面向电子投票的匿名可监管聚合环签名算法

曾捷伦[1]，陈晶[1]，何琨[1]，加梦[2]，吕岚曦[1]，杜瑞颖[1]
（1. 武汉大学国家网络安全学院，湖北 武汉 430040；2. 香港理工大学电子计算学系，香港 999077）

摘要：针对电子投票场景中计票验证效率不足以及对多种投票模式支持不完善的问题，提出了支持多种投票模式的匿名可监管聚合环签名算法，在保障投票者隐私的同时确保操作合法性。通过环签名技术实现去中心化配置与验证者身份隐藏，防止验证过程被追踪或操控。通过聚合机制，将多份投票封装为紧凑验证对象，提高计票验证效率。为监管同一签名者的投票次数，引入链接机构并设计可链接标签以支持单投与多投。安全分析与实验结果表明，所提算法在保障匿名监管的同时能有效降低计算开销，其中链接阶段相较k-LRS方案平均减少约98.29%的时间开销。

关键词：电子投票；环签名；身份隐藏；可链接
中图分类号：TP309
文献标志码：A
doi：10.11959/j.issn.1000−436x.TXXB260133

Anonymous and accountable aggregate ring signature algorithm for electronic voting

Zeng Jielun[1], Chen Jing[1], He Kun[1], Jia Meng[2], Lyu Lanxi[1], Du Ruiying[1]
1. School of Cyber Science and Engineering, Wuhan University, Wuhan 430040, China
2. Department of Computing, The Hong Kong Polytechnic University, Hong Kong 999077, China

Abstract: To address the low tally-verification efficiency and insufficient support for diverse voting modes identified in electronic voting scenarios, an anonymous and accountable aggregate ring signature algorithm supporting multiple voting modes was proposed. The proposed construction was designed to preserve voter privacy while ensuring the legitimacy of voting operations. Ring signature technology was employed to enable decentralized setup and hide verifier identities, reducing the risk of tracing or manipulation during verification, while an aggregation mechanism was introduced to compress multiple ballots into compact verification objects to improve tally-verification efficiency. To regulate the number of votes cast by the same signer, a linking authority was introduced together with linkable tags, enabling support for both single-vote and multiple-vote settings. Security analysis and experimental results show that the proposed algorithm achieves a good balance between anonymous regulation and efficiency, and reduces the time overhead in the linking phase by about 98.29% on average compared with the k-LRS scheme.

Key words: electronic voting, ring signature, identity hiding, linkability

0 引言

自20世纪70年代起，电子投票凭借其计票高效等优势，在从单选投票到累积投票等各类场景中得到了广泛应用。标准的电子投票系统通常由选民、投票人、注册中心、计票中心与监管中心构成。在该系统下，隐私保护是构建选举公信力的核心，主

收稿日期：2026−03−20；修回日期：2026−04−11
通信作者：陈晶，chenjing@whu.edu.cn
基金项目：国家密码科学基金资助项目（No.2025NCSF02027）
Foundation Item：The National Cryptologic Science Fund of China (No.2025NCSF02027)

要涵盖身份隐私、内容隐私与关联隐私。其核心目标是确保投票人的身份、投票内容及二者之间的逻辑关联对任何未经授权的第三方均保持不可见。

随着投票方式复杂化及投票规模扩大，现有方案依据隐私暴露程度可划分为完全隐藏计票与部分隐藏计票。

完全隐藏计票要求选票内容及中间计票信息始终保持密文状态，包括计票机构在内的任何实体均无法获得候选人的中间票数或票数变化趋势。此概念最早由Cohen[1]提出，通常依托同态加密[2-5]、混合网络[6-8]等技术保证最高级别的隐私保护与匿名性。Ordinos系统[2]作为首个可验证的完全计票隐藏投票方案，利用同态聚合与多方安全计算评估密文对应的结果函数，在不揭露中间计票信息的条件下公布选举结果。针对多数判决投票提案，文献[9]通过同态加密、多方安全计算与零知识证明的协同构造，确保计票过程的完全机密性。针对现有方案在应对排序投票等复杂场景时存在的性能瓶颈，Wabartha等[10]结合同态加密与多方安全计算，严格隐藏计票过程与中间结果，仅公开最终席位分配及总投票人数。高改梅等[11]提出了全同态加密与区块链技术结合的电子投票方案，旨在保障投票隐私的同时提升系统的可验证性与防篡改能力。

尽管完全隐藏计票提供了最高安全级别的隐私保护，但其在性能开销与监管效能间存在内生矛盾。此类方案高度依赖高开销技术，计算与通信开销成本较高，在增强匿名性的同时削弱了投票行为的追踪与约束能力，难以兼顾系统隐私性与可监管性。

部分隐藏计票对选举数据的特定部分加密，旨在兼顾系统的可监管性与隐私保护。计票过程及结果数据的可见性，使计票中心或公众能监管投票流程。但部分信息的泄露容易诱发关联分析风险，如典型的意大利攻击[12-13]可通过投票模式推断选民身份。针对意大利攻击，Jamroga等[13]提出了风险限制计票与风险限制验证机制，在保证结果置信度的条件下对计票与验证过程实施监督。Adida等[14]提出的Helios系统是首个在真实选举中成功应用的开放审计电子投票平台。该系统使用同态加密与零知识证明技术保护投票人匿名性，使用开放审计公告板机制实现对投票过程与计票结果的可监管性。为适配更复杂的选举规则，Ramchen等[15]采用多方安全计算与同态加密技术保障选票隐私，并结合可验证多方安全计算实现对复杂计票过程的公开监管与结果审查。Yang等[16]构建了PriScore系统，依托双零知识证明技术保障选票隐私，允许任何验证者依据链上公开信息对同一投票人的重复多投行为进行链接，从而确保投票过程的可监管性。为提升验证效率，Harn等[17]与Wang等[18]分别引入盲签名与聚合签名技术，并结合智能合约实现高效的匿名认证与加密计票保护隐私。Kryvos方案[19]通过轻量级同态承诺提升复杂规则下的验证效率。k-LRS方案[20]构建了基于k次受限环签名的匿名监管系统，通过可链接性约束重复多投行为。然而，该方案的监管判定高度依赖选票间的代数关联，验证存在滞后性。此外，验证开销随次数上限呈非线性增长，在大规模选举中存在性能瓶颈。

尽管部分隐藏计票在隐私保护与公开可验证之间实现了部分平衡，但仍有局限：隐藏边界的模糊性导致明文信息易受推断攻击，诱发身份泄露风险。为兼顾可监管与隐私保护需求，方案需引入额外的多方交互与验证机制，加重计算与通信负载。

综上所述，现有电子投票方案在匿名性、可监管性与系统效率之间仍难以兼顾。完全隐藏计票虽然提供了最强内容隐私与过程匿名性，却因计票过程不可见导致监管失能；部分隐藏计票虽提升了审计透明度，但其现有的可监管机制仍面临监管即时性与高代数开销的问题。针对上述问题，本文面向部分隐藏计票提出了一种基于环签名的匿名可监管电子投票方案，在确保投票人匿名性的同时实现对重复投票行为的有效监管，并兼顾系统的计算效率与可扩展性。本文的主要贡献如下。

（1）针对现有方案监管滞后性与验证开销随阈值k呈非线性增长的问题，构建了结合环签名与聚合机制的匿名监管架构。通过构造可链接标签，实现监管判定与次数k的解耦，将判定复杂度从$O\big(f(k)\big)$降至$O(1)$。

（2）通过理论分析与实验评估验证了本文方案的安全性与高效性，结果表明该方案在匿名性、可监管性与性能间实现了有效平衡，具备实用性。

1　预备知识

1.1　环签名

环签名方案[21]是多项式时间算法，包含3个算法(RGen,RSign,RVerify)，分别用于为用户生成密

钥、签名消息与验证签名与消息。具体形式如下。

（1）$\mathrm{RGen}(1^{\tau}) \to (\mathrm{SK},\mathrm{PK})$：输入安全参数$\tau$，输出私钥SK与公钥PK。

（2）$\mathrm{RSign}_{\mathrm{SK}}(M,R) \to \sigma$：输入消息$M$与环$R=(\mathrm{PK}_1,\cdots,\mathrm{PK}_n)$，输出在消息$M$上对应的环签名$\sigma$。

（3）$\mathrm{RVerify}_R(M,\sigma) \to 0/1$：输入消息$M$与对应的环签名$\sigma$，若签名验证通过则输出1，否则输出0。

需要注意的是，环的大小应至少为2，即$|R| \geqslant 2$，且环中的公钥具备唯一性。

1.2　SM2公钥加密算法

SM2椭圆曲线公钥密码算法[22]是我国公钥密码算法标准[23]，主要包括数字签名算法、密钥交换协议和公钥加密算法。其中公钥加密算法由3个算法组成$(\mathrm{Gen},\mathrm{Enc},\mathrm{Dec})$，具体形式如下。

