

全同态加密算法 FINAL 自举模块的优化设计

杨亚涛^{1,2}, 张思瑶¹, 陈亮宇¹, 刘国鑫²

¹ 北京电子科技学院电子与通信工程系 北京 中国 100070

² 西安电子科技大学通信工程学院 西安 中国 710071

摘要 构建基于格问题的密码系统已成为当前后量子密码领域中的一项关键技术,特别是基于 NTRU 问题构建的高效全同态加密方案,因其在性能和安全性方面的显著优势,受到了学术界的广泛关注。然而,当前的研究表明,在实现同态操作的过程中,现有的方案普遍需要对 NTRU 问题中的参数进行“过度拉伸”,这一操作会增加方案受到格基攻击的风险。为了解决 NTRU 参数过度拉伸所引发的安全性问题,同时进一步提高自举模块的运算速度,本文针对基于 NTRU 和 LWE 问题构造全同态加密方案 FINAL,提出一种利用 NTWE 问题的特性对 FINAL 方案自举模块中密钥切换部分进行优化的方案。与仅基于 NTRU 和 LWE 问题的 FINAL 方案相比,本文方案具有更小、更紧凑的结构,同时在参数选择方面还提供了更强的灵活性。安全性方面,在 NTRU 问题的决策性变体假设下,通过设置低于 NTRU 问题“疲劳点”的参数,本方案证明了其抵抗选择明文攻击的安全性。通过运算性能分析,本文提出的自举算法运算速度比 FINAL 方案快 17%,比 TFHE 算法快 39%。此外,与 FINAL 方案相比,本文方案缩减了密钥的长度,降低至 37.4MB。在保持高效性的同时,本文进一步提升了方案的灵活性和安全性,具有较好的应用前景。

关键词 全同态加密; 自举; 密钥切换; NTRU; LWE; TFHE

中图法分类号 TP309.7 DOI 号 10.19363/J.cnki.cn10-1380/tn.2026.05.06

Optimization on Bootstrapping Module in FINAL Fully Homomorphic Encryption Algorithm

YANG Yatao^{1,2}, ZHANG Siyao¹, CHEN Liangyu¹, LIU Guoxin²

¹ Department of Electronic and Communication Engineering, Beijing Electronic Science and Technology Institute, Beijing 100070, China

² School of Telecommunication Engineering, Xidian University, Xi'an 710071, China

Abstract Constructing cryptographic systems based on lattice problems has become a key technology in the current field of post-quantum cryptography. Notably, the development of efficient fully homomorphic encryption (FHE) schemes, particularly those constructed over the Number Theory Research Unit (NTRU) problem, has garnered widespread attention from the academic community due to their significant advantages in performance and security. However, most of the existing proposals need the so-called 'overstretched' parameters of NTRU problem to enable homomorphic operations, thereby increasing the risk of the system being subjected to lattice-based attacks. To address the security issues arising from the overstretching of NTRU parameters and to further accelerate the computational speed of the bootstrapping module, an optimization scheme has been proposed for the key switching part within the FHE scheme known as FINAL, which is constructed based on the NTRU problem and Learning with Errors (LWE) problems. And this optimization leverages the characteristics of the NTWE problem, which can be seen as a natural combination of the NTRU and module-LWE problems. The proposed schemes based on the NTWE problem have a smaller and more compact structure compared to schemes based solely on NTRU problem. Furthermore, the parameter selection in NTWE-based schemes offers greater flexibility compared to schemes based solely on LWE problem. In terms of security, the scheme is proven to be resistant to the chosen-plaintext attacks under a standard hybrid argument of the decisional NTRU problem by setting parameters that are below the 'fatigue point'. A computational performance analysis has revealed that the bootstrapping algorithm, which has been optimized, is operates at a speed 17% faster than the FINAL scheme and 39% faster than the TFHE scheme. In addition, the proposed scheme reduces the key length from 71 MB to 37.4 MB. By reducing the key length while simultaneously enhancing flexibility and security, the proposed scheme exhibits promising prospects for practical applications.

Key words fully homomorphic encryption algorithm; bootstrapping; key switching; number theory research unit; learning with errors; fully homomorphic encryption over the torus

通讯作者: 杨亚涛, 博士, 教授, 博导, Email: yy2008@163.com。

本课题得到国家自然科学基金资助项目(No. 62476013); 中央高校基本科研业务费资助(No. 3282025039, No. 3282024052)。

收稿日期: 2024-05-10; 修改日期: 2024-08-30; 定稿日期: 2026-04-09

1 引言

在信息时代,数据的价值和重要性日益凸显,随之而来的数据安全和隐私保护问题也变得尤为重要。随着大数据、云计算以及物联网技术的快速发展,对数据进行高效、安全的处理和分析的需求越来越迫切。在这样的背景下,全同态加密(Fully Homomorphic Encryption, FHE)技术应运而生,其可以在不暴露明文的情况下对加密状态下的数据进行处理,该技术提高了数据使用的安全性,也保证了数据的隐私性和完整性。

基于格问题的全同态加密技术是一个富有前景的研究领域。在 2009 年, Gentry^[1]最初提出的全同态加密方案是基于代数数域理想格的性质,随后基于格的代数结构的加密方案不断涌现,包括基于 LWE(Learning With Errors)^[2]、Ring-LWE(Learning With Errors over Rings, RLWE)^[3]、NTRU(Number Theory Research Unit)^[4]和近似整数问题的全同态加密(Fully Homomorphic Encryption, FHE)方案^[5]。其中基于 NTRU 问题的方案较为高效,与基于 RLWE 方案的密文使用一对多项式表示相比,基于 NTRU 方案的密文仅用单个多项式即可表示。因此,基于 NTRU 问题的方案大幅降低了内存需求和运行时间。

2013 年, Bos 等人首次提出了一种基于 NTRU 问题的全同态加密方案,命名为 YASHE^[6],该方案相较于其他同类型加密方案效率更高。然而研究发现,该方案在 NTRU 参数设定上存在“过度拉伸”的现象^[7],引发了其他学者对其安全性的质疑。在 YASHE 方案中,整数的模数 q 相对于多项式环的次数 n 呈指数级增长,但当通过固定 n 值同时增加 q 值的方法来“拉伸”参数时,攻击者可以利用 NTRU 的密集子格结构来发起攻击。因此,基于 NTRU 问题构建全同态加密方案遇到了阻碍。初始版本的攻击是利用了 NTRU 的子域特性来降低搜索格的维数,而后续的研究^[8]揭示了这种攻击实际上是利用了 NTRU 格内的密集子格。由于难以准确评估密集子格攻击对 NTRU 安全性的具体影响,基于 NTRU 问题方案的安全性依旧难以准确判断。2019 年, Genise 等^[9]提出了一种基于矩阵 NTRU 问题的自动机同态加密方案,然而 Lee 等^[10]在 2020 年发现该方案容易遭受到子格攻击的威胁。在此期间,有多位学者利用不同的技术,针对 NTRU 问题的特性提出了加密方案^[11-13],推动了 NTRU 在全同态加密领域的应用。最终, Ducas 等人^[14]提出了一种方法,可以计算出在任何给定 q 值下攻破 NTRU 问题所需的具体成本。至此,对

NTRU 问题的安全性随着参数 q 的增加而降低这一现象有了更深入的理解。上述研究为构建基于 NTRU 问题的全同态加密方案提供了坚实的基础。

作为一种基于环的公钥密码系统, NTRU 问题在 28 年前被引入^[15],而其安全性分析通常是依赖于解决基于 NTRU 格问题的效率。LWE 问题与格问题的关联更为紧密,该问题由 Regev^[16]在 2005 年提出。基于 LWE 问题的密码系统与基于 NTRU 问题的密码系统在参数设置上具有相似性,而其安全性分析则主要依赖于解决 LWE 实例中特定的格问题的效率。在 LWE 问题的不同结构化版本中, RLWE 和 MLWE(Module Learning With Errors)问题在性能上表现较优。RLWE 问题与 NTRU 问题的构造原理相似,其本质是对应了 NTRU 问题的非齐次版本。而 MLWE 问题可以视为 RLWE 问题的一种非完全结构化变体,其构造思路也与 NTRU 问题相似。尽管已经有部分研究将 LWE 的困难性与 NTRU 的困难性相结合,并据此提出了新的加密方案^[17-19],但这些研究文献未能直接阐明加密系统在参数化后的具体安全性。基于对 NTRU 与 LWE 这两个问题的深入理解, Gartner 在 2022 年提出了 NTWE 问题^[20],该问题将 LWE 的困难性与 NTRU 的困难性相结合,并同样对应于 q 元格中的困难问题,其困难度评估与 NTRU 问题及 LWE 问题的评估方法相似。

