

密 级：公开

北京科技大学

University of Science and Technology Beijing



博士学位论文

论文题目：面向区块链智能合约的隐私保护技术研究

学 号： B20180328

作 者： 殷红建

专业名称： 计算机科学与技术

2022年11月24日

面向区块链智能合约的隐私保护技术研究

Research on Privacy-Protection Technologies for Blockchain Smart Contract

研究生姓名：殷红建

指导教师姓名：朱岩

北京科技大学计算机与通信工程学院

北京 100083，中国

Doctor Degree Candidate: Hongjian Yin

Supervisor: Yan Zhu

School of Computer and Communication Engineering

University of Science and Technology Beijing

30 Xueyuan Road, Haidian District

Beijing 100083, P.R.CHINA

分类号: TP309.7

密 级: 公开

U D C:

单位代码: 1 0 0 0 8

北京科技大学博士学位论文

论文题目: 面向区块链智能合约的隐私保护技术研究

作者: 殷红建

指导教师: 朱岩 教授 单位: 北京科技大学

指导小组成员: _____ 单位: _____
_____ 单位: _____

论文提交日期: 2022 年 11 月 24 日

学位授予单位: 北京科技大学

致 谢

衷心感谢我的导师朱岩教授对我博士论文和相关研究的支持和指导，朱教授宽阔的学术视野和敏锐的研究思路，给我的论文选题与研究极大的帮助，您求真务实的科研精神和严谨的治学态度是我学习的榜样。

真诚感谢北大冯荣权教授提供的讨论班学习机会，感谢中科院信工所吕克伟副研究员在讨论班上的指导与帮助。

感谢于汝云师姐、陈娥师姐在学习中提出的宝贵意见。感谢郭光来、陆海、卞中山、姚启钧、文友鹏、李东东、石喆、范雨晴、孟疏桐、李依聪、林鸿杰等师弟师妹们对我的帮助。感谢我的好兄弟代贺鹏、陈广对我的包容和鼓励，给我紧张的博士生涯带来无尽欢乐。

感谢我的父母多年来的付出，感谢我的妻子一直的陪伴和理解，有爱就有一切。

摘要

智能合约是一种存储在区块链上并在满足预定条款和条件时自动执行的计算机代码，它为构建自主可控、高效且易于监管的网络交易和价值交换平台提供了基础。然而，近年来由智能合约引起的区块链安全事件频繁发生并带来的巨大经济损失，暴露出智能合约程序代码在有效性、隐私性和可靠性面临诸多挑战，并引起了学术界和产业界的广泛关注。智能合约中私密数据和当事人身份等敏感信息的泄露问题已成为影响交易安全的严重阻碍，本文据此开展面向区块链智能合约的密码隐私保护技术研究，为密码及安全协议在智能合约中的有效应用提供理论依据和技术基础。

本文通过将密码学安全协议构造技术与区块链智能合约隐私需求相结合，对面向智能合约交易隐私的广播加密机制、面向交易数据安全共享的属性基加密机制以及面向特定场景下合约成员关系证明机制展开研究，本文主要创新性工作如下：

1) 提出了一种面向智能合约交易隐私的双模式身份基广播加密（DM-IBBE）方案，针对智能合约敏感数据访问限定到指定人群和非冲突人群的两种隐私需求，通过在拉格朗日插值曲线下对指定集和冲突集上不同插值点与重构曲线上点的选取，设计了支持“选择”和“排它”两种加密模式的 DM-IBBE 构造分别满足上述隐私需求，进而在智能法律合约语言 SPESC 下给出一种具有隐私保护功能的智能合约架构，保证敏感数据以合约条款形式声明隐私并由编译器将预定义的 DM-IBBE 算法链接到智能合约程序中实施保护，在 DDH 假设下证明了 DM-IBBE 方案的语义安全性以及选择加密模式下接收者匿名性，与其它广播加密方案相比，DM-IBBE 具有更加丰富的加密模式以满足智能合约不同的隐私需求。

2) 提出了一种面向属性基加密的去中心化密钥管理与脚本化密文机制，针对区块链去中心化特征与属性基加密中心化密钥管理相冲突的问题，融合两类加法同态的安全多方计算技术，设计了一种基于密文策略和去中心化密钥的属性基加密方案（CP-DK-ABE）及智能合约化的方案构造，进而通过解密过程中的密钥查询、密文逻辑等操作扩展区块链脚本指令系统，在加密过程中实现针对密文策略中复杂逻辑的密文脚本化，分别在 DBDH 和 DLDH 假设下证明了 CP-DK-ABE 方案的语义安全性以及用户私钥生成算法的隐私性，实现了各方对主密钥的共同管理和用户私钥的协同生成、以及脚本解释器对脚本化密文的自动化解密。

