一种改进的基于LWR的零知识证明方案

尹思彤[1)] 高军涛[1)] 李雪莲[2)]
[1)](西安电子科技大学通信工程学院 西安 710071)
[2)](西安电子科技大学数学与统计学院 西安 710126)

摘 要 格密码体制因其抗量子攻击、可并行计算和存在平均情况下困难问题与最坏情况下困难问题之间的安全性归约等特性，成为后量子密码学的重要研究方向。然而，基于LWE问题及其变体(如RLWE、MLWE)的密码方案存在高斯抽样复杂、公钥尺寸过大等问题。LWR通过消除高斯噪声采样，计算效率更高、公钥及密文尺寸更小，但在构造基于LWR的零知识证明时，仍面临计算开销、存储需求、通信效率以及噪声控制等挑战。本文采用承诺-证明框架，构建了一个高效且安全的基于LWR的非交互式零知识证明系统。与CRYPTO'23上Esgin等人方案相比优势如下：承诺阶段将线性关系和约束关系构造为形式简洁的综合方程，提高了方案的扩展性和灵活性；使用可扩展输出函数以及Pedersen承诺技术生成中间承诺值，在此基础上构建默克尔树，更高效地递归生成哈希树根作为紧凑的公开承诺值，其中对承诺计算中使用到的随机数向量以及生成元向量的生成机制进行优化，抵御重放攻击并降低弱生成元造成的信息泄露风险；证明阶段则基于zk-SNARK协议以非交互方式生成简洁的证明，确保承诺值满足约束关系，并通过约束拒绝抽样的参数来降低重启率，提高验证效率；验证阶段设计多个验证条件，逐层排除无效证明，确保zk-SNARK验证的高效性。通过分析零知识性、模拟可提取性、简洁性等安全属性，证明了方案的安全性、高效性。此外，本文给出了针对不同形式的模数和舍入模数的舍入计算方案，其中一种新的混合模数方案可以在保持多项式乘法高效的同时，通过模数转换和位运算进一步优化模约化和舍入计算的效率，并且将混合模数方案与其他方案进行各项基本操作的时间复杂度以及存储的对比，说明方案的可行性。

关键词 零知识证明；舍入学习；承诺-证明框架；Pedersen承诺；默克尔树；可扩展输出函数
中图法分类号 TP309 **DOI号** 10.11897/SP.J.1016.2025.02450

The Improved LWR-Based Zero-Knowledge Proof Scheme

YIN Si-Tong[1)] GAO Jun-Tao[1)] LI Xue-Lian[2)]
[1)](Department of Communication Engineering, Xidian University, Xi'an 710071)
[2)](Department of Mathematics and Statistics, Xidian University, Xi'an 710126)

Abstract Lattice-based cryptographic constructions have become a pivotal research avenue in post-quantum cryptography owing to their intrinsic resistance to quantum adversaries, their amenability to parallel computation, and the existence of rigorous security reductions from average-case hardness assumptions to worst-case lattice problems. However, schemes predicated on the learning with errors (LWE) problem and its structured variants—namely ring-LWE (RLWE) and module-LWE (MLWE)—are encumbered by the computational intricacies of high-dimensional discrete Gaussian sampling and the attendant inflation of public-key sizes required to embed sufficient entropy. The learning with rounding (LWR) problem ameliorates these impediments by supplanting stochastic Gaussian noise with a deterministic rounding operation, thereby yielding markedly enhanced computational throughput and substantially

收稿日期：2025-02-25；在线发布日期：2025-07-09。本课题得到陕西省重点研发计划(2021ZDLGY06-04)、西安电子科技大学交叉培育项目(21103240011)、国家密码科学基金(2025NCSF02010)资助。**尹思彤**，硕士研究生，主要研究领域为区块链、零知识证明、隐私保护。E-mail：sityin@163.com。**高军涛**(通信作者)，博士，副教授，主要研究领域为量子计算在密码分析中的应用等。E-mail：jtgao@mail.xidian.edu.cn。**李雪莲**，博士，副教授，主要研究领域为信息安全、区块链。

reduced public-key and ciphertext dimensions. Notwithstanding these improvements, the formulation of zero-knowledge proofs (ZKPs) over LWR-based primitives continues to confront substantive obstacles, including but not limited to prover computational overhead, verifier storage demands, communication bandwidth constraints, and the delicate calibration of rounding-induced error accumulation. In this work, we present a non-interactive zero-knowledge (NIZK) proof system grounded in the LWR problem, architected within a commitment-and-proof paradigm that meticulously balances efficiency, security, and modular extensibility. Relative to the scheme introduced by Esgin et al. at CRYPTO 2023, our construction delivers several salient enhancements. During the commitment phase, we amalgamate linear relations and modular constraint relations into a unified ensemble of composite equations, thereby streamlining the proof logic and facilitating seamless extension to more intricate relation sets. Intermediate commitment values are produced by leveraging an extensible-output function (XOF) in conjunction with Pedersen commitment techniques, guaranteeing both computational binding and statistical hiding properties. These intermediate commitments are subsequently organized into a Merkle tree, which permits logarithmic-size public commitments in the form of the Merkle root, affording a compact and verifiable commitment footprint. To bolster security against replay attacks and to attenuate the leakage risks associated with suboptimal generator choices, we optimize the sampling mechanisms for both the randomness vectors and the Pedersen generator vectors, ensuring each commitment instance remains cryptographically unique and unpredictable. In the proof generation phase, we instantiate a succinct zk-SNARK protocol to derive non-interactive proofs that convincingly demonstrate the committed values satisfy the stipulated composite equations. By judiciously selecting the parameters that govern constraint rejection sampling, we effect a significant reduction in prover restart frequency, thereby lowering the overall proof-generation latency and improving prover-side efficiency. During verification, our design employs a hierarchy of layered verification conditions that iteratively cull invalid proofs through successive checks, yielding a verification algorithm which remains practically efficient for real-world deployment. We conduct a rigorous security analysis of the proposed system, establishing its zero-knowledge property under standard cryptographic assumptions, proving simulation extractability to ensure soundness against malicious provers, and verifying succinctness to confirm that proof sizes remain compact. Furthermore, we address the practicalities of modular arithmetic by delineating rounding-computation schemes tailored to various moduli structures and rounding moduli. Central to this discussion is a novel hybrid modulus scheme that preserves the asymptotic efficiency of number-theoretic transform (NTT)-accelerated polynomial multiplication while further optimizing modular reduction and rounding computations via a combination of modulus-conversion techniques and bitwise operations. We supplement this proposal with a comparative evaluation of the hybrid modulus scheme against alternative rounding approaches, analyzing the time complexity of core polynomial operations and storage overhead.

Keywords zero-knowledge proof; learning with rounding; commit-and-prove framework; Pedersen commitment; Merkle tree; extendable output function

1 引 言

零知识证明(Zero-Knowledge Proof,ZKP)允许证明者在不泄露任何额外信息的情况下向验证者证明某一声明的真实性。相较于经典的ZKP方案,基于格上困难问题的ZKP方案提供了更强的安全性保证,并具备抗量子攻击能力。其中,带误差学习问

题(Learning With Errors,LWE)作为格密码学的核心问题之一,与格上最坏情况困难问题存在可归约性,使得基于LWE的ZKP方案在后量子密码学中受到广泛关注。这类方案不仅具有严谨的安全性证明,还能在保证量子安全性的同时支持高效计算,在隐私保护、身份认证及安全多方计算等领域展现出广泛应用潜力。

早期研究如Peikert等人[1]提出的基于LWE问题的非交互式零知识证明(Non-Interactive Zero-Knowledge,NIZK)方案,该方案不仅具有量子安全性,还实现了简洁高效的NP问题证明,在多方通信场景下具有显著优势。然而文献[1-8]表明,这类方案普遍存在计算和存储开销较大的问题,在实际部署中对计算资源要求较高。2021年Lyubashevsky等人[2]提出基于模上带误差学习问题(Module Learning With Errors,MLWE)的非交互式ZKP,通过使用BDLOP承诺并对拒绝采样技术进行优化,证明大小约减小了30%。尽管在效率和适用性上有所改进,但仍存在随机数泄露风险,并且为确保零知识性和抗泄露性,不得不增加模数和格维度。2022年Lyubashevsky等的研究[3]进一步优化了基于MLWE和模短整数解问题(Module Short Integer Solution,MSIS)的交互式证明系统,结合Ajtai和BDLOP方案,缩小证明尺寸,优化计算过程,但在高维度LWE实例下仍然面临效率瓶颈。此外,2023年提出的基于Hint-MLWE的方案[9]构建了更高效的交互式知识证明框架,在理论上具有更低的计算和通信开销,实际应用中仍然依赖于强假设且面临较高的计算开销。

综上所述,基于LWE及其变体的ZKP方案虽然在理论基础、安全性等方面具有优势,但仍需解决计算和存储开销大、效率低等问题。

舍入学习(Learning With Rounding,LWR)是构造基于格的后量子密码原语的一个重要困难问题。文献[10-11]指出LWR可视为LWE的确定性变体,通过确定性的舍入操作来替代LWE中依赖随机误差采样的过程。这一变化不仅提升了计算效率、减小了公钥及密文尺寸,而且由于不再依赖高斯噪声,LWR在大规模样本需求下的效率优势更加突出。此外,这种确定性结构还在一定程度上增强了抗侧信道攻击能力,使其成为格密码体制中广泛应用的底层函数,在构造伪随机函数、陷门函数和同态加密等密码协议中发挥着重要作用,例如F. Esgin等人[12]提出了一种创新的混合精确/松弛的格证明框架(LANES$^+$),并展示了其在舍入证明和可验证伪随机函数中的应用。但其主要局限是没有给出完整的零知识证明系统流程描述,使得难以直接理解和实现整个系统,对后续的研究和应用存在一定影响;并且其中的约束关系被明确定义且固定,这意味着如果应用场景发生变化或者需要更复杂的约束关系,方案可能不够灵活,并且使用BDLOP承诺进行实例化,承诺值较大。

将LWR引入ZKP构造,不仅能够优化LWE的性能瓶颈,还能在保证安全性的同时提供更高效的实现方案。

基于LWR构造ZKP方案存在以下不足:(1)参数选择复杂,文献[11,13]表明,基于LWR的方案需仔细选择参数(如环结构、模数等)以确保安全性;(2)计算开销较高,Ducas等人[14]指出,一些基于LWR的方案在证明生成过程中涉及大量矩阵-向量乘法和模运算,计算复杂度较高,限制其在计算资源受限环境(如物联网设备)中的应用;(3)通信成本较大,Saber方案[15]指出,基于LWR的ZKP方案需传输大量数据(如承诺值、挑战值、响应值等),会导致带宽占用和传输延迟;(4)噪声控制复杂,舍入操作会引入噪声,须对噪声进行有效控制,防止信息泄露;(5)现有ZKP系统在交互式或大规模数据处理场景下,证明生成时间过长或验证延迟较高,在高实时性应用(如支付验证、身份认证)中影响用户体验。

本方案针对上述问题,设计了一种基于LWR的非交互式零知识证明方案,具有后量子安全、灵活可扩展、计算效率高、通信开销低等优势,可以为区块链上信息安全提供零知识性的隐私保护。具体优化措施如下:

(1)优化参数选择与计算效率:采用多项式环结构,针对模数和舍入模数的不同取值(比如是否为数论变换(Number-Theoretic Transform,NTT)模数、是否为2的幂次等),提供多种计算方案,并提出一种新型混合模数计算方案,提升计算效率;利用零知识简洁非交互式知识论证(Zero-Knowledge Succinct Non-Interactive Argument of Knowledge,zk-SNARK)的可信设置生成公共参数,确保参数生成过程高效、安全,并且给出了避免中心化风险的策略。

(2)构造综合方程:综合方程通过矩阵分块形式将多个约束整合为统一框架,新增约束时只需在框架中新增向量,无需重构证明逻辑,提升协议的灵活性与可扩展性;原来的多个变量(如消息向量、辅助向量、随机数向量等)被抽象为单一变量,避免多变量符号冗余,降低协议复杂性,同时后续操作只涉

及对新的单一变量操作，降低初始变量泄露的风险。

(3)降低存储与通信开销：将多个Pedersen承诺值压缩为一个Merkle树根，隐藏中间承诺值，抗碰撞哈希函数的使用确保无法逆向推断消息细节，将树根作为最终的公开承诺值可以减少存储占用和传输开销。

(4)优化随机数与生成元的生成机制：基于SHAKE256的随机数生成机制，通过种子、时间戳和维度适配，可以抵御随机数重用攻击和重放攻击；显式验证生成元的合法性，避免因弱生成元导致承诺可预测的风险。

(5)优化噪声控制机制：将舍入关系转化为误差约束问题，确保噪声在安全范围内，不影响系统的可靠性和安全性。方案通过融合线性关系约束与多项式零化约束来实现误差控制，线性约束建立误差向量与关键变量间的关联关系，通过验证这些关系来确保误差合理性，从而实现对误差的控制；多项式约束基于多项式零点特性对噪声参数施加限制，进一步优化控制精度。在证明阶段通过构建zk-SNARK证明来确保误差向量满足约束，提高了证明的效率和可靠性，从而优化噪声控制。

(6)采用非交互方式，通过哈希函数直接生成挑战值，仅需单次通信即可完成证明，无需验证者在线参与，避免多轮交互造成的通信开销和时序依赖问题，并且挑战值与承诺值、掩蔽值的绑定可以防证明者篡改，也可以防止交互式协议中恶意验证者选择特定挑战来窃取部分秘密信息的问题；zk-SNARK确保生成的证明具有紧凑性；拒绝抽样机制通过合理设定参数减少不必要的重启。

(7)验证阶段构造新的公开信息集，将多个子约束嵌套到一次验证过程中，并设计范数验证、线性关系验证以及挑战值一致性验证，逐层排除无效证明，确保zk-SNARK验证的高效性。

