一种基于同态向量承诺和压缩Σ协议的高效聚合环签名方案

叶青[1]，邵震杰[1]，汤永利[2]，王一川[1]，黑新宏[1]
（1.西安理工大学计算机科学与工程学院，陕西 西安 710048；2.河南理工大学软件学院，河南 焦作 454003）

摘要：针对现有环签名方案在大规模环成员和多签名验证场景下存在通信成本大、验证开销高等问题，设计了一种基于同态向量承诺和压缩∑协议的新型聚合环签名方案。该方案首先基于BLS签名、$(\mathbb{Z}_q,\mathbb{G}_1)$向量承诺、面向$(\mathbb{Z}_q,\mathbb{G}_1)$向量承诺的压缩∑协议生成单个环签名，进一步基于类ElGamal向量承诺及面向类ElGamal向量的∑协议将k个单环签名聚合为一个环签名进行传输与验证。理论分析与实验验证表明，该方案无需可信第三方，满足匿名性与不可伪造性等安全特性，且k个单签名的总签名长度、验证时间均为$O(k)$，与环大小无关，效率显著优于同类方案。

关键词：聚合环签名；向量承诺；压缩∑协议；双线性群
中图分类号：TP309
文献标志码：A
doi：10.11959/j.issn.1000

An efficient aggregate ring signature scheme based on homomorphic vector commitment and compressed Σ protocol

Ye Qing[1], Shao Zhenjie[1], Tang Yongli[2], Wang Yichuan[1], Hei Xinhong[1]
1. School of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710048, China
2. School of Software Engineering, Henan Polytechnic University, Jiaozuo 454003, China

Abstract: Aiming at the problems of high communication cost and large verification overhead in existing ring signature schemes under scenarios of large-scale ring members and multi-signature verification, this paper designs a novel aggregate ring signature scheme based on homomorphic vector commitments and compressed ∑ protocols. The scheme first generates individual ring signatures based on BLS signatures,$(\mathbb{Z}_q,\mathbb{G}_1)$vector commitments, and the compressed ∑ protocol for $(\mathbb{Z}_q,\mathbb{G}_1)$ vector commitments. Furthermore, based on ElGamal-like vector commitments and the ∑ protocol for ElGamal-like vectors, it aggregates k individual ring signatures into one aggregated ring signature for transmission and verification. Theoretical analysis and experimental verification demonstrate that the proposed scheme does not require a trusted third party and satisfies security properties such as anonymity and unforgeability. Moreover, the total signature length and verification time for k individual signatures are both $O(k)$, independent of the ring size, and the efficiency is significantly superior to similar schemes.

Key words: aggregate ring signature, vector commitment, compressedprotocol, bilinear group

0 引言

区块链的技术优势之一在于平衡数据公开透明与用户身份隐私保护，而环签名技术正是实现这一平衡的关键密码学支撑。自2001年Rivest等人[1]首次提出环签名概念以来，因具有“无需环成员协作

收稿日期：XXXX–XX–XX；修回日期：XXXX–XX–XX
通信作者：汤永利，yltang@hpu.edu.cn
基金项目：国家自然科学基金资助项目（No.62472144）陕西省自然科学基础研究计划项目（No.2026JC-YBMS-0727）
Foundation Items：The National Natural Science Foundation of China (No.62472144),Natural Science Basic Research Program of Shaanxi Province (No.2026JC-YBMS-0727)

即可生成签名，以及验证者仅知签名来自环内却无法定位具体签名者”的特性，环签名已在加密货币、区块链、匿名投票等领域广泛应用——例如门罗币[2]便通过环签名技术隐藏交易发起者身份，保障去中心化金融场景下的用户隐私。

然而，随着区块链交易量的增加，传统环签名方案逐渐暴露出效率瓶颈。环签名尺寸随环大小线性增加，会导致区块链交易数据膨胀、手续费上升；而签名验证效率低下，则降低了网络吞吐量，增加系统延迟。国内外关于降低签名长度和提高签名验证效率方面做了大量的工作，例如基于累加器的环签名[3]，基于同态承诺和零知识证明的环签名[4]，这些研究成果主要专注于单个环签名长度和验证效率的极致优化。但在区块链实际场景中，同一时间段内往往产生多笔匿名交易需要批量验证，仅优化单个环签名的性能已难以满足高吞吐需求，如何实现多个环签名的高效聚合与一次性验证，成为亟待解决的关键问题。

2025年，借鉴聚合签名（Aggregate Signature, AS）[5]的思想，Tong等人[6]提出了聚合环签名（Aggregate Ring Signature,ARS）的概念，有效结合环签名的匿名隐私特性与AS的批量压缩优势。普通环签名只能实现单次匿名签名，批量交易场景下每份签名独立存储、逐一验证，通信与计算开销巨大；而ARS可将多个用户生成的多个单环签名合并为一个ARS，仅需一次验证操作即可批量确认所有签名合法性。ARS主要应用于匿名投票、区块链等存在大量匿名核验的场景，有效降低链上存储、网络带宽与节点验证算力消耗。Tong等人形式化定义了ARS的算法框架与安全模型，并基于“BLS签名+密钥同态签名”、向量承诺以及其对应的压缩∑协议给出具体的ARS构造。但分析该方案发现，使用“BLS签名+密钥同态签名”技术生成的环签名存在根本性安全缺陷，难以同时具备匿名性和不可伪造性。具体而言，签名者使用“BLS签名+密钥同态签名”，貌似能实现BLS签名的随机化以达到环签名的匿名性，但随机化的签名并不是真正的环签名，它隐藏着任何人都可伪造的风险。为了验证随机化签名的非伪造性且执行聚合操作，方案中必须存在一个特殊的可信角色—聚合者（Aggregator）。它介于签名者与验证者之间，负责在聚合签名验证前，对批量单个环签名进行聚合。注意到这个角色在执行聚合操作前，需遍历环中公钥以验证单个随机化签名的非伪造性，这样会导致签名者身份完全暴露给聚合者。进一步地，不只是聚合者，任意一个公开验证者都可执行此操作以识别出各单个签名的签名者，从而方案不满足匿名性。

为了克服上述缺陷，本文基于BLS签名、$(\mathbb{Z}_q,\mathbb{G}_1)$向量承诺及其对应压缩∑协议、类ElGamal向量承诺及其对应∑协议构造了一种新的ARS方案。具体地，与同类方案，尤其Tong等人[6]方案相比，本文的创新性在于：

（1）采用新技术构造ARS的高效底层方案。

以往基于累加器、零知识证明或密钥同态签名构造的单环签名方案，仅追求单环签名性能达到最优，或存在安全性缺陷，或技术层面不支持直接聚合，从而难以进行签名的批量传输和验证。本文受Tong等人[6]和Attema等人[7]的启发，基于BLS签名，$(\mathbb{Z}_q,\mathbb{G}_1)$向量承诺及其对应压缩∑协议生成单个环签名。单环签名长度为$O(\log n)$，验证时间为$O(\log n)$，其中n代表环大小，满足匿名性和不可伪造性，且支持签名聚合操作。

（2）采用新技术实现单环签名聚合。Tong等人方案采用可信聚合者这一特殊角色，借助$(\mathbb{Z}_q,\mathbb{G}_1)$向量承诺及其对应压缩∑协议完成单签名聚合，该技术无法实现对承诺值的聚合，本文采用Attema等人[7]提出的类ElGamal向量承诺及其配套的∑协议作为聚合技术，支持对承诺值进行承诺并生成相关零知识证明，进而构造出简洁高效的ARS。

理论分析和实验结果表明，本文所提ARS方案同时具备匿名性和不可伪造性，并且对于k个环签名，本方案的总签名长度和验证代价均为$O(k)$，与环大小无关，效率明显优于目前同类环签名或ARS方案。另外，方案无需可信的聚合者这一特殊角色，签名的聚合操作可由任意用户完成，更适合高频交易的大型区块链场景。

1 相关工作

自Rivest等人[1]于2001年首次提出环签名以来，环签名逐渐发展成为密码学中兼具签名者匿名性与公开可验证性的重要原语。早期环签名研究主要围绕降低签名长度、提升计算效率以及摆脱随机

预言机模型等问题展开。例如，Groth 和 Kohlweiss[8]基于DDH假设，借助1-out-of-N承诺与零知识证明框架构造签名长度为对数级的环签名方案，为短环签名研究奠定了重要基础。随后，Malavolta等人[9]于2017年在标准模型下，基于可重随机化密钥签名与非交互式零知识证明构造了高效环签名方案。

在环签名提供强匿名性的同时，其签名者完全匿名特性也可能导致重复签名、双花攻击或恶意滥用等问题。为在匿名性与责任追溯之间取得平衡，可链接环签名和可追踪环签名逐渐成为重要研究方向。Liu等人[10]提出了LSAG方案，将“同一签名者可关联”的机制引入环签名，使得由同一私钥生成的多个签名能够被识别，从而在不直接暴露签名者身份的前提下实现重复行为检测。Fujisaki和Suzuki[11]进一步提出了可追踪环签名，使得同一标签下针对不同消息的重复签名能够被有效追踪。近年来，Yuen等人[12]提出的DualRing改进了环签名的构造范式；Feng等人[13]和Ye等人[14]的工作则分别从后量子安全性和区块链应用需求出发，推动了可追踪环签名及其相关变体的发展。

随后环签名被进一步引入匿名加密货币和隐私增强型区块链中。Monero[2]通过结合环签名与机密交易技术，形成了环机密交易（Ring Confidential Transactions）协议，同时实现了交易发送者匿名性和交易金额隐藏性。早期的RingCT1.0[2]协议针对m个交易输入，需生成$m+1$个签名，且每个签名大小为$O(n)$，因此在通信开销和验证效率方面存在较大优化空间。另外，RingCT1.0也并未给出RingCT完整的形式化定义和安全模型。随后的RingCT2.0[15]弥补了这一缺陷，并通过引入可信参数提升了协议效率。然而，在开放的区块链环境中，对可信设置的依赖通常被视为不利因素，后续的RingCT 3.0[16]协议进一步消除了对可信设置需求，从而增强了协议在区块链场景中的适用性。另外，Lai等人[17]提出了Omniring协议，能够有效抵御RingCT早期研究尚未充分考虑的实际攻击。Duan等人[18]则基于可链接门限环签名构造了新的RingCT协议，在一定程度上增强了RingCT协议的灵活性和适用范围。然而，这些方案仍面临交易规模增长导致的通信开销和验证成本增加问题，且不支持将多笔交易聚合为单笔交易。

