Vol.2021-CSEC-95 No.7
Vol.2021-SPT-45 No.7
Vol.2021-EIP-94 No.7
2021/11/8

NTRU格子電子署名への中国剰余定理を用いたアグリゲート署名スキームの適用

安ヵ川 彩乃[1,a)]　佐藤 周行[1,b)]

概要：電子署名は電子文書の本人性や非改竄性を保証するために用いられている．量子コンピュータの台頭に対し，耐量子暗号を用いた電子署名方式への需要が高まっているが，耐量子暗号を用いた署名方式の実用化に向けた課題の 1 つとして署名長が挙げられる．この解決策として複数の署名を 1 つに集約し全体の署名長を小さくするアグリゲート署名があるが，格子署名方式ではアグリゲート手法が確立されていない．本研究では，中国剰余定理を用いた格子署名方式のアグリゲート署名スキームを提案する．中国剰余定理の条件を満たす格子生成方法を証明するとともに，既存の NTRU 格子署名方式への適用を検討した．

キーワード：アグリゲート署名，格子暗号，中国剰余定理

Applying aggregate signature scheme based on Chinese remainder theorem to NTRU lattice-based signatures

Abstract: Digital signature is designed to guarantee the origin and identity of the documents. There is an increasing demand for signature schemes using quantum resistant algorithms to prepare for the arrival of quantum computers. However, there are a few issues for application, typically data size. One of the solutions to this issue is the aggregate signature which gathers multiple signatures into one and make the data size smaller. However, the aggregation method has not been established for lattice-based signature schemes. In this paper, we propose a lattice-based signature aggregation method using Chinese remainder theorem, prove its correctness and examine the application on an existing NTRU lattice-based signature scheme.

Keywords: aggregate signature, lattice-based cryptography, Chinese remainder theorem

1. はじめに

インターネット上で契約や手続き，売買が行われるサービスが増える中で，今まで紙の書類に対して印鑑を押したりサインをしたりしたように電子文書が改ざんされていない正式なものであることの証明は不可欠である．電子文書においてこの役割を担う技術が電子署名である．電子署名とは，電子文書に対し署名を発行すると，署名と電子文書から確かに署名者本人による署名がされていることを他者が検証できるようなアルゴリズムである．しかし，単純な電子署名はユーザが署名に使う鍵が漏洩した場合のリスクや，電子文書に対し第三者から署名を得る際のリスク，署名鍵の更新・失効などの運用上の問題があり，ビジネスでの応用範囲は狭かった．

近年，今後台頭すると考えられている量子コンピュータに対し，耐量子性を持つ署名体系への需要も高まっている．格子問題を用いた暗号方式である格子暗号は，耐量子性の観点では「理論的には解読できるが，たとえ量子コンピュータを利用したとしても，現時点では効率的に解くアルゴリズムが見つかっておらず，解読に膨大な時間を要する演算処理が必要なため，事実上解読するのが難しい」という計算量的安全性を持つ暗号である．しかし，格子暗号を用いた電子署名は古典的な方式の署名と比較して署名長が大きくなってしまう問題があり，これを解決する手法として格子電子署名方式へのアグリゲート署名の適用が挙げられる．アグリゲート署名は異なる署名者がそれぞれ異なる書類に対して行った電子署名を 1 つの署名に集約する

[1] 東京大学
The University of Tokyo
[a)] ayanoy@satolab.itc.u-tokyo.ac.jp
[b)] schuko@satolab.itc.u-tokyo.ac.jp

Vol.2021-CSEC-95 No.7
Vol.2021-SPT-45 No.7
Vol.2021-EIP-94 No.7
2021/11/8

方式であり，検証者はそのまとめた署名を 1 回検証するだけで署名者全員の署名を検証することができる．量子コンピュータのよる攻撃を見据えた格子電子署名へのアグリゲート署名方式の研究開発は行われているが，安全性と利便性を兼ね備えた方式はまだ確立されていない．本稿では中国剰余定理を用いた格子暗号アグリゲート署名方式を提案し，既存の NTRU 格子電子署名方式に対してその手法を実装するために行った検討を記す．