（1）$\mathrm{Gen}(1^{\tau}) \to (\mathrm{sk},\mathrm{pk})$：输入安全参数$\tau$，输出私钥sk与公钥pk。

（2）$\mathrm{Enc}(m,\mathrm{pk}) \to c$：输入消息明文$m$与公钥pk，输出对应的密文$c$。

（3）$\mathrm{Dec}(c,\mathrm{sk}) \to m$：输入消息密文$c$与私钥sk，输出消息明文$m$。

1.3　困难性假设

（1）DDH（decisional Diffie-Hellman）问题[24]。设$\mathbb{G}$为阶为素数p的乘法循环群，生成元$g \in \mathbb{G}$。给定(g,g^a,g^b,T)，其中$a,b \in \mathbb{Z}_p$，且$T \in \mathbb{G}$，判定T是否满足$T=g^{ab}$。

DDH困难性假设是指不存在多项式时间算法能以不可忽略的概率完成上述判定。

（2）离散对数（discrete logarithm, DL）问题[25]。设$\mathbb{G}$为阶为素数p的乘法循环群，生成元$g \in \mathbb{G}$。给定(g,g^a)，其中$a \in \mathbb{Z}_p$，要求计算出指数a。

DL困难性假设是指不存在多项式时间算法能以不可忽略的概率从给定的(g,g^a)中恢复指数a。

2　系统模型

2.1　系统架构

匿名可监管的投票系统架构如图1所示。

（1）选民：指若干未具备投票资格的实体集合，选民需在注册中心进行注册才具备投票权。

（2）投票人：指若干具备投票资格的实体集合，具备投票权。

（3）注册中心：指承担身份注册和密钥分发的单个实体。

（4）计票中心：指验证单个选票合规性与次数

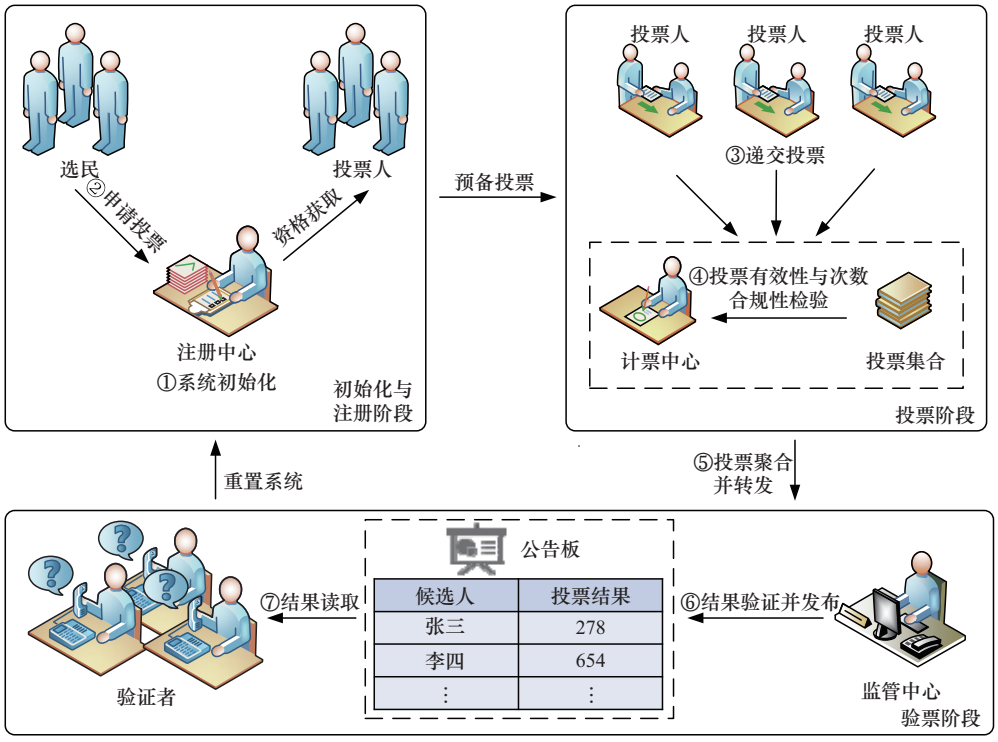

图1　匿名可监管的投票系统架构

有效性的实体，对合规投票进行聚合与链接操作。

（5）监管中心：指验证聚合投票合规性的单个实体，验证结束后将结果发布于公告板。

在初始化与注册阶段，注册中心完成系统初始化并生成相关参数。选民向注册中心提出注册请求，经资格审核后获得公私钥对与注册凭证成为投票人。在投票阶段，投票人依据规则使用其私钥完成签名操作并将选票递交至计票中心。计票中心对投票的合法性与投票次数进行校验，通过者纳入计票并聚合，将聚合投票结果发送至监管中心。经监管中心确认的投票结果将发布于公告板。

2.2　威胁模型

为刻画本文方案在选择消息攻击下的安全性，设存在概率多项式敌手 $\mathcal{A}$，敌手为投票系统中的外部人员或恶意投票人，通过访问查询接口与系统进行交互，能力描述如下。

（1）注册与公钥操控：$\mathcal{A}$ 可发起 $\mathrm{Register}(\mathrm{id})$ 创建任意数量身份，并可通过 $\mathrm{ReplacePK}(\mathrm{id},\mathrm{pk}')$ 在注册或投票前替换任意公开密钥为其指定值。$\mathcal{A}$ 可获得用于构造环签名所需的公钥列表。

（2）选择消息签名与聚合查询：$\mathcal{A}$ 可对任意消息发起 $\mathrm{Sign}(\mathrm{id},m)$，以获得对应的环签名，并可请求计票中心对一组合法签名执行聚合查询以观察聚合输出结果。

（3）伪造与非法投递：$\mathcal{A}$ 试图伪造签名或绕过链接与次数检验机制，使同一投票主体对应的有效计票次数超过系统预设的上限 k。

2.3　安全目标

在上述威胁模型下，本文方案需要同时满足以下安全目标。

（1）匿名性：对于任意不掌握链接私钥的概率多项式敌手，给定有效环签名，其判定该签名由环中某一特定投票人生成的成功概率不高于随机猜测。

（2）不可伪造性：敌手不能使系统接受来自未合法注册身份的投票。

（3）次数受限性：任意概率多项式敌手不能使有效计票次数超过预设上限 k。

若敌手 $\mathcal{A}$ 破坏任一安全目标的成功概率可忽略，则称方案在该安全模型下安全。

本文从匿名性、不可伪造性和次数受限性 3 个方面定义挑战者 $\mathcal{C}$ 与敌手 $\mathcal{A}$ 之间的攻击游戏，相关符号说明如表 1 所示。

表 1　　相关符号说明

符号	说明
pp	系统公共参数
$(\mathrm{lsk},\mathrm{lpk})$	系统链接密钥对
id_i	第 i 个选民身份
$(\mathrm{sk}_i,\mathrm{pk}_i)$	第 i 个选民的公私钥对
B_i	第 i 个选民的链接基参数
$R=(\mathrm{pk}_1,\cdots,\mathrm{pk}_n)$	环公钥集合
n	环大小
m	待签名消息
C	消息密文
tag_i	链接标签
k	投票次数上限
$I=(C,\sigma)$	待检测合法性的签名
$\mathcal{L}=\{C_i,\sigma_i,\mathrm{tag}_i\}_{i=1}^{n}$	合法性检验后的签名集合
σ	环签名
$\hat{\sigma}$	聚合签名

（1）匿名性攻击游戏 $\mathrm{Game}_{\mathcal{A}}^{\mathrm{Anon}}(\tau)$

①系统参数设置。$\mathcal{C}$ 输入安全参数 τ 运行 $\mathrm{Setup}(1^{\tau})$ 生成系统公共参数 pp 与链接私钥 lsk，并初始化注册与公钥列表。

②查询阶段。$\mathcal{A}$ 可自适应访问 $\mathrm{Register}(\mathrm{id})$、$\mathrm{ReplacePK}(\mathrm{id},\mathrm{pk}')$、$\mathrm{Sign}(\mathrm{id},m)$，其中 Sign 仅允许对“诚实生成且未被替换”的密钥进行查询。

③挑战阶段。$\mathcal{A}$ 输出消息 m^* 与两个身份 $\mathrm{id}_0,\mathrm{id}_1$（均对应未被替换的诚实公钥）。$\mathcal{C}$ 随机取 $b\in\{0,1\}$，并生成 $\sigma^*\leftarrow\mathrm{Sign}(m^*,\mathrm{sk}_{\mathrm{id}_b},R,\mathrm{lpk})$，其中 R 为当前公钥列表构成的环，并将 σ^* 返回给 $\mathcal{A}$。随后 $\mathcal{A}$ 可继续访问上述接口，但禁止对 (id_0,m^*) 与 (id_1,m^*) 发起签名查询。最终 $\mathcal{A}$ 输出 $b'\in\{0,1\}$，若 $b'=b$，则敌手赢得该游戏。匿名性优势定义为：

$$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{Anon}}(\tau)=\left|\Pr\left[\mathrm{Game}_{\mathcal{A}}^{\mathrm{Anon}}(\tau)=1\right]-\frac{1}{2}\right| \tag{1}$$

