基于环上容错学习的异步分布式随机信标及其在异步共识中的应用

张宗洋[1)] 李天宇[1)] 胡 斌[1)] 周 游[1)] 周星光[2)]

[1)](北京航空航天大学网络空间安全学院 北京 100191)

[2)](中国民航管理干部学院大数据与人工智能系 北京 100102)

摘 要 安全高效的异步分布式随机信标是异步共识协议能够快速终止的必要保障.2008年以来,比特币中区块链技术的成功促进了分布式应用的蓬勃发展,为分布式应用提供随机性来源的分布式随机信标也受到了学者的广泛研究.现有的分布式随机信标一般选取公开可验证秘密共享、可验证随机函数、可验证延迟函数之一作为基础结构进行设计,其中公开可验证秘密共享结构简单、模块清晰、实用性较强,是现有分布式随机信标协议中数量最多、分析最为透彻的类别,其它结构的协议研究开始时间较晚,设计方法并未固定,在形式化安全定义和安全分析中尚有较大空白等待填补.目前,只有少数基于公开可验证秘密共享的方案支持异步网络模型,并且这些方案都是利用不抗量子的离散对数问题构建的.因此,在后量子安全的异步网络模型下,现有异步共识成果不得不采用期望运行轮数为指数的本地抛币协议来保障全栈量子安全.本文主要研究抗量子安全假设下基于公开可验证秘密共享的方案,具体贡献如下:(1)本文首先基于抗量子困难假设环上容错学习和公开可验证秘密共享的结构范式,设计了具备$O(N\log N)$证明及验证复杂度的公钥更新协议,其中N为多项式次数,该协议保证了节点在每一轮产生随机数份额的过程中采用了一致且正确的新公钥.(2)在此基础上,本文设计了计算复杂度为$O(n)$,通信复杂度为$O(n^2)$的异步分布式随机信标协议,其中n为节点规模.在相同的安全参数下,本文提出的随机信标协议相较于基于离散对数的协议运行时延减小约34%.(3)最后,本文给出了一种基于异步分布式随机信标的异步共识协议,并证明其满足标准的安全需求.

关键词 分布式随机信标;异步网络;量子安全;门限密码;零知识证明

中图法分类号 TP309 **DOI号** 10.11897/SP.J.1016.2024.01813

Ring Learning with Errors Based Asynchronous Distributed Random Beacon and its Application in Asynchronous Consensus

ZHANG Zong-Yang[1)] LI Tian-Yu[1)] HU Bin[1)] ZHOU You[1)] ZHOU Xing-Guang[2)]

[1)](*School of Cyber Science and Technology*, *Beihang University*, *Beijing* 100191)

[2)](*Department of Big Data and Artificial Intelligence*, *Civil Aviation Management Institute of China*, *Beijing* 100102)

Abstract A secure and efficient asynchronous distributed random beacon is necessary to guarantee the termination of asynchronous consensus. Since 2008, the success of blockchain technology in Bitcoin has promoted the vigorous development of distributed applications, and distributed random beacons that provide a source of randomness for distributed applications have also been widely studied by scholars. Although several distributed random beacons have been

收稿日期:2023-08-31;在线发布日期:2024-05-15. 本课题得到国家重点研发计划(2022YFB2702702)、国家自然科学基金面上项目(62372020, 61972017, 72031007)、北京市自然科学基金(M22038, L222050)、中央高校基本科研业务费(YWF-23-L-1032)、中国民航管理干部学院青年教师科研启动基金(23QN06)资助. **张宗洋**,博士,副教授,中国计算机学会(CCF)高级会员,主要研究领域为密码学与区块链. E-mail:zongyangzhang@buaa.edu.cn. **李天宇**,博士研究生,主要研究领域为密码学与区块链. **胡 斌**,博士研究生,主要研究领域为密码学与区块链. **周 游**,硕士研究生,中国计算机学会(CCF)学生会员,主要研究领域为密码学与区块链. **周星光**(通信作者),博士,副教授,主要研究领域为信息安全与密码学. E-mail:zhouxingguang@camic.cn.

proposed, the development of quantum computing necessitates reevaluating the security assumptions underlying existing schemes. In the field of cryptography, it is possible to quickly calculate the order of any element in a multiplication group using quantum Fourier transform, and then efficiently solve mathematical difficulties such as large integer factorization or discrete logarithm. Once quantum computers with a sufficient number of qubits become available, most protocols constructed based on these assumptions will be vulnerable to attacks. Consequently, international standardization organizations, headed by NIST, have initiated quantum-resistant cryptography protocol evaluations, recommending example instances for widely used protocols like public key encapsulations and digital signatures. However, the distributed random beacon protocols and asynchronous consensus have yet to be included in the evaluation scope. Most existing distributed random beacons can be divided into protocols based on public verifiable secret sharing, protocols based on verifiable random function, and protocols based on verifiable delay function. The public verifiable secret sharing model has a simple structure, clear modules, and strong practicality, making it the most numerous and thoroughly analyzed category among existing distributed random beacon protocols, the other structures started relatively late, and the design methods as well as the formal security definitions and analysis were not fixed. At present, only a few public verifiable secret sharing based schemes support asynchronous networks, and these schemes are constructed using the quantum-non-resistant discrete logarithm problem. Therefore, in the context of post-quantum security, existing asynchronous consensus solutions have to rely on local coins with exponential complexity to achieve full-stack quantum security. This paper does further research on public verifiable secret sharing and quantum-resistant security assumptions. The main innovations include the following aspects: (1) A public key update protocol with $O(N\log N)$ prover and verification complexity is designed first, where N is the order of a polynomial. This protocol is based on the quantum-resistant assumption ring learning with errors and the structural paradigm of public verifiable secret sharing, which ensures that the nodes use consistent and correct random seeds in each round of randomness generation. (2) Using the aforementioned public key update protocol, a quantum-resistant asynchronous distributed random beacon protocol based on public verifiable secret sharing is proposed, which ensures that the nodes who use the same random seeds in each round can generate a consistent and secure random beacon. This protocol achieves $O(n)$ computational complexity and $O(n^2)$ communication complexity, where n is the node size of the asynchronous network. Under the same security parameter $\lambda=256$, our protocol, implemented using Python, saves about 34% execution latency with $n=10$ compared to the variant protocol based on the discrete logarithm assumption. (3) This paper presents an asynchronous consensus based on the Quadratic-ABA in WaterBear using our asynchronous distributed random beacon protocol and proves it meets the standard security requirements.

Keywords distributed random beacon; asynchronous network; post-quantum; threshold cryptography; zero-knowledge proof

1 引 言

近年来,量子计算高速发展,基于数学困难问题的公钥密码算法如RSA、ECDSA等面临被量子计算机在多项式时间内破解的风险.因此,密码学家需要审视所有的密码算法和安全协议是否保障量子安全以健全信息安全底层体系.

量子计算的基础是量子电路模型[1],该模型以量子比特作为运算单元.量子比特是最基础的二维

量子系统，由希尔伯特空间中的向量表示，处于0与1的叠加态，并在观测时坍塌为确定的值. 多个量子比特间存在纠缠关系，使得q个量子比特所构成的系统在部分运算中呈现出2^q维的特点，进而在特定计算场景中极大地增强算力.

随着量子电路模型得到广泛认可，许多学者提出了能够解决特定问题的量子算法. 在密码学中，量子算法主要应用于攻击底层数学困难问题. 以最知名的Shor算法[2]为例，该算法能够利用量子傅里叶变换快速测量大乘法群中任意元素的阶数，从而高效解决大整数分解[3]与离散对数[4]等问题.

目前问世的量子计算机所包含的量子比特数尚不足以破解选取了标准参数的密码学方案，但国内外已经开展了相关研究. 以NIST为代表的国际标准化组织已经开始评估后量子密码协议，并针对公钥封装机制[5]、数字签名[6-8]等广泛应用的协议给出了推荐. 国内的研究者也提出了许多性能卓越的后量子密码协议[9-10]，但尚未进行过系统性的评选评估. 一般认为，仅基于编码、格上困难问题、多元二次方程、哈希函数等假设构造的密码协议能够抵抗目前已知的量子算法[11]，并在量子查询复杂度上得到证明，被视为抗量子安全的.

本文所讨论的是异步共识协议[12-15]，该机制能够在通信层面维护系统安全，为分布式系统及其所下辖的云计算、区块链等行业提供必要支持，被广泛应用于分布式金融、互联网、数字藏品、电子存证、企业信息化、供应链与物流、政府及公共服务等领域. 异步共识协议存在大量投票环节，系统中的参与节点在无法判断当前轮优势投票时会通过抛币进行选择.

抛币模块的设计一般有两种设计路线：直接执行本地抛币或使用异步分布式随机信标. 执行本地抛币协议无需密码学假设[16-17]，能够保障量子安全或信息论安全，但期望轮数为指数级；使用异步分布式随机信标(Asynchronous Distributed Random Beacon，ADRB)可以满足常数轮期望终止[18-19]，但现有协议不能兼顾异步网络假设、高效性能指标和量子安全背景的三项设计需求.

国内外鲜有量子安全的高效异步分布式随机信标研究成果的原因在于量子安全的概念提出较晚，强调量子安全的异步共识领域一般将异步分布式随机信标作为黑盒使用，从而着重研究高层协议的量子安全性；而研究分布式随机信标的安全多方计算领域更关注于优先满足同步网络下的量子安全，对异步网络模型研究相对较少. 针对这一异步共识协议与安全多方计算的交叉领域实际问题，本文的核心工作是设计了一个可证安全的异步分布式随机信标协议. 该协议仅基于格上困难问题环上容错学习(Ring Learning with Errors，RLWE)和无碰撞哈希函数假设，满足后量子密码学的一般定义[11]. 它填补了异步共识协议在追求抗量子安全的过程中，理论分析完善但尚无对应抗量子安全抛币模块调用的遗留问题.

本文的全部贡献可以总结为以下方面：

(1)提出了一个基于RLWE的公钥更新协议. 该协议为非交互式零知识证明协议，在多项式环次数N下的证明复杂度和验证复杂度均为$O(N\log N)$，能够为多项式环上的任意元素生成证明并进行验证.

(2)提出了基于RLWE的异步分布式随机信标协议. 该协议采用公开可验证秘密共享的实现结构，在可信初始化的条件下，借助同态映射可产生不限量的随机信标. 对于节点规模n，该协议的计算复杂度为$O(n)$，通信复杂度为$O(n^2)$.

(3)将所提出的异步分布式随机信标协议应用到异步共识协议中，并证明其满足标准安全需求.

(4)在本地单机测试环境下，将本文提出的随机信标协议和基于离散对数的协议进行性能对比. 测试结果表明，当节点规模$n=10$时，本协议所需的时延降低了34%.

2　相关工作

目前，学界针对分布式随机信标进行了大量研究，根据所依赖的密码学原理，现有的分布式随机信标主要可分为基于公开可验证秘密共享(Public Verifiable Secret Sharing，PVSS)的协议、基于可验证随机函数(Verifiable Random Function，VRF)的协议、基于可验证延迟函数(Verifiable Delay Function，VDF)的协议，此外还有少量基于同态映射(Homomorphism，Homo.)或区块链中工作量证明(Proof of Work，PoW)、权益证明(Proof of Stake，PoS)的其它协议. 表1列出了分布式随机信标领域较为重要的研究成果，表中n代表系统中的总节点数，t代表系统中的腐化节点数；c代表协议中哈希值长度等其他参数.

表1 分布式随机信标协议总结

分类	名称	年份	敌手模型	通信复杂度	安全假设	网络模型
PVSS	OptRand[20]	2022	$n=2t+1$	$O(n^2)$	q-SDH, SXDH	同步网络
PVSS	SPURT[21]	2022	$n=3t+1$	$O(n^2)$	DBS	半同步网络
PVSS	GRandPiper[22]	2021	$n=2t+1$	$O(n^2)$	q-SDH, SXDH	同步网络
PVSS	ALBATROSS[23]	2020	$n=2t+1$	$O(n^2)$	DDH	同步网络
PVSS	HydRand[24]	2020	$n=3t+1$	$O(n^2)$	DDH	同步网络
PVSS	RandHerd[25]	2017	$n=3t+1$	$O(c^2\log n)$	DL	同步网络
PVSS	RaandHound[25]	2017	$n=3t+1$	$O(c^2n)$	DL	同步网络
PVSS	RandShare[25]	2017	$n=3t+1$	$O(n^3)$	DL	异步网络
PVSS	Scrape[26][27]	2017	$n=2t+1$	$O(n^4)$	DDH	同步网络
PVSS	Cachin[19]	2005	$n=3t+1$	$O(n^2)$	Sig., DDH	异步网络
VRF, PVSS	GULL[28]	2023	$n=2t+1$	$O(n^2)$	DDH	同步网络
VRF	GLOW[29]	2021	$n=2t+1$	$O(cn\log n)$	DDH	同步网络
VRF	Dfinity[1]Dfinity technology overview series, consensussystem. https://arxiv.org/abs/1805.04548	2018	$n=2t+1$	$O(n^2)$	Sig., CDH	同步网络
VRF	Algorand[30]	2017	$n=3t+1$	$O(cn)$	Hash, BA	半同步网络
PoS, VDF	GasRNG[31]	2024	-	$O(n)$	GGH	同步网络
VDF	STROBE[32]	2023	$n=3t+1$	$O(n^2)$	RSA	异步网络
VDF	CRAFT[33]	2023	$n=2t+1$	$O(n^2)$	TLP	半同步网络
VDF	RandRunner[34]	2021	$n=2t+1$	$O(n^2)$	DL	同步网络
VDF	TARDIS[35]	2021	两方协议	—	TLP	半同步网络
PoW, VDF	RandChain[36]	2020	$n=2t+1$	$O(n^2)$	Hash	同步网络
PoS, VRF	SnowWhite[37]	2019	$n=2t+1$	$O(cn^2)$	Sig., Hash	同步网络
PoS, PVSS	Ouroboros[38]	2017	$n=2t+1$	$O(n^4)$	Sig., Hash	同步网络
PVSS	**ADRB$_{RLWE}$(本协议)**	**—**	$\boldsymbol{n=3t+1}$	$\boldsymbol{O(n^2)}$	**RLWE**	**异步网络**

注:安全假设中,q-SDH代表q强DH(q-Strong Diffie-Hellman)假设,SXDH代表对称外部DH(Symmetric External Diffie-Hellman)假设,DBS代表判定性双线性平方(Decisional Bilinear Squaring)假设,DDH代表判定性DH(Decisional Diffie-Hellman)假设,DL代表离散对数(Discrete Logarithm)假设,Sig. 代表数字签名,Hash代表哈希函数,CDH代表计算性DH(Computational Diffie-Hellman)假设,BA代表拜占庭协议(Byzantine Agreement),GGH代表理性人的博弈论假设,RSA代表大整数分解,TLP代表时间锁问题(Time Lock Puzzle),RLWE代表环上容错学习(Ring Learning With Errors)假设.

