公立はこだて未来大学 2023 年度 システム情報科学実習
グループ報告書

Future University Hakodate 2023 Systems Information Science Practice

Group Report

プロジェクト名

暗号と数理とセキュリティ

Project Name

Cryptography, Mathematics, and Security

グループ名

グループ A

Group Name

Group A

プロジェクト番号/Project No.

17-A

プロジェクトリーダ/Project Leader

猪股莉記　Riki Inomata

グループリーダ/Group Leader

樋口諒　Ryo Higuchi

グループメンバ/Group Member

猪股莉記　Riki Inomata
樋口諒　Ryo Higuchi
砂子祐貴　Yuki Sunako

指導教員

白勢政明　川口聡

Advisor

Masaaki Shirase　Satoshi kawaguchi

提出日

2024 年月日

Date of Submission

July 21, 2024

概要

本プロジェクトはインターネットが広く普及した現代の社会にとって必要不可欠な存在となってる暗号技術について研究を行った. インターネット通信が盛んになった今日では, 企業間での商取引にかかわる通信や政府の外交情報など情報の重要性がますます高まっている. そしてそういった機密情報を扱う上で, その通信の安全性を担保するために様々な暗号技術が開発されてきた一方で, 量子コンピュータの開発によって現代の暗号技術の安全性が脅かされている. そこで, 本グループでは量子コンピュータでも解読が難しいとされている耐量子計算機暗号の一つである格子暗号に注目し研究を進めた. 具体的には格子暗号によるテキストの暗号化/復号を行う web アプリケーションの実現を目標とした.

キーワード　格子暗号, 格子点探索問題, 公開鍵暗号,GGH 方式,LWE 方式, 耐量子計算機暗号, 完全準同型暗号

（※文責: 猪股莉記）

Abstract

This project studied cryptography, which has become indispensable in today's society where the Internet is widely used. Today, as Internet communications become more and more popular, the importance of information, such as communications related to business transactions between companies and diplomatic information from the government, is increasing. While various cryptographic techniques have been developed to ensure the security of such confidential information, the development of quantum computers threatens the security of modern cryptographic techniques. Therefore, this group focused on lattice cryptography, one of the quantum computer-resistant cryptosystems, which is considered to be difficult to decipher even with a quantum computer. Specifically, we aimed to realize a web application that encrypts/decrypts text using lattice cryptography.

Keyword Lattice-based cryptography,lattice point detection problem,public key ,GGH algorithm,LWE algorithm,post-quantum cryptography,fully homomorphic encryption

（※文責: 猪股莉記）

目次

第 1 章　はじめに

1.1　背景

インターネットが普及し、ネットワーク社会と化した現代において、様々なサービスのセキュリティを確保するための基盤技術として暗号技術が用いられている。中でも、公開鍵暗号方式と呼ばれる仕組みの暗号技術は、電子署名や暗号通信など私たちの身の回りの様々な場面で活用されている。一方で、量子コンピュータによって RSA 暗号や楕円曲線暗号などの主流な公開鍵暗号の安全性が危殆化されている。実際に、米国立標準技術研究所は 2030 年までに現在主流である鍵長が 2048bit の RSA 暗号の使用の停止を推奨しており [1]、日本でも同様の基準を設けている [2]。そこで、私たちは量子コンピュータが発達した社会でも活躍されると考えられる耐量子計算機暗号に注目した。中でも格子暗号と呼ばれる暗号技術は量子アルゴリズムへの耐性を有するのみならず、暗号化したまま演算を行える準同型暗号に属し、実用化が期待されている。私たちは格子暗号によるテキストの暗号化と復号を可能とすることを目的とした。

（※文責: 猪股莉記）

1.2　現状における問題点

現状の問題点が 2 つある。一つ目は前述したとおり、現在主流の公開鍵暗号が量子コンピュータが発達することにより破られてしまう点である。二つ目は格子暗号の鍵長が非常に大きい点である。RSA 暗号、楕円曲線暗号の鍵長と比較をしてみると、128 ビット安全を保つための鍵長が RSA 暗号は 3072bit, 楕円曲線暗号は 283bit である [1] のに対し、格子暗号の LWE 方式は 7800000bit,GGH 方式は 280000000bit[3] となっている。またほかにも、暗号化/復号を行う上でのパラメータを厳密に設定しなければならないといった欠点もあり、そのような要因が格子暗号の標準化を遅らせている。

（※文責: 猪股莉記）

1.3　課題の概要

上記の問題点に関わり、課題が 2 つある。一つ目は格子暗号を研究し理解を深めることで、量子コンピュータが発達した社会に適応する能力を身に着けることである。二つ目は、GGH 方式が整数を、LWE 方式がビットを暗号化/復号の対象としているためそれらをどのようにテキストの暗号化/復号と結びつけるかというものである。

（※文責: 猪股莉記）

第 2 章　到達目標

2.1　本プロジェクトにおける目的

本プロジェクトでは格子暗号を用いたテキストの暗号化/復号を可能にすることを目標とした。具体的には格子暗号の中でも GGH 方式と LWE 方式の 2 つの方式において、Web 上で暗号化/復号ができることを目標とした。

（※文責: 猪股莉記）

2.1.1　通常の授業ではなく、プロジェクト学習で行う利点

通常の授業と異なり、複数のコースの生徒が集まり議論することで互いがこれまでコースの授業で培ってきた知見を共有し合うことができる。また、成果物を作成する上で異なる担当分野がある上、その都度その都度で取り組むべき課題や習得するべき技術が変わるので、柔軟性の低い授業形式では成果物を作成することが難しい。

（※文責: 猪股莉記）

2.2　活動記録

本グループでは、格子暗号を用いて web 上で文字を暗号化しそれを復号することを目標として活動を行った。web 上で格子暗号を用いた暗号化システムの実装を果たすために行った各月での活動について以下にまとめた。6 章の方で具体的な活動内容に関してまとめた。