若对任意概率多项式时间敌手 $\mathcal{A}$，$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{Anon}}(\tau)$ 为可忽略量，则方案满足匿名性。

（2）不可伪造性攻击游戏 $\mathrm{Game}_{\mathcal{A}}^{\mathrm{Forge}}(\tau)$

①系统参数设置。同 $\mathrm{Game}_{\mathcal{A}}^{\mathrm{Anon}}(\tau)$ 步骤①。

②查询阶段。同 $\mathrm{Game}_{\mathcal{A}}^{\mathrm{Anon}}(\tau)$ 步骤②。

③挑战阶段。在任意查询后，敌手 $\mathcal{A}$ 输出一个三元组 (m^*,σ^*,R^*)，其中 R^* 为用于验证的环。挑战

者执行单个签名验证算法，检查算法 $\mathrm{Verify}(m^*,\sigma^*,R^*)$ 是否输出1。若验证失败，则敌手失败。若验证成功，则进一步要求 (m^*,σ^*) 并非敌手此前通过查询直接获得的输出。若上述条件均满足，则 $\mathcal{A}$ 赢得此游戏。

本文将不可伪造性优势定义为：

$$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{Forge}}(\tau)=\Pr[\mathrm{Game}_{\mathcal{A}}^{\mathrm{Forge}}(\tau)=1]\leqslant \mathrm{negl}(\tau) \quad (2)$$

若任意概率多项式时间敌手 $\mathcal{A}$ 赢得此游戏的概率可忽略，则称方案满足不可伪造性。

（3）次数受限性攻击游戏 $\mathrm{Game}_{\mathcal{A}}^{\mathrm{kBound}}(\tau)$。

① 系统参数设置。同 $\mathrm{Game}_{\mathcal{A}}^{\mathrm{Anon}}(\tau)$ 的步骤①。

② 查询阶段。同 $\mathrm{Game}_{\mathcal{A}}^{\mathrm{Anon}}(\tau)$ 的步骤②。

③ 挑战阶段。在任意查询后，敌手构造一个待计票集合 $I^*=(C,\sigma)$，并提交至挑战者。挑战者运行链接算法 $\mathrm{Klink}(\mathrm{lsk},k,I^*,R)$。若算法输出拒绝符号⊥，则敌手失败。若输出一个包含链接标签的集合 $\mathcal{L}^*=\{(C,\sigma,\mathrm{tag})\}$，则该集合存在某一链接标签 tag^* 对应的有效投票条目数大于 k，若这些条目均通过单个签名验证，则敌手赢得此游戏。

本文将次数受限性优势定义为：

$$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{kBound}}(\tau)=\Pr[\mathrm{Game}_{\mathcal{A}}^{\mathrm{kBound}}(\tau)=1]\leqslant \mathrm{negl}(\tau) \quad (3)$$

若任意概率多项式时间敌手 $\mathcal{A}$ 赢得此游戏的概率可忽略，则称方案满足次数受限性。

3 方案设计

3.1 形式化定义

（1）$\mathrm{Setup}(1^\tau)$：系统初始化算法。该算法由注册中心执行，输入安全参数 1^τ，输出公共参数 pp 与系统链接私钥 lsk。

（2）$\mathrm{KeyGen}(\mathrm{id},\mathrm{pp})$：选民注册协议。该算法由选民执行，输入身份信息 id 与公共参数 pp，获得对应的私钥 sk_i、公钥 pk_i 与链接基参数 B_i。注册中心将该公钥加入环，使选民成为具备投票资格的投票人。

（3）$\mathrm{Sign}(m,\mathrm{sk}_i,R,\mathrm{lpk})$：签名算法。该算法由投票人执行，输入消息 m、私钥 sk_i、环 $R=(\mathrm{pk}_1,\cdots,\mathrm{pk}_n)$，以及系统链接公钥 lpk，输出环签名 σ 与密文消息 C。

（4）$\mathrm{Verify}(C,\sigma,R)$：单个签名验证算法。该算法由计票中心执行，输入消息密文 C、环签名 σ 和环 $R=(\mathrm{pk}_1,\cdots,\mathrm{pk}_n)$。若签名验证通过则输出1，否则输出0。

（5）$\mathrm{Klink}(\mathrm{lsk},k,I)$：链接算法。该算法由计票中心执行，输入链接密钥 lsk、投票次数上限 k，以及待检验的签名集合 $I=(C,\sigma)$。计票中心对签名集合中的签名执行批量链接与次数统计操作，若任一链接标签对应的签名数量超过投票次数上限 k，则输出⊥；否则，输出通过验证的签名集合 $\mathcal{L}=\{C_i,\sigma_i,\mathrm{tag}_i\}_{i=1}^n$。

（6）$\mathrm{Aggregate}(\mathcal{L})$：聚合签名生成算法。该算法由计票中心执行，输入标签签名集合 $\mathcal{L}$，输出聚合签名 $\hat{\sigma}$。

（7）$\mathrm{AggVerify}(\mathrm{pp},R,\hat{\sigma})$：聚合签名验证算法。该算法由监管中心执行，输入系统公共参数 pp、环 R 和聚合签名 $\hat{\sigma}$。若聚合签名验证通过则输出1，否则输出0。

3.2 系统构造

本节介绍本文提出的匿名可监管的电子投票系统，包括初始化、选民注册、投票和计票4个阶段。

3.2.1 *初始化阶段*

在初始化阶段，注册中心运行系统初始化算法 $\mathrm{Setup}(1^\tau)$ 生成系统公共参数与链接私钥。首先依据安全参数 1^τ 执行双线性群生成算法，得到双线性群参数 $(p,\mathbb{G}_1,\mathbb{G}_2,\mathbb{G}_T,e)$，其中 p 为大素数，$\mathbb{G}_1,\mathbb{G}_2,\mathbb{G}_T$ 均为阶为 p 的循环乘法群，$e:\mathbb{G}_1\times\mathbb{G}_2\rightarrow\mathbb{G}_T$ 为可高效计算的非退化双线性映射。随后，注册中心分别从群 $\mathbb{G}_1,\mathbb{G}_2$ 中选取生成元 $g\in\mathbb{G}_1$ 与 $\omega\in\mathbb{G}_2$，用于后续用户公钥生成与链接密钥构造。初始化哈希函数集合，定义挑战哈希函数 $H_1:\{0,1\}^*\rightarrow\mathbb{Z}_p$，用于将任意长度的消息映射为有限域 $\mathbb{Z}_p$ 中的元素；同时定义链接基哈希函数 $H_2:\mathbb{G}_1\rightarrow\mathbb{G}_2$，用于将用户公钥映射至群 $\mathbb{G}_2$。

注册中心从有限域 $\mathbb{Z}_p^*$ 中随机选取标量 φ_1，并据此计算系统的链接密钥对，其中链接私钥定义为 $\mathrm{lsk}=\varphi_1\in\mathbb{Z}_p$，链接公钥定义为 $\mathrm{lpk}=\omega^{\varphi_1}\in\mathbb{G}_2$。链接公钥 lpk 作为系统公共参数对外发布，链接私钥 lsk 由计票中心秘密持有，用于链接阶段恢复并比较签名对应的链接标签。

为支持投票密文的生成，系统同时设置SM2加密所需的椭圆曲线参数 $\mathrm{pp}_{\mathrm{SM}_2}=(\frac{E}{\mathbb{F}_q},G,n)$，其中 G 为基点，n 为其阶。监管中心随机选取私钥

$d_T \leftarrow \mathbb{Z}_n$，并计算对应公钥$P_T \leftarrow d_T G$。系统同时指定密钥派生函数（key derivation function, KDF）KDF: $\{0,1\}^* \times \mathbb{N} \rightarrow \{0,1\}^l$与完整性校验函数$H$: $\{0,1\}^* \rightarrow \{0,1\}^r$，以分别派生掩码密钥流并生成校验值。最终，输出系统公共参数与链接私钥，形式为：

$$\mathrm{pp} = (\mathbb{G}_1,\mathbb{G}_2,\mathbb{G}_T,p,e,g,\omega,H_1,H_2,\mathrm{lpk},\mathrm{pp}_{\mathrm{SM}_2}),\mathrm{lsk} = \varphi_1 \quad (4)$$

3.2.2 选民注册阶段

在选民注册阶段，选民输入身份信息id与公共参数pp，运行注册算法KeyGen(id,pp)获得投票资格。选民随机采样$x_i \in \mathbb{Z}_p$作为私钥，计算对应公钥$\mathrm{pk}_i = g^{x_i} \in \mathbb{G}_1$。私钥$x_i$由选民秘密保存，公钥$\mathrm{pk}_i$参与环结构构造。利用哈希函数$H_2$对公钥进行映射，得到对应的链接基参数：

$$B_i = H_2(\mathrm{pk}_i)^{x_i} \in \mathbb{G}_2 \quad (5)$$

完成上述步骤后，选民获得投票资格成为投票人，并本地保存$(\mathrm{sk}_i,\mathrm{pk}_i,B_i)$以支持后续环签名生成、验证及指定链接操作。

