CDBS：基于CRYSTALS-Dilithium算法的盲签名方案

杨亚涛[1,2]，常鑫[2]，史浩鹏[2]，王伟[2]，王克[1]
（1.北京电子科技学院电子与通信工程系，北京 100070；2.西安电子科技大学通信工程学院，陕西 西安 710071）

摘 要：为了应对传统盲签名方案在用户端、签名方和验证者交互过程中无法抵御量子计算攻击的这一难题，以NIST选定的后量子数字签名算法CRYSTALS-Dilithium为基础框架，设计了一种新型抗量子计算攻击的盲签名方案CDBS。整体方案采用Fiat-Shamir签名结构，包括密钥生成、盲化、签名、去盲和验证5个阶段，方案内部结合拒绝采样技术防止密钥泄露，使用NTT算法优化多项式计算以提高签名和验证效率。分析表明，所提方案安全性依赖于模误差学习（MLWE）问题和小整数解（SIS）问题，同时满足正确性、盲性和不可伪造性。与其他基于格的盲签名方案相比，所提方案具有较高的安全性，且签名生成过程更高效，占用开销更小。在相同样本参数设置下，所提方案整体开销仅为MBS方案的67.1%。经软件测试验证，实现一次完整的盲签名和验证过程平均仅需657.65 μs。所提方案为CRYSTALS-Dilithium数字签名算法的拓展应用提供了参考。

关键词：盲签名；数字签名；CRYSTALS-Dilithium；格；后量子密码
中图分类号：TN92
文献标志码：A
DOI：10.11959/j.issn.1000–436x.2024129

CDBS: blind signature scheme based on CRYSTALS-Dilithium algorithm

YANG Yatao[1,2], CHANG Xin[2], SHI Haopeng[2], WANG Wei[2], WANG Ke[1]
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: In order to solve the inadequacy of traditional blind signature schemes in resisting quantum computing attacks during interactions between users, signers, and verifiers was addressed, a novel quantum-resistant blind signature scheme named CDBS was proposed. This scheme aimed to enhance security and could resist quantum computing attacks. The scheme was based on the post-quantum digital signature algorithm CRYSTALS-Dilithium, selected by national institute of standards and technology (NIST). The overall scheme adopted the Fiat-Shamir signature structure, which comprised five stages, such as key generation, blinding, signing, unblinding and verification. The scheme incorporated rejection sampling technology to prevent key leakage and used NTT algorithm to optimise polynomial computation, thereby enhancing the efficiency of both signature generation and verification. Security analysis indicated that the scheme satisfied correctness, blindness, and unforgeability based upon module learning with error (MLWE) and small integer solution (SIS) problems. The scheme showed higher security and more efficient signature generation processes with smaller overhead than other lattice-based blind signature schemes. Given the same parameters, the overall computing overhead in the scheme was only 67.1% of the MBS scheme. Through software testing, an average time was only 657.65 μs to complete one blind signature and verification process. This work provided a valuable reference for the extended application on the CRYSTALS-Dilithium digital signature algorithm.

Keywords: blind signature, digital signature, CRYSTALS-Dilithium, lattice, post quantum cryptography

收稿日期：2024–02–02；**修回日期**：2024–06–17
基金项目：北京市自然科学基金资助项目(No.4232034)；中央高校基本科研业务费专项资金资助项目(No.3282024052, No.3282024058)
Foundation Items：Beijing Natural Science Foundation (No.4232034), The Fundamental Research Funds for the Central Universities (No.3282024052, No.3282024058)

0　引言

Chaum[1]于1983年首次提出盲签名（BS, blind signature），并设计了首个基于RSA算法和因子分解问题的盲签名方案。作为一种非常规的数字签名算法，盲签名允许用户对原始消息进行盲化处理，以实现对签名信息的保护，从而使其具有防追踪特性，即签名者无法将签名内容和签名结果相互对应。1995年，Camenisch等[2]提出了一个基于离散对数问题的盲签名方案。1996年，Pointcheval等[3]提出了盲签名的2个安全性定义：盲性和不可伪造性。同年，Abe等[4]提出了部分盲签名的概念。2010年，Rückert[5]提出了首个基于格的盲签名方案，该方案整体基于Lyubashevsky识别方案[6]和Fiat-Shamir构造模式[7]。

随着Shor’s算法[8]和Grover算法[9]等量子算法被相继提出，以及受量子计算机强大计算能力的威胁，目前基于传统方式构造的密码算法已不再安全。美国国家标准与技术研究所（NIST, national institute of standards and technology）于2016年启动了Post-Quantum密码学[10]标准化征集活动，其主要聚焦于2类后量子密码（PQC, post quantum cryptography）算法方案的征集，分别为公钥加密和数字签名。目前，NIST已经完成了第3轮后量子密码算法的标准化过程，并新增了4种算法进入第4轮筛选。

表1　NIST现有PQC算法

方案	算法	类型	特点
公钥加密	CRYSTALS-KYBER	格	开销小、效率高、安全性高
	BIKE	编码	开销大、效率低
	Classic McEliece	编码	密文小、公钥大、效率低
	HQC	编码	开销大、效率低
	SIKE	同源	开销小、效率低、不安全
数字签名	CRYSTALS-Dilithium	格	开销小、效率高、安全性高
	FALCON	格	开销小、效率高、安全性高
	SPHINCS+	哈希	开销小、效率高、安全性低

2022年，NIST宣布了4种待标准化的后量子密码算法，如表1所示。其中，CRYSTALS-KYBER[11]是基于格的公钥加密算法，CRYSTALS-Dilithium[12]和FALCON[13]是基于格的数字签名算法，SPHINCS+[14]是基于哈希的数字签名算法。在公钥加密算法的候选算法中，基于编码的后量子密码算法BIKE、Classic McEliece和HQC以及基于同源的SIKE算法进入第4轮评估，但很快就证实了SIKE算法可被完全破解，因而被宣告退出。

CRYSTALS-Dilithium算法是NIST现有数字签名候选方案中最具潜力的算法之一，其算法结构整体上基于Lyubashevsky所提出的“Fiat-Shamir with Aborts”方案[15]，算法内部通过使用拒绝采样、提取高低阶位等方式，在保障算法安全性的前提下尽可能地减少密钥及签名长度。目前，基于CRYSTALS-Dilithium数字签名方案的研究有不少集中于硬件实现和侧信道攻击等领域。Ricci等[16]利用VHDL语言在现场可编程门阵列（FPGA, field programmable gate array）上实现了CRYSTALS-Dilithium数字签名算法。Land等[17]利用数字信号处理器（DSP, digital signal processing）构造了一种优化型的数论变换（NTT, number theoretic transform）组件，以实现了多项式在硬件上的低时延运算。Beckwith等[18]使用Verilog语言在寄存器传输级（RTL, register transfer level）上设计了一种基于CRYSTALS-Dilithium算法的高效FPGA实现。Karabulut等[19]提出了第一个针对小多项式采样的侧信道攻击，这种攻击方式能以99.99%的概率成功提取CRYSTALS-Dilithium算法中的目标系数。Le等[20]基于CRYSTALS-Dilithium数字签名方案的初始版本，设计了一种MBS（blind signature from module latices）方案，但该方案设计过程复杂、签名结构烦琐，不易于实现。

当前，常规数字签名方案无法满足复杂特殊应用需求，基于传统数学难题的特殊数字签名方案难以有效抵御量子攻击，而现有基于格的盲签名方案普遍存在公私钥开销大或签名尺寸大的问题，同时在密钥生成和签名生成过程中效率低，导致其难以在实际场景中应用。后量子数字签名方案CRYSTALS-Dilithium具有开销小、运算快等优点，本文在继承该方案优势的基础上引入了盲签名的概念，并在其算法框架上构建了一个简单实用的盲签名方案CDBS（CRYSTALS-Dilithium blind signature）。

本文主要的研究贡献如下。

1) 构造了一种仅需3轮交互的高效抗量子盲签名方案CDBS。该方案基于CRYSTALS-Dilithium算法框架，采用Fiat-Shamir结构，结合拒绝采样、

提取高低阶位和NTT等技术，确保其安全性和高效性。理论分析表明，CDBS方案具备正确性、盲性和不可伪造性等盲签名的关键属性。

2) 在软件层面测试了CDBS方案的性能和运行效率。该方案在结构上简单实用，签名参数精简高效，盲化和解盲过程操作便捷且实现迅速，仅需3轮交互即可完成消息的盲化和去盲。经测试，若不计交互时延，CDBS方案完成一轮签名和验证需耗时657.65 μs。