对异步分布式随机信标的研究主要集中在安全多方计算领域. 该协议最早由Rabin[39]提出,借助于数字签名和公钥基础设施,一个可信第三方可以随机生成一个所需长度的随机数,将其作为秘密采用Shamir秘密分享[40]为n个参与者分别生成一个子密钥. 足够数量的参与者可以将子密钥聚合恢复出秘密,并将该随机数的指定位作为各轮抛币的结果. 为防止参与者在秘密分享的聚合中作恶如发送无效份额导致秘密恢复失败,Stadler[41]在1996年提出PVSS,通过在广播份额阶段同时发送供接收者验证的对该份额的承诺来拒绝无效份额. PVSS模型结构简单、模块清晰、实用性较强,是现有分布式随机信标协议中数量最多、分析最为透彻的类别. 2005年,Cachin、Kursawe、Shoup[19]采用PVSS模型,借助随机谕言机(Random Oracle, RO)模型和离散对数(Discrete Logarithm, DL)假设,设计了可证明安全的异步分布式随机信标协议. 2017年,Syta等人[25]也选用PVSS模型提出了不使用密钥分发协议的异步分布式随机信标协议族,但其中应用于异步网络的RandShare通信复杂度为$O(n^3)$,计算复杂度为$O(n^2)$,与同类方案存在性能差距. 最近提出的使用PVSS构造的协议是Das等人[21]的SPURT和Bhat等人[22]的OptRand,两方案均包含了由普通节点运行秘密分享部分并将相关结果发送给领导节点进行验证的步骤,体现出分布式随机信标与共识协议的联系,但此类做法一般不支持异步网络环境.

除PVSS外,Dfinity中选用BLS门限签名构造了基于VRF的分布式随机信标;CRAFT将TARDIS中的两方协议拓展到多方,提出了VDF范畴的公开可验证的时间锁问题,并利用这一原语构

建延迟加密模块实现了分布式随机信标；RandChain等方案结合了区块链挖矿机制，设计了紧密结合公链系统的分布式随机信标. 上述类别的研究开始时间较晚，设计方法并未固定，在形式化安全定义和安全分析中尚有较大空白等待填补.

3　背景知识

3.1　密码学原语

3.1.1　可搜索RLWE问题

可搜索RLWE问题最早由Lyubashevsky、Peikert、Regev[42]提出. 考虑多项式环$\mathbb{Z}_p[X]/f(x)$中$f(x)=X^N+1$(N为2的幂)的特殊情况，在环上取秘密多项式s(即$s=s(x)$，下同)，随机选取公共多项式向量$\boldsymbol{a}\in(\mathbb{Z}_p[X]/f(x))^K$，再随机选取噪声多项式向量$\boldsymbol{e}=(e_1,\cdots,e_k)^T$, $e_i=\sum_{j=0}^{N-1}e_{ij}X^j$，其中$e_{ij}\leftarrow\chi$，$\chi$为取值在$\mathbb{Z}_p$上的概率分布，计算另一公共多项式向量$\boldsymbol{b}=\boldsymbol{a}s+\boldsymbol{e}$.

定义1.　可搜索RLWE问题. 令函数$\mathrm{Supp}(\chi)$代表χ分布的上界，可搜索RLWE问题$\mathrm{RLWE}(N,K,p,\chi)$：给定公钥$(\boldsymbol{a},\boldsymbol{b})$，求取$s'\in\mathbb{Z}_p[X]/f(x)$，使得$\boldsymbol{e}'=\boldsymbol{a}s'-\boldsymbol{b}$满足$\|\boldsymbol{e}'\|_\infty<\mathrm{Supp}(\chi)$，其中向量$\boldsymbol{e}'$的无穷范数$\|\boldsymbol{e}'\|_\infty=\max(e'_{ij})$.

可搜索RLWE问题可归约到理想格上最坏情况的近似最短向量问题，后者的量子查询复杂度为指数级的，被认为是量子安全的困难问题假设.

3.1.2　零知识证明

零知识证明是一类双方协议，其参与者被称为证明者P和验证者V，其中P具有无限计算能力，V为概率多项式时间(Probabilistic Polynomial Time，PPT)验证者. 对于公开的陈述x，秘密的证据w，和一组二元关系R，零知识证明允许证明者说服验证者其知晓证据w即$(x,w)\in R$，且除了声明本身已经揭示的信息之外，协议本身不透露任何关于秘密的信息.

定义2.　零知识证明. 零知识证明由公共字符串生成、证明、验证三个PPT算法组成，即$\mathrm{ZkP}=\{\mathrm{KGen},\mathrm{Prove},\mathrm{Verify}\}$. P和V事先运行KGen，根据安全参数λ约定公共字符串；证明者P运行Prove，对证据w生成证明$proof$；验证者在不掌握w的前提下根据陈述x和公共字符串进行验证$proof$，输出$b=\{0,1\}$代表拒绝或接受证明. 需要指出，P和V在运行Prove和Verify时根据协议的具体构造可以是交互的也可以是非交互的.

零知识证明应满足正确性、特殊可靠性、诚实验证者零知识性的要求[43].

(1)正确性：对于陈述x和证据w与二元关系R，若$(x,w)\in R$，则验证者V以概率p接受证明，其中p不为可忽略函数$\mathrm{negl}(\lambda)$.

(2)特殊可靠性：存在一个PPT算法A_E(知识提取器)，输入m个被接受的证明所产生的三元组$(t_i,c_i,z_i)(i\in[1,\cdots,m])$，能够输出$w'$使得$(x,w')\in R'$. 其中$x$为诚实证明的陈述，三元组中的$t_i$为第$i$个被接受的证明中对随机数的承诺.

(3)诚实验证者零知识性：存在一个PPT算法A_S(模拟器)，将陈述$x\in\mathcal{L}(R)$和挑战$c\in\mathcal{C}$作为输入，输出三元组(t,c,z)，其分布与诚实验证者真实运行协议的过程中所产生的对应值是计算不可区分的. 其中，t为对随机数的承诺，$\mathcal{L}(R)$被定义为$\mathcal{L}(R)=\{x,\exists w\ s.t.\ (x,w)\in R\}$.

3.1.3　承诺协议与RLWE标准承诺

承诺协议是一类双方协议，参与者称为承诺方和验证方，此类协议主要用于确保承诺方事先确定的一个秘密消息不可更改.

定义3.　承诺协议. 承诺协议由公共参数生成、承诺、验证三个概率多项式时间(Probabilistic Polynomial Time，PPT)算法组成，即$\mathrm{Comm}=\{\mathrm{Gen},\mathrm{Com},\mathrm{Ver}\}$. Gen接受参数输入$1^t$，输出用于承诺的公钥$pk_{comm}$；Com接受消息空间上的输入消息$mes$和承诺公钥$pk_{comm}$，输出承诺值$com$和打开承诺所需的打开参数$dec$；Ver将所生成的$pk_{comm}$、$mes$、$com$、$dec$作为输入，输出$b=\{0,1\}$代表拒绝或接受承诺.

承诺协议应满足正确性、计算绑定性、计算隐藏性的要求[43].

(1)正确性：若承诺方和验证方均诚实，则对承诺的验证通过，即：

$$\Pr\left[\begin{array}{c}\mathrm{Ver}(pk_{comm},mes,com,dec)=1:\\ pk_{comm}\leftarrow\mathrm{Gen}(1^t),\\ (com,dec)\leftarrow\mathrm{Com}(mes,pk_{comm})\end{array}\right]=1$$

(2)计算绑定性：任何PPT敌手$\mathcal{A}$都只有可忽略的概率找到承诺的碰撞，即：

$$\Pr\left[\begin{array}{c}\mathrm{Ver}(pk_{comm}, mes, com, dec)=\\ \mathrm{Ver}(pk_{comm}, mes', com, dec'):\\ pk_{comm}\leftarrow\mathrm{Gen}(1^t),\\ (com, mes, dec, mes', dec')\\ \leftarrow\mathcal{A}(pk_{comm})\end{array}\right]=\mathrm{negl}(\lambda)$$

(3)计算隐藏性:任何PPT敌手$\mathcal{A}'$都只有可忽略的概率从承诺中区分秘密消息,即:

$$\Pr\left[\begin{array}{c}b=b':\\ pk_{comm}\leftarrow\mathrm{Gen}(1^t),\\ (mes_0, mes_1)\leftarrow\mathcal{A}_1(pk_{comm});\\ b\leftarrow\{0,1\},\\ (com, dec)\leftarrow\mathrm{Com}(mes_b, pk_{comm});\\ b'\leftarrow\mathcal{A}_2(com, aux)\end{array}\right]=\frac{1}{2}+\mathrm{negl}(\lambda)$$

其中各输入应满足取值空间要求,λ为安全参数,aux为辅助输入,所有的$\leftarrow$包含随机性.

下面介绍RLWE标准承诺方案.

定义4. RLWE标准承诺[44]. RLWE标准承诺即可搜索RLWE问题的标准形式$\boldsymbol{b}=\boldsymbol{a}s+\boldsymbol{e}$,其中:

$$mes=s,\ pk_{comm}=\boldsymbol{a},\ com=\boldsymbol{b},\ dec=\boldsymbol{e}$$
$$\boldsymbol{a}\leftarrow\mathrm{Gen}(1^t),\ (\boldsymbol{b},\boldsymbol{e})\leftarrow\mathrm{Com}(s,\boldsymbol{a}),\ b\leftarrow\mathrm{Ver}(\boldsymbol{a}, s, \boldsymbol{b}, \boldsymbol{e})$$

具体的验证约束和安全级别因参数的选择而不同.

本文在第4节使用了RLWE标准承诺用于构建公钥更新协议,所设计的公钥更新协议的可靠性依托于RLWE标准承诺的计算绑定性. 对于RLWE下的二元关系$\mathcal{R}=\{((\boldsymbol{a},\boldsymbol{b}),(s,\boldsymbol{e})):\boldsymbol{b}=\boldsymbol{a}s+\boldsymbol{e}\wedge\|\boldsymbol{e}\|_\infty<N)\}$,RLWE标准承诺的计算绑定性定义为:若$((\boldsymbol{a},\boldsymbol{b}),(s_1,\boldsymbol{e}_1))\in\mathcal{R}$,$((\boldsymbol{a},\boldsymbol{b}),(s_2,\boldsymbol{e}_2))\in\mathcal{R}$,则$s_1=s_2$.

3.1.4 同态映射与CKKS同态映射方案

同态映射在密码编码学中具有重要意义,它作用在两个相同类型的代数结构间,并使一种代数结构内的运算等价于另一种代数结构内的运算. 下面给出同态映射的定义.

定义5. 同态映射. 一个从集合A到集合A'的映射φ,称为一个对集合A上的二元运算$\odot$和集合A'上的二元运算$\circledcirc$的同态映射,当且仅当对集合A中的任意两个元素a和b,若$a\rightarrow a'$,$b\rightarrow b'$($a',b'\in A'$),则有$a\odot b\rightarrow a'\circledcirc b'$. 可逆的同态映射可以应用在编码与解码中,一般用Encode(·)和Decode(·)表示.

若在集合A和集合A'上定义了加法,则可将该同态关系称之为加法同态;若在集合A和集合A'上定义了乘法,则可将该同态关系称之为乘法同态. 既满足加法同态又满足乘法同态的映射称为全同态映射,仅满足一种同态关系的映射称为半同态映射.

本文在第5节使用了CKKS同态映射方案[45],下面介绍该方案的构造与满足的性质.

定义6. CKKS同态映射方案. CKKS同态映射CKKSEncoder(N, $scale$)的全局输入参数为数据规模N和放缩倍数$scale$,共包含三个子算法:用于确定多项式次数和算法精度的初始化算法Initialize、为指定向量生成多项式的编码算法Encode、将编码后的多项式还原为向量的解码算法Decode. CKKS同态映射的具体流程如算法1至算法3所示.

