を計算する.
4. $\boldsymbol{u}'$ の重みが w 以下であれば, この置換 $\boldsymbol{P}$ を採用し $\boldsymbol{e} = (\boldsymbol{u}', \boldsymbol{0}_k)\cdot\boldsymbol{P}^\top$ を出力する.

$\boldsymbol{u}'$ の重みが w 以下であるため, $\boldsymbol{e}$ の重みも w 以下である. また, $\boldsymbol{e}\cdot\boldsymbol{H}^\top = (\boldsymbol{u}', \boldsymbol{0}_k)\cdot\boldsymbol{P}^\top\boldsymbol{H}^\top = (\boldsymbol{u}', \boldsymbol{0}_k)\cdot\tilde{\boldsymbol{H}}^\top = (\boldsymbol{u}, \boldsymbol{0}_k)\boldsymbol{S}^\top\tilde{\boldsymbol{H}}^\top = (\boldsymbol{u}, \boldsymbol{0}_k)\cdot[\boldsymbol{I}_{n-k} \mid \boldsymbol{Z}]^\top = \boldsymbol{u}$ が成立する. よって, ステップ 4 のチェックを通るならば, $\boldsymbol{e}$ はシンドローム復号問題の解となっている. このような置換は全部で $\binom{n-k}{w}$ 通りあるため, 探索できる確率は $\binom{n-k}{w}/\binom{n}{w}$ となる. 期待計算量は $\mathrm{poly}(n,k)\cdot O(\binom{n}{w}/\binom{n-k}{w})$ となり, 先ほどの列挙法よりも速くなる.

Stern [47] 以降, 空間計算量を犠牲にすることで時間計算量を引き下げるアルゴリズムが多数提案されている. 以下では, Becker と May [15] により時間計算量の表を, 表 3.1 および 3.2 に示す. この表は, 時間計算量を最小化した場合の $R = k/n$ の最悪時 (1/2 の少し下) についてまとめられている. したがって, 問題のパラメータによっては, 表の数値よりも速く解くことが可能となる.

パラメータ設定によっては, $\mathsf{LPN}_{k,n,\tau}$ 問題を $\mathsf{SD}_{k,n,O(\tau n)}$ 問題に置き換えることで, これらの SD 問題用アルゴリズ

[*6] ここで $H(p) = -p\log(p) - (1-p)\log(1-p)$.

表 3.2　確率 1/2 以上で SD 問題を解く場合のパラメータ例 (Half Distance Decoding の場合)

	$\lg(\mathrm{T}ime)/n$	備考
Pra62 (Lee-Brickel)	0,057 6	[34]
Stern89	0,055 7	[47]
BLP	0,055 5	[13]
MMT11	0,053 7	[40]
BJMM12	0,049 4	[10]
MO15	0,047 3	[41]
BM18	0,046 5	[15]

ムも検討する必要がある.

3.1.7　量子アルゴリズムへの耐性

現在のところ多項式時間で LPN 問題を解く量子アルゴリズムは提案されていない. しかし量子アルゴリズムを利用した攻撃の高速化方法を Kachigar と Tillich [33] が提案している[*7]. Esser, Kübler, May [22] は, BKW や Gauss アルゴリズムの変種を量子アルゴリズムで高速化できる点を指摘している.

3.2　代表的な符号に基づく暗号方式の説明

本節では, 符号に基づく代表的な暗号方式と署名方式の説明を行う. 以下では, S_n で n 次対称群を表し, $\mathrm{GL}_k(\mathbb{F}_q)$ で k 次の $\mathbb{F}_q$ 要素正則行列全体がなす群を表す.

3.2.1　暗号方式 1: McEliece 暗号とその変種

McEliece [39] が提案した古典的な暗号方式である.

- k : 安全性パラメータ
- n : サンプルの個数
- τ : 誤差パラメータ (例: $\tau = O(k)$)
- t : 誤り訂正符号の誤り訂正能力 ($t = \Omega(\tau n)$)

鍵生成:　誤り訂正能力が t である $[n,k]_2$-線形符号の生成行列 $\boldsymbol{G}$ を生成する. $\boldsymbol{S} \leftarrow \mathrm{GL}_k(\mathbb{F}_2)$ をランダムに選ぶ. $\boldsymbol{P} \leftarrow S_n$ をランダムに選ぶ. $\tilde{\boldsymbol{G}} = \boldsymbol{SGP}$ とする.
公開鍵を $\tilde{\boldsymbol{G}}$ とし, 秘密鍵を $(\boldsymbol{S}, \boldsymbol{G}, \boldsymbol{P})$ とする.

暗号化:　平文を $\boldsymbol{m} \in \mathbb{F}_2^k$ とする. 乱数 $\boldsymbol{e} \leftarrow \mathsf{Ber}_\tau^n$ を選び, 暗号文 $\boldsymbol{c} = \boldsymbol{m}\tilde{\boldsymbol{G}} + \boldsymbol{e}$ を計算する.

復号:　$\hat{\boldsymbol{v}} = \boldsymbol{cP}^{-1}$ を計算する. $\hat{\boldsymbol{v}}$ を誤り訂正符号で訂正し復号すると $\boldsymbol{m}' = \boldsymbol{mS}$ を得る. $\boldsymbol{m} = \boldsymbol{m}'\boldsymbol{S}^{-1}$ を出力する.

[*7] Kirshanova [30] が Kachigar と Tillich の結果 [33] の改良を提案していたが, 誤りがあったことが報告されている. そのため, 2018 年時点でのベストな量子アルゴリズムは Kachigar と Tillich [33] であると考えられる.

復号の正当性は以下で確認される. $\boldsymbol{c} = \boldsymbol{m}\tilde{\boldsymbol{G}} + \boldsymbol{e}$ として, $\hat{\boldsymbol{v}} = \boldsymbol{c}\boldsymbol{P}^{-1}$ を計算すると,

$$\hat{\boldsymbol{v}} = \boldsymbol{m}\tilde{\boldsymbol{G}}\boldsymbol{P}^{-1} + \boldsymbol{e}\boldsymbol{P}^{-1} = \boldsymbol{m}\boldsymbol{S}\boldsymbol{G} + \boldsymbol{e}\boldsymbol{P}^{-1}$$

を得る. $\boldsymbol{m}\boldsymbol{S}\boldsymbol{G}$ は符号語であり, $\boldsymbol{e}\boldsymbol{P}^{-1}$ は誤りである. $\boldsymbol{e}\boldsymbol{P}^{-1}$ の重みが t 以下であれば, 誤り訂正符号の復号により, $\boldsymbol{m}' = \boldsymbol{m}\boldsymbol{S}$ を得る. よって, 高い確率で復号に成功する.

平文 $\boldsymbol{m}$ および $\tilde{\boldsymbol{G}}$ がランダムであれば, 暗号文 $\boldsymbol{c}$ は LPN 仮定の下で疑似ランダムである. $\tilde{\boldsymbol{G}}$ が疑似ランダムであることを言うためには, McEliece 仮定と呼ばれる仮定が必要となる.

定義 3.8 (McEliece 仮定) $[n,k]_{q(n)}$ -符号のクラス $\mathcal{C}$ を固定する. 敵 $\mathcal{A}$ の優位性を

$$\mathsf{Adv}_{\mathcal{A}}(n) = \left| \Pr[\boldsymbol{S} \leftarrow \mathrm{GL}_k(\mathbb{F}_q), \boldsymbol{G} \leftarrow \mathcal{C}, \boldsymbol{P} \leftarrow S_n : \mathcal{A}(1^n, \tilde{\boldsymbol{G}} = \boldsymbol{S}\boldsymbol{G}\boldsymbol{P}) = 1] - \Pr[\tilde{\boldsymbol{G}} \leftarrow \mathbb{F}_q^{k\times n} : \mathcal{A}(1^n, \tilde{\boldsymbol{G}}) = 1] \right|$$

で定義する. 任意の多項式時間の敵 $\mathcal{A}$ について, その優位性が無視できる関数であるとき, McEliece 仮定が成立するという.

左側の敵は McEliece 暗号の公開鍵 (または Niederreiter 暗号の公開鍵の双対) を受け取っている. そのため, この仮定は, McEliece 暗号の公開鍵はランダムな同サイズの行列と見分けが付かないということを意味する.

3.2.2 暗号方式 2: Niederreiter 暗号とその変種

Niederreiter [42] が 1986 年に提案した. のちに McEliece 暗号と「等価」であることが示された. 詳しくは [35] を参照のこと.

- k : 安全性パラメータ
- n : サンプルの個数
- τ : 誤差パラメータ (例: $\tau = ck$)
- t : 誤り訂正符号の誤り訂正能力 ($t = \Omega(\tau n)$)

鍵生成: 誤り訂正能力が t である $[n,k]$ -線形符号のパリティ検査行列 $\boldsymbol{H} \in \mathbb{F}^{(n-k)\times n}$ を生成する. $\boldsymbol{T} \leftarrow \mathrm{GL}_{n-k}(\mathbb{F})$ をランダムに選ぶ. $\boldsymbol{Q} \leftarrow S_n$ をランダムに選ぶ. $\tilde{\boldsymbol{H}} = \boldsymbol{T}\boldsymbol{H}\boldsymbol{Q}$ とする.
公開鍵を $\tilde{\boldsymbol{H}}$ とし, 秘密鍵を $(\boldsymbol{T}, \boldsymbol{H}, \boldsymbol{Q})$ とする.

暗号化: 平文を $\boldsymbol{e} \in S(n,t)$ とする. 暗号文 $\boldsymbol{d} = \boldsymbol{e}\cdot\tilde{\boldsymbol{H}}^{\top} \in \mathbb{F}^{n-k}$ を計算する.

復号: $\hat{\boldsymbol{w}} = \boldsymbol{d}\cdot\boldsymbol{T}^{-\top}$ を計算する. $\hat{\boldsymbol{w}}$ を誤り訂正符号で訂正し復号し, 誤りとして $\boldsymbol{e}' = \boldsymbol{e}\boldsymbol{P}^{\top}$ を得る. $\boldsymbol{e} = \boldsymbol{e}'\boldsymbol{P}^{-\top}$ を出力する.

復号の正当性は以下で確認される. $\boldsymbol{d} = \boldsymbol{e}\cdot\tilde{\boldsymbol{H}}^{\top}$ として, $\hat{\boldsymbol{w}} = \boldsymbol{d}\boldsymbol{T}^{-\top}$ を計算すると,

$$\hat{\boldsymbol{v}} = \boldsymbol{e}\cdot\tilde{\boldsymbol{H}}^{\top}\boldsymbol{T}^{-\top} = \boldsymbol{e}\cdot\boldsymbol{Q}^{\top}\boldsymbol{H}^{\top}\boldsymbol{T}^{\top}\boldsymbol{T}^{-\top} = \boldsymbol{e}\boldsymbol{Q}^{\top}\cdot\boldsymbol{H}^{\top}$$

を得る. $\boldsymbol{e}\boldsymbol{Q}^{\top}$ の重みが t 以下であれば, 誤り訂正符号の復号により, $\boldsymbol{e}' = \boldsymbol{e}\boldsymbol{Q}^{\top}$ を得る. よって, 高い確率で復号に成功する.

平文 $\boldsymbol{e}$ および $\tilde{\boldsymbol{H}}$ がランダムであれば, 暗号文 $\boldsymbol{d}$ は LPN 仮定の下で疑似ランダムである. $\tilde{\boldsymbol{H}}$ が疑似ランダムであることを言うためには, McEliece 暗号と同様に McEliece 仮定を考えればよい.

3.2.3 暗号方式 3: Alekhnovich 暗号

Alekhnovich は [1] で公開鍵暗号方式を 2 つ提案している. ここではシンプルな 1 つ目の暗号方式を取り上げる. パラメータを以下とする.

- n : 安全性パラメータ
- m : LPN サンプルの個数 (例: $m = 2n + 1$)
- $\tau > 0$: 誤差パラメータ (例: $\tau = n^{-1/2-\epsilon}$)

このとき Alekhnovich 暗号は以下で構成される:

秘密鍵の生成: ランダムに $\boldsymbol{e} \leftarrow \mathsf{Ber}_\tau^m$ を選ぶ.

公開鍵の生成: ランダムに $\boldsymbol{A} \leftarrow \mathbb{F}_2^{n\times m}$ を選ぶ. ランダムに $\boldsymbol{s} \leftarrow \mathbb{F}_2^n$ を選ぶ. $\boldsymbol{b} = \boldsymbol{s}\boldsymbol{A} + \boldsymbol{e} \in \mathbb{F}_2^m$ を計算し, $\boldsymbol{B} = \binom{\boldsymbol{A}}{\boldsymbol{b}}$ とする. $\boldsymbol{M} \in \mathbb{F}_2^{(m-n-1)\times m}$ を $\ker(\boldsymbol{B}^\top)$ の基底とし, 公開鍵を $\boldsymbol{M}$ とする.

暗号化: 平文が 0 の場合, $\boldsymbol{t} \leftarrow \mathbb{F}_2^{m-n-1}$ と $\boldsymbol{f} \leftarrow \mathsf{Ber}_\tau^m$ をランダムに選び, $\boldsymbol{c} = \boldsymbol{t}\boldsymbol{M} + \boldsymbol{f} \in \mathbb{F}_2^m$ を出力する.
平文が 1 の場合, ランダムに $\boldsymbol{c} \leftarrow \mathbb{F}_2^m$ を選び出力する.

復号: 暗号文 $\boldsymbol{c} \in \mathbb{F}_2^m$ について, $\delta = \langle \boldsymbol{c}, \boldsymbol{e}\rangle$ を計算する. δ を出力する.

復号の正当性について以下考察する. 平文が 1 の場合, 復号は確率 $1/2$ で成功する.
一方, 平文が 0 の場合, $\boldsymbol{e} \in \mathrm{Span}(\boldsymbol{B})$ および $\boldsymbol{t}\boldsymbol{M} \in \ker(\boldsymbol{B}^\top)$ より $\boldsymbol{t}\boldsymbol{M}\boldsymbol{e}^\top = 0$ であることに注意すると,

$$\langle \boldsymbol{c}, \boldsymbol{e}\rangle = \boldsymbol{t}\boldsymbol{M} \cdot \boldsymbol{e}^\top + \boldsymbol{f}\boldsymbol{e}^\top = \langle \boldsymbol{f}, \boldsymbol{e}\rangle$$

なので, $\langle \boldsymbol{f}, \boldsymbol{e}\rangle = 0$ であれば復号に成功する. 誤り確率を評価すると, $\tau = n^{-1/2-\epsilon}$ より, $\Pr[\langle \boldsymbol{f}, \boldsymbol{e}\rangle = 1] \approx (1-\tau)^{\tau m} = o(1)$ となり, $1 - o(1)$ の確率で復号に成功する. また, 上記の暗号方式の安全性については, 判定版 LPN 仮定の下で CPA 安全であることが証明される.

ここで紹介した暗号方式は, 1 ビット暗号であり, 復号誤りの確率も高いため実用的ではない. 同様な仮定・パラメータの下で効率的な暗号方式として, 2 つ目の Alekhnovich 暗号や 複数ビットを暗号化できる Lyubashevsky-Peikert-Regev 風の暗号を参考にされたい.

3.2.4 暗号方式 3: Lyubashevsky-Peikert-Regev 風暗号

鍵共有方式として考えられているが, 本稿では暗号方式として書く. McEliece 暗号では公開鍵の疑似ランダム性そのものを McEliece 仮定として導入していた. 一方, Lyubashevsky-Peikert-Regev (LPR) 風暗号では公開鍵の疑似ランダム性を判定版 LPN 仮定から示すことができる.

- k : 安全性パラメータ
- $n = n_1 + n_2$: サンプルの個数
- τ : 誤差パラメータ (例: $\tau = ck$)
- t : 誤り訂正符号の誤り訂正能力 ($t = \Omega(\tau n)$)

鍵生成: 誤り訂正能力が t である $[n_2, \ell]$ -線形符号の生成行列 $\boldsymbol{G}_c$ を生成する. $\boldsymbol{A} \leftarrow \mathbb{F}^{k\times n_1}$ とする. $\boldsymbol{X} \leftarrow \mathsf{Ber}_\tau^{n_1\times n_2}$, $\boldsymbol{Y} \leftarrow \mathsf{Ber}_\tau^{k\times n_2}$ とし, $\boldsymbol{B} = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{Y} \in \mathbb{F}^{k\times n_2}$ とする.
公開鍵を $\tilde{\boldsymbol{G}} = [\boldsymbol{A} \mid \boldsymbol{B}] \in \mathbb{F}^{k\times n}$ とし, 秘密鍵を $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{X})$ とする.

暗号化: 平文を $\boldsymbol{m} \in \mathbb{F}_2^\ell$ とする. 乱数 $\boldsymbol{s} \leftarrow \mathsf{Ber}_\tau^k$ と 乱数 $\boldsymbol{e} \leftarrow \mathsf{Ber}_\tau^n$ を選び, 暗号文 $\boldsymbol{c} = \boldsymbol{s}\tilde{\boldsymbol{G}} + \boldsymbol{e} + (\boldsymbol{0}_{n_1}, \boldsymbol{m}\boldsymbol{G}_c) \in \mathbb{F}^n$ を計算する.

復号: $\boldsymbol{d} = \boldsymbol{c} \cdot \binom{-\boldsymbol{X}}{\boldsymbol{I}_{n_2}}$ を計算する. $\boldsymbol{d}$ を誤り訂正符号で訂正し復号すると $\boldsymbol{m}$ を得る.

復号の正当性は以下で確認される. $\boldsymbol{c} = \boldsymbol{s}\tilde{\boldsymbol{G}} + \boldsymbol{e} + (\boldsymbol{0}_{n_1}, \boldsymbol{m}\boldsymbol{G}_c)$ なので, 前半部を $\boldsymbol{u} = \boldsymbol{s}\boldsymbol{A} + \boldsymbol{e}_1$, 後半部を $\boldsymbol{v} = \boldsymbol{s}\boldsymbol{B} + \boldsymbol{e}_2 + \boldsymbol{m}\boldsymbol{G}_c$ と書く.

$\boldsymbol{d} = \boldsymbol{c} \cdot \binom{-\boldsymbol{X}}{\boldsymbol{I}_{n_2}}$ を計算すると,

$$\boldsymbol{d} = \boldsymbol{v} - \boldsymbol{u}\boldsymbol{X} = \boldsymbol{m}\boldsymbol{G}_c + \boldsymbol{s}\boldsymbol{B} + \boldsymbol{e}_2 - \boldsymbol{s}\boldsymbol{A}\boldsymbol{X} - \boldsymbol{e}_1\boldsymbol{X} = \boldsymbol{m}\boldsymbol{G}_c + (\boldsymbol{e}_2 - \boldsymbol{e}_1\boldsymbol{X} + \boldsymbol{s}\boldsymbol{Y})$$

を得る. $\boldsymbol{m}\boldsymbol{G}_c$ は符号語であり, $\boldsymbol{e}_2 - \boldsymbol{e}_1\boldsymbol{X} + \boldsymbol{s}\boldsymbol{Y}$ は誤りベクトルである. よって, $\boldsymbol{e}_2 - \boldsymbol{e}_1\boldsymbol{X} + \boldsymbol{s}\boldsymbol{Y}$ の重みが t 以下であれば, 誤り訂正符号の復号により, $\boldsymbol{m}$ を得る. よって, 高い確率で復号に成功する.

$\tilde{\boldsymbol{G}}$ がランダムであれば, 暗号文 $\boldsymbol{c}$ は LPN 仮定の下で疑似ランダムである. $\tilde{\boldsymbol{G}}$ が疑似ランダムであることを言うためには, $\boldsymbol{B} = \boldsymbol{A}\boldsymbol{X} + \boldsymbol{Y}$ が疑似ランダムであればよい. これは, パラメータを変更した LPN 仮定の下, 成立する.

3.2.5 署名方式 1: CFS 署名とその変種

Courtois, Finiasz, Sendrier が 2001 年に提案した [18]. のちに, 安全性証明に用いられた仮定が提案パラメータセットでは成り立たないことが示された [23, 24]. しかし後の方式に大きな影響を与えたため, ここに記す. Niederreiter 暗号を思い出すと, 秘密鍵を持っている場合, w 以下の重みの誤りは訂正できる. しかし 訂正可能なシンドロームの集合 $\{\boldsymbol{e}\tilde{\boldsymbol{H}} \in \mathbb{F}^{n-k} \mid \boldsymbol{e} \in \mathcal{S}(n, w)\}$ のサイズは $\mathbb{F}^{n-k}$ のサイズに比べれば圧倒的に少ない. そのため, Full-Domain Hash 法を使おうとして, 文書のハッシュ値を $\boldsymbol{u} \in \mathbb{F}^{n-k}$ に写した場合, 正しく復号できないハッシュ値になることが多い. そこで CFS 署名では, ハッシュ値を $\boldsymbol{u} = \mathsf{Hash}(M, i)$ と i をインクリメントしながら計算する. ハッシュ値が $\{\boldsymbol{e}\tilde{\boldsymbol{H}} \in \mathbb{F}^{n-k} \mid \boldsymbol{e} \in \mathcal{S}(n, w)\}$ に入るものを採用する.

署名鍵と検証鍵: パリティ検査行列 $\tilde{\boldsymbol{H}} \in \mathbb{F}^{(n-k)\times n}$ を検証鍵とする. また秘密鍵を用いると, 重み w 以下の符号語を訂正できることとする.

署名: 文書 M について,

1. $i = 0$ とする
2. $\boldsymbol{u} = \mathsf{Hash}(M, i)$ を計算する
3. 重み w 以下の $\boldsymbol{e}$ で, $\boldsymbol{e} \cdot \tilde{\boldsymbol{H}}^\top = \boldsymbol{u}$ となるものを計算する. なければ $i \leftarrow i + 1$ としてステップ 2 に戻る
4. $\sigma = (\boldsymbol{e}, i)$ を出力する.

検証: 文書 M と $\sigma = (\boldsymbol{e}, i)$ について, $\mathsf{HW}(\boldsymbol{e}) \le w$ と $\boldsymbol{e} \cdot \tilde{\boldsymbol{H}}^\top = \mathsf{Hash}(M, i)$ ならば, 受理する. そうでないならば, 不受理とする.

安全性の根拠として, 以下の 2 つの仮定を必要とする.

- McEliece 仮定: トラップドアが入っている $\tilde{\boldsymbol{H}}$ はランダム符号のパリティ検査行列と区別が付かない
- 探索版 SD 仮定: 探索版 SD 問題が困難

3.3 具体的な暗号方式

本稿では以下の暗号方式を取り上げる.

1. Classic McEliece: ハミング距離を採用, Niederreiter 暗号を採用, 符号の構成が非常に保守的, という観点からこれを取り上げる.
2. DAGS: ハミング距離を採用, McEliece 暗号を採用, Quasi-Dyadic 符号を用いて鍵を圧縮している, という観点からこれを取り上げる.
3. RQC: ランク距離を採用, LPR 風暗号を採用, Quasi-Cyclic 符号を用いて鍵を圧縮している, という特徴からこれを取り上げる.
4. RankSing: ランク距離を採用, CFS 署名を採用という特徴から破れはしているがこれを取り上げる.

表 3.3　本節で扱う主要な符号に基づく暗号技術

文献	暗号化	鍵交換	署名
Classic McEliece [8]	○	○	
DAGS [6]	○	○	
RQC [2]	○	○	
RankSign [5]			○

3.3.1　暗号方式 1: Classic McEliece

- 提案者: Bernstein, Chou, Lange, von Maurich, Misoczki, Niederhagen, Persichetti, Peters, Schwabe, Sendrier, Szefer, Wang.
- 基本方式の説明: Niederreiter 暗号方式に基づいている. 基本符号方式として 2 進 Goppa 符号を利用している.

  鍵生成:　t 誤りを訂正できる 2 進 Goppa 符号のパリティ検査行列 $\boldsymbol{H}$ をランダム生成する. 組織符号化し, $\tilde{\boldsymbol{H}} = [\boldsymbol{I}_{n-k} \mid \boldsymbol{T}]$ とする. 公開鍵を $\boldsymbol{T} \in \mathbb{F}_2^{(n-k)\times k}$ とする. 復号鍵を符号生成に使ったパラメータとする.

  暗号化 $E(pk, \boldsymbol{e})$:　入力を, $pk = \boldsymbol{T} \in \mathbb{F}_2^{(n-k)\times k}$ と $\boldsymbol{e} \in S_H(n,t)$ とする. $\tilde{\boldsymbol{H}} = [\boldsymbol{I}_{n-k} \mid \boldsymbol{T}]$ とし, 暗号文として $\boldsymbol{c} = \tilde{\boldsymbol{H}} \cdot \boldsymbol{e} \in \mathbb{F}_2^{n-k}$ を出力する.

  復号 $D(sk, \boldsymbol{c})$:　重み t のベクトル $\boldsymbol{e}$ を復号する.
  1. $\boldsymbol{c}$ に k 個ゼロを加え, $\boldsymbol{v} = (\boldsymbol{c}, \boldsymbol{0}_k) \in \mathbb{F}^n$ を考える
  2. Goppa 符号の復号アルゴリズムを用いて, $\boldsymbol{v}$ と距離 t 以下にある符号語 $\boldsymbol{d}$ を計算する. (なければ $\perp$ を出力する)
  3. $\boldsymbol{e} = \boldsymbol{v} + \boldsymbol{d}$ とする.
  4. $\mathsf{HW}(\boldsymbol{e}) = t$ かつ $\boldsymbol{c} = \tilde{\boldsymbol{H}}\boldsymbol{e}$ ならば $\boldsymbol{e}$ を出力する. (そうでなければ $\perp$ を出力する.)
- 鍵カプセル化方式の説明: 基本方式を決定性の公開鍵暗号とみなし, $\mathsf{HU}_m^{\not\perp}$ 変換をかけたものとみなせる. 以下ではハッシュ関数 $\mathsf{H}\colon \{0,1\}^* \to \{0,1\}^{256}$ を用いる.

  鍵生成:　公開鍵は同じ. 復号鍵に, n ビットのランダム文字列 s を加える.

  鍵カプセル化:　1.　$e \leftarrow S(n,t)$
  2. $C_0 = E(pk, e)$
  3. $C_1 = \mathsf{H}(2, e)$ とし, $C = (C_0, C_1)$ とする.
  4. $K = \mathsf{H}(1, e, C)$ とする.
  5. 暗号文は C, セッション鍵は K.

  デカプセル:　1.　$C = (C_0, C_1) \in \mathbb{F}_2^{n-k} \times \mathbb{F}_2^{\ell}$ とパースする

表 3.4　Classic McEliece のパラメータ. 単位は全て bit とする.