4 月 プロジェクトの決定
5 月 プロジェクトでの活動班の決定 — 教科書の輪読と共有
6 月 格子暗号についての学習と内容の共有 — 楕円曲線班との学習内容の共有
7 月 成果物作成のための環境の決定 — 中間発表
8 月 後期の活動予定の決定 — 各人で指定された内容の学習とプログラミング
9 月 暗号化ソフトウェアを元に、格子暗号により暗号化された暗号文を平文に復号するソフトウェアを 9 月に作成した。暗号化ソフトウェアと同様に GGH 方式と LWE 方式による復号ソフトウェアを作成した。
10 月 入力した文字を暗号化し、復号するシステムを Web ページに搭載する作業を行った。この Web ページは 8,9 月に作成したソフトウェアを利用した。また、Web ページ作成に関する勉強も行い、web のフレームワークには Flask を使用した。
11 月 引き続き、入力した文字を暗号化し、復号するシステムを Web ページに搭載する作業を行った。また、方向性の変更を行った。
12 月 最終発表会に向けた準備 — 最終報告書作成
1 月 最終報告書作成 — 最終講義

（※文責: 樋口諒）

2.3　課題の割り当て

各人の得意分野及び関連性、時間軸のスケジュールを基準に以下のように割り当てた。

猪股莉記... 格子暗号 (LWE 方式) によるテキストの暗号化/復号システム作成。web アプリケーションの LWE 方式ページの作成。web アプリケーションのレイアウト・ルーティング作成
樋口諒... 格子暗号の (GGH 方式) の暗号化/復号システムの作成。web アプリケーションの GGH 方式ページの作成。
砂子祐貴...Flask による web アプリケーションの骨組み作成。ポスターイラスト作成。

（※文責: 猪股莉記）

第 3 章　課題解決のプロセスの概要

前期 (4〜7 月) は格子暗号で文字を暗号化、復号するシステムを作成するための、格子暗号の基礎について学んだ。学んだ内容は、格子、格子暗号、格子暗号における GGH 方式である。次のページについて学んだ内容について記す。

5 月の前半に行ったプロジェクトでの活動班の決定方法はそれぞれの希望に沿って采配を行った。

5 月の後半に行った教科書の輪読については、「離散数学への入門」という教科書の 20 ページほどを輪読した。ここで学んだことは、RSA 暗号や最大公約数とユークリッドの互除法などの暗号に関わることを学習した。

6 月に行った楕円曲線暗号班と格子暗号班の学習内容と共有について、私たちは「暗号理論のための格子数学」を学習し、その学習した内容をプロジェクト時間中を使い楕円曲線暗号班に共有した。

7 月は主に中間発表の準備及び報告書作成を行った。中間発表ではポスター制作やスライドを行い、報告書作成ではグループ報告書やポートフォリオなどを提出した。

後期は (8〜12 月) は前期で学んだ格子暗号の基礎を踏まえて、GGH 方式、LWE 方式の暗号化、復号システムを作成し、それらのシステムを web に搭載した。

8 月及び 9 月では、夏休み中に後期で使用する web フレームワーク Flask の学習や、格子暗号の暗号方式である LWE 方式、GGH 方式において暗号化、復号ができるプログラムを Python で作成した。

10 月及び 11 月では、作成した暗号プログラムを web フレームワーク Flask を用いて web 上に搭載した。12 月及び 1 月では主に中間発表の準備及び報告書作成を行った。中間発表ではポスター制作やスライドを行い、報告書作成ではグループ報告書やポートフォリオなどを作成した

（※文責: 砂子祐貴）

第 4 章　課題解決のプロセスの詳細

4.1　学習内容

4.1.1　格子について

格子、格子基底

格子とは、$\mathbb{R}^m$ を m 次元のユークリッド空間としたとき、$\mathbb{R}^m$ の n 個 $(m \geq n)$ の線形独立なベクトル $\boldsymbol{b}_1, \cdots, \boldsymbol{b}_n$ のすべての整数結合の集合のことである。以下に格子についての式を示す。$L(\boldsymbol{b}_1, \cdots, \boldsymbol{b}_n) = \{\sum_{i=1}^{n} x_i \boldsymbol{b}_i : x_i \in \mathbb{Z}\}$ また整数 n と m はそれぞれ階数および次元と呼ばれ、$\boldsymbol{b}_1, \cdots, \boldsymbol{b}_n$ は線形独立のとき格子基底と呼ばれる。

（※文責: 砂子祐貴）

直交型基底、非直交型基底

格子には良い基底と悪い基底が存在する。良い基底は互いに直交に近いか直交している。対して、悪い基底は互いが平行に近くなっている。良い基底の一つとして直交型基底、悪い基底の一つとして非直交型基底が存在している。直交型基底は互いに直交していて、ベクトルの長さが、隣の格子点への距離に比べて、その距離と同等である特徴を持つ。非直交型基底は、互いに平行に近くなっていて、ベクトルの長さが、隣の格子点への距離に比べて、その距離より長い特徴を持つ。以下に 2 次元における直交型基底と非直交型基底の図を示す。

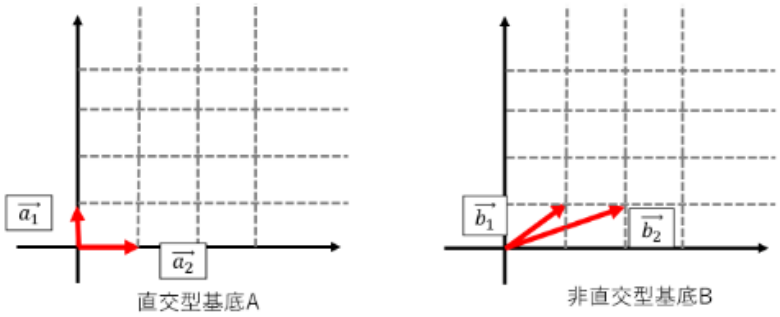

図 4.1　直交型基底と非直交型基底

（※文責: 砂子祐貴）

ユニモジュラ行列

整数行列でその行列式が $det(T) = \pm 1$ である正方行列 T の事を、ユニモジュラ行列と言う。ユニモジュラ行列を T、2 つの基底行列を A,B としたとき、$A = TB$ を満たすユニモジュラ行列 T が存在するならば、$L(A) = L(B)$ となる様な性質を持つ。前述で書かれている $L(A)$ の数学的な意味としては、A の格子基底で作られる格子の事を指す。