本稿の構成は以下の通りである．第 2 章では，格子暗号アグリゲート署名の関連研究について述べる．第 3 章では中国剰余定理を用いた先行研究，第 4 章では第 3 章の先行研究の有効性の調査，及び条件式を満たす格子と基底の生成方法についての検討を記す．第 5 章にて提案手法による既存の NTRU 格子電子署名へのアグリゲート署名の実現方法を記す．第 6 章では第 5 章で検討した手法による実験と考察を述べ，第 7 章では本研究によって得られた Intersection 法の有効性及び既存の格子電子署名への適応についてまとめ，今後の課題を述べる．

2. 関連研究

現在米国国立標準技術研究所が行っている耐量子暗号技術標準化プロジェクトにおいて電子署名の標準化の候補に複数の格子署名方式が挙げられており [1]，実用化に向けた機能性の研究が増えている．複数の署名者による，それぞれ異なる書類に対する署名を 1 つにまとめるアグリゲート署名の応用研究は従来の古典的な電子署名方式では近年盛んであるが [2], [3]，耐量子暗号を用いた方式の研究はまだ少ない．これは格子暗号をベースとした場合は，ペアリング暗号で用いられる双線形写像の加法や乗法演算が成り立たず，アグリゲーション方法が確立していないためである．

格子署名のアグリゲーションに関する先行研究には 1 つの共通の格子から各署名者の署名鍵を生成する研究が多い [4], [5]．しかし，正当な署名者であれば同じ格子から鍵を得た他の署名者の署名を作成できてしまうといったセキュリティ上の欠点がある．また最近は格子暗号電子署名方式の標準化の候補になっている CRYSTALS-Dilithium [6] を簡略化した署名方式についてアグリゲート手法を提案している研究 [7] など，より実用化に向けた手法が求められる．

3. Intersection 法

3.1 格子暗号

$\mathbb{R}^n$ の任意の格子は，実ベクトル空間 $\mathbb{R}^n$ の離散部分群であり，ベクトル空間における基底から得られる整数係数線型結合の全体である．したがって典型的な n 次元の格子 Λ は，基底 $\{\boldsymbol{v_1}, \boldsymbol{v_2}, \ldots, \boldsymbol{v_n}\}$ を用いて次の式 1 のように表すことができる．ただし，1 つの格子を表す基底は一意ではない．

$$\Lambda = \{\sum_{i=1}^{n} a_i \boldsymbol{v_i} | a_i \in \mathbb{Z}\} \tag{1}$$

格子問題はベクトル空間上に規則正しく並んでいる点の集合である格子が与えられたときに，ある条件を満たす格子点（格子上の 1 つの点）を基底を用いて探索する問題の総称である [8]．短く直交に近い基底があれば様々な条件を満たす格子点を求められるが，長い基底からは探索は困難である．そして短く直交に近い基底から格子を生成することはできるが，与えられた格子からは長く平行に近い基底しか求められないため，格子生成に用いた基底を持っている場合のみ効率的に解くことができる．

主な格子暗号には，最近ベクトル問題（Closest Vector Problem; CVP）や，最短ベクトル問題（Short Vector Problem; SVP）を用いた手法がある．短整数解問題（Short Integer Solution; SIS）は SVP の 1 つであり，行列 $\boldsymbol{A} \in \mathbb{Z}_q^{n \times m}$，実数 β，素数 p について，次式を満たす非零ベクトル $\boldsymbol{e} \in \mathbb{Z}^m$ を求める問題である．

$$\boldsymbol{Ae} = \boldsymbol{0}\ (mod\ q)\ and\ |\boldsymbol{e}| \leq \beta \tag{2}$$