パラメータ名	公開鍵長	秘密鍵長	暗号文長
kem/mceliece6960119	8 373 911	8 484 818	1 803
kem/mceliece8192128	10 862 592	10 974 848	1 920

2. $b = 1$ とする
3. $e \leftarrow D(sk, C_0)$ とする. $e = \perp$ であれば, $b = 0$, $e = s$ と上書きする
4. $C_1' = \mathsf{H}(2, e)$ とする.
5. $C_1' \neq C_1$ ならば, $b = 0$, $e = s$ と上書きする.
6. $K = \mathsf{H}(b, e, C)$ を計算する.
7. K を出力する.

表 3.4 に鍵カプセル化方式の鍵長および暗号文長をまとめた. 2 つのパラメータセット (kem/mceliece6960119, kem/mceliece8192128) が, どちらも Category 5 相当として提案されている.

3.3.2　暗号方式 2: DAGS

- 提案者: Banegas, Barreto, Boidje, Cayrel, Dione, Gaj, Gueye, Haeussler, Klamti, N'diaye, Nguyen, Persichetti, Ricardini.
- 基本方式の説明: $k = k' + k''$ とする.

  鍵生成:　McEliece 暗号と同様に $pk = \tilde{\boldsymbol{G}} \in \mathbb{F}^{k \times n}$ とする. ただし, $\tilde{\boldsymbol{G}}$ は Quasi-Dyadic となるように基本符号を選んでいる.

  暗号化 $E(pk, \boldsymbol{m}; \boldsymbol{\rho}, \boldsymbol{e})$:　$\boldsymbol{m} \in \mathbb{F}_q^{k'}$ を平文とし, $\boldsymbol{\rho} \in F_q^{k''}$ と $\boldsymbol{e} \in S_q(n, t)$ を乱数とする. $\boldsymbol{c} = (\boldsymbol{\rho}, \boldsymbol{m}) \cdot \tilde{\boldsymbol{G}} + \boldsymbol{e}$ を出力する.

  鍵生成 $D(sk, \boldsymbol{c})$:　McEliece 暗号の復号を行い, $(\boldsymbol{\rho}, \boldsymbol{m})$ を得る. $\boldsymbol{m}$ を出力する.
- 鍵カプセル化方式の説明: 基本方式を乱択公開鍵暗号方式とみなし, $\mathsf{QFO}_m^{\perp}$ 変換を適用したものとみなせる. 以下ではハッシュ関数 $\mathsf{H}, \mathsf{H}' \colon \{0,1\}^* \to \{0,1\}^{256}$ を用いる.

  鍵生成:　同上.

  鍵カプセル化:　1.　$\boldsymbol{m} \leftarrow \mathbb{F}_q^{k'}$ とする.
  2. $\boldsymbol{r} = (\boldsymbol{\rho}, \boldsymbol{\sigma}) = \mathsf{H}_G(\boldsymbol{m}) \in \mathbb{F}_q^{k'} \times \mathbb{F}_q^{k''}$ を計算する.
  3. $\boldsymbol{e} \in S_q(n, w)$ を $\boldsymbol{\sigma}$ から計算する
  4. $\boldsymbol{c} = E(pk, \boldsymbol{m}; \boldsymbol{\rho}, \boldsymbol{e})$ とする.
  5. $\boldsymbol{d} = \mathsf{H}'(\boldsymbol{m})$ を計算する.
  6. $K = \mathsf{H}(\boldsymbol{m})$ を計算する.
  7. 暗号文を $C = (\boldsymbol{c}, \boldsymbol{d})$ とし, 鍵を K とする.

  デカプセル:　1.　sk を用いて $\boldsymbol{c}$ を復号して, $\boldsymbol{\mu}' = (\boldsymbol{\rho}', \boldsymbol{m}')$ および $\boldsymbol{e}'$ を得る.
  2. 復号に失敗したら, $\perp$ を出力して停止
  3. $HW(\boldsymbol{e}') \neq w$ なら, $\perp$ を出力して停止
  4. $\boldsymbol{r}' = (\boldsymbol{\rho}'', \boldsymbol{\sigma}') = \mathsf{H}_G(\boldsymbol{m}') \in \mathbb{F}_q^{k'} \times \mathbb{F}_q^{k''}$ を計算する.

表 3.5　DGAS のパラメータ. 単位は全て bit とする.

パラメータ名	公開鍵長	秘密鍵長	暗号文長
DAG_1	54 080	1 730 560	4 416
DAG_3	67 584	5 136 384	7 552
DAG_5	92 928	17 842 176	12 928

5. $\boldsymbol{e}'' \in S_q(n, w)$ を $\boldsymbol{\sigma}'$ から計算する
6. $\boldsymbol{d} = \mathsf{H}'(\boldsymbol{m})$ を計算する.
7. $\boldsymbol{e} \neq \boldsymbol{e}''$ or $\boldsymbol{\rho}' \neq \boldsymbol{\rho}''$ or $\boldsymbol{d} \neq \boldsymbol{d}'$ ならば, $\perp$ を出力して停止
8. $K = \mathsf{H}(\boldsymbol{m}')$ を計算し, 出力する.

表 3.5 に鍵カプセル化方式の鍵長および暗号文長をまとめた. 3 つのパラメータセットがそれぞれ Category 1, 3, 5 相当として提案された.

3.3.2.1　攻撃について

Barelli と Couvreur により攻撃が提案されている [7]. 彼らの評価によれば, 公開鍵から秘密鍵を求めるには, $\mathbb{F}_q$ 上の操作を $O(n^{3+\frac{2q}{\#\mathcal{G}}})$ 回必要とのことである. ここで, $\mathcal{G}$ は符号の置換群である. [*8] 具体的なパラメータに当てはめて $\mathbb{F}_q$ 上の演算回数を見積もると, DAG_1 で 2^{70}, DAG_3 で 2^{80}, DAG_5 で 2^{58} となる. Grobner 基底を求めるアルゴリズムを用いて実験的に解いた場合は DAG_1 19 分, DAG_5 で 1 分未満であった.

この攻撃を受け, v2 というパラメータセットが提案されている.

3.3.3　暗号方式 3: RQC

- 提案者: Aguilar Melchor, Blazy, Aragon, Deneuville, Bettaieb, Gaborit, Bidoux, Zemor.
- 基本方式の説明: 符号版の LPR 暗号方式に基づき, 公開鍵暗号を構成している. 基本となる符号にランク距離用の符号を採用している. 以下では $n' = n_1 = n_2$ とする. $\mathcal{R} = \mathbb{F}_{q^m}[X]/(X^{n'} - 1)$ とする. ランク重み w までの誤りを訂正できる $[n', \ell]$ -線形符号を採用し, その符号化・復号アルゴリズムを $\mathsf{encode}, \mathsf{decode}$ とする.

  鍵生成:　$a \leftarrow \mathcal{R}$, $x, y \leftarrow S_R(n', w)$ とし, $b = ax + y$ を計算する. $pk = (a, b) \in \mathcal{R}^2$ とし, $sk = (x, y)$ とする.

  暗号化 $E(pk, m, s, e_1, e_2)$:　$c = (u, v) = (sa + e_1, sb + e_2 + \mathsf{encode}(m))$ を出力する. ($e \leftarrow S_R(n', w_e)$, $e_1, e_2 \leftarrow S_R(n', w_r)$ としている)

  復号 $D(sk, c)$:　$c = (u, v)$ に対して, $\mathsf{decode}(v - ux)$ を出力する.
- 鍵カプセル化方式: 基本方式を乱択な公開鍵暗号とみなし, $\mathsf{QFO}^{\perp}$ 変換を適用したものとみなせる. 以下ではハッシュ関数 H, H': $\{0,1\}^* \to \{0,1\}^{256}$ を用いる. また, XOF [*9] として H_G: $\{0,1\}^{\to}\{0,1\}^*$ も用いる.

  鍵生成:　同上

  鍵カプセル化:　1.　$m \leftarrow \mathbb{F}_{q^m}^{\ell}$ をランダムにとる.

  2.　$\theta \leftarrow \mathsf{H}_G(m)$ を計算する. θ から s, e_1, e_2 を生成する.

  3.　$c = (u, v) = E(pk, m, s, e_1, e_2)$ を計算する. $d = \mathsf{H}'(m)$ とする. $K = \mathsf{H}(m, c)$ とする.

[*8] 符号 $\mathcal{C} \subset \mathbb{F}_{q^m}^n$ について, $Perm(\mathcal{C}) = \{\sigma \in S_n \mid \sigma(\mathcal{C}) = \mathcal{C}\}$ のことを符号 $\mathcal{C}$ の置換群と呼ぶ.

[*9] eXtendable-Output Functions の略. SHAKE128 や SHAKE256 が例として知られている.

表 3.6　RQC のパラメータ. 単位は全て bit とする.

パラメータ名	公開鍵長	秘密鍵長	暗号文長
RQC-I	11 926	11 926	12 438
RQC-II	21 922	21 922	22 434
RQC-III	28 078	28 078	28 590

4. 暗号文 $C = (c, d)$, セッション鍵 K を出力する.

デカプセル: 1. $m' \leftarrow D(sk, c)$ を計算する.
2. $\theta' = \mathsf{H}_G(m')$ を計算する. θ' から s', e'_1, e'_2 を生成する.
3. $c \neq E(pk, m', s', e'_1, e'_2)$ or $d \neq d'$ ならば $\perp$ を出力して停止.
4. $K = \mathsf{H}(m, c)$ を出力する.

表 3.6 に鍵カプセル化方式の鍵長および暗号文長をまとめた. 3 つのパラメータセットがそれぞれ Category 1, 3, 5 相当として提案された.

なお, x, y を作る際のシードだけ覚えておくことにすれば, 秘密鍵長を縮めることが可能である（例えば $|sk| = 512$ bits）. 同様に公開鍵生成の a を疑似乱数で生成することにすれば, $|pk'| = |pk|/2 + 512$ bits と約半分になる.

3.3.4　署名方式 1: RankSign

- 提案者: Nicolas Aragon, Olivier Ruatta, Philippe Gaborit, Gilles Zemor, Adrien Hauteville.
- 基本方式の説明: ランク距離ベースの CFS 署名である. augmented Low-Rank Parity-Check 行列から符号を構成している点が新しい.

  鍵生成: $\boldsymbol{H} \in \mathbb{F}_{q^m}^{(n-k)\times n}$ を LRPC 符号のパリティ検査行列とする. $\boldsymbol{R} \leftarrow \mathbb{F}_{q^m}^{(n-k)\times t}$, $\boldsymbol{Q} \leftarrow \mathrm{GL}_{n+t}(\mathbb{F}_q)$ をランダムの生成する.

  さらに, $[\boldsymbol{R} \mid \boldsymbol{H}] \cdot \boldsymbol{Q}$ を パリティ検査行列とする $[n+t, k+t]_{q^m}$ 符号 $\mathcal{C}$ を考える. この符号 $\mathcal{C}$ の組織パリティ検査行列を検証鍵 $pk = \tilde{\boldsymbol{H}} = [I_{n-k} \mid \boldsymbol{R}'] \in \mathbb{F}_{q^m}^{(n-k)\times(n+t)}$ とする. （すなわち, ある行列 $\boldsymbol{P}$ が存在して, $\tilde{\boldsymbol{H}} = \boldsymbol{P} \cdot [\boldsymbol{R} \mid \boldsymbol{H}]\boldsymbol{Q}$ とできる）$sk = (\boldsymbol{P}, [\boldsymbol{R} \mid \boldsymbol{H}], \boldsymbol{Q})$ を秘密鍵とする.

  署名: M を平文として,
  1. $\mathsf{seed} \leftarrow \{0,1\}^\ell$
  2. $\boldsymbol{s} = \mathsf{H}_G(M, \mathsf{seed})$
  3. $\boldsymbol{e}' \leftarrow S_{q^m}(n+t, t')$
  4. $\boldsymbol{s}' = \boldsymbol{s} - \boldsymbol{e}' \cdot \tilde{\boldsymbol{H}}^\top$
  5. $(e_1, e_2, \ldots, e_t) \in \mathbb{F}_{q^m}^t$ をランダムに選び, 線形空間 $T = \langle e_1, e_2, \ldots, e_t \rangle$ とする.
  6. $\boldsymbol{s}'' = \boldsymbol{s}'\boldsymbol{P}^{-\top} - (e_1, e_2, \ldots, e_t) \cdot \boldsymbol{R}^\top$ を計算する.
  7. 行列 $\boldsymbol{H}$, 部分空間 T およびシンドローム $\boldsymbol{s}''$ に対して符号の復号を行い, 重み $w' = w - t'$ 以下の 誤りベクトル $(e_{t+1}, e_{t+2}, \ldots, e_{n+t})$ を計算する. $\boldsymbol{s}''$ が T -復号不可能な場合, Step1 に戻る.
  8. $\boldsymbol{e} = \boldsymbol{e}' + (e_1, e_2, \ldots, e_{n+t}) \cdot \boldsymbol{Q}^{-\top}$ とする.
  9. 署名として, $\sigma = (\mathsf{seed}, \boldsymbol{e})$ を出力する.

  検証: $\boldsymbol{e} \cdot \tilde{\boldsymbol{H}} = \mathsf{H}_G(M, \mathsf{seed})$ かつ $w(e) \leq r$ ならば, 署名を受理する. そうでないならば, 不受理とする.

表 3.7　RankSign のパラメータ．単位は全て bit とする．

パラメータ名	検証鍵長	署名長
RankSign I	80 640	11 008
RankSign II	96 768	12 000
RankSign III	155 520	17 280
RankSign IV	228 480	23 424

表 3.7 に鍵長と署名長をまとめた．

3.3.4.1　攻撃について

Debris-Alazard と Tillich により, すべてのパラメータ例が破られている [20]．署名生成の効率性を重視した結果, 非常に低いランク重みの符号語が構成した符号に多く含まれていた．彼らはこのような符号語を探索する代数的アルゴリズムを提案し, またその符号語を元にして秘密鍵相当の情報を計算できることを示した．低ランク重み符号語を探索するために一部で多変数連立二次多項式を解く必要がある．RankSign IV のパラメータ設定であっても, 実験的には 260 秒程度で符号語を 1 つ探索できると報告されている．（また色々な方法で変数除去を行うと, すべてのパラメータ設定で方程式の数が変数の数より多いことが示されている (Prop.4) ）したがって, パラメータの取り方だけでなく構成方法自体を変える必要がある．

3.4　まとめ

符号に基づく暗号技術は McEliece により 40 年以上前に提案されており, パラメータは改訂されているものの, いまだに破られていない暗号方式である．Classic McEliece などのように, 公開鍵や秘密鍵は長いものの, 暗号文は短い方式が多い．LPN 問題は学習理論や符号理論から派生した問題であり, 誤り確率 τ が十分大きい場合の LPN 問題を多項式時間で効率的に解くことは困難であると予想されている．

共通鍵や公開鍵の分野で多くの方式が LPN 問題に基づいて提案されている．LWE 問題と比較した場合, 利点としては,

- $\mathbb{F}_2$ およびその拡大体を基に構成するため, ハードウェア構成との相性が良い点
- 誤差分布としてベルヌーイ分布やその一般化した分布を用いるため, 誤差のサンプリングが容易である点

が挙げられる．一方, 欠点として,

- 鍵や暗号文のサイズが大きくなりやすい点
- ID ベース暗号や完全準同型暗号といった発展的な応用が少ない点

が挙げられる．

暗号方式のパラメータ設定の際には, 3.1 節で挙げたさまざまなアルゴリズムを考慮する必要がある．アルゴリズムの高速化について盛んに研究されており, 動向を注視する必要がある．また, 攻撃に用いられるアルゴリズムの研究は理論的なものが多く, 攻撃実験報告は小さいパラメータに対して行ったものが多い．そのため, 攻撃実験に関する研究もこれから非常に重要である．

公開鍵や秘密鍵を圧縮しようと特殊な符号を採用したり, 距離の定義を変える提案も多くある．これらは解読攻撃を

受けることも多く, 提案されて日の浅い暗号・署名方式については注視が必要である.

第 3 章の参考文献

[1] M. Alekhnovich. More on average case vs approximation complexity. *Computational Complexity*, 20(4):755–786, 2011.

[2] C. Aguilar Melchor, O. Blazy, N. Aragon, J.-C. Deneuville, S. Bettaieb, P. Gaborit, L. Bidoux, G. Zemor. RQC NIST PQC 標準化 Round 1 投稿資料. `https://csrc:nist:gov/projects/post-quantum-cryptography/round-1-submissions`

[3] B. Applebaum, D. Cash, C. Peikert, A. Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In S. Halevi, editor, *Advances in Cryptology - CRYPTO 2009, 29th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2009. Proceedings*, volume 5677 of *Lecture Notes in Computer Science*, pages 595–618. Springer, 2009.

[4] S. Arora and R. Ge. New algorithms for learning in presence of errors. In L. Aceto, M. Henzinger, and J. Sgall, editors, *Automata, Languages and Programming - 38th International Colloquium, ICALP 2011, Zurich, Switzerland, July 4-8, 2011, Proceedings, Part I*, volume 6755 of *Lecture Notes in Computer Science*, pages 403–415. Springer, 2011.

[5] N. Aragon, O. Ruatta, P. Gaborit, G. Zemor, A. Hauteville. RankSign NIST PQC 標準化 Round 1 投稿資料. `https://csrc:nist:gov/projects/post-quantum-cryptography/round-1-submissions`

[6] G. Banegas, P. S. L. M. Barreto, B. O. Boidje, P.-L. Cayrel, G. N. Dione, K. Gaj, C. T. Gueye, R. Haeussler, J. B. Klamti, O. N'diaye, D. T. Nguyen, E. Persichetti, J. E. Ricardini. DAGS NIST PQC 標準化 Round 1 投稿資料. `https://csrc:nist:gov/projects/post-quantum-cryptography/round-1-submissions`

[7] É. Barelli and A. Couvreur. An efficient structural attack on NIST submission DAGS. In Peyrin and Galbraith [44], pages 93–118.

[8] D. J. Bernstein, T. Chou, T. Lange, I. von Maurich, R. Misoczki, R. Niederhagen, E. Persichetti, C. Peters, P. Schwabe, N. Sendrier, J. Szefer, W. Wang. Classic McEliece NIST PQC 標準化 Round 1 投稿資料. `https://csrc:nist:gov/projects/post-quantum-cryptography/round-1-submissions`

[9] A. Blum, M. L. Furst, M. J. Kearns, and R. J. Lipton. Cryptographic primitives based on hard learning problems. In D. R. Stinson, editor, *Advances in Cryptology - CRYPTO '93, 13th Annual International Cryptology Conference, Santa Barbara, California, USA, August 22-26, 1993, Proceedings*, volume 773 of *Lecture Notes in Computer Science*, pages 278–291. Springer, 1993.

[10] A. Becker, A. Joux, A. May, and A. Meurer. Decoding random binary linear codes in $2^{n/20}$: How

$1 + 1 = 0$ improves information set decoding. In D. Pointcheval and T. Johansson, editors, *Advances in Cryptology - EUROCRYPT 2012 - 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15-19, 2012. Proceedings*, volume 7237 of *Lecture Notes in Computer Science*, pages 520–536. Springer, 2012.

[11] A. Blum, A. Kalai, and H. Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. *J. ACM*, 50(4):506–519, 2003.

[12] D. J. Bernstein and T. Lange. Never trust a bunny. In J.-H. Hoepman and I. Verbauwhede, editors, *Radio Frequency Identification. Security and Privacy Issues - 8th International Workshop, RFIDSec 2012, Nijmegen, The Netherlands, July 2-3, 2012, Revised Selected Papers*, volume 7739 of *Lecture Notes in Computer Science*, pages 137–148. Springer, 2012.

[13] D. J. Bernstein, T. Lange, and C. Peters. Smaller decoding exponents: Ball-collision decoding. In P. Rogaway, editor, *Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings*, volume 6841 of *Lecture Notes in Computer Science*, pages 743–760. Springer, 2011.

[14] L. Both and A. May. Optimizing BJMM with nearest neighbors: Full decoding in $2^{2n/21}$ and McEliece security. In *WCC 2017*, 2017. See `http://wcc2017.suai.ru/proceedings.html`.

[15] L. Both and A. May. Decoding linear codes with high error rate and its impact for LPN security. In Lange and Steinwandt [37], pages 25–46.

[16] E. R. Berlekamp, R. J. McEliece, and H. C. A. van Tilborg. On the inherent intractability of certain coding problems. *IEEE Trans. Information Theory*, 24(3):384–386, 1978.

[17] S. Bogos and S. Vaudenay. Optimization of LPN solving algorithms. In J.-H. Cheon and T. Takagi, editors, *Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I*, volume 10031 of *Lecture Notes in Computer Science*, pages 703–728, 2016.

[18] N. Courtois, M. Finiasz, and N. Sendrier. How to achieve a McEliece-based digital signature scheme. In C. Boyd, editor, *Advances in Cryptology - ASIACRYPT 2001, 7th International Conference on the Theory and Application of Cryptology and Information Security, Gold Coast, Australia, December 9-13, 2001, Proceedings*, volume 2248 of *Lecture Notes in Computer Science*, pages 157–174. Springer, 2001.

[19] S. Devadas, L. Ren, and H. Xiao. On iterative collision search for LPN and subset sum. In Y. Kalai and L. Reyzin, editors, *Theory of Cryptography - 15th International Conference, TCC 2017, Baltimore, MD, USA, November 12-15, 2017, Proceedings, Part II*, volume 10678 of *Lecture Notes in Computer Science*, pages 729–746. Springer, 2017.

[20] T. Debris-Alazard and J.-P. Tillich. Two attacks on rank metric code-based schemes: RankSign and an IBE scheme. In Peyrin and Galbraith [44], pages 62–92.

[21] A. Esser, F. Heuer, R. Kübler, A. May, and C. Sohler. Dissection-BKW. In H. Shacham and A. Boldyreva, editors, *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part II*, volume 10992 of *Lecture Notes in Computer Science*, pages 638–666. Springer, 2018.

[22] A. Esser, R. Kübler, and A. May. LPN decoded. In J. Katz and H. Shacham, editors, *Advances in*

Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part II, volume 10402 of *Lecture Notes in Computer Science*, pages 486–514. Springer, 2017.

[23] J.-C. Faugère, V. Gauthier-Umaña, A. Otmani, L. Perret, and J.-P. Tillich. A distinguisher for high rate McEliece cryptosystems. In *2011 IEEE Information Theory Workshop, ITW 2011, Paraty, Brazil, October 16-20, 2011*, pages 282–286. IEEE, 2011.

[24] J.-C. Faugère, V. Gauthier-Umaña, A. Otmani, L. Perret, and J.-P. Tillich. A distinguisher for high-rate McEliece cryptosystems. *IEEE Trans. Information Theory*, 59(10):6830–6844, 2013.

[25] Q. Guo, T. Johansson, and C. Löndahl. Solving LPN using covering codes. In P. Sarkar and T. Iwata, editors, *Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I*, volume 8873 of *Lecture Notes in Computer Science*, pages 1–20. Springer, 2014.

[26] H. Gilbert, M. J. B. Robshaw, and Y. Seurin. HB$^{\#}$: Increasing the security and efficiency of HB^{+}. In N. P. Smart, editor, *Advances in Cryptology - EUROCRYPT 2008, 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Istanbul, Turkey, April 13-17, 2008. Proceedings*, volume 4965 of *Lecture Notes in Computer Science*, pages 361–378. Springer, 2008.

[27] J. Håstad. Some optimal inapproximability results. *J. ACM*, 48(4):798–859, 2001.

[28] S. Heyse, Eike Kiltz, Vadim Lyubashevsky, Christof Paar, and Krzysztof Pietrzak. Lapin: An efficient authentication protocol based on ring-LPN. In A. Canteaut, editor, *Fast Software Encryption - 19th International Workshop, FSE 2012, Washington, DC, USA, March 19-21, 2012. Revised Selected Papers*, volume 7549 of *Lecture Notes in Computer Science*, pages 346–365. Springer, 2012.

[29] P. Kirchner. Improved generalized birthday attack. *IACR Cryptology ePrint Archive*, 2011:377, 2011.

[30] E. Kirshanova. Improved quantum information set decoding. In Lange and Steinwandt [37], pages 507–527.

[31] K. Kaminakaya and N. Kunihiro. LPN 問題に対する BKW アルゴリズムの拡張. *SCIS 2015*, pages 3E1–3, 2015.

[32] J. Katz, J. S. Shin, and A. D. Smith. Parallel and Concurrent Security of the HB and HB^{+} protocols. *J. Cryptology*, 23(3):402–421, 2010.

[33] G. Kachigar and J.-P. Tillich. Quantum information set decoding algorithms. In T. Lange and T. Takagi, editors, *Post-Quantum Cryptography - 8th International Workshop, PQCrypto 2017, Utrecht, The Netherlands, June 26-28, 2017, Proceedings*, volume 10346 of *Lecture Notes in Computer Science*, pages 69–89. Springer, 2017.

[34] P. J. Lee and E. F. Brickell. An observation on the security of McEliece's public-key cryptosystem. In C. G. Günther, editor, *Advances in Cryptology - EUROCRYPT '88, Workshop on the Theory and Application of of Cryptographic Techniques, Davos, Switzerland, May 25-27, 1988, Proceedings*, volume 330 of *Lecture Notes in Computer Science*, pages 275–280. Springer, 1988.

[35] Y. Li, R. H. Deng, and X. Wang. On the equivalence of McEliece's and Niederreiter's public-key cryptosystems. *IEEE Trans. Information Theory*, 40(1):271–273, 1994.

[36] É. Levieil and P.-A. Fouque. An improved LPN algorithm. In R. De Prisco and M. Yung, editors, *Security and Cryptography for Networks, 5th International Conference, SCN 2006, Maiori, Italy, September 6-8, 2006, Proceedings*, volume 4116 of *Lecture Notes in Computer Science*, pages 348–359. Springer, 2006.

[37] T. Lange and R. Steinwandt, editors. *Post-Quantum Cryptography - 9th International Conference, PQCrypto 2018, Fort Lauderdale, FL, USA, April 9-11, 2018, Proceedings*, volume 10786 of *Lecture Notes in Computer Science*. Springer, 2018.

[38] V. Lyubashevsky. The parity problem in the presence of noise, decoding random linear codes, and the subset sum problem. In C. Chekuri, K. Jansen, J. D. P. Rolim, and L. Trevisan, editors, *Approximation, Randomization and Combinatorial Optimization, Algorithms and Techniques, 8th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, APPROX 2005 and 9th InternationalWorkshop on Randomization and Computation, RANDOM 2005, Berkeley, CA, USA, August 22-24, 2005, Proceedings*, volume 3624 of *Lecture Notes in Computer Science*, pages 378–389. Springer, 2005.