3.2.3 投票阶段

在投票阶段，投票人首先对选票内容进行加密，再使用签名算法Sign$(m,\mathrm{sk}_i,R,\mathrm{lpk})$完成环签名的生成与提交。

步骤 1　选票内容的加密。投票人将明文消息M与公钥P_T作为输入运行SM2加密算法，其中明文消息由候选标识choice、提交时间戳timestamp和随机数$\rho \in \{0,1\}^*$组成。

$$M = (\mathrm{choice},\mathrm{timestamp},\rho) \quad (6)$$

投票人从有限域中随机选择临时密钥$\theta \in \mathbb{Z}_q^*$，计算椭圆曲线上的点$C_1 = \theta G$，并计算共享点$S = \theta P_T = (x_2,y_2)$，若共享点坐标非法，则重新采样$\theta$。投票人将$(x_2,y_2)$按照固定格式拼接后作为KDF的输入，得到$t = \mathrm{KDF}(x_2\|y_2,|M|)$：若输出的$t$为全零比特串则拒绝采样并重新采样$\theta$，并重复上述步骤。投票人对明文消息进行掩码处理$C_2 = M \oplus t$，计算校验值$C_3 = H(x_2,\|M\|y_2)$以支持解密后的完整性验证，输出密文$C = (C_1,C_2,C_3)$。

步骤 2　环签名的生成。完成选票内容加密后，投票人将公共参数pp、密文C、由n个合法投票人公钥构成的环$R = (\mathrm{pk}_1,\cdots,\mathrm{pk}_n)$、自身索引$s$、私钥$\mathrm{sk}_s = x_s$，以及链接参数$B_s$作为输入并执行环签名算法。

投票人随机选取$t \in \mathbb{Z}_p$，结合链接参数计算$K_a = B_s\mathrm{lpk}$和$K_b = \omega^t$，以支持后续签名验证与链接过程。随机选取$r_t \in \mathbb{Z}_p$，计算$T_1 = \mathrm{lpk}^{r_t}$，$T_2 = \omega^{r_t}$。随机选取$u \in \mathbb{Z}_p$，计算中间量$D_s = H_2(\mathrm{pk}_s)^u$。随后投票人计算起始挑战值$c_{s+1} = H_1(C\|R\|K_a\|K_b\|D_s)$。投票人对索引$i \in \{1,\cdots,n\} \setminus \{s\}$依次按照模$n$的顺序递增计算：对于索引$i$选择$s_i \in \mathbb{Z}_p$，计算$D_i = H_2(\mathrm{pk}_i)^{s_i} B_s^{c_i}$与中间挑战值$c_{i+1} = H_1(C\|R\|K_a\|K_b\|D_i)$。投票人将最终得到的标量值记为$c_{n+1}$，由于索引按照模$n$的顺序递增计算，$c_{n+1}$最终记为$c_1$。投票人根据自身私钥计算$s_s = u - c_s x_s \bmod p$，设置响应向量$S = (s_1,\cdots,s_n)$、$e = H_3\left(K_a\|K_b\|B_s\|T_1\|T_2\|c_1\right)$与$z_t = r_t - et \bmod p$，并最终输出环签名与密文消息分别为：

$$\sigma = (K_a,K_b,B_s,T_1,T_2,c_1,S,z_t)$$
$$C = (C_1,C_2,C_3) \quad (7)$$

3.2.4 计票阶段

在计票阶段，计票中心先执行单个环签名验证算法Verify(C,σ,R)以判断选票合法性，并通过链接算法Klink(lsk,k,I)识别并约束多次投票行为，随后使用聚合签名生成算法Aggregate$(\mathcal{L})$聚合有效选票并提交至监管中心。

步骤 1　单个签名验证。计票中心输入消息密文C、环签名$\sigma = (K_a,K_b,B_s,T_1,T_2,c_1,S,z_t)$，以及环$R = (\mathrm{pk}_1,\cdots,\mathrm{pk}_n)$，运行单个签名验证算法。计票中心解析响应向量$S = (s_1,\cdots,s_n)$，设置初始挑战值$c_1' = c_1$，并依照环中公钥顺序依次重建承诺值与挑战值：对于第i个公钥pk_i，计算$D_i = H_2(\mathrm{pk}_i)^{s_i} B_s^{c_i'}$，$c_{i+1}' = H_1(C\|R\|K_a\|K_b\|D_i)$，以及$e = H_3(K_a\|K_b\|B_s\|T_1\|T_2\|c_1)$，完成后得到$c_{n+1}'$。若$c_{n+1}' = c_1$且$\mathrm{lpk}^{z_t}(K_a B_s - 1)^e = T_1$，$\omega^{z_t} K_b^e = T_2$成立，则签名结构一致验证通过输出1，否则判定该选票无效并输出0。

步骤 2　投票次数检测。计票中心输入链接密钥lsk、投票次数上限k，以及待检验的签名集合$I = \{(C_j,\sigma_j)\}$，运行链接算法。

计票中心初始化空的有效集合$\mathcal{L}$和计数映射$\mathcal{C}[\cdot]$（用于记录各链接标签的出现次数）。计票中心先对I中每一项(C_j,σ_j)解析出其中的$(K_{a,j},K_{b,j})$，再结合系统链接私钥lsk与双线性映射计算该选票对应的链接标签。

$$\text{tag}_j \leftarrow K_{a,j}K_{b,j}^{-\text{lsk}} \tag{8}$$

随后计票中心执行 $\mathcal{C}[\text{tag}_j] \leftarrow \mathcal{C}[\text{tag}_j]+1$ 更新计数映射。若更新后 $\mathcal{C}[\text{tag}_j] > k$，计票中心据此判定出现超额投票行为，输出 $\perp$；否则，计票中心将三元组 $(C_j,\sigma_j,\text{tag}_j)$ 加入有效集合 $\mathcal{L}$，并输出有效集合 $\mathcal{L}$。

在本文方案中，系统通过设定总投票次数上限 k 对投票者可提交的有效选票数量进行约束，从而在统一框架下支持不同投票模式。当 $k=1$ 时，对应单选投票模式；当 $k>1$ 时，对应累积投票模式。

步骤 3 投票的聚合。在完成批量投票次数检验后，计票中心输入有效选票集合 $\mathcal{L}=(C_j,\sigma_j,\text{tag}_j)$。执行聚合算法输出聚合签名 $\hat{\sigma}$，并将其发送至监管中心。

计票中心初始化一个列表 S 与计数映射 $\mathcal{C}[\cdot]$，其中 S 用于存放聚合后的有效选票条目，$\mathcal{C}[\cdot]$ 用于记录每一个链接标签在有效选票中出现的次数，为监管中心提供可核对的计数信息。计票中心依次遍历集合 $\mathcal{L}$ 中的三元组 $(C_j,\sigma_j,\text{tag}_j)$，直接将 $(C_j,\sigma_j,\text{tag}_j)$ 加入列表 S，同时以 tag_j 作为索引更新计数映射，执行累加操作 $\mathcal{C}[\text{tag}_j] \leftarrow \mathcal{C}[\text{tag}_j]+1$。完成全部有效选票的遍历与更新后，计票中心输出聚合结果式(9)，并发送至监管中心。

$$\hat{\sigma}=(S,\mathcal{C}) \tag{9}$$

步骤 4 聚合投票的验证。监管中心输入系统公共参数 pp、环 R 和聚合签名 $\hat{\sigma}=(S,\mathcal{C})$，运行聚合签名验证算法 $\text{AggVerify}(\text{pp},R,\hat{\sigma})$，若聚合签名验证通过则输出 1，否则输出 0。监管中心解析系统公共参数 pp，初始化一个空的计数映射 $\mathcal{C}'$，以独立统计聚合结果中各链接标签出现的次数。依次遍历 S 中的条目 $(C_j,\sigma_j,\text{tag}_j)$，解析签名 $\sigma_j=(K_{a,j},K_{b,j},c_{j,1},S_j)$，其中 $S_j=(s_{j,1},\cdots,s_{j,n})$。

以 $c_{j,1}$ 作为初始挑战值，按环中公钥顺序依次计算 $L_{j,i}=g^{s_{j,i}}\text{pk}_i^{c_{j,i}'}$，并重新计算 $c_{j,i+1}'=H_1(C_j\|R\|K_{a,j}\|K_{b,j}\|L_{j,i})$。完成 n 次迭代后得到 $c_{j,n}$ 并比较 $c_{j,n}=c_{j,1}$，若两者相等，则说明该签名结构在环顺序上保持一致，验证通过输出 1；否则，监管中心判定该条目无效输出 0。