この問題では 1 つの格子に対する 2 つの基底について，短く直交に近い基底を署名（秘密）鍵，長い基底を検証（公開）鍵とする．Babai のアルゴリズム [9] より，署名鍵から式 2 を満たす短いベクトル $\boldsymbol{e}$ を署名として求めることができる．一方，検証鍵の基底を用いれば，求められた $\boldsymbol{e}$ が式 2 を満たすことが確認できるが，量子計算機を用いても検証鍵からは効率的に短く直交に近い基底や式 2 を満たす短いベクトルを求められないという困難性がある．

3.2 Intersection 法の定義

Boneh らによる Intersection 法は，中国剰余定理 [10] を元にした格子暗号の署名方式である [11]．加工したデータに対する署名と，用いた加工方法に対する署名を作成し，1 つの署名としてまとめて出力することができる．中国剰余定理より，2 つの異なる n 次元の格子 Λ_1, Λ_2 が $\Lambda_1 + \Lambda_2 = \mathbb{Z}^n$ を満たす時，2 つのベクトル $\boldsymbol{u}_1 \in \mathbb{Z}^n/\Lambda_1, \boldsymbol{u}_2 \in \mathbb{Z}^n/\Lambda_2$ について，

$$\begin{cases} \boldsymbol{\sigma} = \boldsymbol{u}_1\ mod\ \Lambda_1 \\ \boldsymbol{\sigma} = \boldsymbol{u}_2\ mod\ \Lambda_2 \end{cases}$$

を満たすベクトル $\boldsymbol{\sigma} \in \mathbb{Z}^n$ が格子 $\Lambda_1 \cap \Lambda_2$ を法として存在する．このことから，格子 Λ_1, Λ_2 上でそれぞれ作られた署名 $\boldsymbol{u}_1, \boldsymbol{u}_2$ を $\Lambda_1 \cap \Lambda_2$ を法とする 1 つのベクトル $\boldsymbol{\sigma}$ で表すことができる．この時 $\boldsymbol{\sigma}$ は格子 $\Lambda_1 \cap \Lambda_2$ の短い基底から求められる．このように格子暗号による署名方式で得た 2 個の異なる署名を，1 つの署名にまとめられることが示された．

3.3 Intersection 法による順序なし格子暗号アグリゲート署名

Intersection 法を $t(\geq 3)$ 個の格子に拡張できるという仮定を元に，アグリゲート署名に用いた Lu らの研究 [12] は，各署名者に対し，法 q においてランダムに生成された m 次元の格子をそれぞれ鍵として与える署名方式であり，格子は行列 $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$ をパリティ検査行列として式 3 のように定義される集合 $\Lambda_a^{\perp}(\boldsymbol{A})$ とする．

$$\Lambda_a^{\perp}(\boldsymbol{A}) = \boldsymbol{y} \in \mathbb{Z}^m : \boldsymbol{Ay} = \boldsymbol{0} \pmod q \tag{3}$$

q, m, セキュリティパラメタ n を入力とする確率論的アルゴリズム [13] はランダムな格子 $\Lambda_a^{\perp}(\boldsymbol{A})$ を生成し，その出力は行列 $\boldsymbol{A}$ とその格子の短い基底 $\boldsymbol{T}$ である．署名方式の全体の流れは次の通りである．

設定 鍵発行センタ（Key Generation Center; KGC）は t 人のユーザ u_i $(i = 1, \ldots, t)$ に対し，格子生成アルゴリズムによって出力した格子行列 $\boldsymbol{A}_i$ とその格子の短い基底 $\boldsymbol{T}_i$ をそれぞれ検証鍵，署名鍵として発行する．加えて，KGC は m 次元の格子 Λ_a とその基底 $\boldsymbol{T}_a$ をランダム生成し，次の連立方程式を満たすような t 個の格子 $\Lambda_1, \Lambda_2, \ldots, \Lambda_t$ を求める．

$$\begin{cases} \Lambda_1 + \Lambda_2 + \cdots + \Lambda_t = \mathbb{Z}^m & (4) \\ \Lambda_1 \cap \Lambda_2 \cap \cdots \cap \Lambda_t = \Lambda_a & (5) \end{cases}$$