2　基础知识

2.1　符号表示

(1)$\mathcal{R}_{q,d}=\mathbb{Z}_q[X]/(X^d+1)$表示多项式环，$d$为2的幂次，模数$q\geqslant 2$。

(2)粗体小写字母$\boldsymbol{a}$表示列向量；$\boldsymbol{a}^{\mathrm{T}}$表示向量$\boldsymbol{a}$的转置；粗体大写字母$\boldsymbol{A}$表示矩阵；$\boldsymbol{A}^{\mathrm{T}}$表示矩阵$\boldsymbol{A}$的转置。

(3)对于向量$\boldsymbol{x}\in\mathcal{R}_{q,d}^n$，$\vec{\boldsymbol{x}}\in\mathbb{Z}_q^{dn}$表示在$\mathbb{Z}_q$中$\boldsymbol{x}$的(串联)系数向量。一般来说用$\vec{\boldsymbol{x}}$表示$\mathbb{Z}_q$上的向量，用$\boldsymbol{x}$表示$\mathcal{R}_{q,d}$上的向量。

(4)标量写作$x\in\mathbb{Z}_q$；向量写作$\boldsymbol{x}\in\mathbb{Z}_q^n$，表示长度为$n$的向量，其中每个元素属于$\mathbb{Z}_q$。

(5)对于$\boldsymbol{x}\in\mathcal{R}_{q,d}$，用$\lfloor\boldsymbol{x}\rceil_p$表示$\lfloor(p/q)\cdot\boldsymbol{x}\rceil$，舍入是按坐标进行的。

(6)对于$\mathbb{Z}_q$上的向量$\vec{\boldsymbol{x}}$和$\vec{\boldsymbol{y}}$，$\vec{\boldsymbol{x}}\circ\vec{\boldsymbol{y}}$表示分量相乘。○表示在一组元素上的分量相乘。

(7)对于以下形式的环上多项式：$w=w_0+w_1X+\cdots+w_{d-1}X^{d-1}\in\mathcal{R}$，定义$\ell_\infty$和$\ell_2$范数：$\|w\|_\infty=\max_j\|w_j\|_\infty$，$\|w\|=\sqrt{\|w_0\|_\infty^2+\cdots+\|w_{d-1}\|_\infty^2}$。

(8)定义向量$\boldsymbol{w}=(w_1,\cdots,w_m)\in\mathcal{R}^m$的$\ell_\infty$和$\ell_2$范数$\|\boldsymbol{w}\|_\infty=\max_j\|w_j\|_\infty$，$\|\boldsymbol{w}\|=\sqrt{\|w_1\|^2+\cdots+\|w_m\|^2}$。

(9)Rot(•)表示输入在$\mathbb{Z}_q$上的代表矩阵。

(10)⊗：张量积。

2.2　理论基础

2.2.1　拒绝抽样

算法1.　$\mathrm{Rej}(\boldsymbol{z},\boldsymbol{c},\phi,T)$.

(1)$\sigma=\phi T$；$\mu(\phi)=e^{12/\phi+1/(2\phi^2)}$；$u\leftarrow[0,1)$；

(2)IF$u>\frac{1}{\mu(\phi)}\cdot\exp\left(\frac{-2\langle\boldsymbol{z},\boldsymbol{c}\rangle+\|\boldsymbol{c}\|^2}{2\sigma^2}\right)$，THEN RETURN 1；

(3)IF$\|\boldsymbol{z}\|_\infty>12\sigma$，THEN RETURN 1(由Lyu12[16]可知$\Pr[\|\boldsymbol{z}\|_\infty>12\sigma;\boldsymbol{z}\leftarrow D_\sigma^1]<2^{-100}$)；

(4)ELSE RETURN 0.

2.2.2　LWR及其扩展(环LWR、矩阵LWR)

定义1.　LWR分布。假设$N,p,q,m\in\mathbb{Z}^+$，其中$N\geqslant 1$是安全参数，$q\geqslant p\geqslant 2$。给定秘密向量$\boldsymbol{s}\in\mathbb{Z}_q^N$，LWR分布$\mathrm{L}_{\mathrm{LWR}}$定义为$\mathbb{Z}_q^{N\times m}\times\mathbb{Z}_p^m$上的分布$(\boldsymbol{A},\boldsymbol{b})$：

$$\mathrm{L}_{\mathrm{LWR}}=\{(\boldsymbol{A},\boldsymbol{b})|\boldsymbol{A}\xleftarrow{\$}\mathbb{Z}_q^{N\times m},\boldsymbol{b}=\lfloor\boldsymbol{A}^{\mathrm{T}}\boldsymbol{s}\rceil_p\in\mathbb{Z}_p^m\},$$

其中，$\boldsymbol{A}\xleftarrow{\$}\mathbb{Z}_q^{N\times m}$表示矩阵$\boldsymbol{A}$的元素都独立地从$\mathbb{Z}_q$上均匀随机选取。$\boldsymbol{b}=\lfloor\boldsymbol{A}^{\mathrm{T}}\boldsymbol{s}\rceil_p\in\mathbb{Z}_p^m$表示通过矩阵与向量的乘积$\boldsymbol{A}^{\mathrm{T}}\boldsymbol{s}$舍入至模$p$得到的向量。

定义2.　RLWR分布。假设$N,p,q\in\mathbb{Z}^+$，$q\geqslant p\geqslant 2$。定义多项式环$\mathcal{R}_q=\mathbb{Z}_q[x]/(f(x))$，其中$f(x)=\sum_{i=0}^N f_ix^i$是$N$次多项式。给定环元素$s\in\mathcal{R}_q$，RLWR分布$\mathrm{L}_{\mathrm{RLWR}}$定义为$\mathcal{R}_q\times\mathcal{R}_p$上的分布

(a,b)：

$$L_{RLWR}=\{(a,b)|a\xleftarrow{\$}\mathcal{R}_q,b=\lfloor a\cdot s\rceil_p\in\mathcal{R}_p\},$$

其中，$a\xleftarrow{\$}\mathcal{R}_q$表示环元素$a$均匀随机地从$\mathcal{R}_q$中选取。RLWR分布中$b(x)$的计算分为三步：

(1)整数域上的多项式乘法：$b_1(x)=a(x)\cdot s(x)$；

(2)模约化：$b_2(x)=\big(b_1(x)\bmod f(x)\big)\bmod q$；

(3)舍入计算：$b(x)=\lfloor(p/q)\cdot b_2(x)\rceil$。

定义3. MLWR分布。令$N,p,q,m,n\in\mathbb{Z}^+$，$q\geqslant p\geqslant 2$。定义多项式环$\mathcal{R}_q$如上。给定向量$\boldsymbol{s}\in\mathcal{R}_q^m$，MLWR分布$L_{MLWR}$定义为$\mathcal{R}_q^{m\times n}\times\mathcal{R}_p^n$上的分布$(\boldsymbol{A},\boldsymbol{b})$：

$$L_{MLWR}=\{(\boldsymbol{A},\boldsymbol{b})|\boldsymbol{A}\xleftarrow{\$}\mathcal{R}_q^{m\times n},\boldsymbol{b}=\lfloor\boldsymbol{A}^{\mathrm{T}}\boldsymbol{s}\rceil_p\in\mathcal{R}_p^n\},$$

其中，$\boldsymbol{A}\xleftarrow{\$}\mathcal{R}_q^{m\times n}$表示矩阵$\boldsymbol{A}$的每个元素都取自$\mathcal{R}_q$上的均匀随机分布。特别地，当$m=n=1$时即为RLWR。

LWR的伪随机性体现在舍入操作隐藏了底层的线性关系，使得输出结果与真正的随机值难以区分。具体来说，LWR通过将线性方程的计算结果舍入到固定模数的粗糙子集空间，替代了LWE中的随机噪声。这种舍入操作虽然是确定性的，却能有效破坏输入与输出之间的直接线性关系，使得输出看起来像随机生成的。攻击者即便掌握部分信息，也难以通过计算恢复原始未舍入的值。在适当的参数设置下，LWR问题的计算复杂性被证明至少与LWE问题一样困难，确保了其在密码学应用中的安全性[17]。

2.2.3 可扩展输出函数

可扩展输出函数(Extendable Output Function，XOF)允许从固定长度的输入种子生成任意长度的输出比特流。选择合适的可扩展输出函数可以保证生成伪随机数的安全性和效率。核心特性包括：

(1)伪随机性与抗预测性：在随机种子输入下，输出的任意子序列均满足伪随机性要求。即使攻击者知道部分输出值，也无法推导内部状态或其他输出，确保生成过程的不可预测性。

(2)确定性与一致性：对于相同输入，始终产生相同输出，确保随机数生成过程的可重复性和一致性。

(3)输出长度无关性：输出长度不受输入长度限制，可根据需求动态调整，从而提供极高的灵活性。此外，多次调用XOF生成的随机数相互独立，且不受其他输出干扰。

(4)抗碰撞性与单向性：安全性基于Keccak算法特有的Sponge结构，也依赖于哈希函数的单向性和抗碰撞性等属性，即无法从输出反推输入，也无法找到不同输入映射到同一输出。

安全哈希算法Keccak(Secure Hash Algorithm Keccak，SHAKE)是SHA-3家族中的一种XOF，有两个主要变种：SHAKE128和SHAKE256。SHAKE128提供至少128位的安全性，性能较高，适合一般安全需求的场景；SHAKE256提供至少256位的安全性，适合高敏感性或抗量子攻击场景。

本方案是基于LWR问题的零知识证明，安全性需求较高，为确保足够的抗量子攻击能力和未来适应性，选用SHAKE256作为可扩展输出函数。

2.2.4 Merkle树

Merkle树是一种树形数据结构，将大量数据分块，每个数据块通过哈希函数生成哈希值，再递归组合这些哈希值，最终生成根哈希。任何数据块的改变都会导致根哈希变化，因此Merkle树可以高效地验证数据的完整性。

在承诺方案中，Merkle树可将多个承诺组合为一个根值，减小数据公开部分的大小，同时确保原始数据的完整性和可验证性。

Merkle树使用的哈希函数需具有抗碰撞性，确保不同的输入不会产生相同的哈希值，防止伪造数据；并且在区块链等安全协议中，抗碰撞性确保数据的不可伪造性和不可篡改性，防止攻击。

3 方案设计与分析

在基于格的隐私保护协议中，所需的协议功能归结为：在底层环$\mathcal{R}_{q,d}$上构建用于证明以下形式关系的零知识协议：$\boldsymbol{As}+\boldsymbol{Bm}+\boldsymbol{Cr}=\boldsymbol{t}$，其中$\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}$是公共矩阵，$\boldsymbol{t}$是公共向量，$\boldsymbol{s}$是具有小系数的高维消息，$\boldsymbol{m}$是具有大系数的低维消息(通常是协议中的辅助项，有助于证明$\boldsymbol{s}$中元素之间的关系)，$\boldsymbol{r}$是随机数向量。$\boldsymbol{s}$是构成证明者证据的秘密向量，往往表示二进制或三进制消息，在某些集合中具有小坐标，例如$S=\{-1,0,1\}$；而$\boldsymbol{m}$作为辅助项往往需要表示更多内容，$\boldsymbol{r}$需要足够的随机性和高维度来保证隐藏性和安全性，因此二者在$\mathcal{R}_{q,d}$中取值。此外秘密向量还满足额外的约束关系：$\boldsymbol{R}_{\mathrm{s}}\boldsymbol{s}+\boldsymbol{R}_{\mathrm{m}}\boldsymbol{m}=\boldsymbol{u}\bmod q$，其中矩阵$\boldsymbol{R}_{\mathrm{s}},\boldsymbol{R}_{\mathrm{m}}$以及向量$\boldsymbol{u}$是公开的。

为了使用LWR进行设计，将线性方程组进行构造，得到一个综合方程$\begin{pmatrix}\boldsymbol{A} & \boldsymbol{B} & \boldsymbol{C}\\ \boldsymbol{R}_{\mathrm{s}} & \boldsymbol{R}_{\mathrm{m}} & \boldsymbol{0}^{N\times r}\end{pmatrix}\begin{pmatrix}\boldsymbol{s}\\ \boldsymbol{m}\\ \boldsymbol{r}\end{pmatrix}=\begin{pmatrix}\boldsymbol{t}\\ \boldsymbol{u}\end{pmatrix}$。令$\boldsymbol{K}=\begin{pmatrix}\boldsymbol{A} & \boldsymbol{B} & \boldsymbol{C}\\ \boldsymbol{R}_{\mathrm{s}} & \boldsymbol{R}_{\mathrm{m}} & \boldsymbol{0}^{N\times r}\end{pmatrix}, \boldsymbol{w}=\begin{pmatrix}\boldsymbol{s}\\ \boldsymbol{m}\\ \boldsymbol{r}\end{pmatrix}, \boldsymbol{v}=\begin{pmatrix}\boldsymbol{t}\\ \boldsymbol{u}\end{pmatrix}$。因此构造后的新关系为$\boldsymbol{Kw}=\boldsymbol{v}$。在$\mathbb{Z}_q$上，给定公共矩阵$\boldsymbol{K}$和舍入模数$p$，按照系数进行舍入计算得到$\boldsymbol{h}=\lfloor\boldsymbol{Kw}\rceil_p \bmod p=\lfloor(p/q)\cdot\boldsymbol{Kw}\rceil \bmod p$，误差向量为$\boldsymbol{e}=\boldsymbol{Kw}-(q/p)\cdot\boldsymbol{h}\pmod q$。在舍入证明中，需要对$\boldsymbol{e}$精确证明，以确保其坐标在$\left[0,\left(q/p\right)-1\right]$中。

