doi:10.12068/j.issn.1005－3026.2024.05.002

基于区块链和简短可链接环签名的安全电子投票方案

吴淇毓，杨 帆，周福才，冯 达
（东北大学 软件学院，辽宁 沈阳 110169）

摘 要：围绕当前多数电子投票方案存在中心化、未能实现匿名投票等问题，提出基于区块链和简短可链接环签名的安全电子投票方案．通过设计一个简短可链接环签名方案，结合区块链技术，构造了一个具体、安全、高效的电子投票方案，给出了其系统模型、算法形式化定义以及详细描述．与已有方案相比，该方案允许大规模人员投票，且支持多个投票选项以及自动计票等多个功能．安全分析和实验结果表明，该方案在满足匿名性、不可伪造性和隐私性等更多安全目标的同时在计算开销方面仍具有优势．

关 键 词：电子投票；区块链；简短可链接环签名；安全计算；去中心化
中图分类号：TP 309.2 **文献标志码**：A **文章编号**：1005-3026(2024)05-0619-09

Secure Electronic Voting Scheme Based on Blockchain and Short Linkable Ring Signature

WU Qi-yu, *YANG Fan*, *ZHOU Fu-cai*, *FENG Da*
(School of Software, Northeastern University, Shenyang 110169, China. Corresponding author: ZHOU Fu-cai, E-mail: fczhou@mail.neu.edu.cn)

Abstract: Surrounding the problems that most of the current electronic voting schemes are centralized and unable to achieve anonymous voting, a secure electronic voting scheme based on blockchain and short linkable ring signature is proposed. By designing a short linkable ring signature scheme and combining with blockchain technology, a specific, secure and efficient electronic voting scheme is constructed. The system model, the formal definition of the algorithms, and the detailed description are provided. Compared with the existing schemes, this scheme allows for large-scale voting, and supports multiple voting options and automatic counting of votes. The security analysis and experimental results show that this scheme can meet more security goals such as anonymity, unforgeability, and privacy, while still with advantages in terms of computational overhead.

Key words: electronic voting; blockchain; short linkable ring signature; secure computation; decentralizaiton

投票是人们进行集体决策时常用的一种手段．传统的纸质投票存在诸多弊端，如需要借助纸张记录投票、需要人工统计投票结果、投票过程不透明等．随着互联网的飞速发展，电子投票应运而生．电子投票是一种基于互联网和密码学技术的新型投票方式．投票者可以利用电子设备进行方便快捷的远程无监督投票．与传统纸质投票相比，电子投票具有高效便捷、经济实用等诸多优点．然而，目前电子投票系统存在诸多弊端，如：①中心化程度过高，过度依赖可信第三方机构来计票，一旦被攻击者入侵，投票结果将可能被篡改[1]；②投票者个人隐私容易被泄露[2]，未实

收稿日期：2023-01-02
基金项目：国家自然科学基金资助项目(62072090,62202090,62173101)；辽宁省自然科学基金资助项目(2022-YGJC-24,2022-BS-077)．
作者简介：吴淇毓(1994-)，女，吉林长春人，东北大学博士研究生；周福才(1964-)，男，吉林长春人，东北大学教授，博士生导师．

现匿名投票,将会影响投票决策.

近年来,区块链技术因其具有去中心化、不可篡改、可追溯等特点[3],为安全电子投票的发展提供了新思路. Zhao等[4]首次结合比特币设计了一个电子投票协议,然而其中利用的货币激励制度并不适用于大范围的投票场景. Lee等[5]提出了一个基于区块链的电子投票协议,并通过实例描述了适用于全国性的投票.但该方案依靠了可信第三方来收集选票以及保护投票者隐私. Jason等[6]提出了一种基于盲签名和预付比特币卡的投票协议,缺点是未能很好地保护投票者的身份信息. McCorry等[7]提出了一种适用于小规模董事会选举的电子投票方案,该方案利用以太坊为基础架构,实现了去中心化以及自计票功能.该方案仅支持2个投票候选项. Dimitriou等[8]提出了一种基于区块链的无收据性电子投票方案,通过向每个选民提供一个随机化令牌,确保了投票的抗胁迫性和无收据性等附加属性.但该方案需要基于一个较强的假设,即假设选举机构交给选民的随机化令牌是可信任的. Russo等[9]提出了一个可扩展的、去中心化的区块链电子投票系统.该系统保证了投票的公开性、匿名性,并且可以防止重复投票,缺点是不能保证投票的公平性.

综上所述,现有安全电子投票方案仍不能同时满足匿名性、唯一性、可验证性、公平性、不可伪造性、隐私性以及鲁棒性等要求,此外还存在无法自动计票、不支持多个投票选项、不适合大规模人员投票等问题,从而导致电子投票不能被广泛地推广和应用.因此,如何构造能够同时满足上述多种安全性要求、支持更多功能的安全电子投票方案具有重要的研究意义.

本文旨在结合密码学和区块链技术,设计一个基于区块链和简短可链接环签名的安全电子投票方案,在满足预期安全性要求的同时,降低算法的开销,解决当前功能的局限性,使之更加适用于实际应用.

1 预备知识

1.1 双线性映射

设G_1,G_2和G_T是3个阶为素数p的循环群,设g_1是群G_1的生成元,g_2是群G_2的生成元.设双线性映射e: $G_1\times G_2\to G_T$满足如下属性:

1)双线性.对于循环群内的任何元素$u\in G_1$,$v\in G_2$和整数$a,b\in Z_p$,其中Z_p为$[0,p-1]$的整数,存在$e(u^a,v^b)=e(u,v)^{ab}$.

2)非退化性.$e(g_1,g_2)\neq 1$.

3)可计算性.对于任何元素u,v,存在一个多项式时间算法能够计算$e(u,v)$.