综上所述,格密码学为后量子密码学的发展奠定了重要的理论基础。其中, NTRU 和 LWE 作为该领域的两个核心问题,深入理解它们的特性,准确评估它们的具体安全性,对进一步推动该领域的研究至关重要。NTWE 问题的提出,实际上是对上述两个核心问题的融合,并有望推动基于格问题的全同态加密技术取得新的突破。

在 2022 年, Bonte 等人^[21]运用自举^[22]技术,构建了一种基于 NTRU 问题的全同态加密方案,并将其命名为 FINAL(Faster FHE Instantiated with NTRU and LWE)。该方案的构建,受到了 Micciancio 等人^[23]在 2021 年对 FHEW(Bootstrapping Homomorphic Encryption in Less Than a Second)^[24]与 TFHE(Fully Homomorphic Encryption over the Torus)^[25]两种全同态加密方案进行比较分析的启发,他们的分析指出两种方案性能上存在差异的根本原因在于采用了不同的自举算法。此外, TFHE 在密钥以二进制形式表示时展现出更优的运算性能,而当密钥表示扩展至三元或更高维数时, FHEW 的性能则显著超越了 TFHE。在内存使用方面, TFHE 所需的自举密钥长度明显小于 FHEW。针对 TFHE 方案的性能优化,已有

学者提出了若干方案^[26-28], 特别是针对其可编程自举程序的优化。基于上述研究成果, Bonte 等人以 FHEW 方案为基础, 对 TFHE 方案进行扩展以及优化, 巧妙地融合了两种加密方案的优势, 设计出了 FINAL 方案, 该方案采用了 NTRU 矩阵构建基础框架, 并基于标准 NTRU 问题构建了类 GSW(Gentry-Sahai-Waters)方案, 进而使得 FHEW 的框架也能适用于 NTRU 框架。FINAL 的基本方案具有快速自举的特性, 其运行时间与该类型中相对高效的 TFHE 方案接近。由于 FINAL 方案中参数的选取有效避免了 NTRU 问题的“过度拉伸”现象, 因此可以基于 NTRU 问题来构建具有竞争性的全同态加密方案, 同时也能保证数据的安全性和完整性。此外, FINAL 方案还设计了一种结合 LWE 问题和 NTRU 问题的 NGS(NTRU-GSW-like encryption Scheme)方案, 相较于 TFHE 方案, 其自举算法计算速度更快, 并且密钥长度显著缩短, 这些改进增强了全同态加密方案的实现效率以及实用性。上述研究成果对于深入研究全同态加密算法中自举模块算法的特性及其发展趋势具有重要意义, 为实现算法的进一步优化提供了新的思路和方法。

本文的主要贡献如下:

(1) 提出了一种利用 NTWE 问题对全同态加密算法 FINAL 进行优化的方案。本文对 FINAL 算法中自举模块的密钥切换部分进行了研究, FINAL 中 NGS 方案的自举模块设计结合了 LWE 与 NTRU 问题, 与 NTWE 问题有着相似的构造思路。因此本文利用 NTWE 问题, 重新设计了 FINAL 方案中自举模块的密钥切换部分, 并替换了原方案的切换密钥, 进而缩短了切换密钥的长度, 提高了灵活性和安全性, 同时也实现了方案性能与资源利用的优化。

(2) 对优化后的全同态加密算法进行了全面的性能分析。首先, 本文通过选择特定的参数对优化后算法的安全性进行了分析。随后, 本文分析了算法的运算性能, 验证了该方案能够在合理时间范围内有效地执行常用的加法和乘法运算。最后, 将本文优化后的方案与全同态加密领域中相对高效的 TFHE 以及 FINAL 方案进行了对比分析, 在设置适合参数的情况下, 对比切换密钥长度、自举密钥长度、 R_Q 的乘法次数、快速傅里叶变换(Fast Fourier Transform, FFT)次数以及运行时间, 得到具体优化后的结果: 我们的密文自举算法的运算速度比 TFHE 算法快 39%, 比 FINAL 算法快 17%, 密钥长度相较于 TFHE 方案和 FINAL 方案都有所缩减。经过本文方案优化后能有效抵御选择明文攻击, 其中的密钥交换算法

可以大幅降低自举模块的计算开销, 从而有效提升了方案的整体运算效率。

2 基础知识

2.1 分布

2.1.1 离散高斯分布

离散高斯分布^[29]是定义在整数集 $\mathbb{Z}$ 上的概率分布, 是一种在离散空间中模拟连续分布的方法, 该分布对集合 $\mathbb{Z}$ 中的每个元素采用与其在实数集中高斯分布下的概率质量函数 (Probability Mass Function, PMF) 值以成比例的概率进行抽样。离散高斯分布用于生成随机样本或进行随机抽样, 以此保证加密过程的安全性。首先我们将高斯函数定义如下:

$$\rho_{\sigma,c}(x) = \exp\left(-\frac{|x-c|^2}{2 \cdot \sigma^2}\right)$$

其中, $\sigma, c \in \mathbb{R} > 0$, σ 为分布的标准差, c 为分布的均值, 进而可以得出: $\rho_{\sigma,c}(\mathbb{Z}) = \sum_{i=-\infty}^{\infty} \rho_{\sigma,c}(i)$ 。具有标准差 σ 和均值 c 的离散高斯分布是在整数集 $\mathbb{Z}$ 上的分布, 其概率密度函数为 $\rho_{\sigma,c}(x)/\rho_{\sigma,c}(\mathbb{Z})$, 其中 $x \in \mathbb{Z}$ 。如果 $c=0$, 则可以用 χ_{σ} 表示这个分布。

2.1.2 Sub 高斯分布

格密码学中一个非常重要的研究对象是 Sub 离散高斯分布^[29], 本文将进一步阐述 Sub 高斯分布的定义: 在 $\mathbb{R}$ 上的随机变量 V 是 α -Sub 高斯的, 当其满足以下关系式:

$$E[\exp(t \cdot V)] \leq \frac{1}{2} \exp(\alpha^2 \cdot t^2)$$

其中, $\forall t \in \mathbb{R}$ 。从定义中可知, $\text{Var}(V) \leq \alpha^2$ 。

接下来将其定义拓展到向量, 即若 x 是 $\mathbb{R}^n$ 上的离散随机向量, 设 x 的分量为 x_i , 则称 x_i 是 α_i -Sub 高斯, 称 x 是 β -Sub 高斯分布, 其中 $\beta = \max_{i \in [n]} \alpha_i$ 。

2.2 分解

对于固定的整数 q 和 B , 设 $\ell := \lceil \log_B q \rceil$, 并且定义:

$$g_{q,B} := (B^0, \dots, B^{\ell-1})$$

当 q 和 B 在上下文中明确之后, 可简写为 g 。对于任何 $k \in \mathbb{Z}_q$, 我们用 $[-q/2, q/2)$ 中的任一整数来表示 k , 并定义它在以 B 为底的有符号分解为

$$g^{-1}(k) = (k_0, \dots, k_{\ell-1})$$

对于每个整数 $|k_i| \leq B/2$, $i \in [\ell]$ 可得

$$g^{-1}(k) \cdot g = k$$

对于任何 $f \in R_Q$, 定义:

$$g^{-1}(f) := \sum_{i=0}^{N-1} g^{-1}(f_i) X^i$$

很显然如下等式成立:

$$g^{-1}(f) \cdot g = \sum_{i=0}^{N-1} g^{-1}(f_i) \cdot g \cdot X^i = \sum_{i=0}^{N-1} f_i \cdot X_i = f$$