（※文責: 砂子祐貴）

4.1.2　格子暗号について

格子暗号の種類

格子暗号は「格子問題」と呼ばれる数学的問題の困難性を安全性の根拠とする公開鍵暗号の総称である。これまでに提案されている主な方式は

・「AD 方式」
・「GGH 方 式」
・「NTRU 方式」
・「LWE 方式」

の 4 つである。私たちはこの中の「GGH 方式」と「LWE 方式」についての 2 つについてプログラミングを行った。

（※文責: 砂子祐貴）

公開鍵暗号方式について

格子暗号は共通鍵方式ではなく公開鍵暗号方式である。公開鍵暗号方式とは復号に必要とされる鍵を秘密鍵、暗号化に必要とされる鍵を公開鍵として、受信者は公開鍵、秘密鍵を作成する。そして秘密鍵を受信者が秘密に保持し、公開鍵を利用者に公開する。2 つの鍵を作成するメリットとして、秘密鍵が奪われない限り、暗号化した情報が基本的に復号されないという特徴が挙げられる。下記に公開鍵暗号方式の手順を以下に示す。なお、平文とは暗号化されていないデータである。

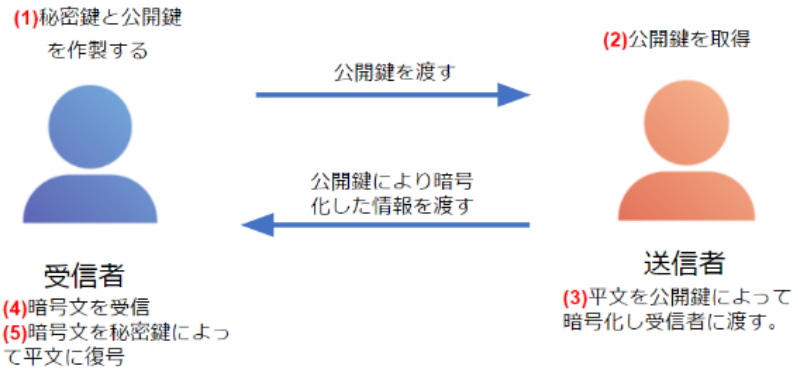

図 4.2　公開鍵暗号方式

(1) 受信者が秘密鍵と公開鍵を作成し、公開鍵を送信者に渡す
(2) 送信者が公開鍵を取得する
(3) 送信者が平文を公開鍵によって暗号化した暗号文を受信者に渡す
(4) 受信者が暗号文を取得する
(5) 受信者が暗号文を秘密鍵によって平文に復号する

（※文責: 砂子祐貴）

格子問題の困難性

格子暗号には様々な困難性が存在しているが、ここでは「格子問題」と「直交基底と非直交基底」の 2 つについて記述する。

まずは「格子問題」について記述する。「格子問題」とはある条件を満たす格子点を探索する問題である。「格子問題」には最短ベクトル問題、最近ベクトル問題、LWE 問題などが存在している。今回プログラミングを行った GGH 方式と LWE 方式では、GGH 方式では最近ベクトル問題の困難性を、LWE 方式では LWE 問題の困難性を利用している。後の方で最近ベクトル問題と最短ベクトル問題について記述するため、ここでは最短ベクトル問題のみ簡単な説明を行う。最短ベクトル問題は、原点に最も近い格子点を探す問題である。また、最近ベクトル問題と LWE 問題は最短ベクトル問題に帰着することができ、これを利用して格子暗号に攻撃を行うという手法も存在している。

次に「直交基底と非直交基底」について記述する。基本的に格子暗号は先ほど記述した格子問題の困難性を安全性の根拠としている。しかし、他に格子暗号として困難性を産んでいる要因が存在しているので説明する。GGH 方式では格子問題の他にも「直交基底と非直交基底」による格子全体の把握の困難性が利用されている。直交基底とは、それぞれのベクトルが全て直交する様な格子基底のことである。逆に非直交基底とは、それぞれのベクトルが非直交である基底である。性質として、直交型基底は格子点の全体像を把握しやすいため、格子問題を解くのが容易であるが、非直交型基底は格子点の全体像を把握しにくいため、格子問題を解くのが困難である。更に、直交型基底から同じ格子の非直交型基底を求めるのは簡単である。ユニモジュラ行列 T, 直交型基底 A, とすると、$B = AT$ は $L(B) = L(A)$ となり、B は非直交型基底となる。しかし、非直交型基底から同じ格子の直交型基底を求めるのは困難である。非直交型基底を B とすると、$L(B) = L(A)$ となる直交型基底 A やユニモジュラ行列 T を見つけるのは困難である。このような格子基底の状態による特徴および格子問題の困難性を利用して、直交型基底を秘密鍵、非直交型基底を公開鍵とした、安全性要件を満たす公開鍵暗号を構成したものが格子暗号の GGH 方式である。また、この後に記述する LWE 方式は安全性の観点から直交基底は利用しない。

（※文責: 樋口諒）

格子暗号の長所

格子暗号の長所は大きく 2 つあり、まず一つ目は量子アルゴリズム攻撃への耐性を有することである。現在使用されている公開鍵暗号である RSA 暗号や楕円曲線暗号が量子コンピュータによって破られることが危惧されている一方で、格子暗号をはじめとする耐量子計算機暗号はその耐性の強さから注目をあびている。二つ目のメリットは暗号化状態処理技術を有する点で、これによりデータを暗号化したまま処理することができる。従来の暗号技術であれば、暗号化したデータを処理する際に一度復号する必要があり、そこが攻撃者にとっての狙い目であった一方で、格子暗号はその問題点をクリアしている。このことは、医療・金融分野における企業間でのデータ共有や、クラウドサービスの運用に活用することができる。具体的には格子暗号を用いることで第三者が提供しているクラウドサービスにおいて、管理者が使用者のデータを盗用・改ざんできないようにすることができる。

（※文責: 猪股莉記）

格子暗号の短所

格子暗号の短所は大きく二つあり、一つ目は演算時間が長く、鍵長が大きくなってしまう点である。したがって計算能力の低い IoT デバイス等での実装が困難になる。二つ目は格子暗号を実用化する上での性質が完全に表明されていない点である。解読の困難性や得意分野などが未だ実験段階であるため、パラメータの設定などに注意しなければならない。