1.2 q-SDH问题

设q为正整数,给定一个$(q+2)$元组$(g_1,g_2,sg_2,s^2g_2,\cdots,s^qg_2)$作为输入,如果任意概率多项式时间算法都不能够以不可忽略的概率计算出$(r,\frac{1}{r+s}g_1)$,其中$s,r\in Z_p^*$,Z_p^*为$[1,p-1]$的整数,则称求解q-SDH问题是困难的.

1.3 可链接环签名

可链接环签名[10](linkable ring signature,LRS)是环签名[11](ring signature,RS)的扩展,它可以额外验证2个签名是否由同一签名者生成.LRS方案通常由4个概率多项式时间算法组成,具体算法如下:

1)密钥生成算法. $\mathrm{LRS.keygen}(1^\lambda)\to(k_\mathrm{p},k_\mathrm{s})$.输入为安全参数$\lambda$,输出为一对公私钥,其中$k_\mathrm{p}$为公钥,$k_\mathrm{s}$为私钥.

2)签名算法. $\mathrm{LRS.sign}(1^\lambda,1^n,k_\mathrm{s},L,m)\to\sigma$.输入为安全参数$\lambda$、组大小$n$、私钥$k_\mathrm{s}$、公钥列表$L$,以及消息$m$,输出为签名$\sigma$.

3)验证算法. $\mathrm{LRS.verify}(1^\lambda,1^n,L,m,\sigma)\to\{1,0\}$.输入为安全参数$\lambda$、组大小$n$、公钥列表$L$、消息$m$以及签名$\sigma$.输出为1,表示签名验证成功,该签名为有效签名;输出为0,表示签名验证失败,拒绝接受该签名.

4)链接算法. $\mathrm{LRS.link}(1^\lambda,1^n,L,m_1,m_2,\sigma_1,\sigma_2)\to\{1,0\}$.输入为安全参数$\lambda$、组大小$n$、公钥列表$L$、消息$m_1$和$m_2$以及签名$\sigma_1$和$\sigma_2$.输出为1,表示$\sigma_1$和$\sigma_2$为同一签名者生成;输出为0,表示$\sigma_1$和$\sigma_2$为不同签名者生成.

1.4 区块链技术

区块链是以区块为单位按照时间顺序连接起来的有序账本.区块链中的每个数据区块都包含区块头和区块体.区块头包含父区块的哈希值、版本号、时间戳、随机数、挖矿难度值、当前区块哈希值和Merkle根.区块体包含交易计数器和交易记录列表.区块链因其巧妙的结构设计,具有去中心化、不可篡改性、匿名性、自治性、开放性等特点.

以太坊是一个开源、分布式的公共可编程区块链平台,其中一个重要概念是智能合约.智能

合约是一段代码和数据的集合,它是由事件驱动的、有状态的、在可信公开的区块链上能够根据预设条件自动运行的程序.其中封装了预设状态和规则代码,当外部的输入数据或事件满足触发条件时,智能合约会按照预设的规则执行相应操作,经过处理的响应最终会被写入区块链中.

2　模型与定义

2.1　系统模型架构

系统模型如图1所示,由组织者、投票者和验证者这三方实体组成.组织者负责发起投票活动、初始化相关参数以及验证投票者身份.投票者在向组织者成功申请注册后,可以参与投票活动.验证者可以看作是以太坊节点,负责对投票者提交的选票进行核实、统计选票并公布结果.

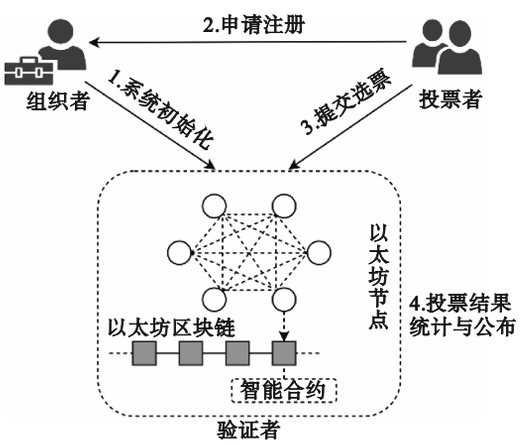

图1　系统模型

Fig. 1　System Model

具体流程分为系统初始化阶段、投票阶段以及投票结果统计与公布阶段.其中系统初始化阶段主要包括组织者编译并部署智能合约、生成系统公共参数、设置投票活动相关参数;投票阶段主要包括投票者向组织者申请注册、提交选票信息;投票结果统计与公布阶段主要包括验证者根据秘密份额计算门限私钥对选票解密、统计解密后的选票并公布投票结果.

2.2　形式化定义

各阶段主要包括5个概率多项式时间算法,其形式化定义如下:

1) 初始化算法. $\mathrm{setup}(1^{\lambda},t)\rightarrow(P,T)$. 以安全参数$\lambda$和系统时间$t$为输入.输出公共参数$P$和时间节点$T$.

2) 注册算法. $\mathrm{register}(P,T,t,\mathrm{ID}_i)\rightarrow k_i$ or $\perp$. 以P,T,t和投票者身份信息ID_i为输入.若注册成功,则输出为密钥信息k_i;否则$\perp$,即中止程序.

3) 密钥生成算法. $\mathrm{keygeneration}(P,t,k_{\mathrm{s},i})\rightarrow(s_i,K_{\mathrm{P}})$. 以$P,t$和投票者私钥$k_{\mathrm{s},i}$为输入.输出为投票者的秘密份额$s_i$和系统门限公钥$K_{\mathrm{P}}$.

4) 投票算法. $\mathrm{vote}(P,t,K_{\mathrm{P}},k_i,v_i)\rightarrow\{1,0\}$. 以$P,t,K_{\mathrm{P}},k_i$和投票选项$v_i$为输入.若投票成功,输出为1;否则输出为0.

5) 结果统计算法. $\mathrm{count}(P,t,\{s_1,s_2,\cdots,s_t\})\rightarrow\mathrm{res}$. 以$P,t$和投票者的秘密份额$\{s_1,s_2,\cdots,s_t\}$为输入.输出为投票结果res.