步骤 5 选票内容的解密。在解密阶段，监管中心输入选票密文 $C=(C_1,C_2,C_3)$ 与解密私钥 d_T，运行 SM2 解密算法获得选票明文，其中 $C_1=(x_1,y_1)\in E(\mathbb{F}_p)$ 为曲线点，C_2 为掩码后的密文主体，C_3 为完整性校验值。监管中心首先利用私钥 d_T 计算共享点 $\mathcal{S}=d_TC_1=(x_2,y_2)$，将坐标按照固定格式编码为 $Z=x_2\|y_2$。接着调用 KDF，计算 $t=\text{KDF}(Z,|C_2|)$，若输出为全零比特，则拒绝该密文。最后监管中心将密文数据按位异或恢复投票明文 $M=C_2\oplus t$。通过计算完整性校验值 $C_3'=H(x_2\|M\|y_2)$ 确保密文未被篡改，若 $C_3'\neq C_3$ 则拒绝该选票，随后将获得的明文 M 用于计票。

4 安全性分析

4.1 正确性

签名的形式为 $\sigma=(K_a,K_b,B_s,T_1,T_2,c_1,S,z_t)$，其中 $S=(s_1,\cdots,s_n)$。大小为 n 的环形式为 $R=(\text{pk}_1,\cdots,\text{pk}_n)$，对于任意 $i\in\{1,\cdots,n\}$，有公钥 $\text{pk}_i=g^{x_i}$。设签名者在环中的索引为 s，其对应的私钥为 x_s。

（1）单个签名验证

若签名 σ 为索引为 s 的诚实签名者按签名算法生成，则签名阶段在索引 s 处满足：

$$L_s=g^{s_s}\text{pk}_i^{c_s}=g^{u-c_sx_s}(g^{x_s})^{c_s}=g^u \tag{10}$$

与验证阶段得到的中间量 L_s 一致。

对于非签名者位置 $i\neq s$ 的情况，由于 s_i 为随机选取，因此有：

$$L_i=g^{s_i}\text{pk}_i^{c_i}=g^{s_i}(g^{x_i})^{c_i}=g^{s_i+x_ic_i} \tag{11}$$

式(11)与签名阶段生成的中间量内容一致。由式(10)和式(11)可知，对于所有 $i\in\{1,\cdots,n\}$，验证方重新计算得到的 L_i 与签名生成阶段对应位置计算结果一致。因此，哈希值 $H_1(C\|R\|K_a\|K_b\|L_i)$ 的计算结果与签名生成阶段保持一致，最终满足：

$$c_{n+1}=c_1 \tag{12}$$

故验证判定条件式(12)成立，单个环签名验证通过。

（2）k 次签名验证

在签名生成阶段，签名者随机选取随机数 $t\in\mathbb{Z}_p$，计算链接参数：

$$K_a=B_s\text{lpk}^t,K_b=\omega^t \tag{13}$$

计票中心在持有链接私钥 $\text{lsk}=g^{\varphi_1}$ 的条件下对签名 σ 计算链接标签：

$$\text{tag}_j \leftarrow K_{a,j}K_{b,j}^{-\text{lsk}} \tag{14}$$

将式(13)代入式(14)并化简，可得：

$$\mathrm{tag} = B_{s_j}\mathrm{lpk}^{t}(\omega^{t})^{-\mathrm{lsk}} = B_{s_j} \tag{15}$$

由式(15)可知，对于同一投票人生成的任意多份签名，其对应的链接标签完全相同，故可判断相同投票人的投票次数是否超出最大允许投票次数上限。

（3）聚合签名验证

计票中心在完成单个签名验证与投票次数检验后，将所有满足约束的有效选票组织为聚合对象 $\hat{\sigma} = (S,\mathcal{C})$。监管中心在聚合验证阶段其判定条件包含两个部分。

①聚合对象每一份环签名均为合法签名，同单个签名验证步骤，此处不再赘述。

②聚合对象中声明的技术信息与实际包含的条目数量一致。

完成步骤①中的单个签名验证后，监管中心对集合中的每一个条目 $(\mathcal{C},\sigma,\mathrm{tag}) \in S$ 执行计数更新操作 $\mathcal{C}'[\mathrm{tag}_j] \leftarrow \mathcal{C}'[\mathrm{tag}_j] + 1$。完成上述操作后可得到计数映射 $\mathcal{C}'[\mathrm{tag}] = |C,\sigma,\mathrm{tag} \in S|$，若满足 $\mathcal{C}'[\mathrm{tag}] = \mathcal{C}[\mathrm{tag}]$，即表示满足条件②的要求，监管中心会正确接受聚合对象 $\hat{\sigma}$。

4.2　匿名性

定理1　在随机预言机模型下，若DDH假设在群 $\mathbb{G}_2$ 上成立，则针对任意多项式时间的敌手 $\mathcal{A}$，其匿名性优势为可忽略函数。

证明　通过4个计算不可区分的游戏与归约进行证明，记敌手在游戏 Game i 中获胜的概率为 $\Pr[G_i]$。

Game 0：真实匿名性游戏 $\mathrm{Game}_{\mathcal{A}}^{\mathrm{Anon}}(\tau)$。挑战者 $\mathcal{C}$ 选取比特 $b \in \{0,1\}$，使用真实私钥为 $\mathcal{A}$ 构造挑战签名。

Game 1：随机化头部组件 (K_a^{*},K_b^{*})，B_s^{*} 保持真实。初始与查询阶段同 Game 0。在生成挑战签名阶段，$\mathcal{C}$ 随机选取 $T \leftarrow \mathbb{G}_2$，令：

$$K_a^{*} = B_s^{*}T, K_b^{*} = \omega^{t} \tag{16}$$

随机选取 $e,z_t \leftarrow \mathbb{Z}_p$，对 H_3 的对应输入编程输出 e，并令：

$$T_1^{*} = \mathrm{lpk}^{z_t}(K_a^{*}(B_s^{*})^{-1})^{e}, T_2^{*} = \omega^{z_t}K_b^{*e} \tag{17}$$

其余签名分量的生成方式同 Game 0。

Game 2：在 Game 1 的基础上，进一步将 B_s^{*} 替换为均匀随机群元素。初始与查询阶段同 Game 0。在生成挑战签名阶段，$\mathcal{C}$ 随机选取 $R \leftarrow \mathbb{G}_2$，令 $B_s^{*} = R$，哈希链中对应计算替换为：

$$D_i = H_2(\mathrm{pk}_i)^{s_i}R^{c_i} \tag{18}$$

其余部分模拟方式同 Game 1，T_1^{*}、T_2^{*} 关于 R 重新设定。

Game 3：初始与查询阶段同 Game 0。在生成挑战签名阶段，$\mathcal{C}$ 随机选取 $c_1^{*} \leftarrow \mathbb{Z}_p$，$s_1,\cdots,s_n \leftarrow \mathbb{Z}_p$，对于 $i = 1,\cdots,n$ 顺序计算 $D_i = H_2(\mathrm{pk}_i)^{s_i}R^{c_i}$，通过对 H_1 编程输出设定为 $c_{i+1}^{*} = H_1(C^{*}\|R^{*}\|K_a^{*}\|K_b^{*}\|D_i)$，最终确保 $c_{n+1}^{*} = c_1^{*}$，挑战签名的生成与 b 完全无关。

引理1　若DDH假设在群 $\mathbb{G}_2$ 上成立，则对任意概率多项式敌手 $\mathcal{A}$，Game 0 与 Game 1 在计算上不可区分，即：

$$|\Pr[G_0 = 1] - \Pr[G_1 = 1]| \leqslant \mathrm{Adv}_{\mathcal{B}}^{\mathrm{DDH}}(\tau) \tag{19}$$

证明　假设存在一个概率多项式敌手 $\mathcal{A}$ 能以不可忽略的优势区分 Game 0 与 Game 1，构造模拟器 $\mathcal{B}$ 求解DDH问题。给定DDH实例 $(\omega,\omega^{\varphi_1},\omega^{t},T)$，设 $\mathrm{lpk} = \omega^{\varphi_1}$，此时 $\mathcal{B}$ 不掌握对应的链接私钥 $\mathrm{lsk} = \varphi_1 \in \mathbb{Z}_p$。令 $K_a^{*} = B_s^{*}T, K_b^{*} = \omega^{t}$，随机选取 e，$z_t \leftarrow \mathbb{Z}_p$，对 H_3 的对应输入编程输出 e，并令 $T_1^{*} = \mathrm{lpk}^{z_t}(K_a^{*}(B_s^{*})^{-1})^{e}, T_2^{*} = \omega^{z_t}K_b^{*e}$。若等式 $T = \omega^{\varphi_1 t} = \mathrm{lpk}^{t}$ 成立，则 $K_a^{*} = B_s^{*}\mathrm{lpk}^{t}$ 与 Game 0 中真实签名分布一致；若 T 为群 $\mathbb{G}_2$ 上的均匀随机元素，则 K_a^{*} 在 $\mathbb{G}_2$ 上均匀随机，与 Game 1 分布一致，由DDH假设两者不可区分。证毕。