3) CDBS方案具有较好的综合性能。相较于MBS方案，CDBS方案的整体开销减少了24.5%，具备开销小、效率高和安全性强等优点。

1 预备知识

1.1 符号说明

在本文中，令R和R_q分别表示环$\frac{Z[X]}{X^n+1}$和$\frac{Z_q[X]}{X^n+1}$，其中，q是一个奇素数。在CRYSTALS-Dilithium数字签名方案中，模q和n的值是固定的，其中，$n=256$，$q=2^{23}-2^{13}+1$。对任意多项式环$\frac{Z_q[X]}{X^n+1}$，本文用小写字母表示环上的元素（如a），用小写粗体表示向量（如$\boldsymbol{b}$），且在默认情况下，所有向量均为列向量，用大写粗体（如$\boldsymbol{A}$）表示矩阵，用$(\boldsymbol{A})^{\mathrm{T}}$来表示$\boldsymbol{A}$的转置。同时引入以下2个概念。

1) 模块化缩减[21]。对于任意一个正整数α，若α是偶数，则定义$r'=r \bmod^{\pm}\alpha$是在区间$r'\in\left[-\frac{\alpha}{2},\frac{\alpha}{2}\right]$上使$r'\equiv r \bmod \alpha$成立的唯一元素；若$\alpha$是奇数，则区间范围可表示为$r'\in\left[-\frac{\alpha-1}{2},\frac{\alpha-1}{2}\right]$。

2) 无穷范数和欧氏范数[22]。对任意元素$w\in\mathbb{Z}_q$，若以$\|w\|_\infty$表示$\left|w \bmod^{\pm} q\right|$的值，则对于环上系数$w=w_0+w_1X+\cdots+w_{n-1}X^{n-1}\in R$，可分别对其无穷范数和欧氏范数作如下定义。

$$\|w\|_\infty=\max_i\left|w_i\right| \tag{1}$$

$$\|w\|=\sqrt{\left|w_0\right|^2+\left|w_1\right|^2+\cdots+\left|w_{n-1}\right|^2} \tag{2}$$

类似地，对于任意向量$\boldsymbol{w}=(w_1,w_2,\cdots,w_k)\in R^k$，其无穷范数和欧氏范数可分别表示为

$$\|\boldsymbol{w}\|_\infty=\max_i\left|w_i\right| \tag{3}$$

$$\|\boldsymbol{w}\|=\sqrt{\left|w_1\right|^2+\cdots+\left|w_k\right|^2} \tag{4}$$

本文用S_η表示任意$w\in R$且满足$\|w\|_\infty\leqslant\eta$的元素，同时用$\tilde{S}_\eta$表示集合$\{w \bmod^{\pm}2\eta : w\in R\}$。

1.2 格理论基础与困难性假设

定义1 格（Lattice）[23]。令$\mathcal{L}$是m维欧氏空间$\mathbb{R}^m$中n（$m>n$）个线性无关向量组$\boldsymbol{a}_1,\boldsymbol{a}_2,\cdots,\boldsymbol{a}_n\in\mathbb{R}^n$中所有系数构成的线性组合，即

$$\mathcal{L}(\boldsymbol{A})=\left\{\sum_{i=1}^{n}x_i\boldsymbol{a}_i\,|\,x_i\in\mathbb{Z},i\in(1,\cdots,n)\right\} \tag{5}$$

其中，矩阵$\boldsymbol{A}=[\boldsymbol{a}_1,\boldsymbol{a}_2,\cdots,\boldsymbol{a}_n]\in\mathbb{R}^{n\times n}$为格$\mathcal{L}$的一组基，这种形式也被称为标准格。

定义2 q元格[23]。对于任意正整数q、m、n和矩阵$\boldsymbol{A}\in\mathbb{Z}^{n\times m}$，$q$元格可定义为

$$\Lambda_q^{\perp}(\boldsymbol{A})=\{z\in\mathbb{Z}^m:\boldsymbol{A}z=0(\bmod q)\} \tag{6}$$

对于任意向量$\boldsymbol{e}\in\mathbb{Z}_q^n$，$q$元格的陪集定义为

$$\Lambda_q^{e}(\boldsymbol{A})=\{z\in\mathbb{Z}^m:\boldsymbol{A}z=\boldsymbol{e}(\bmod q)\} \tag{7}$$

其中，n为格Λ的秩，m为格Λ的维数。$\Lambda_q^{e}(\boldsymbol{A})$可看作$\Lambda_q^{\perp}(\boldsymbol{A})$的平移，也被称作$\Lambda_q^{\perp}(\boldsymbol{A})$的陪集，即$\Lambda_q^{e}(\boldsymbol{A})=\Lambda_q^{\perp}(\boldsymbol{A})+d$，$d\in\Lambda_q^{e}(\boldsymbol{A})$。

定义3 模误差学习（MLWE, module learning with error）问题[24]。首先预置一个安全参数λ，其大小为2的n次幂（n为向量的维数）。令模$q=q(\lambda)$且$q>2$，假设存在一个不可约多项式$f(x)=x^n+1$和模q下的多项式环$R_q=\frac{\mathbb{Z}_q}{f(x)}$。若用$U$表示均匀分布，$S$表示$R_q$上的一个概率分布，$s\leftarrow S$则表示一组由这种概率分布所获取的采样数据。一组MLWE样本数据$\mathcal{L}_{q,k,l,S}$可表示为$\boldsymbol{s}_1\leftarrow S^k$、$\boldsymbol{s}_2\leftarrow S^l$、$\boldsymbol{A}\leftarrow U(R_q^{k\times l})$和$\boldsymbol{A}\leftarrow U(R_q^{k\times l})$。MLWE问题可分为搜索型s-MLWE和判定型d-MLWE两类，s-MLWE要求在给定参数$(\boldsymbol{A},t=\boldsymbol{A}\boldsymbol{s}_1+\boldsymbol{s}_2)\leftarrow L_{q,k,l,S}$的条件下恢复向量$\boldsymbol{s}_1$和$\boldsymbol{s}_2$；d-MLWE则被用于区分概率分布$(\boldsymbol{A},\boldsymbol{t}=\boldsymbol{A}\boldsymbol{s}_1+\boldsymbol{s}_2)\leftarrow L_{q,k,l,S}$和$(\boldsymbol{A},\boldsymbol{t})\leftarrow U(R_q^{k\times l}\times R_q^k)$。

定义4 小整数解（SIS, small integer solution）问题[25]。给定参数集合$\{n,m,q,\beta\}$和均匀随机矩阵$\boldsymbol{A}\in{\mathbb{Z}_q}^{m\times n}$，所谓小整数解问题就是寻找非零整数向量$\boldsymbol{y}\in{\mathbb{Z}_q}^m$使其满足$\|y\|\leqslant\beta$且$\boldsymbol{A}\boldsymbol{y}=0(\bmod q)$。实际上，SIS问题可看作一个$q$阶$m$维格上的最短向量问题（SVP, shortest vector problem），其中$\Lambda_q^{\perp}(\boldsymbol{A})=\{z\in\mathbb{Z}^m:\boldsymbol{A}z=0(\bmod q)\}$，则此时SIS问

题就成为一个在格 $\Lambda_q^{\perp}(\boldsymbol{A})$ 上寻找最短向量的问题。

2　盲签名模型及 CRYSTALS-Dilithium 内部函数

2.1　盲签名

2.1.1　系统模型

盲签名主要包括密钥生成、盲化、签名、去盲和验证5个阶段，且由用户、签名方和验证者互传参数完成整体过程，如图1所示，具体步骤如下。

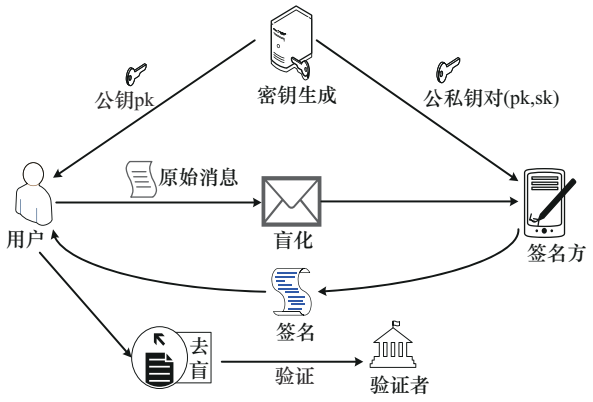

图1　盲签名具体流程

1) 密钥生成。系统初始化并产生相应的安全参数和系统参数，以系统参数作为输入，由密钥生成算法生成对应的公钥pk和私钥sk。

2) 盲化。由用户U生成盲化因子，对原始消息进行盲化处理并获得盲化后的消息，从而隐藏初始消息的具体内容。

3) 签名。签名方S使用私钥sk对盲化后的消息进行签名，获得盲化后的签名结果，随后签名方S将生成的盲签名发送给用户U进行下一阶段的运算。

4) 去盲。由用户U对已生成的盲签名结果进行去盲处理，获得正常的数字签名结果。

5) 验证。验证者利用已公开的系统参数、验证方法和签名者的公钥pk对确定消息的数字签名进行验证，若验证通过，则输出1，否则输出0。

2.1.2　安全性模型

一个安全且完备的盲签名方案至少要具备2个性质：盲性和不可伪造性，可对其安全性和安全模型做如下定义。

定义5　盲性[26]。盲性是指签名方对盲化后的消息进行数字签名，但无法知道或区别原始消息的具体内容。盲性是盲签名的重要性质，可以保护消息方的隐私性和匿名性。

假设存在攻击者E和挑战者C，则可对盲签名盲性的安全模型做如下定义。

1) 系统建立阶段。挑战者C运行密钥生成算法，得到系统公共参数和对应密钥，并将生成的参数和公钥发送给攻击者E。

2) 预备阶段。由攻击者E随机选取2个等长且不同的明文消息 M_0 和 M_1，并将其发送给挑战者C。

3) 挑战阶段。由挑战者C随机选取整数 $k \in \{0,1\}$，并随机选择一个原始消息 M_k（k=0或1）进行盲化，同时输出盲化后的消息 c_b^*（b=0或1）。