为进一步降低环签名的通信开销与验证成本，近年来研究者开始关注ARS在隐私交易中的应用与将零知识证明压缩技术引入环签名构造。Teng等人[19]于2025年提出面向隐私增强区块链ARS方案Agg-DualRing。该方案在DualRing[12]环签名基础上引入聚合思想和内积证明机制，有效压缩了签名大小。Attema等人[20]于2020年首次提出压缩∑协议，将同态向量承诺对应的零知识证明尺寸由线性级压缩为对数级；随后，Attema等人[7]将该协议推广至双线性电路关系中，构造了一系列面向双线性电路的同态向量承诺算法及对应的压缩∑协议，例如：$(\mathbb{Z}_q,\mathbb{G}_1)$向量承诺、$(\mathbb{Z}_q,\mathbb{G}_1,\mathbb{G}_2)$向量承诺以及$(\mathbb{G}_T)$向量承诺（也称为类ElGamal向量承诺），为环签名的尺寸压缩与高效验证提供了新的技术路径。Tong等人[6]基于Attema等人提出的压缩∑协议，针对区块链交易场景提出支持聚合能力的环签名方案，使得同一环内多个签名能够被合成一个更加紧凑的聚合签名。尽管此方案在降低通信开销方面取得了一定进展，但其验证时间仍随环大小线性增长，难以满足大规模区块链系统中的高效验证需求。此外，Tong等人[6]的方案还存在一定的匿名性问题。

综上，如何保证匿名性、不可伪造等安全性的同时，实现更短签名长度、更低的验证复杂度，仍是当前环签名及其区块链应用研究中的重要挑战。

2　预备知识

2.1　符号说明

$\mathbb{Z}_q$为模q剩余类；$[1,n]$代表集合$\{1,\cdots,n\}$；$h\xleftarrow{R}\mathbb{G}$表示在群$\mathbb{G}$中任意随机选取一元素$h$；$|\mathbb{G}|$和$|S|$分别表示群$\mathbb{G}$中元素大小，集合$S$中元素个数。

2.2　双线性映射

双线性群$(\mathbb{G}_1,\mathbb{G}_2,\mathbb{G}_T,q,e,G,H)$[21]由3个$q$阶代数群$\mathbb{G}_1$、$\mathbb{G}_2$、$\mathbb{G}_T$和一个双线性映射$e:\mathbb{G}_1\times\mathbb{G}_2\to\mathbb{G}_T$构成。$G,H$及$e(G,H)$分别为群$\mathbb{G}_1,\mathbb{G}_2$和$\mathbb{G}_T$的生成元。对于向量$\boldsymbol{G}\in\mathbb{G}_1^n$，$\boldsymbol{H}\in\mathbb{G}_2^n$，定义内积$e(\boldsymbol{G},\boldsymbol{H})=\sum_{i=1}^{n}e(G_i,H_i)$。双线性映射$e$具有以下性质：

（1）双线性：对所有$P\in\mathbb{G}_1$、$Q\in\mathbb{G}_2$及$a,b\in\mathbb{Z}_q$，满足$e(aP,bQ)=e(P,Q)^{ab}$。

（2）非退化性：存在$P\in\mathbb{G}_1$、$Q\in\mathbb{G}_2$，使得

$e(P,Q)\neq 1$，其中1代表$\mathbb{G}_T$的单位元。

（3）可计算性：对所有$P\in\mathbb{G}_1$、$Q\in\mathbb{G}_2$，存在高效算法可计算$e(P,Q)$。

2.3 知识证明

知识证明（Proof of Knowledge, PoK）[22]是交互式协议。证明者$\mathcal{P}$在不泄露私有证据的前提下，向验证者$\mathcal{V}$证明自身持有满足条件的证据。定义二元关系$R_f\subseteq\mathcal{X}\times\mathcal{W}$，其中$x\in\mathcal{X}$为公开声明信息，$w\in\mathcal{W}$为私有证据满足约束$(x;w)\in R_f$。协议满足：

（1）完备性：合法$(x;w)\in R_f$生成的证明Π必然被验证者$\mathcal{V}$接受。

（2）知识可靠性：若不诚实证明者$\mathcal{P}^*$可通过$\mathcal{V}$的验证，则存在多项式时间算法$\mathcal{E}$，可在多项式时间内提取证据w。

（3）诚实验证者零知识性（HVZK）：模拟算法$\mathcal{S}$可模拟出与真实交互过程不可区分的会话脚本。

（4）证据不可区分性：关于同一公开声明x，若存在两组合法证据$w_1,w_2\in\mathcal{W}$，验证者无法从交互记录区分证明者实际使用的证据。

2.4 向量承诺方案

定义1（向量承诺[23]）：非交互式向量承诺方案由概率多项式时间（Probabilistic Polynomial Time，PPT）算法组成，记作(Setup, Com, Open, Verify)：

（1）Setup：输入安全参数λ、向量维度q，该算法生成并输出公开参数pp。

（2）Com：设随机数空间为R，消息明文空间为M，承诺值空间为C。输入k维消息序列$m_1,\cdots,m_k\in M$，算法输出对应向量承诺值$\mathbf{Com}\in C$。

（3）Open：基于公开参数与原始消息序列，生成证明Π，证明序列$m_1,\cdots,m_k$为对应被承诺消息。

（4）Verify：若证明Π有效输出1；否则输出0。

定义2（$(\mathbb{Z}_q,\mathbb{G}_1)$向量承诺方案[24-25]）：面向向量空间$\mathbb{Z}_q^{n_1}\times\mathbb{G}_1^{n_2}$，$n_1,n_2\geqslant 0$的承诺方案定义如下：

（1）Setup：$h\xleftarrow{R}\mathbb{G}_T$，$\boldsymbol{g}=(g_1,\cdots,g_{n_1})\xleftarrow{R}\mathbb{G}_T^{n_1}$，$\boldsymbol{H}=(H_1,\cdots,H_{n_2})\xleftarrow{R}\mathbb{G}_2^{n_2}$。

（2）Com：选取$\gamma\xleftarrow{R}\mathbb{Z}_q$，$(\boldsymbol{x},\boldsymbol{y},\gamma)\mapsto h\gamma+<\boldsymbol{g},\boldsymbol{x}>+e(\boldsymbol{y},\boldsymbol{H})\in\mathbb{G}_T$。

（3）Open：承诺方生成一个证明Π，证明向量$(\boldsymbol{x},\boldsymbol{y})$为被承诺向量。

（4）Verify：若Π有效输出1；否则输出0。

上述同态承诺方案在DDH假设下具有计算隐藏性，在SXDH假设[25]下，具有计算绑定性。

定义3（$\mathbb{G}_T$向量承诺方案[25-26]，类ElGamal向量承诺）：面向$\mathbb{G}_T^{n_T}$向量空间的承诺方案具体如下：

（1）Setup：选取$\boldsymbol{g}\xleftarrow{R}\mathbb{G}_T^{n_T},h\xleftarrow{R}\mathbb{G}_T$。

（2）Com：选取$\gamma\xleftarrow{R}\mathbb{Z}_q$，$(\boldsymbol{x},\gamma)\mapsto\begin{pmatrix}h\gamma\\ \boldsymbol{g}\gamma+\boldsymbol{x}\end{pmatrix}\in\mathbb{G}_T^{n_T+1}$。

（3）Open：承诺方生成一个证明Π，证明向量$\boldsymbol{x}$为被承诺向量。

（4）Verify：若Π有效输出1；否则输出0。

上述同态承诺方案在DDH假设下具有计算隐藏性和绑定性。

2.5 压缩Σ协议技术

在本文构造ARS方案中，签名者需对向量进行承诺，并为承诺向量生成知识证明。基础Σ协议（见图1中Π_0协议）可实现该知识证明，但证明脚本长度随承诺向量维度线性增加，通信开销较高。为提升证明紧凑性与协议效率，本方案在单环签名生成阶段采用压缩Σ协议[7]。该协议通过递归折叠机制压缩高维证明脚本，在完整保留完备性、特殊可靠性、HVZK性等核心安全性质基础上，将证明脚本长度降至对数级。

压缩Σ协议（见图1中Π_1协议）将证据划分为可压缩和不可压缩两部分来分别处理。设$\Psi:\mathbb{G}_1\times\mathbb{G}_2\to\mathbb{G}_T$为$\mathbb{Z}_q$上的同态映射，$\mathbb{G}_1$为可压缩证据空间，$\mathbb{G}_2$为非可压缩证据空间。对任意公开声明$y\in\mathbb{G}_T$，证明者需证明自身持有有效证据$x=(x_1,x_2)\in\mathbb{G}_1\times\mathbb{G}_2$，满足$y=\Psi(x_1,x_2)$。对应关系定义为：$R_\Psi=\{(y;x):y=\Psi(x)\}$。

设可压缩空间$\mathbb{G}_1=\mathbb{G}_0\times\mathbb{G}_0$，则证据$x_1\in\mathbb{G}_1$可拆分为$x_1=(x_1^L,x_1^R)\in\mathbb{G}_0^2$。证明者计算辅助承诺$a=\Psi((0,x_1^L),0),b=\Psi((x_1^R,0),0)$，结合挑战值$c\in\mathbb{Z}_q$折叠证据，生成短响应$z=x_1^L+cx_1^R$。同时，更新声明$y'=a+cy+c^2b$与同态映射关系$\Psi'(z,x_2)=\Psi((cz,z),cx_2)$，将原约束$y=\Psi(x_1,x_2)$