[39] R. J. McEliece. A public-key cryptosystem based on algebraic coding theory. *Jet Propulsion Laboratory DSN Progress Report*, 42–44:114–116, January and February 1978. `https://ipnpr.jpl.nasa.gov/progress_report2/42-44/44N.PDF`.

[40] A. May, A. Meurer, and E. Thomae. Decoding random linear codes in $\tilde{O}(2^{0.054n})$. In D. H. Lee and X. Wang, editors, *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*, volume 7073 of *Lecture Notes in Computer Science*, pages 107–124. Springer, 2011.

[41] A. May and I. Ozerov. On computing nearest neighbors with applications to decoding of binary linear codes. In E. Oswald and M. Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I*, volume 9056 of *Lecture Notes in Computer Science*, pages 203–228. Springer, 2015.

[42] H. Niederreiter. Knapsack-type cryptosystems and algebraic coding theory. *Problemy Upravleniia i Teorii Informatsii*, 15:19–34, 1986.

[43] R. Nojima, H. Imai, K. Kobara, and K. Morozov. Semantic security for the McEliece cryptosystem without random oracles. *Des. Codes Cryptography*, 49(1-3):289–305, 2008.

[44] T. Peyrin and S. D. Galbraith, editors. *Advances in Cryptology - ASIACRYPT 2018 - 24th International Conference on the Theory and Application of Cryptology and Information Security, Brisbane, QLD, Australia, December 2-6, 2018, Proceedings, Part I*, volume 11272 of *Lecture Notes in Computer Science*. Springer, 2018.

[45] E. Prange. The use of information sets in decoding cyclic codes. *IRE Trans. Information Theory*, 8(5):5–9, 1962.

[46] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6):34:1–34:40, 2009. Preliminary version was presented at STOC 2005.

[47] J. Stern. A method for finding codewords of small weight. In G. D. Cohen and J. Wolfmann, editors,

Coding Theory and Applications, 3rd International Colloquium, Toulon, France, November 2-4, 1988, Proceedings, volume 388 of *Lecture Notes in Computer Science*, pages 106–113. Springer, 1988.

[48] B. Zhang, L. Jiao, and M. Wang. Faster algorithms for solving LPN. In M. Fischlin and J.-S. Coron, editors, *Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I*, volume 9665 of *Lecture Notes in Computer Science*, pages 168–195. Springer, 2016.

第 4 章

多変数多項式に基づく暗号技術

多変数多項式に基づく公開鍵暗号は, 多変数公開鍵暗号 (Multivariate Public Key Cryptosystems）と呼ばれる. 多変数公開鍵暗号は, 暗号方式, 署名方式の構成に用いることができ, その多くは (有限体上の）多変数多項式写像の代入評価を暗号化および署名検証に用いているため, 効率的なアルゴリズムを持つことが特徴の一つである.

多変数公開鍵暗号の安全性は主に, MP 問題および IP 問題を解く計算の困難性に拠っている. 本報告書では, MP 問題に対する解読方法と, 現在提案されている多くの方式が採用している双極型システムを中心に解説する. 双極型システムによる方式は, 安全性のために MQ 問題の解読困難性を必要とする. また, IP 問題の拡張問題である EIP 問題の解読困難性も安全性を高めるために必要となる.

双極型システム以外の方式としては, MP 問題の解読困難性を安全性の根拠とする対話型認証方式を紹介する. Fiat-Shamir 変換を用いることで多変数多項式に基づく署名方式が構成される.

4.1 多変数多項式に基づく暗号技術の安全性の根拠となる問題

4.1.1 多変数公開鍵暗号について

1984 年に Ong らが提案した署名方式 [28] において, 初めて公開鍵暗号に “多変数”が取り入れられた. この方式は, メッセージ m に対し, 合成数 N を法とする 2 変数合同方程式

$$x^2 + hy^2 \equiv m \bmod N$$

の解 $(x, y) = (x_0, y_0)$ を署名とするものであり, N の素因数分解の困難性を安全性のベースとしている. しかし, この署名方式は, 1987 年に Pollard と Schnorr によって N の素因数分解を用いることなく解読された [34]. その後, 変数の個数を増やした方式 [38] や, 非可換環を利用した方式 [37] などに拡張されたが, いずれも既に解読されている. これらの方式も, 素因数分解問題を安全性のベースとして構成されており, 多変数公開鍵暗号には含まれない.

1988 年に松本と今井が提案した暗号方式 [27] において, 初めて多変数公開鍵暗号のアイデアが示された. この暗号方式は松本-今井方式 (あるいは C^* 方式）と呼ばれている. 1995 年, Patarin は松本-今井方式を解読し [29], 翌 96 年に松本-今井方式を拡張した HFE (Hidden Field Equation）を提案した [30]. HFE 自体は, 1999 年に Kipnis と Shamir によって解読された [26] が, HFE_v^- [30] など, HFE の派生方式には致命的な攻撃を免れているものがある.

一方, Patarin は 1997 年に (balanced) Oil-Vineger 方式と呼ばれる署名方式 [31] を発表した. 松本-今井方式や HFE は, 基礎体となる有限体とは別に, その拡大体を補助的に利用して構成しているのに対し, Oil-Vineger 方式は, 基礎体以外の有限体を構成に利用しない点が一つの特徴である. (前者のような構成法を big field 法, 後者のような構成法を single field 法という.) Oil-Vineger 方式は, 1998 年に Kipnis と Shamir によって解読された [25] が, 安全性を強化

した UOV (Unbalanced Oil-Vinegar 方式）が Kipnis, Patarin らによって翌 99 年に提案されている [24]. また, 2005 年には UOV を多層化した署名方式 Rainbow が, Ding と Schmidt により提案されている [12].

これまで説明した方式はいずれも, 双極型システム [10] と呼ばれる多変数公開鍵暗号の構成法を用いている. しかし, 多変数公開鍵暗号は必ずしも双極型システムで構成される必要はなく, 実際, ソニー株式会社の作本, 白井, 樋渡が 2011 年に提案した 3-pass と 5-pass の認証方式 [35] は双極型システムを利用していない. この 5-pass の認証方式に Fiat-Shamir 変換を適用した署名方式が, 2016 年に提案されている [6].

暗号方式では, HFE 以降様々な方式が提案されたが, 現在致命的な攻撃を免れているもの (および提案年）としては, Simple Matrix 方式 (2013) [41], EFC (の変種）(2016) [40], HFERP (2018) [23], EFLASH (2018) [4] がある.

4.1.2 多変数公開鍵暗号の安全性の根拠となる問題とその解読計算量

$\mathbb{F}_q$ で位数 q の有限体を表し, $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ で (代数的に独立な）変数の集合を表すものとする. $\mathbf{x}$ に関する $\mathbb{F}_q$ 上の多変数多項式の組, すなわち, 多変数多項式 $p_i(\mathbf{x})$ $(i = 1, 2, \ldots, m)$ により, $P(\mathbf{x}) = (p_1(\mathbf{x}), p_2(\mathbf{x}), \ldots, p_m(\mathbf{x}))$ と表されるものを ($\mathbb{F}_q$ 上の）多変数多項式系と呼ぶことにする. この多変数多項式系 $P(\mathbf{x})$ は代入評価により, $\mathbb{F}_q^n$ から $\mathbb{F}_q^m$ への写像を構成する. この (多変数多項式）写像を $P : \mathbb{F}_q^n \to \mathbb{F}_q^m$ と表すことにする. 記号の違いにより, 多変数多項式系 $P(\mathbf{x})$ と多変数多項式写像 P を区別する.

多変数公開鍵暗号の正式な定義はないと思われるが, ここでは MP 問題 (Multivariate Polynomial）または IP 問題 (Isomorphisms of Polynomials）の解読困難性をベースとした公開鍵暗号のことを多変数公開鍵暗号と理解することにする. MP 問題, IP 問題は次のように記述される.

MP 問題　多変数多項式系 $P(\mathbf{x}) = (p_1(\mathbf{x}), p_2(\mathbf{x}), \ldots, p_m(\mathbf{x}))$ と $\mathbf{d} \in \mathbb{F}_q^m$ に対して, 変数 $\mathbf{x}$ に関する方程式系

$$P(\mathbf{x}) = \mathbf{d}$$

の解 (存在するなら）少なくとも 1 つ求めよ.

IP 問題　S, T をそれぞれ, $\mathbb{F}_q^n$, $\mathbb{F}_q^m$ 上のアフィン同型写像とする. 多変数多項式系 $P(\mathbf{x}) = (p_1(\mathbf{x}), p_2(\mathbf{x}), \ldots, p_m(\mathbf{x}))$ に対し, 多変数多項式系 $\tilde{P}(\mathbf{x})$ を合成により, $\tilde{P}(\mathbf{x}) = T \circ P(\mathbf{x}) \circ S$ で定める. このとき, $P(\mathbf{x})$, $\tilde{P}(\mathbf{x})$ の情報から S, T を求めよ.

MP 問題は, $\mathbf{d}$ を左辺に移行して $P(\mathbf{x})$ に吸収させることができるので, 右辺を $\mathbf{0}^m$ (m 次の零ベクトル）として表現される場合もある. MP 問題において, $P(\mathbf{x})$ のすべての成分 $p_i(\mathbf{x})$ が 1 次以下となる場合, MP 問題は単に線形方程式を解く問題となり, ガウスの消去法などで多項式時間で解を求めることが可能である. よって, MP 問題を考える場合は通常, $p_i(\mathbf{x})$ の次数はすべて 2 以上であると仮定する. 特に, $p_i(\mathbf{x})$ の次数がすべて 2 となるとき, MP 問題は MQ 問題 (Multivariate Quadratic）と呼ばれる. $\mathbb{F}_q = \mathbb{F}_2$ の場合, MQ 問題は NP 完全であることが知られている [20].

上記の IP 問題は IP2S 問題とも呼ばれる. IP 問題の T を恒等写像で固定し, S のみを求める IP 問題の別バージョンが存在し, これを IP1S 問題と呼ぶ. Patarin は, IP2S 問題や IS1S 問題を利用した認証方式を提案している [30]. IP 問題をベースとした公開鍵方式が, MP 問題 (MQ 問題）のそれに比べて提案された数が圧倒的に少ないことと, NIST PQC 標準化プロジェクトに IP 問題をベースとした公開鍵方式が投稿されなかったことなどを考慮し, 本報告書では, IP 問題についてこれ以上考察しないこととする. 但し, IP 問題の拡張問題である以下の EIP 問題 (Extended Isomorphisms of Polynomials）は, MP 問題 (MQ 問題）をベースとする公開鍵方式の構造の 1 つである双極型システムの安全性に関わることがあるため, 次節で言及する.

EIP 問題　多変数多項式系 $F(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x}))$ が, $\mathbb{F}_q^n, \mathbb{F}_q^m$ 上のアフィン同型写像 S, T と多変数多項式系のあるクラス $\mathcal{C}$ に属する多変数多項式系 $G(\mathbf{x})$ により, $F(\mathbf{x}) = T \circ G(\mathbf{x}) \circ S$ で与えられているとする. このとき, 分解 $F(\mathbf{x}) = T' \circ G'(\mathbf{x}) \circ S'$ で, S', T' は $\mathbb{F}_q^n, \mathbb{F}_q^m$ 上のアフィン同型写像, $G'(\mathbf{x}) \in \mathcal{C}$ なるものを見つけよ.

本節の残りでは, MP 問題に対する解読手法とその計算量について解説する. MP 問題に対する一般的解読方法として, 総当たり法や XL [42], Gröbner 基底攻撃 [10] が知られている. Gröbner 基底攻撃とは, イデアルの Gröbner 基底計算 [16, 17] を利用する MP 問題の解読方法である. 解読したい MP 問題の右辺の $\mathbf{d}$ を左辺に移行し, $P(\mathbf{x})$ の中に吸収させてしまうことにより, MP 問題は,

$$P(\mathbf{x}) = (p_1(\mathbf{x}), p_2(\mathbf{x}), \ldots, p_m(\mathbf{x})) = \mathbf{0}^m$$

の求解問題と表現できる. これは, イデアル $I = \langle p_1(\mathbf{x}), p_2(\mathbf{x}), \ldots, p_m(\mathbf{x}) \rangle$ で定義される代数多様体の $\mathbb{F}_q$-有理点を求める問題と同値になる. さらに, イデアル I は $I' = \langle p_1(\mathbf{x}), p_2(\mathbf{x}), \ldots, p_m(\mathbf{x}), x_1^q - x_1, x_2^q - x_2, \ldots, x_n^q - x_n \rangle$ に変更することができる.

ここで, イデアル I' を通常の辞書式項順序に関する Gröbner 基底に変形する. I' の Krull 次元が 0 となる, すなわち, I' が $\mathbb{F}_q[\mathbf{x}]$ の極大イデアルとなるとき, I' の Gröbner 基底 $\tilde{I}$ は以下の形で表される.

$$\tilde{I} = \left\{ \begin{array}{c} \tilde{p}_1(x_1, x_2, \ldots, x_n), \\ \vdots \\ \tilde{p}_{i_2-1}(x_1, x_2, \ldots, x_n), \\ \tilde{p}_{i_2}(x_2, x_3, \ldots, x_n), \\ \vdots \\ \tilde{p}_{i_3}(x_3, x_4, \ldots, x_n), \\ \vdots \\ \tilde{p}_{l-1}(x_{n-1}, x_n), \\ \tilde{p}_l(x_n) \end{array} \right\}$$

この形から, $\tilde{I}$ で定義される代数多様体の有理点 (の 1 点) は x_n の値から逐次的に x_1 まで求めることができる. この有理点を $\mathbf{v} \in \mathbb{F}_q^n$ とすると, これは I で定義される代数多様体の $\mathbb{F}_q$-有理点でもあるので, $\mathbf{x} = \mathbf{v}$ が MQ 問題の解となる. これが Gröbner 基底攻撃の概要である.

Gröbner 基底攻撃の計算量を記述するための準備を行う. $p_{m+1}(\mathbf{x}) = x_1^q - x_1, p_{m+2}(\mathbf{x}) = x_2^q - x_2, \ldots, p_{m+n}(\mathbf{x}) = x_n^q - x_n$ とおく. これにより, $I' = \langle p_1(\mathbf{x}), p_2(\mathbf{x}), \ldots, p_{m+n}(\mathbf{x}) \rangle$ と表すことができる. 各 $p_i(\mathbf{x})$ $(i = 1, 2, \ldots, m+n)$ に対し, その最高次斉次部分を $p_i^h(\mathbf{x})$ (d_i 次斉次多項式) と表し, $\mathbb{F}_q[\mathbf{x}]$ の斉次イデアルを J を

$$J = \langle p_1^h(\mathbf{x}), p_2^h(\mathbf{x}), \ldots, p_{m+n}^h(\mathbf{x}) \rangle$$

で定める. $d \geq 0$ に対し, $\mathbb{F}_q[\mathbf{x}]_d$ で d-次斉次多項式のなす $\mathbb{F}_q[\mathbf{x}]$ の部分ベクトル空間を表し, $J_d := J \cap \mathbb{F}_q[\mathbf{x}]_d$ とする. 次数環 $\mathbb{F}_q[\mathbf{x}]/J = \bigoplus_{d=0}^{\infty} \mathbb{F}_q[\mathbf{x}]_d/J_d$ の Hilbert 級数は

$$\mathrm{HS}_{\mathbb{F}_q[\mathbf{x}]/J}(t) = \sum_{d=0}^{\infty} \dim_{\mathbb{F}_q}(\mathbb{F}_q[\mathbf{x}]_d/J_d)\, t^d \in \mathbb{Z}[[t]] \quad \text{(形式的べき級数)}$$

で定義される. J の Krull-次元が 0 となるとき, $\mathrm{HS}_{\mathbb{F}_q[\mathbf{x}]/J}(t)$ は多項式となる. このとき, $d_{\mathrm{reg}} = \deg(\mathrm{HS}_{\mathbb{F}_q[\mathbf{x}]/J}(t)) + 1$ とおき, これを正則性の次数 (degree of regularity) と呼ぶ. この準備の下, Gröbner 基底攻撃の計算量は以下のように

表される [1].

$$\mathcal{O}\left(\left(\begin{array}{c} n+d_{\mathrm{reg}} \\ n \end{array}\right)^{\omega}\right). \tag{4.1}$$

ここで, $2 \leq \omega \leq 3$ は連立線形方程式を解くために利用するアルゴリズムにより定まる定数である.

任意の $S(t) \in \mathbb{Z}[[t]]$ に対し, $[S(t)]_+ \in \mathbb{Z}_{>0}[[t]]$ で, $S(t)$ の最初に現れる非正係数の次数以降 (この項も含む) を切り捨てた多項式を表すことにする. もし,

$$\mathrm{HS}_{\mathbb{F}_q[\mathbf{x}]/J}(t) = \left[\frac{\prod_{i=1}^{m+n}(1-t^{d_i})}{(1-t)^n}\right]_+ = \left[\left(\prod_{i=1}^{m}(1-t^{d_i})\right)\left(\frac{1-t^q}{1-t}\right)^n\right]_+$$

を満たすならば, $p_1(\mathbf{x}), p_2(\mathbf{x}), \ldots, p_{m+n}(\mathbf{x})$ は半正則であるといわれる. q がある程度大きいならば, 任意の m, n に対して, $p_1(\mathbf{x}), p_2(\mathbf{x}), \ldots, p_m(\mathbf{x})$ の係数をランダムに選ぶと, $p_1(\mathbf{x}), p_2(\mathbf{x}), \ldots, p_{m+n}(\mathbf{x})$ は多くの場合に半正則となることが実験的に確かめられる. この現象を理論的に保証するものではないが, これに関連する予想として Fröberg の予想 [19] というものがあり, この現象の根拠に用いられる場合がある. (予想の正確な記述には準備が必要なため, ここでは述べない.) ところが, 多変数公開鍵暗号 (特に暗号方式) に現れる多くの多変数多項式系に対しては, 半正則にはならず, グレブナー基底攻撃が半正則な場合よりも効果的に働いてしまうことは興味深い事実である. $p_i(\mathbf{x})$ がすべて 2 次で, 半正則ならば, m が一定の場合や $m = \alpha n$ (α: 定数) の場合, Gröbner 基底攻撃の計算量は $\mathcal{O}(2^{\mathcal{O}(n)})$ となることが知られている [1].

Gröbner 基底攻撃には別の計算量も知られている. この記述も準備が必要である. まず, 次数環 A を $A = \mathbb{F}_q[\mathbf{x}]/(x_1^q, x_2^q, \ldots, x_n^q)$ で定め, A_k で A の k 次斉次部分を表すことにする. 任意の正の整数 d に対し, $\psi_d : \bigoplus_{i=1}^m A_{d-d_i} \to A_d$ を $\psi_d(b_1, b_2, \ldots, b_m) = \sum_{i=1}^m b_i\, p_i^h(\mathbf{x})$ で定める. ($p_{m+1}^h(\mathbf{x}) = x_1^q, p_{m+2}^h(\mathbf{x}) = x_2^q, \ldots, p_{m+n}^h(\mathbf{x}) = x_n^q$ は使わないことに注意.) R_d を $R_d = \ker \psi_d$, すなわち, $\sum_{i=1}^m b_i\, p_i^h(\mathbf{x}) = 0$ となる $(b_1, b_2, \ldots, b_m) \in \bigoplus_{i=1}^m A_{d-d_i}$ の全体とする. T_d を以下の形の元で生成される R_d の部分空間とする.

$$\begin{array}{ll} (1) & (0, \ldots, 0, b\, p_j^h(\mathbf{x}), 0, \ldots, 0, b\, p_i^h(\mathbf{x}), 0 \ldots, 0) \quad (b \in A_{d-d_i-d_j}), \\ (2) & (0, \ldots, 0, b\,((p_i^h)^{q-1} - 1), 0, \ldots, 0) \quad (b \in A_{d-d_i(q-1)}). \end{array}$$

ここで, (1) の非零成分は左から第 i 成分と第 j 成分であり, (2) の非零成分は第 i 成分である. すべての正の整数 d に対し, 商 R_d/T_d を考えたとき, $R_d/T_d \neq 0$ となる最小の d が存在する. この d を d_{FF} と表し, これを最小降下次数 (first fall degree) と呼ぶ [11, 15]. 言い換えると, 非自明な $\sum_{i=1}^m b_i\, p_i^h(\mathbf{x}) = 0$ なる関係を生み出す最小の次数である. この準備の下, 2 つ目の Gröbner 基底攻撃の計算量は以下のようになる.

$$\mathcal{O}\left(\left(\begin{array}{c} n+d_{\mathrm{FF}} \\ n \end{array}\right)^{\omega}\right). \tag{4.2}$$

ここで, $2 \leq \omega \leq 3$ は連立線形方程式を解くために利用するアルゴリズムにより定まる定数である. 同じ Gröbner 基底攻撃に対して, (4.1), (4.2) の 2 つの計算量があるのは, 計算量の見積もり方 (あるいは使うアルゴリズム) が少し違うためである. (どちらも Faugère の F4/F5 アルゴリズムの計算量と説明はされている.) d_{reg} と d_{FF} は, 多くの場合で同じ値になると予想されている.

Gröbner 基底攻撃は, 総当たり法とミックスすることで攻撃がより効率的になることがある. いくつかの変数に適当に値を代入し, より少ない変数の MP 問題を構成して Gröbner 基底攻撃を適用する. 解が見つからなければ, 変数への代入からやり直すという攻撃である. これをハイブリッド攻撃という.

4.2 代表的な多変数多項式に基づく暗号方式の説明

4.2.1 双極型システム

多変数公開鍵暗号の多くの方式で採用されている双極型システム [10] と呼ばれる構造について説明する．多変数多項式系 $P(\mathbf{x})$ のとり方によっては, 多くの $\mathbf{d} \in \mathbb{F}_q^m$ に対する MP 問題が効率的に計算できる場合がある．例えば, $K = \mathbb{F}_{q^n}$ を $\mathbb{F}_q$ の n 次拡大体とし, $\mathbb{F}_q$-線形同型写像 $\phi : \mathbb{F}_q^n \xrightarrow{\sim} K$ を 1 つ固定する．K 上のべき乗写像 $\mathcal{P}_l : K \ni X \mapsto X^l \in K$ (l : 正の整数) を考え, 多変数多項式写像 P_l を $P_l = \phi^{-1} \circ \mathcal{P}_l \circ \phi$ と定義すると, これに対応する多変数多項式系 $P_l(\mathbf{x})$ に対しては, MP 問題が効率的に解読できる．実際, $P_l(\mathbf{x}) = \mathbf{d}$ の方程式系は, ϕ を通して, K 上の 1 変数方程式 $X^l = \mathbf{d}'$ の形に変換することができ, これは l 乗根計算を用いて効率的に解を求めることができる．このような性質を持つ $P(\mathbf{x})$ は単独ではなく, 系統的に構成できる場合がある．上の例の場合も, l の値を動かすことで, $P_l(\mathbf{x})$ 達のなすクラス (=多変数多項式系のある集合）が構成できる．

双極型システムでは, まず, MP 問題が効率的に計算できる多変数多項式系 $P(\mathbf{x})$ のクラス $\mathcal{C}_{\mathrm{cent}}$ を 1 つ固定する．$G(\mathbf{x}) \in \mathcal{C}_{\mathrm{cent}}$ と $\mathbb{F}_q^n, \mathbb{F}_q^m$ 上のアフィン同型写像 S, T を合成した多変数多項式系 $F(\mathbf{x}) = T \circ G(\mathbf{x}) \circ S$ をトラップドア付き一方向関数として利用しようというのが, 双極型システムのアイデアである．ただし, $F(\mathbf{x})$ が実際にトラップドア付き一方向関数となるかどうかは $\mathcal{C}_{\mathrm{cent}}$ のとり方に依存する．双極型システムの鍵生成は次のように行う．

1. $G(\mathbf{x}) \in \mathcal{C}_{\mathrm{cent}}$ をランダムに選ぶ.
2. $\mathbb{F}_q^n, \mathbb{F}_q^m$ 上のアフィン同型写像 S, T をランダムに選ぶ.
3. $F(\mathbf{x}) = T \circ G(\mathbf{x}) \circ S$ とする.

このとき, 公開鍵は $F(\mathbf{x})$, 秘密鍵は $G(\mathbf{x}), S, T$ となる．(実践的には, S, T よりも S^{-1}, T^{-1} を秘密鍵として保持したほうがよい.) $G(\mathbf{x})$ を ($F(\mathbf{x})$ の）中心写像とよぶ．双極型システムは暗号方式, 署名方式の構成に用いられる．暗号方式の暗号化・復号は次のように行う．

暗号化　平文 $M \in \mathbb{F}_q^n$ に対し, $C = F(M)$ を計算する．C が暗号文となる.
復号　暗号文 $C \in \mathbb{F}_q^m$ に対し, (1) $B_1 = T^{-1}(C)$, (2) $B_2 = G^{-1}(B_1)$, (3) $M' = S^{-1}(B_2)$ の順に計算する．M' が平文と一致する.

復号が成功するためには基本的に, $G(\mathbf{x})$ (あるいは $F(\mathbf{x})$）が単射である必要がある．少し条件を緩めて, 「$G(\mathbf{x})$ (あるいは $F(\mathbf{x})$）の逆像の要素がすべて計算可能」とすることもできる．この場合, M' が複数得られることになるので, ハッシュ値などを用いて平文 M と一致する M' を特定する．署名方式の場合, 署名生成・検証は次のように行う．

署名生成　メッセージ $M \in \mathbb{F}_q^m$ に対し, (1) $B_1 = T^{-1}(M)$, (2) $B_2 = G^{-1}(B_1)$, (3) $\sigma = S^{-1}(B_2)$ の順に計算する．σ が署名となる.
検証　署名 $\sigma \in \mathbb{F}_q^n$ に対し, $M' = F(\sigma)$ を計算する．$M = M'$ ならば署名を受理し, それ以外は不受理とする.

署名生成がいつでも実行できるためには, どのようなメッセージ $M \in \mathbb{F}_q^m$ に対しても, $B_2 = G^{-1}(B_1)$ の計算ができる, すなわち, $G(\mathbf{x})$ (あるいは $F(\mathbf{x})$）が全射である必要がある.

双極型システムを用いて構成された方式は, $\mathcal{C}_{\mathrm{cent}}$ の選び方に応じて方式の名称が決まる．例えば, $\mathcal{C}_{\mathrm{cent}} = \{P_l(\mathbf{x}) \mid l$ の q-ハミング重みが 2$\}$ とした双極型システムの暗号方式が, 松本-今井方式である．(q-ハミング重みの条件は, $F(\mathbf{x}), G(\mathbf{x})$ が 2 次多項式からなることを保証する.)