2.3　设计目标

基于上述系统模型,应当满足以下特性:

1) 匿名性. 不允许知道哪些投票者参与了投票.

2) 唯一性. 每位投票者只能投一票,不能重复投票.

3) 可验证性. 计票阶段结束后,投票结果应可公开验证.

4) 公平性. 没有任何机构或个人能够在投票期结束前获得投票的中间结果,防止其影响剩余投票者的后续投票决策.

5) 不可伪造性. 不允许伪造选票.

6) 隐私性. 选票内容具有私密性,结果公布前任何组织和个人都无法知晓.

7) 鲁棒性. 在输入错误、网络过载或被蓄意攻击的情况下,投票系统应是稳定的.

3　方案的描述

3.1　简短可链接环签名算法

在介绍安全投票方案之前,首先基于LRS构造一个简短可链接环签名方案(short linkable ring signature, SLRS),该方案作为重要的密码学工具用来构造安全高效的电子投票方案.该方案由7个概率多项式时间算法组成,具体描述如下:

1) 初始化算法. $\mathrm{SLRS.setup}(1^{\lambda})\rightarrow p_{\mathrm{p}}$. 给定$\lambda$,选择双线性对$e_1:G_1\times G_1\rightarrow G_2$. 令参数$\delta=(g_1,g_p)$,其中$g_p=(sg_1,s^2g_1,\cdots,s^qg_1)$,随机选择正整数$s\in Z_p^*$,$q$为可以累加的最大环成员个数.根据$\lambda$生成动态累加器$(f,g)$[12],定义为$f:Z_p\times Z_p\rightarrow Z_p$;$g:Z_p\rightarrow G_1$,$u\in Z_p^*$,即$f(u,x)\mapsto(x+s)u$,$g:u\mapsto ug_1$,其中$x$表示变量.选择单向哈希函数$H_0:\{0,1\}^*\rightarrow G_1$和单向哈希函数$H_1:\{0,1\}^*\rightarrow Z_p^*$. 选择群内元素$\gamma_1,\gamma_2,Q\in G_1$,正整数$s_{\mathrm{m}}\in Z_p^*$,计算群内元素$Q_p=s_{\mathrm{m}}Q$,生成公共参数$p_{\mathrm{p}}=(e_1,g_1,p,t,f,g,$

$u,\gamma_1,\gamma_2,Q,Q_p,H_0,H_1\big)$.

2）密钥生成算法. SLRS. keygen$\big(p_{\mathrm{p}},\mathrm{id}_i\big)\rightarrow\big(k_{\mathrm{p},i},k_{\mathrm{s},i}\big)$. 计算哈希值$Q_i=H_0\big(\mathrm{id}_i\big)$,其中$i\in[1,n]$,$n$为环成员数量,$\mathrm{id}_i$为环成员$i$的身份信息. 计算群内元素$d_i=s_{\mathrm{m}}Q_i$,随机选择正整数$x_i\in Z_p^*$,计算环成员$i$的公钥$k_{\mathrm{p},i}=x_i\big(g_1+Q_i\big)$,环成员$i$的私钥$k_{\mathrm{s},i}=x_id_i$.

3）环公钥生成算法. SLRS. pkgen$\big(p_{\mathrm{p}},U\big)\rightarrow k_{\mathrm{Rp}}$. 给定公共参数$p_{\mathrm{p}}$以及环成员的身份信息集合$U$,其中$U=\big\{\mathrm{id}_i\big\}_{i=1}^{n}$. 计算哈希值集合$X=\big\{H_1\big(\mathrm{id}_i\big)\big\}_{i=1}^{n}$,生成环公钥$k_{\mathrm{Rp}}=g\big(f\big(u,X\big)\big)$.

4）环私钥生成算法. SLRS. rskgen$\big(p_{\mathrm{p}},U,k_{\mathrm{s},i}\big)\rightarrow k_{\mathrm{Rs},i}$. 首先计算哈希值集合$X'=\big\{H_1\big(\mathrm{id}_j\big)\big\}_{j=1,j\neq i}^{n}$和哈希值$h_i=H_1\big(\mathrm{id}_i\big)$. 其次计算累加器值$W=g\big(f\big(u,X'\big)\big)$,利用其为环成员$i$生成环私钥$k_{\mathrm{Rs},i}=\big(h_i,k_{\mathrm{s},i},W\big)$.

5）签名生成算法. SLRS. sign$\big(p_{\mathrm{p}},m,k_{\mathrm{s},i},k_{\mathrm{Rp}},k_{\mathrm{Rs},i}\big)\rightarrow\sigma_i$.$m$表示消息,$\sigma_i$表示环成员$i$得到的签名值. 首先验证公式(1)和公式(2)是否成立.

$$e_1\big(h_i+Q_p,k_{\mathrm{s},i}\big)\overset{?}{=}e_1\big(Q,Q\big),\tag{1}$$

$$e_1\big(h_ig_1+g_p,W\big)\overset{?}{=}e_1\big(g_1,k_{\mathrm{Rp}}\big).\tag{2}$$

若两个等式不同时成立,则执行SLRS. sign算法重新生成环私钥;若两个等式均成立,则随机选择正整数$r_1,r_2,k_1,k_2,k_3,k_4,k_5\in Z_p^*$,计算链接标签$L_0=h_ig_1$,$L_1=k_{\mathrm{s},i}g_1+r_1Q_p$和$L_2=r_1Q$. 计算中间参数$U_1=k_{\mathrm{s},i}+r_1g_1$,$U_2=W+r_2Q$,$R=r_1\gamma_1+r_2\gamma_2$,$T_1=k_1\gamma_1+k_3\gamma_2$,$T_2=k_2\gamma_1+k_4\gamma_2-k_5R$. 计算双线性配对值$\Pi_1=e_1\big(Q,U_1\big)^{-k_5}e_1\big(L_2,g_1\big)^{k_2}e_1\big(Q_p,g_1\big)^{k_1}e_1\big(L_0+g_p,W\big)$. 随机选择正整数$a_1,a_2\in Z_p^*$,计算哈希值$c$和双线性配对值$\Pi_2$如下:

$$c=H_1\Bigg(m\parallel k_{\mathrm{Rp}}\parallel L_0\parallel L_1\parallel L_2\parallel R\parallel\Pi_1\parallel e\Bigg(\frac{a_1g_1+g_p}{a_2+r_2},k_{\mathrm{s},i}\Bigg)\Bigg),$$

$$\Pi_2=e\big(g_1,U_2\big)^{-k_5}e\big(g_1,Q\big)^{k_4}e\big(g_p,Q\big)^{k_3}e\big(L_0+g_p,W\big)^{-c}.$$