4) 猜测阶段。由攻击者E猜测 b 的值，若 $b = k$ 则攻击者E获得胜利，即盲签名方案盲性不存在。

定义6　不可伪造性[27]。不可伪造性是指在盲签名方案中，只有签名方才能生成有效的签名，而其他任何机构都不能伪造签名，这种安全性保证了盲签名的真实性和完整性。

在随机预言机模型中，假设存在攻击者E和挑战者C。通过一系列模拟博弈游戏后，在挑战者C与攻击者E的模拟对抗中，若攻击者E在下述匿名性证明游戏中的获胜优势是可以忽略的，则可认为该盲签名方案满足不可伪造性。定义盲签名不可伪造性的安全性模型如下。

1) 初始化阶段。挑战者C进行系统初始化，生成公私钥对并将公钥发送给攻击者E。

2) 查询阶段。攻击者E可以向挑战者C发送签名、验证和哈希等查询，并接收挑战者C的响应。此过程可重复多次，直至攻击者选择结束。

3) 挑战阶段。攻击者E提出挑战，即伪造签名或提供否认的证据，试图欺骗挑战者C。

4) 判定阶段。挑战者C根据攻击者E的输出和自身的私钥，判断攻击者是否成功伪造了有效且可验证的签名。最终，挑战者输出一个比特（1或0），表示挑战的成功或失败。

2.2　CRYSTALS-Dilithium基础算法模块

2.2.1　多项式生成算法

矩阵扩展算法ExpandA使用可扩展输出长度的哈希函数（SHAKE-256），将随机数种子 $\rho \in \{0,1\}^{256}$ 映射到矩阵 $\boldsymbol{A} \in R_q^{k \times l}$ 中，但由于矩阵 $\boldsymbol{A}$ 的运算仅包含乘法操作，因此矩阵扩展算法ExpandA不以

$\boldsymbol{A} \in R_q^{k \times l} = \left[\frac{\mathbb{Z}_q[X]}{X^{256}+1}\right]^{k \times l}$ 的形式直接输出，而是以 $\boldsymbol{A}$ 在 NTT 域中的表示形式 $\hat{\boldsymbol{A}} \in \mathbb{Z}_q^{256}$ 作为输出的结果。在密钥生成阶段，Dilithium 使用哈希函数 SHAKE-256 生成密钥向量 $(\boldsymbol{s}_1,\boldsymbol{s}_2) \in S_\eta^l \times S_\eta^k$。在签名生成阶段，函数 ExpandMask 通过随机数种子 ρ' 和随机数 k 的映射生成多项式向量 $\boldsymbol{y} \in S_{\gamma 1}^l$，其具体过程如下。首先，分别独立计算向量 $\boldsymbol{y}$ 中的 l 个系数，且这些系数均为 $S_{\gamma 1}^l$ 中的多项式。然后，对于第 i（$0 \leqslant i < l$）个系数，将 ρ' 的 48 B 和用二进制表示的 $k+i$ 按比特顺序连接并输入 SHAKE-256 中，最终所获得的输出即函数 ExpandMask 的输出结果。同时，算法 SampleInBall 用于在签名生成和验证过程中构造多项式，该多项式中有 τ 个系数的值为+1或−1，其余系数均置为 0，具体过程如算法 1 所示。

算法 1 SampleInBall(ρ)

输入 ρ

输出 $\boldsymbol{c}$

1) 初始化 $a \to (a(r),a(r^3),\cdots,a(r^{511})):R_q \to \prod_i \frac{\mathbb{Z}_q[X]}{X-r^i}$

2) for $a \to (a(r),a(r^3),\cdots,a(r^{511})):R_q \to \prod_i \frac{\mathbb{Z}_q[X]}{X-r^i}$ to 255

3) $a \to (a(r),a(r^3),\cdots,a(r^{511})):R_q \to \prod_i \frac{\mathbb{Z}_q[X]}{X-r^i}$

4) $s \leftarrow \{0,1\}$

5) $c_i = c_j$

6) $c_j = (-1)^s$

2.2.2 多项式在 NTT 域中的运算

在多项式计算中，由于所选择的模数 q 存在 512 次原根 r，可令 r=1 753，则分圆多项式 $X^{256}+1$ 在模 q 下可以分解为线性因式 $X-r^i$ 的形式，其中 $a \to \left(a(r),a(r^3),\cdots,a(r^{511})):R_q \to \prod_i \frac{\mathbb{Z}_q[X]}{X-r^i}\right)$。由中国剩余定理可知，分圆环 R_q 同构于环的乘积，即满足 $\frac{\mathbb{Z}_q[X]}{X-r^i} \cong \mathbb{Z}_q$，且由于环上的乘法是逐点的，因此环上元素间的乘法更加易于实现，即可通过傅里叶变换计算。

$$a \to (a(r),a(r^3),\cdots,a(r^{511})):R_q \to \prod_i \frac{\mathbb{Z}_q[X]}{X-r^i} \tag{8}$$

此时，由于在运算中满足等式 $X^{256}+1 = X^{256}-r^{256} = (X^{128}-r^{128})(X^{128}+r^{128})$，可计算如下映射。

$$\frac{\mathbb{Z}_q[X]}{X^{256}+1} \to \frac{\mathbb{Z}_q[X]}{X^{128}-r^{128}} \frac{\mathbb{Z}_q[X]}{X^{128}+r^{128}} \tag{9}$$

随后，可继续按照上述方法对多项式进行缩减，在有限域中，这种快速傅里叶变换也被称为数论变换。定义多项式 $a \in R_q$，其在 NTT 域中的表示形式为 $\hat{a} = \mathrm{NTT}(a) \in \mathbb{Z}_q^{256}$，则其具体形式可表示为

$$\hat{a} = \mathrm{NTT}(a) = (a(r_0),a(-r_0),\cdots,a(r_{127}),a(-r_{127})) \tag{10}$$

2.2.3 抗碰撞哈希函数 CRH

CRYSTALS-Dilithium 使用 SHA-3 算法的可扩展输出函数（XOF, extendable output function）SHAKE-256 作为哈希函数来确保消息的完整性和不可篡改性。在签名生成阶段，消息被输入 SHAKE-256 中，然后将其哈希值与其他计算结果一起用于生成签名或直接用作伪随机数生成器。在验证签名阶段，同样会将消息输入 SHAKE-256 中，用于验证消息的完整性。

2.2.4 比特分解算法 $\mathrm{Power2Round}_q$

函数 $\mathrm{Power2Round}_q$ 通过输入参数 r、d，将元素 r 分解为 $r = r_1 2^d + r_0$ 的形式，并输出 r_0 和 r_1，其中 $r_0 = r \bmod^{\pm} 2^d$，$r_1 = \frac{r-r_0}{2^d}$，具体过程如算法 2 所示。

算法 2 $\mathrm{Power2Round}_q(r,d)$

输入 r，d

输出 $\frac{r-r_0}{2^d}$，r_0

1) $r = r_1 2^d + r_0$

2) $r_0 = r \bmod^{\pm} 2^d$

2.2.5 比特分解算法 $\mathrm{Decompose}_q$ 及其相关算法

首先，定义整数 α 且其为 $q-1$ 的倍数。此时，$\mathrm{Decompose}_q$ 的定义类似于 $\mathrm{Power2Round}_q$，但不同之处在于在内部算法的具体实现中，$\mathrm{Decompose}_q$ 函数使用 α 替换了函数 $\mathrm{Power2Round}_q$ 中的 2^d。同时，定义函数 $\mathrm{HighBits}_q$ 和 $\mathrm{LowBits}_q$ 分别用于提取算法 $\mathrm{Decompose}_q$ 的输出结果 r_1 和 r_0。函数 $\mathrm{MakeHint}_q$ 则使用 $\mathrm{HighBits}_q$ 生成一个提示 $\boldsymbol{h}$，函数 $\mathrm{UseHint}_q$ 则使用生成的提示 $\boldsymbol{h}$ 恢复高阶位。其中，比特分解算法 $\mathrm{Decompose}_q$ 的具体数学定义如算法 3 所示。