署名 ユーザ u_i は，署名鍵 $\boldsymbol{T}_i$，署名鍵衝突困難性のあるハッシュ関数 H_1 を用いて，署名する文書 ϖ_i に対し次の式 6 の格子問題を満たす署名 $\boldsymbol{e}_i$ を求める．ただし s はガウシアンパラメタとする．

$$\boldsymbol{A}_i\boldsymbol{e}_i = H_1(\varpi_i) \pmod q \ and\ ||\boldsymbol{e}_i|| \leq s\sqrt{m} \tag{6}$$

検証 検証鍵 $\boldsymbol{A}_i$ を用いて確かに署名 $\boldsymbol{e}_i$ が式 6 を満たすことを検証する．

アグリゲーション KGC が生成した t 個の格子 $\Lambda_1, \Lambda_2, \ldots, \Lambda_t$ を用いると拡張した Intersection 法により，以下の連立方程式 7 をベクトル $\boldsymbol{e}$ について解くことができる．$\boldsymbol{e}$ は格子 Λ_a を法とし，その基底 T_a より求められる．

$$\begin{cases} \boldsymbol{e}_1 = \boldsymbol{e}\ mod\ \Lambda_1 \\ \boldsymbol{e}_2 = \boldsymbol{e}\ mod\ \Lambda_2 \\ \quad \vdots \\ \boldsymbol{e}_t = \boldsymbol{e}\ mod\ \Lambda_t \end{cases} \tag{7}$$

このように得られたアグリゲート署名 $\boldsymbol{e}$ は文書列 $(\varpi_1, \varpi_2, \ldots, \varpi_t)$ の署名であり，全ての署名者の検証鍵に対応する．

アグリゲート署名の検証 H_1 と異なる衝突困難性のあるハッシュ関数 H_2 と各署名者の検証鍵を用いて式 8 が成立することで，有効であると示される．

$$\begin{aligned} &H_2(H_1(\varpi_1), H_1(\varpi_2), \ldots, H_1(\varpi_t)) \\ &= H_2(\boldsymbol{A}_1(\boldsymbol{e}\ mod\ \Lambda_1)\ mod\ q, \ldots, \boldsymbol{A}_t(\boldsymbol{e}\ mod\ \Lambda_t)\ mod\ q) \end{aligned} \tag{8}$$

この署名方式は式 6, 8 の個々の署名およびアグリゲート署名の検証アルゴリズムの正当性が証明されている．加えて，SIS 問題の困難性が適応的選択文書攻撃に対して存在的偽造不可な安全性（EUF-CMA 安全性）を持ち，個々の署名の有効性がアグリゲート署名の有効性の必要十分条件であることが示されている．またこの手法では，先行研究 [4], [5] の課題であった署名鍵の偽造を防いでいる．

4. Intersection 法の有効性の検討

4.1 行った検討の目的と概要

Lu らの研究 [12] は格子問題を用いたアグリゲート署名方式の課題であった正当な署名者による偽造を Intersection 法によって防ぐことができる．加えて，他の格子問題を用いた手法に比べて署名長が小さいという利点があるが，Intersection 法の 3 個以上の格子への拡張についての証明がされていない．そのため，Intersection 法の有効性が示せれば，格子暗号を用いた順序なしアグリゲート署名のスタンダードとなりうる．したがって拡張された Intersection 法の有効性を調べるために，中国剰余定理を用いて問題の再定義を行い，3 個以上の格子への拡張の数学的な正当性と，条件式を満たす格子と基底の生成方法について確かめた．