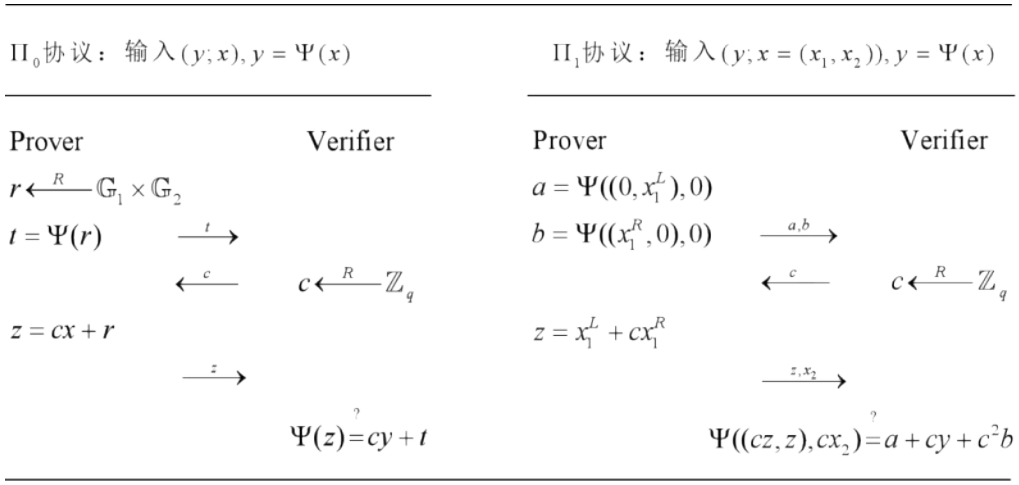

图1　基础Σ协议和压缩Σ协议

等价转化为$y'=\Psi'(z,x_2)$的知识证明。该折叠在不改变证明目标、保证证明等价性前提条件下，实现脚本规模减半。

针对$\mathbb{G}_1=\mathbb{G}_0^n$的高维向量场景，上述单次折叠机制可递归执行，实现高维证据的逐层压缩。Π_0、Π_1的多轮递归折叠形成的完整压缩Σ协议可表示为：$\Pi_c=\underbrace{\Pi_1\diamond\Pi_1\cdots\Pi_1}_{\mu次}\diamond\Pi_0$，式中迭代压缩次数$\mu=\lceil\log_2 n\rceil$。协议先通过$\mu$轮递归折叠，将$n$维原始证据压缩为单个$\mathbb{G}_0$元素，最终基于第$\mu$轮递归折叠后的声明与证据运行基础Σ协议完成证明。若向量维度n非2的整数次幂，可通过补零方式补齐维度以适配递归规则。

上述交互式压缩Σ协议，可通过Fiat-Shamir启发式算法转化为非交互式协议。协议共执行μ轮递归折叠，在最终的第μ轮递归折叠过程中，证明者首先计算本轮对应的两组辅助承诺项：$a_\mu=\Psi_\mu((0,x_{1_\mu}^L),0)$，$b_\mu=\Psi_\mu((x_{1_\mu}^R,0),0)$。随后通过哈希函数生成对应挑战值：$c_\mu=\mathcal{H}(a_\mu,b_\mu,x)$，依据该挑战值完成本轮证据折叠运算与声明更新：

$$z_\mu=x_{1_\mu}^L+c_\mu x_{1_\mu}^R \tag{1}$$

$$y_\mu=a_\mu+c_\mu y_{\mu-1}+c_\mu^2 b_\mu \tag{2}$$

经过μ轮递归折叠压缩后，初始高维证据简化为单维证据$x_{1_\mu}^L,x_{1_\mu}^R\in\mathbb{G}_0$，从而$z_\mu\in\mathbb{G}_0$。全部递归折叠结束后，进入最后一轮Σ协议证明，随机选取参数$r\xleftarrow{R}\mathbb{G}_0\times\mathbb{G}_2$，计算映射结果：$t=\Psi_\mu(r)$，验证者返回最终挑战$c=\mathcal{H}(t,x)$。得到最终响应值为：

$$z=c\cdot(z_\mu,x_2)+r \tag{3}$$

Π_c输出的非交互式压缩Σ协议完整证明结构为：$\Pi=(a_1,b_1,\cdots,a_\mu,b_\mu,t,z)\in\mathbb{G}_T^{2\mu+1}\times\mathbb{G}_0\times\mathbb{G}_2$。

2.6　面向类El Gamal向量承诺的Π_{EG}协议

压缩Σ协议适用于打开双线性群的$\mathbb{G}_1$或$\mathbb{G}_2$上的向量承诺，而对于定义在$\mathbb{G}_T$上的向量承诺（见定义3）则完全没有压缩效果，文献[7]巧妙地将类ElGamal向量承诺中$\mathbb{G}_T$向量的知识证明转化为随机数γ的知识证明，以此构造出高效的Σ协议Π_{EG}（如图2所示）。定义同态映射$f:\mathbb{G}_T^{n_T}\times\mathbb{Z}_q\to\mathbb{G}_T^{n_T+1}$，$\Pi_{EG}$协议以证明$\mathbb{G}_T$承诺向量满足约束$f(\boldsymbol{x})=y$为目标，证明关系为$R_{EG}=\{(P,y;\boldsymbol{x},\gamma):P=Com(\boldsymbol{x},\gamma),f(\boldsymbol{x})=y\}$。协议输出的完整证明脚本为：$\Pi_{agg}=(A,t,\phi)$。

该协议通信开销为常数级与向量维度无关，同时具备完备性、特殊诚实验证者零知识性与特殊可靠性。本文将其应用于ARS方案中签名聚合阶段。

2.7　ARS定义及安全模型

文献[6]给出ARS定义与安全模型，具体如下：

定义4：ARS方案由算法Setup、KeyGen、Sign、Verify、AggreSign、AggreVerify组成。

$pp\leftarrow Setup(1^\lambda)$：PPT算法，输入安全参数$\lambda$，输出公共参数$pp$;

$(sk,pk)\leftarrow \mathrm{KenGen}(pp)$：PPT算法，输入公共参数，输出私钥$sk$和公钥$pk$。

$\sigma\leftarrow \mathrm{Sign}(m,sk,\mathcal{R})$：PPT算法，输入待签名消

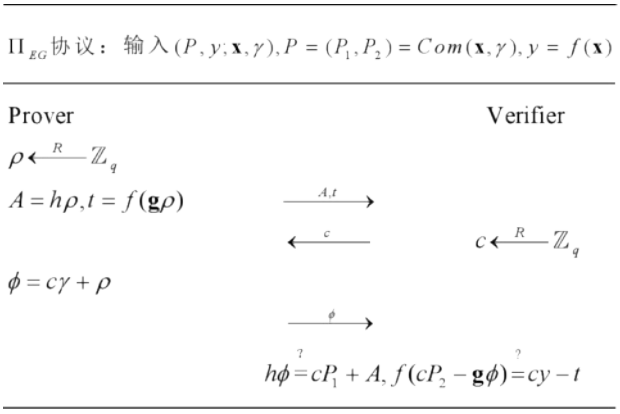

图2 Π_{EG}协议

息m、签名者私钥sk和公钥环集合$\mathcal{R}\subseteq\{pk_1,\cdots,pk_n\}$，其中包含$sk$对应的公钥，输出一个单环签名$\sigma$。

1/0 ← Verify$(m,\mathcal{R},\sigma)$：确定多项式时间算法，输入公钥集合$\mathcal{R}$、消息m和签名σ，输出1代表σ有效，输出0代表σ无效。

$\sigma_{agg}/\perp\leftarrow$ AggreSign$(\{m_i\}_{i\in S},\{\sigma_i\}_{i\in S},S,\mathcal{R})$：PPT算法，输入消息集合$\{m_i\}_{i\in S}$，对应单环签名集合$\{\sigma_i\}_{i\in S}$、环$\mathcal{R}$以及签名者集合$S\subseteq\{1,\cdots,n\}$。若所有单签名有效，则输出ARS$\sigma_{agg}$，否则输出$\perp$。

1/0 ← AggreVerify$(\{m_i\}_{i\in S},\mathcal{R},\sigma_{agg})$：确定多项式时间算法，输入消息集合、环和ARS$\sigma_{agg}$，输出1表示ARS有效，输出0表示无效。

正确性(Correctness)：ARS正确性包括普通验证正确性和聚合验证正确性两方面。

定义5 普通验证正确性：对任意公私钥对$(pk_i,sk_i)\leftarrow$ KeyGen(1^λ)、任意环$\mathcal{R}\subseteq\{pk_1,\cdots,pk_n\}$、任意合法单签名$\sigma\leftarrow$ Sign$(m,sk_s,\mathcal{R})$，Verify$(m,\mathcal{R},\sigma)$输出0的概率是可忽略的，即$\Pr[\text{Verify}(m,\mathcal{R},\sigma)=0]\leqslant negl(\lambda)$。

定义6 聚合验证正确性：对任意公私钥对$(pk_i,sk_i)\leftarrow$ KeyGen(1^λ)、任意环$\mathcal{R}\subseteq(pk_1,\cdots,pk_n)$、任意合法单签名$\sigma_i\leftarrow$ Sign$(m_i,sk_i,\mathcal{R}_i)$，任意合法聚合签名$\sigma_{agg}\leftarrow$ AggreSign$(\{m_i\}_{i\in S},\{\sigma_i\}_{i\in S},S,\mathcal{R})$，AggreVerify$(\{m_i\}_{i\in S},\mathcal{R},\sigma_{agg})$输出0的概率是可忽略的，$\Pr[\text{AggreVerify}(\{m_i\}_{i\in S},\mathcal{R},\sigma_{agg})=0]\leqslant negl(\lambda)$。

安全特性方面，除一般环签名的匿名性、不可伪造性外，ARS还需具备聚合匿名性和聚合不可伪造性。Tong等人[6]的ARS安全模型允许PPT敌手在实验过程中不受限制地访问4种预言机：OKeyGen、OSign、OAggreSign和OCorr，预言机都能访问四个集合：环$\mathcal{R}$、环中被腐化成员集合$\mathcal{R}_{corr}$、签名者身份与密钥信息集合L_{sig}、签名询问集合Q。下面为预言机与安全属性的形式化定义。

（1）OKeyGen(i)：若$(i,\cdot,\cdot)\notin L_{sig}$，则生成密钥对$(pk_i,sk_i)\leftarrow KeyGen(1^\lambda)$，将$(i,pk_i,sk_i)$添加至$L_{sig}$，$pk_i$给$\mathcal{R}$,并将$pk_i$返回给攻击者。