其中, 数字分解 g^{-1} 可以是确定的, 也可以是随机的^[30-31]。

2.3 NTRU 问题

通常采用三元密钥来对 NTRU 问题进行实例化, 在实例化过程中, 密钥是通过从 $\{-1, 0, 1\}$ 上的特定分布中提取并生成的, 其中 0 出现的概率是 1/2, 1 和 -1 出现的概率则各为 1/4。这种分布近似于标准差 $\sigma = 1/\sqrt{2}$ 的离散高斯分布。

根据文献[32]可以将 NTRU 问题划分为反循环版本和矩阵版本。每个版本均包含了计算型和决策型变体。

NTRU: 设 $N > 0$, $Q > 1$ 为整数, 且 $R := Z[X]/\langle X^N + 1 \rangle$ 。设 $\sigma > 0$ 为实数, $g, f \leftarrow \chi_\sigma^N$, f 在 R_Q 中可逆。

(N, Q, σ) -NTRU 计算性问题是已知 $h = g \cdot f^{-1} \bmod Q$ 的情况下恢复多项式 f 和 g 。

(N, Q, σ) -NTRU 决策性问题涉及区分多项式 h 与从 R_Q 中抽样得到的随机多项式之间的差异。

矩阵 NTRU: 设 $n > 0$, $q > 1$ 为整数, $\sigma > 0$ 为实数。设 $G, F \leftarrow \chi_\sigma^{n \times n}$, F 为模 q 可逆。

(n, q, σ) -NTRU 矩阵计算问题是在已知矩阵 H 的情况下, 恢复出矩阵 F 和 $G := G \cdot F^{-1} \bmod q$ 。

(n, q, σ) -NTRU 矩阵决策问题旨在区分矩阵 H 与 $Z_q^{n \times n}$ 上的随机矩阵之间的差异。决策 MNTRU 问题定义如下: 设 k 是整数, q 是素数, R 是数字域 K 的整数环, ψ 是 R_q 上的分布。设 $\bar{F} \in R_q^{k \times k}$ 是从 ψ 中采样的元素, 并假设 $\bar{F}$ 是可逆的。 k 秩决策的 MNTRU 问题是区分形式为 $\bar{h} = \bar{g} \cdot \bar{F}^{-1} \in R_q^k$ 的样本与在 R_q^k 中 $\bar{g} \leftarrow \psi^k$ 的随机样本之间的差异。

2.4 LWE 问题

本文方案中所考虑 LWE 问题是基于 MLWE 分布进行定义的, 具体而言, LWE 问题的定义如下所述:

设 $N > 0$, $q > 1$ 是整数, χ 是 $\mathbb{Z}_q$ 上的一个误差

概率分布。密文向量 $s \in \mathbb{Z}_q^N$, 随机向量 $a \in \mathbb{Z}_q^N$, 误差向量 $e \leftarrow \chi$, 输出 LWE 分布:

$$(a, b = a \cdot s + e \bmod q) \in \mathbb{Z}_q^N \times \mathbb{Z}_q$$

(n, q, χ, m) -LWE 计算问题是在给出 (a, b) 的情况下, 求解密文向量 s 。

(n, q, χ, m) -LWE 判定问题是区分 LWE 分布与 $\mathbb{Z}_q^N \times \mathbb{Z}_q$ 上的均匀分布之间的差异。

MLWE 分布定义如下: 设 q 为素数, R 为数字域 k 的整数环, 对于 $\bar{s} \in R_q^k$ 和 ψ 在 R_q 上的分布, MLWE 分布 $A_{s, \psi}$ 的一个样本是由 $(\bar{a}, b = \bar{a} \cdot \bar{s} + e)$ 给出, 其中 $\bar{a} \leftarrow U(R_q^k)$, $e \leftarrow \psi$ 。

在原定义的 MLWE 与 RLWE 分布问题中, 密文为对偶理想 R_q' 中的元素, 并且其误差分布是连续的。当 R 被定义为二次环分数域中的整数环时, 其密文元素与使用 R_q 中的密文元素具有等价性。

与本文所提出方案密切相关的是标准形式下的决策 MLWE 问题, 其定义如下: 设 q 为素数, R 为数字域 K 上的整数环, ψ 是 R_q 上的某一特定分布, 标准形式的决策 MLWE 问题是将样本从 $A_{s, \psi}$ 的随机分布中区分出来。其中, 在 $R_q^k \times R_q$ 中, $\bar{s}$ 代表了一个包含从指定分布 ψ 中采样得到的元素所构成的向量。

接下来介绍 LWE 基本加密方案。

LWE.SecretKeygen(1^λ): 随机均匀选取向量 $s \in \mathbb{Z}_q^N$, 输出密钥;

LWE.Enc(s, m): $m \in \{0, 1\}$, 随机均匀选取向量 $a \in \mathbb{Z}_q^N$, 选取误差向量 $e \leftarrow \chi$, 计算 $b \leftarrow a \cdot s + e$, 输出 $c_1 = a, c_2 = (a \cdot m + b) \in \mathbb{Z}_q$, 即 (c_1, c_2) 。其中, 设 $\Delta := \lfloor q/4 \rfloor$;

LWE.Dec(s, c): 输入 (s, c) , 输出 $\left\lfloor \frac{4}{q} ((c_2 + c_1 \cdot s) \bmod q) \right\rfloor \bmod 2$ 。

2.5 NTWE 问题

NTWE 问题^[20]的提出将 LWE 的困难性与 NTRU 的困难性相结合, 并同样对应于 q 元格中的困难问题。NTWE 问题的实例在形式上与 NTRU 问题相似, 为 $h = g \cdot f^{-1}$, 其中 $g \leftarrow \psi_1$ 和 $f \leftarrow \psi_2$ 。然而, NTWE 问题的实例在构造上与标准 NTRU 问题不同, 其并未采用 $\psi_1 = \psi_2$, 也并未预设 g 为一个数

值较小的元素。相反地, NTWE 问题的分布 ψ_1 是一个模 LWE 分布。

NTWE 分布 $W(\bar{s}, f, \psi)^{[20]}$: 设 q 为素数, k 为整数, n 为 2 的幂次方, $R = \mathbb{Z}[X]/(X^n + 1)$, ψ 为 R_q 上的某一分布。此外, 设 $\bar{s}$ 是 R_q^k 中的向量, f 是 R_q 中的可逆元素。NTWE 分布 $W(\bar{s}, f, \psi)$ 的样本由下述式子给出:

$$(\bar{a}, b = (\bar{a} \cdot \bar{s} + e)f^{-1}) \in R_q^k \times R_q$$

其中 $a \leftarrow U(R_q^k)$ 和 $e \leftarrow \psi$ 。

决策 NTWE 问题 $(DNTWE(\psi, h))^{[20]}$: 令 ψ 为 R_q 上的分布, h 为整数。决策 NTWE 问题 $DNTWE(\psi, h)$ 的实例从未知分布 D 中取样, 该分布 D 有两种可能: 一是均匀随机的分布, 二是满足 $\bar{s} \leftarrow \psi^k$ 和 $f \leftarrow \psi^*$ 的 $W(\bar{s}, f, \psi)$ 分布。DNTWE(ψ, h) 问题旨在确定如何在最多给出 h 个样本的条件下, 判断未知分布 D 中的性质。

对于搜索 NTWE 问题, 不需要恢复用于生成 NTWE 分布的实际密文 $\bar{s}$ 。相反, 对于某个特定的 i , 可以生成与实际密文一致的 NTWE 分布, 进而可以恢复 $\bar{s}X^i$ 和 fX^i 。

搜索 NTWE 问题 $SNTWE(\psi, h)^{[20]}$: 设 ψ 为 R_q 上的分布, h 为整数。搜索 NTWE 问题 $SNTWE(\psi, h)$ 旨在确定如何从 $W(\bar{s}, f, \psi)$ 分布中最多给出 h 个样本的情况下, 恢复某个 i 对应的 $\bar{s}X^i$ 和 fX^i 的情况, 其中 $\bar{s} \leftarrow \psi^k$ 和 $f \leftarrow \psi^*$ 。