算法1. CKKS同态映射初始化Initialize

输入:(N, $scale$)

// 环上多项式次数、放缩倍数

输出:$InstanceParameter$

// 实例参数

1. 确认N为2的幂,计算分圆多项式次数$M=2N$和输入规模$N'=N/2$
2. 计算M次分圆多项式的本原单位根$x_1,\cdots,x_N$,其中$x_1=e^{(2\pi/M)i}$,$x_t={x_1}^{2t-1}$,i为虚数单位
3. 计算正交基底$\boldsymbol{b}=(b_0,\cdots,b_{N-1})$,其中$b_t=({x_1}^t,\cdots,{x_N}^t)$
4. $InstanceParameter=(M, N', (x_1,\cdots,x_N), \boldsymbol{b})$

算法2. CKKS同态映射编码Encode

输入:$((y_0,\cdots,y_{N'-1}), InstanceParameter)$

// $\mathbb{C}^{N'}$上的输入向量、实例参数

输出:$P(x)$

// 编码多项式

1. 将输入的每个元素乘$scale$得到$\boldsymbol{z}'=(z_0,\cdots,z_{N'-1})$
2. 利用共轭根扩展为N长复数向量$\boldsymbol{z}=(z_0,\cdots,z_{N-1})$,其中$z_t(t<N)$的共轭复根为$z_{N-t-1}$
3. 计算多项式的k次系数${\alpha'}_k=\langle z_t,b_t\rangle/N$,得到向量$\boldsymbol{\alpha}'=({\alpha'}_0,\cdots,{\alpha'}_{N-1})$
4. 将向量$\boldsymbol{\alpha}'$中的每个系数取实部并圆整,得到系数向量$\boldsymbol{\alpha}=(\alpha_0,\cdots,\alpha_{N-1})\in\mathbb{Z}^N$
5. 根据系数向量$\boldsymbol{\alpha}$生成多项式$P(x)\in\mathbb{Z}[X]/(X^N+1)$

算法3. CKKS同态映射解码 Decode

输入:($P(x)$, $InstanceParameter$)

// 编码多项式、实例参数

输出:$\boldsymbol{y}$

// 解码后的向量

1. 计算向量$z=(P(x_1),\cdots,P(x_{N'}))$

2. 将向量z的每一个元素除以$scale$并输出得到向量y

CKKS同态映射算法满足加法同态性、乘法同态性和数乘不变性，属于全同态映射. 同时，通过调整放缩倍数$scale$，CKKS同态映射编码后的精度损失和解码后的错误概率是可达到任意小：

1. 加法同态性：$\forall \boldsymbol{y},\bar{\boldsymbol{y}}\in\mathbb{C}^{N'},\ \forall\varepsilon>0,\ \exists scale>0,\ s.t.\ \left|Encode_{scale}(\boldsymbol{y})+Encode_{scale}(\bar{\boldsymbol{y}})-Encode_{scale}(\boldsymbol{y}+\bar{\boldsymbol{y}})\right|<\boldsymbol{P}_{\varepsilon}$，其中$\boldsymbol{P}_{\varepsilon}=\sum_{j=0}^{N-1}\varepsilon X^j$.

2. 乘法同态性：$\forall \boldsymbol{y},\bar{\boldsymbol{y}}\in\mathbb{C}^{N'},\ \forall\varepsilon>0,\ \exists scale>0,\ s.t.\ \left|Encode_{scale}(\boldsymbol{y})\cdot Encode_{scale}(\bar{\boldsymbol{y}})-Encode_{scale}(\boldsymbol{y}\cdot\bar{\boldsymbol{y}})\right|<\boldsymbol{P}_{\varepsilon}$，其中$\boldsymbol{y}\cdot\bar{\boldsymbol{y}}$定义为向量各分量分别相乘，$\boldsymbol{P}_{\varepsilon}=\sum_{j=0}^{N-1}\varepsilon X^j$.

3. 数乘不变性：$\forall a\in\mathbb{C},\ \boldsymbol{y}\in\mathbb{C}^{N'},\ \forall\varepsilon>0,\ \exists scale>0, s.t.\ \left|a\cdot Encode_{scale}(\bar{\boldsymbol{y}})-Encode_{scale}(a\cdot\bar{\boldsymbol{y}})\right|<\boldsymbol{P}_{\varepsilon}$，其中$\boldsymbol{P}_{\varepsilon}=\sum_{j=0}^{N-1}\varepsilon X^j$.

3.1.5　异步分布式随机信标协议

异步分布式随机信标是一类门限协议，能够在可信初始化的前提下产生不限量的随机信标，是异步共识协议能够快速终止的必要保障，下面给出异步分布式随机信标的定义.

定义7.　异步分布式随机信标. 异步分布式随机信标协议由密钥生成、份额揭示、分享验证、聚合输出四个算法组成，即：

ADRB＝

$\{\text{Gen, Reveal, ShareVrfy, ShareCombine}\}$

Gen(n,k,t)：该算法的输入包括参与节点总数n，聚合随机信标所需门限值k，可容忍的敌手数量t. 该算法由可信第三方运行，产生公共参数vk并广播，同时产生n组公私钥对(pk_i,sk_i). 可信第三方广播pk_i并将sk_i发送给对应的参与者P_i.

Reveal(C,sk_i)：该函数由各诚实参与方P_i分别运行，以硬币唯一标识符C，参与方私钥sk_i作为输入，产生硬币份额$\overline{pk}_i$和证据$\overline{vk}_i$，其中$\overline{pk}_i$也被称之为临时公钥. 各诚实参与方应广播$(\overline{pk}_i,\overline{vk}_i)$.

ShareVrfy$(pk_i,\overline{pk}_i,vk,\overline{vk}_i)$：该函数由收到$(\overline{pk}_i,\overline{vk}_i)$的另一参与方$P_j(j\neq i)$运行. 输入公钥$pk_i$、硬币份额$\overline{pk}_i$、公共参数$vk$和证据$\overline{vk}_i$，用以验证硬币份额$\overline{pk}_i$的有效性，输出0或1.

ShareCombine$(C,(\overline{pk}_{\alpha_1},\overline{pk}_{\alpha_2},\cdots,\overline{pk}_{\alpha_k}))$：该函数由各诚实参与方$P_i$分别运行. 从收到的$(\overline{pk}_i,\overline{vk}_i)$中选取$k$个通过验证的$\overline{pk}_i$，将对应的参与方编号$i$记为$\alpha_1,\cdots,\alpha_k$. 输入硬币的唯一标识符$C$和$(\overline{pk}_{\alpha_1},\overline{pk}_{\alpha_2},\cdots,\overline{pk}_{\alpha_k})$，输出随机信标$F(C)$（其中$F$为伪随机函数）. 可以将$F(C)$作为种子利用随机数生成器RG进行延展.

异步分布式随机信标协议应满足正确性、鲁棒性和不可预测性.

(1)正确性：在第三方正确初始化协议、诚实参与者的所有消息都递送完毕的前提下，所有的诚实参与者都以$1-\text{negl}(\lambda)$的概率取得同一个随机信标$F(C)$.

(2)鲁棒性：敌手仅有可忽略的概率伪造出可以通过验证的$(\overline{pk}_i,\overline{vk}_i)$.

(3)不可预测性：敌手掌握小于k个可通过验证的份额时，仅有可忽略的概率输出有效的随机信标$F(C)$.

3.2　系统模型

3.2.1　网络及敌手模型

本文采用完全异步的网络模型，即由可靠异步认证信道组成的点对点网络，收到消息的节点能够确认消息的来源节点. 纯异步网络中消息的传输时延不存在上限，但最终一定可以送达. 敌手在纯异步网络下可以完全控制信道中消息的调度，对消息进行阻塞、重排，但不能对消息进行插入、删除或修改.

本文采用静态的拜占庭敌手腐化模型，即假设系统中共有n个节点，其中有t个恶意节点完全由敌手控制（保证$3t+1\leqslant n$）. 在协议执行之前，敌手任意选取t个节点进行腐化，其初始状态（私钥等）将完全被敌手获知. 在协议执行过程中，敌手可以控制恶意节点执行任意行为，即节点会出现拜占庭式错误，但敌手不能对腐化的节点进行更换.

3.2.2　初始化模型

本文采用可信初始化模型. 每个节点由整数i唯一标识，其中$i\in[1,n]$，节点P_i及其公钥pk_i等都是公开的. 本文仅需在初始化阶段引入可信第三方，用于选取公共参数，生成并分发公私钥对(pk_i,sk_i)，广播公共值等，后续过程无需可信第三方的参与.

4 基于RLWE的公钥更新协议

4.1 协议概述

本文提出的异步分布式随机信标协议通过将节点P_i的公钥作为随机信标份额参与代数运算的方法使诚实节点聚合出正确且一致的随机信标.为使敌手不能够在所有诚实节点都激活同一次异步分布式随机信标协议前获知随机信标的值,需要各节点自主生成临时公钥pk_{new}以代替长期公钥pk_{old}作为份额$seed_i$参与代数运算.因此,异步分布式随机信标协议需要将本协议作为子模块,使接收节点P_j利用广播份额的节点P_i的公钥pk_{old}对份额pk_{new}进行验证,以保证诚实节点在每一轮产生随机数份额的过程中采用了一致且正确的新公钥,进而能够聚合出相同的随机信标.

在公钥密码系统中,公钥更新是指在同一代数结构下,依靠替换公共参数D_{old}为D_{new}的方法,将公钥由pk_{old}更换为pk_{new}(如在乘法群$\mathbb{Z}_p$中更改生成元,而私钥指数不变).公钥更新是一个零知识证明协议,由证明者将新公钥pk_{new}和证明$proof$.该方案能够在不泄露私钥sk的前提下使网络中的其它参与者相信pk_{new}是通过公钥更新生成的新公钥.

为了适配基于PVSS结构的异步分布式随机信标设计,需要使用非交互的公钥更新协议.在随机谕言机模型下,可通过Fiat-Shamir变换[46]将零知识证明协议中的挑战值从由验证者生成替换为由随机谕言机生成,以构造非交互的公钥更新协议.

首先给出随机谕言机的定义.

定义8. 随机谕言机[47].随机谕言机$\mathcal{O}$是一个真随机函数.更具体的,随机谕言模型假设存在一个公共的随机函数H,该函数仅能通过查询可被视为黑盒构造的谕言机获取数值,即当输入x时返回输出$H(x)$.

Fiat-Shamir变换将公共参数D_{old},D_{new},公钥pk_{old},pk_{new},使用两公共参数对随机数的承诺$\overline{pk_{old}}$,$\overline{pk_{new}}$作为随机谕言机$\mathcal{O}$的输入,并将随机函数H的输出作为挑战值.通过该变换,证明者P不再需要等待验证者V返回挑战,因此可实现一次通信的非交互式协议.下面给出基于RLWE问题的非交互式公钥更新协议的形式化定义.

定义9. 非交互式RLWE公钥更新协议.非交互式RLWE公钥更新协议:

$$\text{RLWEPkUpdate}=\{\text{KGen, Prove, Verify}\}$$

由三个算法KGen、Prove和Verify组成,协议中各参数的取值空间如下:

(1)协议的私钥空间$\mathcal{S}$为多项式环:

$\mathbb{Z}_p[X]/(X^N+1)$,其中p为素数.

(2)协议的公钥空间$\mathcal{P}$为由多项式环组成的长度为K的列向量$(\mathbb{Z}_p[X]/(X^n+1))^K$.

(3)协议的公共参数:

$$\boldsymbol{a}_{old},\ \boldsymbol{a}_{new}\in(\mathbb{Z}_p[X]/(X^N+1))^K.$$

(4)挑战空间:

$$\mathcal{C}=\{d\in\{0,1\}^N:||d||_1<\kappa\wedge deg(d)<N/2\}.$$

(5)挑战谕言机$H_c:(\mathbb{Z}_p[X]/(X^N+1))^{6k}\rightarrow$.

(6)KGen(1^λ):产生公共参数$p\equiv 3\ mod\ 8$,N为2的幂;多项式向量$\boldsymbol{a}_{old},\ \boldsymbol{a}_{new}\leftarrow(\mathbb{Z}_p[X]/(X^N+1))^K$,误差多项式$\boldsymbol{e}_{old},\ \boldsymbol{e}_{new}\leftarrow\mathcal{S}$,$\mathcal{S}$为$\mathcal{P}$的子集;证明者拥有私钥$s\in\mathbb{Z}_p[X]/(X^N+1)$作为证据,验证者拥有证明者经过验证的旧公钥$\boldsymbol{b}_{old}=\boldsymbol{a}_{old}s+\boldsymbol{e}_{old}$和未经过验证的新公钥$\boldsymbol{b}_{new}=\boldsymbol{a}_{new}s+\boldsymbol{e}_{new}$.

(7)Prove$(s,\ \boldsymbol{a}_{old},\ \boldsymbol{b}_{old},\ \boldsymbol{e}_{old},\ \boldsymbol{a}_{new},\ \boldsymbol{b}_{new},\ \boldsymbol{e}_{new})$:证明者选取临时的随机误差$\overline{\boldsymbol{e}_{old}}$和$\overline{\boldsymbol{e}_{new}}$,并计算称为临时公钥的随机数$\overline{\boldsymbol{b}_{old}}$和$\overline{\boldsymbol{b}_{old}}$,以生成对$s$的证明$proof$.

(8)Verify$(\boldsymbol{a}_{old},\ \boldsymbol{b}_{old},\ \boldsymbol{a}_{new},\ \boldsymbol{b}_{new},\ proof)$:验证者根据公共参数和证明$proof$,验证$\boldsymbol{b}_{new}$的真实性,并输出1或0代表接受或拒绝.

4.2 协议流程

仅由证明者发送1次消息构造的非交互式零知识公钥更新协议共包括新公钥的证明和验证两个算法,具体流程分别如算法4和算法5所示.算法中的H_c为随机谕言机,在实际部署时一般用安全的哈希函数替换.