（※文責: 猪股莉記）

4.1.3　GGH 方式について

GGH 方式は最近ベクトル問題の困難性に則って作成された暗号方式である。最近ベクトル問題の説明は以下に示す。GGH 方式では、直交型基底を秘密鍵、非直交型基底を公開鍵として安全性を担保したものである。GGH 方式は「鍵生成」「暗号化」「復号」の三つの手順から成っている。

（※文責: 砂子祐貴）

最近ベクトル問題

最近ベクトル問題とは、次元 n において、ランダムに決められた点 C とある格子が与えられたときに、点 C に一番近い格子点 G を求める問題である。この問題は直交型基底の格子だと解きやすいが、非直交基底の格子だと解くのが困難になる。なぜかという直交型基底は格子の全体像を把握しやすいため点 C に一番近い格子点を探索することができるが、非直交型基底は格子の全体像を把握しにくいため解くのが困難になる。

例として 2 次元ベクトルにおける最近ベクトル問題について説明する。まずは直交型基底で最近ベクトルを解くときのことを考える。x 軸上に並行している基底ベクトルの 1 つを a_1、もう 1 つの y 軸上に並行している基底ベクトルを a_2 としたとき、ある格子点 G を次のように計算する。

$G=m_1a_1+m_2a_2$

この時、格子点 G が第 2 象限にある時 m_1 は負で m_2 が正であることは簡単に分かるが、非直交型基底になると基底ベクトル係数 m_2、m_1 を求めるのは困難になってしまう。

（※文責: 砂子祐貴）

「鍵生成」

・受信者は、まず次元 n を定め、全利用者に公開する。そして、自身の秘密鍵と公開鍵を設定する。ここではわかりやすいように $n = 2$ の場合の例を示す。

・直交型の基底 A=(a_1,a_2)（a_1,a_2 はそれぞれ 2 次元ベクトル）を決定し、A を秘密鍵として受信者だけが秘密に保持する。

・同じ格子を構成する非直交型の基底 B=(b_1,b_2)(b_1,b_2 はそれぞれ 2 次元ベクトル) を作成し、B を公開鍵として全員に公開する。

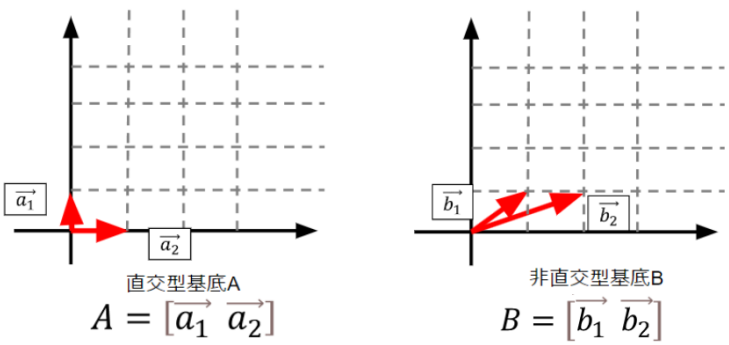

図 4.3　公開鍵、秘密鍵の生成

（※文責: 砂子祐貴）

「暗号化」

・送信者は、送りたい平文をあらかじめ決めておいた n に応じで n 分割する。ここでは送りたい平文を 2 つの整数（m_1,m_2）に分ける。

・次に、この平文（m_1,m_2）と公開鍵 B=(b_1,b_2) を利用して格子点 G=m_1b_1+m_2b_2 を計算する。

・さらに、$L(B)$ からなる格子点 G の座標を $(G_x$,$G_y)$ としたとき、C=$(G_x + r_x$,$G_y + r_y)$ を計算する。ここでの（r_x,r_y）は、点 C に最も近い格子点が G となるようなランダムな実数を選ぶようにする。そしてこの点 C が暗号化情報となるため、この点 C を受信者に送る。

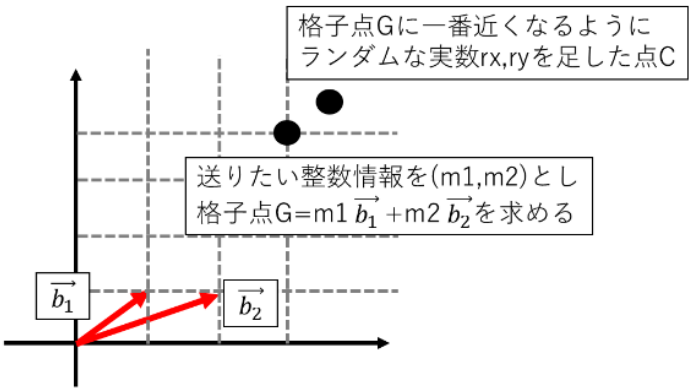

図 4.4　暗号化

（※文責: 砂子祐貴）

「復号」

・秘密鍵 A は直交型基底であり、格子の全体像を把握しやすいという性質を有するため、暗号文 C から格子点 $G(G_x, G_y)$ を簡単に求めることができる。

・受信者は秘密鍵 A を持っているため、点 C から格子点 G を求めることができる。

・受信者が格子点 G を求めた後、公開鍵 $B = (b_1, b_2)$ を用いて、下記の連立方程式を解くことにより平文 $(m_1$,$m_2)$ を復号することができる。

$$\begin{cases} G_x = m_1 b_{1x} + m_2 b_{2x} \\ G_y = m_1 b_{1y} + m_2 b_{2y} \end{cases}$$