方案使用Commit-and-Prove框架，包含Gen、Com、Prove、Verify四个算法。目标是证明元组$\left(\boldsymbol{e},\boldsymbol{w},\vec{\boldsymbol{b}}\right)\in\mathcal{L}\left(mp,ulp\right)$即$\left(\left(mp,ulp\right),\left(\boldsymbol{e},\boldsymbol{w},\vec{\boldsymbol{b}}\right)\right)\in R'_{rnd}$的知识。其中，子约束分别是：线性关系、二进制展开、多项式零化验证、向量范数约束。$\mathcal{L}\left(mp,ulp\right)$表达式如下所示：

$$\mathcal{L}\left(mp,ulp\right)=\left\{\begin{array}{c}\left(\boldsymbol{e},\boldsymbol{w},\vec{\boldsymbol{b}}\right):\boldsymbol{e}=\boldsymbol{Kw}-\dfrac{q}{p}\boldsymbol{h}\bmod q\\ \wedge\vec{\boldsymbol{e}}=\boldsymbol{G}\vec{\boldsymbol{b}}\\ \wedge P\left(\vec{\boldsymbol{e}},\vec{\boldsymbol{b}}\right)=0 \bmod q\forall P\in mp\\ \wedge\left\|c\cdot\boldsymbol{w}\right\|_{\infty}\leqslant\gamma\\ \wedge\left\|c\right\|_{\infty}\leqslant\gamma_c,\text{其中}\gamma、\gamma_c\ll q\end{array}\right\},$$

接下来详细介绍方案的四个算法。

3.1　参数设定与生成$\mathrm{Gen}\left(1^{\lambda}\right)$

(1)选择合适的参数q和d，定义多项式环$\mathcal{R}_{q,d}=\mathbb{Z}_q[x]/f(x)$，其中$f(x)=x^d+1$，$d$是2的幂次；

(2)选择抗碰撞哈希函数$\mathcal{H}:\{0,1\}^{*}\to\mathcal{C}\subseteq\mathcal{R}_{q,d}$；

(3)使用zk-SNARK的可信设置，输入安全参数λ，生成公共参数pp和用于生成/验证证明的密钥对$\left(pk,vk\right)$。

3.2　承诺阶段$\mathrm{Com}_{pp}\left(\boldsymbol{e},\boldsymbol{w},\vec{\boldsymbol{b}},(\boldsymbol{K},\boldsymbol{h})\right)$

(1)选择私有向量$\boldsymbol{s}=\{-1,0,1\}^{m_1}$，$\boldsymbol{m}\in\mathcal{R}_{q,d}^{m_2}$和随机向量$\boldsymbol{r}\in\mathcal{R}_{q,d}^{r}$；

(2)在$\mathcal{R}_{q,d}$上均匀随机采样得到以下矩阵：$\boldsymbol{A}\in\mathcal{R}_{q,d}^{n\times m_1}$，$\boldsymbol{B}\in\mathcal{R}_{q,d}^{n\times m_2}$，$\boldsymbol{C}\in\mathcal{R}_{q,d}^{n\times r}$，$\boldsymbol{R}_{\mathrm{s}}\in\mathcal{R}_{q,d}^{N\times m_1}$，$\boldsymbol{R}_{\mathrm{m}}\in\mathcal{R}_{q,d}^{N\times m_2}$；

(3)计算线性关系与约束关系：

线性关系：$\boldsymbol{As}+\boldsymbol{Bm}+\boldsymbol{Cr}=\boldsymbol{t}\bmod q\in\mathcal{R}_{q,d}^{n}$；

约束关系：$\boldsymbol{R}_{\mathrm{s}}\boldsymbol{s}+\boldsymbol{R}_{\mathrm{m}}\boldsymbol{m}=\boldsymbol{u}\bmod q\in\mathcal{R}_{q,d}^{N}$；

(4)构造综合方程$\begin{pmatrix}\boldsymbol{A} & \boldsymbol{B} & \boldsymbol{C}\\ \boldsymbol{R}_{\mathrm{s}} & \boldsymbol{R}_{\mathrm{m}} & \boldsymbol{0}^{N\times r}\end{pmatrix}\begin{pmatrix}\boldsymbol{s}\\ \boldsymbol{m}\\ \boldsymbol{r}\end{pmatrix}=\begin{pmatrix}\boldsymbol{t}\\ \boldsymbol{u}\end{pmatrix}$，即$\boldsymbol{Kw}=\boldsymbol{v}$；

(5)对上式进行舍入运算，得到舍入关系①$R_{rnd}=\left\{\left((\boldsymbol{K},\boldsymbol{h});\boldsymbol{w}\right):\boldsymbol{w}\in\mathcal{R}_{q,d}^{m_1+m_2+r}\wedge\boldsymbol{h}=\lfloor\boldsymbol{Kw}\rceil_p \bmod p\right\}$，其中$\boldsymbol{K}\in\mathcal{R}_{q,d}^{(n+N)\times(m_1+m_2+r)}$；

(6)依赖以下事实，构造新的关系：

事实：令$\overrightarrow{\boldsymbol{Kw}}\in\mathbb{Z}_q^{(N+n)d}$，$\vec{\boldsymbol{h}}\in\mathbb{Z}_q^{(N+n)d}$，其中$q>p$，$p$整除$q$。那么$\vec{\boldsymbol{h}}=\left\lfloor\overrightarrow{\boldsymbol{Kw}}\right\rceil_p$当且仅当存在$\vec{\boldsymbol{e}}\in\mathbb{Z}_q^{(N+n)d}$使得$\vec{\boldsymbol{e}}\in\left[q/p\right]^{(N+n)d}$并且$\vec{\boldsymbol{e}}=\overrightarrow{\boldsymbol{Kw}}-(q/p)\cdot\vec{\boldsymbol{h}}\bmod q$。

综上，给定公共$(\boldsymbol{K},\boldsymbol{h})$，证明者证明$(\boldsymbol{w},\boldsymbol{e})$的知识满足如下关系②：

$$R'_{rnd}=\left\{\begin{array}{l}\left((\boldsymbol{K},\boldsymbol{h});(\boldsymbol{w},\boldsymbol{e})\right):\boldsymbol{w}\in\mathcal{R}_{q,d}^{m_1+m_2+r}\wedge\\ \boldsymbol{e}=\boldsymbol{Kw}-\dfrac{q}{p}\cdot\boldsymbol{h}\bmod q\wedge\vec{\boldsymbol{e}}\in\left[\dfrac{q}{p}\right]^{(N+n)d}\end{array}\right\}。$$

根据事实知①与②等价，因此可以先证明②；

(7)设置$(2,k)$使得$\dfrac{q}{p}=2^k$，k表示元素进行二进制分解后的位数；

(8)计算$\vec{\boldsymbol{b}}\in\mathbb{Z}_q^{(N+n)dk}$作为$\boldsymbol{e}$的系数的二进制分解；

(9)定义关于$\left(\vec{\boldsymbol{e}},\vec{\boldsymbol{b}}\right)$的多项式操作如下：$P\left(\vec{\boldsymbol{e}},\vec{\boldsymbol{b}}\right)=\bigcirc_{i\in[2]}\left(\vec{\boldsymbol{b}}-\vec{i}\right)$，其中$\vec{i}=(i,i,\cdots,i)$。可以证明$\vec{\boldsymbol{b}}$的坐标都位于有效的二进制范围$[0,1]$中；

(10)在$\mathbb{Z}_q$上$\left(\vec{\boldsymbol{e}},\vec{\boldsymbol{b}}\right)$坐标下多元多项式集合$mp=\{P\}$；

(11)生成Gadget矩阵$\boldsymbol{G}=\boldsymbol{I}_{(N+n)d}\otimes\boldsymbol{g}'$，其中Gadget向量$\boldsymbol{g}'=\left(2^0,2^1,2^2,\cdots,2^{k-1}\right)$。Gadget矩阵的作用是将二进制分解的系数向量$\vec{\boldsymbol{b}}$线性组合，得到原始误差向量$\vec{\boldsymbol{e}}$，因此重构的线性关系为$\vec{\boldsymbol{e}}=\boldsymbol{G}\vec{\boldsymbol{b}}$；

(12)欲证明的线性关系的公开信息的集合$ulp=\left\{\left(\boldsymbol{K},-\boldsymbol{I}_{(N+n)d},\dfrac{q}{p}\cdot\boldsymbol{h}\right),\left(\boldsymbol{I}_{(N+n)d},\boldsymbol{G}\right)\right\}$；

(13)对于任意$c\in\mathcal{C}$，设置公共参数$\eta\geqslant\left\|c\cdot\boldsymbol{w}\right\|$、$\eta_e\geqslant\left\|c\cdot\boldsymbol{e}\right\|$以及$\phi$、$\phi_e$；其中$\eta\geqslant\left\|c\cdot\boldsymbol{w}\right\|$确保掩蔽向量的采样范围足够大，不会泄露$\boldsymbol{w}$的信息；$\eta_e\geqslant\left\|c\cdot\boldsymbol{e}\right\|$

确保误差不会太大，可以有效控制舍入误差的影响；

(14)采样掩蔽向量，其中$\boldsymbol{j}$用来掩蔽秘密向量$\boldsymbol{w}$，$\boldsymbol{y}$用来掩蔽误差向量$\boldsymbol{e}$，因此掩蔽向量维数应分别为$\dim(\boldsymbol{w})$、$\dim(\boldsymbol{e})$。即$\boldsymbol{j}\leftarrow D_{\phi\eta,d}^{m_1+m_2+r}$、$\boldsymbol{y}\leftarrow D_{\phi,\eta,,d}^{N+n}$；

(15)独立随机采样种子ρ、ρ_g、$\rho_h\xleftarrow{\$}\{0,1\}^{256}$，其中$\rho_g\neq\rho_h$避免后续$g_i$和$h_i$存在直接线性相关性；

(16)令$\vec{\boldsymbol{eb}}=\left(\vec{\boldsymbol{y}},\vec{\boldsymbol{e}},\vec{\boldsymbol{b}}\right)\in\mathbb{Z}_q^{d(N+n)(k+2)}$。接着采用Pedersen向量承诺生成中间承诺值，并利用Merkle树得到紧凑的哈希树根，输出对$\vec{\boldsymbol{eb}}$的承诺值t。

①计算生成元向量：

i. 选择有限域F_{q_1}的乘法群$\mathrm{F}_{q_1}^*$作为基本运算群，其中q_1是大素数，使得$\mathrm{F}_{q_1}^*$的阶为q_1-1；

ii. 生成元向量：利用种子ρ_g、ρ_h计算生成元：

$g_i=\mathrm{SHAKE256}(\rho_g\|i)\bmod q_1$；

$h_i=\mathrm{SHAKE256}(\rho_h\|i)\bmod q_1$；

其中，$i=1,2,\cdots,d(N+n)k$。

iii. 验证g_i、h_i是否为$\mathrm{F}_{q_1}^*$的生成元：

对q_1-1的每个素因子p'，验证$g_i^{(q_1-1)/p'}\neq1$，$h_i^{(q_1-1)/p'}\neq1$；

如果某个g_i或h_i不满足生成元条件，则更新种子$\rho_g\leftarrow\rho_g+1$或$\rho_h\leftarrow\rho_h+1$，重新计算；

得到的生成元向量分别为：

$\boldsymbol{g}=\left(g_1,g_2,\cdots,g_{d(N+n)k}\right)$，$\boldsymbol{h}=\left(h_1,h_2,\cdots,h_{d(N+n)k}\right)$。

②生成随机数向量：

需要对向量$\vec{\boldsymbol{y}}\in\mathbb{Z}_q^{d(N+n)}$、$\vec{\boldsymbol{e}}\in\mathbb{Z}_q^{d(N+n)}$、$\vec{\boldsymbol{b}}\in\mathbb{Z}_q^{d(N+n)k}$进行承诺。为确保随机数向量具有足够的随机性、灵活性和安全性，使用种子ρ、时间戳ts和SHAKE256，为每个向量生成对应的随机数向量：

$$\boldsymbol{r}_y=\mathrm{SHAKE256}\left(\rho\left\|\text{“}\vec{\boldsymbol{y}}\text{”}\right\|ts\right)\bmod q_1;$$

$$\boldsymbol{r}_e=\mathrm{SHAKE256}\left(\rho\left\|\text{“}\vec{\boldsymbol{e}}\text{”}\right\|ts\right)\bmod q_1;$$

$$\boldsymbol{r}_b=\mathrm{SHAKE256}\left(\rho\left\|\text{“}\vec{\boldsymbol{b}}\text{”}\right\|ts\right)\bmod q_1。$$

通过调整SHAKE256输出长度和切分方法，可以确保随机数向量维数和消息向量维数匹配。具体来说，SHAKE256输出比特流中每个元素映射到$\mathbb{Z}_{q_1}$中时，至少需要$\log_2(q_1)$比特。步骤一，确定需要的比特长度：向量$\boldsymbol{y}$维数为$d(N+n)$，总比特需求为$L=d(N+n)\cdot\log_2(q_1)$。步骤二，生成比特流：调用SHAKE256，指定输出比特流长度为L，输入为种子、标签和时间戳。步骤三，将生成的比特流按$\log_2(q_1)$比特为一块进行分割，每块表示一个整数；对每个整数取模q_1，得到一个$\mathbb{Z}_{q_1}$中的元素；重复此步骤，直到生成$d(N+n)$个元素，即得到了维数为$d(N+n)$的随机数向量$\boldsymbol{r}_y$。$\boldsymbol{r}_e$和$\boldsymbol{r}_b$生成方法类似。

③生成向量承诺：

利用生成元向量和随机数向量，为消息向量$\vec{\boldsymbol{y}}$、$\vec{\boldsymbol{e}}$、$\vec{\boldsymbol{b}}$生成对应的向量承诺，承诺形式为：

$$t_y=\sum_{i=1}^{d(N+n)}y_i\cdot g_i+r_{y,i}\cdot h_i\bmod q_1;$$

$$t_e=\sum_{i=1}^{d(N+n)}e_i\cdot g_i+r_{e,i}\cdot h_i\bmod q_1;$$

$$t_b=\sum_{i=1}^{d(N+n)k}b_i\cdot g_i+r_{b,i}\cdot h_i\bmod q_1;$$

其中，g_i、h_i为生成元向量的分量，y_i、e_i、b_i为消息向量的分量，$r_{y,i}$、$r_{e,i}$、$r_{b,i}$为随机数向量的分量。

④构建Merkle树：

计算承诺的哈希值，作为Merkle树的叶节点：

$L_1=\mathcal{H}(t_y)$、$L_2=\mathcal{H}(t_e)$、$L_3=\mathcal{H}(t_b)$。这里的$\mathcal{H}$可以选择SHA-256、SHA-3等主流哈希函数，不仅保证了叶节点大小固定且较小，适用于向量承诺值过长的情况，同时增强承诺值的隐私性，并且固定长度的输入更适配哈希函数的计算逻辑，可优化Merkle树性能。从下往上两两合并，生成父节点及最终树根：$P_1=\mathcal{H}(L_1\|L_2)$、$R=\mathcal{H}(P_1\|L_3)$。