3 优化方案设计

本文对 FINAL 算法中自举模块的密钥切换部分进行了研究, 该算法的自举模块与 NTWE 问题有着密切的关联, 其中的 NGS 方案融合了 LWE 和 NTRU 问题, 构造思路与 NTWE 相似。因此, 本文利用 NTWE 问题, 重新设计了自举模块中密钥切换部分, 并替换了原方案的切换密钥。这一改进不仅缩短了切换密钥的长度, 提高了灵活性和安全性, 同时也实现了方案性能和资源利用方面的优化。

3.1 算法描述

基于 LWE 方案和基于 NTRU 的 NGS 方案定义如下。

1. 参数生成 $NGS.ParamGen(1^\lambda)$: 接收安全参数

并输出参数元组 (N, Q, ζ, B, ℓ) , 其中 B 用于分解密文, $\ell := \lceil \log_B(Q) \rceil$ 。

2. 密钥生成 $NGS.KeyGen$: 实现的功能为从 $f' \leftarrow \chi_\zeta^N$ 中取样, 并计算 $f := 1 + 4 \cdot f'$ 直到 R_Q 中存在 f^{-1} 。输出 $sk := f$ 。

3. 标量加密算法 $NGS.EncS(sk, m)$: 将作为 R_Q 上三元多项式中元素的明文 m , 加密成标量密文, 过程具体为给定一个三元多项式 m , 从 $g \leftarrow \chi_\zeta^N$ 中取样, 定义 $\Delta := \lfloor Q/4 \rfloor$, 输出 $c = g/f + \Delta \cdot m \in R_Q$, 则称 c 为 m 的标量加密。

4. 向量加密算法 $NGS.EncVec(sk, m)$: 通过在 R_Q 上使用“小工具向量”, 将明文 m 加密成一个新的向量密文, 过程具体为给定 $m \in M$, 对于 $0 \leq i \leq \ell - 1$, 取样 $g_i \leftarrow \chi_\zeta^N$ 。定义 $g := (g_0, \dots, g_{\ell-1})$, $g = (B^0, B^1, \dots, B^{\ell-1})$ 。输出 $C = g/f + g \cdot m \in R_Q^\ell$, 则称 C 为 m 的向量加密。

此外, 在自举算法的流程中, 由于方案中的 NGS 密文从未被解密, 因此省略了解密过程。

3.2 外部积

在本文优化的方案中, 加密后的多项式 $m \in R := \mathbb{Z}[X]/(X^N + 1)$ 是由以下两种密文格式组成。

标量: 标准 NTRU 密文, 加密 m 为 $g/f + \Delta m \in R_Q$ 。

向量: 加密 m , $c = g/f + g \cdot m \in R_Q^\ell$, 其中 g 是工具向量, $\ell \approx \log(Q)$ 。

接下来在以上这两种密文类型之间定义一个外部积操作, 该操作能够将标量 NTRU 密文(即 R_Q 的元素)与向量 NTRU 密文(即 R_Q 上的向量)相乘, 从而生成新的标量密文。假设有一个标量加密 $c := g/f + \Delta \cdot u \in R_Q$ 的三元多项式 u 和一个向量加密 $C := g/f + g \cdot v \in R_Q^\ell$ 的消息 $v \in M$ 。那么, c 与 C 的外积定义如下式所示:

$$c \odot C := g^{-1}(c) \cdot C \in R_Q$$

由于 $g^{-1}(c) \cdot g = c$, 则 $c_{\text{mult}} = c \odot C$ 等于下式:

$$\begin{aligned} c_{\text{mult}} &:= (g^{-1}(c) \cdot g) / f + (g^{-1}(c) \cdot g \cdot v) \\ &= \underbrace{(g^{-1}(c) \cdot g + g \cdot v)}_{g_{\text{mult}}} / f + \Delta \cdot u \cdot v \end{aligned}$$

因此, 只要噪声项 g_{mult} 足够小, 那么 c_{mult} 就是

乘积 $u \cdot v$ 的有效标量加密。进而假设 $v \in M$, 则 $\|u \cdot v\| \leq \|u\| < 2$ 成立。特别注意, 要满足条件 $\|u \cdot v\| < 4$, 否则将其乘以 Δ 会导致舍入错误, 并产生格式错误的密文。

3.3 模交换

由于 NTRU 与 RLWE 问题的构造思路相似, 所以基于 RLWE 问题方案的模交换部分可以应用于基于 NTRU 问题的方案中。进一步地, 它也适用于在 NTRU 和 LWE 问题基础上构建的 NGS 方案。

模交换操作涉及对密文的缩放处理并引入一个微小的舍入误差, 并随后在解密阶段通过乘以密钥 f 的操作实现还原。即给定某个消息 μ 的一个三元多项式密文 $c = g/f + \Delta \cdot \mu \in R_Q$, 由于满足 $\lfloor y \rfloor = y + \varepsilon$, 进而可以用 c 乘以 q/Q 并对其进行舍入, 得到一个定义为模 q 的密文。

根据参考文献[24]的分析, 定义随机舍入函数如下:

设 $Q, q \in \mathbb{Z}$, 且 $1 < q < Q$ 。随机舍入函数 $[\cdot]_{Q,q} : \mathbb{Z}_Q \rightarrow \mathbb{Z}_q$ 可定义为 $[z]_{Q,q} := \lfloor q \cdot z / Q \rfloor + B$, 其中 $B \in \{0,1\}$ 是伯努利随机变量, 并且满足 $\Pr[B=1] = (q \cdot z / Q) - \lfloor q \cdot z / Q \rfloor \in [0,1]$ 。

其中, 舍入误差 $\varepsilon := [z]_{Q,q} - (q \cdot z / Q)$ 为 1-sub 高斯分布。通过将上述的舍入方法扩展至多项式、向量以及矩阵, 我们对模交换进行如下定义:

$$\text{ModSwitch}(c) = \sum_{i=0}^{N-1} [c_i]_{Q,q} \cdot X^i \in R_q$$

设密文 $c = g/f + \lfloor Q/4 \rfloor \cdot \mu \in R_Q$, 进而我们可以推论出:

$\text{ModSwitch}(c)$ 是 R_q 中 μ 的标量加密, 并且满足:

$$\begin{aligned} \text{Var}(\text{err}(\text{ModSwitch}(c))) &= \text{Var}(q \cdot \text{err}(c) / Q) + \varepsilon \\ &\quad + 4 \cdot \varepsilon \cdot f' \\ &= \text{Var}(q \cdot \text{err}(c) / Q) + \text{Var}(\varepsilon) \\ &\quad + 16 \cdot \text{Var}(\varepsilon \cdot f') \\ &\leq (q/Q)^2 \cdot \text{Var}(\text{err}(c)) \\ &\quad + 1 + 16 \cdot N \cdot \zeta^2 \end{aligned}$$

3.4 利用 NTWE 问题优化后的密钥切换部分

本文方案所提出的利用 NTWE 问题对 FINAL 方案自举模块中密钥切换部分的优化设计思路如下所述:

与 FINAL 方案类似, 本文的基本方案中密文定义为 $(g + \Delta \cdot (m, 0)) \cdot F^{-1} \in \mathbb{Z}_q^n$, 经过模交换操作, 生

成了 NTRU 形式的密文 $c = g/f + \varepsilon + \Delta \cdot \mu \in R_Q$, 其中 μ 是一个多项式, 其常数项满足 $m \in \{0,1\}$ 且 $\Delta := \lfloor q/4 \rfloor$ 。为了完善自举部分的功能, 我们需要采取密钥切换技术, 以重新获得 LWE 形式的密文 $c' = (g' + \Delta \cdot (m, 0)) \cdot F^{-1} \in \mathbb{Z}_q^n$ 。

为了提升自举部分的效率和可行性, 本文方案对 FINAL 中的密钥切换过程进行了优化, 将其细化为密钥生成算法和密钥切换算法两个主要环节。

首先是优化密钥切换部分的密钥生成算法, 该算法的主要目标是将密文格式从 NGS 方案生成的密文转换为加密同一信息的 LWE 密文。算法具体的设计如下: 我们首先设定 LWE 样本 (A, b) 为密钥 s 加密后的结果, 然后计算标量 NGS 密文 $c = g/f + \varepsilon + \Delta \cdot m \in R_Q$ 作为密钥 f 加密后的结果, 其中 ε 是模交换后的舍入误差。接着, 我们将切换密钥重新设计, 定义为经过 NTWE 问题优化后的 LWE 样本向量:

$$\text{ksk}_{\text{NTRU} \rightarrow \text{LWE}} := (A, b := (A \cdot s + e) \cdot P \cdot f_0)$$