计算中间参数$s_1=k_1+cr_1$,$s_2=k_2+cr_1h_i$,$s_3=k_3+cr_2$,$s_4=k_4+cr_2h_i$和$s_5=k_5+ch_i$. 令环成员i的签名为$\sigma_i=(m,c,R,U_1,U_2,T_1,T_2,\Pi_1,\Pi_2,s_1,s_2,s_3,s_4,s_5,L_0,L_1,L_2)$.

6）签名验证算法. SLRS.verify$\big(p_{\mathrm{p}},\sigma_i,k_{\mathrm{Rp}}\big)\rightarrow\{1,0\}$. 验证式(3)~式(6)是否成立.

$$T_1\overset{?}{=}s_1\gamma_1+s_3\gamma_2-cR,\tag{3}$$

$$T_2\overset{?}{=}s_2\gamma_1+s_4\gamma_2-s_5R,\tag{4}$$

$$\Pi_1\overset{?}{=}e\big(Q,U_1\big)^{-s_5}e\big(L_2,g_1\big)^{s_2}e\big(Q_p,g_1\big)^{s_1}\cdot e\big(g_1,Q\big)^{c}e\big(Q_p,U_1\big)^{-c}e\big(g_1,V\big),\tag{5}$$

$$\Pi_2\overset{?}{=}e\big(g_1,U_2\big)^{-s_5}e\big(g_1,Q\big)^{s_4}e\big(g_p,Q\big)^{s_3}e\big(g_p,U_2\big)^{-c}.\tag{6}$$

若等式均成立,则签名合法,输出为1,接受签名;若不成立,则输出为0,拒绝签名.

7）链接算法. SLRS.link$\big(\sigma,\sigma'\big)\rightarrow\{1,0\}$. 以签名$\sigma$和$\sigma'$为输入,验证其链接标签$L_0$和$L_0'$是否相等. 若相等,则表示两个签名是来自同一用户,输出为1;若不相等,则表示两个签名为不同用户所签,输出为0.

3.2 安全电子投票算法详细设计

本小节将构造的简短可链接环签名方案与区块链技术相结合,实现安全电子投票方案. 其中具体流程包括系统初始化阶段、投票阶段以及投票结果统计与公布阶段,各阶段算法详细描述如下:

1）初始化阶段.

① 组织者将编写完成的智能合约编译并部署到以太坊区块链,同时公开合约的地址和代码,任何人都可以验证该合约与组织者公布的是否相同,确保投票程序没有被篡改.

② 组织者调用setup$(1^{\lambda},t)\rightarrow(P,T)$初始化算法生成系统公共参数$P$和时间节点$T$,并发送到以太坊智能合约. 合约会在对应的时间节点执行相应的代码,从而确保投票流程按照预定的时间节点进行. 初始化算法具体步骤如表1所示,其中t_{r}表示注册截止时间,t_{b}表示投票开始时间,t_{e}表示投票结束时间,$t_{\mathrm{s,s}}$表示计算秘密份额与门限公钥开始时间,$t_{\mathrm{b,s}}$表示上传秘密份额开始时间,$t_{\mathrm{e,s}}$表示上传秘密份额结束时间,t_{d}表示解密并统计选票时间节点.

③ 组织者设置完上述参数后,以太坊智能合约的状态自动更新.

2）投票阶段.

① 投票者$V_i\big(i\in[1,n]\big)$调用register$\big(P,T,t,\mathrm{ID}_i\big)\rightarrow k_i$ or $\perp$注册算法向组织者申请注册. 注册算法具体步骤如表2所示.

② 注册成功后,投票者$V_i\big(i\in[1,n]\big)$调用

表 1　初始化算法
Table 1　Setup algorithm

初始化算法:$\text{setup}(1^{\lambda}, t)\rightarrow(P,\ T)$
输入:$1^{\lambda}, t$
输出:$P,\ T$
1 Get $p_{\text{p}} \leftarrow \text{SLRS. setup}\left(1^{\lambda}\right)$
2 Generate $P=p_{\text{p}}$
3 Set $t_{\text{r}},\ t_{\text{b}},\ t_{\text{e}}, t_{\text{s,s}},\ t_{\text{b,s}},\ t_{\text{e,s}},\ t_{\text{d}}$ from t
4 Get $T=(t_{\text{r}},\ t_{\text{b}},\ t_{\text{e}}, t_{\text{s,s}},\ t_{\text{b,s}},\ t_{\text{e,s}},\ t_{\text{d}})$
5 Return $P,\ T$