算法3　$\mathrm{Decompose}_q(r,\alpha)$

输入　r，α

输出　r_1，r_0

1) $r = r \bmod^{+} q$

2) $r_0 = r \bmod^{\pm} \alpha$

3) if $r - r_0 = q - 1$

4) then $r_1 = 0$，$r_0 = r_0 - 1$

5) else $r_1 = \dfrac{r - r_0}{\alpha}$

此外，定义2个函数$\mathrm{HighBits}_q$和$\mathrm{LowBits}_q$分别用于提取高低位比特分解算法$\mathrm{Decompose}_q$的2个输出结果，其数学表达式分别如算法4和算法5所示。

算法4　$\mathrm{HighBits}_q(r,\alpha)$

输入　$r,\alpha(r_1,r_0) = \mathrm{Decompose}_q(r,\alpha)$

输出　r_1

算法5　$\mathrm{LowBits}_q(r,\alpha)$

输入　$r,\alpha(r_1,r_0) = \mathrm{Decompose}_q(r,\alpha)$

输出　r_0

提示$\boldsymbol{h}$的生成算法$\mathrm{MakeHint}_q$的数学定义如算法6所示。

算法6　$\mathrm{MakeHint}_q(z,r,\alpha)$

输入　z，r，α

输出　1或0

1) $r_1 = \mathrm{HighBits}_q(r,\alpha)$

2) $\boldsymbol{v}_1 = \mathrm{HighBits}_q(r + z,\alpha)$

3) if $r_1 \neq v_1$ return 1

4) else return 0

此外，在签名认证过程中使用的用于恢复高阶位的算法$\mathrm{UseHint}_q$的数学定义如算法7所示。

算法7　$\mathrm{UseHint}_q(h,r,\alpha)$

输入　h，r，α

输出　$(r_1 \pm 1) \bmod m$

1) $m = \dfrac{q - 1}{\alpha}$

2) $(r_1,r_0) = \mathrm{Decompose}_q(r,\alpha)$

3) if $h = 1 \&\& r_0 > 1$

4) return $(r_1 + 1) \bmod^{+} m$

5) if $h = 1 \&\& r_0 \leqslant 1$

6) return $(r_1 - 1) \bmod^{+} m$

3　CDBS方案构造

3.1　盲签名方案设计

基于CRYSTALS-Dilithium数字签名方案的算法框架[28]，本文设计了一种具备抗量子特性的盲签名方案。该方案包括密钥生成、盲化、签名、去盲和验证5个阶段，并经过用户U与签名方S之间的3轮交互，完成一次盲化、签名和去盲过程。该方案首先需选定一个安全参数ζ，同时为控制密钥长度，签名、盲化和验证3个阶段均以随机数种子ρ生成矩阵$\boldsymbol{A}$作为算法起点位置。此外，由于每个消息可能需要迭代多次才能完成签名，本文使用抗碰撞哈希（CRH, collision resistant hash）函数计算消息的初始摘要，并在整个签名过程中使用该初始摘要代替明文消息，其详细的交互流程如图2所示。

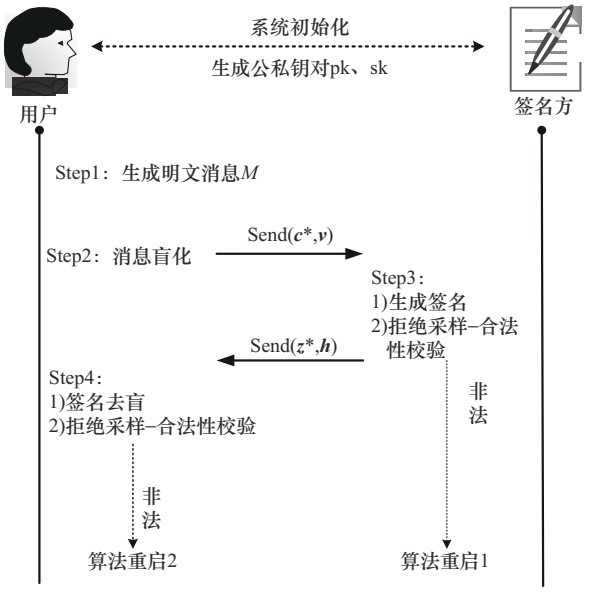

图2　盲签名交互流程

同时，本文所设计的CDBS方案完成一次完整盲签名生成及验证过程需历经密钥生成、盲化、签名、去盲和验证5个阶段，所设计方案的完整流程如算法8~算法11所示。

3.1.1　密钥生成

CDBS方案的密钥生成算法模块如算法8所示。

算法8　密钥生成

输出　$\mathrm{pk} = (\rho,\mathrm{tr},\boldsymbol{t}_1)$，$\mathrm{sk} = (K,\boldsymbol{s}_1,\boldsymbol{s}_2,\boldsymbol{t}_0)$

1) $\zeta \leftarrow \{0,1\}^{256}$

2) $(\rho,\tau,K) \in \{0,1\}^{256 \times 3} = H(\zeta)$

3) $(\boldsymbol{s}_1,\boldsymbol{s}_2) \in S_\eta^l \times S_\eta^k = H(\tau)$

4) $\boldsymbol{A} \in R_q^{k \times l} = \mathrm{ExpandA}(\rho)$

5) $\boldsymbol{t} = \boldsymbol{A}\boldsymbol{s}_1 + \boldsymbol{s}_2$

6) $(\boldsymbol{t}_1,\boldsymbol{t}_0) = \text{Power2Round}_q(\boldsymbol{t},d)$

7) $\text{tr} \in \{0,1\}^{384} = \text{CRH}(\rho\|\boldsymbol{t}_1)$

在密钥生成阶段，首先，进行系统初始化并随机生成一个256位的安全参数ζ，以ζ作为输入通过SHAKE-256算法生成随机参数集(ρ,τ,K)，以参数ρ为随机数种子生成矩阵$\boldsymbol{A} \in R_q^{k\times l}$并输出结果（矩阵$\boldsymbol{A}$在NTT域中可表示为$\hat{\boldsymbol{A}} \in R_q^{k\times l}$）。然后，采用均匀采样生成密钥向量$(\boldsymbol{s}_1,\boldsymbol{s}_2) \in S_\eta^l \times S_\eta^k$，并计算生成私钥的组成参数$\boldsymbol{t} = \boldsymbol{A}\boldsymbol{s}_1 + \boldsymbol{s}_2$，其中$\boldsymbol{A}\boldsymbol{s}_1$在NTT域中的计算可记作$\text{NTT}^{-1}(\hat{\boldsymbol{A}}\,\text{NTT}(\boldsymbol{s}_1))$。最后，运行比特分解算法$\text{Power2Round}_q$提取向量$\boldsymbol{t}$的高位比特$\boldsymbol{t}_1$，并输出整体方案的公钥$\text{pk} = (\rho,\text{tr},\boldsymbol{t}_1)$及私钥$\text{sk} = (K,\boldsymbol{s}_1,\boldsymbol{s}_2,\boldsymbol{t}_0)$。

3.1.2 盲化

在盲化前，由签名者生成向量$\boldsymbol{y} \in S_{\gamma 1}^l$，同时计算生成参数$\boldsymbol{w} = \boldsymbol{A}\boldsymbol{y}$并将其发送给用户，参与盲化过程。在盲化阶段，首先，由用户通过公钥中的随机数种子ρ生成初始矩阵$\boldsymbol{A}$，其在NTT域中表示形式为$\hat{\boldsymbol{A}} \in \mathbb{Z}_q^{256}$，随后用户将初始消息$M$和参数tr拼接后输入CRH函数中，构建初始摘要$\mu = \text{CRH}(\text{tr}||M)$，接着利用初始摘要构建参数$\rho'$，并将其与重启参数$i$输入函数ExpandMask中生成掩蔽向量$\boldsymbol{x} \in S_{\gamma 1}^l$。然后，生成盲化因子$p$并计算构造多项式$\boldsymbol{v} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w} + \boldsymbol{t}_1 p2^d$以参与到整体盲化进程中，此时多项式$\boldsymbol{A}\boldsymbol{x}$的运算也需在NTT域中并记作$\text{NTT}^{-1}(\hat{\boldsymbol{A}}\text{NTT}(\boldsymbol{x}))$，随后通过算法$\text{HighBits}_q$提取向量$\boldsymbol{v}$的高位比特$\boldsymbol{v}_1$并将其和初始摘要连接起来，输入哈希函数中得到待盲化的信息参数c。最后，引入盲化因子p进行计算得到盲化信息c^*，待盲化结束后，由用户传递盲化信息$(c^*,\boldsymbol{v})$至签名方，其具体过程如算法9所示。