这一步确保了所有承诺值的关联性，树根R可作为整个承诺结构的唯一标识，同时记录叶节点到树根的路径信息way。由于验证阶段无需重构树结构，而是隐式地基于路径信息进行验证，因此way需要包含所有必要信息，具体包括：叶节点索引(本文提到的三个叶节点L_1、L_2、L_3的索引可以分别记为0、1、2)、叶节点哈希值(L_1、L_2、L_3)、兄弟节点的哈希值及位置(如+1表示兄弟节点在右侧，−1表示兄弟节点在左侧)、父节点的哈希值、上层兄弟节点的哈希值及位置。

⑤承诺和开口值的输出：

$t=R$：公开承诺值，用于后续的证明验证；

$t'=\left(\vec{\boldsymbol{y}},\vec{\boldsymbol{e}},\vec{\boldsymbol{b}},\boldsymbol{w},\boldsymbol{r}_y,\boldsymbol{r}_e,\boldsymbol{r}_b,way\right)$：私密的开口值，保留向量和随机数的私密性。

3.3 证明阶段 $\mathrm{Prove}_{pp}\left((mp,ulp),(\boldsymbol{K},\boldsymbol{h}),(t;t')\right)$

(1)计算挑战值：

解析$(t;t')=\left(t;\left(\vec{\boldsymbol{y}},\vec{\boldsymbol{e}},\vec{\boldsymbol{b}},\boldsymbol{w},\boldsymbol{r}_y,\boldsymbol{r}_e,\boldsymbol{r}_b,way\right)\right)$；

令$(\boldsymbol{A},\boldsymbol{B})=(\boldsymbol{K},-\boldsymbol{I}_{N+n})$；

使用$\boldsymbol{j}$和$\boldsymbol{y}$进行掩蔽，计算$\boldsymbol{F}=\boldsymbol{Aj}+\boldsymbol{By}$；

生成挑战$c=\mathcal{H}(pp,mp,ulp,t,\boldsymbol{F})$；

(2)生成证明信息：$\boldsymbol{f}=\boldsymbol{y}+c\cdot\boldsymbol{e},\boldsymbol{z}=\boldsymbol{j}+c\cdot\boldsymbol{w}$；

(3)拒绝抽样：

如果$\mathrm{Rej}(\boldsymbol{f},c\boldsymbol{e},\phi_e,\eta_e)$则重启；

如果$\mathrm{Rej}(\boldsymbol{z},c\boldsymbol{w},\phi,\eta)$则重启；

(4)生成zk-SNARK证明：

1)构造新的公开信息：

具体格式如下：$ulp'=\left(\begin{pmatrix}\boldsymbol{I}_{(N+n)d} & \boldsymbol{I}_{N+n}\otimes\mathrm{Rot}(c)\boldsymbol{0}\\ \boldsymbol{0} & -\boldsymbol{I}_{(N+n)d} & \boldsymbol{G}\end{pmatrix},\begin{pmatrix}\vec{\boldsymbol{f}}\\ \vec{\boldsymbol{0}}\end{pmatrix}\right)$，即$\begin{pmatrix}\boldsymbol{I}_{(N+n)d} & \boldsymbol{I}_{N+n}\otimes\mathrm{Rot}(c) & \boldsymbol{0}\\ \boldsymbol{0} & -\boldsymbol{I}_{(N+n)d} & \boldsymbol{G}\end{pmatrix}\begin{pmatrix}\vec{\boldsymbol{y}}\\ \vec{\boldsymbol{e}}\\ \vec{\boldsymbol{b}}\end{pmatrix}=\begin{pmatrix}\vec{\boldsymbol{f}}\\ \vec{\boldsymbol{0}}\end{pmatrix}$等价于$\begin{cases}\vec{\boldsymbol{f}}=\vec{\boldsymbol{y}}+c\cdot\vec{\boldsymbol{e}}\\ \vec{\boldsymbol{e}}=\boldsymbol{G}\vec{\boldsymbol{b}}\end{cases}$；

2)证明者根据承诺值生成zk-SNARK证明：$\pi_N\leftarrow\mathrm{Prove}_{pk}((mp,ulp'),(t;t'))$；

3)生成最终证明$\pi=(\pi_N,(c,\boldsymbol{z},\boldsymbol{f}))$。

最终公开输出(t,π)。

3.4　验证阶段$\mathrm{Veirfy}_{pp}((mp,ulp),(\boldsymbol{K},\boldsymbol{h}),(t,\pi))$

(1)与前面承诺阶段一样，设置mp,ulp；

解析$\pi=(\pi_N,(c,\boldsymbol{z},\boldsymbol{f}))$；

(2)范数验证：若$\|\boldsymbol{f}\|_\infty>12\phi_e\eta_e$或者$\|\boldsymbol{z}\|_\infty>12\phi\eta$，直接返回0；

(3)线性关系验证：计算$\boldsymbol{F}'=\boldsymbol{Az}+\boldsymbol{Bf}-c\frac{q}{p}\boldsymbol{h}$，此处$(\boldsymbol{A},\boldsymbol{B})=(\boldsymbol{K},-\boldsymbol{I}_{N+n})$；

(4)挑战值一致性验证：

计算$c'=\mathcal{H}(pp,mp,ulp,t,\boldsymbol{F}')$；

如果$c'\neq c$，直接返回0；

(5)zk-SNARK有效性验证：

像证明阶段一样，设置ulp'；

$1/0\leftarrow\mathrm{Ver}_{vk}((mp,ulp'),t,\pi_N)$，使用zk-SNARK的验证密钥$vk$进行验证，检查$\pi_N$是否为有效知识证明。

(6)返回验证结果：如果所有检查均通过，返回1，表示验证成功；否则返回0。

3.5　方案分析

(1)通过构造综合方程并舍入，来压缩计算过程中传输和存储的数据量。$\boldsymbol{v}$的元素在模q范围内，每个元素用$\log_2(q)$比特表示；$\boldsymbol{h}$是舍入后得到的，元素在模p范围内，每个元素用$\log_2(p)$比特表示。向量维数是$(N+n)$，因此节省的比特数量为$(N+n)(\log_2(q)-\log_2(p))$。

(2)说明构造综合方程后得到的合成向量$\boldsymbol{w}$具有小范数，原因如下：小范数向量可以降低矩阵运算的计算复杂度；大范数可能泄露真实值的特定分量或规律，小范数能更好地隐藏秘密向量的细节，防止泄露隐私信息。

通过对$\boldsymbol{w}$进行范数上界约束，间接约束$\boldsymbol{s}$的范数，有效地隐藏秘密信息。即使$\boldsymbol{s}$的范数较大，只要最终合成的$\boldsymbol{w}$的范数符合预定上界，秘密信息依然安全。直接约束$\boldsymbol{w}$的范数比单独约束$\boldsymbol{s}$的范数更为简单有效，减少了计算量。根据$\boldsymbol{z}=\boldsymbol{j}+c\cdot\boldsymbol{w}$并且$\boldsymbol{z}$的范数存在上界，可知$\boldsymbol{w}$的范数也存在上界。

(3)使用零知识证明来隐藏叶节点和路径信息，因此验证阶段无需重构Merkle树。最终的公开证明结构为：$\pi=(\pi_N,(c,\boldsymbol{z},\boldsymbol{f}))$，其中$\pi_N$是零知识证明，用于验证树根$R$的完整性和一致性；$(c,\boldsymbol{z},\boldsymbol{f})$是公开的挑战值、混淆值，用于对承诺的真实性进行额外验证。这样验证者只需要验证π_N是否有效以及验证$(c,\boldsymbol{z},\boldsymbol{f})$的正确性，确保承诺满足初始的约束关系，而无需重构Merkle树，避免了逐层检查树结构带来的高成本。在使用zk-SNARK生成π_N时，t_y、t_e、t_b和way作为证明的隐含输入，证明者能够向验证者证明他们知道哈希路径，但不会泄露路径细节。zk-SNARK的零知识性确保验证者只能验证路径有效性，确认证明者确实知道如何从叶节点到根节点进行哈希计算，但无法从证明中推断路径内容，从而确保数据的隐私性。

(4)对偶攻击目前仅在使用指数多条短向量，并使用独立性假设的前提下，才能够比原始攻击的复杂度低[18]。然而，Ducas等人[19]指出，在指数多条短向量的情况下独立性假设不成立。目前对偶攻击的复杂度尚无定论，本文采用原始攻击对算法安全性进行评估。根据Alkim等人提出的评估方法[20]，在模数$q=2^{13}$，舍入模数$p=2^9$，多项式次数$d=256$的情况下，使用原始攻击求解该参数下的LWR问题，BKZ算法的分块大小$beta$至少为707。因此，该参数下的经典安全强度为$2^{0.292beta}=2^{206.4}$，量子安全强度为$2^{0.265beta}=2^{187.4}$，即可以提供206位的经典安全强度和187位的量子安全强度，与相同参数设置

下Saber方案[15]的199位经典安全强度、181位量子安全强度接近，甚至更优，以抵御未来算法优化。

(5)方案中使用基于离散对数问题的Pedersen承诺方案但在后量子场景下仍能保持安全的原因是，哈希函数在量子攻击下保持单向性和抗碰撞性。中间承诺值t_y、t_e、t_b以及中间哈希值L_1、L_2、L_3均不公开，只有经过多次哈希函数迭代生成的Merkle树根R公开。如果攻击者想要从R反推出L_1、L_2、L_3，再到t_y、t_e、t_b，最后反推出$\vec{\boldsymbol{y}},\vec{\boldsymbol{e}},\vec{\boldsymbol{b}}$，则需要进行多次哈希函数的逆运算。由于采用的是SHA-256、SHA-3(包括SHAKE256等)这种抗量子哈希函数，量子计算机的Grover算法只能将原像攻击的复杂度从$O(2^n)$降低到$O(2^{n/2})$(n为哈希值的长度)，对于足够长的哈希值(如256位)，2^{128}的复杂度在当前以及未来一段时间内仍然是难以实现的。因此，哈希函数的单向性在量子计算环境下仍能保障一定的安全性。此外，哈希函数的抗碰撞性使得攻击者很难构造出不同的叶节点组合使得最终树根相同。在量子计算环境下，Grover算法可以将传统的基于生日悖论的碰撞攻击的复杂度从$O(2^{n/2})$降低到$O(2^{n/4})$，但抗量子哈希函数的内部结构和设计可以有效抵御这种攻击，比如SHA-3的海绵结构在理论上对量子攻击具有更强的抵御能力，其抗碰撞性不依赖于生日悖论假设，而是基于更严格的数学证明[21]。因此使用Pedersen承诺再结合哈希树技术，不仅可以保证计算的高效性和简洁性，还能在量子环境下保持安全。

(6)在参数设定与生成部分采用zk-SNARK的可信设置进行参数生成时，传统单一协调者模式会引入中心化风险。为规避这种风险，提出以下策略：引入安全多方计算(Multi-Party Computation，MPC)协议[22]，替代单一协调者角色，任意参与者均可在无需准入许可的情况下轮流对通用参考字符串进行随机化并销毁自身“有毒废物”，从而将信任假设降低为“至少有一名诚实参与者”即可保证整个SNARK设置算法的零知识性和完备性不被破坏。此外，借鉴Zcash Powers of Tau公共参数生成仪式的增量式更新机制，引入可更新(Perpetual)模式，允许新参与者随时加入并注入随机数，体现“越多人越安全(More-the-Merrier)”的扩展性，彻底消除对单一可信方的依赖[23]。通过以上策略，可以将参数生成从“一次性可信仪式”转变为“持续、动态、可更新的多方参与过程”，即使初始参与者中存在恶意行为，后续诚实参与者仍可以通过注入新鲜随机数“清洗”潜在风险，确保zk-SNARK系统的长期安全性与去中心化可信性。

4 LWR的三项基本操作与参数选取

RLWR和MLWR实现包含多项式乘法、模约化和舍入计算三项基础操作。各操作均有多种适用于不同参数的实现方案，且实现效率受参数影响大。

4.1 多项式乘法

多项式乘法往往是基于RLWR、RLWE等假设的密码体制中耗时最高的操作。常用的多项式乘法快速实现算法有Karatsuba、Toom-Cook、NTT、快速傅里叶变换(Fast Fourier Transformation，FFT)以及稀疏乘法算法。NTT算法理论复杂度低、实际效率高，但大多都是针对特定参数的。常用模数是素数或者某些特定形式的合成数(例如$q=2^kp-1$，其中p是素数)。

如果q和p取2的幂次，则模约化、舍入计算都较易实现。此时在二进制表示中，舍入操作变成简单的位操作，效率较高。

综上，NTT乘法实现大多要求模数q为NTT模数(通常取奇素数)，而采用按位与实现模整数、先加法再移位实现舍入计算则要求模数q的因子包含2的幂次。参数q不能同时满足上述限制条件。为提升模约化、舍入计算的效率，参数q和p通常取2的幂次[24]。

4.2 模约化

模约化涉及模多项式和模整数两种操作。模多项式采用减法实现；模整数主要运算开销是除法。模数q通常取特殊值，利用特殊模整数实现方法：q为2的幂次时，采用按位与操作；q是NTT模数时，采用模数为q的NTT乘法算法，可以省略模整数运算。

4.3 舍入计算

舍入计算的主要开销是除法。目前RLWR、MLWR参数q和p通常取2的幂次。这样q/p是2的幂次，可以使用先加法再移位计算舍入值。

4.4 参数选取

要保证LWR中参数选择可以满足安全性。q和p的选择决定了后续使用何种多项式乘法。

4.4.1 方案a(q取NTT模数$q=2^kp-1$，p为奇素数)