4.2 拡張された中国剰余定理による問題の再定義

中国剰余定理 [10] は整数とその剰余に関する定理から，一般の単位元を持つ環とそのイデアルについて拡張されている．単位元を持つ環 R とその部分集合である k 個のイデアル $I_1, I_2, \ldots, I_k$ について，どの 2 つのイデアルも互いに素である，すなわち式 9 を満たすとする．

$$I_i + I_j = R\ (i \neq j) \tag{9}$$

このとき中国剰余定理では任意の $a_1, a_2, \ldots, a_k \in R$ について，連立方程式 10 を満たす $x \in R$ がイデアル $I = \bigcap_{i=1}^{k} I_i$ を法として存在すると示されている．

$$\begin{cases} x \equiv a_1 \pmod{I_1} \\ x \equiv a_2 \pmod{I_2} \\ \quad \vdots \\ x \equiv a_k \pmod{I_k} \end{cases} \tag{10}$$

よって Lu らの研究は，環 $\mathbb{Z}^m$ のイデアルである格子 Λ_i $(i = 1, \ldots, t)$ とベクトル $\boldsymbol{e}_i$ $(i = 1, \ldots, t)$ について拡張

された中国剰余定理を解く問題に帰着する．Intersection 法を提示した Boneh らの研究では $t = 2$ について成立することが証明されているが，$t \geq 3$ での証明はオープンな問題となっている．Lu らの研究では式 4, 5 を満たすことで中国剰余定理の条件を満たすとしているため，これらの式の十分性を調べた．

4.3 式 4 の正当性について

Lu らの研究では環 $\mathbb{Z}^m$ に式 4 を満たすイデアルとして格子 Λ_i $(i = 1, 2, \ldots, t)$ を用いている．第 4.2 節より，Intersection 法を 3 個以上の格子に拡張する場合には，t 個の格子 Λ_i $(i = 1, 2, \ldots, t)$ が互いに素なイデアルである，すなわち式 9 より，次の式 11 を満たす必要がある．

$$\Lambda_i + \Lambda_j = \mathbb{Z}^m \ (i \neq j) \tag{11}$$

よって，式 4 は必要条件ではあるが条件が足りないことが示された．

また，すべての i, j について式 11 を満たすような格子生成アルゴリズムも必要である．利用できるアルゴリズムとして素イデアル格子の生成アルゴリズムが挙げられる．素イデアルとは，イデアルのうち $x, y \in R, xy \in I$ のとき，$x \in I$ または $y \in I$ を満たすものである．素イデアルは素数をもとに生成するため，同じ環の任意の異なる 2 つの素イデアルは式 11 を満たす．このことから 2 個の格子の Intersection 法を提示した Boneh らの研究では，次元数やパラメタを入力とし素イデアルな格子とその生成元を出力するアルゴリズム PrinceGen ([14] Theorem 3.1) を仮定して用いているが，証明はされていない．

4.4 式 5 の正当性と格子生成方法について

式 5 の積集合について，中国剰余定理や Intersection 法の過程では各イデアルからその積集合を求めているが，Lu らの研究では，逆に Intersection 法に基づいて得られる式 7 のベクトル $\boldsymbol{e}$ の法となる格子 Λ_a とその基底 $\boldsymbol{T}_a$ を先に生成し，式 4, 5 をともに満たすような t 個の格子 Λ_i $(i = 1, 2, \ldots, t)$ を求めている．これはアグリゲート署名 $\boldsymbol{e}$ を求めるとき基底 $\boldsymbol{T}_a$ が用いられるが，一度生成された格子について短い基底を求めることは格子暗号署名方式の安全性を保証する格子問題に当たり困難な計算であるからである．このような理由から先にアグリゲート署名の法となる格子とその基底を定める工程が行われているが，上記の条件を満たすような t 個の格子 Λ_i $(i = 1, 2, \ldots, t)$ の生成方法は明示されていない．

4.5 本研究における仮定

本研究では 1 つの施設や組織内のシステムという条件下で，あらかじめ新しい鍵が発行される際には KGC がシステム内の既存の検証鍵と新しい検証鍵が互いに素であることを確認することとする．このとき組織内限定のシステムとすることで鍵の総数を抑えられるため，既存の検証鍵に対し互いに素な新しい検証鍵が高次元にて必ず存在すると仮定する．このような条件で実装を目指すことで，式 4 を満たす格子を既存のアルゴリズムで生成することができる．