表 2　注册算法
Table 2　Registration algorithm

注册算法:$\text{register}\left(P, T, t, \text{ID}_i\right)\rightarrow k_i$ or $\perp$
输入:P, T, t, ID_i
输出:k_i or $\perp$
1 If $t<t_{\text{r}}$
2　For $i=1$ to $i=n$ do
3　　If $\text{ID}_i=\text{true}$
4　　Get $(k_{\text{p},\ i}, k_{\text{s},\ i})\leftarrow \text{SLRS. keygen}\left(P, \text{ID}_i\right)$
5　　Get $k_{\text{Rp}} \leftarrow \text{SLRS. } k_{\text{Rp}}\text{gen}\left(P, U\right)$
6　　Get $k_{\text{Rs},\ i} \leftarrow \text{SLRS. } k_{\text{Rp}}\text{gen}\left(P, U, k_{\text{s},\ i}\right)$
7　　Save $\left\{\text{ID}_i, k_{\text{p},\ i}\right\}$
8　　Return $k_i=(k_{\text{s},\ i}, k_{\text{Rp}}, k_{\text{Rs},\ i})$
9　Else
10　　Return $\perp$
11　End For
12 Else
13　Return $\perp$

keygeneration$\left(P, t, k_{\text{s},\ i}\right)\rightarrow\left(s_i, K_{\text{P}}\right)$密钥生成算法,为其他投票者 $V_j\left(j\in[1,n], j\neq i\right)$计算子秘密份额、聚合来自其他投票者的子秘密份额并秘密保存,之后利用拉格朗日插值法计算门限公钥[13]并在以太坊区块链公布.密钥生成算法具体步骤如表3所示.其中F_p表示有限域,$E(F_p)$表示F_p上的椭圆曲线,G表示$E(F_p)$上的点.

③ 投票者 $V_i\left(i\in[1,n]\right)$使用随机分配的以太坊账号$E_i$和密码登录投票系统,同时$E_i$中有一定量以太币用于投票交易过程中的消费.

④ 若 $t_{\text{b}}\leqslant t\leqslant t_{\text{e}}$, V_i 调用 $\text{vote}\left(P, t, K_{\text{P}}, k_i, v_i\right)\rightarrow\{1,0\}$投票算法进行投票.投票算法具体步骤如表4所示.其中d表示$[1,p-1]$的一个随机参数,$C\left(v_i\right)$表示选票的加密值,由(C, C_{m}) 2个参数构成.

3)投票结果统计与公布阶段.

表 3　密钥生成算法
Table 3　Key generation algorithm

密钥生成算法:$\text{Keygeneration}\left(P, t, k_{\text{s},i}\right)\rightarrow\left(s_i, K_{\text{P}}\right)$
输入:$P, t, k_{\text{s},i}$
输出:s_i, K_{P}
1 If $t_{\text{b,s}}\leqslant t\leqslant t_{\text{s,s}}$
2　Generate $\left\{a_{i,k}\middle\vert k=1,2,\cdots,t-1\right\}\in F_p,\ a_{i,k}\neq 0$
3　Compute $f_i(x)=K_{\text{S},i}+a_{i,1}x+\cdots+a_{i,t-1}x^{t-1}\ (\text{mod}\ p)$
4　Compute $f_i(0)=K_{\text{S},i}$
5　For $i=1$ to $i=n$ do
6　　For $j=1$ to $j=n$ do
7　　Compute $s_{i,j}=f_i\left(x_j\right)\ (\text{mod}\ p)$
8　　End For
9　End For
10　Compute $\lambda_{i,k}=a_{i,k}\cdot G,\ k\in\{1,2,\cdots,t-1\}$
11　Compute $\text{left}=s_{i,j}\cdot G\ (\text{mod}\ p)$
12　Compute $\text{right}=s_k\cdot G+\sum_{k=1}^{t-1}x_j{}^{k}\ \lambda_{i,k}\ (\text{mod}\ p)$
13　If left = right
14　　Compute $s_i=\sum_{j=1}^{n}s_{j,i}\ (\text{mod}\ p)$
15　Else
16　　Compute $s_{i,j}=f_i\left(x_j\right)\ (\text{mod}\ p)$
17　Compute $\beta_i=s_i\cdot G\ (\text{mod}\ p)$
18　Set $x=0$
19　Compute $K_{\text{P}}=(\sum_{i=1}^{t}s_i\prod_{i=1,i\neq j}^{t}\frac{x-x_i}{x_j-x_i})\cdot G\ (\text{mod}\ p)$
20　Return s_i, K_{P}
21 Else
22　Return $\perp$

表 4　投票算法
Table 4　Voting algorithm

投票算法:$\text{vote}\left(P, t, K_{\text{P}}, k_i, v_i\right)\rightarrow\{1,0\}$
输入:$P, t, K_{\text{P}}, k_i, v_i$
输出:1 or 0
1 If $t_{\text{b}}\leqslant t\leqslant t_{\text{e}}$
2　Choose a random element $d\in[1, p-1]$
3　Compute $C=d\cdot G\ (\text{mod}\ p)$
4　Compute $C_{\text{m}}=v_i+d\cdot K_{\text{P}}\ (\text{mod}\ p)$
5　Set $C\left(v_i\right)=(C, C_{\text{m}})$
6　Get $\sigma(C\left(v_i\right))\leftarrow \text{SLRS. sign}\left(P, C\left(v_i\right), k_{\text{s},i}, K_{\text{Rp}}, K_{\text{Rs},i}\right)$
7　If $1\leftarrow \text{SLRS. verify}\left(P,\ \sigma(C\left(v_i\right)), K_{\text{Rp}}\right)$
8　If $0\leftarrow \text{SLRS. link}(\sigma(C\left(v_i\right)))$
9　　Return 1
10　Else
11　　Return 0
12 Else
13　Return 0
14 Else
15 Return $\perp$

① 若$t>t_e$,即投票活动结束,自动进入投票结果统计与公布阶段.若此时投票者们上传的秘密份额s_i小于τ个,则终止统计;否则,验证者调用$\text{count}\left(P,t,\{s_1,s_2,\cdots,s_t\}\right)\rightarrow\text{res}$结果统计算法对选票进行解密.结果统计算法具体步骤如表5所示.

