基于全同态加密的区块链电子投票方案

高改梅[1]，邸国霞[1]，刘春霞[1]，杨玉丽[2]，党伟超[1]，张爱贞[1]
（1. 太原科技大学计算机科学与技术学院，山西 太原 030024；
2. 太原理工大学计算机科学与技术学院（大数据学院），山西 晋中 030600）

摘　要：在数字化投票系统中，全同态加密（FHE）与区块链技术的结合保障了电子投票的安全性和隐私性，但现有方案因 FHE 算法复杂的计算过程导致系统整体性能较差，尤其是在计票效率和公平性方面，因此提出一种基于 FHE 的区块链电子投票方案（BCEVS-FHE）。该方案首先通过优化 BFV（Brakerski-Fan-Vercauteren）FHE 算法中噪声因子的影响，降低加解密过程中的计算开销，从而提高计票效率；然后利用 SM2 数字签名算法对投票者生成的选票信息进行签名，确保投票者无法否认其投票行为，防止身份信息假冒与欺诈；接着引入智能合约对加权计票的加权方式进行改进，确保投票者权重的不可伪造性和不可篡改性，保障投票过程的公平公正；最后通过私有区块链方式将所有交易信息都存储到链上，确保整个投票过程不可篡改和可追溯。实验结果表明，该方案不仅在隐私性、机密性、安全性、唯一性和可验证性等安全属性上得到了保障，而且在公平性和可移动性等功能属性上表现出色。综合来看，BCEVS-FHE 满足电子投票协议的安全需求，还具有较高的实际应用潜力，对于数字化投票系统的广泛应用具有重要的研究价值。

关键词：区块链；电子投票；全同态加密；智能合约；签名算法
源代码链接：https://gitee.com/haha0502/test.git
中图分类号：TP309　　**文献标志码：**A　　**DOI：**10.19678/j.issn.1000-3428.0069981

Blockchain Electronic Voting Scheme Based on Fully Homomorphic Encryption

GAO Gaimei[1], DI Guoxia[1], LIU Chunxia[1], YANG Yuli[2], DANG Weichao[1], ZHANG Aizhen[1]
(1. School of Computer Science and Technology, Taiyuan University of Science and Technology, Taiyuan 030024, Shanxi, China;
2. College of Computer Science and Technology (College of Data Science),
Taiyuan University of Technology, Jinzhong 030600, Shanxi, China)

【Abstract】In digital voting systems, the combination of Fully Homomorphic Encryption (FHE) and blockchain technology guarantees the security and privacy of E-voting. The overall performance of existing schemes is constrained owing to the complex computation process of FHE algorithm, especially in terms of vote-counting efficiency and fairness. To address these issues, this paper proposes a Blockchain E-voting Scheme based on Fully Homomorphic Encryption (BCEVS-FHE). This scheme optimizes the Brakerski-Fan-Vercauteren (BFV) FHE algorithm by mitigating the impact of noise factor to reduce the computational overhead during encryption and decryption, thereby improving the vote-counting efficiency. The SM2 digital signature algorithm is used to sign ballot information generated by voters, ensuring that the voters could not deny their voting behavior and preventing identity impersonation and fraud. Furthermore, smart contracts are introduced to improve the weighting method used for vote tallying. Consequently, the unforgeability and non-tampering of voter weights are ensured, thereby guaranteeing the fairness and impartiality of the voting process. Finally, all transaction information is stored in the chain using a private blockchain, ensuring that the entire voting process is tamperproof and fully traceable. Experimental results show that BCEVS-FHE not only guarantees security attributes such as privacy, confidentiality, security, uniqueness, and verifiability but also excels in functional attributes such as fairness and mobility. Overall, BCEVS-FHE meets the security requirements of E-voting protocols and has high potential for practical applications, which is of significant research for the widespread application of digital voting systems.

【Key words】blockchain; E-voting; Fully Homomorphic Encryption (FHE); smart contract; signature algorithm

基金项目：山西省基础研究计划（202303021221017）；太原科技大学纪检监察研究项目（JWYB202310）。

作者简介：高改梅（CCF 会员），女，副教授、博士，主研方向为区块链技术、网络安全、密码学；邸国霞（通信作者），硕士研究生；刘春霞，副教授、硕士；杨玉丽，讲师、博士；党伟超，副教授、博士；张爱贞，硕士研究生。

收稿日期：2024-06-07　　**修回日期：**2024-09-18　**E-mail：**dguoxia@163.com

0 引言

传统纸质投票受限于规模、效率和环保等问题，电子投票（E-voting）应运而生。E-voting 概念在 1981 年首次被 CHAUM[1] 提出，基于公钥密码系统，实现了投票和计票流程的全面覆盖，显著提升了公正性、效率和广泛性。全球约 50 个国家引入 E-voting[2]，但其可靠性问题限制了在全国性大规模投票系统中的应用。因此，提升 E-voting 的可靠性、高效性、公平性、及时性、准确性等要素仍是 E-voting 实施和使用的关键。E-voting 方案按照不同的加密方式分为 3 类：基于 mix-net 的 E-voting[3-5]，基于盲签名的 E-voting[6-8] 和基于同态加密的 E-voting[9-10]。

全同态加密（FHE）技术[11] 具备在无需解密的情况下对选票进行加法和乘法运算的能力，这一特性恰好满足了 E-voting 系统对数据处理的需求，能够有效解决选票安全性问题。但是由于其复杂的数学结构和运算过程，执行计算操作所需的时间和资源开销较大，有很多学者针对该算法进行改进，或将其与其他算法相结合进行应用。FAN 等[12] 通过改进 FHE 中重线性化方法、简化自举步骤，提高了基于 LWE（Learning With Error）问题的 FHE 方案的计算速度。BEHERA 等[13] 为了实现基于 LWE 问题的 FHE 算法，提出了一种高速现场可编程逻辑门阵列（FPGA）。利用线性代数方程分析了一种新的 FHE 算法，但是当明文模数增大时，乘法噪声依旧会线性增长。KIM 等[14] 针对 BFV（Brakerski-Fan-Vercauteren）中的关键瓶颈——同态乘法过程进行了优化，不仅减少了算法复杂度，还引入了延迟缩放技术和剩余数系统（Residue RNS）中的解密优化策略，从而提升了算法的实用性。YUAN 等[15] 将 Paillier 同态加密算法与去中心化方案结合，提出了一种基于同态加密和去中心化的 E-voting 方案，选票的签名和两层加密避免了选票在传输过程中的篡改，利用第三方计票中心进行计票，计票效率不高且第三方公正性和可信度可能导致计票结果不公正或被篡改，影响选举的公平性和合法性。

为保证 E-voting 的高效性和公平性，有学者提出利用区块链技术的去中心化、开放性、匿名性和不可篡改等特性，为 E-voting 提供了强有力的技术支撑。KHAN 等[16] 通过区块链技术和现有的投票机制，建立一种去中心化的 E-voting 方法，取代集中式 E-voting，保证投票者的公平性以及数据传输和验证的安全性。DE MIRANDA 等[17] 提出了一种基于区块链技术、完全同态加密（TFHE）、代币化及权益证明机制的去中心化投票系统，保障了投票隐私和系统可持续性，支持无须可信第三方的验证过程。WANG 等[18] 利用区块链技术在无线网络环境中实现自计票 E-voting 系统，该系统利用 ElGamal 同态加密算法对选票进行加密，通过智能合约执行整个过程，将全部文件和信息存储在区块链上，但是在公开透明的同时会带来隐私泄露的风险，并且大量的加密选票和智能合约执行影响区块链网络实时性和高效性。NAIDU 等[19] 提出了一种利用区块链技术和同态加密的新的 E-voting 系统，侧重于加密选民信息，而不是加密选票信息。这种新的 E-voting 系统的主要优点是它在保护选民隐私的同时可以对投票结果进行统计分析，但是这种方式没有发挥同态加密的最大优势。杨亚涛等[20] 提出区块链与 BFV FHE 相结合的 E-voting 系统，加密选票实现密文统计，确保投票过程抗操纵，在注册和选票阶段采用 SM2 签名算法以及 SM4 加密算法，双方共同监管选票，确保信息分离，在提交选票后，利用区块链智能合约验证合法性并自计票，取代可信第三方，降低虚假计票和信息泄露风险，但是该系统加解密效率较低，不支持加权计票，需要可信双方监管完成选票的获取，存在监管双方达成某种默契对选票进行不公正处理的风险。综上所述，E-voting 的研究取得了显著的进展，但仍存在计票效率低、公平性不足的问题，为解决 E-voting 存在的问题，本文提出基于全同态加密的区块链电子投票方案（BCEVS-FHE）。本文的主要贡献如下：

1）针对 BFV FHE 计算开销大的问题，深入剖析了算法中影响噪声增长的各项因子，通过针对性地对计算方式进行优化和改良，提出了新的算法 BFV-New。BFV-New 的提出旨在显著提升整个投票流程中加解密以及计票操作的效率，从而缓解原算法在计算开销方面的压力，推动区块链 E-voting 系统的实际应用与发展。

2）优化了 SM2-1 数字签名算法，投票者使用其私钥对选票进行签名，并使用其公钥验证签名的真实性。该优化显著提升了签名和验证的效率，并有效防止选票被篡改或伪造的风险，从而保护了投票者的权益。

3）为了确保加权计算过程中的公平性和安全性，优化了加权计算方法，并利用智能合约实现了自动化的加权计算和计票功能，从而消除了投票者手动提交权重以及依赖第三方计票中心的需要。智能合约的自动执行确保了计票过程的客观性和准确