（※文責: 砂子祐貴）

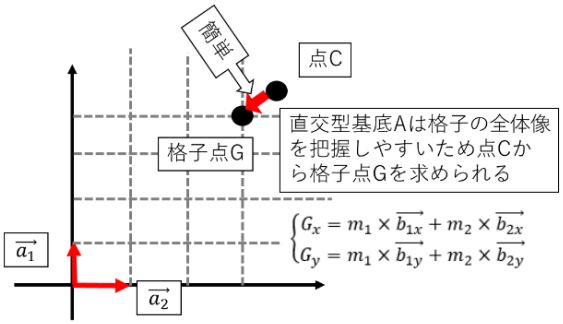

図 4.5　復号

GGH 方式の課題

GGH 方式には、安全性を担保するために必要な鍵の長さが 280000000bit 必要である点が大きな課題として挙げられる。それに対して、LWE 方式も 7800000bit 必要であり現在使用されている RSA 暗号 (3024bit) などよりも大きくなっているが、GGH 方式とは異なり鍵の長さを短くできると言われており研究が進んでいる。GGH 方式は鍵の長さが短くできないとされており、現在では研究があまり行われていない。

（※文責: 樋口諒）

4.1.4　LWE 方式について

概要

LWE 方式は格子暗号の方式の一つであり、LWE 問題という数学問題の困難性を安全性の根拠としている。座標が整数全体で表される格子ではなく素数 q により制限された、有限体上の格子を利用している。米国立標準技術研究所は耐量子暗号アルゴリズムとしてこの LWE 問題の一つを利用した方式を標準化することを発表している [4]。

LWE 問題とは

LWE 問題とはノイズが入った連立合同式を解くことを指す (図 4.6)。これを行列とベクトルで表すと図 4.7 のようになり、つまり LWE 問題とは素数 q を法とする有限体 Zq 上からとった誤差 e を付加した連立合同式について、A,b が与えられたとき、$As + e \equiv b(\mod q)$ を満たすベクトル s を求める問題と言える。

$$\begin{cases} 14s_1 + 15s_2 + 5s_3 + 2s_4 \equiv 8 \ (\mathrm{mod}\ 17) \\ 13s_1 + 14s_2 + 14s_3 + 6s_4 \equiv 16 (\mathrm{mod}\ 17) \\ 6s_1 + 10s_2 + 13s_3 + s_4 \equiv 12 (\mathrm{mod}\ 17) \\ 10s_1 + 4s_2 + 12s_3 + 16s_4 \equiv 3 \ (\mathrm{mod}\ 17) \end{cases}$$

図 4.6　LWE 問題の例

$$\mathbf{A} \cdot \mathbf{s} + \mathbf{e} = \mathbf{b} \pmod q$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1m} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nm} \end{bmatrix} \cdot \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_n \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ \vdots \\ e_m \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_m \end{bmatrix}$$

図 4.7　LWE 問題を行列で表した図

暗号化

ここでは 1bit を暗号化する例を説明する。

1. 共通パラメータとして格子の次元 n、ベクトル空間の次元 m(>n)、法とする素数 q、非直交型の基底 A(n 行 m 列)、実数 α を選択する

2. 各成分を $q-1$ 以下の正整数とした n 行ベクトル s をランダムに選択する

3. 各成分が実数 α により定まる確率分布に基づき選択された整数である m 行ベクトル e を選択する

4. 基底 A、秘密鍵 s、エラー項 e を組み合わせ $T = A \cdot s + e(\mod q)$ を作成し、T を公開鍵とする

5. 各成分が実数 α により定まる確率分布に基づき選択された整数である m 行ベクトル r を選択する

6. n 行 m 列の行列 A と r をかけ暗号文 $C1 = r \cdot A(\mod q)$ を計算する
7. 平文が 1 の場合 $M = \frac{q+1}{2}$,0 の場合 M=0 と設定したうえで、公開鍵 T と r を用いて $C2 = r \cdot T - M(\mod q)$ を作成し、C1,C2 を暗号文とする。

復号

1. 暗号文 C1,C2 と秘密鍵 s を用いて $w = C1 \cdot s - C2(mod\ q)$ を計算し、下のような条件式で平文を獲得する。

$$\text{平文} = \begin{cases} 1 & (\frac{-(q+1)}{2} < w < \frac{-(q+1)}{4} \text{または} \frac{(q+1)}{4} < w < \frac{(q+1)}{2}) \\ 0 & (\frac{-(q+1)}{4} < w < \frac{(q+1)}{4}) \end{cases} \tag{4.1}$$

（※文責: 猪股莉記）

4.2 成果物について (GGH 方式)

4.2.1 簡単なアルゴリズム紹介

第 4 章で GGH 方式の暗号化から復号までの手順を記述したため、ここではどのように鍵を作成し、日本語をどのように座標に当てはめて、GGH 方式を用いて暗号化し復号してるのか、具体例を使いながら説明していく。

まず、鍵の作成に関して説明する。最初に全てのベクトルが直交する様な基底を作成する。ここでは、それぞれのベクトルの長さが 3 から 9 の長さになるようにした。この格子は秘密鍵となる。ユニモジュラ行列を作成する。ユニモジュラ行列の作成には秘密鍵と同じ次元の単位行列に三つの変換のうち一つを行い、その後に変換によって作成した行列の全ての積を求めることによって、ユニモジュラ行列を作成している。私たちのプログラムでは、ユニモジュラ行列の作成に 4000 個の行列をかけ算することによって作成している。ここで、三つの変換について説明する。一つ目の変換は、単位行列の一つの要素に-1 を掛け合わせるという変換である。ここで、要素の選択は乱数によってランダムに決定している。二つ目の変換は、行を入れ替えるという変換になっている。行の選択も乱数によりランダムで決定している。三つ目の変換は、行列の要素に別の行列の要素を整数倍したものを加えるという変換になっている。行の選択は乱数によってランダムに選択している。また、これらの変換を行う確率は、一つ目の変換と二つ目の変換がそれぞれ 20% の確率で変換を行い、三つ目の変換は 60% の確率で変換を行う。以上の作業により、GGH 方式による通信の秘密鍵と公開鍵を作成した。