双極型システムが安全であるためには, MP 問題の解読困難性が必要となる. 実際, 暗号方式における MP 問題 $F(\mathbf{x}) = C$ や, 署名方式における MP 問題 $F(\mathbf{x}) = M$ が許容時間内で求解可能であるならば, 双極型システムの安全性は崩れてしまう. これと関連するのが, IP 問題の拡張問題である EIP 問題である.

EIP 問題 多変数多項式系 $F(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x}))$ が, $\mathbb{F}_q^n, \mathbb{F}_q^m$ 上のアフィン同型写像 S, T と多変数多項式系のあるクラス $\mathcal{C}$ に属する多変数多項式系 $G(\mathbf{x})$ により, $F(\mathbf{x}) = T \circ G(\mathbf{x}) \circ S$ で与えられているとする. このとき, 分解 $F(\mathbf{x}) = T' \circ G'(\mathbf{x}) \circ S'$ で, S', T' は $\mathbb{F}_q^n, \mathbb{F}_q^m$ 上のアフィン同型写像, $G'(\mathbf{x}) \in \mathcal{C}$ なるものを見つけよ.

双極型システムにおいて, $\mathcal{C} = \mathcal{C}_{\mathrm{cent}}$ としたときの EIP 問題が解読可能であるならば, 分解 $F(\mathbf{x}) = T' \circ G'(\mathbf{x}) \circ S'$ を用いて, 対応する MP 問題は容易に解読可能となる. EIP 問題の解読困難性は, クラス $\mathcal{C}$ の選び方に依存しており, $\mathcal{C} = \mathcal{C}_{\mathrm{cent}}$ であるならば, その方式に応じて個々に解析される必要がある.

4.2.2 Simple field 法と big field 法

中心写像 $G(\mathbf{x})$ や 公開鍵 $F(\mathbf{x})$ は理論的には一般次数の多変数多項式が利用できるが, 実践的にはよく 2 次多項式が利用される. これは鍵長を抑えるためである. ここでは, 2 次多項式からなる中心写像 (および公開鍵) を利用した多変数公開鍵暗号の方式を 2 つ紹介する. 双極型システムの代表的な構成法として, simple field 法と big field 法がある. Simple field 法は中心写像の構成に $\mathbb{F}_q$ 以外の有限体は利用しない. 一方, big field 法は中心写像の構成に $\mathbb{F}_q$ の n 次拡大体 $\mathbb{F}_{q^n}$ を利用する. Simple field 法の代表として署名方式 UOV, big field 法の代表として署名方式 HFE_v^- について説明する.

4.2.2.1 署名方式 UOV

UOV [24] の中心写像は 2 次多項式ではあるが, 変数の一部に代入を行うことで, 逆写像計算を線形写像の逆写像計算に帰着させることができる. UOV に対して, 致命的な攻撃は今のところ報告されていない. 但し, 安全性を考慮すると, 署名長をメッセージ長の約 3 倍以上にする必要がある.

v, o を正の整数とし, $m = o$, $n = v+o$ とする. 2 次多項式からなる多変数多項式系 $G(\mathbf{x}) = (g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_m(\mathbf{x}))$ を次の形で与える.

$$g_k(\mathbf{x}) = \sum_{\substack{1 \le i \le v \\ v+1 \le j \le n}} \alpha_{i,j}^{(k)} x_i x_j + \sum_{1 \le i \le j \le v} \beta_{i,j}^{(k)} x_i x_j + \sum_{1 \le i \le n} \gamma_i^{(k)} x_i + \eta^{(k)} \quad (k = 1, 2, \ldots, m).$$

ここで, $\alpha_{i,j}^{(k)}, \beta_{i,j}^{(k)}, \gamma_i^{(k)}, \eta^{(k)} \in \mathbb{F}_q$ である. $G(\mathbf{x})$ は逆写像を効率的に計算することができる. 具体的には, 任意の $\mathbf{c} = (c_1, c_2, \ldots, c_m) \in \mathbb{F}_q^m$ に対し, $G^{-1}(\mathbf{c})$ (の一つ) が以下のように計算できる.

1. $b_1, b_2, \ldots, b_v \in \mathbb{F}_q$ をランダムにとる.

2. $g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_m(\mathbf{x})$ に $x_1 = b_1, x_2 = b_2, \ldots, x_v = b_v$ を代入し, $x_{v+1}, x_{v+2}, \ldots, x_n$ に関する 1 次の多項式系 $\bar{g}_1(x_{v+1}, x_{v+2}, \ldots, x_n), \ldots, \bar{g}_m(x_{v+1}, x_{v+2}, \ldots, x_n)$ を得る. 1 次方程式

$$\begin{cases} \bar{g}_1(x_{v+1}, x_{v+2}, \ldots, x_n) = c_1 \\ \qquad\qquad \vdots \\ \bar{g}_m(x_{v+1}, x_{v+2}, \ldots, x_n) = c_m \end{cases}$$

の解を Gauss の消去法などで計算し, それを $b_{v+1}, b_{v+2}, \ldots, b_n$ とおく. もし解がなければ **1** に戻る.

3. $(b_1, b_2, \ldots, b_n)$ を返す.

$\alpha_{i,j}^{(k)}, \beta_{i,j}^{(k)}, \gamma_i^{(k)}, \eta^{(k)} \in \mathbb{F}_q$ を動かしてできる $G(\mathbf{x})$ の集合を $\mathcal{C}_{\mathrm{UOV}}$ としたとき, $\mathcal{C}_{\mathrm{cent}} = \mathcal{C}_{\mathrm{UOV}}$ として構成される双極型システムの署名方式を UOV と呼ぶ. 但し, アフィン同型写像 T は UOV の安全性には貢献しないので, 通常は T を恒等写像で選ぶ. (UOV の多層化である Rainbow に対しては, 一般の T が必要である.)

鍵生成

1. $G(\mathbf{x}) \in \mathcal{C}_{\mathrm{UOV}}$ をランダムに選ぶ.
2. $\mathbb{F}_q^n$ 上のアフィン同型写像 S をランダムに選ぶ.
3. $F(\mathbf{x}) = G(\mathbf{x}) \circ S$ とする.

公開鍵は $F(\mathbf{x})$, 秘密鍵は $G(\mathbf{x}), S$ である.

署名生成

1. メッセージ $M \in \mathbb{F}_q^m$ に対し, 上で説明した方法で $G^{-1}(\mathbf{c})$ の要素の 1 つ $\mathbf{b} \in \mathbb{F}_q^n$ を求める.
2. $\sigma = S^{-1}(\mathbf{b})$.

検証

1. 署名 $\sigma \in \mathbb{F}_q^n$ に対し, $M' = F(\sigma)$ を計算する. $M = M'$ ならば署名を受理し, それ以外は不受理とする.

4.2.2.2 署名方式 HFE_v^-

HFE_v^- [30] は, 暗号方式 HFE [30] の安全性を強化しつつ署名方式に変形したものである. HFE は, Kipnis と Shamir によって解読された [26] が, HFE_v^- に対しては, 致命的な攻撃は今のところ報告されていない. まず, 暗号方式 HFE について簡単に述べる. HFE は松本-今井方式の拡張方式として提案されたものである. $\mathbb{F}_q$ の n 次拡大体 $K = \mathbb{F}_{q^n}$ をとり, $\mathbb{F}_q$-線形同型写像 $\phi : \mathbb{F}_q^n \to K$ を固定する. D を正の整数として, K 上の 1 変数多項式

$$\mathcal{G}(X) = \sum_{0 \le i \le j}^{q^i+q^j \le D} \alpha_{i,j} X^{q^i+q^j} + \sum_{0 \le i}^{q^i \le D} \beta_i X^{q^i} + \gamma \qquad (\alpha_{i,j}, \beta_i, \gamma \in K)$$

をとる. (HFE 多項式と呼ばれる.) このとき, 多変数多項式写像 $G : \mathbb{F}_q^n \to \mathbb{F}_q^n$ を $G = \phi^{-1} \circ \mathcal{G} \circ \phi$ と定めると, 対応する多変数多項式系 $G(\mathbf{x})$ の成分はすべて 2 次多項式となる. $\alpha_{i,j}, \beta_i, \gamma \in K$ を動かしてできる $G(\mathbf{x})$ の集合を $\mathcal{C}_{\mathrm{HFE}}$ としたとき, $\mathcal{C}_{\mathrm{cent}} = \mathcal{C}_{\mathrm{HFE}}$ として構成される双極型システムの暗号方式を HFE と呼ぶ. $G(\mathbf{x})$ による $\mathbf{c} \in \mathbb{F}_q^n$ の逆写像計算は以下のように行われる.

1. $B = \phi(\mathbf{c}) \in K$ を計算する.
2. $A = \mathcal{G}^{-1}(B)$ を Cantor-Zassenhaus アルゴリズムなどを用いて計算する.
3. $\phi^{-1}(A)$ を計算する.

2 の計算を効率的に実行するためには D をある程度小さくとる必要がある.

次に, HFE_v^- について説明する. 正の整数 a と v を固定する. まず, $\mathcal{G}(X)$ は次のように変更される.

$$\mathcal{G}(X) = \sum_{0 \le i \le j}^{q^i+q^j \le D} \alpha_{i,j} X^{q^i+q^j} + \sum_{0 \le i}^{q^i \le D} \beta_i(x_{n+1}, x_{n+2}, \ldots, x_{n+v}) X^{q^i} + \gamma(x_{n+1}, x_{n+2}, \ldots, x_{n+v}) \qquad (\alpha_{i,j} \in K). \quad (4.3)$$

ここで, $\beta_i(x_{n+1}, x_{n+2}, \ldots, x_{n+v}), \gamma(x_{n+1}, x_{n+2}, \ldots, x_{n+v})$ は共に $\mathbb{F}_q^v$ から $K(\simeq \mathbb{F}_q^n)$ への多項式写像であり, $\beta_i(x_{n+1}, x_{n+2}, \ldots, x_{n+v})$ は 1 次多項式写像, $\gamma(x_{n+1}, x_{n+2}, \ldots, x_{n+v})$ は 2 次多項式写像である. 多変数多項式

系 $G(\mathbf{x})$ は, 多変数多項式写像 $G = \phi^{-1} \circ \mathcal{G} \circ (\phi \times id_v) : \mathbb{F}_q^{n+v} \to \mathbb{F}_q^n$ により定める. $\alpha_{i,j} \in K$ と多項式写像 $\beta_i(x_{n+1}, x_{n+2}, \ldots, x_{n+v}), \gamma(x_{n+1}, x_{n+2}, \ldots, x_{n+v})$ を動かしてできる $G(\mathbf{x})$ の集合を $\mathcal{C}_{\mathrm{HFE}_v^-}$ と定める. 基本的には, これを $\mathcal{C}_{\mathrm{cent}} = \mathcal{C}_{\mathrm{HFE}_v^-}$ として構成される双極型システムを考えるのであるが, 双極型システムも若干変更する. S は $\mathbb{F}_q^{n+v}$ 上のアフィン同型写像のままでよいが, T は $\mathbb{F}_q^n$ から $\mathbb{F}_q^{n-a}$ への最大ランクのアフィン写像と変更する. 公開鍵は通常の双極型システムと同じように, $F(\mathbf{x}) = T \circ G(\mathbf{x}) \circ S$ と定める. よって, F は $\mathbb{F}_q^{n+v}$ から $\mathbb{F}_q^{n-a}$ への多変数多項式写像となる.

鍵生成

1. $G(\mathbf{x}) \in \mathcal{C}_{\mathrm{HFE}_v^-}$ をランダムに選ぶ.
2. $\mathbb{F}_q^{n+v}$ 上のアフィン同型写像 S と, 最大ランクのアフィン写像 $T : \mathbb{F}_q^n \to \mathbb{F}_q^{n-a}$ をランダムに選ぶ.
3. $F(\mathbf{x}) = T \circ G(\mathbf{x}) \circ S$ とする.

公開鍵は $F(\mathbf{x})$, 秘密鍵は $G(\mathbf{x}), S, T$ である.

署名生成

1. メッセージ $M \in \mathbb{F}_q^{n-a}$ に対し, $\mathbf{c} = T^{-1}(M) \in \mathbb{F}_q^n$ (の 1 つ) を計算する.
2. $B = \phi(\mathbf{c}) \in K$ を計算する.
3. $\mathbf{b} \in \mathbb{F}_q^v$ をランダムに選び, $A = \mathcal{G}^{-1}(B \| \mathbf{b})$ を Cantor-Zassenhaus アルゴリズムなどを用いて計算する. $\mathcal{G}^{-1}(B \| \mathbf{b})$ が存在しない場合は, **1** に戻る.
4. $\mathbf{e} = \phi^{-1}(A)$ を計算する.
5. $\sigma = S^{-1}(\mathbf{e})$ を計算する.

検証

1. 署名 $\sigma \in \mathbb{F}_q^{n+v}$ に対し, $M' = F(\sigma)$ を計算する. $M = M'$ ならば署名を受理し, それ以外は不受理とする.

HFE_v^- の公開鍵から得られる多変数多項式系は, 一般には半正則にはならず, 最小降下次数に関する以下の評価が知られている [13].

$$d_{\mathrm{FF}} \leq \begin{cases} \frac{(q-1)\cdot(r-1+a+v)}{2} + 2 & q: \text{偶数 かつ } r+a: \text{奇数}, \\ \frac{(q-1)\cdot(r+a+v)}{2} + 2 & \text{その他}. \end{cases}$$

ここで, $r = \lfloor \log_q(D-1) \rfloor + 1$ である. Gröbner 基底攻撃の計算量は (4.2) であるため, d_{FF} の値がなるべく大きくなるようパラメータを選択する必要がある.

4.3 具体的な暗号方式

多変数公開鍵暗号の具体的な方式として, Rainbow [9], Gui [8], MQDSS [7] の 3 つの署名方式について説明する. これらはいずれも NIST PQC 標準化プロジェクトに投稿されている. この他, NIST PQC 標準化プロジェクトに投稿されたものとして, HiMQ-3 [39], LUOV [3], GeMSS [5], DualModeMS [18] (いずれも署名方式) などがあるが, HiMQ-3, LUOV は Rainbow (あるいは UOV) に構造が近く, GeMSS, DualModeMS は Gui と同じ HFE_v^- をベースとした方式であるため, 代表して Rainbow, Gui, MQDSS の 3 つを説明することとした.

NIST PQC 標準化プロジェクト Round1 の資料を基に, 安全性パラメータも記述している. 安全性レベルが k ビットであるとは, k ビット鍵を持つブロック暗号の鍵探索と同等の安全性を持っていることを意味する.

表 4.1　本節で扱う主要な多変数多項式に基づく暗号技術

文献	暗号化	鍵交換	署名
Rainbow [12, 9]			○
Gui [33, 8]			○
MQDSS [6, 7]			○

4.3.1 Rainbow

4.3.1.1 Rainbow の概要

署名方式 Rainbow [12] は, 双極型システムを用いており, 署名方式 UOV [24] を多層化した構造を持っている. UOV を多層化することにより, UOV よりも短い署名長が実現できる.

正の整数 $t, v_1, o_1, o_2, \ldots, o_t$ に対し, $v_{i+1} = v_i + o_i$ により, $v_2, v_3, \ldots, v_{t+1}$ を帰納的に定める. また, $i = 1, 2, \ldots, t$ に対し, $S_i = \{1, 2, \ldots, v_i\}$, $O_i = \{v_i + 1, v_i + 2, \ldots, v_{i+1}\}$ とおく. S_i の個数は v_i で, O_i の個数は o_i である. 変数の個数を $n = v_{t+1}$, 式数を $m = n - v_1$ とする多変数多項式系 $G(\mathbf{x}) = (g_{v_1+1}(\mathbf{x}), g_{v_1+2}(\mathbf{x}), \ldots, g_n(\mathbf{x}))$ を次の形で与える.

$$g_k(x_1, \ldots, x_n) = \sum_{i \in O_h, j \in S_h} \alpha_{i,j}^{(k)} x_i x_j + \sum_{i,j \in S_h, i \leq j} \beta_{i,j}^{(k)} x_i x_j + \sum_{i \in S_{h+1}} \gamma_i^{(k)} x_i + \eta^{(k)} \quad (k = v_1+1, \ldots, n).$$

但し, h は k が属する層番号, すなわち, "$k \in O_h$" で定まる整数 $1 \leq h \leq t$ である. $\alpha_{i,j}^{(k)}, \beta_{i,j}^{(k)}, \gamma_i^{(k)}, \eta^{(k)} \in \mathbb{F}_q$ を動かしてできる $G(\mathbf{x})$ の集合を $\mathcal{C}_{\text{Rainbow}}$ と定め, これを Rainbow の中心写像のクラスとする. Rainbow に必要なパラメータは, 有限体の位数 q, および, $t, v_1, o_1, o_2, \ldots, o_t$ である. $\mathcal{H} : \{0,1\}^* \to \mathbb{F}_q^m$ を暗号学的ハッシュ関数とする.

鍵生成

1. $G(\mathbf{x}) \in \mathcal{C}_{\text{Rainbow}}$ をランダムに選ぶ.
2. $\mathbb{F}_q^n, \mathbb{F}_q^m$ 上のアフィン同型写像 S, T をランダムに選ぶ.
3. S^{-1}, T^{-1} を計算する.
4. $F(\mathbf{x}) = T \circ G(\mathbf{x}) \circ S$.

公開鍵は $F(\mathbf{x})$, 秘密鍵は $G(\mathbf{x}), S^{-1}, T^{-1}$ である. 次に, 署名生成である. メッセージを $M \in \{0,1\}^*$ とする.

署名生成

1. $\mathbf{h} \leftarrow \mathcal{H}(M)$.
2. $\mathbf{c} = (c_{v_1+1}, \ldots, c_n) \leftarrow T^{-1}(\mathbf{h})$.
3. $b_1, b_2, \ldots, b_{v_1} \in \mathbb{F}_q$ をランダムにとる.
4. $h = 1, 2, \ldots, t$ に対し, 以下を実行：

$g_{v_h+1}(\mathbf{x}), \ldots, g_{v_{h+1}}(\mathbf{x})$ に $x_1 = b_1$, $x_2 = b_2$, …, $x_{v_h} = b_{v_h}$ を代入し, $x_{v_h+1}, \ldots, x_{v_{h+1}}$ に関する 1 次の多項式系 $\bar{g}_{v_h+1}(x_{v_h+1}, \ldots, x_{v_{h+1}}), \ldots, \bar{g}_{v_{h+1}}(x_{v_h+1}, \ldots, x_{v_{h+1}})$ を得る. 1 次方程式

$$\begin{cases} \bar{g}_{v_h+1}(x_{v_h+1}, \ldots, x_{v_{h+1}}) = c_{v_h+1} \\ \vdots \\ \bar{g}_{v_{h+1}}(x_{v_h+1}, \ldots, x_{v_{h+1}}) = c_{v_{h+1}} \end{cases}$$

の解を Gauss の消去法などで計算し, それを $b_{v_h+1}, \ldots, b_{v_{h+1}}$ とおく. もし解がなければ **3** に戻る.

5. $\mathbf{b} \leftarrow (b_1, \ldots, b_n)$.

6. $\sigma \leftarrow S^{-1}(\mathbf{b})$.

σ が署名となる. 最後に検証である.

検証

1. $\mathbf{h} \leftarrow \mathcal{H}(M)$.

2. $\mathbf{h}' \leftarrow F(\sigma)$.

3. $\mathbf{h} == \mathbf{h}'$ を返す.

検証者は, $\mathbf{h} = \mathbf{h}'$ のとき, 署名を受理し, それ以外は不受理とする.

4.3.1.2 Rainbow のパラメータ選択

Rainbow 自体は, 層数 t を自由に設定できるが, NIST PQC 標準化プロジェクト Round1 の資料 [9] では, 2 層 ($t = 2$) の Rainbow が提案されている. 2 層の Rainbow の設計に必要なパラメータは, 有限体の位数 q, 層のサイズを決めるパラメータ v_1, o_1, o_2 である. [9] では, 以下のようにパラメータに関するデータが示されている.

(q, v_1, o_1, o_2)	安全性レベル	公開鍵長	秘密鍵長	署名長
$(16, 32, 32, 32)$	128 bits	148.5 kB	97.9 kB	512 bits
$(31, 36, 28, 28)$	128 bits	148.3 kB	103.7 kB	624 bits
$(256, 40, 24, 24)$	128 bits	187.7 kB	140.0 kB	832 bits
$(31, 64, 32, 48)$	192 bits	512.1 kB	371.4 kB	896 bits
$(256, 68, 36, 36)$	192 bits	703.9 kB	525.2 kB	1,248 bits
$(256, 92, 48, 48)$	256 bits	1,683.3 kB	1,244.4 kB	1,632 bits

[9] では, 公開鍵 $F(\mathbf{x})$ とメッセージ M から作られる MQ 問題に対する攻撃と Rainbow に対する EIP 問題に対する攻撃 (UOV 攻撃 [25], MinRank 攻撃 [21], HighRank 攻撃 [21], Rainbow-Band-Separation 攻撃 [14]) について解析され, 上記のパラメータが見積もられている.

4.3.2 Gui

4.3.2.1 Gui の概要

署名方式 Gui [33] は, 双極型システムを用いており, HFE_v^- を基本構造として構成されている. Gui の中心写像のクラスは, § 4.2.2.2 の $\mathcal{C}_{\mathrm{HFE}_v^-}$ でとる. Gui では, 有限体は $\mathbb{F}_2$ ($q = 2$) に固定されている. Gui に必要なパラメータは, 拡大次数 n, (4.3) に現れる D と v, m のサイズを決める a, および, 署名生成で中心写像の逆写像計算の回数を決める k である. $\mathcal{H} : \{0,1\}^* \to \mathbb{F}_q^{n-a}$ を暗号学的ハッシュ関数とする.

鍵生成

1. $G(\mathbf{x}) \in \mathcal{C}_{\mathrm{HFE}_v^-}$ をランダムに選ぶ.

2. $\mathbb{F}_q^{n+v}, \mathbb{F}_q^n$ 上のアフィン同型写像 S, T をランダムに選ぶ.

3. S^{-1}, T^{-1} を計算する.

4. $F(\mathbf{x}) = \mathrm{Proj}_{n-a} \circ T \circ G(\mathbf{x}) \circ S$. ($\mathrm{Proj}_{n-a} : \mathbb{F}_q^n \to \mathbb{F}_q^{n-a}$ は, 最初の $n-a$ 成分への射影を表す.)

公開鍵は $F(\mathbf{x})$, 秘密鍵は $G(\mathbf{x}), S^{-1}, T^{-1}$ である. 次に, 署名生成である. メッセージを $M \in \{0,1\}^*$ とする.

署名生成

1. $l \leftarrow \lceil k \cdot \log_2 q \cdot (n-a)/|\mathcal{H}| \rceil$ ($|\mathcal{H}|$ は, $\mathcal{H}$ の出力長を表す).
2. $\tilde{\mathbf{h}} \leftarrow \mathcal{H}(M) \| \mathcal{H}(\mathcal{H}(M)) \| \cdots \| \mathcal{H}^l(M)$ ($\mathcal{H}^l$ は, $\mathcal{H}$ の l 回合成を表す).
3. $S_0 \leftarrow \mathbf{0}^{n-a}$.
4. $i = 1, 2, \ldots, k$ に対し, 以下を実行：
  4-1 $(\tilde{\mathbf{h}}_{(i-1)\log_2 q \cdot (n-a)+1} \| \cdots, \| \tilde{\mathbf{h}}_{i \log_2 q \cdot (n-a)})$ を $\mathbb{F}_q^{n-a}$ の元に変換したものを $\mathbf{d}_i$ とおく.
  4-2 $F^{-1}(\mathbf{d}_i \oplus S_{i-1})$ の 1 つ $\mathbf{e} = (e_1, \ldots, e_{n+v})$ を § 4.2.2.2 に記載したアルゴリズムを使って求める.
  4-3 $S_i \leftarrow (e_1, \ldots, e_{n-a})$, $X_i \leftarrow (e_{n-a+1}, \ldots, e_{n+v})$.
5. $\sigma \leftarrow (S_k \| X_k \| \cdots \| X_1)$.

σ が署名となる. 最後に検証である.

検証

1. $l \leftarrow \lceil k \cdot \log_2 q \cdot (n-a)/|\mathcal{H}| \rceil$.
2. $\tilde{\mathbf{h}} \leftarrow \mathcal{H}(M) \| \mathcal{H}(\mathcal{H}(M)) \| \cdots \| \mathcal{H}^l(M)$.
3. $i = 1, 2, \ldots, k$ に対し, 以下を実行：
  3-1 $(\tilde{\mathbf{h}}_{(i-1)\log_2 q \cdot (n-a)+1} \| \cdots, \| \tilde{\mathbf{h}}_{i \log_2 q \cdot (n-a)})$ を $\mathbb{F}_q^{n-a}$ の元に変換したものを $\mathbf{d}_i$ とおく.
4. $i = k-1, \ldots, 0$ に対し, 以下を実行：
  4-1 $S_i \leftarrow F(S_{i+1} \| X_{i+1}) \oplus \mathbf{d}_{i+1}$.
5. $S_0 == \mathbf{0}^{n-a}$ を返す.

検証者は, $S_0 = \mathbf{0}^{n-a}$ のとき, 署名を受理し, それ以外は不受理とする.

4.3.2.2 Gui のパラメータ選択

NIST PQC 標準化プロジェクト Round1 の資料 [8] では, 以下のようにパラメータに関する 3 つのデータが示されている.

名称	(n, D, a, v, k)	安全性レベル	公開鍵長	秘密鍵長	署名長
Gui-184	$(184, 33, 16, 16, 2)$	128 bits	416.3 kB	19.1 kB	360 bits
Gui-312	$(312, 129, 24, 20, 2)$	192 bits	1,955.1 kB	59.3 kB	504 bits
Gui-448	$(448, 513, 32, 28, 2)$	256 bits	5,789.2 kB	155.9 kB	664 bits

[8] では, 公開鍵 $F(\mathbf{x})$ とメッセージ M から作られる MQ 問題に対する攻撃と識別攻撃 [32], Gui に対する EIP 問題に対する攻撃 (Kipnis-Shamir 攻撃 [26]) について解析され, 上記のパラメータが見積もられている.