また，本手法を第 5.1 節にて後述する格子と多項式としての特徴を併せ持つ NTRU 格子暗号を用いた電子署名方式へ実装する．このとき多項式における中国剰余定理より，式 5 を満たすアグリゲート署名を拡張ユークリッドの互除法 [15] から求めることができる．これにより第 4.3 節で言及した各格子の生成とアグリゲート署名生成の順序に関する課題を解決することができる．

5. NTRU 格子電子署名方式 FALCON へのアグリゲート署名の実現

5.1 NTRU 格子電子署名方式 FALCON の概要

FALCON [16] は NTRU 格子暗号 [17] を用いた電子署名であり，NTRU は自然数 N と素数 q を法とする整数が係数である多項式環 $R = \mathbb{Z}_q[x]/(x^N - 1)$ で定義される．多項式 $f(x), g(x) \in R$ を署名鍵とすると，検証鍵は $h(x) = f(x)^{-1} \cdot g(x)$ と定義され，式 12 で表される $2N$ 次元の NTRU 格子に基づく格子署名方式とみなすことができる．

$$M_h^{NTRU} = \begin{pmatrix} I & \boldsymbol{h} \\ 0 & qI \end{pmatrix} \tag{12}$$

また NTRU 格子署名の特徴は，耐量子署名の中で署名長が短いことである．FALCON では署名鍵によって作成された署名は検証鍵と同様に N 次元の多項式であり，係数は 0 を中心としたガウス分布に従う．FALCON は耐量子暗号技術標準化プロジェクトの最終選考候補となっている電子署名方式である．

5.2 FALCON 上でのアグリゲート署名の実現

本研究ではすべての検証鍵があらかじめ互いに素な多項式であることを仮定している．t 個の署名のアグリゲーションを行う時，$t-1$ 個の署名を集約したアグリゲート署名に新たに 1 つ署名を加える形で行うことで，拡張ユークリッドの互除法より t 個の署名のアグリゲート署名が求められる．実験手法は以下の通りである．

設定 KGC は $t-1$ の署名をまとめたアグリゲート署名 sig'_{agg} の検証鍵 vk'_{agg} と，新しい署名を作成するユーザ u_t の検証鍵 vk_t について，ユークリッドの互除法より最大公約数を求め，互いに素であることを確認し，ユーザ u_t の鍵を発行する．

署名・検証 ユーザ u_t の署名鍵 sk_t を用いて任意の書類への FALCON スキームの署名 sig_t を作成し，vk_t を用いて検証する．

Vol.2021-CSEC-95 No.7
Vol.2021-SPT-45 No.7
Vol.2021-EIP-94 No.7
2021/11/8

```
REQUIRE vk_1, vk_2 ∈ R
ENSURE A · vk_1 + B · vk_2 = 1
quo = [0, 0, vk_1 // vk_2]
rem = [vk_1, vk_2, vk_1 mod vk_2]
while rem[i] ≠ 0 do
  i ← i + 1
  qu, r ← rem[i − 2]/rem[i − 1]
  quo[i] ← qu, rem[i] ← r
end while
A ← 1
B ← −quo[i − 1] mod q
for j in [i − 1..2] do
  tmp ← B
  B ← A − quo[j] · B mod q
  A ← tmp
end for
lc ← rem[i − 1][0]
lcinv ← lc^{−1}
A ← A · lcinv mod q, B ← B · lcinv mod q
return A, B
```

図 1 $A \cdot vk_1 + B \cdot vk_2 = 1$ を満たす A, B を求めるアルゴリズム．
Fig. 1 Algorithm to calculate A, B that satisfies $A \cdot vk_1 + B \cdot vk_2 = 1$.