次に日本語の座標化に関して説明する。例えば、「あいう」という文字列を暗号化する場合、まず「あいう」を Unicode で整数値に変換すると、「4196743176255353094534」となる。この整数列を 2 桁ずつに分割して「41,96,74, ··,45,34」のようにする。その後に、この配列の頭に、何処でこの分割された配列が終了するのかを格納する。更に、「バイト列の長さ」と「整数列の長さ」を順に配列の後ろに格納する。先ほどの例を用いると、分割した後の配列の長さは 13「13,41,96,74, ··,45,34,22,9」のようになる。その後に、格子の次元に合わせるために 0 を格納していく。先ほどの例を用いると、「13,41,96,74, ··,45,34,22,9,0,0,0 ··,0」となる。その後に、この配列を暗号化する。暗号化の処理については先ほども述べた様に省略する。乱数に関して、暗号化での乱数の範囲は-0.5 から 2.4 とした。復号した後の処理では、配列から、「配列の長さ」、「バイト列の長さ」、「整数列の長

さ」の値を取り出した。更に余分な 0 を排除して、「41,96,74, ··,45,34」という状態に戻し、これらを結合することによって日本語に戻している。また、格子基底の次元は 200 と設定した。

（※文責: 樋口諒）

4.2.2　web ページ上での動作について

web ページ上での動作を簡単に説明する。アクセスすると、LWE 方式と GGH 方式の二つを選択する画面に遷移する。そこで、GGH 方式を選択すると、「encrypt」ボタンと「decrypt」ボタンが出てくる画面に遷移する。

図 4.8　web ページ 1

そこから以下の順で暗号化と復号を行う。「encrypt」を押すことにより、暗号化を行うページに遷移する。このページで暗号化したい文字を入力し、「Encrypt」ボタンを押すことで、秘密鍵と公開鍵と暗号文の 3 つの CSV ファイルがダウンロードされる。

図 4.9　web ページ 2

復号の際には、先ほど説明したもう一つの「decrypt」ボタンを押し、復号ページの飛び、その後に先ほどダウンロードしたファイルをアップし、「Decrypt」を押すことで復号されて画面に平文が表示されるような web アプリケーションとなっている。

図 4.10　web ページ 3

（※文責: 樋口諒）

4.3　成果物について (LWE 方式)

4.3.1　工夫点

先述したものを基本に暗号化と復号を行っている。相違点として、bit ではなくテキストを暗号化するために下記のようなシステムを作成した。

暗号化

1. 入力された文字をバイト列に変換し、バイト列をビット列に変換する
2. 先ほどと同じ手順で平文ビット一つ一つに対して、暗号文 C1,C2 を作成する (手順は 4.1.4 の 1〜7 の通り)
3.C1,C2 の要素 (整数) を unicode 文字に変換する。この際元の値が 0 だと unicode 化した結果が半角スペースになってしまい復号がうまくいかないので半角スペースは”無”という漢字に変換した。

復号

1. 与えられた Unicode 暗号文をビット一つ一つに分けて復号しなければいけないので、一回変換を行うために必要なサイズごとに分けて復号を行う。
2.unicode 暗号文を復号し得た C1,C2 と秘密鍵 s を用いて 4.1.4 の手順でビット列を得る。
3. 得たビット列をバイト列に変換し、平文を獲得する

web アプリケーション

Web アプリケーション上での暗号化/復号の手順を説明する。
1. タイトル画面から LWE 方式ページに遷移すると Encryption ボタンと Decryption ボタンがあるので Encryption ボタンを押し暗号化ページに遷移する
2. テキスト入力欄に暗号化したい文章を入力し Encrypt ボタンをクリックする
3.save ボタンが出現するのでクリックすると C1unicode 暗号文 (txt ファイル) と C2unicode 暗号文 (txt ファイル) と秘密鍵ファイル (csv ファイル) がダウンロードされる
4.LWE 方式ページの home に戻り、Decryption ボタンをクリックし復号ページに遷移する
5. ダウンロードした 3 つのファイルを選択し、decrypt ボタンをクリックする
6. 復号が成功したら画面に平文が出力される。失敗した場合エラーのアラートが出力される。

（※文責: 猪股莉記）

第 5 章　今後の課題と展望

5.1　システム全体

今後の課題として、「格子暗号による電子署名を用いた安全なメールのやり取りの実現」が挙げられる。

背景でも述べたように、量子コンピューターの登場によって現在主流になっている RSA 暗号や楕円曲線暗号の安全性が危険視されている。RSA 暗号や楕円曲線暗号が破られてしまうと今まで個人で所有してきた情報資産はすべて奪われてしまう。これらの事を防ぐため、まずは対量子計算機暗号である格子暗号を用いて安全なメールのやり取りを実現しようと考えたのが経緯である。

これを実現するためには主な問題点が 4 つ存在する。まず一つ目は、様々な方式がある格子暗号において、どの方式を使って電子署名を行うのか、どの方式が適しているのかが定まっていない点である。二つ目は格子暗号を用いた電子署名によってメールを行う上で、その環境をどのように構築するかが定かでない点である。仮想的なメールのやり取りが行える web ページを自分たちで作成する必要があると考えられる。三つ目は格子暗号のアルゴリズムを内装したソフトウェアおよび web ページの作成に係る技術の習得である。python や javascript の習得が必須になることが想定される。4 つ目は、通常の電子署名において本人確認の証明として用いられる電子証明書が認証局によって発行されるのだが、その工程をどのように仮想的に再現するかである。

全体の課題として「テキストファイルを介さず web 上で格子暗号の暗号化、復号をするシステムを作成」が挙げられる。

後期に作成した格子暗号を用いての web 上での文字の復号に関して、現在は暗号化情報のテキストファイルを選び、秘密鍵のテキストファイルを選んでからでないと平文への復号はできない。なぜこのようなことになったのかというと、今回の web 作成で使用したフレームワーク、Flask の変数の容量が少なかったことに起因する。Flask の変数容量が少なかったがために秘密鍵情報と暗号化情報が変数に保存できず、テキストファイルとして保存するしかなかった。しかし、Django という web フレームワークは拡張性が高いためこのような問題にも対処できると思われる。今後は、web のフレームワーク Django という拡張性の高いものを使い、この問題を解決していきたい。