此时，多项式乘法可用NTT来加速实现，并且可以省略模整数运算。此时由于q的取值，比值q/p

不再是2的幂次，但是$(q+1)/p=2^k$，可以使用先加法再移位操作计算舍入值。虽然此时按照之前的舍入误差计算方式无法得到准确的舍入误差，但仅有少量输入的舍入值存在误差，且误差可以通过一次减法修正。实际舍入误差应该为$\boldsymbol{e}_{\text{real}}=\boldsymbol{Kw}-(q+1)/p\cdot\boldsymbol{h}$，而方案中的误差向量计算公式为$\boldsymbol{e}=\boldsymbol{Kw}-(q/p)\cdot\boldsymbol{h}$，因此可以设置修正项$\boldsymbol{e}_{\text{adjustment}}=-(1/p)\cdot\boldsymbol{h}$，因此$\boldsymbol{e}_{\text{real}}=\boldsymbol{e}+\boldsymbol{e}_{\text{adjustment}}$，$\boldsymbol{e}_{\text{real}}$即为真实误差。

4.4.2　方案b(q和p均为2的幂次)

此时不采用NTT而是采用Karatsuba算法或者Toom-Cook算法对多项式进行乘法运算，后续的模数和舍入操作使用高效的位运算来处理。其中Karatsuba适用于中等规模多项式乘法，而Toom-Cook作为Karatsuba的扩展版本，更适用于较大规模的多项式乘法。

4.4.3　方案c(混合模数方案)

该方案可以在加速多项式乘法的同时，确保模约化和舍入操作的高效性。主要思想是先在一个较大的域内利用NTT加速多项式乘法，然后通过模数转换回较小的域再执行模约化和舍入操作，以确保结果的准确性。

假设从矩阵$\boldsymbol{K}$中选择多项式$k(x)$，从向量$\boldsymbol{w}$中选择多项式$w(x)$，阶数均小于d，q是NTT模数，$q'=2^k$：

(1)多项式乘法$h_1(x)=k(x)\cdot w(x)$：

①将两个多项式变换到NTT域(模数q)，即NTT($k(x)$)和NTT($w(x)$)；

②在NTT域中进行逐点相乘：

$\text{NTT}\left(h_1(x)\right)=\text{NTT}\left(k(x)\right)\cdot\text{NTT}\left(w(x)\right)$；

③使用逆NTT变换将乘积变换回系数表示$h_1(x)=\text{NTT}^{-1}\left(\text{NTT}\left(h_1(x)\right)\right)$；

得到的结果$h_1(x)$是定义在模数q下的多项式。

(2)模约化：

对$h_1(x)$取模X^d+1计算得到$h_2(x)$：

$h_2(x)=h_1(x)\bmod\left(X^d+1\right)$，即把$h_2(x)$的阶降低到$d$，仍在模数$q$下。

将$h_2(x)$从模数q转换到模数q'：

将每个系数$h_2(x)_i\in\mathbb{Z}_q$映射到$h'_2(x)_i\in\mathbb{Z}_{q'}$。即$h'_2(x)=h_2(x)\bmod q'$。通常来说，可以取$q'$为最接近$q$的2的幂次，这样可以保持更多精度，减少误差。具体来说，首先计算$\log_2(q)$，取$\log_2(q)$得到最接近但不大于q的指数，这时对应的2的幂次为$a=2^{\lfloor\log_2(q)\rfloor}$，接着计算$b=2^{\lfloor\log_2(q)\rfloor+1}$。因此候选幂次的区间为$\left[2^{\lfloor\log_2(q)\rfloor},2^{\lfloor\log_2(q)\rfloor+1}\right]$。分别计算$q$与两侧幂次的差值$\Delta_a=|q-a|$，$\Delta_b=|b-q|$，选择差值较小的一个即为与$q$最接近的2的幂次。若$\Delta_a=\Delta_b$，优先选择$b$可确保对更大规模数据的支持。由于对数计算、取整、幂运算以及加减和比较运算均可在常数时间内完成，因此时间复杂度为$O(1)$。因为$q'=2^k$，所以模q'的操作可以通过按位与来实现：$x\bmod q'=x\&\left(q'-1\right)$，其中$q'-1=2^k-1$是一个全1的二进制编码。

(3)舍入计算：$h(x)_i=\dfrac{p}{q'}\cdot h'_2(x)_i$，其中$p=2^{k'}$是舍入模数，是$q'$的因子。此时舍入操作可以通过按位右移来完成。例如$\dfrac{p}{q'}=\dfrac{2^{k'}}{2^k}=2^{k'-k}$，这样舍入操作可以表示为$\dfrac{p}{q'}\cdot x=x>>\left(k-k'\right)$，其中>>表示按位右移。

最终输出结果$h(x)$，定义在模数p下。

接下来对以上三种方案以及其他研究中各项操作的时间复杂度、存储需求进行对比，对比结果如表1所示。假设d为多项式阶数。

综上，如果追求多项式乘法的极致效率且对模约化和舍入操作的常规效率可接受，可以选择方案a；如果不希望固定模数为NTT形式并且更看重模约化和舍入的高效位操作，则推荐使用方案b；如果希望兼顾多项式乘法效率和模约化、舍入的高效性，则推荐使用方案c。并且与其他方案相比，方案c在时间复杂度以及存储需求方面表现较好。

5　证　明

定理1.　方案具有完全完备性.

完全完备性意味着如果一个陈述是真实的，那么诚实证明者总能说服验证者这个陈述是真实的。即如果证明者拥有满足约束关系的秘密，且按协议步骤执行，验证者将始终接受生成的承诺和证明。

证明. 假设证明者遵循了协议中的所有步骤，给出了有效的承诺和证明。验证者根据公开信息和挑战值，进行逐步验证：

(1)范数检查：

由于证明者多次重复进行拒绝采样，以高概率

表1 选取不同参数方案的时间复杂度、存储需求的对比

方案	多项式乘法	模约化		模数转换	舍入计算	误差修正	总时间复杂度	存储需求
		模多项式	模整数					
a	$O(d\log d)$ (NTT)	$O(d)$ (减法)	省略	无	$O(d)$ (先加法再移位)	$O(d)$	$O(d\log d)$	大(NTT模数较大，中间结果存储量高)
b	$O(d^{1.585})$(Karatsuba)或$O(d^{1.465})$(Toom-Cook3)	$O(d)$ (减法)	$O(d)$ (按位与)	无	$O(d)$ (移位)	无	$O(d^{1.585})$(Karatsuba)或$O(d^{1.465})$ (Toom-Cook3)	小(q,p均为2的幂次，存储友好)
c	$O(d\log d)$ (NTT)	$O(d)$ (减法)	$O(d)$ (按位与)	$O(d)$ (系数映射)	$O(d)$ (移位)	无	$O(d\log d)$	中(需临时存储NTT域乘法结果，模数转换后存储量降低，整体较平衡)
文献[15]	$O(d^{1.585})$ (Karatsuba/Toom-Cook)	$O(d)$ (减法)	$O(d)$ (按位与)	$O(d)$	$O(d)$ (比特截取)	$O(d)$	$O(d^{1.585})$	较小(模数为2的幂次，密钥和密文大小紧凑)
文献[25]	$O(d\log d)$ (FFT)	$O(d\log d)$	$O(d)$	$O(d)$	$O(d)$	$O(d^2)$ (重线性化)	$O(d^2)$	大(需存储评估密钥，存储开销随安全参数呈多项式增长)

生成满足范数约束的$\boldsymbol{z}$和$\boldsymbol{f}$。因此范数检查通过。

(2)挑战值一致性：

证明者计算的挑战值c来自$(pp, mp, ulp, t, \boldsymbol{F})$，验证者根据类似的输入计算$c'$。协议假设证明者和验证者使用的是同一个确定性哈希函数$\mathcal{H}$，因此只要$\boldsymbol{F}=\boldsymbol{F}'$，就有$c=c'$。

验证阶段首先计算$\boldsymbol{F}'=\boldsymbol{Az}+\boldsymbol{Bf}-c\cdot(q/p)\cdot\boldsymbol{h}$，将$\boldsymbol{f}=\boldsymbol{y}+c\cdot\boldsymbol{e}, \boldsymbol{z}=\boldsymbol{j}+c\cdot\boldsymbol{w}$代入上式后可以得到$\boldsymbol{F}'=\boldsymbol{A}(\boldsymbol{j}+c\cdot\boldsymbol{w})+\boldsymbol{B}(\boldsymbol{y}+c\cdot\boldsymbol{e})-c\cdot(q/p)\cdot\boldsymbol{h}$，计算后展开得到$\boldsymbol{F}'=\boldsymbol{Aj}+\boldsymbol{By}+c(\boldsymbol{Aw}+\boldsymbol{Be}-(q/p)\cdot\boldsymbol{h})$。又因为$(\boldsymbol{A},\boldsymbol{B})=(\boldsymbol{K}, -\boldsymbol{I}_{N+n})$，$\boldsymbol{e}=\boldsymbol{Kw}-(q/p)\cdot\boldsymbol{h}\bmod q$，故$c(\boldsymbol{Aw}+\boldsymbol{Be}-(q/p)\cdot\boldsymbol{h})=c(\boldsymbol{Kw}-\boldsymbol{e}-(q/p)\cdot\boldsymbol{h})=0$。因此$\boldsymbol{F}'=\boldsymbol{Aj}+\boldsymbol{By}=\boldsymbol{F}$。挑战值一致性成立。

(3)zk-SNARK验证：

π_N的验证隐含地检查了以下两组关系：

①秘密向量与误差向量之间的线性关系$\boldsymbol{e}=\boldsymbol{Kw}-(q/p)\cdot\boldsymbol{h}\bmod q$：证明者构造了满足上述关系的$(\boldsymbol{e}, \boldsymbol{w}, \vec{\boldsymbol{b}})$，并通过$\pi_N$对该关系生成零知识证明。由于zk-SNARK具有完备性，验证者总能接受这一证明。

②分解关系$\vec{\boldsymbol{e}}=\boldsymbol{G}\vec{\boldsymbol{b}}$：类似地，证明者提供了$\vec{\boldsymbol{b}}$使得$\vec{\boldsymbol{e}}$的每一维可以分解成$\boldsymbol{G}$的线性组合。此关系由$t$和$\pi_N$隐式证明。

由于zk-SNARK本身保证了完备性，因此验证者对π_N的验证一定会通过。

结论：如果证明者正确地生成t和π，验证者将总是接受，协议是完备的。

证毕。

定理2. 方案具有知识可靠性.

知识可靠性是比标准可靠性更强的概念，确保如果一个恶意证明者设法提出了一个可接受的证明，那么存在一个有效的提取器，在给定敌手源代码和随机硬币的情况下，可以有效地提取证据[25-26]。

证明. 使用标准倒带论证，得到两个可接受的协议输出$\pi=(\pi_N, (c, \boldsymbol{z}, \boldsymbol{f}))$，$\pi'=(\pi'_N, (c', \boldsymbol{z}', \boldsymbol{f}'))$，其中$c\neq c'$，输入均为$(pp, mp, ulp, t, \boldsymbol{F})$。

根据$\boldsymbol{F}'=\boldsymbol{Az}+\boldsymbol{Bf}-c\cdot(q/p)\cdot\boldsymbol{h}$，得到$\mathcal{R}_{q,d}$上$\bar{c}\cdot(q/p)\cdot\boldsymbol{h}=\boldsymbol{A}\bar{\boldsymbol{z}}+\boldsymbol{B}\bar{\boldsymbol{f}}$(!)，其中$\bar{c}=c-c'$，$\bar{\boldsymbol{z}}=\boldsymbol{z}-\boldsymbol{z}'$，$\bar{\boldsymbol{f}}=\boldsymbol{f}-\boldsymbol{f}'$。并且$\|\bar{\boldsymbol{f}}\|\leqslant 12\phi_e\eta_e$，$\|\bar{\boldsymbol{z}}\|\leqslant 12\phi\eta$成立。

假设存在一个提取器ε。在生成挑战$c\leftarrow\mathcal{H}(pp, mp, ulp, t, \boldsymbol{F})$之前，执行了具有绑定性的Pedersen承诺，概率多项式时间敌手无法找到两个不同的开口。因此当在两组转录本c和c'上运行ε时，返回的承诺开口都是相同的，除了可忽略不计的概率。考虑以上几点，使用ε来提取$\vec{\boldsymbol{eb}}^*=(\vec{\boldsymbol{y}}^*, \vec{\boldsymbol{e}}^*, \vec{\boldsymbol{b}}^*)\in\mathbb{Z}_q^{d(N+n)(k+2)}$，使得：

(1)$P(\vec{\boldsymbol{eb}}^*)=0\bmod q$对于所有$P\in mp$成立；

(2)$\boldsymbol{L}\cdot\begin{pmatrix}\vec{\boldsymbol{y}}^*\\ \vec{\boldsymbol{e}}^*\\ \vec{\boldsymbol{b}}^*\end{pmatrix}=\begin{pmatrix}\vec{\boldsymbol{f}}\\ \vec{\boldsymbol{0}}\end{pmatrix}\bmod q, ulp=\left(\boldsymbol{L}, \begin{matrix}\vec{\boldsymbol{f}}\\ \vec{\boldsymbol{0}}\end{matrix}\right)$，其中$\boldsymbol{L}=\begin{pmatrix}\boldsymbol{I}_{(N+n)d} & \boldsymbol{I}_{N+n}\otimes\mathrm{Rot}(c) & \boldsymbol{0}\\ \boldsymbol{0} & -\boldsymbol{I}_{(N+n)d} & \boldsymbol{G}\end{pmatrix}$；

(3)$\bigcirc_{i\in[2]}\left(\vec{\boldsymbol{b}}-\vec{i}\right)=0 \bmod q$，其中$\vec{i}=(i,\cdots,i)$。