引理2　若DDH假设在群 $\mathbb{G}_2$ 上成立，则对任意概率多项式敌手 $\mathcal{A}$，Game 1 与 Game 2 在计算上不可区分，即：

$$|\Pr[G_0 = 1] - \Pr[G_1 = 1]| \leqslant \mathrm{Adv}_{\mathcal{B}}^{\mathrm{DDH}}(\tau) \tag{20}$$

证明　构造模拟器 $\mathcal{B}'$ 求解DDH实例 $\left(H_2(\mathrm{pk}_{\mathrm{id}_b}),H_2(\mathrm{pk}_{\mathrm{id}_b})^{x_{\mathrm{id}_b}},X\right)$，即判断 $X = H_2(\mathrm{pk}_{\mathrm{id}_b})^{x_{\mathrm{id}_b}}$ 是否成立，其中 $B_s^{*} = X$。若 $X = H_2(\mathrm{pk}_{\mathrm{id}_b})^{x_{\mathrm{id}_b}}$，则 B_s^{*} 为真实链接基，敌手对应 Game 1 的分布；若 X 为群 $\mathbb{G}_2$ 上的均匀随机元素，则敌手对应 Game 2 的分布。注意到，Game 1 中 K_a^{*} 已均匀随机，故 B_s^{*} 的替换不影响 K_a^{*} 的分布，由 $\mathbb{G}_2$ 上DDH假设两者不可区分。证毕。

引理3　DDH假设在群 $\mathbb{G}_2$ 上成立，则对任意概率多项式敌手 $\mathcal{A}$，Game 2 与 Game 3 在计算上不可区分，即：

$$|\Pr[G_2=1]-\Pr[G_3=1]|\leqslant q_{H_1}+\frac{q_s}{p} \quad (21)$$

证明 在真实签名中签名者由随机数u导出$s_s=u-c_sx_s$，由于映射$f(u)=u-c_sx_s$是$\mathbb{Z}_p$上的双射，直接采样$s_s\leftarrow\mathbb{Z}_p$与由随机数$u$导出的分布完全等价。模拟器$\mathcal{B}$通过对$H_1$编程确保${c_{n+1}}^*={c_1}^*$，模拟失败仅发生在哈希碰撞时，概率上限为$q_{H_1}+\frac{q_s}{p}$，由于$p$为大素数，该偏差可忽略。证毕。

综上所述，在Game 3中挑战签名的生成与b无关，故$\Pr[\mathrm{Exp}_{\mathcal{A}}^{G_3}=1]=\frac{1}{2}$，可得：

$$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{Anon}}(\tau)=\left|\Pr\left[\mathrm{Game}_{\mathcal{A}}^{\mathrm{Anon}}(\tau)=1\right]-\frac{1}{2}\right|\leqslant \mathrm{Adv}_{\mathcal{B}}^{\mathrm{DDH}}(\tau)+\mathrm{Adv}_{\mathcal{B}}^{\mathrm{DDH}}(\tau)+q_{H_1}+\frac{q_s}{p} \quad (22)$$

因此，针对任意概率多项式时间敌手$\mathcal{A}$，其匿名性优势为可忽略函数，定理1证毕。

4.3 不可伪造性

定理2 在随机预言机模型下，若DL问题在群$\mathbb{G}_2$困难，则针对任意多项式时间的敌手$\mathcal{A}$在不可伪造攻击游戏$\mathrm{Game}_{\mathcal{A}}^{\mathrm{Forge}}(\tau)$中获胜的概率为可忽略函数。

证明 本文通过游戏与归约证明上述结论。

Game 0：即$\mathrm{Game}_{\mathcal{A}}^{\mathrm{Forge}}(\tau)$为真实不可伪造游戏。

Game 1：猜测目标诚实公钥索引。初始与查询阶段同Game 0。设系统中诚实且未被替换的公钥数量为N，$\mathcal{C}$在游戏开始时额外引入随机目标索引，对应某诚实生成且未被替换的公钥pk_t。$\mathcal{A}$输出(m^*,σ^*,R^*)后，若$\mathrm{pk}_t\notin R$，则判定$\mathcal{A}$失败；否则，按Game 0获胜条件判定。

Game 2：猜测目标诚实公钥索引。与Game 1相同，差异在于$\mathcal{C}$回答$\mathrm{Sign}(\mathrm{id},m)$查询时，若所用环$R$包含目标公钥$\mathrm{pk}_t$，则$\mathcal{C}$不再调用真实签名算法，返回模拟签名。具体而言，$\mathcal{C}$随机选取$B_{\mathrm{sim}}\leftarrow\mathbb{G}_2$，$c_1,s_1,\cdots,s_n\leftarrow\mathbb{Z}_p$，对$i=1,\cdots,n$顺序计算：

$$D_i=H_2(\mathrm{pk}_i)^{s_i}B_{\mathrm{sim}}^{c_i} \quad (23)$$

设置$c_{i+1}=H_1(m\|R\|K_a\|K_a\|D_i)$为随机预言机$H_1$的输出，一致性响应机制确保最终验证条件$c_{n+1}=c_1$成立。随机选取$e$，$z_t\leftarrow\mathbb{Z}_p$，设置$H_3$的输出为$e$，令$T_1=\mathrm{lpk}^{z_t}(K_aB_{\mathrm{sim}}^{-1})^e$，$T_2=\omega^{z_t}K_b^e$。当环不包含$\mathrm{pk}_t$时，仍依照真实签名方式签名。

引理4 对任意概率多项式时间敌手$\mathcal{A}$，有$\Pr[\mathrm{Game}\ 0=1]\leqslant N\Pr[\mathrm{Game}\ 1=1]$。

证明 若敌手$\mathcal{A}$在Game 0中获胜，则其输出环R^*至少包含一个诚实未被替换的公钥。Game 1随机选取目标公钥pk_t，其落入该集合的概率至少为$\frac{1}{N}$，故$\Pr[\mathrm{Game}\ 1=1]\geqslant\frac{\Pr[\mathrm{Game}\ 0=1]}{N}$，结论成立。证毕。

引理5 在随机预言机模型下，对任意概率多项式时间敌手$\mathcal{A}$使$\Pr[\mathrm{Game}\ 1=1]$不可忽略，则存在多项式时间算法$\mathcal{B}$能以不可忽略求解$\mathbb{G}_2$上的离散对数问题。

证明 本文构造DL求解器$\mathcal{B}$。$\mathcal{B}$的输入为$(H_2(\mathrm{pk}_t),Z)$，其中$Z=H_2(\mathrm{pk}_t)^{x_t}$，输出$x_t$。第一步进行模拟过程，当$\mathcal{B}$模拟整个Game 2的交互过程时，将目标公钥设置为$\mathrm{pk}_t=Y$，其余诚实公钥$\mathrm{pk}_i=g^{x_i}$正常生成。随机预言机由$\mathcal{B}$维护查询表。对于Sign查询，当环不包含pk_t时，$\mathcal{B}$使用已知私钥真实签名；当环包含pk_t时，$\mathcal{B}$依照Game 2的规则输出模拟签名，这使敌手在$\mathcal{B}$的模拟中看到的分布与Game 2一致。

第二步获取私钥相关方程。当$\mathcal{A}$在Game 1中获胜时，输出(m^*,σ^*,R^*)，且$\mathrm{pk}_t\in R^*$并通过验证。对环$R^*=(\mathrm{pk}_1,\cdots,\mathrm{pk}_n)$中索引位置$t$，验证计算：

$$D_t=H_2(\mathrm{pk}_t)^{s_t}(B_s^*)^{c_t} \quad (24)$$

在求解阶段，$\mathcal{B}$以不可忽略概率得到两份对同一(m^*,R^*)的有效伪造$\sigma^{*(1)}$与$\sigma^{*(2)}$，使其在目标位置对应挑战值不同$c_t^{(1)}\neq c_t^{(2)}$，对应的D_t仍然满足一致性关系，即：

$$H_2(\mathrm{pk}_t)^{s_t^{(1)}}(B_s^*)^{c_t^{(1)}}=D_t=H_2(\mathrm{pk}_t)^{s_t^{(2)}}(B_s^*)^{c_t^{(2)}} \quad (25)$$

由于$\mathbb{G}_2$为素数阶循环群，设$B_s^*=H_2(\mathrm{pk}_t)^{\alpha}$，整理可得：

$$\alpha\equiv\frac{s_t^{(1)}-s_t^{(2)}}{c_t^{(2)}-c_t^{(1)}}\ (\mathrm{mod}\,p) \quad (26)$$

若$B_s^*=Z=H_2(\mathrm{pk}_t)^{x_t}$，则$\alpha=x_t$，$\mathcal{B}$可在多项式时间内求出DL实例解$x_t$。若$\Pr[\mathrm{Game}\ 1=1]$不可忽略，则$\mathcal{B}$求解$\mathbb{G}_2$上DL的概率也不可忽略，与DL假设矛盾。证毕。