表5 结果统计算法

Table 5 Result counting algorithm

结果统计算法:$\text{count}\left(P,t,\{s_1,s_2,\cdots,s_t\}\right)\rightarrow\text{res}$
输入:$P,t,\{s_1,s_2,\cdots,s_t\}$
输出:res
1 If $t>t_e$
2 If $t_{b,s}\leqslant t\leqslant t_{e,s}$
3 Generate $\left\{\left(x_1,F(x_1)\right),\left(x_2,F(x_2)\right),\cdots,\left(x_t,F(x_t)\right)\right\}$ by $\{s_1,s_2,\cdots,s_t\}$
4 Compute $F(x)=\sum_{i=1}^{t}s_i\prod_{i=1,i\neq j}^{t}\frac{x-x_i}{x_j-x_i}$
5 Compute $K_S=F(0)$
6 Compute $v_i=C_m-K_S\cdot C\ (\text{mod}\ p)$
7 For $i=1$ to $i=n$ do
8 Compute $v_i=v_i+1$
9 End For
10 Return $r=\{v_i\}$
11 Else
12 Return $\perp$
13 Else
14 Return $\perp$

② 系统调用智能合约中写好的计票方法对解密后的选票进行统计.

③ 系统公布每一个候选项的统计结果并记录在以太坊区块链上,任何人都可以验证最终的投票结果.

4 理论分析及实验

4.1 理论分析

4.1.1 正确性

本方案是正确的,表示如果方案步骤都是正确执行、产生的结果都是按照正确步骤计算得到的,且利用了提供公共可验证性的智能合约,则以太坊区块链上的每一笔交易都会被所有参与的以太坊节点验证和接受,最终公布的投票结果必然是所有选票的正确统计结果.具体证明过程略.

4.1.2 安全性

1) 匿名性. 投票者V_i使用门限加密算法对选票加密得到$C(v_i)$,再使用SLRS签名算法对其签名得到$\sigma\left(C(v_i)\right)$,将$\left(C(v_i),\sigma\left(C(v_i)\right)\right)$发送给以太坊智能合约进行验证,最后发布在以太坊区块上.由签名算法可知,在$\sigma\left(C(v_i)\right)$计算过程中,除了消息$m$和链接标签$L_0$之外,其他参数均为随机产生.而$L_0=h_{\text{id}_j}g_1$,根据椭圆曲线离散对数难题,想要计算出$h_{\text{id}_j}$的概率是可忽略的.因此任何人无法将$v_i$与投票者$V_i$对应,保证了投票的匿名性.

2) 唯一性. V_i投票后,智能合约会验证$\sigma\left(C(v_i)\right)$中的链接标签是否与之前的$L_0$相等.若相等,则表示该选票来自相同投票者,将拒绝该选票.因此,保证了投票的唯一性.

3) 可验证性. ①公共可验证性.在达到t_e时间节点后,门限私钥重构后被公开,因此公众可以解密选票并通过智能合约中的计票方法来重新计票,以此验证投票的最终结果与公布的结果是否一致;②私有可验证性. $C(v_i)$和$\sigma\left(C(v_i)\right)$通过智能合约验证后同时被存储到以太坊区块中,$V_i$随时可以对比自己的投票信息是否与以太坊区块中的记录一致.因此,保证了投票的可验证性.

4) 公平性. V_i须在t_e时间节点之前完成投票,并使用门限公钥K_P加密选票,将密文选票存储在以太坊区块中.只有在结果统计与公布阶段,才会对选票进行解密以及统计公布.在此之前,即投票阶段中,没有投票者可以提前获得中间明文投票结果.因此,在t_e时间节点之前,已投选票不会影响后续投票者的选择,保证了投票的公平性.

5) 不可伪造性. 一方面,基于区块链技术,数据一旦写入以太坊区块链,若敌手想篡改区块中的投票信息,则必须同时修改此区块之后的所有区块信息,伪造的难度和代价极高;另一方面,若敌手伪造选票v_i'使得$C(v_i')\neq C(v_i)$,还需同时调用SLRS.sign算法伪造选票对应的签名$\sigma\left(C(v_i')\right)\neq\sigma\left(C(v_i)\right)$.为此,敌手需在保持环公钥$k_{\text{Rp}}$不变的同时计算出环私钥$k_{\text{Rs},i}$.由于SLRS.rskgen算法中用到的动态累加器[13]是基于q-SDH困难假设构造的,因此,基于q-SDH难题[14],敌手在不知道私钥$k_{s,i}$的情况下计算出环私钥的概率是可忽略的,即伪造一个有效选票合法签名的概率是可忽略的.因此,保证了选票的不可伪造性.

6) 隐私性. 在投票阶段t_b到t_e时间段内,投票选项v_i经过门限加密算法加密后存储在以太

坊区块中.在不知道私钥的情况下,任何人无法获取到有关明文选票的任何信息.因此,保证了选票的隐私性.

7) 鲁棒性.基于以太坊分布式、去中心化等特点,投票阶段产生的数据都同步在以太坊网络节点上,各个节点都是平等且互不影响的.考虑到成本,攻击者几乎不可能破坏所有的以太坊节点.因此,能够确保正常运行,保证投票的鲁棒性.

4.1.3　方案对比

本节将对本文方案与已有相关工作文献[5]、文献[6]和文献[7]进行对比.如表6所示,主要对比方案是否去中心化、可扩展、支持弃权、自动计票、支持多个候选项、满足安全性要求以及适合大规模投票等.

表6　现有方案对比

Table 6　Comparisons with existing schemes

功能	文献[5]	文献[6]	文献[7]	本文方案
去中心化	×	×	√	√
可扩展	√	×	×	√
支持弃权	√	√	√	√
自动计票	×	×	√	√
支持多个候选项	√	√	×	√
七大安全性要求	×	×	×	√
适合大规模投票	√	×	×	√

由表6可知,文献[5]和文献[6]未实现去中心化,这是由于它们虽利用了区块链技术,但却依赖于第三方管理员.同时,文献[5]和文献[6]也并不支持自动计票功能,而文献[7]和本文方案借助以太坊平台,通过智能合约对投票结果进行统计与公布,实现了自动计票功能,并且在整个投票过程中没有第三方中心机构的参与,实现了去中心化.此外,文献[6]由于借助了比特币卡,不具有扩展性,并不适合大规模投票场景.文献[7]也只适合小规模投票,仅支持只有2个候选项的投票活动.而文献[5]和本文方案都考虑到了大规模投票的需求,方案具有可扩展性,适用于投票人数较多、投票选项较多的大规模投票场景.除此之外,文献[5]、文献[6]和文献[7]和本文方案都支持弃权功能,然而文献[5]、文献[6]和文献[7]均不能同时满足七大安全性要求.综上所述,本文方案与现有方案相比在功能性方面具有很大优势.