算法9 盲化算法

输入 $(M,\boldsymbol{w})$，$\text{pk} = (\rho,\text{tr},\boldsymbol{t}_1)$

输出 $(c^*,\boldsymbol{v})$

8) $\boldsymbol{A} \in R_q^{k\times l} = \text{ExpandA}(\rho)$

9) $\mu \in \{0,1\}^{384} = \text{CRH}(M\|\text{tr})$

10) $i = 0,\rho' \in \{0,1\}^{384} = \text{CRH}(K\|\mu)$

11) $\boldsymbol{x} \in S_{\gamma_1}^l = \text{ExpandMask}(\rho',i)$

12) $p \in B_\tau = \text{SampleInBall}(H(\text{tr}\|M))$

13) $\boldsymbol{v} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w} + \boldsymbol{t}_1 p2^d$

14) $\boldsymbol{v}_1 = \text{HighBits}_q(\boldsymbol{v},2\gamma_2)$

15) $\tilde{c} \in \{0,1\}^{256} = H(\boldsymbol{v}_1\|\mu)$

16) $c \in B_\tau := \text{SampleInBall}(\tilde{c})$

17) $c^* = c + p$

3.1.3 签名

在签名阶段，签名方得到用户所传递的盲化信息$(c^*,\boldsymbol{v})$后，使用私钥$\text{sk} = (K,\boldsymbol{s}_1,\boldsymbol{s}_2,\boldsymbol{t}_0)$对盲化信息进行签名并判断签名的合法性。具体地，签名方首先提取向量$\boldsymbol{v}$的高位比特并计算得到$\boldsymbol{z} = \boldsymbol{y} + c^*\boldsymbol{s}_2$的值，其中多项式$c^*\boldsymbol{s}_2$的运算需在NTT域中进行，即$c^*\boldsymbol{s}_2 \leftarrow \text{NTT}^{-1}(\hat{c}^*\hat{\boldsymbol{s}}_2)$，随后运行比特算法$\text{Decompose}_q$分解向量$\boldsymbol{v} - c^*\boldsymbol{s}_2$的高低位比特，最后根据拒绝采样定理校验签名是否合法。由于所生成签名满足不等式$\|\boldsymbol{z}\|_\infty < \gamma_1 - \beta$，此时签名的分布就会近似服从指定区间上的均匀分布，以防止密钥泄露，进一步保证了方案安全性。同时，若$\|\boldsymbol{r}_0\| \geqslant \gamma_2 - \beta$或$\|c^*\boldsymbol{t}_2\|_\infty \geqslant \gamma_2$，则签名中断，并由用户在步骤11)处重启并重新构造盲化和签名过程。若签名有效，则生成提示$\boldsymbol{h}$用于后续的验证过程并将参数$(\boldsymbol{z},\boldsymbol{h})$传递给用户，具体过程如算法10所示。

算法10 签名算法

输入 $(c^*,\boldsymbol{v})$，$\text{sk} = (K,\boldsymbol{s}_1,\boldsymbol{s}_2,\boldsymbol{t}_0)$

输出 $(\boldsymbol{z},\boldsymbol{h})$

18) $\boldsymbol{v}_1 = \text{HighBits}_q(\boldsymbol{v},2\gamma_2)$

19) $\boldsymbol{z} = \boldsymbol{y} + c^*\boldsymbol{s}_1$

$\Rightarrow c^*\boldsymbol{s}_1 \leftarrow \text{NTT}^{-1}(\hat{c}^*,\hat{\boldsymbol{s}}_1)$

20) $(\boldsymbol{r}_1,\boldsymbol{r}_0) = \text{Decompose}_q(\boldsymbol{v} - c^*\boldsymbol{s}_2,2\gamma_2)$

$\Rightarrow c^*\boldsymbol{s}_2 \leftarrow \text{NTT}^{-1}(\hat{c}^*,\hat{\boldsymbol{s}}_2)$

21) if $\|\boldsymbol{z}\|_\infty \geqslant \gamma_1 - \beta$ or $\|\boldsymbol{r}_0\| \geqslant \gamma_2 - \beta$ or $\|c^*\boldsymbol{t}_0\|_\infty \geqslant \gamma_2$ $i = i + l$ then goto 11)

22) else $\boldsymbol{h} = \text{MakeHint}_q(-c^*\boldsymbol{t}_0,\boldsymbol{v} - c^*\boldsymbol{s}_2 + c^*\boldsymbol{t}_0,2\gamma_2)$

3.1.4 去盲

用户获得盲化后的签名参数$\boldsymbol{z}$后，首先使用去盲因子$\boldsymbol{x}$对$\boldsymbol{z}$进行去盲操作并获得参数$\boldsymbol{z}^*$，随后判断该参数是否满足安全性条件$\|\boldsymbol{z}^*\|_\infty < \gamma_1 - \beta$，若满足，则方案输出一个合法且有效的签名$\Sigma = (\boldsymbol{z}^*,\boldsymbol{h},\tilde{c})$；否则，签名过程中断，即所生成的盲签名不是一个有效的签名，方案返回至初始阶段并重启，其具体过程如算法11所示。

算法11　去盲算法

输入　$(z,\boldsymbol{h})$

输出　$\Sigma=(z^*,\boldsymbol{h},\tilde{c})$

23) $z^*=z+\boldsymbol{x}$

24) if $\|z^*\|_\infty<\gamma_1-\beta$

output $\Sigma=(z^*,\boldsymbol{h},\tilde{c})$

25) else restart

3.1.5　验证

在验证阶段，验证者利用公钥$\mathrm{pk}=(\rho,\mathrm{tr},\boldsymbol{t}_1)$和明文$M$对签名$\Sigma=(z^*,\boldsymbol{h},\tilde{c})$进行验证，具体地，验证者首先通过公钥和明文计算初始摘要值即$\mu=\mathrm{CRH}(\mathrm{CRH}(\rho\|\boldsymbol{t}_1)\|M)$，随后使用签名、公钥和算法$\mathrm{UseHint}_q$恢复向量$\boldsymbol{v}$的高位比特$\boldsymbol{v}_1$，并使用与生成$\tilde{c}$相同的方式计算出其估计值$\tilde{c}'$，若二者相等且满足条件$\|z\|_\infty<\gamma_1-\beta$，则验证通过，返回1；否则，返回0表示验证失败。需要注意的是，本阶段计算$\boldsymbol{A}z^*-c\boldsymbol{t}_1 2^d$时，仍需在NTT域中进行，即需计算$\mathrm{NTT}^{-1}(\hat{\boldsymbol{A}}\,\mathrm{NTT}(z^*)-\mathrm{NTT}(c)\,\mathrm{NTT}(\boldsymbol{t}_1 2^d))$的值。

算法12　验证算法

输入　$\mathrm{pk}=(\rho,\mathrm{tr},\boldsymbol{t}_1)$，$M$，$\Sigma=(z^*,\boldsymbol{h},\tilde{c})$

输出　0或1

26) $\boldsymbol{A}\in R_q^{k\times l}=\mathrm{ExpandA}(\rho)$

27) $\mu\in\{0,1\}^{384}=\mathrm{CRH}(\mathrm{CRH}(\rho\|\boldsymbol{t}_1)\|M)$

28) $\boldsymbol{v}_1'=\mathrm{UseHint}_q(\boldsymbol{h},\boldsymbol{A}z^*-c\boldsymbol{t}_1 2^d,2\gamma_2)$

29) if $\|z^*\|_\infty<\gamma_1-\beta$ or $\tilde{c}=\tilde{c}'=H(\boldsymbol{v}'\|\mu)$ return 1

30) else return 0

3.2　方案正确性证明

3.2.1　相关引理及推论

利用下述的引理及推论来分析数字签名方案CRYSTALS-Dilithium内部算法的相关属性，进而证明本文方案的正确性和安全性。

引理1　设q和α满足$q>2\alpha$，其中$q\equiv 1(\mathrm{mod}\,\alpha)$且$\alpha$是偶数，假设$\boldsymbol{r}$和$\boldsymbol{z}$是元素在$R_q$中的向量，其中$\|z\|_\infty\leqslant\dfrac{\alpha}{2}$，设$\boldsymbol{h}$和$\boldsymbol{h}'$均为向量，则方案内部算法$\mathrm{HighBits}_q$、$\mathrm{UseHint}_q$和$\mathrm{MakeHint}_q$满足以下特性。

1) $\mathrm{UseHint}_q(\mathrm{MakeHint}(z,\boldsymbol{r},\alpha),\boldsymbol{r},\alpha)=\mathrm{HighBites}(\boldsymbol{r}+z,\alpha)$。

2) 若$\boldsymbol{v}_1=\mathrm{UseHint}(\boldsymbol{h},\boldsymbol{r},\alpha)$成立，则存在不等式$\left\|\boldsymbol{r}-\boldsymbol{v}_1\alpha\right\|\leqslant\alpha+1$。若向量$\boldsymbol{h}$中1的数量为$w$，则在进行模$q$的中心约减后，$\boldsymbol{r}-\boldsymbol{v}_1\alpha$的系数除$w$个外，其余系数的幅值均不会超过$\dfrac{\alpha}{2}$。