（2）OSign$(m,i,\mathcal{R})$：若$i\in\mathcal{R}$、$i\notin\mathcal{R}_{corr}$且$(i,pk_i,sk_i)\in L_{sig}$，该预言机运行$Sign(m,sk_i,\mathcal{R})$返回普通环签名给攻击者，并将$(m,i,\mathcal{R})$添加到$Q$中。

（3）OAggreSign$(\{m_i\}_{i\in S},S,\mathcal{R})$：若$S\subseteq\mathcal{R}\wedge|S\cap\mathcal{R}_{corr}|\leqslant|S|-1$且对$i\in S$，都有$(i,pk_i,sk_i)\in L_{sig}$，该预言机为每个$i\in S$，计算单环签名$\sigma_i$，运行AggreSign$(\{m_i\}_{i\in S},\{\sigma_i\}_{i\in S},S,\mathcal{R})$返回一个ARS$\sigma_{agg}$给攻击者，并将$(\{m_i\}_{i\in S},S,\mathcal{R})$添加至$Q$。

（4）OCorr(i)：若存在$(i,pk_i,sk_i)\in L_{sig}$，则返回sk_i给攻击者，并将访问过的pk_i添加至$\mathcal{R}_{corr}$。

匿名性(Anonymity)：ARS方案的匿名性包括普通匿名性和聚合匿名性两方面。

普通匿名性确保单环签名不泄露真实签名者身份，对应攻击者$\mathcal{A}$与挑战者$\mathcal{C}$之间的游戏$Game_{\mathcal{A}}^{Anno}(1^\lambda)$见图3。$\mathcal{A}$赢得该游戏优势定义为：

$$Adv_{\mathcal{A}}^{Anon}=\left|\Pr[b'=b]-\frac{1}{2}\right| \tag{4}$$

定义7 对于任意PPT攻击者$\mathcal{A}$，$Adv_{\mathcal{A}}^{Anno}$是可忽略的，则ARS方案满足普通匿名性。

聚合匿名性确保聚合过程仍能保护诚实签署者身份，对应攻击者$\mathcal{A}$与挑战者$\mathcal{C}$之间的游戏$Game_{\mathcal{A}}^{AAnno}(1^\lambda)$见图3。$\mathcal{A}$赢得该游戏的优势定义为：

$$Adv_{\mathcal{A}}^{AAnon}=\left|\Pr[b'=b]-\frac{1}{2}\right| \tag{5}$$

定义8 对于任意PPT攻击者$\mathcal{A}$，$Adv_{\mathcal{A}}^{AAnon}$是可忽略的，则ARS方案满足聚合匿名性。

不可伪造性(Unforgeability)：ARS方案的不可伪造性包括普通不可伪造性和聚合不可伪造性两方面。

普通不可伪造性确保环外成员无法产生该环的有效签名，对应攻击者$\mathcal{A}$与挑战者$\mathcal{C}$之间的游戏

$Game_{\mathcal{A}}^{Anno}(1^{\lambda})$

Run $pp \leftarrow \text{Setup}(1^{\lambda})$
$O \leftarrow \{O\text{KeyGen}, O\text{Sign}, O\text{Corr}\}$
$(m^*, s_0^*, s_1^*, \mathcal{R}^*) \leftarrow \mathcal{A}^O(pp)$
$b \xleftarrow{R} \{0,1\}$
$\sigma_b \leftarrow O\text{Sign}(m^*, sk_{s_b}^*, \mathcal{R}^*)$
$b' \leftarrow \mathcal{A}^O(\sigma_b)$
if $s_0 \in \mathcal{R}_{corr} \vee s_1 \in \mathcal{R}_{corr}$, *or*
$(m^*, s_0^*, \mathcal{R}^*) \in Q \vee (m^*, s_1^*, \mathcal{R}^*) \in Q$
  return 0
return 1 *iff* $b = b'$
otherwise, *return* 0

$Game_{\mathcal{A}}^{AAnno}(1^{\lambda})$

Run $pp \leftarrow \text{Setup}(1^{\lambda})$
$O \leftarrow \{O\text{KeyGen}, O\text{Sign}, O\text{Corr}, O\text{AggreSign}\}$
$(M^*, S_0^*, S_1^*, \mathcal{R}^*) \leftarrow \mathcal{A}^O(pp)$
$b \xleftarrow{R} \{0,1\}$
$\sigma_{agg_b} \leftarrow O\text{AggreSign}(M^*, S_b^*, \mathcal{R}^*)$
$b' \leftarrow \mathcal{A}^O(\sigma_{agg_b})$
if $|S_0^* \cap \mathcal{R}_{corr}| \geq |S_0^*|$, *or*
  $|S_1^* \cap \mathcal{R}_{corr}| \geq |S_1^*|$, *or*
$(M^*, S_0^*, \mathcal{R}^*) \in Q \vee (M^*, S_1^*, \mathcal{R}^*) \in Q$
  return 0
return 1 *iff* $b = b'$, *otherwise*, *return* 0

图3　匿名性游戏

$Game_{\mathcal{A}}^{Unforge}(1^{\lambda})$ 见图4。$\mathcal{A}$ 赢得该游戏的优势定义为：

$$Adv_{\mathcal{A}}^{Unforge} = \Pr[b = 1] \tag{6}$$

定义9 对于任意PPT攻击者 $\mathcal{A}$，$Adv_{\mathcal{A}}^{Unforge}$ 可忽略，则ARS方案满足普通不可伪造性。

聚合不可伪造性指攻击者即使与某个单签名集合的部分签名者合谋，也难以伪造出关于该单签名集合的有效ARS，对应攻击者 $\mathcal{A}$ 与挑战者 $\mathcal{C}$ 之间的游戏 $Game_{\mathcal{A}}^{AUnforge}(1^{\lambda})$ 见图4。$\mathcal{A}$ 赢得该游戏的优势为：

$$Adv_{\mathcal{A}}^{AUnforge} = \Pr[b = 1] \tag{7}$$

定义10 对于任意PPT攻击者 $\mathcal{A}$，$Adv_{\mathcal{A}}^{AUnforge}$ 可忽略，则ARS方案满足聚合不可伪造性。

3　ARS方案

3.1　方案技术路线

本文所提ARS方案定义在双线性群 $(q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, G, H)$ 上。方案包含单环签名和ARS两部分。单环签名生成算法中，签名者首先用私钥对消息 m 和环 $\mathcal{R}$ 进行BLS签名得到 σ_s。接下来，采用了文献[27]的 $n-1$ 阶多项式 $p(X) = 1 + \sum_{j=1}^{n-1} a_j X^j \in \mathbb{Z}_q[X]$ 技术，实现将真实签名隐藏于伪造签名之中。计算 $\tilde{\sigma}_i = p(i)\sigma_i (i \in [1,n])$，满足 $p(s)=1$，而当 $i \neq s$ 时 $p(i) = 0$，其中 s 为真实签名者。

$Game_{\mathcal{A}}^{Unforge}(1^{\lambda})$

Run $pp \leftarrow \text{Setup}(1^{\lambda})$
$O \leftarrow \{O\text{KeyGen}, O\text{Sign}, O\text{Corr}\}$
$(m^*, \mathcal{R}^*, \sigma^*) \leftarrow \mathcal{A}^O(pp)$
if : $\text{Verify}(m^*, \mathcal{R}^*, \sigma^*) = 1$ *and*
  $\mathcal{R}^* \subseteq L_{sig}$ *and*
  $\mathcal{R}^* \cap \mathcal{R}_{corr} = \varnothing$ *and*
  $(m^*, \cdot, \mathcal{R}^*) \notin Q$
  return $b = 1$
else, *return* $b = 0$

$Game_{\mathcal{A}}^{AUnforge}(1^{\lambda})$

Run $pp \leftarrow \text{Setup}(1^{\lambda})$
$O \leftarrow \{O\text{KeyGen}, O\text{Sign}, O\text{Corr}, O\text{AggreSign}\}$
$(\{m_i\}_{i \in S}^*, \mathcal{R}^*, \sigma_{agg}{}^*) \leftarrow \mathcal{A}^O(pp)$
if : $\text{AggreVerify}(\{m_i\}_{i \in S}^*, \mathcal{R}^*, \sigma_{agg}{}^*) = 1$ *and*
  $\mathcal{R}^* \subseteq L_{sig}$ *and*
  $|\mathcal{R}^* \cap \mathcal{R}_{corr}| \leq |S| - 1$ *and*
  $(\{m_i\}_{i \in S}^*, \cdot, \mathcal{R}^*) \notin Q$
  return $b = 1$
else, *return* $b = 0$

图4　不可伪造性游戏

然后，对向量$\boldsymbol{x}=(a_1,\cdots,a_{n-1},\tilde{\sigma}_1,\cdots,\tilde{\sigma}_n)\in\mathbb{Z}_q^{n-1}\times\mathbb{G}_1^n$进行同态承诺得到$P$。最后，以等式$f_i(\boldsymbol{x})=e(\tilde{\sigma}_i,H)-\sum_{j=1}^{n-1}a_j i^j e(\mathcal{H}(m,\mathcal{R}),y_i)-e(\mathcal{H}(m,\mathcal{R}),y_i)=0$作为约束条件，对承诺$P$和约束$f_i$生成基于非交互的压缩Σ协议$\Pi_c$的证明Π，输出单环签名为$(P,\Pi)$。由于诚实的BLS签名$\sigma_s$必然满足$e(\sigma_s,H)=e(\mathcal{H}(m,\mathcal{R}),y_s)$，进而满足$e(\tilde{\sigma}_s,H)=p(s)e(\mathcal{H}(m,\mathcal{R}),y_s)$，而且关于其他环成员的伪签名，也有有$e(\tilde{\sigma}_i,H)=p(i)e(\mathcal{H}(m,\mathcal{R}),y_i)(i\neq s)$成立。所以，对于$i\in[1,n]$，以$f_i(\boldsymbol{x})=e(\tilde{\sigma}_i,H)-p(i)e(\mathcal{H}(m,\mathcal{R}),y_i)=e(\tilde{\sigma}_i,H)-\sum_{j=1}^{n-1}a_j i^j e(\mathcal{H}(m,\mathcal{R}),y_i)-e(\mathcal{H}(m,\mathcal{R}),y_i)=0$作为约束条件，可使诚实环签名通过验证，并同时过滤非法环签名。