算法4. 公钥更新证明Prove

输入:$(s,\ \boldsymbol{a}_{old},\ \boldsymbol{b}_{old},\ \boldsymbol{e}_{old},\ \boldsymbol{a}_{new},\ \boldsymbol{b}_{new},\ \boldsymbol{e}_{new})$

// 私钥,两公共参数,两公钥,两误差多项式

输出:$(c,(z_s,\ \boldsymbol{z}_{e\_old},\ \boldsymbol{z}_{e\_new}))$

// 挑战,响应

1. 均匀选取临时私钥$\bar{s}\leftarrow\mathbb{Z}_p[X]/(X^N+1)$和临时误差$\overline{\boldsymbol{e}_{old}},\ \overline{\boldsymbol{e}_{new}}\in\mathcal{E}$
2. 计算公共参数取值a_{old}下的临时公钥
   $\overline{\boldsymbol{b}_{old}}=\boldsymbol{a}_{old}\bar{s}+\overline{\boldsymbol{e}_{old}}$和$\overline{\boldsymbol{b}_{new}}=\boldsymbol{a}_{new}\bar{s}+\overline{\boldsymbol{e}_{new}}$
3. 计算挑战c,
   $c=H_c(\boldsymbol{a}_{old},\ \boldsymbol{b}_{old},\ \overline{\boldsymbol{b}_{old}},\ \boldsymbol{a}_{new},\ \boldsymbol{b}_{new},\ \overline{\boldsymbol{b}_{new}})$

4 计算响应$z_s=sc+\bar{s}$,

$$z_{e\_old}=\boldsymbol{e}_{old}c+\overline{\boldsymbol{e}_{old}},\ z_{e\_new}=\boldsymbol{e}_{new}c+\overline{\boldsymbol{e}_{new}}$$

算法5.　公钥更新验证Verify

输入:$(\boldsymbol{a}_{old},\ \boldsymbol{b}_{old},\ \boldsymbol{a}_{new},\ \boldsymbol{b}_{new},\ (c,(z_s,\ z_{e_{old}},\ z_{e_{new}})))$

// 两公共参数,两公钥,证明

输出:$isVerified$

// 是否验证通过的布尔值

1. 验证$z_{e_old},\ z_{e_new}$的范围:

$$\|z_{e\_old}\|_\infty,\ \|z_{e\_new}\|_\infty<(\kappa+1)N$$

2 验证$c=H_c(\boldsymbol{a}_{old},\ \boldsymbol{b}_{old},\ \boldsymbol{a}_{old}z_s-\boldsymbol{b}_{old}c+z_{e_{old}},\ \boldsymbol{a}_{new},\ \boldsymbol{b}_{new},\boldsymbol{a}_{new}z_s-\boldsymbol{b}_{new}c+z_{new})$

3 $isVerified\leftarrow\{0,1\}$

//验证通过接受证明输出1,否则输出0

特别的,对该方案进行实例化时,还需要确定各误差多项式向量的系数服从的分布χ. 这里χ取均值$\mu=0$,标准差$\sigma=N^{3/4}$的高斯分布. 根据Benhamouda等人[43]和Lyubashevsky等人[48]的分析,该挑战-响应的知识误差为$1/\left\langle\begin{matrix}N/2\\ \kappa\end{matrix}\right\rangle$,当$N=256$,$\kappa=32$时误差率小于$2^{-100}$. $N\geqslant 2^8$时有$\Pr[|\chi(e)|<N]=1-\mathrm{negl}(\lambda)$. 因此,可以将$z_{e_old}$和$z_{e_new}$的取值范围验证阈值设置为:

$$\|z_{e\_old}\|_\infty,\ \|z_{e\_new}\|_\infty<(\kappa+1)N$$

4.3　协议分析

4.3.1　协议安全性分析

根据4.1节提出的公钥更新协议构造范式,4.2节的非交互式RLWE公钥更新协议中,Prove算法的第二步实为RLWE标准承诺. 在证明本协议是零知识公钥更新协议前,需要首先证明该步骤针对安全参数$\lambda=256$的安全需求满足计算绑定性.

引理1.　取$\lambda=256$,RLWE下的二元关系:

$$\mathcal{R}=\{((\boldsymbol{a},\ \boldsymbol{b}),(s,\ \boldsymbol{e})):\ \boldsymbol{b}=\boldsymbol{a}s+\boldsymbol{e}\wedge\|\boldsymbol{e}\|_\infty<N)\}$$

在表2的参数选取下满足RLWE标准承诺的计算绑定性.

表2　适用于RLWEPkUpdate的RLWE推荐参数

符号	含义	取值
N	多项式环中的多项式次数	256
K	RLWE假设中的向量长度	3
p	多项式环的系数所模素数	64位强素数 $p\equiv 3\ mod\ 8$
σ	RLWE噪声分布的标准差	$N^{3/4}$

证明. 我们证明取$N=256$时能够满足$\lambda=256$的安全需求. 根据N和p的参数选取,多项式环$\mathbb{Z}_p[X]/(X^N+1)$的模多项式(X^N+1)有$(X^N+1)=\alpha(X)\cdot\beta(X)$,且$\alpha(X)$和$\beta(X)$为不可约多项式[49],因此第$i$个多项式乘积$a_is$有至少$p^{N/2}$种取值,故$\boldsymbol{b}=\boldsymbol{a}s+\boldsymbol{e}$有至少$p^{KN/2}$种取值.

使用反证法,假设$((\boldsymbol{a},\ \boldsymbol{b}),(s_1,\ \boldsymbol{e}_1)),((\boldsymbol{a},\ \boldsymbol{b}),(s_2,\ \boldsymbol{e}_2))\in\mathcal{R}$,即:

$$\boldsymbol{b}=\boldsymbol{a}s_1+\boldsymbol{e}_1\wedge\boldsymbol{b}=\boldsymbol{a}s_2+\boldsymbol{e}_2\wedge$$
$$\|\boldsymbol{e}_1\|_\infty<N\wedge\|\boldsymbol{e}_2\|_\infty<N\wedge s_1\neq s_2$$

对$\boldsymbol{b}=\boldsymbol{a}s_1+\boldsymbol{e}_1$和$\boldsymbol{b}=\boldsymbol{a}s_2+\boldsymbol{e}_2$作差,有$\boldsymbol{a}\Delta s+\Delta\boldsymbol{e}=0$,其中$\Delta s=s_1-s_2$,$\Delta\boldsymbol{e}=\boldsymbol{e}_1-\boldsymbol{e}_2$,分析可知$\Delta s$最多有$p^N$种取值;考虑到$\sigma=N^{3/4}$,$\Delta\boldsymbol{e}$最多有$(4N)^{KN}$种取值($|e_{ij}|<2N$).

考虑碰撞集合$\Delta s\times\Delta\boldsymbol{e}$和$\boldsymbol{b}$取值集合的元素大小,并代入$p$, N, K的取值,有:

$$\frac{p^N(4N)^{KN}}{p^{KN/2}}=\left(\frac{2^{64}2^{30}}{2^{96}}\right)^N=\frac{1}{4^N}=2^{-2\lambda}\tag{1}$$

因此,产生碰撞的概率为可忽略的,即有:

$$\Pr\left[\begin{pmatrix}a_1\Delta s\\ \vdots\\ a_K\Delta s\end{pmatrix}=\begin{pmatrix}\Delta e_1\\ \vdots\\ \Delta e_K\end{pmatrix}:\ a,\ b\leftarrow\left(\frac{\mathbb{Z}_p[X]}{X^N+1}\right)^K\right]\leqslant\mathrm{negl}(N)\tag{2}$$

证毕.

下面证明4.2节的非交互式RLWE公钥更新协议是零知识公钥更新协议.

定理1.　若H_c为随机谕言机,则4.2节的非交互式RLWE公钥更新协议是零知识公钥更新协议.

证明. 即证本协议满足正确性、特殊可靠性、诚实验证者零知识性.

正确性:对于一个诚实的证明者,在4.2节给出的挑战空间L1范数约束κ和标准差σ选取下,有$1-\mathrm{negl}(\lambda)$的概率正确计算响应:

$$\begin{aligned}\overline{\boldsymbol{b}_{old}}&=c(\boldsymbol{b}_{old}-\boldsymbol{b}_{old})+\overline{\boldsymbol{b}_{old}}\\&=(sc\boldsymbol{a}_{old}-sc\boldsymbol{a}_{old})+(c\boldsymbol{e}_{old}-c\boldsymbol{e}_{old})+(\boldsymbol{a}_{old}\bar{s}+\overline{\boldsymbol{e}_{old}})\\&=(sc+\bar{s})\boldsymbol{a}_{old}-c(\boldsymbol{a}_{old}s+\boldsymbol{e}_{old})+\left(c\boldsymbol{e}_{old}+\overline{\boldsymbol{e}_{old}}\right)\\&=z_s\boldsymbol{a}_{old}-c\boldsymbol{b}_{old}+z_{e_{old}}\end{aligned}\tag{3}$$

同理$\overline{\boldsymbol{b}_{old}}=z_s\boldsymbol{a}_{new}-c\boldsymbol{b}_{new}+z_{e_new}$. 因此:

$$H_c\left(\boldsymbol{a}_{old},\ \boldsymbol{b}_{old},\ \overline{\boldsymbol{b}_{old}},\ \boldsymbol{a}_{new},\ \boldsymbol{b}_{new},\ \overline{\boldsymbol{b}_{new}}\right)=H_c\begin{pmatrix}\boldsymbol{a}_{old},\ \boldsymbol{b}_{old},\ z_s\boldsymbol{a}_{old}-c\boldsymbol{b}_{old}+z_{e\_old},\\ \boldsymbol{a}_{new},\ \boldsymbol{b}_{new},z_s\boldsymbol{a}_{new}-c\boldsymbol{b}_{new}+z_{e\_new}\end{pmatrix}\tag{4}$$

特殊可靠性:考虑二元关系$\mathcal{R}_1$和$\mathcal{R}_2$,其中:

$$\mathcal{R}_1=\{((\boldsymbol{a},\ \boldsymbol{b}),(s,\ \boldsymbol{e})):\ \boldsymbol{b}=\boldsymbol{a}s+\boldsymbol{e}\wedge\left|\left|\boldsymbol{e}\right|\right|_\infty<N\}\}$$

$$\mathcal{R}_2=\{((\boldsymbol{a},\ \boldsymbol{b}),(s,\ \boldsymbol{e},\ c)):\ \boldsymbol{b}=\boldsymbol{a}s+c^{-1}\boldsymbol{e}\wedge\|\boldsymbol{e}\|_\infty<2N\wedge c\in\mathcal{C})\}$$

存在PPT算法A_E,输入2个被接受的证明所产生的三元组

$(\overline{\boldsymbol{b}_{old}}_1, c_1, (z_{s1}, \boldsymbol{z}_{e\_old1}))$, $(\overline{\boldsymbol{b}_{old}}_2, c_2, (z_{s2}, \boldsymbol{z}_{e\_old2}))$

可由$\overline{\boldsymbol{b}_{old}}=z_s\boldsymbol{a}_{old}-c\boldsymbol{b}_{old}+\boldsymbol{z}_{e_old}$计算

$$\overline{\boldsymbol{b}_{old}}_1-\overline{\boldsymbol{b}_{old}}_2=(z_{s1}-z_{s2})\boldsymbol{a}_{old}-(c_1-c_2)\boldsymbol{b}_{old}+(\boldsymbol{z}_{e\_old1}-\boldsymbol{z}_{e\_old2})$$

由RLWE承诺绑定性有$\overline{\boldsymbol{b}_{old}}_1=\overline{\boldsymbol{b}_{old}}_2$,因此

$$(c_1-c_2)\boldsymbol{b}_{old}=(z_{s1}-z_{s2})\boldsymbol{a}_{old}+(\boldsymbol{z}_{e\_old1}-\boldsymbol{z}_{e\_old2})$$

记:

$$\Delta c=c_1-c_2,\ \Delta z=z_{s1}-z_{s2},\ \boldsymbol{\Delta e}=\boldsymbol{z}_{e\_old1}-\boldsymbol{z}_{e\_old2}$$

由Δc在$\mathbb{Z}_p[X]/(X^n+1)$上有逆[48],有

$$\boldsymbol{b}_{old}=\frac{\Delta z}{\Delta c}\boldsymbol{a}_{old}+\frac{1}{\Delta c}\boldsymbol{\Delta e}$$

PPT算法A_E输出$w'=(\frac{\Delta z}{\Delta c}, \frac{1}{\Delta c}\boldsymbol{\Delta e}, \Delta c)$满足

$$((\boldsymbol{a}_{old}, \boldsymbol{b}_{old}), (\frac{\Delta z}{\Delta c}, \frac{1}{\Delta c}\boldsymbol{\Delta e}, \Delta c))\in\mathcal{R}_2$$

选用另一对三元组$(\overline{\boldsymbol{b}_{new}}_1, c_1, (z_{s1}, \boldsymbol{z}_{e_new1}))$, $(\overline{\boldsymbol{b}_{new}}_2, c_2, (z_{s2}, \boldsymbol{z}_{e_new2}))$同理.

诚实验证者零知识性:考虑二元关系

$$\mathcal{R}=\{(((\boldsymbol{a}_1, \boldsymbol{b}_1), (\boldsymbol{a}_2, \boldsymbol{b}_2)), ((s, \boldsymbol{e}_1), (s, \boldsymbol{e}_2))):\boldsymbol{b}_1=\boldsymbol{a}_1s+\boldsymbol{e}_1\wedge\boldsymbol{b}_2=\boldsymbol{a}_2s+\boldsymbol{e}_2\wedge\|\boldsymbol{e}_1\|_\infty<N\wedge\|\boldsymbol{e}_2\|_\infty<N\}$$

根据定义2,构造PPT算法A_S作为模拟器,A_S将陈述$statement=((\boldsymbol{a}_{old}, \boldsymbol{b}_{old}), (\boldsymbol{a}_{new}, \boldsymbol{b}_{new}))$和挑战$c$作为输入,并计算

$$\overline{\boldsymbol{b}_{old}}'=z_s'\boldsymbol{a}_{old}-c\boldsymbol{b}_{old}+\boldsymbol{z}_{e\_old}'$$

$$\overline{\boldsymbol{b}_{new}}'=z_s'\boldsymbol{a}_{new}-c\boldsymbol{b}_{new}+\boldsymbol{z}_{e\_new}'$$

其中$z_s'\in\mathbb{Z}_p[X]/(X^N+1)$, $\boldsymbol{z}_{e_old}', \boldsymbol{z}_{e_new}'\in(\mathbb{Z}_p[X]/(X^N+1))^K$且$\|\boldsymbol{z}_{e_old}'\|_\infty$, $\|\boldsymbol{z}_{e_new}'\|_\infty<(\kappa+1)N$为随机生成的三个元素.$A_S$输出三元组

$$((\overline{\boldsymbol{b}_{old}}', \overline{\boldsymbol{b}_{new}}'), c, (z_s', \boldsymbol{z}_{e\_old}', \boldsymbol{z}_{e\_new}'))$$

该三元组的分布与诚实验证者真实运行协议的过程中所产生的对应值是计算不可区分的.

因此,4.2节的非交互式RLWE公钥更新协议是零知识公钥更新协议.