4.3.3 MQDSS

4.3.3.1 MQDSS の概要

MQDSS [6] は, ソニー株式会社の作本, 白井, 樋渡により提案された 5-pass の認証方式 [35] に対し, Fiat-Shamir 変換を適用したものである. 特に, 双極型システムでは構成されておらず, 攻撃対象となる MQ 問題にトラップドアの構造は入っていない. すなわち, MQ 問題は一般的となり, トラップドアによる脆弱性はない. 実際, MQ 問題の average

case での解読困難性を仮定すると, ランダムオラクルモデルの下, MQDSS は EUF-CMA 安全であることが示されている [6]. この 5-pass の認証方式 (および, MQDSS) を記述するために, コミットメント Com を 1 つ用意する. (コミットメントについては, [22] を参照のこと.) 5-pass の認証方式の鍵生成は次のように行う.

1. $\mathbb{F}_q$ 上の多項式系 $F(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_m(\mathbf{x}))$ で, すべての $f_i(\mathbf{x})$ が 2 次多変数多項式で, かつ定数項を持たないものをランダムに構成する. (写像としては, $F : \mathbb{F}_q^n \to \mathbb{F}_q^m$ となる.)

2. $\mathbf{s} \in \mathbb{F}_q^n$ をランダムに選ぶ.

3. $\mathbf{v} = F(\mathbf{s}) \in \mathbb{F}_q^m$ を計算する.

このとき, $\mathbf{s}$ を秘密鍵とし, $(F(\mathbf{x}), \mathbf{v})$ を公開鍵とする. また, 多変数多項式写像 $G : \mathbb{F}_q^{2n} \to \mathbb{F}_q^m$ を

$$G(\mathbf{x}, \mathbf{y}) = F(\mathbf{x} + \mathbf{y}) ? F(\mathbf{x}) ? F(\mathbf{y}) \tag{4.4}$$

で定義する. $G(\mathbf{x}, \mathbf{y})$ は双一次写像であり, 公開鍵 $F(\mathbf{x})$ から誰でも作ることができる. 認証は次のように実行される.

証明者 $((F(\mathbf{x}), \mathbf{v}), \mathbf{s})$		検証者 $(F(\mathbf{x}), \mathbf{v})$
$\mathbf{r}_0, \mathbf{t}_0 \in \mathbb{F}_q^n$, $\mathbf{e}_0 \in \mathbb{F}_q^m$ をランダムに選ぶ.		
$\mathbf{r}_1 \leftarrow \mathbf{s} - \mathbf{r}_0$		
$c_0 \leftarrow \mathrm{Com}(\mathbf{r}_0, \mathbf{t}_0, \mathbf{e}_0)$		
$c_1 \leftarrow \mathrm{Com}(\mathbf{r}_1, G(\mathbf{t}_0, \mathbf{r}_1) + \mathbf{e}_0)$	$\xrightarrow{(c_0, c_1)}$	
	$\xleftarrow{\alpha}$	$\alpha \in \mathbb{F}_q$ をランダムに選ぶ.
$\mathbf{t}_1 \leftarrow \alpha \mathbf{r}_0 - \mathbf{t}_0$		
$\mathbf{e}_1 \leftarrow \alpha F(\mathbf{r}_0) - \mathbf{e}_0$	$\xrightarrow{(\mathbf{t}_1, \mathbf{e}_1)}$	
	$\xleftarrow{\mathrm{Ch}}$	$\mathrm{Ch} \in \{0, 1\}$ をランダムに選ぶ.
$\mathrm{Ch} = 0$ ならば, $\mathrm{Rsp} \leftarrow \mathbf{r}_0$		
$\mathrm{Ch} = 1$ ならば, $\mathrm{Rsp} \leftarrow \mathbf{r}_1$	$\xrightarrow{\mathrm{Rsp}}$	
		$\mathrm{Ch} = 0$ ならば, $\mathrm{Rsp} = \mathbf{r}_0$ と見なし, 次をチェック.
		$c_0 \stackrel{?}{=} \mathrm{Com}(\mathbf{r}_0, \alpha \mathbf{r}_0 - \mathbf{t}_1, \alpha F(\mathbf{r}_0) - \mathbf{e}_1)$
		$\mathrm{Ch} = 1$ ならば, $\mathrm{Rsp} = \mathbf{r}_1$ と見なし, 次をチェック.
		$c_1 \stackrel{?}{=} \mathrm{Com}(\mathbf{r}_1, \alpha(\mathbf{v} - F(\mathbf{r}_1)) - G(\mathbf{t}_1, \mathbf{r}_1) - \mathbf{e}_1)$

“$\stackrel{?}{=}$” の等式が成り立つならば 1 を, そうでなければ 0 を検証者は出力する. この認証において, 不正証明者が到達できる (最大の) 成功確率は $\frac{1}{2} + \frac{1}{2q}$ となる.

MQDSS の鍵生成, 署名生成, 検証アルゴリズムを記述する. MQDSS の設計に必要なパラメータは, 有限体の位数 q, 変数の個数 (および式数) n, ラウンド数 r の 3 つである. 暗号学的ハッシュ関数を 3 つ用意する : $\mathcal{H} : \{0,1\}^* \to \{0,1\}^k$ (k は安全性パラメータ), $H_1 : \{0,1\}^* \to \mathbb{F}_q^r$, $H_2 : \{0,1\}^* \to \{0,1\}^r$.

鍵生成は 5-pass の認証方式のそれと基本的に同じであるが, $m = n$ とする点に注意する.

鍵生成

1. $\mathbb{F}_q$ 上の多項式系 $F(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_n(\mathbf{x}))$ で, すべての $f_i(\mathbf{x})$ が 2 次多変数多項式で, かつ定数項を持たないものをランダムに構成する. (写像としては, $F : \mathbb{F}_q^n \to \mathbb{F}_q^n$ となる.)

2. $\mathbf{s} \in \mathbb{F}_q^n$ をランダムに選ぶ.

3. $\mathbf{v} = F(\mathbf{s}) \in \mathbb{F}_q^n$ を計算する.

$\mathsf{pk} = (F(\mathbf{x}), \mathbf{v})$, $\mathsf{sk} = \mathbf{s}$ がそれぞれ公開鍵, 秘密鍵となる. 次に, 署名生成である. メッセージを $M \in \{0,1\}^*$ とする.

署名生成

1. $R \leftarrow \mathcal{H}(\mathsf{sk}\|M)$.
2. $D \leftarrow \mathcal{H}(\mathsf{pk}\|R\|M)$.
3. $\mathbf{r}_0^{(1)},\ldots,\mathbf{r}_0^{(r)},\mathbf{t}_0^{(1)},\ldots,\mathbf{t}_0^{(r)},\mathbf{e}_0^{(1)},\ldots,\mathbf{e}_0^{(r)} \in \mathbb{F}_q^n$ をランダムに選ぶ.
4. $j=1,2,\ldots,r$ に対し, 以下を実行：
  4-1 $\mathbf{r}_1^{(j)} \leftarrow \mathbf{s}-\mathbf{r}_0^{(j)}$.
  4-2 $c_0^{(j)} \leftarrow \mathrm{Com}(\mathbf{r}_0^{(j)},\mathbf{t}_0^{(j)},\mathbf{e}_0^{(j)})$.
  4-3 $c_1^{(j)} \leftarrow \mathrm{Com}(\mathbf{r}_1^{(j)},G(\mathbf{t}_0^{(j)},\mathbf{r}_1^{(j)})+\mathbf{e}_0^{(j)})$.　($G$ は (4.4) で定義される双一次写像)
  4-4 $\mathsf{com}^{(j)} := (c_0^{(j)},c_1^{(j)})$.
5. $\sigma_0 \leftarrow \mathcal{H}(\mathsf{com}^{(1)}\|\mathsf{com}^{(2)}\|\cdots\|\mathsf{com}^{(r)})$.
6. $\mathsf{ch}_1=(\alpha^{(1)},\ldots,\alpha^{(r)}) \leftarrow H_1(D,\sigma_0) \in \mathbb{F}_q^r$.
7. $j=1,2,\ldots,r$ に対し, 以下を実行：
  7-1 $\mathbf{t}_1^{(j)} \leftarrow \alpha^{(j)}\mathbf{r}_0^{(j)}-\mathbf{t}_0^{(j)},\ \mathbf{e}_1^{(j)} \leftarrow \alpha^{(j)}F(\mathbf{r}_0^{(j)})-\mathbf{e}_0^{(j)}$.
  7-2 $\mathsf{resp}_1^{(j)} := (\mathbf{t}_1^{(j)},\mathbf{e}_1^{(j)})$.
8. $\sigma_1 \leftarrow (\mathsf{resp}_1^{(1)}\|\mathsf{resp}_1^{(2)}\|\cdots\|\mathsf{resp}_1^{(r)})$.
9. $\mathsf{ch}_2=(b^{(1)},\ldots,b^{(r)}) \leftarrow H_2(D,\sigma_0,\mathsf{ch}_1,\sigma_1) \in \{0,1\}^r$.
10. $j=1,2,\ldots,r$ に対し, 以下を実行：
  10-1 $\mathsf{resp}_2^{(j)} \leftarrow \mathbf{r}_{b^{(j)}}^{(j)}$.
11. $\sigma_2 \leftarrow (\mathsf{resp}_2^{(1)}\|\mathsf{resp}_2^{(2)}\|\cdots\|\mathsf{resp}_2^{(r)}\|c_{1-b^{(1)}}^{(1)}\|c_{1-b^{(2)}}^{(2)}\|\cdots\|c_{1-b^{(r)}}^{(r)})$.

$\sigma=(R,\sigma_0,\sigma_1,\sigma_2)$ が署名となる. 最後に, 検証である.

検証

1. $D \leftarrow \mathcal{H}(\mathsf{pk}\|R\|M)$.
2. $\mathsf{ch}_1=(\alpha^{(1)},\ldots,\alpha^{(r)}) \leftarrow H_1(D,\sigma_0) \in \mathbb{F}_q^r$.
3. $\mathsf{ch}_2=(b^{(1)},\ldots,b^{(r)}) \leftarrow H_2(D,\sigma_0,\mathsf{ch}_1,\sigma_1) \in \{0,1\}^r$.
4. σ_1 を $\sigma_1=(\mathsf{resp}_1^{(1)}\|\mathsf{resp}_1^{(2)}\|\cdots\|\mathsf{resp}_1^{(r)})$ なる連結と見なす.
5. σ_2 を $\sigma_2=(\mathsf{resp}_2^{(1)}\|\mathsf{resp}_2^{(2)}\|\cdots\|\mathsf{resp}_2^{(r)}\|c_{1-b^{(1)}}^{(1)}\|c_{1-b^{(2)}}^{(2)}\|\cdots\|c_{1-b^{(r)}}^{(r)})$ なる連結と見なす.
6. $j=1,2,\ldots,r$ に対し, 以下を実行：
  6-1 $\mathsf{resp}_1^{(j)}$ を $\mathsf{resp}_1^{(j)}=(\mathbf{t}_1^{(j)},\mathbf{e}_1^{(j)})$ と見なす.
  6-2 $b^{(j)}=0$ ならば, $\mathbf{r}_0^{(j)} := \mathsf{resp}_2^{(j)},\ c_0^{(j)} \leftarrow \mathrm{Com}(\mathbf{r}_0^{(j)},\alpha^{(j)}\mathbf{r}_0^{(j)}-\mathbf{t}_1^{(j)},\alpha^{(j)}F(\mathbf{r}_0^{(j)})-\mathbf{e}_1^{(j)})$.
    そうでなければ, $\mathbf{r}_1^{(j)} := \mathsf{resp}_2^{(j)},\ c_1^{(j)} \leftarrow \mathrm{Com}(\mathbf{r}_1^{(j)},\alpha^{(j)}(\mathbf{v}-F(\mathbf{r}_1^{(j)}))-G(\mathbf{t}_1^{(j)},\mathbf{r}_1^{(j)})-\mathbf{e}_1^{(j)})$.
  6-3 $\mathsf{com}^{(j)} := (c_0^{(j)},c_1^{(j)})$.
7. $\sigma_0' \leftarrow \mathcal{H}(\mathsf{com}^{(1)}\|\mathsf{com}^{(2)}\|\cdots\|\mathsf{com}^{(r)})$.
8. $\sigma_0' == \sigma_0$ を返す.

検証者は, $\sigma_0'=\sigma_0$ のとき, 署名を受理し, それ以外は不受理とする.

4.3.3.2　MQDSS のパラメータ選択

NIST PQC 標準化プロジェクト Round1 の資料 [7] では, 以下のようにパラメータに関する 2 つのデータが示されている.

名称	(q, n, r)	安全性レベル $(= k/2)$	公開鍵長	秘密鍵長	署名長
MQDSS-31-48	$(31, 48, 269)$	128 bits	62 B	32 B	32882 B
MQDSS-31-64	$(31, 64, 403)$	192 bits	88 B	48 B	67800 B

このパラメータは, 古典計算機による攻撃であるハイブリッド攻撃と, ハイブリッド攻撃の一部を Grover のアルゴリズムで置き換えた量子攻撃の計算量から見積もられている. なお, [7] では, 公開鍵 $F(\mathbf{x})$ は必要となる度に可変長出力関数 (XOF) を用いて, k ビットの文字列から係数を構成するという手法を用いているため, 公開鍵長を (単純な $F(\mathbf{x})$ の係数集合より) 小さくすることができている.

4.4 まとめ

1988 年に松本-今井方式が提案されて以来, 様々な多変数公開鍵暗号の暗号方式, 署名方式が提案されてきた. 暗号方式は, 松本-今井方式をはじめ, 提案された多くの方式が破られてしまった. 現在致命的な攻撃を免れているものとしては, Simple Matrix 方式 (2013), EFC (の変種) (2016), HFERP (2018), EFLASH (2018) などがあるが, いずれも提案されてまだ間もなく, NIST PQC 標準化プロジェクトには投稿されなかった. 署名方式では, 1996 年に HFE_v^- が, 1999 年に UOV が提案された. また, 2005 年には UOV を多層化した署名方式 Rainbow が提案された.

NIST PQC 標準化プロジェクトには, 多変数公開鍵暗号の署名方式が多く投稿された. この中では, UOV 系が 3 件, HFE_v^- 系が 3 件含まれていた. この報告書では具体的な方式として, NIST PQC 標準化プロジェクトにも投稿されている Rainbow (UOV 系代表), Gui (HFE_v^- 系代表), MQDSS の 3 つの署名方式を取り上げ, 概要を説明した. 双極型システムで構成されている Rainbow と Gui は, 多変数多項式系を公開鍵, 秘密鍵とするため鍵長が膨大になるが, 署名長は比較的小さいという特徴を持つ. また, 検証は多項式への代入評価のみで計算されるため, 効率的で安定したパフォーマンスが得られる. 一方, 対話型認証と Fiat-Shamir 変換で構成されている MQDSS は, 鍵長はそれほど大きくはないが, 署名長が比較的大きいという特徴を持つ.

一般に, 多変数公開鍵暗号では, MP 問題に対するグレブナー基底攻撃の計算量の観点から, 半正則と呼ばれる条件を満たす多変数多項式系を利用することが望ましい. 署名方式 Rainbow は, 半正則な多変数多項式系を利用できている. 一方で, 暗号方式に対しては, 半正則な多変数多項式系を用いた構成が難しいのが現状である.

第 4 章の参考文献

[1] M. Bardet, J.-C. Faugère, B. Salvy, B.-Y. Yang. Asymptotic expansion of the degree of regularity for semi-regular systems of equations. In Effective Methods in Algebraic Geometry (MEGA), pp. 71–74, 2004.

[2] L. Bettale, J.-C. Faugère, and L. Perret. Hybrid approach for solving multivariate systems over finite fields. J. Math. Cryptol. vol. 3, Issue 3, pp. 177-197, 2009.

[3] W. Beullens, B. Preneel, A. Szepieniec, F. Vercauteren. LUOV. NIST PQC 標準化プロジェクト Round 1 候補資料. https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions

[4] R. Cartor, D, Smith-Tone. EFLASH: A New Multivariate Encryption Scheme. SAC'18, to be appeared in Springer LNCS, 2018.

[5] A. Casanova, J.-C. Faugère, G. Macario-Rat, J. Patarin, L. Perret, J. Ryckeghem. GeMSS. NIST PQC 標準化プロジェクト Round 1 候補資料. https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions

[6] M.-S. Chen, A. Hülsing, J. Rijneveld, S. Samardjiska, P. Schwabe. From 5-pass MQ-based identification to MQ-based signatures. ASIACRYPT'16, Springer LNCS vol. 10032, pp. 135–165, 2016.

[7] M.-S. Chen, A. Hülsing, J. Rijneveld, S. Samardjiska, P. Schwabe. MQDSS. NIST PQC 標準化プロジェクト Round 1 候補資料. https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions

[8] J. Ding, M.-S. Chen, A. Petzoldt, D. S. Schmidt, B.-Y. Yang. Gui. NIST PQC 標準化プロジェクト Round 1 候補資料. https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions

[9] J. Ding, M.-S. Chen, A. Petzoldt, D. S. Schmidt, B.-Y. Yang. Rainbow. NIST PQC 標準化プロジェクト Round 1 候補資料. https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions

[10] J. Ding, J. E. Gower, D. S. Schmidt. Multivariate Public Key Cryptosystems. Advances in Information Security 25, Springer, 2006.

[11] J. Ding and T. J. Hodges. Inverting HFE Systems Is Quasi-Polynomial for All Fields. CRYPTO'11, Springer LNCS vol. 6841, pp. 724–742, 2011.

[12] J. Ding and D. Schmidt. Rainbow, a New Multivariable Polynomial Signature Scheme. ACNS'05 Springer LNCS vol. 3531, pp. 164–175, 2005.

[13] J. Ding, B.-Y. Yang. Degree of Regularity for HFEv and HFEv-. PQCrypto'13, Springer LNCS vol. 7932, pp. 52–66, 2013.

[14] J. Ding, B.-Y. Yang, C.-H. O. Chen, M.-S. Chen, C. M. Cheng. New Differential-Algebraic Attacks and Reparametrization of Rainbow. ACNS'08, Springer LNCS vol. 5037, pp. 242–257, 2008.

[15] V. Dubois and N. Gama. The Degree of Regularity of HFE Systems. ASIACRYPT'10, Springer LNCS vol.

6477, pp. 557–576, 2010.

[16] J.-C. Faugère. A new efficient algorithm for computing Gröbner bases (F4). J. Pure Appl. Algebra, vol. 139 pp. 61–88, 1999.

[17] J.-C. Faugère. A new efficient algorithm for computing Gröbner bases without reduction to zero (F5). Proceedings of ISSAC'02, pp. 75–83. ACM Press, 2002.

[18] J.-C. Faugère, L. Perret, J. Ryckeghem. DualModeMS. NIST PQC 標準化プロジェクト Round 1 候補資料. https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions

[19] R. Fröberg. An inequarity for Hilbert series of graded algebras. Math. Scand., vol. 56, no. 2, pp. 117–144, 1985.

[20] M. R. Garay and D. S. Johnson. A Guide to the Theory of NP-Completeness. In Computers and Intractability, W.H.Freeman, 1979.

[21] L. Goubin and N.T. Courtois. Cryptanalysis of the TTM Cryptsystem. ASIACRYPT'00, Springer LNCS vol. 1976, pp. 44–57, 2000.

[22] S. Halevi and S. Micali. Practical and Provably-Secure Commitment Schemes from Collision-Free Hashing. CRYPTO, Springer LNCS vol. 1109, pp. 201–215, 1996.

[23] Y. Ikematsu, R. Perlner, D. Smith-Tone, T. Takagi, J. Vates. HFERP - A New Multivariate Encryption Scheme. PQCrypto'18, Springer LNCS vol. 10786, pp. 396- 416, 2018.

[24] A. Kipnis, L. Patarin, L. Goubin. Unbalanced Oil and Vinegar Schemes. EUROCRYPT'99, Springer LNCS vol. 1592, pp. 206–222, 1999.

[25] A. Kipnis, and A. Shamir. Cryptanalysis of the Oil and Vinegar Signature Scheme. CRYPTO'98, Springer LNCS vol. 1462, pp. 257–266, 1998.

[26] A. Kipnis, and A. Shamir. Cryptanalysis of the HFE Public Key Cryptosystem by Relinearization. CRYPTO'99. Springer LNCS, vol. 1666, pp. 19–33 1999

[27] T. Matsumoto and H. Imai. Public Quadratic Polynomial-tuples for Efficient Signature Verification and Message Encryption. EUROCRYPT'88, Springer LNCS vol. 330, pp. 419–453, 1988.

[28] H. Ong, C.P. Schnorr and A. Shamir. An efficient signature scheme based on quadratic equations. Proc. 16th ACM Symp. Theory Comp., pp. 208–216, 1984.

[29] J. Patarin. Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt' 88. CRYPTO'95, Springer LNCS vol. 963, pp. 248–261, 1995.

[30] J. Patarin. Hidden Fields Equations (HFE) and Isomorphisms of Polynomials (IP): Two New Families of Asymmetric Algorithms. EUROCRYPTO'96, Springer LNCS vol. 1070, pp. 33–48, 1996.

[31] J. Patarin. The Oil and Vineger Signature Scheme. Dagstuhl Workshop on Cryptography, 1997.

[32] R. Perlner, A. Petzoldt, D. Smith-Tone. Improved Cryptanalysis of HFEv- via Projection. PQCrypto'18, Springer LNCS vol. 10786, pp. 375–395, 2018.

[33] A. Petzoldt, M.-S. Chen, B.-Y. Yang, C. Tao, J. Ding. Design principles for HFEv-based multivariate signature schemes. ASIACRYPT'15, Springer LNCS vol. 9742, pp. 311–334, 2015.

[34] J.M. Pollard and C.P. Schnorr. An efficient solution of the congruence $x^2 + ky^2 \equiv m \pmod{n}$. IEEE Trans. Inf. Theory, IT-33, pp. 702–709, 1987.

[35] K. Sakumoto, T. Shirai, H. Hiwatari. public-key identification schemes based on multivariate quadratic polynomials. CRYPTO'11, Springer LNCS vol. 6841, pp. 706–723, 2011.

[36] K. Sakumoto, T. Shirai, H. Hiwatari. On Provable Security of UOV and HFE Signature Schemes against Chosen-Message Attack. PQCrypto'11, Springer LNCS vol. 7071, pp. 68–82, 2011.

[37] T. Satoh and K. Araki. On construction of signature scheme over a certain noncommutative ring. IEICE Trans. Fundamentals, E80-A, pp. 702–709, 1997.

[38] A. Shamir. Efficient signature schemes based on birational permutations. CRYPTO'93, Springer LNCS vol. 773, pp. 1–12, 1994.

[39] K.-A. Shim. HiMQ3. NIST PQC 標準化プロジェクト Round 1 候補資料. https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions

[40] A. Szepieniec, J. Ding, B. Preneel. Extension Field Cancellation: A New Central Trapdoor for Multivariate Quadratic Systems. PQCrypto'16, Springer LNCS vol. 9606, pp. 182–196, 2016.

[41] C. Tao, A. Diene, S. Tang, J. Ding. Simple Matrix Scheme for Encryption. PQCrypto'13, Springer LNCS vol. 7932, pp. 231–242, 2013.

[42] B.-Y. Yang and J.-M. Chen. All in the XL Family: Theory and Practice. ICISC 2004, Springer LNCS vol. 3506, pp. 67–86, 2005.

第 5 章

同種写像に基づく暗号技術

本章では同種写像に基づく暗号技術についてまとめる. 同種写像に基づく暗号技術の安全性は, 種々の同種写像問題を解く計算の困難性に依存している.

5.1 節では, 安全性の根拠となる問題として, 同種写像問題の一般形, SIDH (Supersingular Isogeny Diffie–Hellman) 鍵共有の安全性を根拠づける問題, CSIDH (Commutative Supersingular Isogeny Diffie–Hellman) 鍵共有の安全性を根拠づける問題の順に, その概要を記述していく. そして, その最終節 (5.1.4 節) で, 3 つの基本同種写像問題の漸近的解読時間を使用するアルゴリズムと共に比較する. 5.2 節では, 代表的な暗号方式として, SIDH 鍵共有, CSIDH 鍵共有を取り上げる. 5.3 節では, SIDH 鍵共有に基づいた具体的な同種写像に基づく暗号方式として, NIST 標準化に提案された SIKE (Supersingular Isogeny Key Encapsulation) を, まず解説する. そして次節で同種写像に基づく認証付き鍵共有・グループ鍵共有に触れたあと, 同種写像に基づく署名方式についての現状, 特に CSIDH ベースの SeaSign 署名方式を概説する.

本章では, 超特異楕円曲線を用いた暗号技術しか扱わないが, 通常楕円曲線に基づく CRS (Couveignes–Rostovtsev–Stolbunov) 鍵共有法を改良した De Feo ら [20] の方式, それ自体は実用的な性能にはまだ遠いが, しかし, そこで得られた知見は, CSIDH 鍵共有を構成していくのに大きく寄与したことが知られている.

同種写像の数学的詳細については, De Feo の概説記事 [17] や Washington の楕円曲線の教科書 [52] を参照のこと. また, Galbraith–Vercauteren による同種写像関連問題のサーベイ [30] も参照する.

■記法 $x \leftarrow_R X$ は, x を集合 X から一様ランダムにサンプリングすることを表す. 以下では, 有限体上に定義された楕円曲線のみを扱い, 同種写像暗号では, 多くの場合, モンゴメリ型の楕円曲線定義式 $E_{a,b} : by^2 = x^3 + ax^2 + x$ が用いられる. 標数 p の有限体 $\mathbb{F}$ 上定義された楕円曲線 E に対し, O_E は E の無限遠点であり, $\mathbb{F}$ の拡大体 $\mathbb{K}$ に対して, $\mathbb{K}$-有理点群は $E(\mathbb{K}) = \{(x, y) \in \mathbb{K}^2 \,|\, (x, y) \text{ は } E \text{ の定義式を満たす }\} \cup \{O_E\}$ で与えられる. また, E の r-ねじれ群は $E[r] = \{P \in E(\overline{\mathbb{F}}_p) \,|\, rP = O_E\}$ で与えられる. また, 有限体を指定せず, 単に $P \in E$ と書いたときは, $P \in E(\overline{\mathbb{F}}_p)$ を意味する.

5.1 同種写像に基づく暗号技術の安全性の根拠となる問題

同種写像問題の一般形, SIDH 鍵共有の安全性の根拠となる問題, CSIDH 鍵共有の安全性の根拠となる問題, それぞれの定義と, 現在までに知られているそれら問題に対する解析状況について概説する.

5.1.1 同種写像問題の一般形

同種写像とは, 2 つの楕円曲線 E, E' の間の写像 ϕ であり, E の座標 (x, y) の有理式で与えられると共に, 楕円曲線の加法構造に関する準同型性, 即ち $\phi(P+Q) = \phi(P) + \phi(Q)$, を有する写像である. (その正確な定義は, 前掲の各文献を参照のこと.) また, E, E' の間に, 同種写像 ϕ が存在する時に, E と E' は同種であるという.