性,预防和打击了任何人为干预和舞弊行为,为投票过程提供了更加可靠的技术支持。

1 预备知识

1.1 SM2-1 数字签名算法

SM2 算法基于椭圆曲线密码学(ECC)算法的公钥密码算法标准,由中国国家密码管理局发布,包括 SM2-1 数字签名算法、SM2-2 密钥交换协议和 SM2-3 公钥加密算法。在 SM2 算法中,公钥和私钥可用于执行加密和解密操作,同时也可用于生成和验证数字签名[21]。

SM2-1 数字签名算法是一种高效、安全和可靠的加密技术,在 BCEVS-FHE 中投票者生成选票之后利用 SM2-1 数字签名算法的私钥对选票进行签名,确保了选票的合法性和真实性,在选票被计入前,利用公钥对签名进行验证,确保消息的完整性和不可否认性。

1.2 BFV FHE 算法

2012 年,FAN 和 VERCAUTEREN 将基于 LWE 困难问题的 FHE 方案修改为基于 RLWE(Ring Learning With Errors)困难问题的 FHE 方案,即产生了 BFV FHE 算法[13]。该算法建立在多项式环 $R=\mathbb{Z}[x]/(x^N+1)$ 上。在 BCEVS-FHE 系统中,对选票进行加解密和计票计算,主要包括表 1 中的私钥生成、公钥生成、密钥生成、加密算法、解密算法、同态加法运算、同态乘法运算 7 个算法。

表 1 BFV FHE 算法过程

Table 1 BFV FHE algorithm process

算法	过程
私钥生成算法	随机选取 s,s 为一个系数为−1、0、1 的多项式,输出私钥 $sk=s$
公钥生成算法	输入私钥 s,选取 a 为随机多项式,其系数模为 q;e 为一个足够小的噪声多项式,输出公钥 $pk=([-as+e]_q,a)$
密钥生成算法	输入私钥 s,选取 a_i 和 e_i,a_i 和 e_i 的生成方式同上,$i\in\{0,1,\cdots,l\}$,输出 $rlk=([-(a_is+e_i)+w^is^2]_q,a_i)$,其中,$w$ 为对数的底数,$l=\lfloor\log_w(q)\rfloor$
加密算法	输入明文 m,公钥 pk,同时构建 3 个随机量:e_1,e_2,u,前两个 e_1,e_2 为噪声,u 和 s 类似,为系数−1、0、1 类似的多项式,计算 $ct=([\Delta[m]_t+pk_0u+e_1]_q,[pk_1u+e_2]_q)$,其中 $\Delta=\lfloor q/t\rfloor$
解密算法	输入私钥 s,$c_0=ct[0]$,$c_1=ct[1]$,输出 $m'=\left[\left\lfloor\frac{t}{q}[c_0+c_1s]_q\right\rfloor\right]_t$
同态加法运算	输入密文 ct 和 ct',输出 $(ct[0]+ct'[0],ct[1]+ct'[1])$
同态乘法运算	输入密文 ct 和 ct',计算 $ct\cdot ct'$ 令: $c_0=\left[\frac{t}{q}ct[0]ct'[0]\right]_q$ $c_1=\left[\frac{t}{q}(ct[0]ct'[1]+ct[1]ct'[0])\right]_q$ $c_2=\left[\frac{t}{q}ct[1]ct'[1]\right]_q$ (重线性化)将 c_2 以 w 为底表示为 $c_2=\sum_{i=0}^{l}c_2^{(i)}w^i$,令 $c'_0=c_0+\sum_{i=0}^{l}rlk[i][0]c_2^{(i)}$, $c'_1=c_1+\sum_{i=0}^{l}rlk[i][1]c_2^{(i)}$,输出 (c'_0,c'_1)

1.3 RLWE 困难问题

RLWE 困难问题分布:给定一个环 R_q,均匀随机选取 m 个环多项式 $a_i\in R_q$,以及一个秘密环多项式 $s\in R_q$,同时从概率分布 χ 中选取噪声 $e_i\in R_q$,其中 χ 是参数为 σ 的离散高斯分布,令 $b_i=a_i\cdot e_i$,则 (a_i,b_i) 为 $R_q\times R_q$ 上的 RLWE 分布 $A_{s,\chi}$[22]。

RLWE 困难问题判定:给定样本分布 (a_i,b_i),能否以不可忽略的优势区分 (a_i,b_i) 为 RLWE 分布 $A_{s,\chi}$ 和 $R_q\times R_q$ 上的随机均匀分布。

1.4 区块链技术

区块链是一种新型的分布式数据库,从本质上整合了共识机制、加密算法、网络通信、分布式架构以及智能合约等一系列新兴的信息技术。这些技术的有机结合赋予了区块链诸多独特属性,包括去中心化、开放性、透明度、可追溯性和信息不可篡改性[23]。私有链作为一种全封闭的区块链,提供了卓越的交易处理能力和速度,大大减少了在网络内达成共识所需的时间,并且私有链通过严格的访问权限控制,确保了在没有相应权限的情况下,任何人均无法获取区块链上的个人数据[24]。

BCEVS-FHE 使用私有区块链技术构建了一个基于 FHE 的 E-voting 系统,为 E-voting 系统

面临的公正性、透明度、可信度问题提供了解决方案。

2 BFV-New FHE 方案

BFV FHE 方案允许用户直接在密文上进行运算，而得到的运算结果与在明文上直接进行相同运算的结果完全一致。这一特性有效保护了用户的敏感信息，避免了隐私泄露的风险。但是，同态运算过程中，密文会引入噪声，尤其在乘法运算中，噪声随运算次数指数增长，影响密文清晰度和安全性。因此，需对噪声进行有效控制，同时降低噪声对加解密效率和投票系统性能的影响。本章深入分析 BFV 方案的噪声，找出影响噪声的关键因素，并探索改进 BFV-New FHE 方案，提高加解密效率，优化投票系统的性能。

2.1 BFV FHE 算法噪声分析

2.1.1 解密算法

对于私钥持有方，得到密文 $ct=([\Delta[m]_t+pk_0u+e_1]_q,[pk_1u+e_2]_q)$ 后，需要对其进行解密，将 ct 展开，再代入公钥 pk，得到 $ct=([\Delta[m]_t-aus+eu+e_1]_q,[au+e_2]_q)$，其中，$eu$、$e_1$ 为噪声因子，aus 为噪声多项式。通过利用私钥 s 可消去 aus 噪声多项式，将私钥 s 代入可得 $c_0+c_1s=\Delta[m]_t+ue+e_1s+e_0=\Delta[m]_t+v \bmod q$，其中 $v=ue+e_1s+e_0$。

然而，当噪声水平达到一定程度时，其在解密过程中将变得不容忽视，这可能导致解密操作失败。为确保解密的准确性，要求噪声满足：

$$\|v\|_\infty<\frac{q}{2t}-\frac{r_t(q)}{2} \tag{1}$$

式中：$r_t(q)=q-t\Delta$。

2.1.2 同态加法运算

同态加法运算实际上是对两个密文 ct 和 ct' 进行相加，其中 $ct=(c_0,c_1)$ 与 $ct'=(c'_0,c'_1)$ 是对明文 m 和 m' 经过加密后的结果：

$$c_0+c'_0+(c_1+c'_1)\cdot s=\Delta[m+m']_t+v+v'-r_t(q)u \bmod q$$

式中：$\|u\|_\infty\leqslant 1$。

加法密文 $ct_{add}=([c_0+c'_0]_q,[c_1+c'_1]_q)$ 是对明文 $[m+m']_t$ 进行加密。

在执行加法操作时，加法噪声接近于密文 ct 与 ct' 各自的噪声之和，所以在叠加的过程中很有可能会突破安全阈值，因此要求噪声 v_{add} 满足：

$$\|v_{add}\|_\infty=\|v+v'+r_t(q)u\|_\infty\leqslant\|v\|_\infty+\|v'\|_\infty+r_t(q) \tag{2}$$

2.1.3 同态乘法运算

同态乘法运算首先计算 R 中的两个密文 $ct=(c_0,c_1)$ 和 $ct'=(c'_0,c'_1)$ 的乘积：

$$(c_0+c_1\cdot s)\cdot(c'_0+c'_1\cdot s)=(\Delta[m]_t+v+qk)\cdot(\Delta[m']_t+v'+qk')=\frac{q}{t}\Delta[m\cdot m']_t+\frac{q}{t}v_{tensor}+\frac{q^2}{t}k_{tensor} \tag{3}$$

式中：$k=(c_0+c_1\cdot s-\Delta[m]_t-v)/q$，$k'=(c'_0+c'_1\cdot s-\Delta[m']_t-v')/q$，$[m]_t\cdot[m']_t=[m\cdot m']_t+tr_m$，$\|r_m\|_\infty\leqslant\delta_R t/2$，$k$ 和 k' 的噪声界限为 $(\delta_R B_{key}+3)/2$，δ_R 为常数，B_{key} 为多项式系数；$k_{tensor}=[m]_t\cdot k'+[m']_t\cdot k+tk\cdot k'+r_m$；$v_{tensor}=\frac{tv\cdot v'}{q}+\frac{t\Delta}{q}([m]_t\cdot v+[m']_t\cdot v)+t(v\cdot k'+v'\cdot k)-r_t(q)([m]_t\cdot k'+[m']_t\cdot k+r_m+\frac{\Delta}{q}[m]_t\cdot[m']_t)$。