其中 $A \in \mathbb{Z}_q^{Nn \times Ln}$, $e \leftarrow \chi_{\sigma_e}^{N \cdot L}$, $P = I_N \otimes g_{q, B_{\text{ksk}}}$, $f_0 := \text{col}_0(\Phi(f)) \in \mathbb{Z}^N$ 。

接下来优化密钥切换部分的密钥切换算法, 具体设计如下: 给定模交换的输出 $c = g/f + \varepsilon + \Delta \cdot m \in R_Q$ 和切换密钥 (Key-Switching Key, ksk), 定义函数:

$$\text{KeySwitch}(c, \text{ksk}) := y \cdot \text{ksk} \in \mathbb{Z}_q^n$$

其中 $y := (g_{q, B_{\text{ksk}}}^{-1}(c_0), \dots, g_{q, B_{\text{ksk}}}^{-1}(c_{N-1})) \in \mathbb{Z}^{N \cdot L}$ 。

最终实现了密钥切换功能, 即给定密文 $c = g/f + \varepsilon + \Delta \cdot m \in R_Q$, 经过密钥切换操作将 NGS 密文转换为 LWE 密文, 具体步骤如下: $\text{KeySwitch}_{\text{NTRU} \rightarrow \text{LWE}}(c, \text{ksk}_{\text{NTRU} \rightarrow \text{LWE}})$:

- ① $\text{ksk}_{\text{NTRU} \rightarrow \text{LWE}} := (A, b := (A \cdot s + e) \cdot P \cdot f_0)$
- ② $a \leftarrow \text{KeySwitch}(c, A)$
- ③ $b \leftarrow \text{KeySwitch}(c, b)$
- ④ 输出: $c' := (a, b)$

接下来, 对 c 的系数向量进行分解, 然后乘以 $\text{ksk}_{\text{NTRU} \rightarrow \text{LWE}}$ 的两个分量, 并定义:

$$y := g^{-1}(\phi(c)) \in \mathbb{Z}^{N \cdot L}$$

计算 $c' := (a, b) = (y \cdot A, y \cdot b) \in \mathbb{Z}_q^{n+1}$, 则可以得到下述等式:

$$\begin{aligned}
b &= y \cdot b = y \cdot ((\bar{A} \cdot \bar{s} + \bar{e}) \cdot f^{-1}) \\
&= (y \cdot \bar{A} \cdot \bar{s} \cdot f^{-1} + y \cdot \bar{e} \cdot f^{-1}) \\
&= a \cdot s + y \cdot \bar{e} + g_0 + \varepsilon \cdot ((1, 0) + 4 \cdot \phi(f')) \\
&\quad + 4 \cdot \varepsilon' \cdot \phi(m) \cdot \phi(f') + \Delta \cdot m_0
\end{aligned}$$

其中 $\varepsilon \in (-1/2, 1/2]$, m_0 是 m 的常数项。满足上述等式则说明 (a, b) 是有效的 LWE 密文 m_0 。

3.5 自举设计

自举模块的操作由自举密钥生成算法和自举算法两部分组成。为了处理 NTRU 问题的三元密文, 本文方案定义了三元 CMux 门。

对于给定 $f_i \in \{-1, 0, 1\}$, 定义密钥 $\text{bsk}_{i,0}$ 和 $\text{bsk}_{i,1}$:

$$\begin{cases}
f_i = -1 \Rightarrow \text{bsk}_{i,0} := \text{NGS.EncVec}(0) \wedge \text{bsk}_{i,1} \\
\quad := \text{NGS.EncVec}(1) \\
f_i = 0 \Rightarrow \text{bsk}_{i,0} := \text{NGS.EncVec}(0) \wedge \text{bsk}_{i,1} \\
\quad := \text{NGS.EncVec}(0) \\
f_i = 1 \Rightarrow \text{bsk}_{i,0} := \text{NGS.EncVec}(1) \wedge \text{bsk}_{i,1} \\
\quad := \text{NGS.EncVec}(0)
\end{cases}$$

根据上述密钥定义, 将 CMux 门定义为

$$\text{CMux}_i(c_i) := 1 + (X^{c_i} - 1) \cdot \text{bsk}_{i,0} + (X^{-c_i} - 1) \cdot \text{bsk}_{i,1}$$

其中 1 是一个简单的、无噪声的加密, 即可简单地使用 g 来表示。并且可以得到 $\text{CMux}_i(c_i) = \text{NGS.EncVec}(X^{c_i \cdot f_i})$ 。其中, 由 $\text{CMux}_i(c_i)$ 加密的消息属于 M , 符合本文在 3.2 节中外部乘积的定义。

接下来对组成自举模块的两个算法分别进行描述。自举密钥生成算法负责生成用于自举操作的特殊密钥, 算法具体描述如下。

算法 1: 自举密钥生成

输入: $s \in \mathbb{Z}_q^n$ - LWE 基本加密方案的密钥

输出: bsk - 自举密钥

步骤 1 $(s_0, \dots, s_{n-1}) \leftarrow \text{col}_0(s)$

步骤 2 for $i \leftarrow 0$ to $n-1$ do

 Compute bsk_i accordingly to (2.1)

步骤 3 Return $\text{bsk} := \{(\text{bsk}_{i,0}, \text{bsk}_{i,1}) : 0 \leq i \leq n-1\}$

自举密钥是执行自举算法的前提, 为方案提供了在加密情况下进行复杂计算的条件。

自举算法允许算法在加密数据上执行复杂的计算, 是实现全同态加密的关键技术。在自举流程的主循环结束后, 可以得到形式为 NTRU 的密文, 然后通过模交换和密钥切换模块将 NTRU 密文转化为 LWE 密文。算法具体描述如下:

算法 2: 自举算法

输入:

$ct = (ct.a, ct.b) \in \mathbb{Z}_q^{n+1}$ - LWE 加密方案的密文, 加密的明文 $m \in \{0, 1\}$

$\{\text{bsk}_i\}_{0 \leq i \leq n-1}$ - 自举密钥, 其中 $\text{bsk}_i \in R_{Q,N}^{\ell}$

$\text{ksk}_{\text{NTRU} \rightarrow \text{LWE}}$ - NGS 密钥 $f \in R$ 转换为 LWE 基本方案密钥 $s \in \mathbb{Z}_q^n$ 的切换密钥

输出: $ct' \in \mathbb{Z}_q^{n+1}$ - 加密相同明文信息 m 的密文

步骤 1: $(c_0, \dots, c_{n-1}, c_n) \leftarrow \left\lfloor \frac{2 \cdot N \cdot ct}{q} \right\rfloor$

步骤 2: $\text{ACC}_0 \leftarrow \left\lfloor \frac{Q}{8} \right\rfloor \cdot X^{N/2} \cdot \sum_{i=0}^{N-1} X^i$

步骤 3: FOR $i \leftarrow 0$ TO $n-1$ DO

$c_{\text{Mux}} \leftarrow \text{CMux}_i(c_i)$

$\text{ACC} \leftarrow \text{ACC} \odot c_{\text{Mux}}$

步骤 4: $\text{ACC} \leftarrow \text{ACC} + \left\lfloor \frac{Q}{8} \right\rfloor \cdot \sum_{i=0}^{N-1} X^i$

步骤 5: $\text{ACC} \leftarrow \text{ModSwitch}(\text{ACC})$

步骤 6: $ct' \leftarrow \text{KeySwitch}(\text{ACC}, \text{ksk})$

返回 ct'

自举密钥生成算法和自举算法共同构成了自举模块的核心, 它们使得方案在保持数据加密的同时, 能够对数据执行复杂的计算操作。这对于构建安全、灵活且功能强大的全同态加密方案至关重要。