同種写像 ϕ は, その核 $C = \ker(\phi)$ によって決まるので, ϕ の定義域曲線 E に対して ϕ の値域となる楕円曲線を E/C と書き表す, すなわち, $\phi : E \to E/C$. 核 $C = \ker(\phi)$ の位数がセキュリティパラメータ λ の多項式サイズであれば, $C = \ker(\phi)$ となる ϕ を効率的に計算するアルゴリズムが Vélu によって与えられている [51]. (モンゴメリ型楕円曲線に対する Vélu の公式に関しては, [44] を参照のこと.) その効率的な同種写像基本演算の合成が同種写像暗号での公開鍵生成, 暗号化, そして復号を与える. そして, その合成における基本演算の組み合わせ方法が, 秘密鍵情報を与える.

つまり, 同種な楕円曲線の間の同種写像を計算することを要求する次の同種写像問題が, 具体的な暗号方式の安全性を根拠づける次節以降の問題の共通テンプレートを与える.

定義 5.1 (一般形同種写像問題 [30]) *2 つの同種な楕円曲線 E, E' に対して, 同種写像 ϕ を計算せよ. (ϕ のコンパクトな表現を与えよ.)*

ここで, 「ϕ のコンパクトな表現」とは, 例えば, $\deg(\phi)$ が小素数 ℓ_i によって $\prod \ell_i^{e_i}$ となっている場合には, この分解に沿って ϕ を分解した各 ℓ_i 次同種写像の像に現れる値域楕円曲線 (又は j 不変量) の列挙で与えられる. また, SIDH 鍵共有の設定では, 核の生成点が, 同種写像のコンパクト表現を与える. これまで, 同種写像暗号では, 小素数の積に分解する次数をもつ同種写像しか考えない.

定義 5.1 において, ϕ の次数が多項式サイズであれば, 上記問題は簡単に解けるので, ϕ の次数は通常は指数サイズのものを考える. また, Galbraith ら [30] は, j 不変量を使って, 上記問題を定式化しているが, CSIDH 鍵共有では, $\mathbb{F}_p$-有理な楕円曲線のみを対象とするので, $\overline{\mathbb{F}}_p$-同型であるが $\mathbb{F}_p$-同型でないツイスト曲線を判別して扱う必要が生じるため, 上ではあえて, より素朴な形を採用して, 2 つの同種な楕円曲線 E, E' を使って同種写像問題を提示した.

同種写像問題の初期の考察には, 自己準同型環計算を扱った Kohel の博士論文 [36] や Galbraith による同種写像問題に関する研究 [25] 及び Couveignes と Rostovtsev–Stolbunov による初期の暗号応用への提案 [13, 45] がある. その後, Charles らによる同種写像に基づいたハッシュ関数の提案 [8] は, 同種写像一方向性関数を一方向性の観点からだけでなく, 衝突困難性の観点からも見直すことになり, 初期の同種写像暗号の研究では重要な役割を果たした. 特に, 同種写像グラフがエクスパンダーグラフであることに着目して暗号に応用した意義は大きい.

■超特異同種写像問題と通常同種写像問題 標数 p の有限体上の楕円曲線 E の p-ねじれ群 $E[p]$ が, $E[p] = \{O_E\}$ の時, E を超特異楕円曲線といい, そうでない時, E を通常楕円曲線という. 超特異楕円曲線の j 不変量は, $\mathbb{F}_{p^2}$ の要素である. つまり, 超特異 j 不変量の個数は, 有限個であり, 具体的に $p/12 + \epsilon$(但し $\epsilon = 0, 1, 2$) で与えられる. 超特異, 通常という楕円曲線の性質は, 同種写像によって保存されるため, 同種写像問題も, この 2 つの性質によって, 超特異同種写像問題と通常同種写像問題という 2 つの問題に分類される.

5.1.2 SIDH 鍵共有の安全性の根拠となる問題

超特異楕円曲線間の同種写像問題の困難性に基づく鍵共有法として, SIDH 鍵共有 (5.2.1 節参照) が知られているが, その安全性の根拠となる計算問題を [19, 30, 26, 21] に従ってまとめる.

一般の超特異同種写像問題と異なるのは, 標数 p が特殊な形をしており, それに従って問題内で扱う同種写像の次数

が決まることと, 問題の入力に補助点が加わっていることである. (実際に, 超特異同種写像問題と SIDH 同種写像問題の解読計算量が異なることが 5.1.4 節に記載される.)

■SIDH 鍵共有の公開パラメータ　SIDH 鍵共有で, 公開パラメータは $pp_{\mathtt{sidh}} = (\ell_{\mathtt{A}}, \ell_{\mathtt{B}}, e_{\mathtt{A}}, e_{\mathtt{B}}, f; E, P_{\mathtt{A}}, Q_{\mathtt{A}}, P_{\mathtt{B}}, Q_{\mathtt{B}})$ で与えられる. ここで, $p+1 = f \cdot \ell_{\mathtt{A}}^{e_{\mathtt{A}}} \ell_{\mathtt{B}}^{e_{\mathtt{B}}}$ で, p は素数, $\ell_{\mathtt{A}}, \ell_{\mathtt{B}}$ は 2 つの小素数である (f は小さい正整数で, 多くの場合 $f = 1$). 例えば, $\ell_{\mathtt{A}} = 2, \ell_{\mathtt{B}} = 3$. E は, $\mathbb{F}_{p^2}$ 上定義された超特異楕円曲線であり, $P_{\mathtt{A}}, Q_{\mathtt{A}}$ は, $E[\ell_{\mathtt{A}}^{e_{\mathtt{A}}}]$ の基底, $P_{\mathtt{B}}, Q_{\mathtt{B}}$ は, $E[\ell_{\mathtt{B}}^{e_{\mathtt{B}}}]$ の基底である. また, アリス (Alice) 側, ボブ (Bob) 側の乱数空間を, それぞれ $\mathcal{K}_{\mathtt{A}} = \mathbb{Z}/\ell_{\mathtt{A}}^{e_{\mathtt{A}}}\mathbb{Z}$, $\mathcal{K}_{\mathtt{B}} = \mathbb{Z}/\ell_{\mathtt{B}}^{e_{\mathtt{B}}}\mathbb{Z}$ とする. [*1]

■SIDH 同種写像問題, SIDH 問題, およびその判定版問題　SIDH 鍵共有に関して離散対数問題にあたる基本問題は, 定義 5.2 の SIDH 同種写像問題であり, それに基づいて CDH, DDH 問題にあたるのは, それぞれ定義 5.4 の SIDH 問題, 定義 5.5 の判定版 SIDH 問題である.

定義 5.2 (SIDH 同種写像問題 [19, 30]) *SIDH* 鍵共有公開パラメータ $pp_{\mathtt{sidh}}$ と, そこで定義された E と $\ell_{\mathtt{A}}^{e_{\mathtt{A}}}$-同種な $E_{\mathtt{A}}$ と $P'_{\mathtt{B}}, Q'_{\mathtt{B}} \in E_{\mathtt{A}}[\ell_{\mathtt{B}}^{e_{\mathtt{B}}}]$ が与えられた時, $P'_{\mathtt{B}} = \phi_{\mathtt{A}}(P_{\mathtt{B}}), Q'_{\mathtt{B}} = \phi_{\mathtt{A}}(Q_{\mathtt{B}})$ となる次数 $\ell_{\mathtt{A}}^{e_{\mathtt{A}}}$ の同種写像 $\phi_{\mathtt{A}} : E \to E_{\mathtt{A}}$ を計算せよ. 特に, $\phi_{\mathtt{A}}$ の核 $\ker(\phi_{\mathtt{A}})$ の生成元 $R_{\mathtt{A}} \in E[\ell_{\mathtt{A}}^{e_{\mathtt{A}}}]$ を計算せよ.

ここでは, [30] に従って, 上記問題を SIDH 同種写像問題と呼ぶが, それは, [19] では CSSI 問題と呼ばれている. また, $\ell_{\mathtt{A}} \neq \ell_{\mathtt{B}}$ であるので, SIDH 鍵共有は非対称であり, $\phi_{\mathtt{B}}$ の計算を要求する (上とは異なる) 問題も定義できる. その定義は自明であるので, ここでは省略する. また, 次の判定版 SIDH 同種写像問題に関しても事情は同じである.

従来の離散対数問題に対しては存在しない, 定義 5.2 の判定版の問題が Galbraith–Vercauteren [30] によって以下のように定式化されている.

定義 5.3 (判定版 SIDH 同種写像問題 [30]) *SIDH* 鍵共有公開パラメータ $pp_{\mathtt{sidh}}$ と, $E_{\mathtt{A}}$ と $P'_{\mathtt{B}}, Q'_{\mathtt{B}} \in E_{\mathtt{A}}[\ell_{\mathtt{B}}^{e_{\mathtt{B}}}]$, $0 < n \leq e_{\mathtt{A}}$ が与えられた時, $P'_{\mathtt{B}} = \phi_{\mathtt{A}}(P_{\mathtt{B}}), Q'_{\mathtt{B}} = \phi_{\mathtt{A}}(Q_{\mathtt{B}})$ となる次数 $\ell_{\mathtt{A}}^{n}$ の同種写像 $\phi_{\mathtt{A}} : E \to E_{\mathtt{A}}$ が存在するかどうかを判定せよ.

ここでは, [30] に従って, 上記問題を判定版 SIDH 同種写像問題と呼ぶが, 補助点情報を含まず, 次数に関する条件も異なる類似の問題が, [19] では DSSI 問題と呼ばれている.

注目すべきは, Galbraith–Vercauteren [30] 及び Thormarker [50] によって, SIDH 同種写像問題から判定版 SIDH 同種写像問題への (多項式時間) 帰着が示されていることである. つまり, 両問題は, 多項式時間帰着の意味で同程度に困難な問題であり, 攻撃しやすく見える判定版 SIDH 同種写像問題を攻撃の対象にすれば十分であることがわかる.

定義 5.4 (SIDH 問題 [19, 26]) *SIDH* 鍵共有公開パラメータ $pp_{\mathtt{sidh}}$ に対し, 乱数 $k_{\mathtt{A}} \leftarrow_R \mathcal{K}_{\mathtt{A}}$ による点 $R_{\mathtt{A}} = P_{\mathtt{A}} + k_{\mathtt{A}} Q_{\mathtt{A}}$ によって核 $\ker(\phi_{\mathtt{A}}) = \langle R_{\mathtt{A}} \rangle$ が生成される同種写像を $\phi_{\mathtt{A}} : E \to E_{\mathtt{A}}$, 乱数 $k_{\mathtt{B}} \leftarrow_R \mathcal{K}_{\mathtt{B}}$ による点 $R_{\mathtt{B}} = P_{\mathtt{B}} + k_{\mathtt{B}} Q_{\mathtt{B}}$ によって核 $\ker(\phi_{\mathtt{B}}) = \langle R_{\mathtt{B}} \rangle$ が生成される同種写像を $\phi_{\mathtt{B}} : E \to E_{\mathtt{B}}$ とする. 超特異楕円曲線 $E, E_{\mathtt{A}}, E_{\mathtt{B}}$ とその上の点 $\phi_{\mathtt{A}}(P_{\mathtt{B}}), \phi_{\mathtt{A}}(Q_{\mathtt{B}}) \in E_{\mathtt{A}}$, $\phi_{\mathtt{B}}(P_{\mathtt{A}}), \phi_{\mathtt{B}}(Q_{\mathtt{A}}) \in E_{\mathtt{B}}$ が与えられた時, $E/\langle R_{\mathtt{A}}, R_{\mathtt{B}} \rangle$ の j 不変量を計算せよ.

ここでは, [26] に従って, 上記問題を SIDH 問題と呼ぶが, それは, [19] では SSCDH 問題と呼ばれ, [21] では SI-CDH 問題と呼ばれている.

[*1] [19] など SIDH 鍵共有の初期の文献では, 同種写像 $\phi_{\mathtt{A}}$ の核生成点 $R_{\mathtt{A}}$ を $R_{\mathtt{A}} = m_{\mathtt{A}} P_{\mathtt{A}} + n_{\mathtt{A}} Q_{\mathtt{A}}$ としていたが, SIKE 提案 [34] など最近の文献では, $R_{\mathtt{A}} = P_{\mathtt{A}} + k_{\mathtt{A}} Q_{\mathtt{A}}$ としているので, ここでは (アリス側の) 乱数空間は $\mathcal{K}_{\mathtt{A}} = \mathbb{Z}/\ell_{\mathtt{A}}^{e_{\mathtt{A}}}\mathbb{Z}$ として乱数は $k_{\mathtt{A}} \leftarrow_R \mathcal{K}_{\mathtt{A}}$ と生成する.

定義 5.5 (判定版 SIDH 問題 [19, 21]) *SIDH* 鍵共有公開パラメータ $pp_{\mathtt{sidh}}$ に対し, 乱数 $k_{\mathtt{A}} \leftarrow_R \mathcal{K}_{\mathtt{A}}$ による点 $R_{\mathtt{A}} = P_{\mathtt{A}} + k_{\mathtt{A}}Q_{\mathtt{A}}$ によって核 $\ker(\phi_{\mathtt{A}}) = \langle R_{\mathtt{A}} \rangle$ が生成される同種写像を $\phi_{\mathtt{A}} : E \to E_{\mathtt{A}}$, 乱数 $k_{\mathtt{B}} \leftarrow_R \mathcal{K}_{\mathtt{B}}$ による点 $R_{\mathtt{B}} = P_{\mathtt{B}} + k_{\mathtt{B}}Q_{\mathtt{B}}$ によって核 $\ker(\phi_{\mathtt{B}}) = \langle R_{\mathtt{B}} \rangle$ が生成される同種写像を $\phi_{\mathtt{B}} : E \to E_{\mathtt{B}}$ とする. 次の *2* つの分布を判別せよ.

- 超特異楕円曲線 $E, E_{\mathtt{A}}, E_{\mathtt{B}}$ とその上の点 $\phi_{\mathtt{A}}(P_{\mathtt{B}}), \phi_{\mathtt{A}}(Q_{\mathtt{B}}) \in E_{\mathtt{A}}$, $\phi_{\mathtt{B}}(P_{\mathtt{A}}), \phi_{\mathtt{B}}(Q_{\mathtt{A}}) \in E_{\mathtt{B}}$ および $E/\langle R_{\mathtt{A}}, R_{\mathtt{B}} \rangle$ の j 不変量からなる分布, または
- 超特異楕円曲線 $E, E_{\mathtt{A}}, E_{\mathtt{B}}$ とその上の点 $\phi_{\mathtt{A}}(P_{\mathtt{B}}), \phi_{\mathtt{A}}(Q_{\mathtt{B}}) \in E_{\mathtt{A}}$, $\phi_{\mathtt{B}}(P_{\mathtt{A}}), \phi_{\mathtt{B}}(Q_{\mathtt{A}}) \in E_{\mathtt{B}}$ および超特異 j 不変量の集合から一様ランダムにサンプリングされた j 不変量からなる分布.

ここでは, SIDH 問題の判定版であるので, 上記問題を判定版 SIDH 問題と呼ぶが, それは, [19] では SSDDH 問題と呼ばれ, [21] では SI-DDH 問題と呼ばれている. また, 定義 5.5 は, [19, 21] 内の定義と少し異なる. [19, 21] では「超特異 j 不変量の集合から一様ランダムにサンプリングされた j 不変量」の代わりに, 「(適切な) ランダム点 $R'_{\mathtt{A}}, R'_{\mathtt{B}}$ による $E/\langle R'_{\mathtt{A}}, R'_{\mathtt{B}} \rangle$ の j 不変量」としている. 定義 5.5 の方が適切と思われる.

■超特異楕円曲線 ℓ-同種写像グラフ 上記問題は, 全て超特異楕円曲線間の ℓ-同種写像からなる同種写像グラフ (パイザーグラフ [43] ともいう) の上の問題として定式化するのが自然である. その頂点集合は, 超特異 j 不変量で与えられて, その辺集合は, ℓ-同種写像で与えられる. そして, このグラフは, ラマヌジャングラフという最適な拡張率をもつエクスパンダーグラフになっていることが安全性上重要であり, その帰結として, グラフ上, 比較的少ないステップ数のウォークを行うことで, 終点分布が一様分布を大変良い精度で近似することがわかる ([43, 8] を参照). これにより, 頂点数 $O(p)$ のグラフでの一様分布がセキュリティパラメータ λ の多項式時間で得られる.

■自己準同型環計算問題との関係 同種写像問題の初期の考察である Kohel の博士論文 [36] は, その題目が示す通り, 有限体上の楕円曲線の自己準同型環を計算するアルゴリズムを示したものであり, その過程で自然に同種写像計算問題が関連することが示されている.

その後, 2014 年に, Kohel ら [37] により, 同種写像問題と同種写像グラフでの道探索問題の関係が調べられ, それを基にして, Eisenträger ら [16] によって, いくつかのヒューリスティックな仮定の下では, 同種写像問題と自己準同型環計算問題の間に両方向の多項式時間帰着が付くことが示された. 特殊な楕円曲線は, 簡明な自己準同型環を持つことから, そのような楕円曲線を始点に選んだ場合の安全性についても, [42, 16] において調べられている.

5.1.3 CSIDH 鍵共有の安全性の根拠となる問題

素体 $\mathbb{F}_p$ 上定義された超特異楕円曲線間の同種写像問題の困難性に基づく鍵共有法として, CSIDH 鍵共有 (5.2.2 節参照) が 2018 年になって Castryck らによって提案された [7]. その安全性の根拠となる計算問題を [13, 48] に従ってまとめる.

■CSIDH 鍵共有の公開パラメータ CSIDH 鍵共有で, 公開パラメータは $pp_{\mathtt{csidh}} = (\mathcal{O}, (\mathfrak{l}_1, \mathfrak{l}_2, \ldots, \mathfrak{l}_n), E, B)$ で与えられる. ここで, $\mathcal{O}$ は虚 2 次代数体の整環 (オーダー), $\mathfrak{l}_1, \mathfrak{l}_2, \ldots, \mathfrak{l}_n$ はノルムが小さい奇素数 ℓ_i になる $\mathcal{O}$ の素イデアルで, ℓ_i は $\ell_i = \mathfrak{l}_i\bar{\mathfrak{l}}_i$ $(i = 1, 2, \ldots, n)$ と 2 個の異なる素イデアル $\mathfrak{l}_i, \bar{\mathfrak{l}}_i$ の積に分解している. そして $p + 1 = 4 \cdot \ell_1 \cdots \ell_n$ とした時, p は素数である必要がある. 小奇素数 ℓ_i は, 例えば, $\ell_1 = 3, \ell_2 = 5, \ldots$ である. E は, $\mathbb{F}_p$ 上定義されて, $\mathcal{O}$ を $\mathbb{F}_p$-自己準同型環にもつ超特異楕円曲線である. B は指数 e_i のノルムの上限値, すなわち $-B \le e_i \le B$ となる指数 e_i を CSIDH 鍵共有では使う.

■CSIDH 鍵共有の抽象形　CSIDH 鍵共有での基本演算は, $\mathbb{F}_p$-自己準同型環に虚 2 次代数体の整環 (オーダー)$\mathcal{O}$ をもつ楕円曲線集合 X に対する $\mathcal{O}$ のイデアル類群 $G = \mathrm{cl}(\mathcal{O})$ の群作用 $(g, x) \mapsto gx \in X$ (但し $g \in G, x \in X$) として理解できる. その群作用は, 自由かつ推移的である. この群作用の詳細に関しては, 5.2.1 節を参照. その記法に従えば, CSIDH 鍵共有における同種写像問題は, この群作用の (G に関する) 逆関数 $(x, gx) \mapsto g$ を計算する問題と理解できる. このことから, CSIDH 鍵共有基本演算は, 一方向性群作用の特殊例であり, また, このような一方向性群作用をもつ (等質) 空間 X は, Hard Homogeneous Space (HHS) と呼ばれる [13, 48][*2].

■CSIDH ベクトル化問題, CSIDH 並列化問題　CSIDH 鍵共有に関して離散対数問題にあたる基本問題は, 以下の CSIDH ベクトル化問題 1(CSIDH Vectorization 問題 1) であり, 更に, それに基づいた CDH 問題は, CSIDH 並列化問題 1(CSIDH Parallelization 問題 1) である.

定義 5.6 (CSIDH ベクトル化問題 1 [13, 7, 48]) *CSIDH* 鍵共有公開パラメータ $pp_{\mathtt{csidh}}$ と, $\mathbb{F}_p$ 上定義されており $\mathbb{F}_p$-自己準同型環 $\mathcal{O}$ をもつ超特異楕円曲線 $E, E_{\mathtt{A}}$ が与えられた時, $E_{\mathtt{A}} = [\mathfrak{a}]E$ となる $\mathcal{O}$ のイデアル $\mathfrak{a}$ を計算せよ. 但し, $\mathfrak{a}$ の E への作用が効率的に計算可能な場合に限る. 例えば, $\mathfrak{a}$ が小さい次数のイデアル積で与えられる場合などである.

ここでは, [13, 48] に従って, 上記問題を CSIDH ベクトル化問題 1 と呼ぶが, それは, [7] では CSIDH 鍵復元 (CSIDH Key Recovery) 問題と呼ばれている.

定義 5.7 (CSIDH 並列化問題 1 [13, 7, 48]) *CSIDH* 鍵共有公開パラメータ $pp_{\mathtt{csidh}}$ と, $\mathbb{F}_p$ 上定義されており $\mathbb{F}_p$-自己準同型環 $\mathcal{O}$ をもつ超特異楕円曲線 $E, E_{\mathtt{A}} = [\mathfrak{a}]E, E_{\mathtt{B}} = [\mathfrak{b}]E$ (但し, $\mathfrak{a}, \mathfrak{b}$ は群作用が効率的に計算できる $\mathcal{O}$ のイデアル) が与えられた時, $[\mathfrak{a}\mathfrak{b}]E = [\mathfrak{b}]E_{\mathtt{A}} = [\mathfrak{a}]E_{\mathtt{B}}$ を計算せよ.

ここでは, [13, 48] に従って, 上記問題を CSIDH 並列化問題 1 と呼ぶ.

現在, イデアル類群 $G = \mathrm{cl}(\mathcal{O})$ の構造計算を多項式時間で行う (古典) アルゴリズムは知られていないため, G 上の一様分布からの効率的なサンプリング法も知られていない. よって, 近似的にその一様サンプリングを行う効率的な (秘密鍵) サンプリング法を用いて CSIDH 鍵共有は与えられる (5.2.2 節参照). それに従って, 上記の問題もそれぞれ修正されて, それらを CSIDH ベクトル化問題 2, CSIDH 並列化問題 2 として以下に与える.

定義 5.8 (CSIDH ベクトル化問題 2 [7, 18]) *CSIDH* 鍵共有公開パラメータ $pp_{\mathtt{csidh}}$ と, $\mathbb{F}_p$ 上定義されており $\mathbb{F}_p$-自己準同型環 $\mathcal{O}$ をもつ超特異楕円曲線 E, および $[-B, B]^n \subset \mathbb{Z}^n$ から一様に選んだ $(e_1, e_2, \ldots, e_n)$ により $\mathfrak{a} = \prod_{i=1}^{n} \mathfrak{l}_i^{e_i}$ となる $\mathfrak{a}$ によって $E_{\mathtt{A}} = [\mathfrak{a}]E$ となる $E_{\mathtt{A}}$ が与えられた時, $\mathfrak{a}$ と同値な $\mathfrak{a}'$, *i.e.*, $\mathfrak{a}' \in [\mathfrak{a}]$ を計算せよ.

定義 5.9 (CSIDH 並列化問題 2 [7, 18]) *CSIDH* 鍵共有公開パラメータ $pp_{\mathtt{csidh}}$ と, $\mathbb{F}_p$ 上定義されており $\mathbb{F}_p$-自己準同型環 $\mathcal{O}$ をもつ超特異楕円曲線 $E, E_{\mathtt{A}} = [\mathfrak{a}]E, E_{\mathtt{B}} = [\mathfrak{b}]E$ (但し, $\mathfrak{a}, \mathfrak{b}$ は共に, $[-B, B]^n \subset \mathbb{Z}^n$ から一様に選んだ $(e_1, e_2, \ldots, e_n)$ により $\prod_{i=1}^{n} \mathfrak{l}_i^{e_i}$ と表されるイデアル) が与えられた時, $[\mathfrak{a}\mathfrak{b}]E = [\mathfrak{b}]E_{\mathtt{A}} = [\mathfrak{a}]E_{\mathtt{B}}$ を計算せよ.

■CSIDH ベクトル化問題に対する準指数時間での量子攻撃　G の X への作用が自由かつ推移的であるなら, 群作用逆問題は, 隠れシフト問題に帰着されて, それは更に二面体群に関する隠れ部分群問題 (DHSP) に帰着する. DHSP には, 準指数時間で動く量子アルゴリズムが知られているので, 一般に一方向性群作用に基づいた暗号方式は, 量子計算機に対して準指数時間安全性しかもたない.

[*2] 相川ら [55] は, 強等質空間という訳語をあてている.

5.1.4 節で, CSIDH ベクトル化問題に対して, 準指数時間の量子アルゴリズムがあることに触れる. それにより, CSIDH 並列化問題も準指数時間で量子攻撃可能であることがわかるが, 最近, その逆, 並列化問題を解くオラクルを用いてベクトル化問題を解く多項式時間量子帰着アルゴリズムが提案されている [24].

5.1.4 3 つの基本同種写像問題の漸近的解読時間比較

前節までに見た超特異同種写像問題, SIDH 同種写像問題, CSIDH ベクトル化問題という 3 つの基本問題に対して漸近的解読時間が最速の既知アルゴリズムについて, 表 5.1 と, それに続く箇条書きでまとめる. そのヘッドライン, 例えば, [**超特異・古典**] は, 超特異同種写像問題に対する古典アルゴリズムに関する説明である.

表 5.1 3 つの同種写像問題の漸近的解読時間比較. ここで, $L_p[\alpha, c] = \exp\left((c+o(1))\,(\log p)^{\alpha}(\log\log p)^{1-\alpha}\right)$ とする.