アグリゲーション 拡張ユークリッドの互除法を用いて，**図 1** のアルゴリズムより式 13 を満たす多項式 A, B を求める．

$$A \cdot vk'_{agg} + B \cdot vk_t \equiv 1 \ (mod \ \ q) \tag{13}$$

A, B を用いて式 14 よりアグリゲート署名 sig_{agg} を $mod\ (vk'_{agg} \cdot vk_t)$ において求める．

$$sig_{agg} \equiv A \cdot vk'_{agg} \cdot sig_t + B \cdot vk_t \cdot sig'_{agg} \tag{14}$$

また $vk_{agg} = vk'_{agg} \cdot vk_t$ を生成されたアグリゲート署名 sig_{agg} の検証鍵とする．

アグリゲート署名の検証 sig_{agg} をユーザ u_i の検証鍵 vk_i で割った余り sig_i をそれぞれ求め，これについて vk_i を用いて検証する．

このように，本手法では公開されている署名と検証鍵のみを用いてアグリゲーションを行っており，署名者の署名鍵に関する情報が得られることはない．よって，第三者によるアグリゲーションが可能である．

6. 実験と考察

7. 実験環境

本実験は FALCON の Python 用パッケージ [18] を使用して Python 3.8 で本手法によるアグリゲート署名の実装を行った．各パラメタは署名と検証鍵の次元 $n = 128, 512, 1024$，多項式環の法 $q = 12 \times 1024 + 1$ とした．

7.1 実験結果

実験を実行した結果として，検証鍵及び署名する文書が異なる t 個の署名から 1 個のアグリゲート署名が正しく生成できることを確認した．n を FALCON の設定に則って変化させたところ，t 個の署名のアグリゲート署名の長さについて**表 1** の結果を得た．本手法によって，t 個の署名の合計の次元数を t だけ小さくなることが示された．

表 1 アグリゲート署名の署名長（t: 署名数，n: 署名に用いた多項式の次数）．
Table 1 Length of aggregate signatures. (t: number of individual signatures, n: degree of the signature polynomial).

scheme	signature length
FALCON only	tn
FALCON+Ours	$tn - t$

一方で本手法ではアグリゲート署名の検証を行う際にアグリゲート署名から各署名を求めるが，sig_i は各署名 sig_i を検証鍵 vk_i で割った商を c_i として式 15 のように求められる．

$$sig_i = sig_{agg} \ mod \ vk_i + c_i \cdot vk_i \tag{15}$$

検証鍵と署名の次数が等しいことから，sig_i の計算にトラップドアとして $c_i \ (i = 1, \ldots, t)$ の情報が必要であり，署名に加えて保存する必要がある．

7.2 考察

第 5.3 節にて得られたアグリゲート署名長は多項式の中国剰余定理において求められるアグリゲート署名 sig_{agg} が検証鍵，すなわち元の 2 つの署名の検証鍵の積を法としてただ 1 つ存在することから，次の式より求められる．

$$\begin{aligned}
deg(vk_{agg}) &= deg(vk_1 \cdot vk_2) \\
&= deg(vk_1) + deg(vk_2) - 1 \\
&= 2n - 1 \\
deg(sig_{agg}) &= deg(vk_1 \cdot vk_2) - 1 \\
&= 2n - 2 \\
&\vdots \\
deg(sig_{agg}) &= deg(vk_{agg} \cdot vk_t) - 1 \\
&= tn - t
\end{aligned}$$

また，アグリゲート署名から t 個の署名を求めるために新たなトラップドアが t だけ必要となるため，全体としてデータサイズがアグリゲーション前と等しくなってしまっている．これは検証鍵と署名の次数が等しいことに起因する．署名の次数を小さくすることはセキュリティ上困難であり，トラップドア情報を圧縮することで解決することができると考えられる．

8. おわりに

本稿では，最近の耐量子暗号を用いた電子署名方式に対するアグリゲート署名に注目し，中国剰余定理を用いたアグリゲート署名方式の有効性についての検討と提案手法の NTRU 格子電子署名方式の FALCON への実装をまとめた．第 4 章の検討結果から，再定義した問題は論文内の条件式やアルゴリズムでは解くことができず，Intersection 法による順序なし格子暗号アグリゲート署名の先行研究は未完であると示すことができた．さらに，格子における中国剰余定理の条件を満たすような仮定を置くことで，第 5 章の FALCON への実装において全体の署名長を小さくすることができた．