3) 提出了一种面向合约成员身份认证的零知识对偶集合成员关系证明 (ZKDMP) 协议, 针对智能合约中动态群组成员的身份认证及判定过程中的用户隐私泄露问题, 通过构造聚合函数实现子集压缩到密码空间元素的表示方法, 将子集表示的规模降低到理论下限, 并引入安全聚合函数 (SAF) 的概念, 在子集压缩到随机元素过程中将元素与集合间属于和不属于的判定问题转变为元素删除和插入下的聚合函数求解问题, 进而给出了基于正负成员关系判定架构下的 ZKDMP 协议及投票合约案例, 证明了该协议在 SDH 假设下的正/负完整性、完备性以及零知识性, 该协议具有支持动态元素添加与删除、集合元素数目不受限制以及更加严格的集合关系证明优势。

上述研究结果表明, 双模式身份基广播加密和基于密文策略和去中心化密钥的属性基加密是保护区块链智能合约交易数据隐私的有效途径, 零知识集合成员关系证明协议进一步保护了交易参与方身份隐私。这些工作将为面向区块链智能合约更加安全且高效的密码方案的构造提供理论与实践支撑。

关键词: 智能合约, 隐私保护, 广播加密, 属性基加密, 零知识证明

Research on Privacy-Protection Technologies for Blockchain Smart Contract

Abstract

Smart contracts are programs stored on blockchain that can be automatically executed according to the predetermined terms of contracts. It provides a basis for building a self-controllable, efficient and easy-monitoring transaction platform. However, in recent years, blockchain security incidents caused by smart contracts have occurred frequently and brought huge economic losses. These security risks expose that the smart contract program codes are facing many challenges in terms of effectiveness, reliability and privacy-protection, thereby attracting extensive attention from academia and industry. The leakage of sensitive information in smart contracts, such as private data and the identity of the parties, has become a serious obstacle to the realization of secure transactions. Based on this, the thesis carries out researches on cryptographic technologies oriented at smart contract privacy protection. These researches will provide theoretical and technical bases for the effective application of cryptographic security protocols in smart contracts.

By combining cryptographical security protocol construction technology with the privacy requirements of blockchain smart contracts, this thesis conducts researches on broadcast encryption for transaction privacy, attribute-based encryption for transaction data security sharing, and set membership proof for specific contracts scenarios. The main contributions of this thesis are as follows:

1) A dual-mode identity-based broadcast encryption (DM-IBBE) scheme is proposed. There are two types of privacy requirements for sensitive data in smart contract, that is, access to sensitive data is limited to the designated people and non-conflict people. By selecting different interpolation points on the designated set and conflict set as well as these points on the reconstructed curve under the Lagrangian interpolation, this thesis proposed the DM-IBBE scheme with selective and exclusive encryption mode to meet the above privacy requirements, respectively. Then, an architecture of smart contracts with privacy protection is introduced under smart legal contract language SPESC. In this architecture, the sensitive transaction data are declared in the form of contract terms and protected by the compiler linking predefined DM-IBBE algorithms into smart contract programs. Moreover, under the decisional Diffie-Hellman assumption, this scheme is proved to be semantically secure and selective encryption mode is anonymous. Compared with other broadcast encryption schemes, DM-IBBE has more varied encryption modes to

meet the different privacy requirements of smart contracts.

2) A decentralized key and scripted ciphertext mechanism for attribute-based encryption is proposed. First, aiming at the problem of the conflict between the decentralization of blockchain and the centralized key management of attribute-based encryption, this thesis proposed a ciphertext-policy decentralized-key attribute-based encryption (CP-DK-ABE) scheme by combining two types of secure multi-party computation with additive homomorphism. Furthermore, the blockchain's script system is extended by adding key query and ciphertext logic opcodes, and complex access policy logic in ciphertext are scripted in the encryption process. In addition, this scheme is proved to be private in key generation and semantically secure under the DBDH and DLDH assumptions, respectively. CP-DK-ABE realizes the decentralized management of master key, the cooperative generation of user private key, and the automatic decryption of scripted-ciphertext by script interpreter.