	古典計算機による解読時間	量子計算機による解読時間
超特異同種写像問題	$\tilde{O}(\sqrt{p})$	$\tilde{O}(\sqrt[4]{p})$
SIDH 同種写像問題	$\tilde{O}(\sqrt[4]{p})$	$\tilde{O}(\sqrt[6]{p})$
CSIDH ベクトル化問題	$\tilde{O}(\sqrt[4]{p})$	$L_p[1/2, \sqrt{3}/2]$

- [**超特異・古典**] Galbraith [25] による中間一致攻撃で, 解読時間は $\tilde{O}(\sqrt{p})$ である.
- [**超特異・量子**] Biasse ら [4] によって時間計算量が $\tilde{O}(\sqrt[4]{p})$ の量子アルゴリズムが知られている. これは, $\mathbb{F}_p$ 上の超特異楕円曲線の同種写像問題に対する準指数時間量子アルゴリズム [9] と Grover アルゴリズムに基づく $\tilde{O}(\sqrt[4]{p})$ の道探索アルゴリズムを結合したものである.
- [**SIDH・古典**] SIDH 同種写像問題では, $\Theta(\sqrt{p})$ 次の同種写像問題に Galbraith による中間一致攻撃 [25] を適用するので, その解読時間は $\tilde{O}(\sqrt[4]{p})$ である.
- [**SIDH・量子**] $\Theta(\sqrt{p})$ 次の同種写像問題に, 谷のクロー探索 (claw finding) アルゴリズム [49] を適用して, その解読時間は $\tilde{O}(\sqrt[6]{p})$ である.
- [**CSIDH・古典**] $\mathbb{F}_p$ 上の超特異楕円曲線の同種写像問題に対する Delfs–Galbraith [15] の (古典) アルゴリズムを適用して, その解読時間は $\tilde{O}(\sqrt[4]{p})$ である.
- [**CSIDH・量子**] Childs ら [9] による通常同種写像問題に対する量子準指数時間アルゴリズムは, CSIDH ベクトル化問題に対しても有効である. (既に, 5.1.3 節でも触れた.) Galbraith–Stolbunov [29] によって改善アルゴリズムが提案された. また, 最近, Biasse [3] らは, メモリ使用量を削減するとともに, 量子メモリ使用を古典メモリ使用に転嫁するアルゴリズムを提案することで, より実現性を高める努力が行われている.

■CSIDH ベクトル化問題量子アルゴリズムの詳細な解析 特に, CSIDH ベクトル化問題に対する量子攻撃の正確な見積もりは, 与えられた安全性レベルを達成する p のビット長を決めるのに重要である. 従って, SIDH 鍵共有と CSIDH 鍵共有のどちらが効率的かという問題と関連して, この詳細な評価は, 現在, 注目を集めている.

Bonnetain–Schrottenloher [6] は, 詳細に CSIDH 攻撃量子アルゴリズムを検討して, これまで考えられていたより効率的に攻撃可能であると主張している. それにより, 彼らは, Castryck ら [7] が 56 ビット量子安全性レベルと主張していたパラメータが, 実際には 38 ビットレベルの量子安全性しか確保できないのでないか, という試算を述べている.

また, 一方, Bernstein ら [2] は, CSIDH 群作用を行う量子回路のサイズを具体的に見積もることで, 上記の準指数時間アルゴリズムが, 従来考えられていたより計算オーバーヘッドが大きいのでないか, つまり, 攻撃するのはより困難で

あろうと主張している.

前者は, CJS (Childs–Jao–Soukharev) アルゴリズム [9] そのものを対象としているのに対し, 後者は, そこでサブルーチンとして使う CSIDH 群作用の量子回路を対象としているので, 両者は相補的である. これにより, 今後も更なる詳細解析が必要とされる.

5.2 代表的な同種写像に基づく暗号方式の説明

De Feo と Jao によるディフィー–ヘルマン型の SIDH 鍵共有 [19] によって, 同種写像に基づいた公開鍵暗号が始めて与えられた. 最近になり, CRS 鍵共有のアイデアに基づき CRS 鍵共有よりも実現性が高い CSIDH 鍵共有 [7] も提案されているので, それもまとめる.

5.2.1 SIDH 鍵共有

De Feo ら [19] に従って SIDH 鍵共有を記述する. 5.1.2 節で定義したように, $p+1 = f \cdot \ell_{\mathtt{A}}^{e_{\mathtt{A}}} \ell_{\mathtt{B}}^{e_{\mathtt{B}}}$ で, p は素数, $\ell_{\mathtt{A}}, \ell_{\mathtt{B}}$ は 2 つの小素数である, 例えば, $\ell_{\mathtt{A}} = 2, \ell_{\mathtt{B}} = 3$, そして $\ell_{\mathtt{A}}^{e_{\mathtt{A}}} \approx \ell_{\mathtt{B}}^{e_{\mathtt{B}}} = 2^{\Theta(\lambda)}$ となるように素数 p を生成する. 但し, ここで λ はセキュリティパラメータ. その時, $\mathbb{F}_{p^2}$ 上定義された超特異楕円曲線 E の $\mathbb{F}_{p^2}$-有理点群は, $E(\mathbb{F}_{p^2}) \simeq (\mathbb{Z}/(p \pm 1)\mathbb{Z})^2 \supseteq (\mathbb{Z}/\ell_{\mathtt{A}}^{e_{\mathtt{A}}}\mathbb{Z})^2 \oplus (\mathbb{Z}/\ell_{\mathtt{B}}^{e_{\mathtt{B}}}\mathbb{Z})^2$ となる. アリスは, 2 次元空間 $(\mathbb{Z}/\ell_{\mathtt{A}}^{e_{\mathtt{A}}}\mathbb{Z})^2$ 内の 1 次元部分空間を自分の秘密同種写像 $\phi_{\mathtt{A}}$ の核 $\ker \phi_{\mathtt{A}} = \langle R_{\mathtt{A}} \rangle \subset E[\ell_{\mathtt{A}}^{e_{\mathtt{A}}}]$ として, ボブは, 2 次元空間 $(\mathbb{Z}/\ell_{\mathtt{B}}^{e_{\mathtt{B}}}\mathbb{Z})^2$ 内の 1 次元部分空間を自分の秘密同種写像 $\phi_{\mathtt{B}}$ の核 $\ker \phi_{\mathtt{B}} = \langle R_{\mathtt{B}} \rangle \subset E[\ell_{\mathtt{B}}^{e_{\mathtt{B}}}]$ とする. ここで, $(\mathbb{Z}/\ell_{\mathtt{A}}^{e_{\mathtt{A}}}\mathbb{Z})^2$ 内の 1 次元部分空間は, $\ell_{\mathtt{A}}^{e_{\mathtt{A}}+1}$ 個, つまり $O(\sqrt{p})$ 個あることに注意する. $(\mathbb{Z}/\ell_{\mathtt{B}}^{e_{\mathtt{B}}}\mathbb{Z})^2$ 内の 1 次元部分空間についても同様. そして, アリスとボブの間の SIDH 鍵共有の骨格は, 以下の可換図式によって与えられる.

$$
\begin{array}{ccc}
E & \xrightarrow{\phi_{\mathtt{A}}} & E_{\mathtt{A}} = E/\langle R_{\mathtt{A}} \rangle \\
{\scriptstyle \phi_{\mathtt{B}}}\downarrow & & \downarrow{\scriptstyle \phi_{\mathtt{AB}}} \\
E_{\mathtt{B}} = E/\langle R_{\mathtt{B}} \rangle & \xrightarrow{\phi_{\mathtt{BA}}} & E/\langle R_{\mathtt{A}}, R_{\mathtt{B}} \rangle
\end{array}
$$

但し
$\ker \phi_{\mathtt{A}} = \langle R_{\mathtt{A}} \rangle \subset E[\ell_{\mathtt{A}}^{e_{\mathtt{A}}}]$,
$\ker \phi_{\mathtt{B}} = \langle R_{\mathtt{B}} \rangle \subset E[\ell_{\mathtt{B}}^{e_{\mathtt{B}}}]$,
$\ker \phi_{\mathtt{BA}} = \langle \phi_{\mathtt{B}}(R_{\mathtt{A}}) \rangle \subset E_{\mathtt{B}}[\ell_{\mathtt{A}}^{e_{\mathtt{A}}}]$,
$\ker \phi_{\mathtt{AB}} = \langle \phi_{\mathtt{A}}(R_{\mathtt{B}}) \rangle \subset E_{\mathtt{A}}[\ell_{\mathtt{B}}^{e_{\mathtt{B}}}]$.

上記の可換図式をプロトコルとして成り立たせるために, $E[\ell_{\mathtt{A}}^{e_{\mathtt{A}}}] = \langle P_{\mathtt{A}}, Q_{\mathtt{A}} \rangle$, $E[\ell_{\mathtt{B}}^{e_{\mathtt{B}}}] = \langle P_{\mathtt{B}}, Q_{\mathtt{B}} \rangle$ となる基底 $(P_{\mathtt{A}}, Q_{\mathtt{A}}), (P_{\mathtt{B}}, Q_{\mathtt{B}})$ を選び, 公開パラメータ $pp_{\mathtt{sidh}} = (\ell_{\mathtt{A}}, \ell_{\mathtt{B}}, e_{\mathtt{A}}, e_{\mathtt{B}}, f; E, P_{\mathtt{A}}, Q_{\mathtt{A}}, P_{\mathtt{B}}, Q_{\mathtt{B}})$ を生成する. また, アリス側, ボブ側の乱数空間を, それぞれ $\mathcal{K}_{\mathtt{A}} = \mathbb{Z}/\ell_{\mathtt{A}}^{e_{\mathtt{A}}}\mathbb{Z}$, $\mathcal{K}_{\mathtt{B}} = \mathbb{Z}/\ell_{\mathtt{B}}^{e_{\mathtt{B}}}\mathbb{Z}$ とする. この公開パラメータの下, $SK_{\mathtt{Alice}} = SK_{\mathtt{Bob}}$ を共有鍵とする SIDH 鍵共有は以下で与えられる.

アリス		ボブ
$k_{\mathtt{A}} \leftarrow_R \mathcal{K}_{\mathtt{A}}$: アリスの秘密鍵,		$k_{\mathtt{B}} \leftarrow_R \mathcal{K}_{\mathtt{B}}$: ボブの秘密鍵,
$R_{\mathtt{A}} = P_{\mathtt{A}} + k_{\mathtt{A}} Q_{\mathtt{A}}$,	$\xrightarrow{E_{\mathtt{A}}, \phi_{\mathtt{A}}(P_{\mathtt{B}}), \phi_{\mathtt{A}}(Q_{\mathtt{B}})}$	$R_{\mathtt{B}} = P_{\mathtt{B}} + k_{\mathtt{B}} Q_{\mathtt{B}}$,
$\phi_{\mathtt{A}} : E \to E_{\mathtt{A}} = E/\langle R_{\mathtt{A}} \rangle$,	$\xleftarrow{E_{\mathtt{B}}, \phi_{\mathtt{B}}(P_{\mathtt{A}}), \phi_{\mathtt{B}}(Q_{\mathtt{A}})}$	$\phi_{\mathtt{B}} : E \to E_{\mathtt{B}} = E/\langle R_{\mathtt{B}} \rangle$,
$R_{\mathtt{BA}} = \phi_{\mathtt{B}}(P_{\mathtt{A}}) + k_{\mathtt{A}}\, \phi_{\mathtt{B}}(Q_{\mathtt{A}})$,		$R_{\mathtt{AB}} = \phi_{\mathtt{A}}(P_{\mathtt{B}}) + k_{\mathtt{B}}\, \phi_{\mathtt{A}}(Q_{\mathtt{B}})$,
$SK_{\mathtt{Alice}} = j(E_{\mathtt{B}}/\langle R_{\mathtt{BA}} \rangle)$.		$SK_{\mathtt{Bob}} = j(E_{\mathtt{A}}/\langle R_{\mathtt{AB}} \rangle)$.

図 5.1 SIDH 鍵共有の概要.

ここで, $\langle \phi_{\mathtt{B}}(P_{\mathtt{A}}) + k_{\mathtt{A}}\, \phi_{\mathtt{B}}(Q_{\mathtt{A}}) \rangle = \langle \phi_{\mathtt{B}}(R_{\mathtt{A}}) \rangle = \ker \phi_{\mathtt{BA}}$ かつ $\langle \phi_{\mathtt{A}}(P_{\mathtt{B}}) + k_{\mathtt{B}}\, \phi_{\mathtt{A}}(Q_{\mathtt{B}}) \rangle = \langle \phi_{\mathtt{A}}(R_{\mathtt{B}}) \rangle = \ker \phi_{\mathtt{AB}}$ であるので, j-不変量を用いることで, 鍵共有の正しさ $SK_{\mathtt{Alice}} = j(E_{\mathtt{B}}/\ker \phi_{\mathtt{BA}}) = j(E/\langle R_{\mathtt{A}}, R_{\mathtt{B}} \rangle) = j(E_{\mathtt{A}}/\ker \phi_{\mathtt{AB}}) = SK_{\mathtt{Bob}}$ が得

られることに注意する.

■**安全性**　例えば, アリスの公開鍵は, 自身の楕円曲線 $E_{\mathtt{A}}$ と共に, $E_{\mathtt{A}}$ 上の点 $\phi_{\mathtt{A}}(P_{\mathtt{B}}), \phi_{\mathtt{A}}(Q_{\mathtt{B}})$ を補助情報として含むことから, その安全性は, 判定版 SIDH 問題の困難性に基づく. また, Galbraith ら [27] によって, SIDH 鍵共有では公開鍵検証が困難であることが指摘されて, それを利用した能動的 (アクティブ) 攻撃があることが示された. その攻撃を回避するためには, Kirkwood らの対策 [35] が知られている.

■**実装研究**　SIDH 鍵共有は, 一般的に, 他の耐量子公開鍵暗号と比べて, データサイズは短く抑えられる利点をもつ反面, 演算処理時間が長くなる傾向にあるので, いかに計算時間を短縮するかというのが重要な課題である.

Costello ら [11, 10] により, 洗練された SIDH 実装研究が始まった. 最近も, Hutchinson–Karabina [32] において, SIDH 鍵共有内での同種写像の最適計算法を並列計算設定で考えた時の高速化について提案されている.

モバイル機器・組み込み機器向け実装研究では, Seo ら [47] によって, 32 ビットの ARMv7-A プロセッサ上で実用上問題ない実行時間が達成できることが示された. 一方, Koppermann ら [38] によって, 32 ビット ARMv7-M4 などの計算能力の低いプロセッサ上では実用的なソフトウェア実装を実現するのは難しいのではないか, という指摘もなされている.

5.2.2　CSIDH 鍵共有

Castryck ら [7] により提案された CSIDH 鍵共有を記述する. CSIDH 鍵共有は, 一方向性群作用をもつ等質空間 (X, G) 上で構成される. ここで, $X = \mathcal{E}\ell\ell_p(\mathcal{O})$ は, $\mathbb{F}_p$ 上定義されて $\mathbb{F}_p$-有理自己準同型環が固定された虚 2 次整環 (オーダー)$\mathcal{O}$ である超特異楕円曲線の集合であり, $G = \mathrm{cl}(\mathcal{O})$ は $\mathcal{O}$ のイデアル類群である. Castryck ら [7] は, 5.1.3 節で定義した CSIDH 並列化問題 2 の判定版問題の困難性に基づいて, CSIDH 鍵共有方式を提案した.

K を虚 2 次代数体, $\mathcal{O} \subset K$ をその整環とする, すなわちランク 2 の自由 $\mathbb{Z}$-加群である K の部分環である. $\mathcal{O}$ の分数イデアルは, $\alpha \in K^*$ と $\mathcal{O}$-イデアル $\mathfrak{a}$ によって $\alpha\mathfrak{a}$ と表される K 内の $\mathcal{O}$-部分加群である. $\mathfrak{a}\mathfrak{b} = \mathcal{O}$ となる $\mathcal{O}$-分数イデアル $\mathfrak{b}$ が存在する時に ($\mathcal{O}$-分数イデアル)$\mathfrak{a}$ は可逆であるという. そして, そのような $\mathfrak{b}$ が存在するならば, $\mathfrak{a}^{-1} = \mathfrak{b}$ と定義する. 可逆分数イデアルの集合 $I(\mathcal{O})$ はイデアル積に関してアーベル群をなす. この群には主イデアルからなる部分群 $P(\mathcal{O})$ が含まれており, $\mathcal{O}$ のイデアル類群は商群 $\mathrm{cl}(\mathcal{O}) = I(\mathcal{O})/P(\mathcal{O})$ によって定義される. どのイデアル類 $[\mathfrak{a}] \in \mathrm{cl}(\mathcal{O})$ にも整イデアルが存在してその代表として使うことができる. $\mathcal{O}$ のどの整イデアル $\mathfrak{a}$ も $\mathcal{O}$-イデアルの積として $\mathfrak{a}_s \not\subseteq \pi\mathcal{O}$ となる $\mathfrak{a}_s$ によって $(\pi\mathcal{O})^r\mathfrak{a}_s$ と表せる. ここで, π は, p 乗フロベニウス写像. この表示により, 整イデアル $\mathfrak{a}$ に対して楕円曲線 $E/E[\mathfrak{a}]$ とそこへの $N(\mathfrak{a})$ 次同種写像 $\varphi_{\mathfrak{a}} : E \to E/E[\mathfrak{a}]$ が以下のように定義される. ここで, $N(\mathfrak{a}) = \#\left(\mathcal{O}/\mathfrak{a}\right)$ は $\mathfrak{a}$ のノルムである. $\varphi_{\mathfrak{a}}$ の分離的な部分は $E[\mathfrak{a}] = \cap_{\alpha\in\mathfrak{a}_s} \ker\alpha$ を核にもつ同種写像であり, 純非分離的な部分はフロベニウス写像 π の r 回の繰り返しで与えられる. 同種写像 $\varphi_{\mathfrak{a}}$ および値域曲線 $E/E[\mathfrak{a}]$ は共に $\mathbb{F}_p$ 上定義されており $\mathbb{F}_p$-同型を除いて一意的に決まる. ここで主イデアルは自己準同型写像に対応しているので, 2 つのイデアルが同じイデアル類に属することと, 対応する同種写像が同型な値域曲線を与えることとは同値である. 更に, $\mathcal{E}\ell\ell_p(\mathcal{O})$ に属する 2 つの楕円曲線間の $\mathbb{F}_p$-同種写像 ψ はすべて上記の対応により可逆な $\mathcal{O}$-イデアルから得られる. そして分離部分 $\mathfrak{a}_s$ は ψ から $\mathfrak{a}_s = \{\alpha \in \mathcal{O} \mid \ker\alpha \supseteq \ker\psi\}$ によって復元できる. その対応は以下の定理にまとめられる.

定理 5.10 ([53, 46, 7]) *$\mathcal{O}$ を虚 2 次代数体内の整環 (オーダー) とする. もし $\mathcal{E}\ell\ell_p(\mathcal{O})$ が空集合でなければ, イデアル類群 $\mathrm{cl}(\mathcal{O})$ は $\mathcal{E}\ell\ell_p(\mathcal{O})$ に以下のように作用する.*

$$\begin{array}{rcl} \mathrm{cl}(\mathcal{O}) \times \mathcal{E}\ell\ell_p(\mathcal{O}) & \to & \mathcal{E}\ell\ell_p(\mathcal{O}) \\ ([\mathfrak{a}], E) & \mapsto & E/E[\mathfrak{a}], \end{array}$$

ここで, $\mathfrak{a}$ は類 $[\mathfrak{a}]$ を代表する整イデアルであり, 上の作用は自由である. 更に, $\mathcal{E}\ell\ell_p(\mathcal{O})$ が超特異楕円曲線を含む場合には, この作用は推移的であり, 含まない場合は, この作用は丁度 *2* つの軌道からなる.

以下では $E/E[\mathfrak{a}]$ を $[\mathfrak{a}]E$ と書くことにする. 上記の群作用に基づいて, 以下のように CSIDH 鍵共有プロトコル (Fig. 5.2) を定義する. 下の図で, $\mathfrak{a} \leftarrow \mathrm{cl}(\mathcal{O})$ と書いたのは, 実際にはイデアル類群 $\mathrm{cl}(\mathcal{O})$ からのサンプリングとして, 定義 5.8 の CSIDH ベクトル化問題および定義 5.9 の CSIDH 並列化問題に記載された $\mathfrak{a}$ のサンプリング法を用いる. モンゴメリ型楕円曲線 $E : y^2 = x^3 + ax^2 + x$ に対して, 係数 a は, E のモンゴメリ係数と呼ばれる. CSIDH 鍵共有では, 始点曲線 $E : y^2 = x^3 + x$ に対して, アリスとボブによって計算される楕円曲線はすべてモンゴメリ型楕円曲線である.

アリス		ボブ
$\mathfrak{a} \leftarrow \mathrm{cl}(\mathcal{O})$: アリスの秘密鍵,	$\xrightarrow{[\mathfrak{a}]E}$	$\mathfrak{b} \leftarrow \mathrm{cl}(\mathcal{O})$: ボブの秘密鍵,
$[\mathfrak{a}]\,E$ (のモンゴメリ係数) を計算,	$\xleftarrow[[\mathfrak{b}]E]{}$	$[\mathfrak{b}]\,E$ (のモンゴメリ係数) を計算,
$SK_{\texttt{Alice}} = [\mathfrak{a}]\,([\mathfrak{b}]E)$ のモンゴメリ係数.		$SK_{\texttt{Bob}} = [\mathfrak{b}]\,([\mathfrak{a}]E)$ のモンゴメリ係数.

図 5.2　CSIDH 鍵共有の概要

イデアル類群 $\mathrm{cl}(\mathcal{O})$ は可換なので, $[\mathfrak{a}]\,([\mathfrak{b}]E) = [\mathfrak{a}\mathfrak{b}]E = [\mathfrak{b}\mathfrak{a}]E = [\mathfrak{b}]\,([\mathfrak{a}]E)$ であり, そのモンゴメリ係数を考えれば $SK_{\texttt{Alice}} = SK_{\texttt{Bob}}$ となるので, アリスとボブは同じ鍵を共有できる. その計算アルゴリズムについては, Castryck ら [7] および Meyer–Reith [41] を参照のこと. その安全性は, 判定版の CSIDH 並列化問題 2 の困難性に基づく.

5.3　具体的な暗号方式

5.3.1 節で, NIST 標準化に提案された SIKE 公開鍵暗号と SIKE 鍵カプセル化方式 (KEM) を概説して, 5.3.2 節で, 同種写像に基づく認証付き鍵共有・グループ鍵共有についての現状を簡単に述べる. そして, 5.3.3 節で, 同種写像に基づいた署名方式の現状, 特に, 最近研究が始まった CSIDH ベースの署名方式である SeaSign 署名について少し詳しく説明する.

表 5.2　本節で扱う主要な同種写像に基づく暗号技術

文献	暗号化	鍵交換	署名
SIKE [34]	○	○	
SeaSign [18]			○

5.3.1　SIKE：SIDH ベース公開鍵暗号と鍵カプセル化方式

NIST 標準化に提案された SIKE [34] について述べる. 主たる提案者は, Waterloo 大学の David Jao であり, 他に企業研究者は Microsoft, InfoSec Global, Amazon, TI から, 大学研究者は Florida Atlantic 大学, UVSQ & Inria, Radboud 大学から名を連ねている.

モンゴメリ型楕円曲線 $E_{a,b}/\mathbb{F}_{p^2} : by^2 = x^3 + ax^2 + x$ を用いたスカラー倍算, 同種写像計算を用いて構成されており, 素数 p は, $p = 2^{e_2}3^{e_3} - 1$ であり, 始点曲線は $E_{0,1}/\mathbb{F}_{p^2} : y^2 = x^3 + x$ を用いている. 指数部分 (e_2, e_3) には, 3 つの値 $(e_2, e_3) = (250, 159), (372, 239), (486, 301)$ が採用されており, この公開パラメータ値は, それぞれ、SIKEp503,

SIKEp751, SIKEp964 と呼ばれている．ここで, 値 503, 751, 964 は, それぞれ素数 p のビット長を表す．これらのパラメータは, それぞれ, 安全性レベル $k = 128, 192, 256$ を満たすとされるが, その根拠は, 各パラメータに対して, 古典, 量子計算機のそれぞれについて, 安全性目標と安全性見積もりを計算した下の表 5.3 により与えられる．ここで, 古典安全性見積り, 量子安全性見積りは, それぞれ図 5.1 で, [**SIDH・古典**], [**SIDH・量子**] 欄に記載された見積り値を使って与えられている．

表 5.3　安全性レベルを決めるための安全性見積もり

	安全性レベル k	古典安全性目標 2^{k-1}	古典安全性見積もり $\min(\sqrt{2^{e_2}}, \sqrt{3^{e_3}})$	量子安全性目標 $\sqrt{2^k}$	量子安全性見積もり $\min(\sqrt[3]{2^{e_2}}, \sqrt[3]{3^{e_3}})$
SIKEp503	128	2^{127}	$1.00 \cdot 2^{125}$	2^{64}	$1.26 \cdot 2^{83}$
SIKEp751	192	2^{191}	$1.00 \cdot 2^{186}$	2^{96}	$1.00 \cdot 2^{124}$
SIKEp964	256	2^{255}	$1.45 \cdot 2^{238}$	2^{128}	$1.02 \cdot 2^{159}$

以下で, 関数 $\mathtt{isogen}_\ell(sk)$ は, 始点曲線 E から秘密鍵 sk に従って, ℓ-冪次の同種写像を計算した終点の曲線とその上の補助点からなる SIDH 鍵共有の公開鍵である．また, $\mathtt{isoex}_\ell(pk, sk)$ は, スカラー sk を使って pk に含まれる楕円曲線から ℓ-冪次の同種写像を計算した終点の曲線の j 不変量 (SIDH 鍵共有の共有鍵) を表す．以下では, 鍵空間 (乱数空間) を $\mathcal{K}_2 = \mathbb{Z}/2^{e_2}\mathbb{Z}$, $\mathcal{K}_3 = \mathbb{Z}/3^{e_3}\mathbb{Z}$ とする．

■SIKE 公開鍵暗号　通常の DH 鍵共有方式を基に ElGamal 暗号を構成するのと同じ要領で, SIDH 鍵共有方式を公開鍵暗号に変換したものが SIKE 公開鍵暗号である．

- **鍵生成：**　鍵空間 $\mathcal{K}_3$ からランダムな鍵を生成する：$sk_3 \leftarrow_R \mathcal{K}_3$．$sk_3$ に従って, 同種写像を計算してその値域曲線及び補助点を生成する：$pk_3 = \mathtt{isogen}_3(sk_3)$．秘密鍵 $sk = sk_3$, 公開鍵 $pk = pk_3$ とする．
- **暗号化：**　公開鍵 $pk = pk_3$ とメッセージ $m \in \mathcal{M}$ を入力とする．鍵空間 $\mathcal{K}_2$ からランダムな値を生成する：$sk_2 \leftarrow_R \mathcal{K}_2$．$sk_2$ に従って, 同種写像を計算してその値域曲線及び補助点を生成する：$c_0 = \mathtt{isogen}_2(sk_2)$．$pk_3$ と sk_2 から, SIDH 共有鍵にあたる j 不変量を計算する：$j = \mathtt{isoex}_2(pk_3, sk_2)$．鍵導出関数 F を用いてビット列へ変換する：$h = F(j)$．メッセージ m をマスクする：$c_1 = h \oplus m$．暗号文 (c_0, c_1) を出力する．
- **復号：**　秘密鍵 $sk = sk_3$ と暗号文 (c_0, c_1) を入力とする．c_0 と sk_3 から, SIDH 共有鍵にあたる j 不変量を計算する：$j = \mathtt{isoex}_2(c_0, sk_3)$．鍵導出関数 F を用いてビット列へ変換する：$h \leftarrow F(j)$．c_1 のマスクを取り除く：$m \leftarrow h \oplus c_1$．メッセージ m を出力する．

定理 5.11 ([34]) *SIKE 公開鍵暗号は, ランダムオラクルモデルにおいて, SIDH 問題困難性の仮定の下で, 選択平文攻撃に対して識別不可 (IND-CPA) 安全である．*

■SIKE 鍵カプセル化方式　SIKE 鍵カプセル化方式 (KEM) は, ランダムオラクルモデルで, SSDDH 仮定の下で IND-CCA 安全である．岡本–藤崎変換を基にした Hofheinz らの変換 [31] を施して, IND-CPA 安全な公開鍵暗号を, IND-CCA 安全な鍵カプセル化方式に変換したものである．

以下では, 前段落に記載した SIKE 公開鍵暗号の暗号化, 復号をそれぞれ $\mathtt{Enc}, \mathtt{Dec}$ と記載して, 本段落での KEM 暗号化, KEM 復号内でサブルーチンとして用いる．

- **鍵生成：**　鍵空間 $\mathcal{K}_3$ からランダムな鍵を生成する：$sk_3 \leftarrow_R \mathcal{K}_3$．$sk_3$ に従って, 同種写像を計算してその値域

曲線及び補助点を生成する：$pk_3 = \texttt{isogen}_3(sk_3)$. ランダムなビット列 s を生成する：$s \leftarrow_R \{0,1\}^n$. 秘密鍵 $sk = (s, sk_3)$, 公開鍵 $pk = pk_3$ とする.