ARS部分接收k个单环签名，首先提取单签名的承诺部分生成向量$\boldsymbol{x}=(P_1,\cdots,P_k)$，把所有单签名验证条件统一转化为约束关系$F_i(\boldsymbol{x})=0$，$F_i(\boldsymbol{x})$具体形式见3.3节。进而根据定义3生成关于$\boldsymbol{x}$的类ElGamal向量承诺$\boldsymbol{P}_{agg}=Com(\boldsymbol{x},\gamma)$，在此基础上通过面向类ElGamal向量承诺的Σ协议非交互版本Π_{EG}生成向量$\boldsymbol{x}$的知识证明Π_{agg}，最终输出ARS为$(\boldsymbol{P}_{agg},\Pi_{agg})$。

验证算法，包括单环签名和ARS验证算法，即可由Π_c和Π_{EG}对应验证算法得到。

3.2 具体方案

本文所提ARS方案具体构造如下：

（1）$pp\leftarrow\text{Setup}(1^\lambda)$输入安全参数$\lambda$，输出公开参数$pp$。$pp$包含双线性群、哈希映射$\mathcal{H}:\{0,1\}^*\rightarrow\mathbb{G}_1$等。

（2）$(x_i,y_i)\leftarrow\text{KeyGen}(pp)$：随机选取私钥$x_i\in\mathbb{Z}_q$，计算公钥$y_i=x_iH$，输出公私钥对$(x_i,y_i)$。

（3）$\sigma\leftarrow\text{Sign}(m,x_s,\mathcal{R})$：输入消息$m$，签名者私钥$x_s$与环成员集合$\mathcal{R}$，生成环签名$\sigma_s=(P,\Pi)$：

1）计算$\sigma_s=x_s\cdot\mathcal{H}(m,\mathcal{R})\in\mathbb{G}_1$；

2）对于$j\in[1,n-1]$，随机选取$a_j\in\mathbb{Z}_q$，构造多项式$p(X)=1+\sum_{j=1}^{n-1}a_jX^j\in\mathbb{Z}_q[X]$，满足$p(s)=1$，$p(i)=0(i\in[1,n],i\neq s)$；

3）计算$\tilde{\sigma}_s=p(s)\cdot\sigma_s\in\mathbb{G}_1$；对于$i\in[1,n]$，$i\neq s$，设置$\tilde{\sigma}_i=0$；

4）构造向量$\boldsymbol{x}_1=(a_1,\cdots,a_{n-1})\in\mathbb{Z}_q^{n-1}$，$\boldsymbol{x}_2=(\tilde{\sigma}_1,\cdots,\tilde{\sigma}_n)\in\mathbb{G}_1^n$，$\boldsymbol{x}=(\boldsymbol{x}_1,\boldsymbol{x}_2)\in\mathbb{Z}_q^{n-1}\times\mathbb{G}_1^n$；

5）选取$h\xleftarrow{R}\mathbb{G}_T$，$\gamma\xleftarrow{R}\mathbb{Z}_q$，$\boldsymbol{g}=(g_1,\cdots,g_{n-1})\in\mathbb{G}_T^{n-1}$、$\boldsymbol{H}=(H_1\cdots,H_n)\in\mathbb{G}_2^n$，计算向量承诺$P=h\gamma+<\boldsymbol{g},\boldsymbol{x}_1>+e(\boldsymbol{x}_2,\boldsymbol{H})\in\mathbb{G}_T$；

6）构造单环签名约束$f_i:\mathbb{Z}_q^{n-1}\times\mathbb{G}_1^n\rightarrow\mathbb{G}_T$：$f_i(\boldsymbol{x})=e(\tilde{\sigma}_i,H)-\sum_{j=1}^{n-1}a_j i^j e(\mathcal{H}(m,\mathcal{R}),y_i)-e(\mathcal{H}(m,\mathcal{R}),y_i)=0$；

7）以证据$\boldsymbol{x}=(\boldsymbol{x}_1,\boldsymbol{x}_2)\in\mathbb{Z}_q^{n-1}\times\mathbb{G}_1^n$、约束关系$f_i(\boldsymbol{x})=0$为输入，运行非交互压缩Σ协议$\Pi_c$，证明向量$\boldsymbol{x}$满足约束关系$f_i(\boldsymbol{x})=0$，生成单签证明$\Pi=(a_1,b_1,\cdots,a_\mu,b_\mu,t,z)$；

8）输出环签名$\sigma_s=(P,\Pi)$。

（4）$1/0\leftarrow\text{Verify}(m,\mathcal{R},\sigma)$：运行$\Pi_c$的验证算法，若证明Π有效则输出1，接受签名；否则输出0，拒绝签名。验证等式为$F(P,a_1,b_1,\cdots,a_\mu,b_\mu,t,z)=0$。

（5）

$\sigma_{agg}/\perp\leftarrow\text{AggreSign}(\{m_i\}_{i\in S},\{\sigma_i\}_{i\in S},S,\mathcal{R})$：输入$S$中签名者的单环签名集合，假设$|S|=k$，输出一个ARS$\sigma_{agg}=(\boldsymbol{P}_{agg},\Pi_{agg})$：

1）对k个签名执行单环签名验证算法，若验证通过则继续，否则返回⊥；

2）构造承诺向量$\boldsymbol{P}=(P_1,\cdots,P_k)\in\mathbb{G}_T^k$；

3）按照本文定义3，计算聚合承诺$\boldsymbol{P}_{agg}=\begin{pmatrix}h\gamma\\g_1\gamma+P_1\\\vdots\\g_k\gamma+P_k\end{pmatrix}\in\mathbb{G}_T^{k+1}$；

4）构造聚合签名约束$F_i(P_i,a_{i1},b_{i1},\cdots,a_{i\mu},b_{i\mu},t_i,z_i)=0,i\in[1,k]$；

5）以证据$\boldsymbol{P}\in\mathbb{G}_T^k$、约束关系$F_i(P_i,a_{i1},b_{i1},\cdots,a_{i\mu},b_{i\mu},t_i,z_i)=0$为输入，通过非交互$\Pi_{EG}$协议，生成ARS证明$\Pi_{agg}=(A,t,\phi)$；

6）输出ARS签名$\sigma_{agg}=(\boldsymbol{P}_{agg},\Pi_{agg})$。

（6）$1/0\leftarrow\text{AggreVerify}(\{m_i\}_{i\in S},\mathcal{R},\sigma_{agg})$：运行$\Pi_{EG}$算法，验证$\Pi_{agg}$，若有效接受ARS，否则拒绝。

3.3 讨论

（1）关于约束条件的批量验证。采用文献[28]第5.1节中的多项式摊销技术可以通过随机线性组合将 n 个单签名约束 $f_i(\boldsymbol{x})=0$ 聚合为单个约束 $f(\boldsymbol{x})=\sum_{i=1}^{n} f_i(\boldsymbol{x})\rho^{i-1}$ 或将 k 个签名的约束 $F_i(\boldsymbol{x})=0$ 聚合为 $F(\boldsymbol{x})=\sum_{i=1}^{k} F_i(\boldsymbol{x})\rho^{i-1}$ 进行验证，仅用一次证明操作，可以至少 $1-(n-1)/q$ 或 $1-(k-1)/q$ 的概率证明所有约束成立，不增加证明者向验证者传输的数据通信成本。借助该技术，便可批量验证约束条件，进一步提升算法验证时间。

（2）关于ARS约束关系 $F_i(\boldsymbol{x})=0$ 的具体化。参与聚合的第 i 个单环签名为 $\sigma_i=(P_i,\Pi_i)$，$i\in[1,k]$，其中 P_i 为第 i 个环签名的同态向量承诺，$\Pi_i=(a_{i1},b_{i1},\cdots,a_{i\mu},b_{i\mu},t_i,z_i)$ 为该签名的压缩Σ协议证明。Π_i 的初始声明为 $y_i^{(0)}=(Com(\boldsymbol{x}),f_1(\boldsymbol{x}),...f_n(\boldsymbol{x}))=(P_i,\underbrace{0,\cdots,0}_{n})$，依据压缩Σ协议递归折叠规则，对每一轮折叠 $l\in[1,\mu]$，递归更新 $y_i^{(l)}=a_{il}+c_{il}\cdot y_i^{(l-1)}+c_{il}^2\cdot b_{il}$。经过 μ 轮递归折叠后，最终声明为 y_i^{μ}。压缩Σ协议最终合法验证等式：$\Psi_{\mu}(z_i)=c_i\cdot y_i^{(\mu)}+t_i$ 为第 μ 轮折叠后的同态映射。将等式移项整理，则聚合阶段第 i 个独立约束关系为：$F_i(P_i,a_{i1},b_{i1},\cdots,a_{i\mu},b_{i\mu},t_i,z_i)=\Psi_{\mu}(z_i)-c_i\cdot y_i^{(\mu)}-t_i=0$。其中，$P_i$ 通过初始声明 $y_i^{(0)}=(P_i,\underbrace{0,\cdots,0}_{n})$ 被引入递归计算，包含在 y_i^{μ} 中；$a_{i1},b_{i1},\cdots,a_{i\mu},b_{i\mu}$ 为各轮产生的证明脚本；t_i,z_i 为基础Σ协议中的承诺和响应。

以聚合两个单环签名为例，设 $\sigma_1=(P_1,\Pi_1)$ 和 $\sigma_2=(P_2,\Pi_2)$，其中 $\Pi_1=(a_{11},b_{11},\cdots,a_{1\mu},b_{1\mu},t_1,z_1)$，$\Pi_2=(a_{21},b_{21},\cdots,a_{2\mu},b_{2\mu},t_2,z_2)$。聚合时，首先运行单环签名验证算法，声明分别由最初的 $y_1^{(0)}=(P_1,0,\cdots,0)$ 和 $y_2^{(0)}=(P_2,0,\cdots,0)$ 递归计算得到 μ 轮声明 y_1^{μ} 和 y_2^{μ}，则两组独立约束为：$F_1=\Psi_{\mu}(z_1)-c_1y_1^{(\mu)}-t_1=F_2=\Psi_{\mu}(z_2)-c_2y_2^{(\mu)}-t_2=0$。随后构造承诺向量 $\boldsymbol{P}=(P_1,P_2)$，依据类ElGamal向量承诺算法计算聚合承诺。为避免并行证明多组等式带来额外通信开销，引入多项式摊销技术，将两组约束线性合并为一个单一约束关系等式：$F(\boldsymbol{P})=F_1+\rho\cdot F_2=(\Psi_{\mu}(z_1)-c_1y_1^{(\mu)}-t_1)+\rho\cdot(\Psi_{\mu}(z_2)-c_2y_2^{(\mu)}-t_2)=0$。若合并约束 $F(\boldsymbol{P})=0$ 成立，则 $F_1=0\wedge F_2=0$ 同时成立的概率不低于 $1-\frac{1}{q}$，满足零知识证明可靠性要求。