（※文責: 砂子祐貴）

5.2　GGH 方式

GGH 方式の課題については第 4 章に記述したため、ここでは私たちが実装した GGH 方式についての課題について記述する。

一つ目は公開鍵の作成に関わる時間の長さである。全体的に実行時間が長いものになってしまっているが、特に、公開鍵の作成に時間を有することが問題として挙げられる。私たちは公開鍵の作成において、ユニモジュラ行列を作成し秘密鍵とユニモジュラ行列を掛け合わせることにより、公開鍵を作成している。このとき、格子の次元は 200 次元で、4000 回の行列のかけ算を行うことで公開鍵の作成を行っているが、この動作における実行時間がとても長くなってしまっている。およそではあるが約 3 秒程度かかってしまう。しかし、時間をこのように有していても、複雑な公開鍵

の作成はあまりできていないので、効率的に公開鍵を作成するようなアルゴリズムを考慮する必要がある。

二つ目は暗号化に関する乱数の設定に関してである。GGH 方式では、平文と公開鍵を掛け合わせたものに更に乱数を加えることにより、盗聴者は最近ベクトル問題を解かなければならなくなり、解読が困難となる。しかし、この乱数の範囲が短すぎると容易に解読され、範囲が長いと復号に長い時間を有してしまうことになってしまう。この部分も実行時間の長さに関わってしまっているため、この乱数の扱いについての学習と考察を行う必要があると考える。

三つ目は暗号文への変換についてである。私たちは文字列を暗号化する際に、一文字ずつ座標の値として入れ込むのではなく、文字列全体を整数化し、その値を 2 桁ずつ分割して平文を作成することで文字列の大きさを小さくしたが、これでは文字列が長かった場合に新たに座標を設定して、二つの暗号文を送らなければならないため、暗号文の大きさが大きくなってしまっている。しかし、一文字ずつ座標の値としてしまえば、暗号文の大きさを大きくしまっている。どちらにしろ、暗号文の大きさが大きくなってしまうため、この暗号文の大きさにバランスについて考察していかなければならないと考える。

（※文責: 樋口諒）

5.3 LWE 方式

今回実装した Web アプリケーションの LWE 方式ページについての今後の課題が 2 つある。
一つ目は暗号文のサイズが非常に大きくなってしまうことである。本 Web アプリケーションでは 1 つのビットを暗号化をすると格子の次元の二乗分のサイズの暗号文が出力される。今回格子の次元は 230 としているため $230 \times 230 = 52900$ 個の要素を持つ行列が暗号文の一つとなる。さらにそれらの数字一つ一つを Unicode に置き換え、その操作を暗号化したい文章のビット分だけ行うので、暗号文のサイズが非常に大きくなり計算時間が長くなってしまう。
二つ目の課題はパラメータの適切な設定についてである。現在は清藤ら [3] の示したパラメータに準じて暗号化/復号を行っているが、Unicode 暗号文についての最適なパラメータを求めることができなかった。
これらの課題を解決するために、今後は Ring-LWE や Compact-LWE など別種類の LWE 暗号についても研究を進め、暗号文のサイズや鍵長を小さくする方法を開発したい。また、パラメータを変更させながら解読必要時間等を計測し、安全性の評価を行うことで最適なパラメータを発見したい。

（※文責: 猪股莉記）

第 6 章　まとめ

6.1　プロジェクト活動

前期の活動

前期では、4 月にプロジェクトの決定を行った。そして、5 月の最初にプロジェクトメンバー全員で暗号に大きく関わる離散数学の教科書を輪読し、暗号に関する知識の習得を行った。教科書の内容として、整数論に関わる章を輪読した。具体的に四則演算の基本や体、mod に関する知識やフェルマーの小定理に関して学習し、輪読した。その後、5 月の末に、楕円曲線暗号を用いてブロックチェーンによる大学内での仮想通貨の実装を目指す楕円曲線暗号班と、格子暗号を用いたメッセージ通信アプリケーションの作成を目指す格子暗号班の 2 つに分かれて活動を行った。我々は格子暗号班として活動を行うことを決定した。また、6 月にはグループ間でそれぞれ学習している暗号理論をお互いに共有した。具体的に共有した内容として、格子の定義、ユニモジュラ行列、GGH 方式、最短ベクトル問題、最近ベクトル問題が挙げられる。更に、6 月中旬に、格子暗号を用いたメッセージ通信アプリケーションの作成を行うことを決定し、それに関する学習を行った。また、格子暗号による暗号通信を紙を用いて仮想的に行った。このとき、ユークリッド空間と格子基底の次元数は共に 2 と設定した。7 月の中間発表では前期で学習した内容を発表した。

夏休みの活動

夏休み中では、8 月に後期の活動予定の決定を行い、その際に各人の作業に関しても決定した。そして、各人で学習とプログラミングを行った。猪股莉記君は、LWE 方式に関するプログラミングと学習を主に行った。樋口諒君は、GGH 方式に関するプログラミングと学習を主に行った。砂子祐貴君は、HTML と Javascript に関する学習や、python を HTML に適用するためのフレームワークの学習を行った。9 月の中旬まで、8 月に決定した学習の続きを行った。

後期の活動

後期では夏休みの間に決定した役割分担のもと、それぞれの学習と共にプログラミングを行った。具体的には、9 月には web アプリケーション作成のための格子暗号のプログラミングと学習、フロントエンドに関する学習と作成を行った。その中で、楕円曲線班とそれぞれの進捗について共有を行った。しかし、時間内でのメッセージ通信アプリケーションの作成が困難と判断し、10 月の中旬に、暗号化を行う web アプリケーションの作成を行った。11 月には 10 月の活動の続きや、最終発表に向けた準備を行った。12 月の最終発表では作成した成果物の作成と、新たに学習しプログラミングを行った LWE 方式について発表をした。第 4 章で学習した内容や、作成した成果物につい具体的に記述した。12 月の中旬から 1 月中旬にかけて、最終報告書の作成などを行った。内容に関してはこの報告書に記述した。

（※文責: 樋口諒）

6.2　プロジェクト活動の成果

作成した成果

作成した成果物に関して、私たちはプロジェクト内では暗号アルゴリズムのプログラミングを目標とし、グループ内では格子暗号によってメッセージ通信が行えるアプリケーションの作成を目標として活動した。しかし、作成が間に合わず、急遽成果物の作成を暗号化と復号を行える web ページの作成に切り替え、作成を行い、成果物の作成を完了した。