$(c_0+c_1\cdot s)\cdot(c'_0+c'_1\cdot s)$ 也可等于 $c_0\cdot c'_0+c_0\cdot c'_1\cdot s+c_1\cdot c'_0\cdot s+c_1\cdot c'_1\cdot s^2$。

令 $ct_{tensor}=(c_0\cdot c'_0,c_0\cdot c'_1+c_1\cdot c'_0,c_1\cdot c'_1)\in R^3$，其次进行缩放操作，缩放操作在 R_q 中进行，并且输出对 q 取模的结果：

$$ct_{scale}=\left[\left[\frac{t}{q}ct_{tensor}\right]\right]_q\in R_q^3 \tag{4}$$

$$\frac{t}{q}ct_{tensor}=\frac{t}{q}(c_{tensor_0}+c_{tensor_1}\cdot s+c_{tensor_2}\cdot s^2)=\Delta[m\cdot m']+v_{tensor}+qk_{tensor} \tag{5}$$

缩放项的四舍五入引入附加的误差项 v_r，使 $ct_{scale_0}+ct_{scale_1}\cdot s+ct_{scale_2}\cdot s^2=\Delta[m\cdot m']+v_{tensor}+v_r \bmod q$，其中 $\|v_r\|_\infty\leqslant(1+\delta_R B_{key}+\delta_R^2 B_{key}^2)/2$。因此，总乘法的噪声界限为 $v_{mult}=v_{tensor}+v_r$，即：

$$\|v_{mult}\|_\infty\leqslant\frac{\delta_R t}{2}\left(\frac{2\|v\|_\infty\|v'\|_\infty}{q}+(4+\delta_R B_{key})(\|v\|_\infty+\|v'\|_\infty)+r_t(q)(\delta_R B_{key}+5)\right)+\frac{1+\delta_R B_{key}+\delta_R^2 B_{key}^2}{2} \tag{6}$$

密文的度和大小在每次乘法之后都会增加，因此需要执行密钥切换操作使度为 2 的密文重线性化程度为 1 的密文，降低未来的通信和计算成本。

2.2 BFV FHE 算法噪声优化

观察上述式(1)、式(2)、式(6)中的噪声界限，可以明显看出 BFV FHE 算法的噪声水平受到 $r_t(q)$ 因子的显著影响。特别是随着明文模量 t 的增加，

噪声增长的速度显著加快。为了解决这一问题，本节提出一种更合理的实例化方法，旨在有效降低噪声，从而提高 BFV 算法的安全性和性能。

观察 BFV 算法中乘法后的噪声界限[即式(6)]，注意到：在明文模数较大时，BFV 中的噪声增长尤为迅速。本文重点关注对噪声幅度起主导作用的项，以此来简化噪声界限。更具体地，如果假设两个密文的噪声由 V 限制，并且 $B_{\text{key}}=1$，那么两个密文相乘后所产生的噪声大小可以合理地近似如下：

$$\delta_R t\left((5+\delta_R)V+\frac{r_t(q)}{2}(\delta_R+5)\right)+\frac{\delta_R^2}{2}\approx \delta_R^2 t\left(V+\frac{r_t(q)}{2}\right) \tag{7}$$

由于噪声随着同态乘法显著增长，在执行完第一次乘法之后，V 变大，$r_t(q)/2<t$。但是，这对于第一次乘法本身并不一定成立，因为 BFV 中的新密文的噪声由 $B_{\text{err}}(2\delta_R B_{\text{key}}+1)\approx 2\delta_R B_{\text{err}}$ 限制。同态加密标准[25]建议使用 $\sigma_{\text{err}}=3.2$ 的误差分布。因此，新密文的噪声估计为 $V_{\text{init}}=2\times 6\times 3.2\times 2\sqrt{N}<77\sqrt{N}$。实际上，由于维度 N 通常不超过 2^{16}，因此新的密文噪声大小不超过 14 bit，而 $r_t(q)$ 可以与 t 一样大。因此，当 $r_t(q)>2^{15}$ 时，它会导致 BFV 中第一次乘法之后较大的噪声增长。举例来说，如果 $t=2^{32}$ 并且 $r_t(q)\approx t/2\approx 2^{31}$，那么第一次乘法之后的噪声将比 $r_t(q)<V_{\text{init}}$ 的情况至少大 16 bit。然而，这种增长不会导致后续乘法操作的噪声进一步急剧增加，因为如式(7)所示，乘法操作后的噪声增长在 V 中是线性增长的。尽管如此，该 16 bit 的噪声差会持续影响后续的计算过程，直到计算结束。特别是在 $t=2^{60}$ 的情况下，这一差异最终可能累积到至少 44 bit，从而使得噪声迅速增长。

解决 $r_t(q)$ 因子带来的影响最直接的方法是选择模 q_i，使得 $q_i=1 \bmod t$，则 $r_t(q)=1$。然而，这种方法的实施却面临一个挑战：需要筛选并测试大量的明文模 t，这无疑增加了问题的复杂性。替代方案是尝试采用特定的值，如 $r_t(q)<\sqrt{N}$，通过反复试验找到满足条件的 $r_t(q)$，但这种方法更多的是一种试错过程，而非真正的解决方案。因此，为了更有效地解决这个问题，需要提出一种更为合理且切实可行的方法。

在加解密和同态加法、乘法时使用的是 Δm(即 $\left\lfloor\frac{q}{t}\right\rfloor m$)，而不是 $\left\lfloor\frac{q}{t}m\right\rfloor$，所以在 $\left\lfloor\frac{q}{t}\right\rfloor$ 与 $\frac{q}{t}$ 之间存在误差，设为 $r_t(q)$，令 $r_t(q)=q-t\Delta$，只要 $r_t(q)\approx 0$，那么就可以减少 BFV 中的噪声增加，这样在解密算法、同态相加、同态相乘中都能实现，且该加密函数还能显著简化 BFV 同态乘法的噪声分析和估计。

2.2.1 解密算法

通过第 2.1.1 节的分析可得原本的解密算法如下：

$$m'=\left[\left\lfloor\frac{t}{q}[c_0+c_1 s]_q\right\rceil\right]_t=\left[\left\lfloor\frac{t}{q}[\Delta[m]_t+v]_q\right\rceil\right]_t \tag{8}$$

现在将 $\Delta m=\left\lfloor\frac{q}{t}m\right\rfloor$ 代入解密算法中得到：

$$m'=\left\lfloor\frac{t}{q}[c_0+c_1 s]\right\rceil=\left\lfloor\frac{t}{q}\left(\frac{q}{t}[m]_t+v+\varepsilon+kq\right)\right\rceil=[m]_t+\left\lfloor\frac{t}{q}(v+\varepsilon)\right\rceil+tk=[m]_t+\left\lfloor\frac{t}{q}(v+\varepsilon)\right\rceil \bmod t \tag{9}$$

式中：$k\in R$，$\left\lfloor\frac{q[m]_t}{t}\right\rceil=\frac{q[m]_t}{t}+\varepsilon$，$\|\varepsilon\|_\infty\leqslant\frac{1}{2}$。

因此，当 $\frac{t}{q}\|v+\varepsilon\|_\infty<\frac{1}{2}$ 时解密就是正确的，即：

$$\|v\|_\infty<\frac{q}{2t}-\frac{1}{2} \tag{10}$$

2.2.2 同态加法运算

在加法运算过程中，密文 ct 与 ct' 之间的相加会导致噪声也进行叠加，这种噪声的累积如果过大，很可能突破预设的安全阈值。一旦噪声水平超出这个安全范围，在进行四舍五入等操作时，就可能直接导致计算错误，从而影响整个加密通信的准确性和安全性。因此，在加法运算中，必须严格控制噪声的叠加和增长，以确保加密通信的可靠性。

通过上节分析可得原本加法噪声如下：

$$\|v_{\text{add}}\|_\infty=\|v+v'+r_t(q)u\|_\infty\leqslant\|v\|_\infty+\|v'\|_\infty+r_t(q) \tag{11}$$

将 $\Delta m=\left\lfloor\frac{q}{t}m\right\rfloor$ 代入后的加法运算如下：

$$c_0+c_1\cdot s+c_0'+c_1'\cdot s=\frac{q}{t}([m]_t+[m']_t)+v+v'+\varepsilon+\varepsilon'=\frac{q}{t}([m+m']_t+tu)+v+v'+\varepsilon+\varepsilon'=\frac{q}{t}([m+m']_t)+v+v'+\varepsilon+\varepsilon' \bmod q \tag{12}$$

此时，加法噪声界限如下：

$$\|v_{\text{new-add}}\|_\infty \leqslant \|v\|_\infty + \|v'\|_\infty + 1 \tag{13}$$

2.2.3 同态乘法运算

通过第 2.1.3 节可得原本的同态乘法运算如下：

$$(c_0 + c_1 \cdot s) \cdot (c_0' + c_1' \cdot s) = (\Delta[m]_t + v + qk) \cdot (\Delta[m']_t + v' + qk') = \frac{q}{t}\Delta[m \cdot m']_t + \frac{q}{t}v_{\text{tensor}} + \frac{q^2}{t}k_{\text{tensor}} \tag{14}$$