4 安全性分析

本文所优化的方案, 其针对选择明文攻击(Against Chosen Plaintext Attacks, CPA)的安全性, 是通过从决策 NTRU 问题的标准混合论证方式直接推导出来的。首先, 根据 MNTRU 问题的假设, 我们把 G/F 看作对 q 的一致模, 从而推导出 $G/F + M$ 的加密具有安全性。然后, 基于循环安全假设, 在矩阵 NTRU 问题下加密 M/F 而非 M , 可以得到一个关键推论: 使用密钥 F 加密的消息具有安全性。由此, 我们可以推导出在基本方案中使用的格式: $(G + M)/F$ 。最后, 自举的安全性遵循弱循环安全假设, 即本文的方案可以用于加密基础方案的密钥, 而基础方案又可以加密本文方案的密钥。如文献[9,26]中所提到的循环安全假设, 均为标准化的且得到了广泛的应用。

4.1 具体安全性

针对 NTRU 问题的安全性研究表明, 对于使用

较大模数 q 的 NTRU 格, 其格约化攻击的性能已经得到了显著提升。根据文献 [7-8,33] 的研究结果, 攻击行为会受到参数“过度拉伸”的影响。然而, Kirchner 与 Fouque 的研究表明, 攻击事件的发生主要归因于参数的选择, 而非 NTRU 格问题本身的结构特性。在设定密文模数 q 的过程中, NTRU 的参数在“过度拉伸”时会导致安全问题的出现, 这个转折点被称作疲劳点。Kirchner 和 Fouque 对疲劳点进行估算, 从而提出了一个渐近上界。为了进一步明确三元 NTRU 问题疲劳点的确切数值, Ducas 等人引入了两个判别事件, 旨在区分标准区域与过度拉伸区域:

私钥恢复 (Secret Key Recovery, SKR) 指的是将一个与私钥向量长度相同的向量嵌入至格基向量中时所触发的事件。

密集子格发现 (Dense Sublattice Discovery, DSD) 指的是一个由私钥生成的密集子格向量被插入到格基向量中时所触发的事件。该向量的长度严格来说比私钥长, 并且在 DSD 事件下, 攻击者可以在以下两种情况中对算法进行攻击: 一种情况是在 DSD 事件之后很快发生 SKR 事件, DSD 事件会与其级联并生成可以用于恢复密钥的密集子格; 另一种情况是被发现的密集子格向量本身就具备解密新生成密文的能力, 无需考虑密钥恢复的过程。

Ducas 等人^[14]的研究表明, 在多元多项式下, NTRU 问题疲劳点的具体值由参数 q 的值决定。超过 q 的阈值, DSD 攻击的效果则超过 SKR 攻击。该研究提出了一个渐近结果, 明确指出了在多元多项式密文的 NTRU 问题中, 疲劳点出现在 $q = n^{2.484+o(1)}$ 处, 并进一步对平均情况进行了分析, 得出对疲劳点的具体预测情况: 当 $n > 100$ 时, 疲劳点为 $q \approx 0.004 \cdot n^{2.484+o(1)}$ 。

4.2 参数选择

参考 Ducas 等人的研究以及他们提供的用于估计 NTRU 问题困难性的相关代码, 在给定的矩阵维度、模数 q 、方差 σ^2 以及密钥分布的情况下, 本文能够确定使用格基规约 (Block Korkin-Zolotarev, BKZ) 破解 NTRU 问题所需的块大小 β 的具体值。为了将 β 转换至具备安全属性的级别, 本文采用了与 TFHE 等价的成本模型。BKZ- β 在 d 维上的运算次数为: $T(d, \beta) := 2^{0.292 \cdot \beta + 16.4 + \log 2(8-d)}$, 其中 n 维 NTRU 问题的 $d = 2 \cdot n$, λ 位的安全级别表示为 $T(d, \beta) \geq 2\lambda$ 。

因此, 为了确定本文方案的参数设定, 我们固定 $N = 1024$ 和三元密文, 进而找到了 $\log Q$ 的最大值,

得到 $\lambda = 128$ 。

本文使用 LWE 估计器^[34]来确定 LWE 问题的参数, 并选择用于外部积和密钥切换的分解基以保证方案的正确性。为了减少自举过程中多项式乘积的计算总量, 我们对前 n_1 个外部乘积使用 B_1 , 对最后 n_2 个外部乘积使用 B_2 , 其中 $n = n_1 + n_2$ 。

基于上述描述, 本文方案的参数选择如下: $n = 610$ 、 $q = 92683 \approx 2^{16.5}$ 、 $N = 2^{10}$ 、 $Q = 912829 \approx 2^{19.8}$ 、 $(B_1, n_1) = (8, 140)$ 、 $(B_2, n_2) = (16, 470)$ 、 $B_{ksk} = 3$ 、 $\ell_1 = 7$ 、 $\ell_2 = 5$ 。

4.3 噪声分析

为了简化噪声分析, 假设所有由向量加密形成的信息都属于以下单项式集合:

$$M = \{\pm b \cdot X^k : b \in \{0, 1\} \text{ and } k \in \mathbb{N}\}$$

本文并非根据每个元素的无限范数边界来分析噪声增长的最坏情况, 而是选择了以平均情况下的噪声为基础进行分析。因此, 本文采用了两种方法来量化噪声的影响, 其一是采用随机小工具分解算法对 g^{-1} 进行实例化, 其二是采用确定性分解并假设 NGS 样本的所有误差系数都是独立且集中的。第一种方法适用于 FHEW 方案, 而后者则可应用于 TFHE 方案中。这两种方法都返回一个 sub 高斯随机变量, 进而假设对于所有 $a \in R_Q$, 选取任一 $\gamma = O(B)$, $g_{q, B_{ksk}}^{-1}(a)$ 是 γ -sub 高斯。

接下来定义标量密文噪声: 设 $c = g/f + \Delta \cdot m \in R_Q$ 。将密文 c 的噪声定义为: $\text{err}(c) := c \cdot f - \Delta \cdot m \in R_Q$, 并将其解释为 $\mathbb{Z}[X]$ 上的多项式, 系数为 $[-Q/2, Q/2]$ 。向量密文噪声的定义如下: 设 $c = g/f + g \cdot m \in R_Q^\ell$ 。将密文 c 的噪声定义为: $\text{err}(c) := c \cdot f - g \cdot m \cdot f \in R_Q$, 并将其解释为系数为 $[-Q/2, Q/2]$ 的 $\mathbb{Z}[X]$ 上的多项式向量。

然后对单项式进行加密的新密文的噪声进行约束, 其中包括特例 $m \in \{0, 1\}$ 。对标量密文噪声的约束定义如下: 设 $c = g/f + \Delta \cdot m \in R_Q$ 是 m 的密文, 如果 m 是一个形式为 $\pm b \cdot X^k$ 的单项式, 对于某个 $b \in \{0, 1\}$, 则满足下式:

$$\text{Var}(\text{err}(c)) \leq \text{Var}(g) + 4 \cdot \zeta^2$$

如果 m 是阶数最多为 $N-1$ 的三元多项式, 则满足下式:

$$\text{Var}(\text{err}(c)) \leq \text{Var}(g) + 4 \cdot N \cdot \varsigma^2$$

此外, 如果 c 是一个新的密文, 则对于单项式 m , $\text{Var}(\text{err}(c)) \leq 5 \cdot \varsigma^2$, 对于三元多项式 m , 方差以 $(4 \cdot N + 1) \cdot \varsigma^2$ 为界。

进一步, 定义外部积的噪声增长: 设 $c := g/f + \Delta \cdot u \in R_Q$, $c := g/f + g \cdot m \in R_Q^\ell$ 。定义 $c_{\text{mult}} := c \odot C$, 可得

$$\begin{aligned} \text{Var}(\text{err}(c_{\text{mult}})) &\leq N \cdot \ell \cdot \gamma^2 \cdot \text{Var}(g) \\ &\quad + \|v\|_2^2 \cdot \text{Var}(g) + 4 \cdot \varsigma^2 \end{aligned}$$

若 $v \in M := \{\pm b \cdot X_k : b \in \{0, 1\} \text{ and } k \in \mathbb{N}\}$, 则:

$$\begin{aligned} \text{Var}(\text{err}(c_{\text{mult}})) &\leq N \cdot \ell \cdot \gamma^2 \cdot \text{Var}(\text{err}(c)) \\ &\quad + \text{Var}(\text{err}(c)) \end{aligned}$$