3) 对于任意向量$\boldsymbol{h}$和$\boldsymbol{h}'$而言，若满足等式$\mathrm{UseHint}(\boldsymbol{h},\boldsymbol{r},\alpha)=\mathrm{UseHint}(\boldsymbol{h}',\boldsymbol{r},\alpha)$，则有$\boldsymbol{h}=\boldsymbol{h}'$。

引理2　若存在不等式$\|\boldsymbol{s}\|_\infty\leqslant\beta$，且满足条件$\left\|\mathrm{LowBits}_q(\boldsymbol{r},\alpha)\right\|_\infty<\dfrac{\alpha}{2}-\beta$，则有如下等式成立。

$$\mathrm{HighBites}(\boldsymbol{r},\alpha)=\mathrm{HighBites}(\boldsymbol{r}+\boldsymbol{s},\alpha)\tag{11}$$

3.2.2　正确性证明

若要证明本文所设计的CDBS方案满足正确性，实际上需证明在$\|z^*\|_\infty<\gamma_1-\beta$成立的条件下，方案满足等式$\tilde{c}=\tilde{c}'=H(\boldsymbol{v}_1'\|\mu)$，该等式成立的条件实际上是满足了等式$\boldsymbol{v}_1'=\mathrm{UseHint}_q(\boldsymbol{h},\boldsymbol{A}z^*-c\boldsymbol{t}_1 2^d,2\gamma_2)=\boldsymbol{v}_1$，已知存在$z^*=z+\boldsymbol{x}$，则

$$\boldsymbol{A}z^*-c\boldsymbol{t}_1 2^d=\boldsymbol{A}(z+\boldsymbol{x})-c\boldsymbol{t}_1 2^d=\boldsymbol{A}z+\boldsymbol{A}\boldsymbol{x}-c\boldsymbol{t}_1 2^d\tag{12}$$

且因为$z=\boldsymbol{y}+c^*\boldsymbol{s}_1$，$c^*=c+p$，则式(12)可表示为

$$\boldsymbol{A}\boldsymbol{x}+\boldsymbol{A}\boldsymbol{y}+\boldsymbol{A}c^*\boldsymbol{s}_1-c\,\boldsymbol{t}_1 2^d=\boldsymbol{A}\boldsymbol{x}+\boldsymbol{A}\boldsymbol{y}+\boldsymbol{A}c^*\boldsymbol{s}_1-(c^*-p)\boldsymbol{t}_1 2^d=\boldsymbol{A}\boldsymbol{x}+\boldsymbol{A}\boldsymbol{y}+\boldsymbol{A}c^*\boldsymbol{s}_1+p\boldsymbol{t}_1 2^d-c^*\boldsymbol{t}_1 2^d\tag{13}$$

已知$\boldsymbol{t}=\boldsymbol{t}_1 2^d+\boldsymbol{t}_0$，代入式(13)可得

$$\boldsymbol{A}(\boldsymbol{x}+\boldsymbol{y})+\boldsymbol{A}c^*\boldsymbol{s}_1+p\boldsymbol{t}_1 2^d-c^*(\boldsymbol{t}-\boldsymbol{t}_0)=\boldsymbol{A}(\boldsymbol{x}+\boldsymbol{y})+\boldsymbol{A}c^*\boldsymbol{s}_1+p\boldsymbol{t}_1 2^d+c^*\boldsymbol{t}_0-c^*\boldsymbol{t}\tag{14}$$

又存在$\boldsymbol{t}=\boldsymbol{A}\boldsymbol{s}_1+\boldsymbol{s}_2$，代入式(14)得

$$\boldsymbol{A}(\boldsymbol{x}+\boldsymbol{y})+\boldsymbol{A}c^*\boldsymbol{s}_1+p\boldsymbol{t}_1 2^d+c^*\boldsymbol{t}_0-c^*(\boldsymbol{A}\boldsymbol{s}_1+\boldsymbol{s}_2)=\boldsymbol{A}(\boldsymbol{x}+\boldsymbol{y})+p\boldsymbol{t}_1 2^d+c^*\boldsymbol{t}_0-c^*\boldsymbol{s}_2\tag{15}$$

已知等式$\boldsymbol{v}=\boldsymbol{A}(\boldsymbol{x}+\boldsymbol{y})+\boldsymbol{t}_1 p2^d$成立，代入式(15)可得

$$\boldsymbol{A}(\boldsymbol{x}+\boldsymbol{y})+p\boldsymbol{t}_1 2^d+c^*\boldsymbol{t}_0-c^*\boldsymbol{s}_2=\boldsymbol{v}+c^*\boldsymbol{t}_0-c^*\boldsymbol{s}_2\tag{16}$$

已知$\boldsymbol{h}=\mathrm{MakeHint}_q(-c^*\boldsymbol{t}_0,\boldsymbol{v}-c^*\boldsymbol{s}_2+c^*\boldsymbol{t}_0,2\gamma_2)$，由引理1可得

$$\boldsymbol{v}_1'=\mathrm{UseHint}_q(\boldsymbol{h},\boldsymbol{A}z^*-c\boldsymbol{t}_1 2^d,2\gamma_2)=\mathrm{UseHint}_q(\boldsymbol{h},\boldsymbol{v}+c^*\boldsymbol{t}_0-c^*\boldsymbol{s}_2,2\gamma_2)=\mathrm{HighBits}_q(\boldsymbol{v}-c^*\boldsymbol{s}_2,2\gamma_2)\tag{17}$$

由于签名截取参数β的特殊设置，使得不等式$\|c^*\boldsymbol{s}_2\|_\infty\leqslant\beta$成立，同时在盲签名构造过程中进行了签名合法性判断（即算法10中的步骤21)），这也使得$\mathrm{LowBits}_q(\boldsymbol{v}-c^*\boldsymbol{s}_2)<\gamma_2-\beta$成立，也就是说$c^*\boldsymbol{s}_2$的存在不会对$\boldsymbol{v}-c^*\boldsymbol{s}_2$的高阶位产生影响，则由引理2可得

$$\mathrm{HighBits}_q(\boldsymbol{v}-c^*\boldsymbol{s}_2,2\gamma_2)=\mathrm{HighBits}_q(\boldsymbol{v}-c^*\boldsymbol{s}_2+c^*\boldsymbol{s}_2,2\gamma_2)=\mathrm{HighBits}_q(\boldsymbol{v},2\gamma_2)=\boldsymbol{v}_1\tag{18}$$

于是$\boldsymbol{v}_1'=\boldsymbol{v}_1$，即本文方案的正确性得以证明。

4 CDBS方案的安全性与效率分析

4.1 安全性分析

4.1.1 盲性分析

由定义5可知，盲签名方案盲性的存在使用户可以在不暴露隐私信息的情况下进行身份认证，确保数据的安全性和隐私性。假设存在攻击者E伪装成签名方与用户U进行盲签名的交互，首先由用户U从任意2个等长且不同的原始消息M_0和M_1中随机选取一个，并对该原始消息M_k（k=0或1）进行盲化处理，同时输出盲化后的数据c_b^*（b=0或1），若在概率统计上有$\left|\Pr\left(b=k\right)\right|>\frac{1}{2}$，则盲性验证失败，也就是说该盲签名方案不具有盲性，且由整体方案的步骤16)（算法9）可知

$$c^*=c+p=\text{SampleInBall}\left(\tilde{c}\right)+p=\text{SampleInBall}\left(H\left(\boldsymbol{v}_1\|\mu\right)\right)+p \tag{19}$$

在本文方案中，哈希函数CRH和H的具体实现算法均为SHAKE-256，不同之处在于函数CRH输出384位哈希值，而函数H输出256位哈希值。由于哈希函数具有单向性和抗碰撞性，同时其输出近似服从均匀分布，则对于不同的原始消息，签名方无法直接通过盲化后的数据进行区分，也就是说M_k是无法被辨别的，因此该盲签名方案是具备盲性的。

4.1.2 不可伪造性分析

由定义6可知，可设计下述安全模型来证明CDBS方案的不可伪造性。假设攻击者E生成k组不同的明文消息$M_i(i=1,2,\cdots,k)$，并与签名方进行k次交互，同时建立索引表T_H、T_S和T_V分别用于记录哈希查询、签名查询和验证查询，假设攻击者E可以通过不可忽略的概率$\text{negl}(\zeta)$成功伪造出第$k+1$次签名，则存在挑战者C可以近似$\text{negl}(\zeta)$的概率解决CRYSTALS-Dilithium所面临的困难问题。首先，系统初始化并通过密钥生成算法，生成整个CDBS方案中的公钥$\text{pk}=(\rho,\text{tr},\boldsymbol{t}_1)$和私钥$\text{sk}=(K,\boldsymbol{s}_1,\boldsymbol{s}_2,\boldsymbol{t}_0)$，随后攻击者通过下述查询伪造签名。

1) 哈希查询。攻击者E向挑战者C请求关于M_i的哈希值，若T_H中存在对应的请求值，则直接返回(M_k,H_k)；若不存在，则随机返回一个哈希值，并将其记录在表中。