证毕.

4.3.2 协议复杂度分析

根据协议流程,本协议的计算开销主要集中在计算临时公钥的多项式向量数乘部分,即常数次的数乘运算$\boldsymbol{P}\cdot\boldsymbol{Q}$, $\boldsymbol{P}\in\mathbb{Z}_p[X]/(X^N+1)$, $\boldsymbol{Q}\in(\mathbb{Z}_p[X]/(X^N+1))^K$.通过离散傅里叶变换(Discrete Fourier Transform,DFT)加速多项式乘法,可以使本运算的时间复杂度下降至$O(KN\log N)$.在实际应用中,K取值相对于N较小,且向量数乘可以并行运算,因此在考虑计算的渐进复杂度时可以忽略.而在通信中,仅有$\boldsymbol{z}_{e_old}$和$\boldsymbol{z}_{e_new}$为多项式向量,且其系数取值为圆整后的高斯分布$\chi(e)$,因此通信复杂度也可忽略K的影响.

根据上述分析,在时间开销上:本协议的证明复杂度为$O(N\log N)$,验证复杂度为$O(N\log N)$.在空间开销上:本协议的通信复杂度为$O(N)$,比特复杂度为$O(KN^2+len(p)N)$.

5 基于RLWE的异步分布式随机信标

5.1 协议流程

如图1所示,本文所设计的基于RLWE的异步分布式随机信标协议基于PVSS结构,共分为4个步骤,分别是密钥生成、份额揭示、分享验证、聚合输出.

由于PVSS中的秘密分发和重组使用Shamir秘密分享,该方案不能够直接处理多项式环上的元素,因此本文使用了定义6中给出的CKKS全同态映射,以将实数向量与多项式环上元素互相转换.

基于RLWE的异步分布式随机信标协议ADRB_{RLWE}共包含算法6至算法9四个子算法,其中密钥生成由可信第三发运行,用于分发参与者P_i的公私钥对.后三个阶段由参与者P_i运行,份额揭示用于计算单次随机信标的份额和对该份额的证明,分享验证用于验证其它节点所发送的随机信标份额的正确性,聚合输出用于通过真实的随机信标份额集合计算随机信标值.本协议将向量到多项式环的全同态映射作为黑盒算法使用,且输入更改为整数上的向量,读者可理解为CKKS同态映射算法.

算法6. ADRB_{RLWE}密钥生成Gen

输入:(n, k, t)

// 参与节点总数n,门限值k,敌手数目t

输出:$(\{f_i(X)\}, \{\boldsymbol{b}_i(\boldsymbol{X})\})$

// n个参与者的公私钥对,秘密分发$f_i(X)$,广播$\boldsymbol{b}_i(\boldsymbol{X})$

1. 均匀选取$k-1$次多项式向量$\boldsymbol{f}(\boldsymbol{X})=\sum_{j=0}^{k-1}\begin{pmatrix}m_{1j}\\ \vdots\\ m_{N'j}\end{pmatrix}X^j(\boldsymbol{f}(\boldsymbol{X})\in\mathcal{F}, \mathcal{F}=\left\{d|d\in\{0,\cdots,(p-1)\}^{\frac{kN}{2}}\right\})$

2. 计算参与方P_i的私钥$f_i(X)=\mathrm{Encode}_M(f(i))$和随机选取的误差多项式$\boldsymbol{e}_i(\boldsymbol{X})$

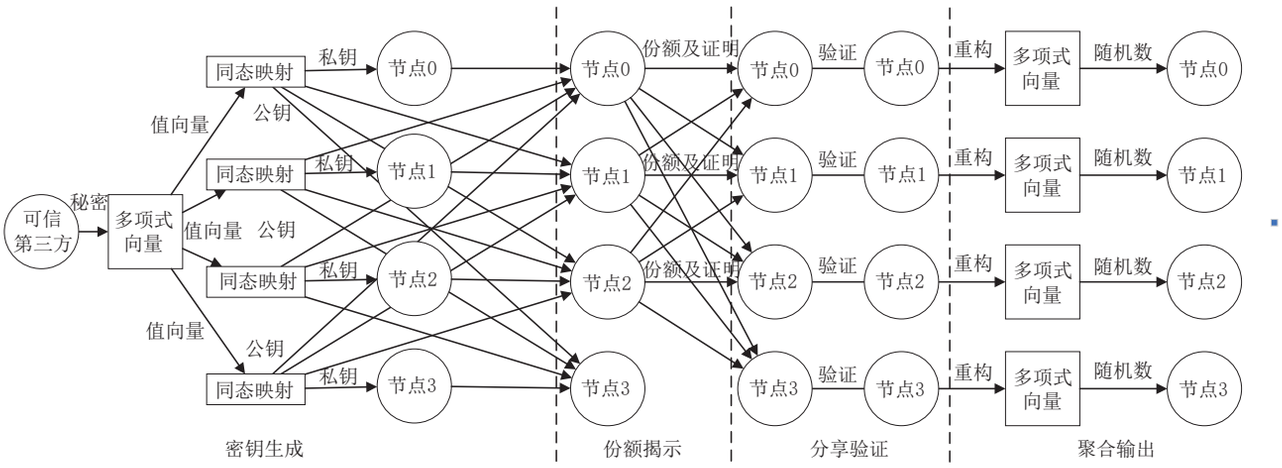

图1　基于RLWE的异步分布式随机信标协议总览

3. 在$(\mathbb{Z}_p[X]/(X^N+1))^K$上均匀选取公共参数$\boldsymbol{a}(\boldsymbol{X})$，计算$P_i$的公钥$\boldsymbol{b}_i(\boldsymbol{X})=\boldsymbol{a}(\boldsymbol{X})f_i(X)+\boldsymbol{e}_i(\boldsymbol{X})$

算法7.　ADRB_{RLWE}份额揭示 Reveal

输入：$(seed, f_i(X))$

// 全局已知的种子，参与方P_i自身的私钥

输出：$(\overline{\boldsymbol{b}(\boldsymbol{X})}, proof)$

// 份额、对份额的证明，在网络内广播该元组

1. 借助谕言机选取抛币公共参数$\overline{\boldsymbol{a}(\boldsymbol{X})}=H_b(seed)$

2. 选取对应的抛币误差多项式$\overline{\boldsymbol{e}_i(\boldsymbol{X})}$计算硬币份额$\overline{\boldsymbol{b}(\boldsymbol{X})}=\overline{\boldsymbol{a}(\boldsymbol{X})}f_i(X)+\overline{\boldsymbol{e}_i(\boldsymbol{X})}$

3. 调用公钥更新生成证明：

$$proof=\mathrm{Prove}\Big(f_i(X),\ \boldsymbol{a}(\boldsymbol{X}),\ \boldsymbol{b}(\boldsymbol{X}),\ \boldsymbol{e}_i(\boldsymbol{X}),\ \overline{\boldsymbol{a}(\boldsymbol{X})},\ \overline{\boldsymbol{b}(\boldsymbol{X})},\ \overline{\boldsymbol{e}_i(\boldsymbol{X})}\Big)$$

算法8.　ADRB_{RLWE}分享验证 Sharevrfy

输入：$(seed, (\overline{\boldsymbol{b}_j(\boldsymbol{X})}, proof_j))$

// 全局已知的种子，份额及证明

输出：$isVerified$

// 若验证通过，则将键值对$(j, \overline{\boldsymbol{b}_j(\boldsymbol{X})})$加入到集合$S$中

1. 借助谕言机选取抛币公共参数$\overline{\boldsymbol{a}(\boldsymbol{X})}=H_b(seed)$

2. 调用公钥更新进行验证：

$$isVerified=\mathrm{Verify}(\boldsymbol{a}(\boldsymbol{X}),\boldsymbol{b}_j(\boldsymbol{X}),\overline{\boldsymbol{a}(\boldsymbol{X})},\overline{\boldsymbol{b}_j(\boldsymbol{X})},proof_j)$$

算法9.　ADRB_{RLWE}聚合输出 ShareCombine

输入：(S, MSB)

// 通过集合的子集，高位取值长度

输出：$seed_g$

// 随机信标

1. 选取集合S中的k个元素$\Big(\alpha, \overline{\boldsymbol{b}_\alpha(\boldsymbol{X})}\Big)$记为$S'$

2. 计算随机信标值：

$$seed_g=H_a\Big(\sum\nolimits_{\alpha\in S}\overline{\boldsymbol{b}_\alpha(\boldsymbol{X})}(n!\prod\nolimits_{\beta=S-\{\alpha\}}(\beta/(\beta-\alpha))),MSB\Big)$$

概述地讲，可信第三方根据总参与节点数n，生成随机信标所需份额k，最大容忍的敌手数目t（要求$3t+1\leqslant n$且$k\leqslant n-t$，一般取$k=n-t$）生成秘密多项式并在多项式上取值以分发公私钥；当所有节点要根据全局已知的种子$seed$生成随机信标时，他们通过随机谕言机H_b分别将$seed$映射为公共参数$\overline{a(X)}$，并调用第4节的公钥更新协议生成证明$proof$，只有收到门限k个验证通过的份额时才会聚合并通过随机谕言机H_a（取输入值的前MSB位）生成本次随机信标$seed_g$，后可通过安全的随机数生成器 RG 拓展为任意长的随机数 RG($seed_g$).

5.2　协议分析

5.2.1　协议安全性分析

如果敌手只有可忽略的概率伪造份额，且只有可忽略的概率在份额数量不足门限k时恢复随机信标，则本协议是一个安全的异步分布式随机信标协议. 非形式化的，由于异步网络内只有t个敌手，因此当所有的诚实节点被激活时，网络中一定至少有$n-t$个正确的份额处于或完成了传输阶段，各节点不会永远等待下去；且敌手难以伪造份额，因此各节点获得的随机信标是一致的. 相比于通信各方直接使用全局已知的$seed$作为H_a的输入或 RG 的种子，分布式协议强调了当网络中少于$k-t$诚实节点广播份额时，敌手不能够获得最终的随机信标，而前者在协议设计时可轻易转化为让所有诚实节点均广播份额（在本文的下一节将给出应用示例），从而永远保障不会出现敌手获得随机信标且网络中所有相关消息传递完毕时存在诚实节点未获得随机信标的情况出现.

下面证明5.1节的协议构造在使用的从$\mathbb{Z}^{N/2}$到$\mathbb{Z}[X]/X^N+1$的同态映射和证明算法 Prove(·)、验

证函数 Verify(·)理想的情况下满足安全需求.

定理2. 若$\mathrm{Encode}_{2N}(\cdot):\mathbb{Z}^{N/2}\to\mathbb{Z}[X]/X^N+1$为理想的同态映射编码,且满足数乘不变性即$a\mathrm{Encode}_{2N}(b)=\mathrm{Encode}_{2N}(ab)$;Prove(·)和 Verify(·)为理想的非交互式零知识证明方案,则5.1节提出的$\mathrm{ADRB}_{RLWE}=\{\mathrm{Gen},\ \mathrm{Reveal},\ \mathrm{ShareVrfy},\ \mathrm{ShareCombine}\}$是异步分布式随机信标协议.

证明. 本方案仅涉及1轮n对n广播,且监听门限$k\leqslant n-t$,因此在异步网络下的可终止性天然得到满足.下证本协议满足正确性、鲁棒性、不可预测性.

$$\mathrm{Encode}_M(f(0))=\mathrm{Encode}_M(\Sigma_{\alpha\in S}(\boldsymbol{f}(\boldsymbol{\alpha})(\Pi_{\beta\in S-\{\alpha\}}\frac{\beta}{\beta-\alpha})))=\Sigma_{\alpha\in S}(f_\alpha(X)(\Pi_{\beta\in S-\{\alpha\}}\frac{\beta}{\beta-\alpha})) \tag{5}$$

注意到$\mathrm{Encode}_M(f(0))\in\mathbb{Z}[X]/X^N+1$,将上式乘以多项式向量$n!\cdot\overline{\boldsymbol{a}(\boldsymbol{X})}$,有:

$$n!\cdot\overline{\boldsymbol{a}(\boldsymbol{X})}\Sigma_{\alpha\in S}(f_\alpha(X)(\Pi_{\beta\in S-\{\alpha\}}\frac{\beta}{\beta-\alpha}))=\Sigma_{\alpha\in S}(\overline{\boldsymbol{a}(\boldsymbol{X})}f_\alpha(X)(n!\ \Pi_{\beta\in S-\{\alpha\}}\frac{\beta}{\beta-\alpha})) \tag{6}$$

特别的,在工程实现中$\left(n!\prod_{\beta\in S-\{\alpha\}}\left(\beta/(\beta-\alpha)\right)\right)$的运算顺序为$n!\prod_{\beta\in S-\{\alpha\}}(1/(\beta-\alpha))\prod_{\beta\in S-\{\alpha\}}(\beta)$,以确保$n!\prod_{\beta\in S-\{\alpha\}}(\beta/(\beta-\alpha))$的值永远为整数.

考虑各节点计算所得值,有:

$$\Sigma_{\alpha\in S}(\overline{\boldsymbol{b}_\alpha(\boldsymbol{X})}(n!\ \Pi_{\beta\in S-\{\alpha\}}\frac{\beta}{\beta-\alpha}))=\Sigma_{\alpha\in S}((\overline{\boldsymbol{a}(\boldsymbol{x})}f_\alpha(X)+\overline{\boldsymbol{e}_\alpha(\boldsymbol{X})})(n!\ \Pi_{\beta\in S-\{\alpha\}}\frac{\beta}{\beta-\alpha}))=n!\cdot\overline{\boldsymbol{a}(\boldsymbol{X})}\mathrm{Encode}_M(f(0))+\Sigma_{\alpha\in S}(\overline{\boldsymbol{e}_\alpha(\boldsymbol{X})}(n!\ \Pi_{\beta\in S-\{\alpha\}}\frac{\beta}{\beta-\alpha})) \tag{7}$$

可知,若随机信标的噪声多项式向量$\sum_{\alpha\in S}(\overline{\boldsymbol{e}_\alpha(\boldsymbol{X})}(n!\prod_{\beta\in S-\{\alpha\}}(\beta/(\beta-\alpha))))$各系数的最高$MSB'$位均为0,则对于任一节点$P_i$,所计算出的

$$seed_g=H_a(\Sigma_{\alpha\in S}(\overline{\boldsymbol{b}_\alpha(\boldsymbol{X})}(n!\ \Pi_{\beta\in S-\{\alpha\}}\frac{\beta}{\beta-\alpha})),MSB)$$

与无误差的参考值

$$H_a\left(n!\cdot\overline{\boldsymbol{a}(\boldsymbol{X})}\mathrm{Encode}_M(f(0)),\ MSB\right)$$

不等的概率为$2^{-\Delta MSB}$,其中$\Delta MSB=MSB'-MSB+1$.k个诚实节点所生成的随机信标均相同的概率为$(1-2^{-\Delta MSB})^k$.