4 安全性分析

定理1：若BLS签名满足正确性，Π_c 和 Π_{EG} 协议满足完备性，则本文所提ARS方案满足正确性。

证明：首先对于单个环签名算法，诚实的单个环签名，必然满足以下等式：

$$\begin{aligned}e(\tilde{\sigma}_i,H)&=p(i)e(\sigma_i,H)=p(i)e(\mathcal{H}(m,\mathcal{R}),y_i)\\&=(\sum_{j=1}^{n-1}a_j i^j+1)(e(\mathcal{H}(m,\mathcal{R}),y_i)\\&=\sum_{j=1}^{n-1}a_j i^j e(\mathcal{H}(m,\mathcal{R}),y_i)+e(\mathcal{H}(m,\mathcal{R}),y_i)\end{aligned}\tag{8}$$

结合 Π_c 的完备性，单个环签名满足正确性；由于采用单个环签名的验证等式 $F_i(\cdot)=0$ 作为ARS的约束条件，每个有效的单个环签名必然满足该约束，再由 Π_{EG} 的完备性知，ARS算法满足正确性。证毕。

定理2：如果BLS签名方案满足不可伪造性，Π_c 和 Π_{EG} 协议满足特殊可靠性，则所提ARS方案满足存在不可伪造性。

证明：首先给出普通不可伪造性证明。对于单环签名方案，下面展示如果存在PPT敌手以不可忽略优势 ε 攻破普通不可伪造性，则可构造出一个以不可忽略概率伪造BLS签名的算法 $\mathcal{B}$，从而证明假设不成立。

$\mathcal{B}$ 作为BLS签名攻击者，随机选取索引 s 嵌入挑战公钥 y_s，模拟ARS挑战者与 $\mathcal{A}$ 进行交互：

（1）当 $\mathcal{A}$ 进行 $O\text{KeyGen}(s)$ 询问时，由于没有对应私钥，$\mathcal{B}$ 将 $(s,\cdot,y_s)$ 添加至 L_{sig}，y_s 给环 $\mathcal{R}$，并将 y_s 返回给A。关于其他成员的公钥询问，正常回应。

（2）当 $\mathcal{A}$ 进行 $O\text{Sign}(m,s,\mathcal{R})$ 询问时，$\mathcal{B}$ 调用BLS签名预言机发起查询，获取 s 关于消息 m 的BLS签名，其余步骤与Sign算法相同，得到ARS签名并返回；同时，将 $(m,s,\mathcal{R})$ 添加到 Q 中。关于其他成员的签名询问，$\mathcal{B}$ 正常回应。

（3）当 $\mathcal{A}$ 进行 $O\text{Corr}(s)$ 询问时，$\mathcal{B}$ 中止；关于其他成员的腐化询问，$\mathcal{B}$ 正常回应。

分析。若经过上述自适应预言机查询后，$\mathcal{A}$ 输

出有效的伪造单环签名。进一步假设该伪造签名对应的环集合包含挑战公钥y_s。伪造签名的核心在于伪造出有效的零知识证明，对于单环签名，意味着攻击者能成功伪造一个有效的压缩Σ协议证明Π。由Π的特殊可靠性，则$\mathcal{B}$能提取出一个不同于证据$\boldsymbol{x}$的向量$\boldsymbol{x'}=(a'_1,\cdots a'_{n-1},S_1,\cdots S_n)$满足$f_i(\boldsymbol{x'})=0$。则有$e(S_i,H)=(1+\sum_{j=1}^{n-1}a'_j i^j)e(\mathcal{H}(m,\mathcal{R}),y_i)$，换言之有$e(\mathcal{H}(m,\mathcal{R}),y_i)=(1+\sum_{j=1}^{n-1}a'_j i^j)^{-1}e(S_i,H)(i\in[1,n])$，即攻击者成功伪造一个关于消息$m$的BLS的签名$\sigma_{\mathrm{BLS}}=(1+\sum_{j=1}^{n-1}a'_j s^j)^{-1}S_s$。

概率分析。$\mathcal{B}$伪造成功需要以下条件成立：

（1）$\mathrm{Succ}_{\mathcal{A}}$：$\mathcal{A}$在环签名游戏成功伪造；

（2）Event_1：伪造环$\mathcal{R}^*$包含挑战公钥y_s；

（3）Event_2：$\mathcal{A}$在所有腐败询问中从未请求y_s的私钥；

（4）Event_3：NIZK提取器成功提取有效证据，

$\Pr[\mathcal{B}成功]=\Pr[\mathrm{Succ}_{\mathcal{A}}\wedge\mathrm{Event}_1\wedge\mathrm{Event}_2\wedge\mathrm{Event}_3]$。

由于有以下关系存在：

$$\Pr[\mathrm{Succ}_{\mathcal{A}}\wedge\mathrm{Event}_1\wedge\mathrm{Event}_2]=\Pr[\mathrm{Succ}_{\mathcal{A}}\wedge\mathrm{Event}_1\wedge\mathrm{Event}_2\wedge\mathrm{Event}_3]+\Pr[\mathrm{Succ}_{\mathcal{A}}\wedge\mathrm{Event}_1\wedge\mathrm{Event}_2\wedge\neg\mathrm{Event}_3]\leqslant\Pr[\mathcal{B}成功]+\Pr[\neg\mathrm{Event}_3]$$

，所以

$$\Pr[\mathcal{B}成功]\geqslant\Pr[\mathrm{Succ}_{\mathcal{A}}\wedge\mathrm{Event}_1\wedge\mathrm{Event}_2]-\Pr[\neg\mathrm{Event}_3]=\Pr[\mathrm{Succ}_{\mathcal{A}}]\cdot\Pr[\mathrm{Event}_1\wedge\mathrm{Event}_2\backslash\mathrm{Succ}_{\mathcal{A}}]-\Pr[\neg\mathrm{Event}_3]$$

。

由Π_c的特殊可靠性:$\Pr[\neg\mathrm{Event}_3]=\mathrm{negl}(\lambda)$假设$Q_u$是系统最大用户/公钥数量,能够得到下限$\Pr[\mathrm{Event}_1\wedge\mathrm{Event}_2\backslash\mathrm{Succ}_{\mathcal{A}}]\geqslant\frac{1}{Q_u}$，则有概率关系

$$\Pr[\mathcal{B}成功]\geqslant\varepsilon\cdot\frac{1}{Q_u}-\mathrm{negl}(\lambda)$$

聚合签名不可伪造性，证明与上述过程类似，区别在于攻击者能成功伪造一个ARS，则他能成功伪造一个有效证明Π_{agg}，由Π_{agg}具有特殊可靠性，则存在一个算法$\mathcal{Y}$输出一个向量$\boldsymbol{P'}=(P_1',\cdots,P_k')$满足$F_i(\boldsymbol{P'})=0$。而$F_i$为$\Pi_c$对应的验证等式，具有同态性与抗伪造性，只有凭借真实证据产生的证明才能通过验证，即使攻击者与单个签名集合中的部分签名者合谋，获取部分证据，只要不拥有全部证据$\boldsymbol{P}$，输出向量$\boldsymbol{P'}$满足$F_i(\boldsymbol{P'})=0$概率就是可忽略的。证毕。

定理3：若BLS签名满足可证明安全性，$(\mathbb{Z}_q,\mathbb{G}_1)$和$\mathbb{G}_T$向量承诺满足隐藏性，$\Pi_c$和$\Pi_{EG}$协议满足HVZK性，则所提ARS方案满足匿名性。

证明：首先给出方案满足聚合匿名性的证明。假设存在PPT攻击者能够以不可忽略的优势攻破方案的聚合匿名性，我们构造一系列不可区分游戏，并由此得出假设与同态承诺绑定性相矛盾的结论。

Game1:

1:$\mathcal{C}$生成$pp\leftarrow\mathrm{Setup}(1^\lambda)$给$\mathcal{A}$;

2:$\mathcal{A}$自适应地进行查询OKeyGen OSign OAggreSign和OCorr预言机,$\mathcal{C}$都据实回答;

3:$\mathcal{A}$输出$(m^*,S_0^*,S_1^*,\mathcal{R}^*)$;

4:$\mathcal{C}$检查若$|S_0^*\cap\mathcal{R}_{corr}|\geqslant|S_0^*|\vee|S_1^*\cap\mathcal{R}_{corr}|\geqslant|S_1^*|$或$(m^*,S_0^*,\mathcal{R}^*)\in Q\vee(m^*,S_1^*,\mathcal{R}^*)\in Q$则中止;否则设置$b=0$,生成$\sigma_0\leftarrow O\mathrm{AggreSign}(m^*,S_0^*,\mathcal{R}^*)$;

5:$\mathcal{A}$给出b的猜测

Game2:

1:$\mathcal{C}$生成$pp\leftarrow\mathrm{Setup}(1^\lambda)$给$\mathcal{A}$;

2:$\mathcal{A}$自适应地进行查询OKeyGen OSign OAggreSign和OCorr预言机,$\mathcal{C}$都据实回答;

3:$\mathcal{A}$输出$(m^*,S_0^*,S_1^*,\mathcal{R}^*)$;

4:$\mathcal{C}$检查若$|S_0^*\cap\mathcal{R}_{corr}|\geqslant|S_0^*|\vee|S_1^*\cap\mathcal{R}_{corr}|\geqslant|S_1^*|$或$(m^*,S_0^*,\mathcal{R}^*)\in Q\vee(m^*,S_1^*,\mathcal{R}^*)\in Q$则中止;否则设置$b=1$,生成$\sigma_1\leftarrow O\mathrm{AggreSign}(m^*,S_1^*,\mathcal{R}^*)$;