现在将 $\Delta m = \left\lfloor \frac{q}{t}m \right\rfloor$ 代入进行简化。

$$\|v_{\text{new-mult}}\|_\infty \leqslant \frac{\delta_R t}{2}\left(\frac{2\|\bar{v}\|_\infty \|\bar{v}'\|_\infty}{q} + (4 + \delta_R B_{\text{key}})(\|\bar{v}\|_\infty + \|\bar{v}'\|_\infty)\right) + \frac{1 + \delta_R B_{\text{key}} + \delta_R^2 B_{\text{key}}^2}{2} \tag{16}$$

BFV FHE 算法与 BFV-New FHE 算法的噪声对比分析如表 2 所示。从表 2 中可以看出，BFV-New FHE 算法的解密、同态加法和同态乘法操作中成功消除了 $r_t(q)$ 这一噪声影响因子。这一改进显著缩小了噪声的取值范围，极大降低了因噪声过高而导致的解密失败的风险。综上，BFV-New 加密方案在噪声控制方面表现优异，有效减轻了噪声对加解密过程的不利影响。同时，该技术并未对 RLWE 安全参数选择产生任何影响，因此在确保安全性的前提下，实现了加密性能的提升，为实际应用提供了有力支持。

令 $\bar{v} = v + \varepsilon$，$ct_{\text{tensor}}$ 计算如下：

$$(c_0 + c_1 \cdot s) \cdot (c_0' + c_1' \cdot s) = \left(\frac{q}{t}[m]_t + \bar{v} + kq\right) \cdot \left(\frac{q}{t}[m']_t + \bar{v}' + k'q\right) = \frac{q^2}{t^2}[m \cdot m']_t + \frac{q}{t}v_{\text{new-tensor}} + \frac{q^2}{t}k_{\text{new-tensor}} \tag{15}$$

式中：$v_{\text{new-tensor}} = [m]_t \cdot \bar{v}' + [m']_t \cdot \bar{v} + \frac{t}{q}\bar{v} \cdot \bar{v}' + t(\bar{v} \cdot k' + \bar{v}' \cdot k)$；$k_{\text{new-tensor}} = [m]_t \cdot k' + [m']_t \cdot k + tk \cdot k' + r_m$。

按 t/q 缩放和舍入之后，乘法的噪声如下：

表 2 BFV 与 BFV-New 噪声对比分析

Table 2 Comparative analysis of noise between BFV and BFV-New

过程	算法	噪声
解密算法	BFV	$\|v\|_\infty < \frac{q}{2t} - \frac{r_t(q)}{2}$
	BFV-New	$\|v\|_\infty < \frac{q}{2t} - \frac{1}{2}$
同态加法运算	BFV	$\|v_{\text{add}}\|_\infty \leqslant \|v\|_\infty + \|v'\|_\infty + r_t(q)$
	BFV-New	$\|v_{\text{new-add}}\|_\infty \leqslant \|v\|_\infty + \|v'\|_\infty + 1$
同态乘法运算	BFV	$\|v_{\text{mult}}\|_\infty \leqslant \frac{\delta_R t}{2}\left(\frac{2\|v\|_\infty \|v'\|_\infty}{q} + (4 + \delta_R B_{\text{key}})(\|v\|_\infty + \|v'\|_\infty) + r_t(q)(\delta_R B_{\text{key}} + 5)\right) + \frac{1 + \delta_R B_{\text{key}} + \delta_R^2 B_{\text{key}}^2}{2}$
	BFV-New	$\|v_{\text{new-mult}}\|_\infty \leqslant \frac{\delta_R t}{2}\left(\frac{2\|\bar{v}\|_\infty \|\bar{v}'\|_\infty}{q} + (4 + \delta_R B_{\text{key}})(\|\bar{v}\|_\infty + \|\bar{v}'\|_\infty)\right) + \frac{1 + \delta_R B_{\text{key}} + \delta_R^2 B_{\text{key}}^2}{2}$

3 方案设计

3.1 BCEVS-FHE 系统模型

BCEVS-FHE 系统模型由 5 个实体组成，包括管理员 A、认证机构 CA、投票者 V、智能合约 SC 和区块链 BC。管理员 A：确定投票时间（包括注册时间、投票开始时间、投票结束时间），生成公私钥对，确定候选人。认证机构 CA：为有投票权的用户颁发注册证书 RC。投票者 V：生成选票并进行加密和签名。智能合约 SC：自动进行加权计票并将结果公布在区块链上。区块链 BC：存储投票信息、候选人信息、投票者信息、进行投票结果的核验等。

BCEVS-FHE 系统模型如图 1 所示，投票流程如下：

1）管理员 A 部署智能合约 SC。

2）投票者 V 向认证机构 CA 提出注册申请。

3）CA 核实投票者身份信息后为有权投票的投票者颁发注册证书 RC。

4）投票者 V 得到 RC 后提交给管理员 A，管理员核实通过后为其分配公私钥对。

5）投票者 V 从智能合约 SC 中获取选票信息。

6）生成选票上传到智能合约 SC。

7）智能合约 SC 进入计票阶段，将计票结果上传到区块链 BC。

8）投票者从区块链 BC 上下载计票结果，核验投票结果。

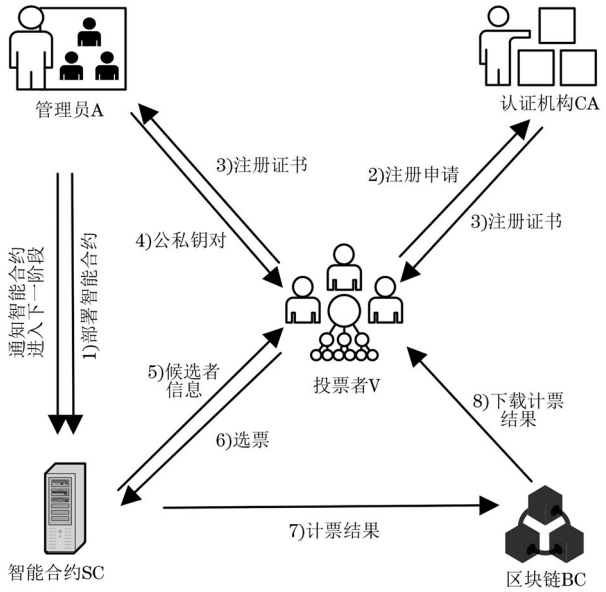

图 1 BCEVS-FHE 系统模型

Fig. 1 BCEVS-FHE system model

投票系统由智能合约精确控制，一旦投票合约启动执行，整个投票流程将自动展开并有序执行。智能合约投票的每一步操作都会被详细记录于日志之中，确保投票过程的透明性和可追溯性。智能合约部署流程如图 2 所示。智能合约投票系统执行流程如下：

1)投票发起者(即管理员)创建管理员智能合约 AdminContract、投票者智能合约 VoteContract、计票智能合约 CountContract，并在合约中确定好候选人名单，之后将合约部署到区块链中。

2)投票者在完成注册后，可从区块链中调用投票合约来进行投票，并且在投票之前对选票内容的合法性进行验证，验证通过后将投票结果加密并按合约规定上传选票。

3)当所有投票者完成投票后，核验计票合约并进行加权计票，然后将密文状态的计票结果上传至区块链。

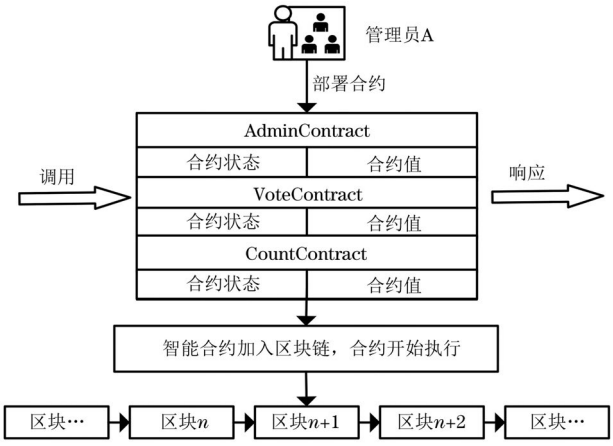

图 2 智能合约部署模型

Fig. 2 Smart contract deployment model

4)计票结束后，智能合约自动解密，将计票结果公布到区块链上，投票者可以重新登录验证结果。

3.2 BCEVS-FHE 投票过程

BCEVS-FHE 投票过程可细分为以下 5 个阶段：初始化阶段，注册阶段，选票生成阶段，计票阶段，验证阶段。系统中关键符号说明如表 3 所示。

表 3 系统中关键符号说明

Table 3 Description of key symbols in the system

符号	描述
$V_i, i\in[1,s]$	投票者集合
$C_j, j\in[1,n]$	候选人集合
A	管理员
CA	认证机构
$SK_{\text{BFV-New}}$	投票者私钥
$PK_{\text{BFV-New}}$	投票者公钥
SK_{sm2}	SM2-1 签名私钥
PK_{sm2}	SM2-1 验证公钥
rlk	BFV 计算密钥
result	计票结果
T_{start}	投票开始时间
T_{end}	投票结束时间

3.2.1 系统初始化

管理员 A 是投票者集合 V_i 共同选举出来的，设置投票过程中用到的参数并且部署智能合约 SC。

首先确定具有投票资格的投票者集合 $V_i(0<i<s)$ 和候选人集合 $C_j(0<j<n)$，投票者信息、候选人信息。注册时间、投票开始时间 T_{start} 和结束时间 T_{end}，部署智能合约 SC。