3) A zero-knowledge dual-membership proof (ZKDMP) protocol is proposed. Aiming at decision of dynamic group members and the problem of user privacy disclosure in the authentication process, this thesis constructed two aggregation functions to compact an arbitrarily-sized subset into an element in a cryptographic space and reach the theoretical lower limit for the representation size of subsets. Also, the concept of secure aggregation function (SAF) is introduced to transform the problem of set membership decision into solving the aggregation function under element deletion and insertion. In addition, this thesis provided detailed security proof of this protocol, including positive completeness, negative completeness, soundness and zero-knowledge. The ZKDMP protocol supports dynamic element addition and deletion, unlimited number of set elements and more strict set membership proof.

All these researches show that the DM-IBBE and CP-DK-ABE are efficient methods to protect the transaction privacy. Furthermore, the ZKDMP protects the identity privacy of parties in the authentication process. These results will provide theoretical and practical supports for more secure and efficient cryptographic technologies in smart contracts.

Key Words: Smart Contract, Privacy-Protection, Broadcast Encryption, Attribute-based Encryption, Zero-knowledge Proof

目 录

致 谢	I
摘 要	III
Abstract	V
插图和附表清单	XI
缩写清单	XIII
1 绪论	1
1.1 研究背景	1
1.2 智能合约系统框架及特征	2
1.3 智能合约安全	4
1.4 国内外研究现状	5
1.4.1 面向智能合约交易隐私的广播加密机制	5
1.4.2 面向交易数据安全共享的属性基加密机制	7
1.4.3 面向特定场景下合约成员关系证明机制	9
1.4.4 目前研究存在的问题	10
1.5 研究目标及研究内容	11
1.5.1 研究目标	11
1.5.2 研究内容及组织结构	12
1.6 研究意义	13
1.7 本章小结	14
2 预备知识	15
2.1 数学基础	15
2.2 困难性假设	16
2.3 交互证明系统	17
2.4 秘密共享方案	18
2.5 区块链智能合约系统	19
2.6 智能法律合约语言	21
2.7 本章小结	22
3 支持交易数据多样化隐私需求的双模式广播加密	23
3.1 研究动机	23
3.2 双模式身份基广播加密方案构造	25
3.2.1 DM-IBBE 方案形式化定义	26
3.2.2 DM-IBBE 构造方法	26

3.2.3 DM-IBBE 方案构造	28
3.3 MD-IBBE 在智能合约交易隐私保护中的应用	31
3.3.1 具有隐私保护功能的智能法律合约实例	32
3.3.2 基于 DM-IBBE 竞拍合约的实现	36
3.4 安全性分析	38
3.4.1 DM-IBBE 方案选择明文攻击下的不可区分性	38
3.4.2 SEM 中选择明文攻击下的匿名性	41
3.5 性能分析	43
3.6 本章小结	47
4 支持可编程密文和去中心化密钥管理的属性基加密	48
4.1 研究动机	48
4.2 区块链数据安全共享系统	49
4.2.1 系统框架	50
4.2.2 区块链数据安全共享的合约化描述	52
4.3 去中心化密钥生成密文策略属性基加密方案	54
4.3.1 去中心化密钥生成构造方法	54
4.3.2 CP-DK-ABE 方案构造	55
4.4 脚本化可编程密文	59
4.4.1 脚本化密文生成	63
4.4.2 脚本驱动的解密	65
4.5 安全性分析	67
4.5.1 Key-generation 算法的隐私性分析	67
4.5.2 CP-DK-ABE 语义安全性分析	71
4.6 性能分析	72
4.7 本章小结	76
5 零知识对偶集合成员关系证明协议	77
5.1 研究动机	77
5.2 子集的安全表示	78
5.2.1 基于零点聚合的子集安全表示	80
5.2.2 基于极点聚合的子集安全表示	84
5.3 集合关系安全证明	88
5.3.1 正集合成员安全判定	89
5.3.2 负集合成员安全判定	91
5.4 零知识对偶集合成员关系证明协议	93

5.5 基于 ZKDMR 的智能合约投票系统.....	95
5.6 安全性分析.....	97
5.7 性能分析.....	102
5.8 本章小结.....	105
6 总结.....	106
参考文献.....	108
作者简历及在学研究成果.....	119
独创性说明	121
关于论文使用授权的说明	121
学位论文数据集.....	123