考虑$\sum_{\alpha\in S}(\overline{\boldsymbol{e}_\alpha(\boldsymbol{X})}(n!\prod_{\beta\in S-\{\alpha\}}(\beta/(\beta-\alpha))))$的最大值.$\prod_{\beta\in S-\{\alpha\}}(\beta-\alpha)$的最小值为$((k/2)!)^2$,在集合$S$中的元素在整数环上相邻且$\alpha_i$为中位数时取到;$\prod_{\beta\in S-\{\alpha\}}\beta$的最大值为$n!/(t+1)!$,在集合$S$中的元素为所有参与节点的$k$个最大编号且$\alpha$为$S$中的最小数值时取到.两限定条件无交集,故$\prod_{\beta\in S-\{\alpha\}}(\beta/(\alpha-\beta))<(n!/(t+1)!)/((k/2)!)^2$.取$n=10$, $\mathrm{len}(p)=64$,$\prod_{\beta\in S-\{\alpha\}}(\beta/(\alpha-\beta))<2^9$,$\sum_{\alpha\in S}(\overline{\boldsymbol{e}_\alpha(\boldsymbol{X})}(n!\prod_{\beta\in S-\{\alpha\}}(\beta/(\beta-\alpha))))<2^{39}$,则$MSB'>25$.取$MSB=1$,存在诚实节点未恢复出随机信标进而可能会造成协议运行更多轮数的概率小于2^{-18},满足一般场景的需要.实际应用中也可通过扩展模数p的长度达到任意小的出错概率.

鲁棒性:由于本部分假设 Prove(·)和 Verify(·)为理想的非交互式零知识证明方案,因此天然满足敌手仅有可忽略的概率伪造出k个可以通过验证的$(\overline{pk}_i,\ \overline{vk}_i)$.在第4章,我们证明了所提出的非交互式RLWE公钥更新协议是零知识公钥更新协议,为计算安全的非交互式零知识证明方案,因此将该理想模型替换为此方案后仍然满足敌手仅有可忽略的概率伪造出任意一个可以通过验证的$(\overline{pk}_i,\ \overline{vk}_i)$.

不可预测性:首先考虑$H_a\left(\sum_{\alpha\in S}\overline{\boldsymbol{b}_\alpha(\boldsymbol{X})},\ (n!\prod_{\beta\in S-\{\alpha\}}\left(\beta/(\beta-\alpha)\right)MSB\right)$的集合大小.由于$\overline{b_\alpha(X)}$是伪随机的,因此$s=\sum_{\alpha\in S}\overline{\boldsymbol{b}_\alpha(\boldsymbol{X})}(n!\prod_{\beta\in S-\{\alpha\}}(\beta/(\beta-\alpha)))$也是伪随机的,则谕言机$H_a$的输出集合大小为$(2^{MSB})^{KN}$,$\Pr[M=m]=1/(2^{MSB})^{KN}$.

接下来考虑$\Pr[M=m|C=c]$:假设敌手获得了$k-1$个份额,将某一分量分别为$(\alpha_1,\ \overline{b_{\alpha_1}(X)}),\cdots,(\alpha_{k-1},\ \overline{b_{\alpha_{k-1}}(X)})$,设$\alpha_k\in\mathbb{Z}_q^*$,记集合$S=\{\alpha_1,\cdots,\alpha_k\}$,敌手根据拉格朗日插值计算:

$$s=\Sigma_{l=1}^{k-1}\left(\overline{b_{\alpha_l}(X)}n!\ \Pi_{\beta\in S-\{\alpha_l\}}\left(\beta/(\beta-\alpha_l)\right)\right)+\overline{b_{\alpha_k}(X)}(n!\ \Pi_{\beta\in S-\{\alpha_k\}}\left(\beta/(\beta-\alpha_k)\right)) \tag{8}$$

记:

$$u=n!\ \Pi_{\beta\in S-\{\alpha_k\}}\left(\beta/(\beta-\alpha_k)\right)$$

$$v=\Sigma_{l=1}^{k-1}\left(\overline{b_{a_l}(X)}n!\ \Pi_{\beta\in S-\{a_l\}}\left(\beta/(\beta-\alpha_l)\right)\right)$$

$$z=\overline{b_{a_k}(X)} \tag{9}$$

则有$s(z)=zu+v$. 由于$u\in\mathbb{Z}_q^*$,故z与s的取值集合形成一一映射. 在第4章已经证明了：

$$\Pr\left[\begin{pmatrix}a_1\Delta s\\ \vdots\\ a_K\Delta s\end{pmatrix}=\begin{pmatrix}\Delta e_1\\ \vdots\\ \Delta e_K\end{pmatrix}:a,\ b\leftarrow\left(\frac{\mathbb{Z}_p[X]}{X^N+1}\right)^K\right]\leqslant \mathrm{negl}(N) \tag{10}$$

因此,z的取值集合可近似为$f_{a_k}(X)$的取值集合大小,即q^N. 由$\overline{b_{a_k}(X)}$的伪随机性：

$$\Pr[M=m|C=c]=\max\left(1/q^N+\mathrm{negl}(\lambda),\ 1/\left(2^{MSB}\right)^{KN}\right) \tag{11}$$

由$N=\lambda$,知$\Pr[M=m]=\mathrm{negl}(\lambda)$, $\Pr[M=m|C=c]=\mathrm{negl}(\lambda)$, 即有即$|\Pr[M=m|C=c]-\Pr[M=m]|=\mathrm{negl}(\lambda)$,本方案是计算安全的,敌手掌握小于$k$个可通过验证的份额时仅有可忽略的概率输出有效的随机信标$seed_g$.

因此,$\mathrm{ADRB}_{RLWE}=\{\mathrm{Gen},\mathrm{Reveal},\mathrm{ShareVrfy},\mathrm{ShareCombine}\}$是异步分布式随机信标协议. 证毕.

5.2.2　协议复杂度分析

根据协议流程,分析初始化和生成一次随机信标的开销. 初始化阶段,主要需要进行n次同态映射运算和n次公钥计算,以CKKS同态映射作为实例化的单次编码开销为$O(N^2)$,单次公钥计算的开销为$O(N\log N)$,因此,初始化阶段的计算开销为$O(N^2n)$,如果实现对n个实例并行计算,则可将计算开销降低为$O(N^2)$. 本阶段在通信开销中主要需要$n+1$次1对n广播用于广播公共参数和公钥,故消息复杂度为$O(n^2)$,通信复杂度为$O\left(\mathrm{len}(p)\cdot KN\cdot n^2\right)$比特.

下面考虑生成单个随机信标的开销. 对于任意的诚实节点P_i,在Reveal阶段主要需要进行一次公钥运算,一次公钥更新证明,计算复杂度为$O(N\log N)$;在ShareVrfy阶段需要进行k次公钥更新验证,在$k=n-t$的默认输入下计算复杂度为$O(n\cdot N\log N)$;在ShareCombine阶段主要需要进行k次多项式向量加法、k^2次$\mathbb{Z}_p$上的求逆运算,考虑异步共识协议的节点数n相对于N较小,故计算复杂度为$O(n\cdot N)$. 综上所述,单个节点生成一个随机信标的计算开销是$O(n\cdot N\log N)$. 生成单个随机信标需要进行1轮n对n广播,故消息复杂度为$O(n^2)$. 考虑到广播内容$\left(\overline{b(X)},\ proof\right)$中$proof$的尺寸较小,故比特复杂度为$O(\mathrm{len}(p)\cdot KN\cdot n^2)$.

因此,初始化和生成单个随机信标的消息复杂度均为$O(n^2)$,通信复杂度均为$O\left(\mathrm{len}(p)\cdot KN\cdot n^2\right)$比特.

6　采用ADRB_{RLWE}抛币的异步共识协议

6.1　协议流程

本章以最新提出的异步共识协议WaterBear[18]中使用的异步二元共识协议Quadratic－ABA为例,给出ADRB_{RLWE}在异步二元共识中的应用,首先给出异步二元共识协议的定义.

定义10.　异步二元共识协议. 异步二元共识协议旨在使系统中所有参与节点对某个二元值达成一致,是实现异步共识协议的基础. 在该协议中,每个参与节点在协议执行前提出一个初始二元值v,协议执行结束后每个节点会决定同一个二元值b. 异步二元共识协议需要满足以下性质：

(1)有效性：如果所有诚实节点输入的值均为b,那么每个诚实节点都会决定b.

(2)一致性：若任意一个诚实节点决定一个比特值b,那么每个诚实节点都会决定b.

(3)可终止性：如果所有诚实节点都能收到输入值,那么每个诚实节点都能决定一个比特b.

(4)整体性：诚实节点不会多次决定.

Quadratic－ABA的消息复杂度为$O(n^2)$,该协议没有使用密码学假设,在使用本地抛币下是天然抗量子的,它可将本地抛币直接替换为异步分布式抛币,以实现常数轮终止的公共硬币异步二元共识协议CC－ABA. 遗憾的是,原文并未给出能够满足抗量子安全的异步分布式抛币协议构造.

本文利用哈希函数H_a将所得随机信标映射为二元值,将所设计的基于RLWE的异步分布式随机信标协议应用到协议中,构造了新的异步二元共识协议CC_{RLWE}－ABA,以作为CC－ABA的实例在抗量子安全的保障下使协议实现常数轮终止,协议流程如过程1和过程2所示.

过程1.　CC_{RLWE}－ABA初始化Format

1. 可信第三方调用ADRB_{RLWE}的$\mathrm{Gen}(n,n-t,t)$,进行一次性可信初始化后退出协议.

2. 定义有效投票：

若为二值票$vote_x(b, r)$,则有效定义为事先收到$t+1$张$vote_{x-1}(b, r)$;

若为弃权票$vote_x(\perp, r)$,则有效定义为本地预投票集合$Set_r==\{0, 1\}$.

3 根据节点P_i的初值$v\in\{0, 1\}$和当前交易号ID调用过程2:RoundProtocol(ID, 0, v).

过程2. CC_{RLWE}－ABA轮协议RoundProtocol

输入:(ID, r, v)

// 当前交易号,当前轮数,当前轮初值

输出:b

// 诚实节点间达成一致的二元值

1. 广播初始投票值$vote_1(v, r)$.

2. 接收各节点的$vote_1(b, r)$和$vote_1(\neg b, r)$并计数:

若收到$t+1$张$vote_1(b, r)$初始投票且未广播$vote_1(b, r)$,则广播$vote_1(b, r)$.

若收到$2t+1$张$vote_1(b, r)$初始投票,则将其加入预投票集合Set_r.

3. 若Set_r不为空且未进行$vote_2$,则针对Set_r中的第一个元素v,广播$vote_2(v, r)$.

4. 接收$n-t$张$b\in Set_r$的$vote_2$预投票:

若均为$vote_2(b, r)$,则广播主投票$vote_3(b, r)$;

若存在$vote_2(b, r)$和$vote_2(\neg b, r)$,则广播主投票$vote_3(\perp, r)$.

5 接收$n-t$张有效的$vote_3$主投票

若均为$vote_3(b, r)$,则广播最终投票$vote_4(b, r)$;

否则,则广播最终投票$vote_4(\perp, r)$.

6 接收$n-t$张有效的$vote_4$最终投票

若均为$vote_4(b, r)$,**输出**b,第$r+1$轮初始值$v=b$,直接进入下一轮,下一轮结束后终止;

否则,若同时存在$vote_4(\perp, r)$和$vote_4(b, r)$,第$r+1$轮初始值$v=b$.

7. 节点P_i根据私钥$f_i(X)$调用$ADRB_{RLWE}$的Reveal($ID\|r, f_i(X)$)得到$\left(\overline{b(X)}, proof\right)$并广播.

8. 接收$\left(\overline{b_j(X)}, proof_j\right)$并调用$ADRB_{RLWE}$的ShareVrfy($ID\|r$, ($\overline{b_j(X)}$, $proof_j$))验证.

9. 选取$n-t$个验证通过的$\left(\overline{b_j(X)}, proof_j\right)$键值对构成集合$S$.

10. 若第$r+1$轮初始值v为空,$v=$ShareCombine(S, MSB),调用RoundProtocol(ID, $r+1$, v)进入下一轮.

过程1对协议本身进行了初始化,通过分发公私钥建立了轮协议中调用$ADRB_{RLWE}$的先决条件,并在此基础上定义了有效投票的概念,为轮协议的运行做好准备;过程2首先在过程1结束后被调用,各节点先后进行了4轮n对n广播并根据本地所收到的$vote_4$结果决定输出或进入下一轮.特别的,在异步二元共识中,即使一个节点决定输出,它也会再运行一轮协议以确保其它节点能够输出相同值.

6.2 协议分析

6.2.1 协议安全性分析

本协议为轮协议,每一步等待的合法消息数量不高于$n-t$,因此在异步网络下一定能够进行至下一轮.在证明CC_{RLWE}－ABA协议满足有效性、一致性、常数轮可终止性、整体性之前,需要先给出一个中间结论.

引理2. CC_{RLWE}－ABA协议不可能同时收到有效的$vote_4(b, r)$和$vote_4(\neg b, r)$.