参数设置完成并发布后，投票者就可以开始注册，同时告知智能合约 SC 开始下一阶段。

3.2.2 注册阶段

投票者 V 向 CA 注册身份，CA 为投票者 V 颁发证书。

投票者 V 提出注册申请并提交个人身份信息给 CA 认证机构，CA 进行身份验证，验证通过后为其颁发注册证书 RC。

然后投票者 V 将 RC 提交给管理员 A，管理员 A 进行审核，审核通过后为其生成唯一的密钥 $PK_{\text{BFV-New}}$、$SK_{\text{BFV-New}}$、PK_{sm2}、SK_{sm2}，审核不通过则不予以生成。

具体生成过程如下：

1)智能合约生成 BFV-New 公私钥对，s 是随机生成的一个多项式，令 $s=SK_{\text{BFV-New}}$，a 为密文空间中随机生成的多项式，其系数模为 q；e 为噪声多项

式,计算公钥 $PK_{\text{BFV-New}}=([-as+e]_q,a)$,即生成的公私钥为 $PK_{\text{BFV-New}}$、$SK_{\text{BFV-New}}$。

2)注册器随机选取私钥 SK_{sm2},根据私钥计算出与之对应的公钥 $PK_{\text{sm2}}=SK_{\text{sm2}}G$,$G$ 为椭圆曲线上的一个点,即生成的公私钥为 PK_{sm2}、SK_{sm2}。

注册完成后管理员 A 公布合法投票者总数及名单,并通知智能合约 SC 进入选票生成阶段。

3.2.3 选票生成阶段

投票者 V 从智能合约 SC 上获取候选人信息后生成选票进行投票,投票完成后对选票进行加密并签名再发送给智能合约 SC。

投票者 V 先从智能合约 SC 上获取加密后的候选人信息(候选人信息是经过哈希计算的,在区块链 BC 上是公开的),利用公钥 $PK_{\text{BFV-New}}$ 解密,解密后选出自己的支持者并生成选票,再使用 BFV-New 同态加密的私钥 $SK_{\text{BFV-New}}$ 加密后得到密文选票。

解密候选人信息,解密过程如下:

1)根据投票者 V 拥有的私钥 $s=SK_{\text{BFV-New}}$,$c_0=ct[0]$,$c_1=ct[1]$,其中 ct 为 BFV-New 加密后的密文,计算:

$$m'=\left[\left\lfloor\frac{t}{q}[c_0+c_1s]_q\right\rceil\right]_t \tag{17}$$

2)对选票进行 BFV-New 加密,加密过程如下:

首先进行密钥生成计算:

$$rlk=([-(a_is+e_i)+w^is^2]_q,a_i) \tag{18}$$

其次利用公钥 $PK_{\text{BFV-New}}$ 加密选票:

$$ct=\left(\left[\left\lfloor\frac{q}{t}m\right\rceil+pk_0u+e_1\right]_q,[pk_1u+e_2]_q\right) \tag{19}$$

3)使用私钥 SK_{sm2} 对选票的哈希值进行签名[21]。

首先计算 $\overline{M}=Z_A\parallel ct$,$e=H_V(\overline{M})$,其中,$Z_A$ 是散列值,$H_V(\ )$是消息摘要为 V bit 的密码杂凑函数;然后选择随机数 $k\in[1,n-1]$,计算椭圆曲线上的点 $(x_1,y_1)=[k]G$,计算 $r=(e+x_1)\bmod n$、$s=[(1+SK_{\text{sm2}})^{-1}\cdot(k-r\cdot SK_{\text{sm2}})]\bmod n$,验证输出的消息 $ct(r,s)$,并提交到智能合约 SC。

3.2.4 计票阶段

投票者 V 提交选票触发智能合约 SC 开始进行验证、验证通过后进行加权计算,且计算完成开始计票。智能合约 SC 对选票进行验证,通过验证的选票才计入。计票结束后用 BFV-New 私钥解密选票结果,将选票结果公布在区块链 BC 上,并共享给所有节点。

1)验证投票者 V 身份合法性和选票合法性。

首先检查 $r',s'\in[1,n-1]$是否成立,如果成立则令 $\overline{M}'=Z_A\parallel M'$,计算 $e'=H_V(\overline{M}')$,$t=(r'+s')\bmod n$,如果 $t=0$,则验证不通过,否则计算椭圆曲线上的点 $(x'_1,y'_1)=[s']G+[t']PK_{\text{sm2}}$,$R=(e'+x'_1)\bmod n$,然后验证 $R=r'$是否成立,如果成立则验证通过,选票是合法的。

2)计票。

智能合约 SC 进行加权计票,加权计算过程如图 3 所示,计算:

$$\text{result}(c_m)=\sum_{i=j=1}^{i=j=m}(\text{BFV-New}_j(c_m)\times w_i) \tag{20}$$

式中:c_m 为候选人;w_i 为 BFV-New 同态加密后的投票者权重。

$\text{BFV-New}_j(c_m)\times w_i$ 表示通过对密文进行同态相乘得到对候选人 c_m 的投票进行加权的结果,计算:

$$\text{BFV-New}_j(c_m)\times w_i=(c_{i0}+c_{i1}\cdot s)(c'_{i0}+c'_{i1}\cdot s) \tag{21}$$

式中:c_{i0}、c_{i1} 分别表示第 i 位投票者 V 选票密文 ct_i 的两位密文;c'_{i0}、c'_{i1}分别表示第 i 位投票者 V 权重密文 ct'_i的两位密文。

$\text{result}(c_m)$表示投票者 V 的选票经过加权计算后再进行同态加法得到最后的投票结果,计算:

$$\text{result}(c_m)=c_{i0}+c_{i1}\cdot s+c'_{i0}+c'_{i1}\cdot s \tag{22}$$

将最终得到每位候选人的计票结果进行一次解密后,得到明文计票结果 Result 为 $\text{Dec}(SK_{\text{BFV}},\text{result}(c_m))$,并将其公布到区块链网络上。

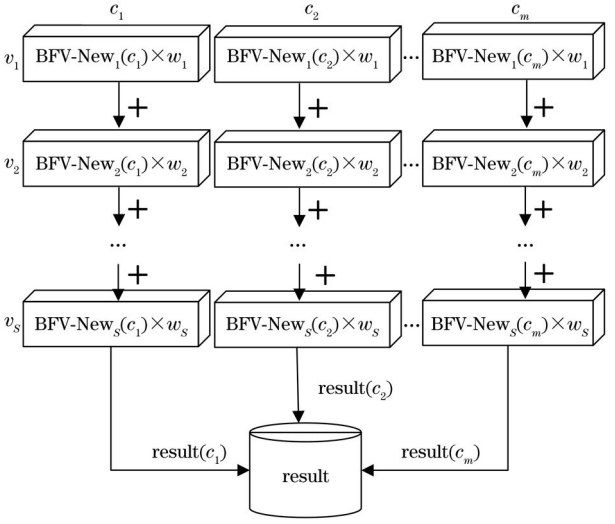

图 3 加权计算过程

Fig. 3 Weighted calculation process

3.2.5 验证阶段

智能合约 SC 将解密后的结果发布到区块链 BC 上,投票者 V 可以主动从区块链 BC 上下载投票结果,进行核验。

4 安全性分析与性能对比分析

4.1 安全性证明

BFV-New 方案是在 BFV FHE 方案的基础上针对噪声进行优化，其安全性仍基于 RLWE 困难问题。BFV-New 的安全性满足选择明文攻击下具有不可区分性(IND-CPA)[25]。

定理 1 假设敌手优势 $\mathrm{Adv}_{\pi,\mathcal{A}}^{\mathrm{IND\text{-}CPA}}(\kappa)=\left|\Pr[b'=b]-\frac{1}{2}\right|$，其中，$\kappa$ 是安全参数。假设在任意多项式时间内敌手的优势 ε 是能够忽略的，即不存在一个算法能够在任意多项式时间内以不可忽略的优势解决 IND-CPA 问题，并且也不存在一个算法能够在任意多项式时间内解决 RLWE 问题。

证明：

1)初始化阶段：算法拥有者称为挑战者 $\mathcal{C}$，算法攻击者称为敌手 $\mathcal{A}$。挑战者 $\mathcal{C}$ 选择一个随机的私钥 SK，利用私钥 SK 和公开的参数来生成公钥 PK，将公钥 PK 公开给敌手。

2)攻击者阶段：敌手 $\mathcal{A}$ 选择任意数量的明文 M，并请求挑战者 $\mathcal{C}$ 返回对应的密文，挑战者 $\mathcal{C}$ 使用公钥 PK 对敌手 $\mathcal{A}$ 选择的明文进行加密，并返回密文给敌手 $\mathcal{A}$。

3)挑战阶段：敌手 $\mathcal{A}$ 选择两个等长的明文 M_1 和 M_2。挑战者 $\mathcal{C}$ 选择一个随机数 $b\in\{0,1\}$，并使用公钥 PK 对 M_b 进行加密，生成挑战密文 $c^*=\mathrm{Enc}_{pk}(M_b,r)$。挑战者 $\mathcal{C}$ 将挑战密文 c^* 发送给敌手 $\mathcal{A}$。