插图清单

图 1-1 区块链智能合约系统框架	3
图 1-2 论文组织结构	12
图 2-1 交互式证明系统一般表示	17
图 2-2 区块链智能合约系统	19
图 2-3 区块链智能合约开发、部署与执行流程	20
图 2-4 SPESC 合约结构.....	22
图 3-1 合约转换过程	23
图 3-2 标底价竞拍执行流程	32
图 3-3 基于 SPESC 编写的标底价竞拍合约.....	34
图 3-4 SPESC 竞拍合约与 DM-IBBE 算法中间关系.....	35
图 3-5 基于 DM-IBBE 的竞拍合约部署流程	36
图 3-6 getPK 动作执行结果	37
图 3-7 Encrypt 动作执行结果.....	37
图 3-8 Decrypt 动作执行结果.....	37
图 3-9 Extraction 算法时间消耗对比.....	46
图 3-10 Encryption 算法时间消耗对比.....	46
图 3-11 Decryption 算法时间消耗对比.....	47
图 4-1 区块链隐私数据安全分享系统	50
图 4-2 公共账本结构	51
图 4-3 SPESC 编写的区块链数据安全分享智能合约.....	53
图 4-4 用户私钥生成过程	56
图 4-5 脚本系统框架图	59
图 4-6 访问树实例	63
图 4-7 加密算法运行时间对比	75
图 4-8 解密算法运行时间对比	75
图 5-1 安全的聚合函数设计路线	79
图 5-2 正集合成员关系判定示意图	90
图 5-3 负集合成员关系判定示意图	92
图 5-4 对偶集合成员关系零知识证明协议	94
图 5-5 智能合约投票系统模型	96
图 5-6 SPESC 语言编写的投票合约.....	97
图 5-7 聚合函数计算开销	102
图 5-8 ZKDMMP 协议中各个算法时间开销	103
图 5-9 证明者时间开销对比	104
图 5-10 验证者时间开销对比	104

附表清单

表 2-1 基于 SPESC 智能法律合约语法介绍	21
表 3-1 不同加密模式下的构造方法	28
表 3-2 四个现有的广播加密方案与本章提出方案之间对比结果	43
表 3-3 相关广播加密方案计算开销对比	44
表 3-4 相关广播加密方案存储开销对比	45
表 4-1 Bitcoin 脚本中部分操作码	60
表 4-2 新添加操作码	61
表 4-3 属性子密文脚本解密过程	66
表 4-4 数据子密文脚本解密过程	67
表 4-5 现有方案与本章提出的 CP-DK-ABE 之间的比较	73
表 4-6 各方案之间计算开销对比	74
表 4-7 各方案之间存储开销对比	74
表 4-8 CP-DK-ABE 中加解密算法运行时间（单位：毫秒）	75
表 5-1 证明者和验证者计算开销对比	103

缩写清单

ABE	Attribute-based Encryption	属性基加密
AHT	Authenticated Hash Table	认证哈希列表
ANO-CPA	Anonymity under Chosen-plaintext Attacks	选择明文攻击下匿名性
BE	Broadcast Encryption	广播加密
BIoT	Blockchain Internet of Things	区块链物联网
CCA	Chosen Ciphertext Attacks	选择密文攻击
CCA2	Adaptive Chosen Ciphertext Attacks	适应性选择密文攻击
CP-ABE	Ciphertext-policy Attribute-based Encryption	密文策略属性基加密
CP-DK-ABE	Ciphertext-policy Decentralized-key Attribute-based Encryption	密文策略去中心化密钥生成属性基加密
CGA	Ciphertect Generation Algorithm	密文生成算法
DABE	Decentralizing Attribute-based Encryption	去中心化属性基加密
DM-IBBE	Dual-mode Identity-based Broadcast Encryption	双模式身份基广播加密
EEM	Exclusive Encryption Mode	排它加密模式
ETH	Ethereum	以太坊
HABE	Hierarchical Attribute-based Encryption	分层属性基加密
IBBE	Identity-based Broadcast Encryption	身份基广播加密
IBE	Identity-based Encryption	身份基加密
IND-CPA	Indistinguishability under Chosen-plaintext Attacks	选择明文攻击下不可区分性
IoT	Internet of Things	物联网
IPS	Interactive Proof System	交互证明系统
JPBC	Java Pairing Based Cryptography	Java 基于配对密码学
JSON	JavaScript Object Notation	JavaScript 对象简谱
LSSS	Linear Secret Sharing Scheme	线性秘密共享方案