今後は，本手法でアグリゲート署名の検証時に必要になる各署名のトラップドア情報について，圧縮する方法を検討する．また生成されるアグリゲート署名の特徴に適した圧縮アルゴリズムを用いることで更なる署名の短縮化を目指す．

参考文献

[1] Post-Quantum Cryptography — CSRC (online), available from ⟨https://csrc.nist.gov/projects/post-quantum-cryptography⟩ (2021.10.5).
[2] 矢内直人，岡田雅之，岡村真吾，et al.：アグリゲート署名を用いた BGPsec AS_Path 検証手法の提案と実装評価，コンピュータセキュリティシンポジウム 2018 論文集，Vol.2018，No.2，pp.333–340(2018).
[3] Li, X., Liu, S., Wu, F., Kumari, S. and Rodrigues, J. J. P. C.: Privacy Preserving Data Aggregation Scheme for Mobile Edge Computing Assisted IoT Applications, *IEEE Internet of Things Journal*, Vol.6, No.3, pp.4755–4763(2019).
[4] Jing, Z.: An efficient homomorphic aggregate signature scheme based on lattice, *Mathematical Problems in Engineering*, Vol.2014(2014).
[5] Zhang, P., Yu, J. and Wang, T.: A homomorphic aggregate signature scheme based on lattice, *Chinese Journal of Electronics*, Vol.21, No.4, pp.701–704(2012).
[6] Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schwabe, P., Seiler, G. and Stehlé, D.: CRYSTALS-Dilithium: A Lattice-Based Digital Signature Scheme, *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp.238–268(2018).
[7] Boudgoust, K. and Roux-Langlois, A.: Compressed linear aggregate signatures based on module lattices, *IACR Cryptol. ePrint Arch.*, Vol.2021, p.263(2021).
[8] Regev, O.: Lattice-based cryptography, In *Annual International Cryptology Conference*, pp.131–141, Springer(2006).
[9] Babai, L.: On Lovász' lattice reduction and the nearest lattice point problem, *Combinatorica*, Vol.6, pp.1–13(1986).
[10] Pei, D., Salomaa, A. and Ding, C.: *Chinese remainder theorem: applications in computing, coding, cryptography*, World Scientific(1996).
[11] Boneh, D. and Freeman, D. M.: Homomorphic signatures for polynomial functions, In *Advances in Cryptology – EUROCRYPT 2011*, pp.149–168, Springer(2011).
[12] Lu, X., Yin, W., Wen, Q., Jin, Z. and Li, W.: A lattice-based unordered aggregate signature scheme based on the intersection method, *IEEE Access*, Vol.6, pp.33986–33994(2018).
[13] Gentry, C., Peikert, C. and Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions, In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pp.197–206, Association for Computing Machinery(2008).
[14] Smart, N. P. and Vercauteren, F.: Fully homomorphic encryption with relatively small key and ciphertext sizes, In *Public Key Cryptography – PKC 2010*, pp.420–443, Springer(2010).
[15] Knuth, D. E.: *Art of computer programming, volume 2: Seminumerical algorithms*, Addison-Wesley Professional(2014).
[16] Fouque, P. A., Hoffstein, J.,Kirchner, P., Lyubashevsky, V.,Pornin, T., Prest, T., Ricosset, T., Seiler, G., Whyte, W. and Zhang, Z.: Falcon: Fast-Fourier lattice-based compact signatures over NTRU, *Submission to the NIST's post-quantum cryptography standardization process*, Vol.36(2018).
[17] Hoffstein, J., Pipher, J. and Silverman, J. H.: NTRU: A ring-based public key cryptosystem, In *Algorithmic Number Theory*, pp.267–288, Springer(1998).
[18] Prest, T.: A python implementation of the signature scheme Falcon (online), available from ⟨https://github.com/tprest/falcon.py⟩ (2021.10.5).