4.1.4　性能对比

1) 对3.1节构造的简短可链接环签名方案与相关工作文献[15]和文献[16]进行对比.主要对比签名方案的存储开销和计算开销.存储开销对比主要包括链接复杂度和签名大小,如表7所示,其中ω表示环成员的总数量.

表7　签名存储开销对比

Table 7　Comparisons of storage overhead of signatures

性能	文献[15]	文献[16]	本文方案
链接复杂度	$O(\omega)$	$O(\omega)$	$O(1)$
签名大小	$O\left(\sqrt{\omega}\right)$	$O(\omega)$	$O(1)$

由表7可知,文献[16]提出的环签名方案的链接复杂度和签名大小均达到了$O(\omega)$级,与文献[15]$O(\sqrt{\omega})$和本文方案的常量级$O(1)$相比,当环成员数量较多时,存储开销较大.文献[15]提出的非基于身份的环签名方案虽然实现了亚线性的签名存储开销,但显然不如本文方案的常量签名存储开销更适合大规模签名场景.此外,本文方案的链接复杂度也为常量.因此,本文方案的存储开销与同类方案相比具有明显优势.

计算开销对比主要包括生成签名计算开销、验证签名计算开销以及换算成统一运算的总计算开销,如表8所示.其中E表示模幂运算、M表示模乘运算、I表示模逆运算、S表示标量乘法以及B表示双线性映射运算.由表8可知,本文签名方案的计算开销远远小于对比方案且为常量,因此,利用其来构造后续的安全电子投票方案可以使投票方案更加高效、适用于大规模投票等实际应用.

表8　签名计算开销对比

Table 8　Comparisons of computational overhead of signatures

性能	文献[15]	文献[16]	本文方案
生成签名	$\left(8+4\sqrt{\omega}\right)E+\left(4+2\sqrt{\omega}\right)M$	$2\omega E+M+2I$	$7E+19S+9B$
验证签名	$2E+\left(8\sqrt{\omega}+8\right)B$	$2\omega E$	$9E+6S+10B$
总计算代价	$\left(1\,565\sqrt{\omega}+3\,004\right)M$	$(960\omega+25)M$	$5\,993.25M$

2) 对本文投票方案与相关工作文献[5]、文献[6]和文献[7]进行对比.主要对比初始化阶段、投票阶段和结果统计与公布阶段的计算复杂度,如表9所示.

表 9 性能对比

Table 9 Comparisons of performance

阶段	文献[5]	文献[6]	文献[7]	本文方案
初始化	$O(1)$	$O(n)$	—	$O(1)$
投票	$O(1)$	$O(n^2)$	$O(n^2)$	$O(n^2)$
结果统计与公布	$O(n)$	$O(n)$	$O(n)$	$O(n)$

由表 9 可知,在初始化阶段,文献[6]的计算复杂度较大,达到了 $O(n)$,而文献[5]与本文方案的计算复杂度均为常量级 $O(1)$. 在投票阶段,文献[5]的计算复杂度为常量级 $O(1)$,相比之下,文献[6]、文献[7]与本文方案的计算复杂度较高,达到了 $O(n^2)$. 在结果统计与公布阶段,对比文献与本文方案的计算复杂度均为 $O(n)$. 尽管本文方案在投票阶段的计算开销表现得不如文献[5],但这是因为本文方案实现了更多的功能目标,从而牺牲了部分性能. 综上所述,本文方案与现有方案相比在性能方面仍具有竞争力.

4.2 实验结果

对本文方案中的各阶段算法可行性以及效率进行测试. 将配置为 Ubuntu 20.04.1 x64、CPU 双核、4 GB 内存的华为云服务器作为投票发起者节点,将配置为 Windows 10 专业版 x64、Intel(R) Core(TM) i7-10 750 H CPU@2.60 GH、16 GB 内存的本地戴尔 PC 机作为投票者节点,利用本地投票者节点来模拟多个投票者客户端. 其中浏览器使用 Google Chrome 98.0.4758(64 位)和 MetaMask 插件(v10.11.3),使用 Truffle 框架编译和部署智能合约至 Ganache 搭建的以太坊区块链环境. 测试结果均为执行 20 次实验测试的平均值.

1) 对本文方案中主要算法的 Gas 消耗进行测试. 设置 Ganache 中的 Gas Limit 为 8×10^7 Gas、投票总人数 $\upsilon=35$ 以及门限阈值 $\tau=0.6\upsilon$,测试结果如图 2 所示.

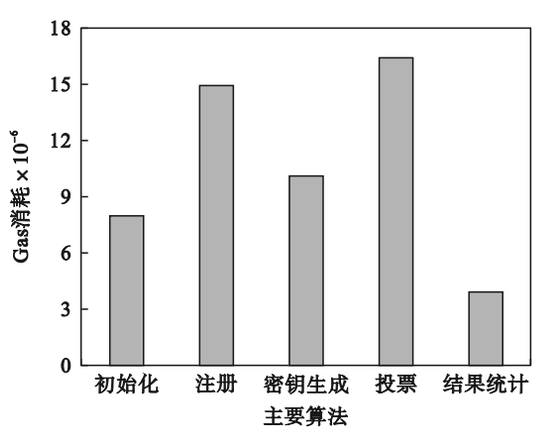

图 2 Gas 消耗测试

Fig. 2 Test of consumed Gas

由图 2 可知,执行主要算法消耗的 Gas 均未超过系统设置的 Gas Limit,满足实际使用需求. 此外,可以看出投票算法消耗的 Gas 较多,这是由于投票者会先在本地前端对选票进行加密和签名,然后提交选票至以太坊智能合约,而智能合约存储这些选票信息需要消耗大量的 Gas.