MA-ABE	Multi-authority Attribute-based Encryption	多中心属性基加密
OPF	Oblivious Pseudorandom Function	遗忘伪随机函数
PKI	Public Key Infrastructure	公钥基础设施
Pos-and-Neg	Positive and Negative	正和负
PPT	Probabilistic Polynomial Time	概率多项式时间
RSK	RootStock	根链
SAF	Security Aggregation Function	安全聚合函数
SEM	Selective Encryption Mode	选择性加密模式
SLC	Smart Legal Contract	智能法律合约
SMP	Set Membership Proof	集合关系证明
SPESC	Specification Language of Smart Contract	智能合约规范语言
Tx	Transaction	交易
VM	Virtual Machine	虚拟机
VSS	Verifiable Secret Sharing	可验证秘密共享
WE	Witness Encryption	证据加密
ZKDMP	Zero-knowledge Dual-membership Proof	零知识对偶集合成员关系证明
ZKP	Zero-knowledge Proof	零知识证明

1 绪论

1.1 研究背景

区块链技术以其去中心化、数据防篡改、网络开放、密码安全、共识同步等特点，一经推出就受到社会各界的广泛研究和关注并被应用于数字货币、供应链管理、智慧医疗、物联网等诸多领域。区块链不仅为可信协作、资源共享、公平交易提供新的执行平台，而且正逐步成为我国数字经济发展中不可或缺的信任基础设施。智能合约（Smart Contract）作为第二代区块链的核心技术已经成为区块链领域研究的重点，智能合约的本质是存储在区块链上的计算机代码，在触发预制条件的情况下可被计算机自动执行。该技术支持面向合约编程，具有可编程性、自动执行、有效监督等特点，能够帮助区块链开发人员在分布式应用软件上更加灵活进行程序开发，并为数据要素的管理和价值释放提供了新的思路。

智能合约作为一种自动执行的合约，它是被直接写入代码的买卖双方之间的协议，其中代码化的协议被部署在中心化区块链网络中。由于智能合约被部署在区块链上，这意味着它们可以在不需要第三方参与的情况下执行。此外，想要破坏合约系统则必须具有巨大的计算能力来覆盖整个区块链网络。正是由于智能合约具有自治、去中心化以及安全等特征，该技术助于解决两个或多个当事人之间的不信任问题并被广泛应用于违约合约、信用执行、金融服务、众筹协议等诸多方面。在 Verified Market Intelligence 公司给出的智能合约市场报告¹中指出，2021 年智能合约市场规模已达到 1.495 亿美元，预计到 2030 年该规模将超过 8 亿美元，年复合增长率达到 26.4%。

随着智能合约产业化进展的不断深入，现有大多数区块链应用开发平台已在其分布式账本服务中支持对智能合约开发，例如以太坊平台（Ethereum, ETH）、基于 Bitcoin 的根链（RootStock, RSK）以及 IBM 公司提出的超级账本 Fabric 平台等，这些平台为开发人员提供了创建智能合约应用程序的简单接口，为智能合约的进一步发展提供了良好的开发环境。截至到 2022 年 10 月，仅在以太坊平台上的合约数量就已突破 5100 万，交易数量达到 17.5 亿笔²。然而，区块链以及智能合约不断发展的同时也带来了一些安全问题，例如对 DAO 攻击来盗窃以太币以及对 Parity Multisig 钱包账户冻结等^[1]。因此，在对智能合约研究时，安全性始终是需要考虑的重点。

¹ www.verifiedmarketresearch.com/product/smart-contracts-market/

² <https://dune.com/hildobby/Ethereum-Overview>

1.2 智能合约及其特征

智能合约的概念最早可追溯到 1994 年，Nick Szabo 在文献[2]中对智能合约定义进行了完善，指出智能合约与人工智能无关而是一组以数字形式指定的承诺，其中包括双方履行这些承诺的协议。这些协议通常是通过计算机网络上的程序或其他形式的数字电子产品来实现，因此智能合约比传统纸质协议“更智能”。尽管该文对智能合约构造和密码技术等方面进行了探讨，但由于当时缺少可靠的合约执行环境，导致智能合约的研究只停留在理论层面。