(2)等价于$\mathcal{R}_{q,d}$上$\boldsymbol{f}=\boldsymbol{y}^*+c\cdot\boldsymbol{e}^*$(*)，(1)等价于$\mathbb{Z}_q$上$\vec{\boldsymbol{e}}^*=\boldsymbol{G}\vec{\boldsymbol{b}}^*$，其中$\boldsymbol{y}^*$和$\boldsymbol{e}^*$是对应于$\vec{\boldsymbol{y}}^*$、$\vec{\boldsymbol{e}}^*$的$\mathcal{R}_{q,d}$中的多项式向量。$q$是素数，根据乘法的零因子性质，可知至少有一个$b_i\equiv i \bmod q$，确保了在模$q$下存在特定的结构和解的唯一性，维度依旧是$d(N+n)k$，因此(3)暗示了$\vec{\boldsymbol{b}}^*\in[2]^{d(N+n)k}$。根据$\boldsymbol{G}$的结构和$\mathbb{Z}_q$上$\vec{\boldsymbol{e}}^*=\boldsymbol{G}\vec{\boldsymbol{b}}^*$，得到$\vec{\boldsymbol{e}}^*\in\left[q/p\right]^{d(N+n)}$。根据上述讨论，对于相同的证据$\vec{\boldsymbol{eb}}^*=\left(\vec{\boldsymbol{y}}^*,\vec{\boldsymbol{e}}^*,\vec{\boldsymbol{b}}^*\right)$，使用ε得到$\mathcal{R}_{q,d}$上$\boldsymbol{f}'=\boldsymbol{y}^*+c'\cdot\boldsymbol{e}^*$(**)。对于$\vec{\boldsymbol{eb}}^*=\left(\vec{\boldsymbol{y}}^*,\vec{\boldsymbol{e}}^*,\vec{\boldsymbol{b}}^*\right)$，将(*)式和(**)式代入(!)式进行计算可以得到$\bar{c}\cdot(q/p)\cdot\boldsymbol{h}=\boldsymbol{A}\bar{\boldsymbol{z}}+\boldsymbol{B}\bar{\boldsymbol{f}}=\boldsymbol{A}\bar{\boldsymbol{z}}+\bar{c}\boldsymbol{B}\boldsymbol{e}^*$。文献[12]表明，$\bar{c}$在$\mathcal{R}_{q,d}$中是可逆的，$(q/p)\cdot\boldsymbol{h}=\boldsymbol{A}\cdot(\bar{\boldsymbol{z}}/\bar{c})+\boldsymbol{B}\boldsymbol{e}^*$，因此提取器可以计算$\boldsymbol{w}^*=\bar{\boldsymbol{z}}/\bar{c} \bmod q$，即$(q/p)\cdot\boldsymbol{h}=\boldsymbol{K}\boldsymbol{w}^*-\boldsymbol{e}^*$并且$\vec{\boldsymbol{e}}^*\in\left[q/p\right]^{d(N+n)}$。关系②$R'_{rnd}$成立。又因为$\boldsymbol{h}\in R_{q,d}^{N+n}$，根据事实得到$\boldsymbol{h}=\lfloor\boldsymbol{K}\boldsymbol{w}\rceil_p$。因此关系①成立。因此关于$\left(\bar{c},\boldsymbol{e}^*,\boldsymbol{w}^*,\vec{\boldsymbol{b}}^*\right)$，$\mathcal{L}(mp,ulp)$成立。

证毕。

定理3.　方案具有零知识性.

零知识性保证了在证明过程中，验证者除了陈述的真实性外，无法获得任何有关陈述的额外信息。一个证明系统是零知识的，如果存在一个模拟器，它能在不使用私有信息的情况下生成与真实证明分布相同的输出。

证明. 通过一系列的游戏序列(见表2)，从真实证明者的游戏开始，到完全模拟游戏结束，证明每一阶段的输出在概率上与前一个实验无法区分。相邻两个Game之间的不可区分性分析如下：

(1)Game 1和Game 0的不可区分性：

①隐藏性：模拟器选择的$\boldsymbol{e}'$、$\vec{\boldsymbol{b}}'$、$\boldsymbol{w}'$、$\boldsymbol{j}'$、$\boldsymbol{y}'$是与真实向量的维度相同的随机向量，并且使用相同的承诺方案(即Pedersen承诺和Merkle树)，确保树根的计算方式与真实承诺结构一致，因此模拟器生成的承诺t''和真实承诺t在分布上是不可区分的。

②绑定性：承诺方案的绑定性保证了t只能对应唯一的$\boldsymbol{e}$、$\boldsymbol{w}$、$\vec{\boldsymbol{b}}$。因此Game 1的输出$\left(t'',\pi\right)$在统计上与Game 0的输出(t,π)无法区分。

(2)Game 2和Game 1的不可区分性：

模拟器通过随机生成挑战c'，并且计算响应操作基于与真实证明完全相同的数学结构。在真实证明中，挑战c通过哈希函数生成，哈希输入包含pp、mp, ulp以及承诺值，而模拟证明中的挑战c'是随机生成的。由于哈希函数的抗碰撞性和随机性，验证者无法区分挑战是随机生成的还是由哈希计算得出的，因此Game 2和Game 1具有不可区分性。

(3)Game 3和Game 2的不可区分性：

模拟器通过随机生成响应$\boldsymbol{f}'$和$\boldsymbol{z}'$，其分布与真实证明中的响应一致。因此Game 3和Game 2具有不可区分性。

(4)Game 4和Game 3的不可区分性：

zk-SNARK的零知识性保证了模拟证明π'_N和真实证明π_N在计算上不可区分，因此Game 4和Game 3具有不可区分性。

表2　零知识性证明的游戏序列

游戏名称	游戏目标	游戏步骤	游戏输出
Game 0：真实证明	真实证明者选择真实的$\boldsymbol{e}$、$\boldsymbol{w}$、$\vec{\boldsymbol{b}}$生成真实的承诺和证明。	1. 输入：pp、mp、ulp； 2. 选择私有的秘密向量$\boldsymbol{e}$、$\boldsymbol{w}$、$\vec{\boldsymbol{b}}$和掩蔽向量$\boldsymbol{j}$、$\boldsymbol{y}$； 3. 承诺生成：使用Pedersen承诺，利用均匀且独立选择的随机数向量$\boldsymbol{r}_y$、$\boldsymbol{r}_e$、$\boldsymbol{r}_b$分别为消息向量$\boldsymbol{y}$、$\boldsymbol{e}$、$\vec{\boldsymbol{b}}$计算承诺值t_y、t_e、t_b； 4. 构造Merkle树根：利用Merkle树技术，构造树根$t=R=\mathcal{H}(\mathcal{H}(L_1\|L_2)\|L_3)$，其中$L_1$、$L_2$、$L_3$分别是承诺值$t_y$、$t_e$、$t_b$的哈希； 5. 生成挑战：使用掩蔽向量计算$\boldsymbol{F}=\boldsymbol{A}\boldsymbol{j}+\boldsymbol{B}\boldsymbol{y}$，其中$(\boldsymbol{A},\boldsymbol{B})=(\boldsymbol{K},-\boldsymbol{I}_{N+n})$，并使用非交互式的方法，利用抗碰撞哈希函数生成挑战：$c\leftarrow\mathcal{H}(pp,mp,ulp,t,\boldsymbol{F})$； 6. 计算响应$\boldsymbol{f}=\boldsymbol{y}+c\cdot\boldsymbol{e}$，$\boldsymbol{z}=\boldsymbol{j}+c\cdot\boldsymbol{w}$； 7. 生成zk-SNARK证明：基于承诺值$t$和约束关系，使用zk-SNARK生成证明$\pi_N$。	承诺t和证明$\pi=\left(\pi_N,(c,\boldsymbol{z},\boldsymbol{f})\right)$

续表

游戏名称	游戏目标	游戏步骤	游戏输出
Game 1：随机生成掩蔽向量并生成伪造承诺(模拟器1)	证明通过随机化秘密信息生成的承诺和证明，与真实分布的不可区分性。	1. 输入：pp,mp,ulp； 2. 选择随机向量和掩蔽向量：模拟器选择随机的秘密向量$\boldsymbol{e}'$、$\vec{\boldsymbol{b}}'$、$\boldsymbol{w}'$和掩蔽向量$\boldsymbol{j}'$、$\boldsymbol{y}'$，这些向量的维度与真实证明中的$\boldsymbol{e}$、$\vec{\boldsymbol{b}}$、$\boldsymbol{w}$、$\boldsymbol{j}$、$\boldsymbol{y}$相同，但它们是完全随机的； 3. 承诺生成：利用Pedersen承诺，利用均匀且独立选择的随机数向量$\boldsymbol{r}'_y$、$\boldsymbol{r}'_e$、$\boldsymbol{r}'_b$分别为消息向量$\boldsymbol{y}'$、$\boldsymbol{e}'$、$\vec{\boldsymbol{b}}'$计算承诺值t'_y、t'_e、t'_b； 4. 构造Merkle树根：利用Merkle树技术，构造树根$t''=R'=\mathcal{H}(\mathcal{H}(L'_1\|L'_2)\|L'_3)$，其中$L'_1$、$L'_2$、$L'_3$分别是承诺值$t'_y$、$t'_e$、$t'_b$的哈希； 5. 生成挑战：使用掩蔽向量计算$\boldsymbol{F}=\boldsymbol{A}\boldsymbol{j}'+\boldsymbol{B}\boldsymbol{y}'$，其中$(\boldsymbol{A},\boldsymbol{B})=(\boldsymbol{K},-\boldsymbol{I}_{N+n})$，并使用非交互式的方法，利用抗碰撞哈希函数生成挑战：$\boldsymbol{c}\leftarrow\mathcal{H}(pp,mp,ulp,t,\boldsymbol{F})$； 6. 计算响应$\boldsymbol{f}=\boldsymbol{y}'+c\cdot\boldsymbol{e}'$，$\boldsymbol{z}=\boldsymbol{j}'+c\cdot\boldsymbol{w}'$； 7. 生成zk-SNARK证明：基于承诺值$t''$和约束关系，使用zk-SNARK生成证明$\pi_N$。	伪造承诺t''和证明$\pi=(\pi_N,(c,\boldsymbol{z},\boldsymbol{f}))$
Game 2：生成随机挑战(模拟器2)	证明通过随机生成的挑战值c'替换哈希生成的挑战值c后的实验仍然不可区分。	1. 输入：pp,mp,ulp； 2. 选择随机向量和掩蔽向量：模拟器选择随机的秘密向量$\boldsymbol{e}'$、$\vec{\boldsymbol{b}}'$、$\boldsymbol{w}'$和掩蔽向量$\boldsymbol{j}'$、$\boldsymbol{y}'$，这些向量的维度与真实证明中的$\boldsymbol{e}$、$\vec{\boldsymbol{b}}$、$\boldsymbol{w}$、$\boldsymbol{j}$、$\boldsymbol{y}$相同，但它们是完全随机的； 3. 承诺生成：利用Pedersen承诺，利用均匀且独立选择的随机数向量$\boldsymbol{r}'_y$、$\boldsymbol{r}'_e$、$\boldsymbol{r}'_b$分别为消息向量$\boldsymbol{y}'$、$\boldsymbol{e}'$、$\vec{\boldsymbol{b}}'$计算承诺值t'_y、t'_e、t'_b； 4. 构造Merkle树根：利用Merkle树技术，构造树根$t''=R'=\mathcal{H}(\mathcal{H}(L'_1\|L'_2)\|L'_3)$，其中$L'_1$、$L'_2$、$L'_3$分别是承诺值$t'_y$、$t'_e$、$t'_b$的哈希； 5. 选择随机挑战值：模拟器独立随机地选择一个随机挑战值c'来模拟真实证明中的挑战生成过程； 6. 计算响应$\boldsymbol{f}=\boldsymbol{y}'+c\cdot\boldsymbol{e}'$，$\boldsymbol{z}=\boldsymbol{j}'+c\cdot\boldsymbol{w}'$； 7. 生成zk-SNARK证明：使用zk-SNARK的生成算法，基于承诺值t''和约束关系，生成证明π_N。	伪造承诺t''和证明$\pi=(\pi_N,(c',\boldsymbol{z},\boldsymbol{f}))$
Game 3：随机选择响应(模拟器3)	证明在完全随机生成响应$\boldsymbol{f}$和$\boldsymbol{z}$的情况下，实验仍然不可区分。	1. 输入：pp,mp,ulp； 2. 选择随机向量和掩蔽向量：模拟器选择随机的秘密向量$\boldsymbol{e}'$、$\vec{\boldsymbol{b}}'$、$\boldsymbol{w}'$和掩蔽向量$\boldsymbol{j}'$、$\boldsymbol{y}'$，这些向量的维度与真实证明中的$\boldsymbol{e}$、$\vec{\boldsymbol{b}}$、$\boldsymbol{w}$、$\boldsymbol{j}$、$\boldsymbol{y}$相同，但它们是完全随机的； 3. 承诺生成：利用Pedersen承诺，利用均匀且独立选择的随机数向量$\boldsymbol{r}'_y$、$\boldsymbol{r}'_e$、$\boldsymbol{r}'_b$分别为消息向量$\boldsymbol{y}'$、$\boldsymbol{e}'$、$\vec{\boldsymbol{b}}'$计算承诺值t'_y、t'_e、t'_b； 4. 构造Merkle树根：利用Merkle树技术，构造树根$t''=R'=\mathcal{H}(\mathcal{H}(L'_1\|L'_2)\|L'_3)$，其中$L'_1$、$L'_2$、$L'_3$分别是承诺值$t'_y$、$t'_e$、$t'_b$的哈希； 5. 选择随机挑战值：模拟器独立随机地选择一个随机挑战值c'来模拟真实证明中的挑战生成过程； 6. 模拟器随机选择响应$\boldsymbol{f}'$和$\boldsymbol{z}'$，其分布与真实的$\boldsymbol{f}$和$\boldsymbol{z}$相同； 7. 生成zk-SNARK证明：使用zk-SNARK的生成算法，基于承诺值t''和约束关系，生成证明π_N。	伪造承诺t''和证明$\pi=(\pi_N,(c',\boldsymbol{z}',\boldsymbol{f}'))$
Game 4：构造模拟证明(完全模拟器)	证明在无真实参数输入情况下，生成的模拟证明仍不可区分。	1. 输入：pp,mp,ulp； 2. 选择随机向量和掩蔽向量：模拟器选择随机的秘密向量$\boldsymbol{e}'$、$\vec{\boldsymbol{b}}'$、$\boldsymbol{w}'$和掩蔽向量$\boldsymbol{j}'$、$\boldsymbol{y}'$，这些向量的维度与真实证明中的$\boldsymbol{e}$、$\vec{\boldsymbol{b}}$、$\boldsymbol{w}$、$\boldsymbol{j}$、$\boldsymbol{y}$相同，但它们是完全随机的； 3. 承诺生成：利用Pedersen承诺，利用均匀且独立选择的随机数向量$\boldsymbol{r}'_y$、$\boldsymbol{r}'_e$、$\boldsymbol{r}'_b$分别为消息向量$\boldsymbol{y}'$、$\boldsymbol{e}'$、$\vec{\boldsymbol{b}}'$计算承诺值t'_y、t'_e、t'_b； 4. 构造Merkle树根：利用Merkle树技术，构造树根$t''=R'=\mathcal{H}(\mathcal{H}(L'_1\|L'_2)\|L'_3)$，其中$L'_1$、$L'_2$、$L'_3$分别是承诺值$t'_y$、$t'_e$、$t'_b$的哈希； 5. 选择随机挑战值：模拟器独立随机地选择一个随机挑战值c'来模拟真实证明中的挑战生成过程； 6. 模拟器随机选择响应$\boldsymbol{f}'$和$\boldsymbol{z}'$，其分布与真实的$\boldsymbol{f}$和$\boldsymbol{z}$相同； 7. 构造模拟zk-SNARK证明π'_N。	伪造承诺t''和证明$\pi=(\pi'_N,(c',\boldsymbol{z}',\boldsymbol{f}'))$