4)猜测阶段：敌手 $\mathcal{A}$ 根据之前获得的密文和挑战密文 $c*$，输出一个 b' 作为对 b 的猜测，若 $b'=b$，则输出 0，$c^*=\mathrm{Enc}_{pk}(M_1,r)$，那么它是真正的密文，在这种情况下，由于敌手 $\mathcal{A}$ 的优势在假设中定义为 ε，因此猜对的概率如下：

$$\Pr[\mathcal{A}(c^*=\mathrm{Enc}_{pk}(M_1,r))=0]=\frac{1}{2}+\varepsilon$$

否则输出 1，$c^*=\mathrm{Enc}_{pk}(M_2,r)$，对敌手 $\mathcal{A}$ 而言完全是随机的，敌手 $\mathcal{A}$ 猜对的概率如下：

$$\Pr[\mathcal{A}(c^*=\mathrm{Enc}_{pk}(M_2,r))=1]=\frac{1}{2}$$

因此，敌手 $\mathcal{A}$ 破解 CP-ABE 问题的优势如下：

$$\mathrm{Adv}_{\pi,\mathcal{A}}^{\mathrm{IND\text{-}CPA}}(\kappa)=$$

$$\frac{1}{2}\Pr[\mathcal{A}(c^*=\mathrm{Enc}_{pk}(M_1,r))=0]+$$

$$\Pr[\mathcal{A}(c^*=\mathrm{Enc}_{pk}(M_2,r))=1]-\frac{1}{2}=$$

$$\frac{1}{2}\left(\frac{1}{2}+\varepsilon+\frac{1}{2}\right)-\frac{1}{2}=\frac{\varepsilon}{2}$$

以上过程证明了敌手能够以 $\frac{\varepsilon}{2}$ 的优势解决 IND-CPA 问题，那么也存在算法能够在任意多项式时间内解决 RLWE 问题，但是这与假设是矛盾的，因此 BFV-New 在 IND-CPA 攻击下是安全的。

4.2 安全性分析

在构建高效、安全且公平的 E-voting 系统时，严格遵循并满足 E-voting 的安全协议标准[26]是至关重要的。为验证 BCEVS-FHE 的可行性，深入分析以下 7 个核心安全属性。

1)隐私性：通过保持投票者和候选人身份的匿名性来确保隐私性。为投票者提供唯一的数字签名以验证选票的真实性，同时确保签名不包含用户识别投票者身份的信息，选票也进行加密处理，无法根据选票信息与投票者身份关联起来。候选人信息以哈希形式记录在区块链中，使得原始数据转换成了一个固定长度的哈希值，而该哈希值在理论上是不可逆的，经过哈希处理的个人隐私数据(哈希值)会被存储在区块链上，任何尝试篡改数据的行为都会被全网节点发现并拒绝，并且 BCEVS-FHE 在本地以太坊私有区块链上进行，以太坊网络由于其庞大的规模和高度分散的哈希算力，使得发动 51%攻击的成本极高，几乎不可能实现，从而保障了候选人身份的隐私性。

2)机密性：整个投票过程都是保密的，选票经过 BFV-New FHE 进行加密，直到得出计票结果前，选票、权重等都是以密文形式存在的，选票内容无法获取和被篡改，在计票结果公布前任何人都无法提前获知最终结果，进一步强化了投票过程的机密性，确保了投票系统的公正性和可信度。

3)安全性：区块链的特性确保了选票无法入侵和篡改。任何数据更改需重新计算该区块及其后续所有区块的哈希值，所需的计算能力是巨大的，这在现实中几乎不可能实现，它使区块链对 BCEVS-FHE 来说是安全的。此外，BCEVS-FHE 中智能合约的安全性通过以下措施得以保障：部署前对合约函数进行独立测试，确保输入条件符合预期，并进行集成测试；采用加密等防御技术处理投票数据；部署后持续监控运行状态和交易数据，并定期进行代码审计。这使得智能合约对 BCEVS-FHE 来说也是安全的。

4)唯一性：投票者有唯一的注册证书 RC，通过验证的投票者只被允许投一票，该票包含在最终投票中，多次投票不作数，如果投票者想投票，他将无

权投票更多。

5)可验证性:区块链技术确保每笔交易都是透明的,全网可验证。任何人都可以验证选举结果与公布的结果相同。

6)公平性:为确保公平,所有选票从投票开始到结束均采取加密形式,以密文的形态存在,投票结果最后会公开上传到区块链,确保投票数据的真实性与可信度。此外,在投票加权的处理过程中,投票者无法得知自己的权重,并且权重是经过 BFV-New 同态加密的,调用智能合约自动执行计算 $\text{BFV-New}_S(c_m)\times w_s$,得到计票结果。在此过程中,投票者无法看到计算过程,确保了计算过程的独立性与保密性,进一步提高选举的公平性。

7)可移动性:投票者可以在任何地方投票。投票系统只有在投票时才可以访问。投票地点不局限于投票系统。本文方案只需要一个具有互联网连接和区块链地址的设备即可访问投票网络。因此,不需要额外的基础设施或投票设备。

4.3 对比分析

将 BCEVS-FHE 与文献[15,17-18,20]方案进行对比分析,如表 4 所示。文献[15]方案采用 Paillier 同态加密算法确保了投票结果在选举结束前不被泄露,但没有利用智能合约取代第三方计票中心,并且无法实现可验证性,增加了潜在的安全风险。文献[17]方案采用了 TFHE 算法,但是该算法基于 Bootstrapping 技术,计算开销比较大,一般仅适用于有限次运算的场景,相比之下 BCEVS-FHE 支持无限次数的同态运算。文献[18]方案利用基于离散对数问题的 ElGamal 加密算法,在量子计算出现后,面临更大的威胁,而 BCEVS-FHE 不仅能够抵抗量子攻击,还显著提升了计票效率,计票时间减少了 22%。文献[20]通过 BFV FHE 算法加密选票,由区块链的智能合约实现自动计票,但是该方案不支持加权计票,且计票效率相对较低。

表 4 不同 E-voting 方案对比分析

Table 4 Comparative analysis of different E-voting schemes

投票方案	同态加密算法	智能合约技术	加权投票	困难性问题	抗量子攻击	计票效率(时间/ms)	可验证性
文献[15]方案	Paillier	否	否	复合剩余类问题	否	6.80	否
文献[17]方案	TFHE	是	否	RLWE 问题	是	1.75	是
文献[18]方案	ElGamal	是	否	离散对数问题	否	1.86	否
文献[20]方案	BFV	是	否	RLWE 问题	是	1.69	是
BCEVS-FHE	BFV-New	是	是	RLWE 问题	是	1.45	是

相较于上述 4 个方案,BCEVS-FHE 显著改进了加解密和计票流程,采用 BFV-New FHE 算法对选票进行加密,这不仅保证了投票数据的安全性,还大幅提升了加解密效率,并且能够抗量子攻击。更为关键的是:利用智能合约完全取代了第三方计票中心的角色实现加权计票功能不仅提高了计票效率,还显著增强了计票过程的安全性。因此,BCEVS-FHE 在保障投票公正性、效率性和安全性方面均取得了显著进步。

4.4 性能分析

4.4.1 BFV-New 同态加密算法分析

通过对 BFV FHE 算法的深入分析发现数论变换(NTT)和乘法操作是开销最大的操作,其计算成本在整体加密效率中占主导地位。如表 5 所示,密钥切换过程中使用的模数数量 m 和密钥切换次数 n 对于这两项操作的复杂度具有显著影响。经过优化后的 BFV-New 算法,NTT 的执行次数由原来的 $14m+7$ 减少到了 $14m$,同时整数乘法操作次数也由原来的 $(10m^2+26m+9)n$ 减少到了 $(9m^2+15)n$,显著提升了系统的运算效率与资源利用率。

表 5 同态乘法运算的计算复杂度对比

Table 5 Comparison of computational complexity of homomorphic multiplication operations

乘法操作	NTT	整数乘
Mult_Old	$14m+7$	$(10m^2+26m+9)n$
Mult_New	$14m$	$(9m^2+15)n$

4.4.2 BFV-New 同态加密算法测试

实验设备为 Intel® Core™ i7-9750H CPU@2.60 GHz 处理器、64 bit 主机,实验环境在 VMware Workstation Pro 上的 Ubuntu 18.04 中搭建。

在基准计算是一个乘积为 $\prod_{i=1}^{2^k} x_i$ 的情况下,比较 BFV 和 BFV-New 算法的噪声增长量和运行时间结果,如表 6 所示,采用 3 bit 混合密钥切换,即 $d_{\text{num}}=3$,t 表示明文模空间,$t=2$ 和 $t=2^{16}+1$,安全参数 $\lambda=2\ 048$,其中,k 表示乘法深度,N 表示环维度,$\text{lb}\ q$ 表示密文模数的大小,e 表示当前噪声幅度。

表 6 乘积 $\prod_{i=1}^{2^k} x_i$ 的噪声增长和运行时间对比

Table 6 Comparison of noise growth and runtime for multiply $\prod_{i=1}^{2^k} x_i$

t	d_{num}	k	params			BFV		BFV-New	
			lb N	lb q_i	lb q	lb e	Time/s	lb e	Time/s
2	3	1	12	31	31	22	0.003	18	0.002
		2	12	45	45	34	0.008	30	0.006
		3	13	31	62	49	0.070	47	0.058
		4	13	39	78	63	0.150	60	0.130
		5	13	47	94	76	0.310	73	0.260
		6	13	55	110	90	0.630	87	0.530
		7	14	44	134	111	3.840	109	3.350
$2^{16}+1$	3	1	13	45	60	44	0.010	36	0.004
		2	13	46	92	66	0.030	63	0.025
		3	14	42	126	403	0.210	95	0.190
		4	14	52	158	132	0.450	125	0.400
		5	14	48	190	161	1.200	166	1.070
		6	14	56	222	189	2.440	184	2.180
		7	14	50	252	220	6.510	214	5.980