知識に関する成果

私たちは先ほど記述したように、活動として、最初の活動で整数論に関する学習を行い、その後に、グループごとに文献を読みプロジェクト内で共有を行った。更に、知識を深め、夏休み中に暗号をプログラミングするための学習を行った。その後にプログラミングを行い成果物を作成した。この活動の中で、知識に関する成果として、暗号に関する知識のを得た。更に、その暗号をプログラミングし、成果物を作成するための知識を得た。具体的に得られた知識の成果は、近年の暗号の状況、暗号アルゴリズム、暗号アルゴリズムに関わる数学の知識、flask に関わる知識などを身につけることができた。

プロジェクト活動に関する成果

プロジェクト活動に関する成果として複数個挙げられる。

一つ目は、スケジューリングの重要性を理解した点である。私たちは、スケジューリングが不十分であったため、目標に及ばず、途中で目標の変更を行わざるを得なかった。このことから、スケジューリングすることの重要性を学んだ。

二つ目は、分担の重要性を理解した点である。目標が達せられなかった原因として分担が的確でなかった点が挙げられる。分担がもっと適切に行うことができたのならば、楽にプロジェクトを進められたのではないかと考えているため、分担の失敗から重要性を学ぶことができた。

三つ目は、共有の重要性を理解した点である。私たちは、時間が少ないことから、共有を行う時間をおろそかにしてしまったため、他の人進捗を理解していなかったため、上記で挙げたスケジューリングと分担が更に問題を抱えることになってしまった。このことから、共有の重要性を理解することができた。

私たちは以上の三つの点を、プロジェクトを行う上で理解することができた。これらの三つを理解したことが、プロジェクト活動を通じて得られた成果である。

（※文責: 樋口諒）

6.3　プロジェクト内の各人の役割

6.3.1　猪股莉記

前期の活動

プロジェクトリーダーとして、各グループの進捗管理や、他グループとの連携や連絡などの様々な事務的な活動を行った。グループ活動内では、格子暗号に関わる文献などを読み、グループメンバーに積極的に共有を行っていた。

後期の活動

前期に引き続き、プロジェクトリーダーとして、各グループの進捗管理や、他グループとの連携や連絡などの様々な事務的な活動を行った。グループ活動内では、LWE 方式のプログラミングを行うと同時に、格子暗号と LWE 方式に対する理解を深めていた。

（※文責: 樋口諒）

6.3.2 樋口諒

前期の活動

グループリーダーとして、グループ内での進捗管理などの活動を行った。グループ活動内では、格子暗号に関わる数学について学習を行い共有を行った。

後期の活動

前期に引き続き、グループリーダーとして、グループ内での進捗管理などの活動を行った。グループ活動内では、GGH 方式のプログラミングを行うと共に、数学的側面の学習を積極的に行った。

（※文責: 樋口諒）

6.3.3 砂子祐貴

前期の活動

前期では 1 年生の情報数学で使用した「離散数学への入門」に述べられている、RSA 暗号や共通鍵方式、公開鍵暗号方式をわかりやすくスライドでプロジェクトメンバーに説明を行った。RSA 暗号とは素因数分解の困難性を利用した暗号方式である。RSA 暗号が主に使われているところは SSL サーバー証明などである。共通鍵方式とは暗号化と復号するのに使う鍵が同じ鍵である暗号方式の事を指す。この暗号方式は 1 つの鍵が晒されることによって復号もできてしまうため、情報を安全に送ることが難しい。そして 6 月には「暗号理論のための格子数学」を学習し、暗号について、格子暗号について理解を深めた。そのほかにも公開鍵暗号方式についても理解を深めた。7 月ではサブポスターやポスター、スライドの作成を行った。夏休み中は主に javascript について学習し、後期に使用する予定だった Flask を扱えるようにした。グループ活動内で、ポスターなどの作成の中心として積極的な活動を行っていた。また、ポスターや報告書内での図の作成をすべて行った。

（※文責: 砂子祐貴）

後期の活動

後期では LWE 方式の格子暗号の暗号の一部を変更し、日本語を暗号化し復号できるようにした。まず入力された日本語を Unicode で 16 進数にし、それを bit 列にして保存した。そしてその bit 列を暗号化し、復号する。復号するときは bit 列をまず復号し、それを 16 進数にし、Unicode で日本語にした。そして時間の関係で断念したメール電子署名の web ページを作成した。断念した web ページも同様に Flask を使用して作成した。11 月は Flask の勉強と並行して断念した web

ページの作成を行っていた。Web ページを作成する際、Flask なぜ Flask を選んだのかというと、django は大規模な web ページを作るのに対して、Flask は拡張性が高く軽量であったからである。私たちが作成しようとしていたものは比較的小規模のものであり、格子暗号は他の暗号方式よりも計算が重くなってしまうため Flask という Web フレームワークを使用した。まとめとして、グループ活動内で、ポスターなどの作成の中心として積極的な活動を行っていた。また、web ページのフロントエンド部分や flask による暗号化プログラミングと javascript の結合を行った。さらに、時間の関係で断念したメッセージアプリケーションの作成を行っていた。

（※文責: 砂子祐貴）

付録 A　新規習得技術

公開鍵暗号方式, 格子暗号 (GGH 方式,LWE 方式), プログラミング言語 PARI-GP,Python,html,css

付録 B　活用した講義

情報ネットワーク, 情報数学, 線形代数学 I,II, データサイエンス入門

参考文献

[1] 経済産業省 (2022) 暗号強度要件（アルゴリズム及び鍵長選択）に関する設定基準. https://www.cryptrec.go.jp/list/cryptrec-ls-0003-2022.pdf (2023/07/14 アクセス)

[2] NIST(2020) Recommendation for Key Management. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r5.pdf (2023/07/14 アクセス)

[3] 清藤ら (2015) 量子コンピュータの解読に耐えうる「格子暗号」の最新動向 https://www.imes.boj.or.jp/research/papers/japanese/kk34-4-7.pdf(2023/07/19 アクセス)

[4] digicert(2023) 米国商務省標準化技術研究所（NIST）が耐量子暗号標準を発表：最新の状況. https://www.digicert.com/jp/blog/nist-pqc-standards-are-here(2023/8/26 アクセス)