2008 年，区块链技术的出现使得智能合约的研究得到快速发展。区块链以其不变性、去中心化、可追溯性等特点，为智能合约提供了可信的执行环境^[3]。以太坊意识到区块链和智能合约结合的可能，并在白皮书《下一代智能合约与去中心化应用平台^[4]》中重新使用了“智能合约”概念，建立了一整套智能合约的规范与架构，极大地推动了智能合约的发展。智能合约与区块链的结合，使得智能合约天然继承了区块链的特征，即公开透明、无需第三方以及自动执行等。因此，区块链智能合约可被定义如下：

定义 1-1（区块链智能合约^[5]） 区块链智能合约是一种存储在区块链上并在满足预定的条款和条件时自动执行的计算机代码。

存储在区块链上的智能合约被视为自动执行的计算机代码，而且该代码指定了双方之间的协议条款并被嵌入到区块链中^[6]。智能合约的执行不需要人工干预，只要指定的条件被满足就可以被执行，因此也被称为“自执行合约”（Self-enforcing Contract）。需要注意的是，智能合约作为合约条款的代码表示，是一组被编写好的计算机算法，可以作为应用软件的一部分^[7]。区块链智能合约是在计算机网络上执行的，其执行过程通过共识协议（Consensus Protocol）保证正确性而不需要信任方的参与。因此，智能合约可以被理解为一种自动化可执行的协议，在不需要预先构建调解方式下通过共识验证实现合同条款的自我执行。

虽然目前智能合约被视为合约条款的代码表示，但不是法律意义上的合同或合约，不具备法律效力。从法律的角度来看，智能合约本质上可被视为包含一套预先定义规则的计算机协议，这些规则通常包括协助、验证和履行合约的手段，以此体现智能合约的法律特征^[8]。在此基础上智能法律合约（Smart Legal Contract, SLC）的概念被提出，其本质上是一种符合法律的智能合约，具体其定义如下。

定义 1-2（智能法律合约^[9]） 智能法律合约是一种含有合同构成要素、涵盖合同缔约方依据要约和承诺达成履行约定的计算机程序。

智能法律合约是一种介于现实法律合同与智能合约之间的过渡性手段。现实法律合同以自然语言为载体，可翻译成由智能法律合约语言撰写的智能法律合约，进而转化为由智能合约语言编写的智能合约。智能法律合约以条款的形式描述交易，具有较强的可读性，因此本文后续所涉及到的智能合约均用智能法律合约形式进行描述。