- **鍵カプセル化：** 公開鍵 $pk = pk_3$ を入力とする. ランダムなビット列 m を生成する：$m \leftarrow_R \{0,1\}^n$. ハッシュ関数 G を用いて乱数値 r を生成する：$r = G(m \,||\, pk_3)$. 公開鍵 pk_3 と乱数値 r を用いて m を暗号化する：$(c_0, c_1) = \texttt{Enc}(pk_3, m; r)$. 鍵導出関数 H を用いて共有鍵 K を生成する：$K = H(m \,||\, (c_0, c_1))$. 暗号文 (c_0, c_1) と共有鍵 K を出力する.
- **デカプセル：** 秘密鍵 $sk = (s, sk_3)$, 公開鍵 $pk = pk_3$ と暗号文 (c_0, c_1) を入力とする. 秘密鍵 sk_3 を用いて (c_0, c_1) を復号する：$m' = \texttt{Dec}(sk_3, (c_0, c_1))$. ハッシュ関数 G を用いて乱数値 r' を生成する：$r' = G(m' \,||\, pk_3)$. r' に従って, 同種写像を計算してその値域曲線及び補助点を生成する：$c_0' = \texttt{isogen}_2(r')$. $c_0' = c_0$ が成立するかどうかによって, 鍵導出関数 H を用いて共有鍵 K を以下のように生成する：もし $c_0' = c_0$ であれば, $K = H(m' \,||\, (c_0, c_1))$, もし $c_0' \neq c_0$ であれば, $K = H(s \,||\, (c_0, c_1))$. 共有鍵 K を出力する.

定理 5.12 ([34]) *SIKE* 鍵カプセル化方式は, ランダムオラクルモデルにおいて, *SIDH* 問題困難性の仮定の下で, 選択暗号文攻撃に対して識別不可 *(IND-CCA)* 安全である.

[34] に従って, 表 5.4 に, SIKE 鍵カプセル化方式の鍵と暗号文のサイズをバイト長で示す. また, [34] に従い, 秘密鍵サイズ欄には正味の秘密鍵サイズとともに公開鍵も加えたサイズも括弧内に併記する.

表 5.4 SIKE 鍵カプセル化方式の鍵・暗号文のサイズ (バイト)

	秘密鍵 (含む公開鍵)	公開鍵	暗号文	共有鍵
SIKEp503	56 (434)	378	402	16
SIKEp751	80 (644)	564	596	24
SIKEp964	100 (826)	726	766	32

2019 年 1 月に公開された NIST PQC 標準化の現況報告書 [1] によると, 他の PQC 候補に比べて最も鍵サイズが小さいことが SIKE の 1 つの特長である. 表 5.4 に見るように, 256 ビットの安全性レベルとされる SIKEp964 鍵カプセル化方式でも, その公開鍵サイズが 750 バイトを下回っていることが [1] で特筆されている.

また, 他にも, NIST 報告書 [1] で指摘された SIKE 方式の特長と弱点を簡単にまとめる. その特長としては, 上に指摘した他の PQC 候補に比べて最も鍵サイズが小さいことの他に, これまでの楕円曲線暗号の実装研究に基づいた高速実装や耐サイドチャネル実装の研究蓄積があること, そして従来の楕円曲線暗号との古典/耐量子安全ハイブリッド方式の構成容易さが指摘されている. その弱点としては, 他の PQC 候補に比べて, 基本問題である同種写像問題の安全性検討がまだ不十分であることや同種写像計算に時間がかかることが挙げられている. (最新の安全性検討報告として [33, 12] がある.)

5.3.2 同種写像に基づく認証付き鍵共有・グループ鍵共有

■認証付き鍵共有 認証付き鍵共有は, 長い研究の歴史を持つが, 同種写像に基づく, 特に SIDH ベースの認証付き鍵共有は, 2018 年になって Galbraith [26], LeGrow ら [39], Longa [40], 藤岡ら [21] によって相次いで提案された. また, 最近, CSIDH に基づく認証付き鍵共有も, 藤岡ら [22] によって提案された.

■**グループ鍵共有**　2 ユーザ間の鍵共有を複数ユーザの鍵共有に拡張するグループ鍵共有も長く研究されてきているが, 同種写像に基づくものとしては, 最近, 古川ら [23] によって SIDH ベースグループ鍵共有が提案された. また, Boneh ら [5] は, 同種写像に基づく非対話グループ鍵共有の可能性を示唆しており, それを応用して 1 ラウンド認証グループ鍵共有も, 藤岡ら [22] によって検討されている.

5.3.3　同種写像に基づく署名方式

SIDH 鍵共有に基づく署名 [28] に簡単に触れた後, CSIDH 鍵共有に基づく署名 [18, 14] 方式である SeaSign 署名を説明する.

■**SIDH ベース署名**　SIDH 鍵共有をベースにして, 1 ビットずつ証明していく一般的なゼロ知識対話証明を適用して, Fiat-Shamir 変換による署名を構成すれば, 効率は非常に悪いが, 元の SIDH 鍵共有と同じ仮定の下で署名方式ができる [54, 28]. Galbraith ら [28] は, SIDH 同種写像問題ではなく (一般的な) 超特異同種写像問題の困難性に基づいた署名方式も構成した.

■**SeaSign 署名: CSIDH ベース署名**　De Feo と Galbraith [18] により, 前節の署名方式よりは効率的な CSIDH ベース署名が提案された. これは, CSIDH 鍵共有の数学的な構造を利用したものであり, 表 5.3.3 に見るように, まだ実用的とは言い難いが, 現実的な計算時間に収まる署名方式となっている. 以下では, [18] で, 「基本形」と呼ばれる SeaSign 署名方式を記載する. [18] では, 更に, 基本形でも使われたパラメータ t と共に, パラメータ s を導入して, 署名が短い方式や公開鍵が短い方式といった変形方式を定義している.

後続研究 [14] において, 更なる高速化が図られているが, まだ実用的とは言い難く, SeaSign 署名を改良することで, 実用的な速度性能を達成できるかどうかを見極めることは, 今後の同種写像ベース署名研究にとって重要な課題である.

以下では, ベクトル $\mathbf{e}$ の各成分は e_i $(i = 1, 2, \ldots, n)$ とする, 即ち, $\mathbf{e} = (e_1, e_2, \ldots, e_n)$ である. ベクトル $\mathbf{f}_k, \mathbf{z}_k$ についても同様の記法を用いる.

- **鍵生成：**　公開パラメータ $pp_{\mathtt{csidh}} = (\mathcal{O}, (\mathfrak{l}_1, \mathfrak{l}_2, \ldots, \mathfrak{l}_n), E, B)$, すなわち, 虚 2 次整環 $\mathcal{O}$, イデアル $\mathfrak{l}_1, \mathfrak{l}_2, \ldots, \mathfrak{l}_n$, $\mathbb{F}_p$-有理な超特異楕円曲線 E, 上限値 B を入力とする. 係数ベクトル $\mathbf{e} \leftarrow_R [-B, B]^n$ を生成する. $E_{\mathtt{A}} = [\prod_{i=1}^n \mathfrak{l}_i^{e_i}]\, E$ を計算して, 秘密鍵 $sk = \mathbf{e}$, 公開鍵 $pk = E_{\mathtt{A}}$ とする.
- **署名生成：**　公開パラメータ $pp_{\mathtt{csidh}}$, メッセージ $\mathtt{msg}$, 公開鍵 $pk = E_{\mathtt{A}}$, 秘密鍵 $sk = \mathbf{e}$ を入力とする. 各 $k = 1, 2, \ldots, t$ に対して, 係数ベクトル $\mathbf{f}_k \leftarrow_R [-(nt+1)B, (nt+1)B]^n$ を生成して, $\mathcal{E}_k = \left[\prod_{i=1}^n \mathfrak{l}_i^{f_{k,i}}\right] E$ を計算する. ハッシュ値を $b_1 || \cdots || b_t = H(j(\mathcal{E}_1), \ldots, j(\mathcal{E}_t), \mathtt{msg})$ とビット分解する. 各 $k = 1, 2, \ldots, t$ に対して, もし $b_k = 0$ であれば, $\mathbf{z}_k = \mathbf{f}_k$ として, もし $b_k = 1$ であれば, $\mathbf{z}_k = \mathbf{f}_k - \mathbf{e}$ として, もし $\mathbf{z}_k \notin [-ntB, ntB]^n$ であれば $\perp$ を出力する. そうでなければ, 署名 $\sigma = (\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_t, b_1, b_2, \ldots, b_t)$ を出力する.
- **署名検証：**　公開パラメータ $pp_{\mathtt{csidh}}$, メッセージ $\mathtt{msg}$, 公開鍵 $pk = E_{\mathtt{A}}$, 署名 σ を入力とする. まず, 署名 σ が, $\sigma = (\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_t, b_1, b_2, \ldots, b_t)$ というデータになっていることを確認する. 各 $k = 1, 2, \ldots, t$ に対して, もし $b_k = 0$ であれば, $\mathcal{E}_k = \left[\prod_{i=1}^n \mathfrak{l}_i^{z_{k,i}}\right] E$ を計算して, もし $b_k = 1$ であれば, $\mathcal{E}_k = \left[\prod_{i=1}^n \mathfrak{l}_i^{z_{k,i}}\right] E_{\mathtt{A}}$ を計算する. ハッシュ値を $b'_1 || \cdots || b'_t = H(j(\mathcal{E}_1), \ldots, j(\mathcal{E}_t), \mathtt{msg})$ とビット分解する. もし, $(b'_1, b'_2, \ldots, b'_t) = (b_1, b_2, \ldots, b_t)$ であれば, 受理を出力して, そうでなければ, 不受理を出力する.

定理 5.13 ([18]) *SeaSign 署名方式は, ランダムオラクルモデルにおいて, CSIDH ベクトル化問題 2 困難性の仮定の下で, 選択文書攻撃に対して存在的偽造不可 (EUF-CMA) 安全である.*

表 5.5　SeaSign 署名基本形, 短署名方式, 短公開鍵方式のデータサイズと処理時間見積り ([18], 2018/9/6 に投稿された版の ePrint 論文). ここで, $(\lambda, n, B, \log_2(p)) = (128, 74, 5, 16, 500)$ かつ第 1 列目では $(s,t) = (1,128)$, 第 2,3 列目では $(s,t) = (16,8)$. 単位 KiB は 1 KiB $= 2^{10}$ バイト $= 1024$ バイトとする. また, 時間見積りは [7] と同様に, $[B,-B]^n$ 内のベクトルが楕円曲線 E に作用する群作用演算 1 回に (多くとも)0.1 秒かかるのを基準にして見積もっている.

	基本形	短署名方式	短公開鍵方式
署名サイズ	19600 B	944 B	3092 B
公開鍵サイズ	63 B	4032 KiB	32 B
秘密鍵サイズ	32 B	16 B	1024 KiB
鍵生成時間見積り	0.1 秒	6554 秒	6554 秒
署名生成・検証時間見積り	123136 秒	474 秒	474 秒

5.4　まとめ

本章では, 同種写像に基づいた SIDH 鍵共有, CSIDH 鍵共有に関連して, その方式記述, 安全性研究, NIST 標準化の動向, そして, それらを基にした署名方式などについてまとめてきた.

[13] によると, Couveignes は, 1997 年の École Normale Supérieure でのセミナーで既に同種写像に基づく暗号技術を提案しており, ほぼ同時期に Kohel[36] や Galbraith[25] も, 同種写像問題に関する研究を始めていた. つまり, 同種写像暗号技術の研究は既に 20 年以上の歴史をもつ. そして, 最近になり, 耐量子計算機暗号の必要性が高まることで, 同種写像暗号技術は注目されて研究が進んでおり, 今後, 更に様々な方向性の研究が進むと思われる. もちろん, 未知の同種写像暗号方式や未知の攻撃が, 今後続く可能性もあるので, 最新研究動向を引き続き監視していく必要があるが, 現在知られた方式や安全性議論に関しても, 今後, 特に注意すべきこと数点について以下にまとめておく.

- NIST 暗号標準化に提案された同種写像暗号は, 5.3.1 節に記載した SIKE 方式だけであるので, SIKE 方式の動向は重要である. 特に IND-CCA 安全性を達成する SIKE 鍵カプセル化方式の安全性・実装研究の動向には, 注目する必要がある.
- SIKE 方式の基である SIDH 鍵共有の効率化, 特に計算時間の短縮や組み込み機器での高速実装法は, 他の耐量子公開鍵暗号と比べた時に, 重要な研究課題である.
- 5.1.4 節で述べたように, CSIDH 鍵共有に対する準指数時間量子アルゴリズムの詳細な解析・見積もりは, まだ今後の課題である. それに基づいて, SIDH 鍵共有と CSIDH 鍵共有の比較が行われていくので, SIKE 方式の標準化という観点からも重要である.
- 5.3.3 節で述べたように, 実用的な同種写像ベース署名は, まだ存在しない. 現在, もっとも実用に近い SeaSign 署名を改良することで達成できるのか, 新たな署名構成法を開拓する必要があるのか, などは今後の課題として残されている.

第 5 章の参考文献

[1] G. Alagic, J. Alperin-Sheriff, D. Apon, D. Cooper, Q. Dang, Y.-K. Liu, C. Miller, D. Moody, R. Peralta, R. Perlner, A. Robinson, and D. Smith-Tone. Status report on the first round of the NIST post-quantum cryptography standardization process. NISTIR 8240, NIST, January 2019.

[2] D. J. Bernstein, T. Lange, C. Martindale, and L. Panny. Quantum circuits for the CSIDH: optimizing quantum evaluation of isogenies. *IACR Cryptology ePrint Archive*, 2018:1059, 2018. To appear in EUROCRYPT 2019.

[3] J.-F. Biasse, A. Iezzi, and M.J. Jacobson Jr. A note on the security of CSIDH. In *INDOCRYPT 2018*, pages 153–168, 2018.

[4] J.-F. Biasse, D. Jao, and A. Sankar. A quantum algorithm for computing isogenies between supersingular elliptic curves. In *INDOCRYPT 2014*, pages 428–442, 2014.

[5] D. Boneh, D. Glass, D. Krashen, K. Lauter, S. Sharif, A. Silverberg, M. Tibouchi, and M. Zhandry. Multiparty non-interactive key exchange and more from isogenies on elliptic curves. In *MATHCRYPT 2018*, 2018. `https://eprint.iacr.org/2018/665`.

[6] X. Bonnetain and A. Schrottenloher. Quantum security analysis of CSIDH and ordinary isogeny-based schemes. *IACR Cryptology ePrint Archive*, 2018:537, 2018.

[7] W. Castryck, T. Lange, C. Martindale, L. Panny, and J. Renes. CSIDH: An efficient post-quantum commutative group action. In *ASIACRYPT 2018, Part III*, pages 395–427, 2018.

[8] D.X. Charles, K. E. Lauter, and E.Z. Goren. Cryptographic hash functions from expander graphs. *J. Crypt.*, 22(1):93–113, 2009.

[9] A.M. Childs, D. Jao, and V. Soukharev. Constructing elliptic curve isogenies in quantum subexponential time. *J. Math. Crypt.*, 8(1):1–29, 2014.

[10] C. Costello, D. Jao, P. Longa, M. Naehrig, J. Renes, and D. Urbanik. Efficient compression of SIDH public keys. In *EUROCRYPT 2017, Part I*, pages 679–706, 2017.

[11] C. Costello, P. Longa, and M. Naehrig. Efficient algorithms for supersingular isogeny Diffie–Hellman. In *CRYPTO 2016, Part I*, pages 572–601, 2016.

[12] C. Costello, P. Longa, M. Naehrig, J. Renes, and F. Virdia. Improved classical cryptanalysis of the computational supersingular isogeny problem. *IACR Cryptology ePrint Archive*, 2019:298, 2019.

[13] J.M. Couveignes. Hard homogeneous spaces. *IACR Cryptology ePrint Archive*, 2006:291, 2006.

[14] T. Decru, L. Panny, and F. Vercauteren. Faster SeaSign signatures through improved rejection sampling. *IACR Cryptology ePrint Archive*, 2018:1109, 2018.

[15] C. Delfs and S.D. Galbraith. Computing isogenies between supersingular elliptic curves over $\mathbb{F}_p$. *Des. Codes*

Cryptography, 78(2):425–440, 2016.

[16] K. Eisenträger, S. Hallgren, K. E. Lauter, T. Morrison, and C. Petit. Supersingular isogeny graphs and endomorphism rings: Reductions and solutions. In *EUROCRYPT 2018, Part III*, pages 329–368, 2018.

[17] L. De Feo. Mathematics of isogeny based cryptography. *CoRR*, abs/1711.04062, 2017.

[18] L. De Feo and S.D. Galbraith. SeaSign: Compact isogeny signatures from class group actions. *IACR Cryptology ePrint Archive*, 2018:824, 2018. We refer to the version which was posted at 6th September, 2018. To appear in EUROCRYPT 2019.

[19] L. De Feo, D. Jao, and J. Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *J. Math. Crypt.*, 8(3):209–247, 2014.

[20] L. De Feo, J. Kieffer, and B. Smith. Towards practical key exchange from ordinary isogeny graphs. In *ASIACRYPT 2018, Part III*, pages 365–394, 2018.

[21] A. Fujioka, K. Takashima, S. Terada, and K. Yoneyama. Supersingular isogeny Diffie–Hellman authenticated key exchange. In *ICISC 2018*, pages 177–195, 2018.

[22] A. Fujioka, K. Takashima, and K. Yoneyama. One-round authenticated group key change from isogenies. *IACR Cryptology ePrint Archive*, 2018:1033, 2018.

[23] S. Furukawa, N. Kunihiro, and K. Takashima. Multi-party key exchange protocols from supersingular isogenies. In *ISITA 2018*. IEEE Xplore, 2019.

[24] S. Galbraith, L. Panny, B. Smith, and F. Vercauteren. Quantum equivalence of the DLP and CDHP for group actions. *IACR Cryptology ePrint Archive*, 2018:1199, 2018.

[25] S.D. Galbraith. Constructing isogenies between elliptic curves over finite fields. *Journal of Computational Mathematics*, 2:118–138, 1999.

[26] S.D. Galbraith. Authenticated key exchange for SIDH. *IACR Cryptology ePrint Archive*, 2018:266, 2018.

[27] S.D. Galbraith, C. Petit, B. Shani, and Y.B. Ti. On the security of supersingular isogeny cryptosystems. In *ASIACRYPT 2016, Part I*, pages 63–91, 2016.

[28] S.D. Galbraith, C. Petit, and J. Silva. Identification protocols and signature schemes based on supersingular isogeny problems. In *ASIACRYPT 2017, Part I*, pages 3–33, 2017.

[29] S.D. Galbraith and A. Stolbunov. Improved algorithm for the isogeny problem for ordinary elliptic curves. *Applicable Algebra in Engineering, Communication and Computing*, 24(2):107–131, 2013.

[30] S.D. Galbraith and F. Vercauteren. Computational problems in supersingular elliptic curve isogenies. *Quantum Information Processing*, 17(10):265, 2018.

[31] D. Hofheinz, C.K. Hövelmanns, and E. Kiltz. A modular analysis of the Fujisaki-Okamoto transformation. In *TCC 2017, Part I*, pages 341–371, 2017.

[32] A. Hutchinson and K. Karabina. Constructing canonical strategies for parallel implementation of isogeny based cryptography. In *INDOCRYPT 2018*, pages 169–189, 2018.

[33] S. Jaques and J.M. Schanck. Quantum cryptanalysis in the RAM model: Claw-finding attacks on SIKE. *IACR Cryptology ePrint Archive*, 2019:103, 2019.

[34] D. Jao, R. Azarderakhsh, M. Campagna, C. Costello, L. De Feo, B. Hess, A. Jalali, B. Koziel, B. LaMacchia, P. Longa, M. Naehrig J. Renes, V. Soukharev, and D. Urbanik. Supersingular isogeny key encapsulation. *submission to NIST PQC Standardization*, 2017.

[35] D. Kirkwood, B.C. Lackey, J. McVey, M. Motley, J.A. Solinas, and D. Tuller. Failure is not an option: Stan-

dardization issues for post-quantum key agreement, 2015. Workshop on Cybersecurity in a Post-Quantum World.

[36] D. Kohel. *Endomorphism rings of elliptic curves over finite fields*. PhD thesis, University of California at Berkeley, 1996.

[37] D. Kohel, K. Lauter, C. Petit, and J.-P. Tignol. On the quaternion ℓ-isogeny path problem. *LMS Journal of Comupational Mathematics*, 17:418–432, 2014.

[38] P. Koppermann, E. Pop, J. Heyszl, and G. Sigl. 18 seconds to key exchange: Limitations of supersingular isogeny Diffie–Hellman on embedded devices. *IACR Cryptology ePrint Archive*, 2018:932, 2018.

[39] J. LeGrow, D. Jao, and R. Azarderakhsh. Modeling quantum-safe authenticated key establishment, and an isogeny-based protocol. *IACR Cryptology ePrint Archive*, 2018:282, 2018.

[40] P. Longa. A note on post-quantum authenticated key exchange from supersingular isogenies. *IACR Cryptology ePrint Archive*, 2018:267, 2018.

[41] M. Meyer and S. Reith. A faster way to the CSIDH. In *INDOCRYPT 2018*, pages 137–152, 2018.

[42] C. Petit. Faster algorithms for isogeny problems using torsion point images. In *ASIACRYPT 2017, Part II*, pages 330–353, 2017.

[43] A.K. Pizer. Ramanujan graphs and Hecke operators. *Bull. AMS*, 23(1):127–137, 1990.

[44] J. Renes. Computing isogenies between montgomery curves using the action of (0, 0). In *PQCrypto 2018*, pages 229–247, 2018.

[45] A. Rostovtsev and A. Stolbunov. Public-key cryptosystem based on isogenies. *IACR Cryptology ePrint Archive*, 2006:145, 2006.

[46] R. Schoof. Nonsingular plane cubic curves over finite fields. *Journal of Combinatorial Theory, Series A*, 46(2):183–208, 1987.

[47] H. Seo, Z. Liu, P. Longa, and Z. Hu. SIDH on ARM: faster modular multiplications for faster post-quantum supersingular isogeny key exchange. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2018(3):1–20, 2018.

[48] B. Smith. Pre- and post-quantum Diffie–Hellman from groups, actions, and isogenies. In *WAIFI 2018*, pages 3–40, 2018.

[49] S. Tani. Claw finding algorithms using quantum walk. *Theor. Comput. Sci.*, 410(50):5285–5297, 2009.

[50] E. Thormarker. Post-quantum cryptography: Supersingular isogeny Diffie–Hellman key exchange. Master's thesis, Stockholm University, 2017.

[51] J. Vélu. Isogénies entre courbes elliptiques. *C.R. Acad. Sc. Paris, Séries A.*, 273:238–241, 1971.

[52] L.C. Washington. *Elliptic Curves: Number Theory and Cryptography*. CRC Press, 2nd edition, 2008.

[53] W.C. Waterhouse. Abelian varieties over finite fields. *Annales scientifiques de l'É.N.S., 4^e série*, 2(4):521–560, 1969.

[54] Y. Yoo, R. Azarderakhsh, A. Jalali, D. Jao, and V. Soukharev. A post-quantum digital signature scheme on supersingular isogenies. In *FC 2017*, pages 163–181, 2017.

[55] 相川勇輔, 山崎努, 小貫啓史, 高木剛. 同種写像暗号 CSIDH の計算量評価と高速化パラメータ. In *SCIS 2019*, 3B3-2, 2019.

CRYPTREC
耐量子計算機暗号の研究動向調査報告書
[CRYPTREC TR-2001-2018]

発行日　2019年 4月5日　第1版
発行者

・〒184-8795
東京都小金井市貫井北町四丁目2番1号
国立研究開発法人情報通信研究機構
（サイバーセキュリティ研究所 セキュリティ基盤研究室）
NATIONAL INSTITUTE OF
INFORMATION AND COMMUNICATIONS TECHNOLOGY
4-2-1 NUKUI-KITAMACHI,KOGANEI
TOKYO,184-8795 JAPAN

・〒113-6591
東京都文京区本駒込二丁目28番8号
独立行政法人情報処理推進機構
（セキュリティセンター セキュリティ技術評価部 暗号グループ）
INFORMATION-TECHNOLOGY PROMOTION AGENCY, JAPAN
2-28-8 HONKOMAGOME,BUNKYO-KU
TOKYO,113-6591 JAPAN