证明.使用反证法,对于诚实节点P_i,若$vote_4(b, r)$和$vote_4(\neg b, r)$同时有效,说明同时有诚实节点发送了$vote_3(b, r)$和$vote_3(\neg b, r)$,即网络中同时出现了$(n-t)$张$vote_2(b, r)$和$(n-t)$张$vote_2(\neg b, r)$.考虑诚实节点共有$n-t$个,每个节点仅投1张$vote_2$,拜占庭节点共有t个,每个节点可以同时投$vote_2(b, r)$和$vote_2(\neg b, r)$,故网络中最多有$n+t$张$vote_2$投票.由于$3t+1\leqslant n$,因此$n+t<2(n-t)$,网络中不会同时产生$(n-t)$张$vote_2(b, r)$和$(n-t)$张$vote_2(\neg b, r)$. 证毕.

定理3. CC_{RLWE}－ABA是抗量子安全的异步常数轮二元共识协议.

证明.即证该协议满足有效性、一致性、常数轮可终止性、整体性.

有效性:若所有诚实节点初值均为v,则Set_r的元素个数为1,即v.因此,任意诚实节点P_i不会接收取值相反的$vote_2(\neg v, r)$,也就不会广播$vote_3(\perp, r)$.P_i最多收到由敌手发出的t个$vote_3(\perp, r)$,对于$vote_4(\perp, r)$将不会认证为有效,因此P_i所收到的$(n-t)$张有效的$vote_4$选票一定均为$vote_4(v, r)$,P_i决定值v并输出.

特别的,根据异步二元共识协议有效性的逆否命题,若P_i决定v,一定存在一个诚实节点P_j第0轮的初值为v.

一致性:如果存在一个诚实节点P_i决定值v并输出,则P_i收到了$(n-t)$张$vote_4(v, r)$,说明至少有$(n-2t)$个诚实节点广播了$vote_4(v, r)$.对于剩余的$(n-t)-(n-2t)$不超过t个诚实节点,所收到的$(n-t)$张有效$vote_4$投票一定包含其它诚实节点广播的$vote_4(v, r)$,因此所有诚实节点下一轮的初值均为v,协议一致性转化为已解决的有效性问题.

常数轮可终止性：考虑$n-t$个诚实节点在接收$n-t$张有效的$vote_4$最终投票后的状态全集，如果存在一个诚实节点P_i决定值v并输出，则所有诚实节点将在最多1轮后达成一致，决定值v并输出；如果所有诚实节点所收到的均为$vote_4(\perp, r)$，则所有诚实节点调用ADRB_{RLWE}，有$1-\mathrm{negl}(\lambda)$的概率产生相同的抛币值作为第$r+1$轮的初值，在这一情况下所有诚实节点将在最多1轮后达成一致；如果部分诚实节点所收到的均为$vote_4(\perp, r)$，以抛币值v'作为第$r+1$轮初值，另一部分诚实节点收到$vote_4(v, r)$和$vote_4(\perp, r)$，以v作为第$r+1$轮初值，则根据ADRB_{RLWE}的鲁棒性和不可预测性，$\Pr[v'=v]=1/2-\mathrm{negl}(\lambda)$，协议期望终止轮数为3轮.

整体性：本协议仅在过程2的步骤6决定并输出，且在输出后诚实节点会终止协议. 因此尽管诚实节点可能会执行多轮协议，但只会决定并输出一次，协议整体性得到保障.

因此，$\mathrm{CC}_{RLWE}-\mathrm{ABA}$是抗量子安全的异步常数轮二元共识协议.　　证毕.

6.2.2　协议复杂度分析

本协议所对应的网络传输结构如图2所示，将$\mathrm{CC}_{RLWE}-\mathrm{ABA}$协议的4次投票分别记为$Vote_1$到$Vote_4$，本协议在这四个阶段各进行一次$n$对$n$广播；取决于协议执行情况，$\mathrm{CC}_{RLWE}-\mathrm{ABA}$协议在$Vote_1$还可能进行第二轮$n$对$n$广播，在揭示抛币份额并验证的Reveal和ShareVrfy阶段进行一轮n对n广播.

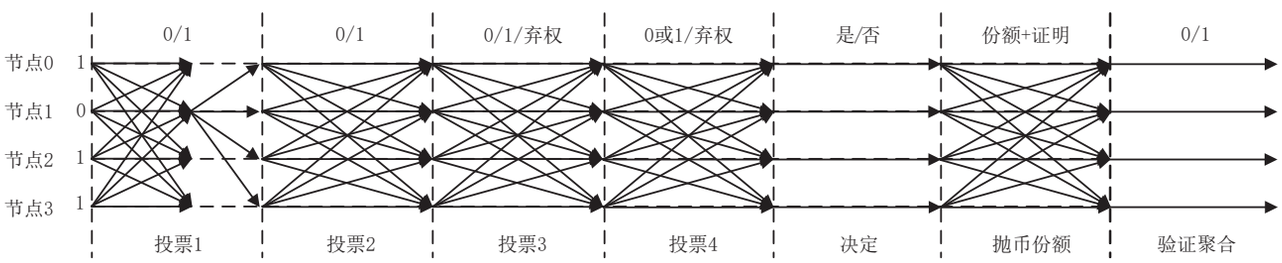

图2　$\mathrm{CC}_{RLWE}-\mathrm{ABA}$协议总览

因此，使用ADRB_{RLWE}的$\mathrm{CC}_{RLWE}-\mathrm{ABA}$协议每轮最多进行6步$n$对$n$广播，期望结束轮数为3轮，总共期望进行广播18次，通信复杂度为$O(n^2)$. 除抛币份额和证明外，各投票阶段的消息长度为一小常数，因此$CC_{RLWE}-ABA$协议的比特复杂度取决于ADRB_{RLWE}子协议，为$O(\mathrm{len}(p)\cdot KN\cdot n^2)$. 各节点的计算复杂度也取决于$\mathrm{ADRB}_{RLWE}$子协议，为$O(n\cdot N\log N)$；由于$N$不会随$n$的增加而增加，可认为$\mathrm{CC}_{RLWE}-ABA$协议的计算复杂度为$O(n)$.

7　实验设计与仿真测试

本文主要对公钥更新协议和异步分布式随机信标协议进行测试，所选择的对比方案是Cachin、Kursawe、Shoup[19]提出的基于离散对数构造的相似结构方案. 该对比方案以结构简单、运行速度快、支持异步网络环境等优点被广泛应用于HB-BFT[12]、Chronos[50]等经典的或最新的异步二元共识协议中.

本测试在Ubuntu22.04虚拟机内进行，硬件配置选用单虚拟核心，8GB内存与40GB的硬盘，主机处理器为Intel酷睿i7-1165G7. 软件环境采用Python3.9，主要使用的第三方库为支持高精度与大数运算的gmpy2和支持并行运算与多项式快速乘法的numpy.

7.1　公钥更新测试

本文在安全参数$\lambda=256$下实现了所设计的基于RLWE的公钥更新RLWEPkUpdate$=\{\mathrm{Prove}, \mathrm{Verify}\}$与作为对照组的同样基于256位安全参数构建的离散对数方案(模数按照NIST推荐，选取MODP6144，因此将该方案简记为MODP6144，所对应的异步分布式随机信标方案记为ADRB_{MODP})，用于比较两方案在证明和验证阶段的效率. 此外，为了验证理论推导中证明、验证阶段$O(N\log N)$的计算复杂度，本文调整N的取值，观察证明、验证时间关于N的变化关系.

如图3所示，令安全参数$\lambda=256$，RLWEPkUpdate方案平均单次证明所需时间为206 ms，平均单次验证所需时间为155 ms；而经典的基于离散对数假设构造的MODP6144方案平均单次证明所需时间为194 ms，平均单次验证所需时间为160 ms. 两协议运行效率相近：尽管MODP6144在证明中

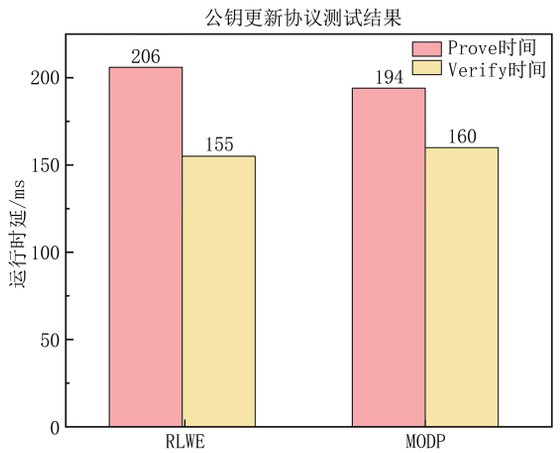

图3 $\lambda=256$ 时 RLWEPkUpdate 与 MODP6144 运行时延对比

略有优势，但在更高层级的异步分布式随机信标协议中，单个节点单轮仅会调用一次证明模块，却会多次调用 RLWEPkUpdate 更有优势的验证模块.

7.2 异步分布式随机信标协议测试

针对所设计的异步分布式随机信标协议 ADRB_{RLWE}，本文针对安全参数 $\lambda=256$ 测试了在不同输入规模下 Gen、Reveal、ShareVrfy、ShareCombine 四个函数的运行时间，ADRB_{RLWE} 的参数沿用 $N=256$, $K=3$, $len(p)=64$，对比方案也沿用与公钥更新协议采用同一模数的 ADRB_{MODP} 以同 ADRB_{RLWE} 对比安全参数 $\lambda=256$ 下协议的运行效率.

图4显示了 ADRB_{RLWE} 和 ADRB_{MODP} 的四个子算法在节点数 $n=4$、7、10 的时延测试结果，两协议除计算随机信标份额与 n 的增长无关外，其它三个子算法耗时相对于 n 线性增长. 至 $n=10$ 时，ADRB_{RLWE} 分发密钥耗时 830 ms，计算随机信标份额耗时 217 ms，验证通过 7 个正确的份额耗时 1029 ms，合并输出用时 4.5 ms；ADRB_{MODP} 分发密钥耗时 1047 ms，计算随机信标份额耗时 250 ms，验证通过 7 个正确的份额耗时 1401 ms，合并输出用时 253 ms.

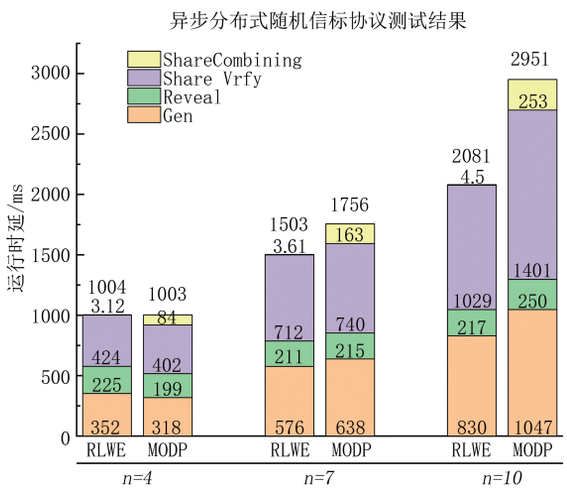

图4 ADRB_{RLWE} 和 ADRB_{MODP} 详细性能分析

纵向对比两异步分布式随机信标协议，主要耗时部分均为 ShareVrfy 子算法，这是由于在该算法处需要逐个调用公钥更新对份额进行验证. 横向对比 ADRB_{RLWE} 和 ADRB_{MODP} 两协议，相同安全参数 $\lambda=256$ 下 ADRB_{RLWE} 的性能略好于 ADRB_{MODP}，在 $n=10$ 下生成单个随机信标的总时延减小了 34%. 这是由于单个节点运行 Reveal 需要执行 1 次 Prove，而运行 ShareVrfy 需要执行 k 次 Verify.

8 总 结

本文针对异步共识协议难以同时满足量子安全和快速终止的关键问题，首先基于RLWE困难假设提出了一个总体性能与现有基于离散对数构造的方案相近的公钥更新协议 RLWEPkUpdate. 在此基础上，本文提出了基于RLWE的异步分布式随机信标协议 ADRB_{RLWE}，该协议采用公开可验证秘密共享的实现结构，在可信初始化借助同态映射可产生不限量的随机信标. 对于节点规模 n, 该协议的计算复杂度为 $O(n)$, 通信复杂度为 $O(n^2)$, 当 $n=10$ 时较同安全参数下基于离散对数构造的方案减小了 34% 的运行时延.

本文提出的 ADRB_{RLWE} 相比同类协议额外满足后量子密码学的需求，特别适用于需要抗量子安全的异步二元共识协议，且协议性能较目前开源代码主流使用的抛币模块具有优势. 因此，本文将该协议特化为抗量子安全的异步分布式抛币协议，并可插拔地应用到学界领先的 Quadratic-ABA 中，证明该改进能够在不降低原有协议安全性的前提下使异步二元共识协议达到常数轮期望终止.

本文为该领域未来的研究提供了如下方向：

(1) 本文仅给出了 ADRB_{RLWE} 在 Quadratic-ABA 中应用的示例. 可以将 ADRB_{RLWE} 结合其它异步二元共识协议或更高层的异步多元可验证拜占庭共识协议，进一步提升异步共识协议的效率.

(2)本文给出的 ADRB_{RLWE} 在节点数多时的噪声较大. 可以通过调整 RLWEPkUpdate 和 ADRB_{RLWE} 的参数选取或进一步优化份额生成及验证流程等方

法降低生成随机信标的时延、灵活调节协议安全性并降低协议失败概率.

(3)本文给出的$ADRB_{RLWE}$仍然需要可信第三方分发公私钥.可以参考Baum等人的研究[51],为$ADRB_{RLWE}$增加预处理阶段,由每个参与者分发一个秘密共同组成秘密多项式,探寻在密钥生成阶段满足同步网络假设时移除可信第三方,实现异步分布式密钥协商协议的可行性.