由引理5及DL假设可得，$\Pr[\mathrm{Game}\ 1=1]$为

可忽略函数，$\Pr[\text{Game } 0 = 1] \leqslant N\Pr[\text{Game } 1 = 1]$且$N$多项式有界，因此$\Pr[\text{Game } 0 = 1]$为可忽略函数，定理2证毕。

4.4　次数受限性

定理 3　在随机预言机模型下，若DL困难，则本文方案满足次数受限性。对于任意多项式时间的敌手$\mathcal{A}$，在次数受限攻击游戏$\text{Game}_{\mathcal{A}}^{\text{kBound}}(\tau)$中获胜的概率为可忽略函数。

证明　通过标签一致性、标签唯一性与不可伪造性来证明敌手$\mathcal{A}$无法生成超过限额的合法签名。

引理 6　标签一致性指任意通过验证的签名σ，其提取标签满足$\text{tag} = K_a K_b^{-\text{lsk}} = B_s$。

证明　存在e和z_t使：

$$\text{lpk}^{z_t}(K_a B_s^{-1})^e = T_1, \omega^{z_t} K_b^e = T_2 \tag{27}$$

由分叉引理，对同一$(K_a, K_b, B_s, T_1, T_2, c_1)$得到两组响应$(e, z_t)$与$(e', z_t')$，$e \neq e'$，可得：

$$\log_{\text{lpk}}(K_a B_s^{-1}) = \frac{z_t - z_t'}{e - e'} \tag{28}$$

$$\log_{\omega}(K_b) = \frac{z_t - z_t'}{e - e'} \tag{29}$$

故$\log_{\text{lpk}}(K_a B_s^{-1}) = \log_{\omega}(K_b) = t$，因此：

$$\text{tag} = K_a K_b^{-\text{lsk}} = B_s \text{lpk}^t \omega^{-\text{lsk}t} = B_s \tag{30}$$

证毕。

引理 7　在DL假设下，标签唯一性指不同投票人的链接标签以不可忽略概率互不相同。

证明　假设存在$i \neq j$使$B_i = B_j$，即：

$$H_2(\text{pk}_i)^{x_i} = H_2(\text{pk}_j)^{x_j} \tag{31}$$

由于合法注册保证$\text{pk}_i \neq \text{pk}_j$，$H_2(\text{pk}_i)$与$H_2(\text{pk}_j)$为对不同输入的独立随机预言机查询结果，在$\mathbb{G}_2$上均匀分布且相互独立。

对于任意固定的x_i，$x_j \in \mathbb{Z}_p^*$，给定$H_2(\text{pk}_i)$，式(31)要求$H_2(\text{pk}_i)$恰好等于$\mathbb{G}_2$中某一特定元素，该时间发生的概率恰为$\frac{1}{p}$。由于p为大素数，该概率可忽略，故不同投票人的B_s以不可忽略概率互不相同。证毕。

假设敌手$\mathcal{A}$在$\text{Game}_{\mathcal{A}}^{k\text{Bound}}(\tau)$中以不可忽略概率获胜，即$\mathcal{A}$使某一tag对应的次数超过$k$。由引理6可知，每张通过验证的选票提取的tag唯一等于签名者的B_s。由引理7可知，不同投票人的B_s互不相同。

若敌手试图规避次数检测，必须在保证签名合法性的前提下篡改链接基参数B_s。具体而言，敌手需要构造一个合法签名使$B_s' \neq B_s$但仍通过验证，这等价于在不知道私钥x_s的情况下伪造一个新的合法签名。由定理2可知，在DL假设下未掌握私钥x_s的敌手无法构造合法签名，故敌手$\mathcal{A}$无法通过替换新的B_s规避次数检测。

因此，敌手$\mathcal{A}$在次数受限攻击游戏$\text{Game}_{\mathcal{A}}^{\text{kBound}}(\tau)$中获胜的概率为可忽略函数，定理3证毕。

5　性能评估

本节与Ordinos方案[2]、Helios方案[14]和k-LRS方案[20]进行性能对比。实验基于Python（版本3.6.9）在Windows Subsystem for Linux 2 (WSL2)环境，运行于Ubuntu 18.04.6 LTS 64位操作系统。本文方案适用Charm-Crypto库，Type-A对称双线性配对，底域素数长度为512位，群阶为160位素数，对应安全强度约为80位，本节中所有实验数据为20次实验结果的平均值。

5.1　通信开销与计算复杂度

如图2所示，本文方案在2~500人规模下的通信开销由0.57 KB增至15.65 KB，整体始终低于Helios方案及两种k-LRS方案。Ordinos方案通信开销基本恒定在7.3 KB，Helios方案增长最快，k-LRS方案则随人数规模增加而上升。在500人规模下，本文方案较上述Helios、k-LRS（k=1）与k-LRS（k=10）方案分别降低约93.30%、75.38%和87.6%。

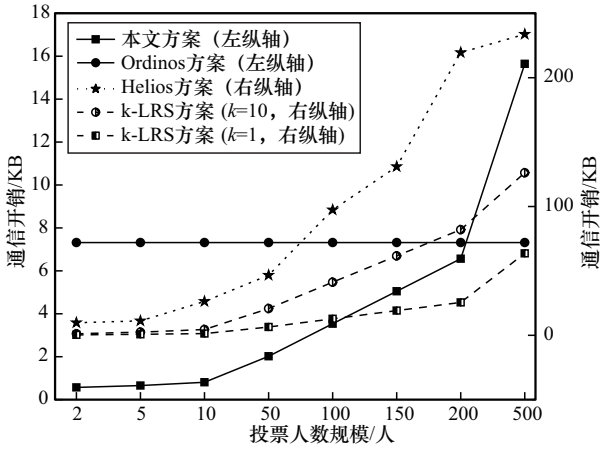

图2　通信开销

算法计算复杂度对比如表2所示，其中n为环规模，k为投票次数上限，Γ为待聚合签名数量。

对比结果表明，本文方案在全流程中实现了算法开销与阈值k的解耦，在链接阶段将重复投票判定从$O(k^3)$降至$O(1)$，提升了高阈值场景下的可扩展性。

表2　　算法计算复杂度对比

算法	k-LRS方案	本文方案
Setup	$O(k)$	$O(1)$
KeyGen	$O(k)$	$O(1)$
Sign	$O(nk)$	$O(n)$
Verify	$O(nk)$	$O(n)$
Klink	$O(k^3)$	$O(1)$
Aggregate	—	$O(\Gamma)$
AggVerify	—	$O(n\Gamma)$

5.2　投票次数上限影响

如图3所示，在总票数为50~200票的条件下，本文方案的链接验证时间整体较低且基本保持稳定，k-LRS方案的链接验证时间随投票次数上限增加而显著上升。本文方案相较于k-LRS方案可平均减少约98.29%的时间开销，故本文方案在链接阶段具备更优的稳定性与效率。

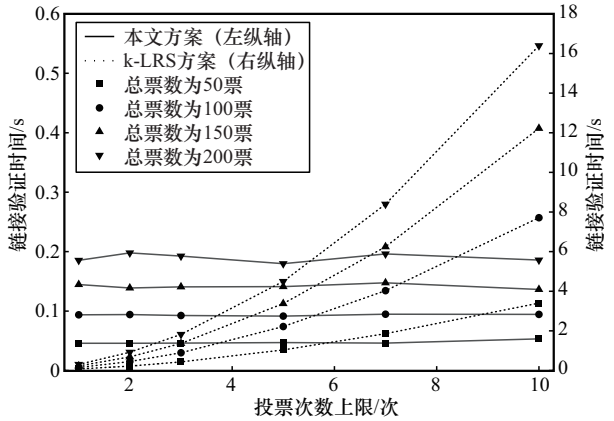

图3　不同人数规模链接验证开销

5.3　投票人数规模影响

如图4所示，在m = 50、k = 10的实验设置下，对比本文方案和k-LRS方案的执行效率。因算法流程差异，Ordinos与Helios方案仅对比投票与验证时间。结果表明，本文方案在整体开销上优于k-LRS方案。本文方案在KeyGen、Sign、Verify与Klink等核心阶段均表现出更高的执行效率，尤其在签名与验证阶段优势更为显著。Aggregate与AggVerify算法作为本文方案特有模块，整体保持在较低开销。

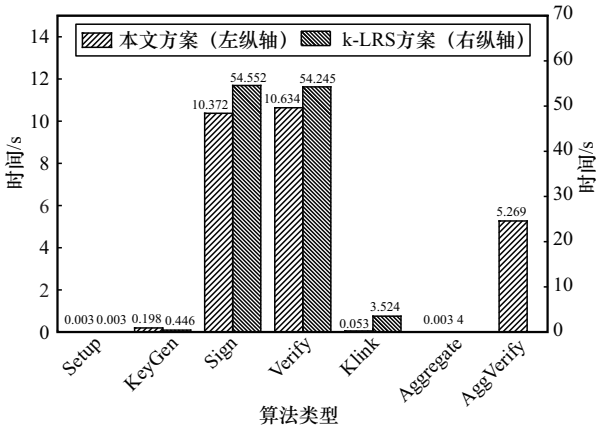

图4　算法整体开销对比

如图5所示，在投票人数规模为2~500人的实验设置下，将本文方案与Ordinos、Helios与k-LRS方案针对投票时间进行对比。结果表明，本文方案在不同投票人数规模下始终保持较低的投票时间开销，尤其在小规模和中等规模场景中优势更为明显；Ordinos方案整体投票时间开销较高且变化相对平缓；k-LRS方案的投票时间则随着投票人数增加呈现显著增长趋势，在大规模场景下尤其突出。上述结果说明，本文方案在不同规模投票场景下具有较优的投票效率。