5:$\mathcal{A}$给出b的猜测

分析。假设$\mathcal{A}$以不可忽略的概率区分σ_0和σ_1，对比$\sigma_0=(\mathrm{Com}(S_0),\Pi_0)$和$\sigma_1=(\mathrm{Com}(S_1),\Pi_1)$，由$\Pi_{EG}$零知识性，$\Pi_0$和$\Pi_1$的可区分概率为$negl(\lambda)$，意味着$\mathcal{A}$区分$\mathrm{Com}(S_0)$和$\mathrm{Com}(S_1)$概率不可忽略，这与同态承诺隐藏性相矛盾。假设不成立，方案满足聚合匿名性。

单环签名匿名性依赖BLS签名的可证明安全性、$(\mathbb{Z}_q,\mathbb{G}_1)$向量承诺方案的隐藏性和$\Pi_c$的HVZK性。其证明过程与上述过程类似，此处不再赘述。证毕。

相较于Tong等人提出的ARS方案[6]，本文方案聚合过程无需可信第三方参与，任意用户均可完

整签名聚合及验证操作。聚合过程执行的是单环签名的模糊验证，单环签名对应的签名者身份信息不会泄露给聚合者，可真正实现聚合匿名性。

5　性能评估

5.1　理论分析

本节给出环签名领域的主流研究成果与本文方案的理论性能对比。鉴于各方案所采用的底层密码技术存在差异，难以对各方案的性能进行精准定量对比。因此本文参照相关文献的通用分析方式，从复杂度层面进行横向对比。

表1　k个签名通信与计算开销对比分析

方案	签名长度	签名时间	验证时间	可聚合性
Duan[18]	$O(kn)$	$O(kn)$	$O(kn)$	×
Malavolta[9]	$O(kn)$	$O(kn)$	$O(kn)$	×
Qin[3]	$O(k)$	$O(kn)$	$O(kn)$	×
Groth[8]	$O(klogn)$	$O(knlogn)$	$O(kn)$	×
DualRing[12]	$O(klogn)$	$O(kn)$	$O(kn)$	×
RingCT3.0[16]	$O(klogn)$	$O(kn)$	$O(kn)$	×
Omniring[17]	$O(klogn)$	$O(kn)$	$O(kn)$	×
Teng[19]	$O(logkn)$	$O(kn)$	$O(k+n)$	□
Tong[6]	$O(logn)$	$O(kn)$	$O(n)$	□
本文	$O(k)$	$O(kn)$	$O(k)$	□

当单个签名数量为k，环成员总数为n时，ARS方案与同类方案的签名长度、签名生成时间、验证时间及可聚合性等方面的理论对比见表1。分析过程不计入公共参数、消息等传输开销，忽略基本代数运算、常数损耗及哈希运算等计算开销，关注承诺及证明尺寸等传输开销，及承诺、证明生成与验证等核心计算。本文方案将k个单签名聚合为1个ARS传输与验证，每个ARS包含k+1个承诺值与固定规模证明数据，签名长度复杂度为$O(k)$。具体而言ARS包括$\boldsymbol{P}_{agg}$和Π_{agg}两部分，$\boldsymbol{P}_{agg}$含$k+1$个群元素，即$\left|\boldsymbol{P}_{agg}\right|=(k+1)\left|\mathbb{G}_T\right|$，$\Pi_{agg}$对应元素规模为$\left|\Pi_{EG}\right|=\left|\mathbb{G}_T\right|+|\mathbb{H}|+\left|\mathbb{Z}_q\right|$。则本文ARS的总长度为$(k+2)\left|\mathbb{G}_T\right|+|\mathbb{H}|+\left|\mathbb{Z}_q\right|$。签名生成方面，所提方案的单环签名生成时间复杂度为$O(n)$，k个签名初生成总开销为$O(kn)$；聚合阶段依托单签名构造承诺向量，Π_{EG}协议生成证明，运算规模与k线性相关，合并后方案整体签名生成复杂度为$O(kn)$。本文方案验证阶段无需拆解底层环结构，仅对长度为k的承诺向量运算，验证复杂度为$O(k)$。RingCT3.0和DualRing在批量验证k个签名时，时间复杂度达$O(klogn)$，而本文方案签名长度为$O(k)$均优于两方案；Teng和Tong的ARS方案，虽然签名长度呈对数级，但为了确保匿名性，一般来讲，$n \gg k$，所以本文方案签名长度仍具有一定优势，并且本文方案大幅缩短了多签名场景下的验证时间。需要说明，Qin等人[3]基于双线性配对与累加器技术提出了常数大小环签名方案，k个签名总长度与本文聚合签名长度相同，均为$O(k)$。但与大多数基于累加器的环签名相似，Qin等人[3]方案签名验证始终绕不开“遍历环中的公钥”，因而k个环签名验证时间为$O(kn)$，且方案需要可信第三方参与。Tong方案生成签名时间复杂度原文是$O(n)$，这是由于单签名生成算法中采用密钥同态签名技术，导致单签名的生成时间为常数，但分析发现该方案不满足匿名性，若弥补方案的匿名性缺陷，至少需要生成签名复杂度提升至$O(kn)$。

综上，方案在签名长度、验证效率方面具有较高效率，且不需可信第三方参与，综合性能更适配大规模环和批量签名核验的区块链交易等业务场景。

5.2　实验对比

为验证本文ARS方案在不同参数规模下的通信开销与计算效率，本文基于 Python 构建实验模型，并在 Windows 11 操作系统、Intel Core i7 处理器和16 GB 内存环境下完成实验。实验主要从签名尺寸和签名验证时间两个方面进行对比。由于环签名方案的性能主要受环大小n和签名者数量k的影响，本文采用控制变量法进行实验，即在分析签名者数量k对性能的影响时固定环大小n，在分析环大小n对性能的影响时固定签名者数量k。

具体而言，在签名尺寸实验中，本文选取2组典型区块链交易参数：环大小n=1024，签名者数量$k\in[1,5]$；签名者数量k=4，环大小$n\in[2^{10},2^{14}]$。其中，$k=1$为单签名者情形，用于作为多签名者聚合场景的基准；$k\in[1,5]$契合区块链实际应用场景，待聚合签名的数量与区块内交易数量一一对应，而单个区块承载的交易数通常为几笔至数十笔。对于环大小，本文选取$n\in[2^{10},2^{14}]$，该范围

覆盖了较大规模匿名集合场景，能够充分体现环大小增加时各方案通信开销。在验证时间实验中，本文设置 2 种典型参数进行实验：环大小 $n = 1024$，签名者数量 $k \in [1,5]$；签名者数量 $k = 4$，环大小 $n \in [2^6,2^{10}]$，这是因为当 n 过大时，部分对比方案验证时间急剧增加，图像对比效果过于悬殊。

本文对实验中的每组参数设置均独立重复运行 100 次。由于各组实验结果波动较小，且本文主要关注不同参数规模下各方案签名尺寸和验证时间的整体变化趋势。因此，验证时间实验以平均值反映不同参数规模下各方案的整体变化趋势，以保证实验结果展示的清晰性，例如：当 n=1024，k=1 时，本文方案的验证时间在[49ms，86ms]波动，均值为 70ms；当 n=64，k=4 时，本文方案的验证时间在[93ms,140ms]范围波动，均值为 121ms。

（1）签名尺寸方面，图 5 展示了本文 ARS 方案与其他方案签名长度的对比结果。其中，图 5（a）给出了当环大小固定为 1024 时，各方案签名长度随签名者数量增长的变化趋势；图 5（b）展示了当签名者数量固定为 4 时，签名尺寸随环大小 n 增长的变化趋势。由图 5(a)可以看出，当环大小固定为 1024 时，随着签名者数量 k 的增加，各方案的签名长度均呈现不同程度的增长。相比之下，本文方案的签名长度增长较为平缓，整体签名尺寸始终保持在较低水平，说明本文方案能够有效控制签名者数量增加带来的通信开销。由图 5(b)可以看出，当签名者数量固定为 $k = 4$ 时，随着环大小 n 从 2^{10} 增加至 2^{14}，部分对比方案的签名长度呈现明显增长趋势，说明这些方案的签名尺寸与环成员数量存在较强相关性。相比之下，本文方案的签名长度随环大小增加变化较小，表明本文方案在大规模环场景下具有较好的存储效率和通信效率。

（2）验证时间方面，本文采用 Charm-Crypto 框架进行实验，安全参数设置 256 位。实验结果如图 6 所示。图 6(a)展示当环大小固定为 $n = 1024$ 时，各方案验证时间随签名者数量 k 增长的变化情况；图 6(b)展示当签名者数量固定为 $k = 4$ 时，各方案验证时间随环大小 n 增长的变化趋势。由图 6(a)可以看出，当环大小固定为 $n = 1024$ 时，随着签名者数量 k 的增加，各方案的验证时间均呈现一定增长趋势。RingCT3.0 和 DualRing 方案的验证时间随 k 增长较为明显，说明其验证开销与签名者数量具有较强相关性。相比之下，本文方案的验证时间增长较为缓慢，说明本文方案在多签名者聚合场景下具有较好的验证效率。由图 6(b)可以看出，当签名者数量固定为 $k = 4$ 时，随着环大小 n 的增加，部分对比方案的验证时间明显上升。相比之下，本文方案的验证时间随环大小增加变化较小，整体趋势较为平缓，表明本文方案的验证效率基本不随环大小 n 的增加而显著变化。因此，本文方案在大规模环体、大批量签名核验和高频交易等应用场景中具有更好的综合适用性。

6　结束语

本文提出一种基于同态向量承诺和压缩 Σ 协议的聚合环签名方案，通过引入 BLS 签名、$(\mathbb{Z}_q,\mathbb{G}_1)$ 向量承诺、面向 $(\mathbb{Z}_q,\mathbb{G}_1)$ 向量承诺的压缩 ∑ 协议、