参 考 文 献

[1] Nielsen M A, Chuang I L. Quantum computation and quantum information. 10th Anniversary Edition. Cambridge, USA: Cambridge university press, 2001

[2] Shor P W. Algorithms for quantum computation: Discrete logarithms and factoring//Proceedings of the 35th IEEE Annual Symposium on Foundations of Computer Science (FOCS). Santa Fe, USA, 1994: 124-134

[3] Bhatia V, Ramkumar K R. An efficient quantum computing technique for cracking RSA using Shor's algorithm// Proceedings of the 5th IEEE International Conference on Computing Communication and Automation (ICCCA). Greater Noida, India, 2020: 89-94

[4] Shor P W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Review, 1999, 41(2): 303-332

[5] Bos J, Ducas L, Kiltz E, et al. CRYSTALS-kyber: A CCA-secure module-lattice-based KEM//Proceedings of the 3rd IEEE European Symposium on Security and Privacy (EuroS&P). London, UK, 2018: 353-367

[6] Ducas L, Kiltz E, Lepoint T, et al. Crystals-dilithium: A lattice-based digital signature scheme//Proceedings of the 20th IACR Transactions on Cryptographic Hardware and Embedded Systems(CHES). Amsterdam, The Netherlands, 2018: 238-268

[7] Fouque P A, Hoffstein J, Kirchner P, et al. Falcon: Fast-Fourier lattice-based compact signatures over NTRU. Submission to the NIST's Post-Quantum Cryptography Standardization Process, 2018, 36(5): 1-75

[8] Bernstein D J, Hülsing A, Kölbl S, et al. The SPHINCS+ signature framework//Proceedings of the 26th ACM SIGSAC Conference on Computer and Communications Security(CCS). London, UK, 2019: 2129-2146

[9] Yin An-Qi, Wang Ding, Guo Yuan-Bo, et al. Provably secure quantum resistance efficient password-authenticated key exchange protocol. Chinese Journal of Computers, 2022, 45(11): 2321-2336 (in Chinese)
(尹安琪,汪定,郭渊博,等.可证明安全的抗量子高效口令认证密钥交换协议.计算机学报,2022,45(11):2321-2336)

[10] Gao Yi-Tian, Chen Li-Quan, Tu Tian-Yang, et al. Post-quantum encryption technology based on BRLWE for internet of things. Chinese Journal of Network and Information Security, 2022,8(05):140-149 (in Chinese)
(高艺恬,陈立全,屠天扬等.基于BRLWE的物联网后量子加密技术研究.网络与信息安全学报,2022,8(05):140-149)

[11] Bernstein D J, Lange T. Post-quantum cryptography. Nature, 2017, 549(7671): 188-194

[12] Miller A, Xia Yu, Croman K, et al. The honey badger of BFT protocols//Proceedings of the 23rd ACM SIGSAC Conference on Computer and Communications Security (CCS). Vienna, Austria, 2016: 31-42

[13] Duan S, Reiter M, Zhang H. BEAT: Asynchronous BFT made practical//Proceedings of the 25th ACM SIGSAC Conference on Computer and Communications Security (CCS). Toronto, Canada, 2018: 2028-2041

[14] Guo B, Lu Y, Lu Z, et al. Speeding dumbo: Pushing asynchronous BFTcloser to practice//Proceedings of the 27th Network and Distributed System Security Symposium (NDSS). San Diego, USA, 2022: 385-402

[15] Gao Y, Lu Y, Lu Z, et al. Dumbo-ng: Fast asynchronous BFTconsensus with throughput-oblivious latency//Proceedings of the 29th ACM SIGSAC Conference on Computer and Communications Security (CCS). Los Angeles, USA, 2022: 1187-1201

[16] Mostéfaoui A, Moumen H, Raynal M. Signature-free asynchronous binary byzantine consensus with $t<n/3$, $O(n^2)$ messages, and $O(1)$ expected time. Journal of the ACM (JACM), 2015, 62(4):1-21

[17] Mostéfaoui A, Moumen H, Raynal M. Signature-free asynchronous byzantine consensus with $t<n/3$ and $O(n^2)$ messages//Proceedings of the 2014 ACM Symposium on Principles of Distributed Computing (PODC). Paris, France, 2014: 2-9

[18] Zhang H, Duan S, Zhao B, et al. WaterBear: Practical asynchronous BFT matching security guarantees of partially synchronous BFT//Proceedings of the 32nd USENIX Security Symposium (USENIX). Anaheim, USA, 2023:5341-5357

[19] Cachin C., Kursawe K., Shoup V. Random oracles in constantinople: Practical asynchronous byzantine agreement using cryptography. Journal of Cryptology, 2005, 18 (3): 219-246

[20] Bhat A, Shrestha N, Kate A, et al. Optrand: Optimistically responsive reconfigurable distributed randomness//Proceedings of the 28th Annual Network and Distributed System Security Symposium (NDSS). San Diego, USA, 2023: 832-849

[21] Das S, Krishnan V, Isaac I M, et al. Spurt: Scalable distributed randomness beacon with transparent//Proceedings of the 42th IEEE Symposium on Security and Privacy (S&P). San Francisco, USA, 2022: 2502-2517

[22] Bhat A, Shrestha N, Luo Z, et al. Randpiper-reconfiguration-friendly random beacons with quadratic communication// Proceedings of the 28th ACM SIGSAC Conference on Computer and Communications Security(CCS). Seoul, Korea, 2021: 3502-3524

[23] Cascudo I, David B. Albatross: Publicly attestable batched randomness based on secret sharing//Proceedings of the 26th

International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT). Daejeon, Korea, 2020: 311-341

[24] Schindler P, Judmayer A, Hittmeir M, et al. Hydrand: Efficient continuous distributed randomness//Proceedings of the 41th IEEE Symposium on Security and Privacy (S&P). Seoul, Republic of Korea, 2020: 73-89

[25] Syta E, Jovanovic P, Kogias E, et al. Scalable bias-resistant distributed randomness//Proceedings of the 38th IEEE Symposium on Security and Privacy (S&P). San Jose, USA, 2017: 444-460.

[26] Cascudo I, Dvid B. Scrape: Scalable randomness attested by public entities//Proceedings of the 15th International Conference on Applied Cryptography and Network Security (ACNS). Kanazawa, Japan, 2017: 537-556

[27] Guo Z, Shi L, Xu M. Secrand: Asecure distributed randomness generation protocol with high practicality and scalability. IEEE Access, 2020(8): 203917-203929

[28] Cascudo I, David B, Shlomovits O, et al. Mt. random: Multi-tiered randomness beacons//Proceedings of the 21st International Conference on Applied Cryptography and Network Security (ACNS). Kamakura, Japan, 2023: 645-674

[29] Galindo D, Liu J, Ordean M, et al. Fully distributed verifiable random functions and their application to decentralised random beacons//Proceedings of the 6th IEEE European Symposium on Security and Privacy (Euro S&P). Vienna, Austria, 2021: 88-102

[30] Gilad Y, Hemo R, Micali S, et al. Algorand: Scaling byzantine agreements for cryptocurrencies//Proceedings of the 26th Symposium on Operating Systems Principles (SOSP). Shanghai, China, 2017: 51-68

[31] Abidha V P, Barakbayeva T, Cai Z, et al. Gas-efficient decentralized random beacons//Proceedings of the 2024 IEEE International Conference on Blockchain and Cryptocurrency (ICBC). Dublin, Ireland, 2024: 1-6

[32] Beaver D, Chalkias K, Kelkar M, et al. STROBE: Streaming threshold random beacons//Proceedings of the 5th Conference on Advances in Financial Technologies (AFT). Princeton, USA, 2023, 7: 1-16

[33] Baum C, David B, Dowsley R, et al. Craft: Composable randomness beacons and output-independent abort MPC from time//Proceedings of the 26th IACR International Conference on Public-Key Cryptography (PKC). Atlanta, USA, 2023: 439-470

[34] Schindler P, Judmayer A, Hittmeir M, et al. Randrunner: Distributed randomness from trapdoor VDFs with strong uniqueness//Proceedings of the 26th Annual Network and Distributed System Security Symposium (NDSS). San Diego, USA, 2021: 116-133

[35] Baum C, David B, Dowsley R, et al. Tardis: a foundation of time-lock puzzles in UC//Proceedings of the 40th International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT). Zagreb, Croatia, 2021: 429-459

[36] Wang G, Nixon M. Randchain: Practical scalable decentralized randomness attested by blockchain//Proceedings of the 2020 IEEE International Conference on Blockchain (Blockchain). Rhodes Island, Greece, 2020: 442-449

[37] Daian P, Pass R, Shi E. SnowWhite: robustly reconfigurable consensus and applications to provably secure proof of stake//Proceedings of the Financial Cryptography and Data Security: 23rd International Conference (FC). Frigate Bay, St. Kitts and Nevis, 2019: 23-41

[38] Kiayias A, Russell A, David B, et al. Ouroboros: a provably secure proof-of-stake blockchain protocol//Proceedings of the 37th Annual International Cryptology Conference (CRYPTO). Santa Barbara, USA, 2017: 357-388

[39] Rabin M. Randomized byzantine generals//Proceedingsof 24th Annual Symposium on Foundations of Computer Science (FOCS). Tucson, USA, 1983: 403-409

[40] Shamir A. How to share a secret. Communications of the ACM, 1979, 22(11): 612-613

[41] Stadler M. Publicly verifiable secret sharing//Proceedings of the 15th International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT). Saragossa, Spain, 1996: 190-199

[42] Lyubashevsky V, Peikert C, Regev O. On ideal lattices and learning with errors over rings. Journal of the ACM, 2013, 60(6): 1-35

[43] Benhamouda F, Krenn S, Lyubashevsky V, et al. Efficient zero-knowledge proofs for commitments from learning with errors over rings//Proceedings of the European Symposium on Research in Computer Security (ESORICS). Vienna, Austria, 2015: 305-325

[44] Xie X, Xue R, Wang M. Zero knowledge proofs from Ring-LWE//Proceedings of the 12th Cryptology and Network Security: 12th International Conference (CANS). Paraty, Brazil, 2013: 57-73

[45] Cheon J H, Kim A, Kim M, et al. Homomorphic encryption for arithmetic of approximate numbers//Proceedings of International Conference on the 23rd Theory and Application of Cryptology and Information Security (ASIACRYPT). Hong Kong, China, 2017: 409-437

[46] Fiat A, Shamir A. How to prove yourself: Practical solutions to identification and signature problems//Proceedings of the 5th International Conference on the Theory and Application of Cryptographic Techniques (EUROCRYPT). Linköping, Sweden, 1986: 186-194

[47] Katz J, Lindell Y. Introduction to modern cryptography. 3rd Edition. Florida: CRC Press, 2020

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

[49] Stehlé D, Steinfeld R, Tanaka K, et al. Efficient public key encryption based on ideal lattices//Proceedings of International Conference on the 15th Theory and Application of Cryptology and Information Security (ASIACRYPT). Tokyo, Japan, 2009: 617-635

[50] Zhang Z, Zhang L, Wang Z, et al. Chronos: An efficient asynchronous byzantine ordered consensus. The Computer Journal, 2024, 67(3): 1153-1162

[51] Baum C, Jadoul R, Orsini E, et al. Feta: Efficient threshold designated-verifier zero-knowledge proofs//Proceedings of the 29th ACM SIGSAC Conference on Computer and Communications Security (CCS). Los Angeles, USA, 2022: 293-306

ZHANG Zong-Yang, Ph. D., associate professor. His research interest focuses on blockchain and cryptography.

LI Tian-Yu, Ph. D. candidate. His research interest focuses on blockchain and cryptography.

HU Bin, Ph. D. candidate. His research interest focuses on blockchain and cryptography.

ZHOU You, M. S. candidate. His research interest focuses on blockchain and cryptography.

ZHOU Xing-Guang, Ph. D., associate professor. Her research interest focuses on information security and cryptography.

Background

Distributed random beacon is a decentralized and trustless system that generates unpredictable random numbers with a public serial number called seed as input in a secure manner. The generated random numbers can be used in blockchain consensus mechanisms, smart contract applications, etc. This research belongs to the field of blockchain and cryptography. There is a similar concept called Distributed Key Generation in this field, which is committed to replacing trusted setups without losing any performance. The well-known concept of Distributed randomness generation is the union of the two.

At present, much research on distributed random beacons has been proposed. According to the Cryptography principles that it relies on, existing distributed random beacons can be divided into protocols based on Public Verifiable Secret Sharing (PVSS), protocols based on Verifiable Random Function (VRF), protocols based on Verifiable Delay Function (VDF), and other not yet systematic protocols. The PVSS model has a simple structure, clear modules, and strong practicality, making it the most numerous and thoroughly analyzed category among existing distributed random beacon protocols, however, it still has the disadvantage of high communication complexity. The research on the other categories of protocols started relatively late, and the design methods were not fixed. There are still significant gaps to be filled in the formal security definition and analysis. It is worth noting that at present, only a few PVSS-based schemes support asynchronous networks, and these schemes are constructed using the quantum-non-resistant discrete logarithm problem.

Therefore, this paper studies the scheme based on PVSS and focuses on the research of quantum-resistant security assumptions. We first proposed a public key update protocol to be used, then a quantum-resistant asynchronous distributed random beacon protocol based on Ring Learning with Errors (RLWE) assumption, which reached the minimum $O(n^2)$ communication complexity and $O(n)$ computational complexityamong similar protocols. Under the same security parameters, our protocol saves about 34% execution latency when compared with the variant protocol based on the discrete logarithm assumption. Note that our proposed scheme has two shortcomings, it needs a trusted third party to setup and its noise will be difficult to control when the node number is large.

This work is supported in part by the National Key Research and Development Program of China (2022YFB2702702), the National Natural Science Foundation of China (62372020, 61972017, 72031007), the Beijing Natural Science Foundation (M22038, L222050), the Fundamental Research Funds for the Central Universities (YWF-23-L-1032), and the Research Foundation for Youth Scholars of Civil Aviation Management Institute of China (23QN06).

Blockchain and cryptography are the main research topics of our research group. In the past few years, we have worked out many research papers that have been published in Computers & Security, Journal of Cryptologic Research, and other journal.