通过表 6 观察到，在明文模空间 $t=2$、安全参数 λ 相同的前提下，BFV-New 算法显著降低了噪声的增长速度，当乘法深度 $k=1$、密钥尺寸 q_i 和密文模数 q 相同时，BFV 算法的噪声增长量 lb $e=22$，BFV-New 的噪声增长量 lb $e=18$。BFV-New 算法的计算效率也有提高，当 $k=1$ 时，BFV 算法的运行时间为 0.003 s 而 BFV-New 算法的运行时间为 0.002 s。在不同的乘法深度下，BFV-New 的噪声控制能力和运行效率更高，当 $k=7$ 时，BFV 算法的噪声增长量 lb $e=111$，BFV-New 的噪声增长量 lb $e=109$，BFV 算法的运行时间为 3.840 s，BFV-New 算法的运行时间为 3.350 s。同样，当明文模空间 $t=2^{16}+1$、$k=1$ 且密钥尺寸 $q_i=45$ 和密文模数 $q=60$ 时，BFV 算法的噪声增长量 lb $e=44$，BFV-New 的噪声增长量 lb $e=36$，BFV 算法的运行时间为 0.010 s 而 BFV-New 算法的运行时间为 0.004 s。在不同的乘法深度下，当 $k=7$ 时，BFV 算法的噪声增长量 lb $e=220$，BFV-New 的噪声增长量 lb $e=214$，BFV 算法的运行时间为 6.510 s，BFV-New 算法的运行时间为 5.980 s。因此，消除影响因子有效控制了 BFV-New 算法的噪声增长，提高了解密算法的正确性和高效性。

在基准计算是一个多项式 $\prod_{i=0}^{2} a_i x^i$ 的情况下，比较 BFV 和 BFV-New 算法的噪声增长量和运行时间结果，如表 7 所示，采用 3 bit 混合密钥切换，即 $d_{\text{num}}=3$，t 表示明文模空间，$t=2$ 和 $t=2^{16}+1$，安全参数 $\lambda=2\ 048$。

通过表 7 观察到在乘法深度 k 为 2、4、8、16、32、48、64 的情况下，当明文模空间 t 相同时，BFV-New 算法的噪声增长和运行时间比原来算法更优，当明文模空间不同时，在较大明文模空间 $t=2^{16}+1$ 下对原来 BFV 的噪声幅度的影响更加显著，并且所需时间增加了近 3 倍，而将影响因子消除后，BFV-New 算法的所需时间降低了 50%，提高了解密算法的正确性。

通过表 6 和表 7 的对比发现，当明文模空间 t 相同、乘法深度 k 相同时，多项式乘法所需要的密钥尺寸和密文模数更大，但是相比于 BFV 算法，BFV-New 算法能够降低密钥和密文大小，从而节省存储空间和通信带宽。

为了进一步分析 BFV 和 BFV-New 算法的性能，对比了它们在不同乘法深度和明文模数下同态乘法的运行时间，实验结果如图 4 和图 5 所示。由图 4 和图 5 可以看出，在明文模数 $t=2$ 且深度不同的情况下，BFV-New 算法运行速度较快，在明文模空间增大为 $t=2^{16}+1$ 且深度不同的情况下，BFV-New 算法运行速度较快。在明文模数和深度都相同时，BFV-New 算法的运行时间更短。

表 7 多项式 $\prod_{i=0}^{2} a_i x^i$ 的噪声增长和运行时间对比

Table 7 Comparison of noise growth and running time for polynomial $\prod_{i=0}^{2} a_i x^i$

t	d_{num}	k	params			BFV		BFV-New	
			lb N	lb q_i	lb q	lb e	Time/s	lb e	Time/s
2	3	2	12	31	31	22	0.003	18	0.002
		4	12	47	47	35	0.009	31	0.006
		8	13	33	66	52	0.071	48	0.059
		16	13	41	82	66	0.150	62	0.130
		32	13	50	99	80	0.310	76	0.260
		48	13	58	116	94	0.470	90	0.390
		64	13	58	116	94	0.640	91	0.530
$2^{16}+1$	3	2	13	33	66	40	0.010	35	0.009
		4	13	49	98	76	0.030	67	0.026
		8	14	44	132	107	0.220	100	0.190
		16	14	55	164	138	0.460	130	0.400
		32	14	50	198	167	1.220	161	1.100
		48	14	58	230	198	1.850	190	1.660
		64	14	58	230	199	2.480	191	2.220

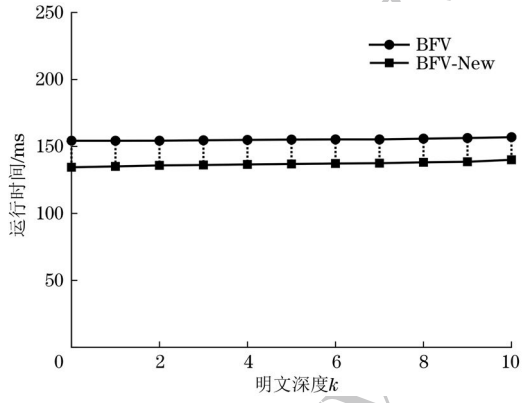

图 4 明文模数 $t = 2$ 时不同深度下同态乘法运行时间对比

Fig. 4 Comparison of homomorphic multiplication running times at different depths when explicit modulus $t = 2$

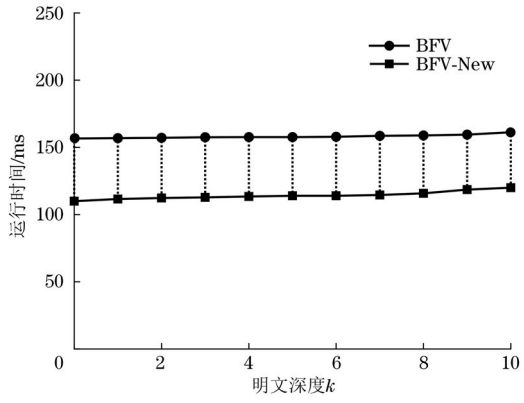

图 5 明文模数 $t = 2^{16} + 1$ 时不同深度下同态乘法运行时间对比

Fig. 5 Comparison of homomorphic multiplication running times at different depths when explicit modulus $t = 2^{16} + 1$

4.4.3 BCEVS-FHE 性能测试

BCEVS-FHE 在本地以太坊私有区块链网络上进行性能测试，系统软件环境配置为 VMware Workstation Pro、Ubuntu 18.04、Palisade 1.10.5、Truffle 4.1.13、Geth 1.8.27、Node 10.3.2、Solc 0.4.24、g++ 6.1。

为全面验证 BCEVS-FHE 的性能，首先进行系统测试，主要聚焦于不同数量的投票者和候选人对计票时间的影响。假设有 2、5、10 名候选人，分别有 10、20、30、40、50 名投票者，对 BCEVS-FHE 的计票时间测试如图 6 所示。从图 6 中可以看出，随着投票者和候选人数量的递增，计票时间呈线性增长，但增长速度相对可控。此外，在投票者数量增加时，候选人数量对计票时间的影响并不显著。

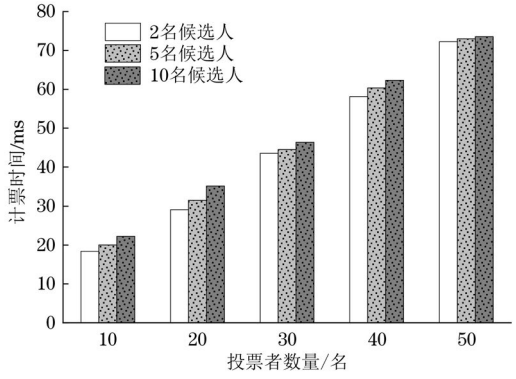

图 6 不同投票者数量下计票时间对比

Fig. 6 Comparison of counting time with different number of voters

基于上述测试结果，可以得出 BCEVS-FHE 在应对一般投票需求时表现出良好的性能和可扩展性，即使在投票者和候选人数量较多的情况下，计票时间仍保持在合理范围内，满足了实际应用中的效率要求。可见，BCEVS-FHE 适用于大多数投票场景，能有效保障投票过程的正确性和高效性。

为了评估 BCEVS-FHE 各阶段所需的运行时间及随投票者数量变化的趋势，记录了初始化阶段、注册阶段、选票生成阶段、计票阶段以及验证阶段的时间消耗如图 7 所示。选择 10、20、30、40 及 50 名投票者，实验结果显示，各个阶段的运行时间与投票者数量呈正相关关系。以 10 名投票者为例：初始化阶段耗时最长，主要工作包括设定投票所需的参数及部署智能合约，耗时为 103.20 ms；注册阶段为投票者生成密钥等，耗时为 43.20 ms；选票生成阶段包括选票的生成、BFV-New 公私钥加密及 SM2-1 签名其哈希值，耗时为 55.36 ms；计票阶段包括验证选票、加权计算并解密公布结果，耗时为 45.00 ms；验证阶段允许有权限的人员核验投票结果，耗时最短，仅为 8.70 ms。