最后, 定义外积序列的噪声: 对于 $1 \leq i \leq k$, 令 $c_i := g_i/f + g \cdot m_i \in R_Q^\ell$, 其中 $m_i \in M$ 。计算 $c_0 = g/f + \Delta \cdot m_0 \in R_Q$, 其中 m_0 是一个三元多项式。如果 $c' := c \odot_{i=1}^k C_i$, 则:

$$\begin{aligned} \text{Var}(\text{err}(c')) &\leq N \cdot \ell \cdot \gamma^2 \cdot \sum_{i=1}^k \text{Var}(g_i) \\ &\quad + \text{Var}(g_0) + 4 \cdot \varsigma^2 \end{aligned}$$

如果所有的密文都是新生成的, 则:

$$\text{Var}(\text{err}(c')) \leq (4 + (k + 1) \cdot N \cdot \ell \cdot \gamma^2) \cdot \varsigma^2$$

5 运算性能分析

在三种基于相同困难问题并具有快速自举框架的全同态加密方案^[24-25,35]中, 效率最高的是 TFHE。因此, 本文将优化后方案的运行结果与 TFHE 和 FINAL 方案的运行结果进行比较。与 TFHE 方案中的门自举类似, 本文方案中自举上的加速可以等效成二进制电路上的加速, 所以本文以自举效率作为基准, 对比各方案的性能。本文使用 C++ 语言进行编程, 实现了自举过程的概念验证。为确保运行结果比较的公正性, 本文方案选择使用 C++ 编写的最新版本的 TFHE 库。

本文实验利用具有 8 GB RAM 和 3.1 GHz 双核英特尔酷睿 i5 的单核机器来进行。

由于快速傅里叶变换决定了自举的运行时间, 我们使用以下公式计算自举算法(算法 2)中的 $\text{ACC} \odot \text{CMux}(c_i)$:

$$((X^{c_i} - 1) \cdot \text{ACC}) \odot \text{bsk}_i + \text{ACC}$$

在 TFHE 库中可以使用 LWE 的相关公式, 则计

算 $(X^{c_i} - 1) \cdot \text{ACC}$ 时并不需要进行多项式的乘法操作, 而是可以通过对 ACC 的系数进行一次负循环移位, 然后在 $\mathbb{Z}_Q$ 中进行 N 次减法来完成。假设自举密钥已经提前完成了对 FFT 的运算, 则外部乘积的计算量只需要考虑 $\ell_i + 1$ 个 FFT 运算和 ℓ_i 个 Hadamard 向量乘积运算。

在 TFHE 中, 每个密文为 $4 \cdot \ell'$ 个环上元素, 其中 $\ell' = 3$, 而本文方案中, 只有 ℓ 个环上元素, 进而可以计算出在 TFHE 中的自举密钥大小为 $4 \cdot n' \cdot \ell' \cdot N' \cdot \log(q') = 31\text{MB}$ 。

在 TFHE 框架中, 每个外部乘积的实现需要执行 $4 \cdot \ell'$ 次 R_q' 上的乘法操作, 总共需要执行 $4 \cdot n' \cdot \ell' = 7560$ 次环上的乘法操作。然而, 在自举过程中, 最耗时的操作是前向 FFT 和后向 FFT。自举密钥相关的 FFT 已经预先完成计算并存储。自举过程中“for”循环的主要工作是将累积的 RLWE 样本进行分解, 进而获得 $2 \cdot \ell'$ 个环元素。通过计算外部乘积生成新的 RLWE 样本, 然后进行两次向后 FFT, 可以计算出环元素的 FFT 操作总次数。因此, TFHE 每次自举需要进行 $2 \cdot n' \cdot (\ell' + 2) = 6300$ 次 FFT 操作。

NGS 方案建立在 LWE 和 NTRU 问题的基础上, 构成该方案的是 n 个 NGS 密文, 其中 ℓ 略大于 TFHE 中使用的 $\ell' = 3$ 。但本文方案中每个密文所需的存储空间、环乘法的执行次数以及 FFT 运算的次数均低于 TFHE 方案。特别地, 每次自举操作中 FFT 的计算次数为 $n_1 \cdot (\ell_1 + 1) + n_2 \cdot (\ell_2 + 1)$, 其中 $\ell_i = \lceil \log_B(Q) \rceil$ 。

考虑到本文方案具体编码的实现过程中, 所有整数均采用 int 类型进行表示, 因此可以假设每个密钥对应的多项式系数占据 32 位的内存空间。

我们运行了 1000 次自举过程, 计算出刷新后的密文噪声的标准差为 $\sigma_{\text{LWE}} = 29.46$ 和 $\sigma_{\text{NTRU}} = 29.85$, 由此可推算出解密失败的概率为 $p_{\text{LWE}} = 1 - \text{erf}\left(92683 / (16 \cdot \sigma_{\text{LWE}} \cdot \sqrt{2})\right) < 2^{-52}$ 和 $p_{\text{NTRU}} = 1 - \text{erf}\left(131071 / (16 \cdot \sigma_{\text{NTRU}} \cdot \sqrt{2})\right) < 2^{-60}$ 。

表 1 中所示的 FTT 和乘法的数量使用上述解释中描述的方案参数进行计算, 我们测量并计算了算法在 NAND 门上自举运算 1000 次后的平均时间作为运行时间。

表 1 表明, 本文方案自举算法比 TFHE 算法快 39%, 比 FINAL 算法快 17%。同时我们的方法生成的密钥长度为 37.4MB, 优于 FINAL。自举模块负责密钥更新, 是提升方案效率的关键所在, 本文方案

表 1 实际运行结果

Table 1 Practical results

	切换密 钥长度 (MB)	自举密 钥长度 (MB)	R_Q 的乘 法次数	FFT 次数	运行 时间 (ms)
TFHE	40	31	7560	6300	66
FINAL	26.3	13	3330	3940	48
本文方案	25.4	12	2460	3940	40

优化密钥交换算法可以大幅降低自举的计算开销, 从而提高方案整体的效率。

6 结论

本文提出一种利用 NTWE 问题对全同态加密算法 FINAL 进行优化的方案。基于 NTWE 问题中所提出的将 NTRU 与 LWE 问题相结合的思路, 优化了全同态加密方案 FINAL 自举模块中的密钥切换部分, 利用 NTWE 问题的特性重新设计切换密钥, 来替换原方案中密钥切换部分的密钥。

本文方案有效地缩短了密钥的长度, 这一特性在资源受限制的环境下尤为重要, 例如在物联网设备或移动通信设备中, 不仅有利于密钥的存储与传输, 而且增加了系统的灵活性。通过这种灵活性, 该方案能够适用于多种应用环境, 可以与其他加密技术或协议结合使用, 以满足具体的安全需求。此外, 本文方案证明了其抵抗选择明文攻击的安全性, 这意味着其能够更有效地防御多种密码攻击和分析手段, 确保了数据的保密性与完整性, 如避免进行解密时噪声的累积, 从而抵抗噪声增长攻击; 避免进行同态运算时密文长度的增加, 从而抵抗维数膨胀攻击。综上所述, 鉴于密钥长度的缩减、系统灵活性的增强以及安全性的提升, 本方案在云计算、物联网、安全通信等领域展现出广阔的应用前景, 为数据保护和隐私保密提供有效的解决方案。