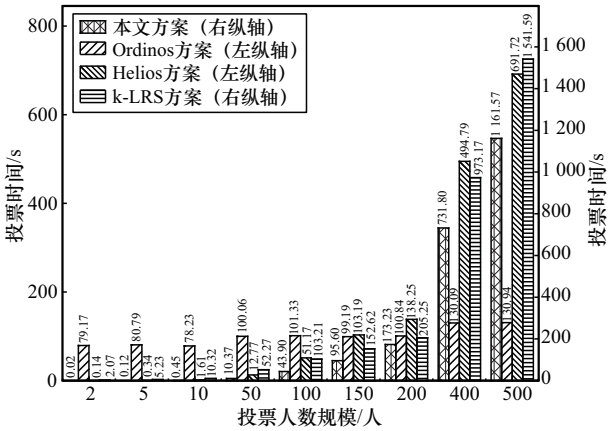

图5　不同投票人数规模下的投票时间

图6对比了本文方案与Ordinos、Helios及k-LRS方案在不同投票人数规模下的验证时间开销。实验结果表明，本文方案在小规模场景下保持较低验证开销。当投票人数由200人增至500人时，Or-

dinos 和 Helios 的验证开销缓慢增长，而 k-LRS 的验证开销显著上升。相比之下，本文方案在大规模场景下仍具有较稳定的验证性能，相较于传统签名类方案具有更好的可扩展性和验证效率。

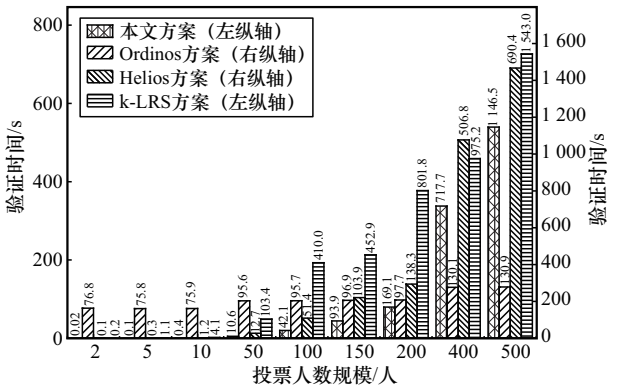

图6　不同投票人数规模下的验证时间

6　结束语

本文提出了一种支持多种投票模式的匿名可监管聚合环签名方案，可在保护投票者身份隐私的同时实现高效投票次数监管。实验结果与安全分析表明，本文方案兼具安全性与效率，尤其在链接阶段，相较于 k-LRS 方案平均减少约 98.29% 的时间开销。

参考文献：

[1] Cohen J D. Improving privacy in cryptographic elections[R]. 1986.

[2] Küsters R, Liedtke J, Müller J, et al. Ordinos: a verifiable tally-hiding E-voting system[C]//Proceedings of the 2020 IEEE European Symposium on Security and Privacy (EuroS&P). Piscataway: IEEE Press, 2020: 216-235.

[3] Anggriane S M, Nasution S M, Azmi F. Advanced e-voting system using Paillier homomorphic encryption algorithm[C]//Proceedings of the 2016 International Conference on Informatics and Computing (ICIC). Piscataway: IEEE Press, 2016: 338-342.

[4] Yuan K, Sang P, Ge J, et al. A timed-release E-voting scheme based on Paillier homomorphic encryption[J]. IEEE Transactions on Sustainable Computing, 2024, 9(5): 740-753.

[5] ElSheikh M, Youssef A M, Hasan M A. Self-tallying E-voting using homomorphic time-lock puzzles and ZK-SNARKs[J]. IEEE Transactions on Network Science and Engineering, 2025, 12(4): 2566-2581.

[6] Jivanyan A, Khachatryan G. New receipt-free E-voting scheme and self-proving mix net as new paradigm[J]. IACR Cryptology EPrint Archive, 2011, 2011: 325.

[7] Haines T, Goré R, Sharma B. Did you mix me? Formally verifying verifiable mix nets in electronic voting[C]//Proceedings of the 2021 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2021: 1748-1765.

[8] Battagliola M, D'Alconzo G, Gangemi A, et al. Enhancing E-voting with multiparty class group encryption[C]//Applied Cryptography and Network Security Workshops. Berlin: Springer, 2025: 251-273.

[9] Canard S, Pointcheval D, Santos Q, et al. Practical strategy-resistant privacy-preserving elections[C]//Computer Security. Berlin: Springer, 2018: 331-349.

[10] Wabartha C, Liedtke J, Huber N, et al. Fully tally-hiding verifiable E-voting for real-world elections with seat-allocations[C]//Computer Security-ESORICS 2023. Berlin: Springer, 2024: 209-228.

[11] 高改梅, 邸国霞, 刘春霞, 等. 基于全同态加密的区块链电子投票方案[J]. 计算机工程, 2026, 52(4): 313-326.
Gao G M, Di G X, Liu C X, et al. Blockchain electronic voting scheme based on fully homomorphic encryption[J]. Computer Engineering, 2026, 52(4): 313-326.

[12] Benaloh J, Moran T, Naish L, et al. Shuffle-sum: coercion-resistant verifiable tallying for STV voting[J]. IEEE Transactions on Information Forensics and Security, 2009, 4(4): 685-698.

[13] Jamroga W, Roenne P B, Ryan P Y A, et al. Risk-limiting tallies[C]//Proceedings of the International Joint Conference on Electronic Voting. Berlin: Springer, 2019: 183-199.

[14] Adida B, Marneffe O D, Pereira O, et al. Electing a university president using open-audit voting: analysis of real-world use of Helios[C]//Proceedings of EVT/WOTE. Berlin: Springer, 2009: 10.

[15] Ramchen K, Culnane C, Pereira O, et al. Universally verifiable MPC and IRV ballot counting[C]//Financial Cryptography and Data Security. Berlin: Springer, 2019: 301-319.

[16] Yang Y, Guan Z S, Wan Z G, et al. PriScore: blockchain-based self-tallying election system supporting score voting[J]. IEEE Transactions on Information Forensics and Security, 2021, 16: 4705-4720.

[17] Harn L, Hsu C, Xia Z, et al. Multiple blind signature for e-voting and e-cash[J]. The Computer Journal, 2023, 66(10): 2331-2338.

[18] Wang B W, Guo F X, Liu Y T, et al. An efficient and versatile e-voting scheme on blockchain[J]. Cybersecurity, 2024, 7: 62.

[19] Huber N, Küsters R, Krips T, et al. Kryvos: publicly tally-hiding verifiable E-voting[C]//Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2022: 1443-1457.

[20] Choi W, Liu X Y, Pang Y X, et al. K-linkable ring signatures and applications in generalized voting[R]. Cryptology ePrint Archive, 2025.

[21] Rivest R L, Shamir A, Tauman Y. How to leak a secret[C]//Advances in Cryptology-ASIACRYPT 2001. Berlin: Springer, 2001: 552-565.

[22] SM2 椭圆曲线公钥密码算法: GM/T 0003-2012[S]. 2012.
SM2 elliptic curve public key cryptographic algorithm: GM/T 0003-2012[S]. 2012.

[23] GM/T 0003.4-2012 SM2 椭圆曲线公钥密码算法 第 4 部分: 公钥加密算法[S]. 2012.
GM/T 0003.4-2012 public key cryptographic algorithm SM2 based on elliptic curves: part 4: public key encryption algorithm[S]. 2012.

[24] Boneh D. The decision Diffie-Hellman problem[C]//Algorithmic Number Theory. Berlin: Springer, 1998: 48-63.

[25] McCurley K S. The discrete logarithm problem[C]//Proceedings of Symposia in Applied Mathematics. Berlin: Springer, 1990: 49-74.

[作者简介]

曾捷伦（1998– ），女，武汉大学博士生，主要研究方向为数字签名、应用密码学等。

加梦（1995– ），女，博士，香港理工大学在站博士后，主要研究方向为应用密码学、区块链安全、分布式安全等。

陈晶（1981– ），男，博士，武汉大学教授，主要研究方向为网络安全、应用密码学、分布式系统安全等。

吕岚曦（2003– ），男，武汉大学硕士生，主要研究方向为应用密码学。

何琨（1986– ），男，博士，武汉大学副教授，主要研究方向为应用密码学、网络安全、云计算安全、区块链安全等。

杜瑞颖（1964– ），女，博士，武汉大学教授，主要研究方向为网络安全、隐私保护等。