由以上实验序列和分析可知Game 0和Game 4具有不可区分性，即真实证明和模拟证明之间不可区分，方案具有零知识性。

证毕。

定理4. 方案具有简洁性.

如果承诺大小和证明大小都是多项式级别的，并且验证算法的时间复杂度仅与公开信息有关，则方案具有简洁性。

证明. 总的证明输出$\delta=\left(t,\left(\pi_N,\left(c,\boldsymbol{z},\boldsymbol{f}\right)\right)\right)$的大小可以近似为：$|\delta|\approx|t|+|\pi_N|+|c|+|\boldsymbol{z}|+|\boldsymbol{f}|$。

其中t是Merkle树根，为哈希函数输出的固定比特长度，记为λ_1；π_N是zk-SNARK证明，证明是简洁且大小固定的，记为λ_2；挑战值c的大小等于哈希函数输出的比特长度λ_1；

$\boldsymbol{y}$和$\boldsymbol{j}$是从离散高斯分布中采样得到的，$|\boldsymbol{y}|=\log_2(12\phi_e\eta_e)$，$|\boldsymbol{j}|=\log_2(12\phi\eta)$，$|c\cdot\boldsymbol{e}|=\lambda_1+\log_2 q$，$|c\cdot\boldsymbol{w}|=\lambda_1+\log_2 q$。因此：

$$\begin{cases}|\boldsymbol{f}|=(N+n)\cdot\max\left(\log_2(12\phi_e\eta_e),\log_2 q\right)+\lambda_1\\|\boldsymbol{z}|=(m_1+m_2+r)\cdot\max\left(\log_2(12\phi\eta),\log_2 q\right)+\lambda_1\end{cases}。$$

总通信复杂度为：

$$(N+n)\cdot\max\left(\log_2(12\phi_e\eta_e),\log_2 q\right)+(m_1+m_2+r)\cdot\max\left(\log_2(12\phi\eta),\log_2 q\right)+4\lambda_1+\lambda_2,$$

其中，$(N+n)$、(m_1+m_2+r)、λ_1、λ_2都是多项式级别。

综上，承诺大小、证明大小以及通信复杂度均是多项式级别。

验证阶段包括：

(1)范数验证：

时间复杂度与$\boldsymbol{z}$和$\boldsymbol{f}$的维度有关，因此为$O(m_1+m_2+r+N+n)$。

(2)线性关系验证：

计算$\boldsymbol{F}'=\boldsymbol{Az}+\boldsymbol{Bf}-c\cdot(p/q)\cdot\boldsymbol{h}$：

其中矩阵-向量乘法$\boldsymbol{Az}$计算的时间复杂度为$O\big((N+n)(m_1+m_2+r)\big)$；$\boldsymbol{Bf}-c\cdot(p/q)\cdot\boldsymbol{h}$中，$\boldsymbol{B}$是负单位矩阵，因此计算$\boldsymbol{Bf}$的复杂度为$O(N+n)$；$c\cdot(p/q)\cdot\boldsymbol{h}$是标量乘法，复杂度为$O(N+n)$。

因此这一阶段的总时间复杂度为：$O\big((N+n)(m_1+m_2+r)\big)+2O(N+n)=O\big((N+n)(m_1+m_2+r)\big)$。

(3)挑战值一致性验证：

哈希函数的时间复杂度与输入的总长度有关，输入长度L_H大小可估算为与N、n、m_1、m_2、r的线性函数相关，记为$O(L_H)$。

检查一致性的复杂度为$O(1)$。

因此这一阶段的总复杂度为$O(L_H)$。

(4)zk-SNARK有效性验证：

验证π_N是否为有效证明。zk-SNARK验证的复杂度与电路大小(即公开信息的维度)相关，假设为$O(S)\approx O(N+n)$，其中S表示电路的复杂度，通常与$\boldsymbol{e}$的维数$n+N$相关。

因此验证算法的总时间复杂度为：

$O(N+n+m_1+m_2+r)+O\big((N+n)(m_1+m_2+r)\big)+O(L_H)+O(S)$。

可以看到，验证算法的时间复杂度与公开信息(如$\boldsymbol{K}$、$\boldsymbol{h}$)的维度$N+n$、m_1+m_2+r有关，与秘密数据(如$\boldsymbol{w}$、$\boldsymbol{e}$的具体值)无关。因此方案具有简洁性。

证毕。

定理5. 方案具有完美正确性.

完美正确性确保协议中每一个步骤和计算都符合预期，并且生成的证明与声明的一致。诚实的证明者严格按照协议步骤生成承诺t和证明π；验证者依照协议步骤验证(t,π)，所有计算均满足公开关系，且验证通过。

证明. 需要证明每一个步骤的正确性：

(1)承诺生成：证明者严格按照协议步骤计算，则t是有效承诺。

(2)证明生成：证明者严格执行挑战值生成、掩蔽值计算、zk-SNARK生成流程，则π是有效证明。

(3)验证过程：验证者严格执行范数检查、挑战值一致性检查、zk-SNARK验证。

如果证明者和验证者分别进行诚实计算和验证，且验证通过，则方案具有完美正确性。

证毕。

定理6. 方案具有模拟可提取性.

标准知识可靠性只能保证，生成有效证明的证明者掌握相关证据，但无法防止证明被篡改或伪造。因此引入模拟知识可靠性[27]，也称为模拟可提取性，是知识可靠性的扩展版本，要求在零知识证明系统中，若证明者能生成验证通过的证明，则存在提取器可从中提取出有效证据，这意味着即使敌手得到了任意数量的模拟证明，他也不能生成新的证明，除非他知道证据[28]。核心要求是：存在一个模拟器和一个提取器，使得即使敌手与模拟器进行交互并试图伪造有效证明，也必须存在一个提取器算法，它能从敌手的行为和模拟证明中提取出对应的秘密证据，并且成功概率不可忽略。

[GM17][29]指出，zk-SNARK本身不直接提供模拟可提取性，因此本方案通过对zk-SNARK的增强设计，确保了模拟可提取性的实现。具体措施如下：

(1)Fiat-Shamir变换的使用：

原本zk-SNARK交互式协议中，恶意证明者可

能影响挑战值绕过模拟可提取性要求。通过Fiat-Shamir变换将其转为非交互式协议，利用哈希函数生成挑战值c。哈希函数的单向性、抗碰撞性使挑战值不可预测且唯一，恶意证明者无法操控，模拟器可借此模拟验证者行为，按规则提取秘密信息。

(2)唯一性标识与随机数绑定：

在承诺阶段引入唯一性标识(如时间戳或会话标识)和随机数生成机制，确保每个证明只能在特定环境中使用。这一设计有效防止了重放攻击，并增强了证明的不可重复性。模拟器能准确模拟验证过程，提取秘密信息。

(3)Pedersen承诺与公开参数的结合：

通过使用Pedersen承诺来确保私有向量和公开参数之间的强绑定关系，采用Merkle树减少公开承诺值的大小。这使得模拟器无法通过伪造承诺或篡改证明来逃避对秘密信息的提取，确保模拟器按照承诺与私有向量的绑定关系合法提取，实现模拟可提取性。

证明.

(1)构造模拟器$\mathcal{S}$

目标：模拟器$\mathcal{S}$在未知$(\boldsymbol{w},\boldsymbol{e})$的情况下，生成与真实承诺和证明不可区分的承诺$t$和证明$\pi$。

输入：接收公共参数pp和公开值$(\boldsymbol{K},\boldsymbol{h})$。

①模拟承诺：模拟器随机选择私密值$\vec{\boldsymbol{y}}$、$\vec{\boldsymbol{e}}$、$\vec{\boldsymbol{b}}$以及随机数向量，计算t_y、t_e、t_b；

②模拟Merkle树根：使用公开信息和t_y、t_e、t_b构建Merkle树根R，作为承诺t；

③模拟证明：随机选择挑战值c，并计算$\boldsymbol{z}$、$\boldsymbol{f}$；基于zk-SNARK的零知识性质，生成一个模拟证明π_N，证明以下关系：$\boldsymbol{f}=\boldsymbol{y}+c\cdot\boldsymbol{e}$、$\vec{\boldsymbol{e}}=\boldsymbol{G}\vec{\boldsymbol{b}}$。

输出：合法的承诺t和完整的模拟证明$\pi=\left(\pi_N,\left(c,\boldsymbol{z},\boldsymbol{f}\right)\right)$。

由于zk-SNARK的零知识性，验证者无法区分π_N是基于真实秘密生成，还是由模拟器生成。且随机生成的t在验证者视角下分布一致。

(2)构造提取器$\mathcal{E}$

目标：提取器从验证通过的证明中提取出有效的证据$\boldsymbol{w}$、$\boldsymbol{e}$，并验证它们是否满足所需的关系。

①提取器的能力：提取器$\mathcal{E}$可以通过zk-SNARK提取机制，访问证明者生成证明时的所有内部状态；

②提取步骤：

提取器运行zk-SNARK提取算法，从证明π_N中提取$\boldsymbol{w}$、$\boldsymbol{e}$；

提取器验证$\boldsymbol{w}$、$\boldsymbol{e}$是否满足约束关系：$\boldsymbol{e}=\boldsymbol{K}\boldsymbol{w}-(q/p)\cdot\boldsymbol{h}\bmod q\wedge\vec{\boldsymbol{e}}\in\left[q/p\right]^{(N+n)d}$。这是证明的核心约束，保证了提取出的证据是有效的，且符合要求。如果$\boldsymbol{w}$、$\boldsymbol{e}$不满足关系，则证明π是无效的。

③提取器的成功性：zk-SNARK的知识可靠性保证，如果π通过验证，则一定能够提取出满足约束的$\boldsymbol{w}$、$\boldsymbol{e}$。

(3)构造游戏序列证明模拟可提取性

目标：通过一系列的游戏序列(见表3)，证明模拟可提取性，即证明通过模拟的承诺和证明无法与真实的承诺和证明区分，并且提取器能够从任何有

表3 模拟可提取性证明的游戏序列

游戏名称	游戏步骤与分析	游戏结论
Game 0：真实协议执行	证明者根据方案的步骤生成t和π，验证者根据验证算法进行验证。	验证者接受证明的概率$\Pr[\text{Game0}]$
Game 1：替换为模拟器生成的t和π	使用模拟器$\mathcal{S}$生成的承诺和证明替换实际的t和π。zk-SNARK的零知识性质和承诺的不可伪造性保证模拟器生成的t和π与真实分布在多项式时间内不可区分。	$\left\lvert\Pr[\text{Game0}]-\Pr[\text{Game1}]\right\rvert\leqslant\text{negl}(\lambda)$
Game 2：引入提取器对zk-SNARK证明进行提取	在Game 1基础上，引入提取器$\mathcal{E}$，对验证通过的π_N提取$\boldsymbol{w}$、$\boldsymbol{e}$。zk-SNARK的知识可靠性保证，如果π_N是有效证明，则提取器一定能够提取满足关系的$\boldsymbol{w}$、$\boldsymbol{e}$。因此Game 1和Game 2的成功概率是相同的。	$\Pr[\text{Game1}]=\Pr[\text{Game2}]$
Game 3：敌手伪造zk-SNARK证明	假设敌手$\mathcal{A}$试图生成一个伪造证明π'使得通过验证但无法提取出有效证据$\boldsymbol{w}$、$\boldsymbol{e}$。哈希函数$\mathcal{H}$的抗碰撞性确保$\mathcal{A}$无法伪造挑战c；zk-SNARK的知识可靠性保证，敌手$\mathcal{A}$无法生成满足验证要求的证明π_N'且不提供有效证据$\boldsymbol{w}$、$\boldsymbol{e}$。因此，敌手伪造π'的成功概率是可忽略的。	$\left\lvert\Pr[\text{Game3}]-\Pr[\text{Game2}]\right\rvert\leqslant\text{negl}(\lambda)$
Game 4：敌手伪造承诺值	假设敌手试图生成一个伪造承诺值$\tilde{t}$，使得$\tilde{t}$通过验证，但无法对应一个合法的证据。Merkle树的完整性和哈希函数$\mathcal{H}$的抗碰撞性保证承诺值不可伪造；Pedersen承诺的绑定性保证敌手无法伪造合法的承诺值$\tilde{t}$。因此，敌手伪造的成功概率是可忽略的。	$\left\lvert\Pr[\text{Game4}]-\Pr[\text{Game3}]\right\rvert\leqslant\text{negl}(\lambda)$
Game 5：完整性验证	提取器的成功性保证提取出的$\boldsymbol{w}$、$\boldsymbol{e}$一定满足约束关系。	$\Pr[\text{Game5}]=\Pr[\text{Game4}]$

效证明中提取出有效证据。

最终结论：$\Pr[\text{Game0}]=\Pr[\text{Game5}]+\text{negl}(\lambda)$。

这表明敌手无法区分模拟生成的t、π与真实生成，且提取器能够可靠地提取有效证据，因此该方案满足模拟可提取性。

证毕。

定理7. 方案具有绑定性.