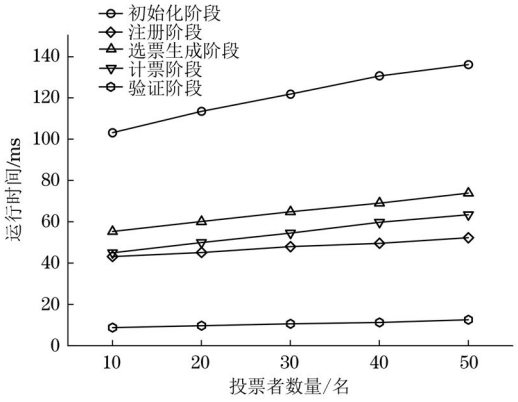

图 7 BCEVS-FHE 各阶段运行时间对比

Fig.7 Comparison of running time for each stage of BCEVS-FHE

为进一步验证 BCEVS-FHE 在计票效率上相较于其他方案具有显著优势，对 BCEVS-FHE 与文献[15，17-18，20]方案进行比较。假设有 1 名候选人，分别有 10、20、30、40、50 名投票者，进行计票时间的对比，结果如图 8 所示。从图 8 中可以看出，随着投票者数量的增加，各方案所需的计票时间也呈现出相应的增长趋势，并且文献[15]方案在计票时间上的增幅远大于其他 3 种方案。与其他 3 种方案相比，BCEVS-FHE 所需的计票时间更短，从而在整体计票效率上实现了一定程度的提升。

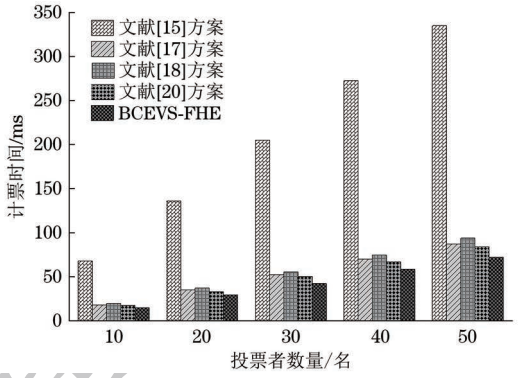

图 8 同类 E-voting 方案计票时间对比

Fig.8 Comparison of counting time of similar E-voting schemes

5 结束语

针对 FHE 与区块链结合的数字化投票方案存在的计票效率低、公平性差的缺陷，本文提出一种基于 FHE 的区块链 E-voting 方案，该方案实现了高效公平的电子计票。该方案使用 BFV-New FHE 算法对选票进行加密处理，保证选票内容的安全性和隐私性。同时，使用 SM2-1 签名算法对选票进行签名，保证选票的真实性。经过加密和签名的选票再上传到区块链，确保信息的透明度和不可篡改性。智能合约技术控制整个投票流程顺利进行，并实现自动化加权计票功能，从而构建了一个安全可靠的 E-voting 方案。由于当前随着投票者和候选人数量的增加，系统的计票时间呈线性增长，因此进一步降低计算开销并确保能够在大规模全国性投票中保持高效性是下一步的研究方向。

参考文献

[1] CHAUM D L. Untraceable electronic mail, return addresses, and digital pseudonyms[J]. Communications of the ACM, 1981, 24(2): 84-90.

[2] IHM Y S, KIM S H. Development of a blockchain-based online secret electronic voting system [J]. IEICE Transactions on Information and Systems, 2022, 105(8): 1361-1372.

[3] ALAM K M R, TAMURA S, RAHMAN S M S, et al. An electronic voting scheme based on revised-SVRM and confirmation numbers[J]. IEEE Transactions on Dependable and Secure Computing, 2021, 18(1): 400-410.

[4] ARANHA D F, BAUM C, GJØSTEEN K, et al. Lattice-based proof of shuffle and applications to electronic voting[C]// Proceedings of Cryptographers' Track at the RSA Conference. Berlin, Germany: Springer International Publishing, 2021: 227-251.

[5] HAINES T, GORÉ R, SHARMA B. Did you mix me? Formally verifying verifiable mix nets in electronic voting[C]// Proceedings of the IEEE Symposium on Security and Privacy. Washington D.C., USA: IEEE Press, 2021: 1748-1765.

[6] KUMAR M, CHAND S, KATTI C P. A secure end-to-end verifiable Internet-voting system using identity-based blind

signature[J]. IEEE Systems Journal, 2020, 14(2): 2032-2041.
[7] KUMAR M, KATTI C P, SAXENA P C. A secure anonymous E-voting system using identity-based blind signature scheme[C]//Proceedings of the 13th International Conference on Information Systems Security. Berlin, Germany: Springer International Publishing, 2017: 29-49.
[8] ZHANG X, ZHANG J Z, XIE S C. A secure quantum voting scheme based on quantum group blind signature [J]. International Journal of Theoretical Physics, 2020, 59(3): 719-729.
[9] FAN X Y, WU T, ZHENG Q H, et al. HSE-voting: a secure high-efficiency electronic voting scheme based on homomorphic signcryption[J]. Future Generation Computer Systems, 2020, 111: 754-762.
[10] FAN X Y, WU T, ZHENG Q H, et al. DHS-voting: a distributed homomorphic signcryption E-voting [C]// Proceedings of the 5th International Conference on Dependability in Sensor, Cloud, and Big Data Systems and Applications. Singapore: Springer, 2019: 40-53.
[11] 熊世强, 何道敬, 王振东, 等. 联邦学习及其安全与隐私保护研究综述[J]. 计算机工程, 2024, 50(5): 1-15.
XIONG S Q, HE D J, WANG Z D, et al. Review of federated learning and its security and privacy protection[J]. Computer Engineering, 2024, 50(5): 1-15. (in Chinese)
[12] FAN J F, VERCAUTEREN F. Somewhat practical fully homomorphic encryption[C]//Proceedings of International Conference on Codes, Cryptology, and Information Security. Berlin, Germany: Springer, 2017: 68-82.
[13] BEHERA S, PRATHURI J R. FPGA-based design architecture for fast LWE fully homomorphic encryption[C]// Proceedings of ICCSDF'21. Singapore: Springer, 2021: 575-584.
[14] KIM A, POLYAKOV Y, ZUCCA V. Revisiting homomorphic encryption schemes for finite fields [C]// Proceedings of the 27th International Conference on the Theory and Application of Cryptology and Information Security. Berlin, Germany: Springer International Publishing, 2021: 608-639.
[15] YUAN K, SANG P, ZHANG S Y, et al. An electronic voting scheme based on homomorphic encryption and decentralization [J]. PeerJ Computer Science, 2023, 9: e1649.
[16] KHAN S, ARSHAD A, MUSHTAQ G, et al. Implementation of decentralized blockchain E-voting[J]. EAI Endorsed Transactions on Smart Cities, 2020, 4(10): 164859.
[17] DE MIRANDA L M B, GARCIA R D, RAMACHANDRAN G S, et al. Blockchain in inter-organizational collaboration: a privacy-preserving voting system for collective decision-making [J]. Journal of Information Security and Applications, 2024, 85: 103837.
[18] WANG L H, LI H L, LI Y N, et al. Self-tallying voting with blockchain in wireless network environment[J]. IEEE Wireless Communications, 2024, 31(5): 142-147.
[19] NAIDU P R, BOLLA D R, G P, et al. E-voting system using blockchain and homomorphic encryption [C]// Proceedings of the IEEE 2nd Mysore Sub Section International Conference (MysuruCon). Washington D.C., USA: IEEE Press, 2022: 1-5.
[20] 杨亚涛, 刘德莉, 刘培鹤, 等. BFV-Blockchainvoting: 支持 BFV 全同态加密的区块链电子投票系统[J]. 通信学报, 2022, 43(9): 100-111.
YANG Y T, LIU D L, LIU P H, et al. BFV-Blockchainvoting: blockchain-based electronic voting systems with BFV full homomorphic encryption [J]. Journal on Communications, 2022, 43(9): 100-111. (in Chinese)
[21] FU J H, ZHOU W H, ZHANG S Z. Fabric blockchain design based on improved SM2 algorithm[J]. International Journal on Semantic Web and Information Systems, 2023, 19(1): 1-13.
[22] 王超, 韩益亮, 段晓巍, 等. 基于 RLWE 困难假设的 NTRU 型代理重加密方案[J]. 密码学报, 2021, 8(5): 909-920.
WANG C, HAN Y L, DUAN X W, et al. NTRU-type proxy re-encryption scheme based on RLWE difficult assumption [J]. Journal of Cryptologic Research, 2021, 8(5): 909-920. (in Chinese)
[23] BELOTTI M, BOŽIĆ N, PUJOLLE G, et al. A vademecum on blockchain technologies: when, which, and how [J]. IEEE Communications Surveys & Tutorials, 2019, 21(4): 3796-3838.
[24] DONG S, ABBAS K, LI M X, et al. Blockchain technology and application: an overview[J]. PeerJ Computer Science, 2023, 9: e1705.
[25] ALBRECHT M, CHASE M, CHEN H, et al. Homomorphic encryption standard[EB/OL]. [2024-05-17]. https://homomorphicencryption.github.io/.
[26] 吴淇毓, 杨帆, 周福才, 等. 基于区块链和简短可链接环签名的安全电子投票方案[J]. 东北大学学报(自然科学版), 2024, 45(5): 619-627.
WU Q Y, YANG F, ZHOU F C, et al. A secure electronic voting scheme based on blockchain and short linkable ring signatures[J]. Journal of Northeastern University (Natural Science), 2024, 45(5): 619-627. (in Chinese)

文字编辑 陆燕菲
栏目编辑 赖玉玲