2) 签名查询。攻击者E向挑战者C请求关于M_i的数字签名，若T_S中存在相应值，则直接返回该签名值Σ_i；若不存在，则通过签名算法生成该签名值，并对应将其记录在表中。

3) 验证查询。攻击者E向挑战者C请求关于M_i的验证查询，若T_V中存在对应值，则直接返回1，若不存在，则执行验证算法，验证成功返回1，否则返回0。

经过若干次盲签名的交互过程，攻击者E将在概率$\text{negl}(\zeta)$内伪造出M_j的签名$\Sigma_j=(\boldsymbol{z}_j^*,\boldsymbol{h}_j,\tilde{c}_j)$，且生成签名满足$\|\boldsymbol{z}_j^*\|_\infty<\gamma_1-\beta$，并在签名验证时使得$c_j^*=\text{SampleInBall}\left(H\left(\boldsymbol{v}_1^j\|\text{CRH}\left(\text{tr}\|M_j\right)\right)\right)+p$成立，此时若该签名合法有效，则攻击者E必须得到私钥$\text{sk}=(K,\boldsymbol{s}_1,\boldsymbol{s}_2,\boldsymbol{t}_0)$，这就相当于解决了MLWE和SIS问题，这与它们是困难问题相互矛盾，因此本文方案具有不可伪造性。

4.2 效率分析

4.2.1 理论分析对比

CDBS方案与CRYSTALS-Dilithium数字签名算法在基础参数的设置下保持一致，取$q=8\,380\,417$、$d=13$。假设安全参数ζ所占开销为256位即32 B，CDBS方案中公钥、私钥和签名长度的理论计算过程如下。

1) 公钥$\text{pk}=(\rho,\text{tr},\boldsymbol{t}_1)$。公钥部分由参数$\rho$、tr和$\boldsymbol{t}_1$组成，已知参数$\rho\in\{0,1\}^{256}$、$\text{tr}\in\{0,1\}^{384}$，因此CDBS方案的公钥长度为$32+48+32k\left(\log\lceil q\rceil-d\right)=(80+320k)$ B。

2) 私钥$\text{sk}=(K,\boldsymbol{s}_1,\boldsymbol{s}_2,\boldsymbol{t}_0)$。私钥部分由参数$K$、$\boldsymbol{s}_1$、$\boldsymbol{s}_2$和$\boldsymbol{t}_0$组成，且已知参数$K\in\{0,1\}^{256}$、$(\boldsymbol{s}_1,\boldsymbol{s}_2)\in S_\eta^l\times S_\eta^k$、$(\boldsymbol{t}_1,\boldsymbol{t}_0)=\text{Power2Round}_q(\boldsymbol{t},d)$，因此在CDBS方案中，私钥长度为$(32+32(k+l)\lceil\log(2\eta+1)\rceil+32kd)$ B。

3) 签名$\Sigma=(\boldsymbol{z},\boldsymbol{h},\tilde{c})$。签名$\Sigma=(\boldsymbol{z},\boldsymbol{h},\tilde{c})$中包含3个参数，向量$\boldsymbol{h}$中的所有多项式最多有$w$个非零系数，则在具体实现中，只需要存储这些非零系数的位置，因此需$(w+k)$ B，则签名长度为$(32l(\log 2\gamma_1)+(w+k)+32)$ B。

若忽略对存储开销影响较小的随机参数因素，在考虑方案中的主要参数（如向量、矩阵等）计算开销的情况下，将CDBS方案的理论开销与其他基于格的盲签名方案的计算开销进行对比，得到如表2所示的对比结果。在对比过程中需要注意的

表2　不同方案的密码参数理论分析对比

方案	公钥	私钥	签名	交互轮数/轮
文献[27]	$2k(\log q)$	$l\log q$	$l\log q$	3
文献[29]	$kl\log q$	$kl\log q$	$k\log q$	2
文献[30]	$kl\log 2q$	$kl\log 2q$	$k\log 12\sigma$	3
CDBS	$k(\log\lceil q\rceil - d)$	$(k+l)\lceil\log(2\eta+1)\rceil$	$l(\log 2\gamma_1)$	3

是，在CDBS方案中矩阵$\boldsymbol{A}\in R_q^{k\times l}$，而在其他方案中矩阵$\boldsymbol{A}\in\mathbb{Z}_q^{m\times n}$，则统一使用$(k,l)$表示生成矩阵的维数，令矩阵$\boldsymbol{A}\in R_q^{k\times l}$或$\boldsymbol{A}\in\mathbb{Z}_q^{k\times l}$。由于私钥范围的大小$\eta$的取值远小于$q$，则$\lceil\log(2\eta+1)\rceil \ll \log q$。同时在基础参数的设置中，若$\log q=23$、$\eta=2$和$\gamma_1=2^{17}$，此时$\log q>\log 2\gamma_1\gg\log 2\eta+1$。综上所述，CDBS方案的签名长度在理论层面具有显著优势。

4.2.2　方案性能对比

文献[20]是基于CRYSTALS-Dilithium数字签名方案初始版本所构建的一种抗量子盲签名方案——MBS方案，本文方案与MBS方案均基于CRYSTALS-Dilithium数字签名基础框架，其参数构成也相似，因此主要对这2个方案进行详细对比分析。参数设置如表3所示。

表3　参数设置

参数	定义	参数值
q	模数	8 380 417
d	$\boldsymbol{t}$中删除的位数	13
γ_1	向量$\boldsymbol{y}$的系数范围	2^{17}
γ_2	低阶舍入范围	$\frac{q-1}{88}$
(k,l)	矩阵维数	(4,4)
η	私钥范围的大小	2
β	$\tau\eta$	78
公钥长度/B	$80+32k(\log\lceil q\rceil-d)$	1 360
私钥长度/B	$32+32(k+l)\lceil\log(2\eta+1)\rceil+32kd$	2 464
签名长度/B	$32l(\log 2\gamma_1)+(w+k)+32$	2 420

在方案性能的对比测试中，主要对CDBS方案和MBS方案[20]在实际开销与软件运行效率2个方面进行测试分析，MBS方案与CDBS方案在公私钥长度及签名长度方面实际开销的对比结果如表4所示。

由对比结果可知，尽管在公钥长度上本文方案是MBS方案的1.04倍，但私钥长度是MBS方案的65.8%，签名长度是MBS方案的57.0%。同时相较于MBS方案，CDBS方案在整体的公私钥长度以及签名长度上减小了32.9%，即CDBS方案的整体开销是MBS方案的67.1%。同时，在算法的实现上CDBS方案仅需3轮交互，即该方案在整体算法的构造上也更为简洁，且易于实现。

表4　方案实际开销对比

方案	公钥长度/B	私钥长度/B	签名长度/B	交互轮数/轮
MBS	1 312	3 744	4 246	4
CDBS	1 360	2 464	2 420	3

4.2.3　性能测试与分析

在Linux操作系统下，使用C语言并调用OpenSSL库对所设计的基于CRYSTALS-Dilithium的盲签名方案进行软件层面的实现。测试环境配置如下。CPU：AMD R5-6600H 3.30 GHz，内存：16.0 GB（4 800 MHz），操作系统：Ubuntu 22.04.1 LTS。在上述实验环境中，随机生成一个59 B的消息并对其进行盲签名，重复运行该方案10 000次，并对这10 000次签名的运行效率取均值并输出具体测试结果。CDBS方案与CRYSTALS-Dilithium方案实际运算效率对比测试结果如图3所示。

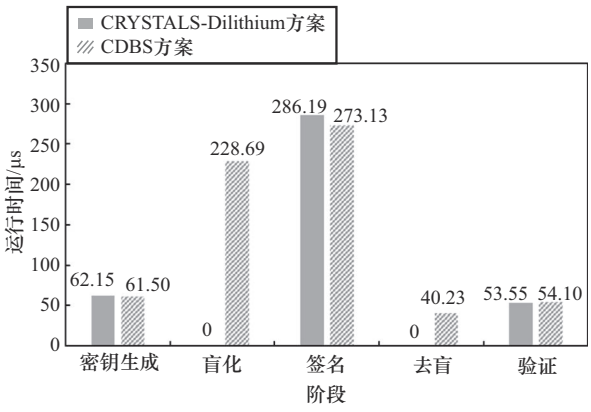

图3　CDBS方案与CRYSTALS-Dilithium方案实际运算效率对比测试

由测试结果可知，整体上CRYSTALS-Dilithium方案进行一轮签名及验证过程平均消耗401.89 μs；而

对于CDBS方案而言，由于在原有算法的基础上增加了盲化和去盲模块，即增加了原方案的算法复杂度，CDBS方案进行一轮签名及验证过程平均消耗657.65 μs。其在增加原方案算法复杂度的同时，整体运算效率也增加了255.76 μs。