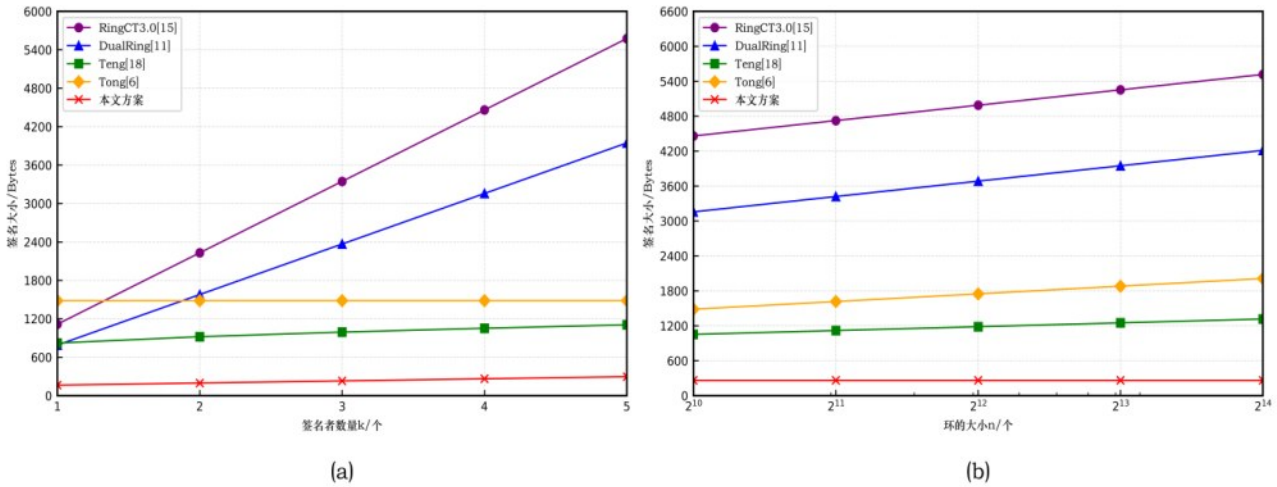

图 5　签名长度对比分析

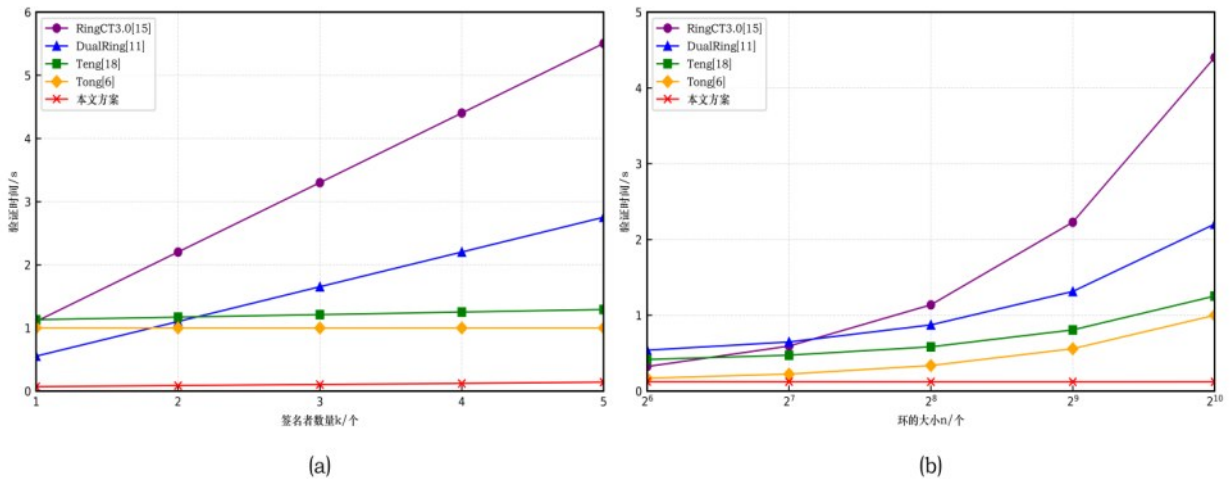

图6　签名验证时间对比分析

类ElGamal向量承诺及面向类ElGamal向量的∑协议，实现了无可信第三方条件下的多个环签名的高效压缩与统一验证。该方案具备匿名性、不可伪造性，并且具备优秀的聚合验证性。理论分析与实验结果表明，本文方案在多用户、大规模环和批量签名验证场景下能够有效降低验证时间和通信负担，更适合区块链隐私交易、匿名投票等高频应用环境。

邵震杰（2001− ），男，西安理工大学硕士生，主要研究方向为环签名、网络安全。

汤永利（1972− ），男，博士，河南理工大学教授，主要研究方向为密码学、抗量子密码。

王一川（1983− ），男，博士，西安理工大学教授，主要研究方向为系统脆弱性分析、网络安全。

黑新宏（1976− ），男，博士，西安理工大学教授，主要研究方向为网络安全、轨道交通安全。

参考文献：

[1] Rivest R L, Shamir A, Tauman Y. How to leak a secret[C]//International Conference on the Theory and Application of Cryptology and Information Security. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001: 552-565.

[2] Shen N. Ring signature confidential transactions for monero[J]. Tech. rep. Cryptology ePrint Archive, Report 2015/1098, 2015.

[3] Qin M J, Zhao Y L, Ma Z J. Practical constant-size ring signature[J]. Journal of Computer Science and Technology, 2018, 33(3): 533-541.

[4] Backes M, Döttling N, Hanzlik L, Kluczniak K, Schneider J. Ring signatures: logarithmic-size, no setup—from standard assumptions[C]//Advances in Cryptology—Eurocrypt 2019. Cham: Springer International Publishing, 2019: 281-304.

[5] Boneh D, Gentry C, Lynn B, et al. Aggregate and verifiably encrypted signatures from bilinear maps[C]//International Conference on the Theory and Applications of Cryptographic Techniques. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003: 416-432.

[6] Tong X, Zhou J, Cao Z, et al. A ring signature with aggregation for ensuring privacy in blockchain transactions[J]. IEEE Internet of Things Journal,2025, 12(12): 21001-21015.

[7] Attema T, Cramer R, Rambaud M. Compressed Σ-protocols for bilinear group arithmetic circuits and application to logarithmic transparent threshold signatures[C]//International Conference on the Theory and Application of Cryptology and Information Security. Cham: Springer International Publishing, 2021: 526-556.

[8] Groth J, Kohlweiss M. One-out-of-many proofs: Or how to leak a secret and spend a coin[J]. Annual International Conference on the Theory and Applications of Cryptographic Techniques, 2015: 253-280.

[9] Malavolta G, Schröder D. Efficient ring signatures in the standard model [J]. International Conference on the Theory and Application of Cryptology and Information Security, 2017: 128-157.

[10] Liu J K, Wei V K, Wong D S. Linkable spontaneous anonymous group signature for ad hoc groups[J]. Australasian Conference on Information Security and Privacy, 2004: 325-335.

[11] Fujisaki E, Suzuki K. Traceable ring signature[J]. International Workshop on Public Key Cryptography, 2007: 181-200.

[12] Yuen T H, Esgin M F, Liu J K, et al. DualRing: Generic construction of ring signatures with efficient instantiations[C]//Annual International Cryptology Conference. Cham: Springer International Publishing, 2021: 251-281.

[13] Feng H, Liu J, Li D, et al. Traceable ring signatures: general framework and post-quantum security[J]. Designs, Codes and Cryptography, 2021, 89(6): 1111-1145.

[14] Ye Q, Lang Y, Guo H, et al. Efficient lattice-based traceable ring signature scheme with its application in blockchain[J]. Information Sciences, 2023, 648: 119536.

[15] Sun S F, Au M H, Liu J K, et al. Ringct 2.0: A compact accumulator-based (linkable ring signature) protocol for blockchain cryptocurrency monero[C]//European Symposium on Research in Computer Security. Cham: Springer International Publishing, 2017: 456-474.

[16] Yuen T H, Sun S, Liu J K, et al. Ringct 3.0 for blockchain confidential transaction: Shorter size and stronger security[C]//International Conference on Financial Cryptography and Data Security. Cham: Springer International Publishing, 2020: 464-483.

[17] Lai R W F, Ronge V, Ruffing T, et al. Omniring: Scaling private payments without trusted setup[C]//Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security. 2019: 31-48.

[18] Duan J, Zheng S, Wang W, et al. Concise RingCT protocol based on linkable threshold ring signature[J]. IEEE Transactions on Dependable and Secure Computing, 2024, 21(5): 5014-5028.

[19] Teng D, Yao Y, Huang C. Optimizing signature space performance in privacy-enhanced blockchains: novel ring signature solutions[J]. EURASIP Journal on Information Security, 2025, 2025(1): 6.

[20] Attema T, Cramer R, Rambaud M. Compressed sigma protocol theory and practical application to plug &play secure algorithmics[C]//Cryptology ePrint Archive,Report 2020/152, 2020.

[21] Galbraith S D, Paterson K G, Smart N P. Pairings for cryptographers [J]. Discrete Applied Mathematics, 2008, 156(16): 3113-3121.

[22] Rackoff C, Simon D R. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack[C]//Annual International Cryptology Conference. Berlin, Heidelberg: Springer Berlin Heidelberg, 1991: 433-444.

[23] Catalano D, Fiore D. Vector commitments and their applications[C]// International Workshop on Public Key Cryptography. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013: 55-72.

[24] Abe M, Fuchsbauer G, Groth J, et al. Structure-preserving signatures and commitments to group elements[J]. Journal of Cryptology, 2016, 29(2): 363-421.

[25] Lai R W F, Malavolta G, Ronge V. Succinct arguments for bilinear group arithmetic: Practical structure-preserving cryptography[C]//Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security. 2019: 2057-2074.

[26] ElGamal T. A public key cryptosystem and a signature scheme based on discrete logarithms[J]. IEEE Transactions on Information Theory, 1985, 31(4): 469-472.

[27] Attema T, Cramer R, Fehr S. Compressing proofs of k-out-of-n partial knowledge[C]//Annual International Cryptology Conference. Cham: Springer International Publishing, 2021: 65-91.

[28] Attema T, Cramer R. Compressed-protocol theory and practical application to plug & play secure algorithmics[C]//Annual International Cryptology Conference. Cham: Springer International Publishing, 2020: 513-543.

叶青（1981− ），女，博士，西安理工大学副教授，主要研究方向为密码学、网络安全。