2) 对本文方案中主要算法的计算开销进行测试. 设置投票总人数 $\upsilon=35$ 以及门限阈值 $\tau=0.6\upsilon$,测试结果如图 3 所示.

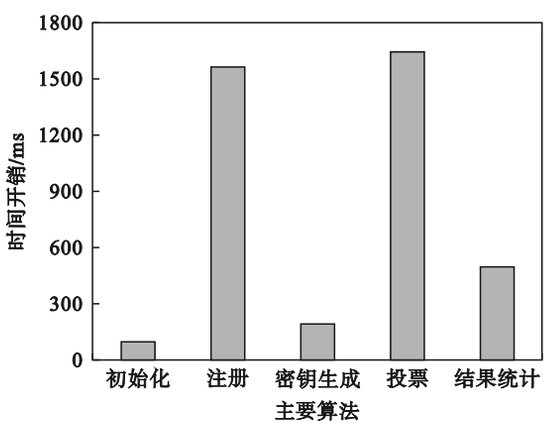

图 3 计算开销测试

Fig. 3 Test of computational overhead

由图 3 可知,各个算法的时间开销均不超过 1 800 ms,能够满足安全电子投票系统的实际使用需求. 其中,注册算法和投票算法的时间开销要远大于其他算法消耗的时间,这是由于简短可链接环签名的生成过程涉及到较多耗时的双线性配对运算,且由于以太坊智能合约缺乏对椭圆曲线密码体制的原生支持,因此涉及到基于椭圆曲线的相关计算均会比其他操作耗时.

5 结 语

本文围绕安全电子投票展开研究,针对现有电子投票方案多数存在中心化、未能实现匿名投票、不支持多个投票选项等问题,提出了一个基于区块链和简短可链接环签名的安全电子投票方案. 首先设计了一个简短可链接环签名方案,在此基础上,结合区块链技术,构造了一个安全高效的电子投票方案. 然后对方案进行了理论分

析和实验评估.与已有方案相比,该方案在安全性、功能和性能方面都具有很大优势.实验评估结果表明了该方案的正确性以及可行性,具有一定的理论和实用价值.

参考文献:

[1] Kubjas I, Pikma T, Willemson J.Estonian voting verification mechanism revisited again [C]//International Joint Conference on Electronic Voting. Bregenz: Springer, 2017: 306−317.

[2] Lubis M, Kartiwi M, Zulhuda S. Current state of personal data protection in electronic voting: criteria and indicator for effective implementation[J].*Telkomnika: Indonesian Journal of Electrical Engineering*, 2018, 16(1): 290−301.

[3] 袁勇,王飞跃.区块链技术发展现状与展望[J].自动化学报,2016,42(4):481−494.
(Yuan Yong, Wang Fei-yue. Blockchain: the state of the art and future trends[J].*Acta Automatica Sinica*, 2016, 42(4): 481−494.)

[4] Zhao Z, Chan T H.How to vote privately using bitcoin[C]// International Conference on Information and Communications Security. Beijing: Springer, 2015: 82−96.

[5] Lee K, James J I, Ejeta T G, et al. Electronic voting service using block-chain[J].*Journal of Digital Forensics, Security and Law*, 2016, 11(2): 123−136.

[6] Jason P C, Yuichi K. E-voting system based on the bitcoin protocol and blind signatures [J]. *IPSJ Transactions on Mathematical Modeling and Its Applications*, 2017, 10(1): 14−22.

[7] McCorry P, Shahandashti S F, Hao F. A smart contract for boardroom voting with maximum voter privacy [C]// International Conference on Financial Cryptography and Data Security. Sliema: Springer, 2017: 357−375.

[8] Dimitriou T.Efficient, coercion-free and universally verifiable blockchain-based voting[J].*Computer Networks*, 2020, 174: 107234.

[9] Russo A, Anta A F, Vasco M I G, et al.Chirotonia: a scalable and secure e-voting framework based on blockchains and linkable ring signatures [C]//2021 IEEE International Conference on Blockchain. Melbourne: IEEE, 2021: 417−424.

[10] Liu J K, Wong D S. Linkable ring signatures: security models and new schemes[C]//International Conference on Computational Science and Its Applications. Heidelberg: Springer, 2005: 614−623.

[11] Rivest R L, Shamir A, Tauman Y.How to leak a secret[C]// Proceedings of the 7th International Conference on the Theory and Application of Cryptology and Information Security. Gold Coast: Springer, 2001: 552 − 565

[12] Nguyen L. Accumulators from bilinear pairings and applications[C]//Topics in Cryptology-CT-RSA 2005. San Francisco: Springer, 2005: 275−292.

[13] Desmedt Y G. Threshold cryptography [J]. *European Transactions on Telecommunications*, 1994, 5(4): 449−458.

[14] Boneh D, Boyen X.Short signatures without random oracles [C]//EUROCRYPT 2004.Interlaken: Springer, 2004: 56−73.

[15] Yuen T H, Liu J K, Au M H, et al. Efficient linkable and/or threshold ring signature without random oracles [J]. *The Computer Journal*, 2013, 56(4): 407−421.

[16] 张文芳,熊丹,王小敏,等.基于RSA公钥密码体制的可选择可转换关联环签名[J].计算机学报,2017,40(5):1168−1180.
(Zhang Wen-fang, Xiong Dan, Wang Xiao-min, et al. Selectively linkable and convertible ring signature based on RSA public key cryptosystem [J]. *Chinese Journal of Computers*, 2017, 40(5): 1168−1180.)