同时，在相同的实验平台上，对MBS方案和CDBS方案进行软件层面的效率测试，其样本参数的选择如表3所示，测试结果如图4所示。

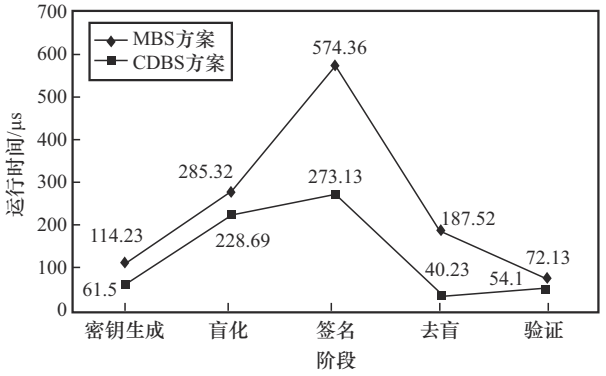

图4 CDBS方案与MBS方案实际运算效率对比测试

通过上述的测试与分析，本文设计的CDBS方案完成一轮完整的签名及验证过程（包括密钥生成、盲化、签名、去盲和验证5个阶段）耗时657.65 μs，其在实现效率上仅为微秒级别且方案流程简洁，具有较好的实际应用可行性。

5 结束语

本文在NIST公布的标准化后量子数字签名算法CRYSTALS-Dilithium的基础上，构造了一种抗量子盲签名方案CDBS。与传统的数字签名方案相比，该方案引入了盲签名的设计思想，可适用于电子票务、匿名支付、隐私保护等多种互联网应用场景。与基于传统数学难题的盲签名方案相比，CDBS方案基于MLWE和SIS的困难性假设，因而具有抗量子计算攻击的安全属性。在整体方案中，其公私钥长度和签名长度较短，从而极大地节约了存储资源，方案内部还使用NTT算法提高了签名速度。同时相较于其他基于格的盲签名方案，本文方案在构造方式上易于实现，在签名构成上更加精简，在安全性上更具优势。

参考文献：

[1] CHAUM D. Blind signatures for untraceable payments[C]//Advances in Cryptology. Berlin: Springer, 1983: 199-203.

[2] CAMENISCH J L, PIVETEAU J M, STADLER M A. Blind signatures based on the discrete logarithm problem[C]//Advances in Cryptology. Berlin: Springer, 1995: 428-432.

[3] POINTCHEVAL D, STERN J. Security proofs for signature schemes[C]//Advances in Cryptology. Berlin: Springer, 1996: 387-398.

[4] ABE M, FUJISAKI E. How to date blind signatures[C]//International Conference on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 1996: 244-251.

[5] RÜCKERT M. Lattice-based blind signatures[C]//International Conference on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 2010: 413-430.

[6] LYUBASHEVSKY V. Lattice signatures without trapdoors[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 2012: 738-755.

[7] LIU Q P, ZHANDRY M. Revisiting post-quantum fiat-shamir[C]//Annual International Cryptology Conference. Berlin: Springer, 2019: 326-355.

[8] LANYON B P, WEINHOLD T J, LANGFORD N K, et al. Experimental demonstration of a compiled version of shor's algorithm with quantum entanglement[J]. Physical Review Letters, 2007, 99(25): 250505.

[9] LONG G L. Grover algorithm with zero theoretical failure rate[J]. Physical Review A, 2001, 64(2): 022307.

[10] YESINA M V, OSTRIANSKA Y V, GORBENKO I D. Status report on the third round of the NIST post-quantum cryptography standardization process[J]. Radiotekhnika, 2022(210): 75-86.

[11] BOS J, DUCAS L, KILTZ E, et al. CRYSTALS-KYBER: a CCA-secure module-lattice-based KEM[C]//Proceedings of the 2018 IEEE European Symposium on Security and Privacy (EuroS&P). Piscataway: IEEE Press, 2018: 353-367.

[12] DUCAS L, KILTZ E, LEPOINT T, et al. CRYSTALS-Dilithium: a lattice-based digital signature scheme[J]. IACR Transactions on Cryptographic Hardware and Embedded Systems, 2018(1): 238-268.

[13] ODER T, SPEITH J, HÖLTGEN K, et al. Towards practical microcontroller implementation of the signature scheme FALCON[C]//International Conference on Post-Quantum Cryptography. Berlin: Springer, 2019: 65-80.

[14] BERNSTEIN D J, HÜLSING A, KÖLBL S, et al. The SPHINCS+signature framework[C]//Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2019: 2129-2146.

[15] LYUBASHEVSKY V. Fiat-Shamir with aborts: applications to lattice and factoring-based signatures[C]//International Conference on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 2009: 598-616.

[16] RICCI S, MALINA L, JEDLICKA P, et al. Implementing CRYSTALS-dilithium signature scheme on FPGAs[C]//Proceedings of the 16th International Conference on Availability, Reliability and Security. New York: ACM Press, 2021: 1-11.

[17] LAND G, SASDRICH P, GÜNEYSU T. A hard crystal-implementing dilithium on reconfigurable hardware[C]//International Conference on Smart Card Research and Advanced Applications. Berlin: Springer, 2022: 210-230.

[18] BECKWITH L, NGUYEN D T, GAJ K. High-performance hardware implementation of CRYSTALS-Dilithium[C]//Proceedings of the 2021 International Conference on Field-Programmable Technology (ICFPT). Piscataway: IEEE Press, 2021: 1-10.

[19] KARABULUT E, ALKIM E, AYSU A. Single-trace side-channel at-

tacks on ω-small polynomial sampling: with applications to NTRU, NTRU prime, and CRYSTALS-Dilithium[C]//Proceedings of the 2021 IEEE International Symposium on Hardware Oriented Security and Trust (HOST). Piscataway: IEEE Press, 2021: 35-45.

[20] LE H Q, SUSILO W, KHUC T X, et al. A blind signature from module latices[C]//Proceedings of the 2019 IEEE Conference on Dependable and Secure Computing (DSC). Piscataway: IEEE Press, 2019: 1-8.

[21] OMONDI A R. Modular reduction[C]//Cryptography Arithmetic. Berlin: Springer, 2020: 105-141.

[22] DUGUET A, ARTIGUES C, HOUSSIN L, et al. Properties, extensions and application of piecewise linearization for euclidean norm optimization in R^2[J]. Journal of Optimization Theory and Applications, 2022, 195(2): 418-448.

[23] 杨亚涛, 韩新光, 黄洁润, 等. 基于RLWE支持身份隐私保护的双向认证密钥协商协议[J]. 通信学报, 2019, 40(11): 180-186.
YANG Y T, HAN X G, HUANG J R, et al. Bidirectional authentication key agreement protocol supporting identity's privacy preservation based on RLWE[J]. Journal on Communications, 2019, 40(11): 180-186.

[24] NGUYEN T T, NGUYEN T T B, LEE H. An analysis of hardware design of MLWE-based public-key encryption and key-establishment algorithms[J]. Electronics, 2022, 11(6): 891.

[25] YANG Y T, ZHANG J M, HUANG J R, et al. Improved authenticated key agreement protocol based on Bi-ISIS problem[J]. The Journal of China Universities of Posts and Telecommunications, 2020, 27(3): 93-102.

[26] RAWAL S, PADHYE S. Untraceability of partial blind and blind signature schemes[C]//Proceedings of the 15th International Conferenceb Information Security and Cryptology (Inscrypt 2019). Berlin: Springer, 2020: 452-459.

[27] 黄秀菊, 杜云飞, 李子臣. 一种理想格上高效盲签名方案[J]. 计算机应用研究, 2022, 39(11): 3461-3464.
HUANG X J, DU Y F, LI Z C. Efficient blind signature scheme on ideal lattice[J]. Application Research of Computers, 2022, 39(11): 3461-3464.

[28] LI X, LU J, LIU D, et al. A high-speed post-quantum crypto-processor for crystals-dilithium[J]. IEEE Transactions on Circuits and Systems II: Express Briefs, 2023(71): 435-439.

[29] ZHANG P Y, JIANG H, ZHENG Z H, et al. A new post-quantum blind signature from lattice assumptions[J]. IEEE Access, 2018, 6: 27251-27258.

[30] LI C Y, TIAN Y, CHEN X B, et al. An efficient anti-quantum lattice-based blind signature for blockchain-enabled systems[J]. Information Sciences, 2021, 546: 253-264.

[作者简介]

杨亚涛（1978– ），男，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，主要研究方向为密码学与通信安全、后量子密码、全同态加密、密码协议和算法等。

常鑫（1997– ），男，甘肃定西人，西安电子科技大学硕士生，主要研究方向为后量子密码、格基后量子签名算法。

史浩鹏（1999– ），男，内蒙古呼伦贝尔人，西安电子科技大学硕士生，主要研究方向为格基后量子签名算法、全同态签名算法。

王伟（1998– ），男，山西运城人，西安电子科技大学硕士生，主要研究方向为信息安全、密码协议和隐私计算。

王克（1992– ），男，河南邓州人，北京电子科技学院讲师，主要研究方向为后量子密码算法设计与分析。