绑定性确保了一个诚实的证明者不可能在不同的运行中使用相同的证明来证明两个不同的陈述。

证明. 由于Pedersen承诺的安全性基于离散对数问题，证明者在不破坏离散对数假设的情况下，无法找到两组不同的$\left(\vec{\boldsymbol{y}},\vec{\boldsymbol{e}},\vec{\boldsymbol{b}}\right)$满足$\boldsymbol{y}\cdot\boldsymbol{g}+\boldsymbol{r}\cdot\boldsymbol{h}=\boldsymbol{y}'\cdot\boldsymbol{g}+\boldsymbol{r}'\cdot\boldsymbol{h}\bmod q_1$；

利用抗碰撞哈希函数确保Merkle树中所有节点不可伪造，从而绑定承诺值。因此t和way保证了承诺值的唯一性，防止证明者作弊。

证毕。

6 结　论

本方案提出了一种更高效、更安全的基于LWR问题的非交互式零知识证明协议。与Esgin等人提出的零知识方案[12]相比，我们方案优势如下：

(1)方案流程完整，逻辑严谨：本方案给出了完整的基于承诺-证明框架的完整零知识证明流程，方便理解和研究。

(2)安全性增强：通过在秘密向量的线性关系之外引入额外的约束条件，确保只有同时满足这两种关系时，秘密向量才被视为合法。这种双重约束机制可以提高攻击者破解系统的难度，从而增强了协议整体的安全性。

(3)数学表示简化：本方案将线性关系和约束关系整合成一个综合方程，使得繁杂的线性方程组可以以简洁的矩阵-向量乘积形式表示，可以涵盖多种类型的关系。这种简化不仅提升了方案的灵活性和扩展性，还使得方案框架更易于理解和实现。

(4)承诺机制优化：方案[12]采用的BDLOP承诺产生的承诺值较大，而本方案采用可扩展输出函数和Pedersen承诺来生成中间承诺值，再基于这些中间值构建Merkle树，递归生成哈希树根作为紧凑的公开承诺值，有效减小公开承诺值的大小，实现存储与通信资源的高效利用；承诺阶段不再需要显式使用消息向量，而是对误差向量、掩蔽值以及二进制分解值分别进行承诺，确保消息向量、辅助向量等不被泄露，其中承诺机制中对于随机数生成和生成元向量的生成进行优化，增强安全性。

(5)证明阶段改进：基于zk-SNARK协议以非交互方式生成证明，确保承诺值满足约束关系，并且降低了交互带来的通信开销，适用于区块链、隐私交易等异步场景。

(6)多种舍入计算方案：方案[12]虽提供了高效的证明框架，但在处理大规模数据或高维问题时，可能效率不高。本方案则引入适合不同形式模数和舍入模数的计算方案，并提出了一种新型混合模数方案，通过保留NTT加速多项式乘法的优势，并优化了模约化和舍入操作，从而提高了整体计算效率。

(7)安全性分析全面：从零知识性、模拟可提取性、简洁性等方面对方案进行了理论分析与证明，确保方案的安全性和高效性。

与现有的格密码方案相比，本方案在证明尺寸和计算复杂性方面表现出显著优势，为后量子安全密码系统的设计与实现提供了新的思路与方法。

在接下来的工作中，将着重研究以下两个方面：(1)随着互联网时代的快速发展，区块链规模可能会有所增长。因此，未来需要深入研究本方案是否足够高效，以应对区块链规模的增长，同时确保能够稳定提供零知识性的隐私保护；(2)为了有效控制噪声，目前方案中选用了向下取整的舍入策略，后续可以探索向上取整或向最近整数取整的可能性。具体来说，舍入策略由向下取整变为向上取整或向最近整数取整，误差区间相应地由$[0,\Delta)$分别变为$(-\Delta,0]$或$[-\Delta/2,\Delta/2]$，这会直接影响误差概率分布、参数设置和多项式约束的构造，从而影响安全性证明和计算效率。因此需进行严谨分析与参数调优，以评估其在降低噪声概率以及提升整体效率方面的潜力。

致　谢　感谢对本文提出建议的所有评审专家。

参　考　文　献

[1] Peikert C, Shiehian S. Noninteractive zero knowledge for np from (plain) learning with errors//Proceedings of the 39th Annual International Cryptology Conference. Santa Barbara, USA, 2019: 89-114

[2] Lyubashevsky V, Nguyen N K, Seiler G. Shorter lattice-based zero-knowledge proofs via one-time commitments// Proceedings of the 24th IACR International Conference on Practice and Theory of Public Key Cryptography. Virtual,

2021: 215-241

[3] Lyubashevsky V, Nguyen N K, Plancon M. Lattice-based zero-knowledge proofs and applications: shorter, simpler, and more general//Proceedings of the 42nd Annual International Cryptology Conference. Santa Barbara, USA, 2022: 71-101

[4] Cheng L. Zero-knowledge proofs based on lattices [Master's Thesis]. Xidian University, Xi'an, Shaanxi, 2021(in Chinese)
(程磊. 基于格的零知识证明[硕士学位论文]. 西安电子科技大学,陕西西安,2021)

[5] Silva R, Campello AC de A Jr, Dahab R. Lwe-based identification schemes//Proceedings of the Information Theory Workshop. Porto, Portugal, 2011: 292-296

[6] Kim S, Wu DJ. Multi-theorem preprocessing nizks from lattices//Proceedings of the 38th Annual International Cryptology Conference. Santa Barbara, USA, 2018: 733-765

[7] Baum C, Bootle J, Cerulli A, et al. Sub-linear lattice-based zero-knowledge arguments for arithmetic circuits//Proceedings of the 38th Annual International Cryptology Conference. Santa Barbara, USA, 2018: 669-699

[8] Yang N, Tian YL. Review of cryptoscheme construction based on lwe problem. Journal of Guizhou University (Natural Sciences), 2022, 39 (03): 93-101, 124 (in Chinese)
(杨楠,田有亮. 基于LWE问题构造密码方案综述. 贵州大学学报(自然科学版), 2022,39(03):93-101,124)

[9] Kim D, Lee D, Seo J, et al. Toward practical lattice-based proof of knowledge from hint-mlwe//Proceedings of the 43rd Annual International Cryptology Conference. Santa Barbara, USA, 2023: 549-580

[10] Okada H, Takayasu A, Fukushima K, et al. A compact digital signature scheme based on the module-lwr problem//Proceedings of the Information and Communications Security. Tokyo, Japan, 2020: 73-90

[11] Alwen J, Krenn S, Pietrzak K, et al. Learning with rounding revisited: new reduction, properties and applications//Proceedings of the 33rd Annual International Cryptology Conference. Santa Barbara, USA, 2013: 57-74

[12] Esgin M F, Steinfeld R, Liu D, et al. Efficient hybrid exact/relaxed lattice proofs and applications to rounding and vrfs//Proceedings of the 43rd Annual International Cryptology Conference. Santa Barbara, USA, 2023: 484-517

[13] Yang RP, Au MH, Lai JZ, et al. Lattice-based techniques for accountable anonymity: composition of abstract stern's protocols and weak prf with efficient protocols from lwr. IACR Cryptology ePrint Archive, 2017(781):1-62

[14] Ducas L, Kiltz E, Lepoint T, et al. Crystals-dilithium: a lattice-based digital signature scheme. IACR Transactions on Cryptographic Hardware and Embedded Systems, 2018, 2018(1): 238-268

[15] D'Anvers J, Karmakar A, Roy S S, et al. Saber: module-lwr based key exchange, cpa-secure encryption and cca-secure kem//Proceedings of the 10th International Conference on Cryptology in Africa. sMarrakesh, Morocco, 2018: 282-305

[16] Lyubashevsky V. Lattice signatures without trapdoors//Proceedings of the 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques. Cambridge, UK, 2012: 738-755

[17] Banerjee A, Peikert C, Rosen A. Pseudorandom functions and lattices//Proceedings of the 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques. Cambridge, UK, 2012: 719-737

[18] MATZOV. Report on the security of lwe: improved dual lattice attack. 2022. Available at https://doi. org/10.5281/zenodo.6412487

[19] Ducas L, Pulles L N. Does the dual-sieve attack on learning with errors even work?//Proceedings of the Annual International Cryptology Conference. Cham, Switzerland, 2023: 37-69

[20] Alkim E, Ducas L, Pöppelmann T, et al. Post-quantum key exchange—a new hope//Proceedings of the 25th USENIX Security Symposium. Austin, USA, 2016:327-343

[21] Hamlin B, Song F. Quantum security of hash functions and property-preservation of iterated hashing//Proceedings of the 10th International Conference on Post-Quantum Cryptography. Chongqing, China, 2019: 329-349

[22] Wang FX, Cohney S, Bonneau J. Sok: trusted setups for powers-of-tau strings. IACR Cryptology ePrint Archive, 2025 (064):1-30

[23] Nikolaenko V, Ragsdale S, Bonneau J, et al. Powers-of-tau to the people: decentralizing setup ceremonies//Proceedings of the 22nd International Conference on Applied Cryptography and Network Security. Abu Dhabi, United Arab Emirates, 2024: 105-134

[24] Jiang ZM, Zhou YB, Zhang R. General implementation algorithms and parameter selection methods for rounding learning over rings and rounding learning over Modules. Chinese Journal of Computers, 2022, 45 (6): 1326-1347 (in Chinese)
(姜子铭,周永彬,张锐. 环上舍入学习和模上舍入学习的通用实现算法与参数选取方法. 计算机学报,2022,45(6):1326-1347)

[25] Luo F, Wang F, Wang K, et al. Fully homomorphic encryption based on the ring learning with rounding problem. IET Information Security, 2019, 13(6): 639-648

[26] Groth J. On the size of pairing-based non-interactive arguments//Proceedings of the 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques. Vienna, Austria, 2016: 305-326

[27] Chase M, Lysyanskaya A. On signatures of knowledge//Proceedings of the 26th Annual International Cryptology Conference. Santa Barbara, California, USA, 2006: 78-96

[28] Atapoor S, Baghery K. Simulation extractability in groth's zk-snark//Proceedings of the 14th International Workshop on Data Privacy Management, the 24th European Symposium on Research in Computer Security, and the 3rd International Workshop on Cryptocurrencies and Blockchain Technology. Luxembourg, 2019: 336-354

[29] Groth J, Maller M. Snarky signatures: minimal signatures of knowledge from simulation-extractable snarks//Proceedings of the 37th Annual International Cryptology Conference. Santa Barbara, USA, 2017: 581-612

YIN Si-Tong, master candidate. Her main research areas include blockchain, zero-knowledge proof, and privacy protection.

GAO Jun-Tao, Ph. D., associate professor. His main research fields include the application of quantum computing in crypt analysis.

LI Xue-Lian, Ph. D., associate professor. Her main research fields are information security and blockchain.

Background

With the rapid development of post-quantum cryptography, lattice-based cryptosystems have become a major focus in the field of cryptography due to their exceptional security and resistance to quantum attacks. Constructing efficient and secure cryptographic protocols based on lattice-hard problems, particularly zero-knowledge proof schemes, plays a crucial role in applications such as privacy protection in blockchain.

Although zero-knowledge proof schemes based on the Learning With Errors (LWE) problem have made some progress, they still face numerous challenges, including low computational efficiency, large sizes of commitment and proof values, and reliance on Gaussian noise sampling. In lattice-based cryptographic problems, the presence of noise is essential, while the Learning With Rounding (LWR) problem generates errors by rounding the computation results, thus avoiding explicit sampling of random errors and further improving computational efficiency. This characteristic makes LWR a promising alternative to LWE.

Inspired by this, we propose using LWR as the underlying problem to construct a zero-knowledge proof scheme. Currently, research on zero-knowledge proofs based on LWR is limited, with recent studies focusing on using rounding problems to construct pseudorandom number generators. However, existing schemes still face challenges such as large commitment and proof sizes, which not only increase storage overhead but may also reduce the feasibility of practical deployment. Furthermore, performance is suboptimal when performing large-scale high-dimensional matrix-vector multiplications.

To address these issues, we propose an improved zero-knowledge proof scheme based on the LWR problem, utilizing the Commit-and-Prove framework. To enhance computational efficiency, we integrate multiple constraint relations into a composite equation and leverage the rounding mechanism to convert the constraints on the secret vector into controls and constraints on the error vector. Additionally, we introduce hash tree techniques, using the root of the hash tree as a public commitment value and combining zk-SNARKs to non-interactively generate proofs. As a result, the publicly revealed commitment and proof values are sufficiently compact, thus reducing storage and transmission overhead. At the same time, we control the noise introduced by the rounding operation to minimize its impact on computational accuracy.

Furthermore, to improve computational performance, we analyze the effect of modulus and rounding modulus selection on computational efficiency and propose a hybrid modulus scheme, ensuring that all three basic operations in the rounding computation achieve optimal performance. Additionally, we explore efficient polynomial multiplication acceleration algorithms to address computational bottlenecks in high-dimensional matrix-vector multiplications.

In summary, the proposed scheme demonstrates superior performance in terms of computational cost, communication cost, and storage requirements, reducing the resource consumption in practical deployment and enhancing its applicability.