参考文献

- [1] Gentry C. A fully homomorphic encryption scheme[M]. *Stanford university*, 2009.
- [2] Brakerski Z, Vaikuntanathan V. Efficient Fully Homomorphic Encryption from (Standard)[J]. *SIAM Journal on Computing*, 2014, 43(2): 831-871.
- [3] Brakerski Z, Vaikuntanathan V. Fully Homomorphic Encryption from Ring-LWE and Security for Key Dependent Messages[M]. *Advances in Cryptology – CRYPTO 2011*. Berlin, Heidelberg: Springer, 2011: 505-524.
- [4] López-Alt A, Tromer E, Vaikuntanathan V. On-the-Fly Multiparty Computation on the Cloud via Multikey Fully Homomorphic Encryption[C]. *The Forty-Fourth Annual ACM Symposium on Theory of Computing*, 2012: 1219-1234.
- [5] van Dijk M, Gentry C, Halevi S, et al. Fully Homomorphic Encryption over the Integers[M]. *Advances in Cryptology – EUROCRYPT 2010*. Berlin, Heidelberg: Springer, 2010: 24-43.
- [6] Bos J W, Lauter K, Loftus J, et al. Improved Security for a Ring-Based Fully Homomorphic Encryption Scheme[M]. *Cryptography and Coding*. Berlin, Heidelberg: Springer, 2013: 45-64.
- [7] Albrecht M, Bai S, Ducas L. A Subfield Lattice Attack on Overstretched NTRU Assumptions: Cryptanalysis of Some FHE and Graded Encoding Schemes[M]. *Advances in Cryptology – CRYPTO 2016*. Berlin, Heidelberg: Springer, 2016: 153-178.
- [8] Kirchner P, Fouque P A. Revisiting Lattice Attacks on Overstretched NTRU Parameters[M]. *Advances in Cryptology – EUROCRYPT 2017*. Cham: Springer International Publishing, 2017: 3-26.
- [9] Genise N, Gentry C, Halevi S, et al. Homomorphic Encryption for Finite Automata[C]. *Advances in Cryptology – ASIACRYPT 2019*, 2019: 473-502.
- [10] Lee C, Wallet A. Lattice analysis on MiNTRU problem[J]. *Cryptology ePrint Archive*, 2020:2230-240.
- [11] Li Z C, Zhang J M, Yang Y T, et al. A Fully Homomorphic Encryption Scheme Based on NTRU[J]. *Acta Electronica Sinica*, 2018, 46(4): 938-944.
(李子臣, 张卷美, 杨亚涛, 等. 基于 NTRU 的全同态加密方案[J]. *电子学报*, 2018, 46(4): 938-944.)
- [12] Yang Y T, Liu B Y, Sun Y F, et al. Fully Homomorphic Masking Defense Scheme Based on NTRU[J]. *Chinese Journal of Computers*, 2019, 42(12): 2742-2753.
(杨亚涛, 刘博雅, 孙亚飞, 等. NTRU 全同态掩码防御方案[J]. *计算机学报*, 2019, 42(12): 2742-2753.)
- [13] Li R Q, Jia C F. A Multi-Key Homomorphic Encryption Scheme Based on NTRU[J]. *Journal of Cryptologic Research*, 2020, 7(5): 683-697.
(李瑞琪, 贾春福. 一个基于 NTRU 的多密钥同态加密方案[J]. *密码学报*, 2020, 7(5): 683-697.)
- [14] Ducas L, van Woerden W. NTRU Fatigue: How Stretched Is Overstretched? [M]. *Advances in Cryptology – ASIACRYPT 2021*. Cham: Springer International Publishing, 2021: 3-32.
- [15] Hoffstein J, Pipher J, Silverman J H. NTRU: A Ring-Based Public Key Cryptosystem[M]. *Algorithmic Number Theory*. Berlin, Heidelberg: Springer, 1998: 267-288.
- [16] Regev O. On Lattices, Learning with Errors, Random Linear Codes, and Cryptography[C]. *The Thirty-Seventh Annual ACM Symposium on Theory of Computing*, 2005: 84-93.
- [17] Peikert C. A Decade of Lattice Cryptography[J]. *Foundations and Trends® in Theoretical Computer Science*, 2016, 10(4): 283-424.
- [18] Stehlé D, Steinfeld R. Making NTRU as Secure as Worst-Case Problems over Ideal Lattices[M]. *Advances in Cryptology – EUROCRYPT 2011*. Berlin, Heidelberg: Springer, 2011: 27-47.
- [19] Wang Y, Wang M Q. Provably Secure NTRUEncrypt over any Cyclotomic Field[M]. *Selected Areas in Cryptography – SAC 2018*. Cham: Springer International Publishing, 2019: 391-417.
- [20] Gärtnert J. NTWE: A Natural Combination of NTRU and LWE[M]. *Post-Quantum Cryptography*. Cham: Springer Nature Switzerland, 2023: 321-353.

- [21] Bonte C, Iliashenko I, Park J, et al. FINAL: Faster FHE Instantiated with NTRU and LWE[M]. *Advances in Cryptology – ASIACRYPT 2022*. ChamSpringer Nature Switzerland2022: 188-215.
- [22] Alperin-Sheriff J, Peikert C. Faster Bootstrapping with Polynomial Error[M]. *Advances in Cryptology – CRYPTO 2014*. Berlin, Heidelberg: Springer, 2014: 297-314.
- [23] Micciancio D, Polyakov Y. Bootstrapping in FHEW-Like Cryptosystems[C]. *The 9th on Workshop on Encrypted Computing & Applied Homomorphic Cryptography*, 2021: 17-28.
- [24] Ducas L, Micciancio D. FHEW: Bootstrapping Homomorphic Encryption in less than a Second[M]. *Advances in Cryptology -- EUROCRYPT 2015*. Berlin, HeidelbergSpringer2015: 617-640.
- [25] Chillotti I, Gama N, Georgieva M, et al. TFHE: Fast Fully Homomorphic Encryption over the Torus[J]. *Journal of Cryptology*, 2020, 33(1): 34-91.
- [26] Carpov S, Gama N, Georgieva M, et al. Privacy-Preserving Semi-Parallel Logistic Regression Training with Fully Homomorphic Encryption[J]. *BMC Medical Genomics*, 2020, 13(S7): 88.
- [27] Guimarães A, Borin E, Aranha D F. Revisiting the Functional Bootstrap in TFHE[J]. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2021: 229-253.
- [28] Chillotti I, Ligier D, Orfila J B, et al. Improved Programmable Bootstrapping with Larger Precision and Efficient Arithmetic Circuits for TFHE[M]. *Advances in Cryptology – ASIACRYPT 2021*. ChamSpringer International Publishing2021: 670-699.
- [29] Genise N, Micciancio D, Peikert C, et al. Improved Discrete Gaussian and Subgaussian Analysis for Lattice Cryptography[M]. *Public-Key Cryptography – PKC 2020*. Cham: Springer International Publishing, 2020: 623-651.
- [30] Genise N, Micciancio D, Polyakov Y. Building an Efficient Lattice Gadget Toolkit: Subgaussian Sampling and More[M]. *Advances in Cryptology – EUROCRYPT 2019*. ChamSpringer International Publishing2019: 655-684.
- [31] Jeon S, Lee H S, Park J. Efficient Lattice Gadget Decomposition Algorithm with Bounded Uniform Distribution[J]. *IEEE Access*, 2021, 9: 17429-17437.
- [32] Lyubashevsky V, Peikert C, Regev O. On Ideal Lattices and Learning with Errors over Rings[M]. *Advances in Cryptology – EUROCRYPT 2010*. Berlin, Heidelberg: Springer, 2010: 1-23.
- [33] Cheon J H, Jeong J, Lee C. An Algorithm for NTRU Problems and Cryptanalysis of the GGH Multilinear Map without a Low-Level Encoding of Zero[J]. *LMS Journal of Computation and Mathematics*, 2016, 19: 255-266.
- [34] Albrecht M R, Player R, Scott S. On the Concrete Hardness of Learning with Errors[J]. *Journal of Mathematical Cryptology*, 2015, 9(3): 169-203.
- [35] Pereira H V L. Bootstrapping Fully Homomorphic Encryption over the Integers in less than One Second[M]. *Public-Key Cryptography – PKC 2021*. ChamSpringer International Publishing2021: 331-359.



杨亚涛 河南平顶山人, 博士, 北京电子科技学院教授、博士生导师, 西安电子科技大学硕士生导师, 中国密码学会理事, 主要研究方向为信息安全、全同态密码、抗量子密码、密码协议和算法。Email: yy2008@163.com



张思瑶 北京人, 硕士研究生, 主要研究领域为密码学与信息安全



陈亮宇 江西赣州人, 电子科技学院硕士生, 主要研究方向为通信工程、信息安全



刘国鑫 安徽合肥人, 硕士研究生, 主要研究领域为密码学和信息安全