基于上述对智能合约的定义，接下来介绍区块链智能合约系统框架，如图 1-1 所示，区块链智能合约架构大致由如下四部分构成^{[10][11]}：

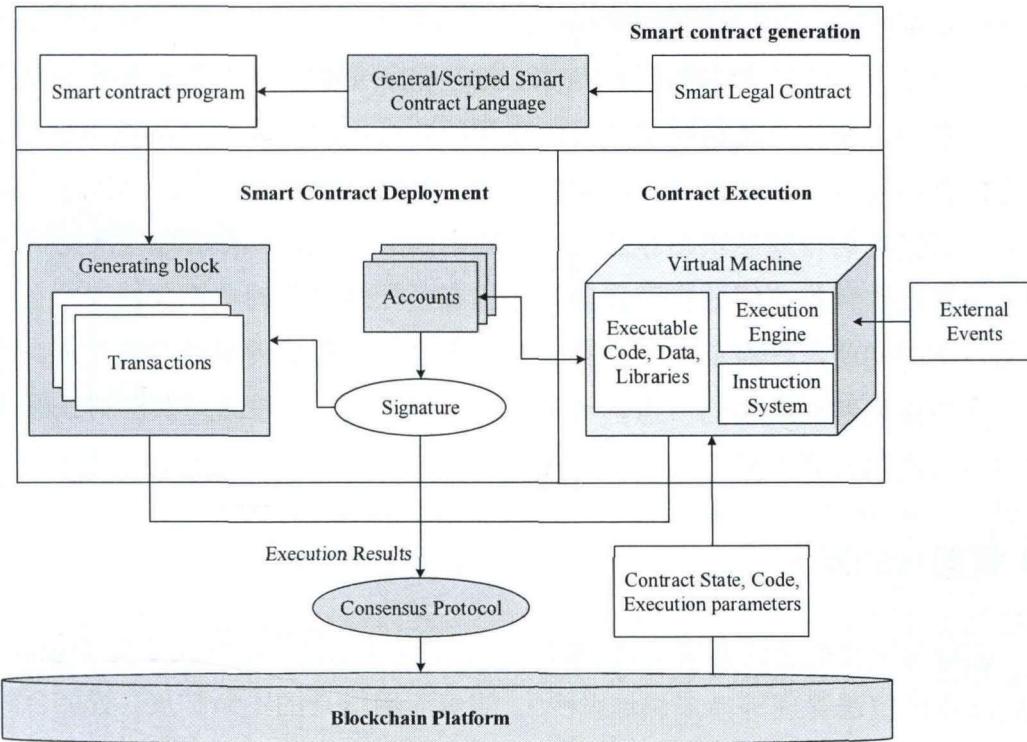


图 1-1 区块链智能合约系统框架

1) 智能合约生成模块：为智能合约开发提供代码开发环境，包括：提供支持交易合同生成智能合约的编程语言规范、开发和编译工具^[12]，帮助开发者根据智能法律转化智能合约程序，并按照智能合约执行模块中指令系统的语法规范将上述程序编译成为可执行平台代码。

2) 智能合约部署模块：提供前述可执行平台代码部署到区块链所需要的工具，包括：建立合约实例并通过缔约双方的协商为变量赋值，再将上述合约实例以及合约信息（如可执行代码、合同账户、当事人签名）封装到区块链交易中，最后通过共识机制将合约部署到区块链。

3) 智能合约执行模块：提供一种可信执行环境来运行合约代码^[13]，包括：接收外部发来的可信事件或内部交易，为可信智能合约执行环境下载相关区块链交易中的合约代码，触发执行机构和指令系统执行合约代码对事件或交易进行响应，并与区块链交互同步合约状态信息^[14]。

4) 区块链平台：为智能合约部署和运行提供可信数据存储环境并存证，在去中心化的区块链网络^[15]基础上，通过屏蔽区块链细节化的各种复杂机制（哈希、对等网络、挖矿等）直接向智能合约使用者提供平台接口，为智能合约平台代码、执行信息、运行状态等提供安全和一致性的共享交易数据库。

综上所述，区块链智能合约兼顾了普通合约与区块链的双重属性，能够在没有可信中介的情况下自动完成价值转移并允许各方就转移结果达成一致，其特征如下：

- 1) 强制约束：受限于区块链存储能力的限制，所存储合约程序的体量不宜过大，且区块链的不可篡改性决定了合约履行必须严格依据合约程序加以实施，履行过程被区块链存证并可作为诉讼证据；
- 2) 自动执行：一旦满足预定义条件，合约条款中的代码将自动执行，执行结束后将程序状态存储于区块链中，在后续条件满足时可恢复状态并继续执行，因此这种自动执行具有条件触发下的非连续性；
- 3) 公开透明：合约及其状态被存储于区块链中并在所有节点间共享，进而区块链网络中的节点都可浏览和执行该合约，通过共识机制来保证节点执行合约的正确性。

1.3 智能合约安全

正是由于智能合约具有上述良好特征，该技术已被广泛应用于电子投票、竞拍以及供应链管理等诸多领域^{[16]-[18]}。随着应用场景的不断扩展，智能合约所面临的安全问题也日益突出。尽管区块链为智能合约提供了近乎完美的去中心化环境，保证智能合约及相关数据都可以被加密并放到不可被更改的账本上^[19]。然而，这远远不能满足智能合约安全需求的全部，智能合约面临的安全问题大致包括以下三类^{[20]-[22]}：

- 1) 有效性：智能合约行为是否符合预期，合约状态是否满足规约条件，法律合同、源程序与转换后的目标代码是否一致；
- 2) 隐私性：智能合约中的敏感信息（包括私密数据、身份信息等）不被暴露给外界，或者不被非授权用户访问；
- 3) 可靠性：通常包括智能合约代码的有界性、可达性和状态无二义性，当这些性质不被满足时，通常会引发安全漏洞及安全风险，例如重入、整数溢出、缓冲区溢出、变量未初始化等。

目前，通过形式化方法验证智能合约以保证其有效性和可靠性的研究已取得较大进展，例如采用定理证明的方式验证智能合约的可靠性^{[23][24]}，通过

基于符号执行的方法验证智能合约程序是否满足特定属性^{[25][26]}，利用模型检查方式验证智能合约的有效性^{[27][28]}等。然而，与其它两方面相比，关于智能合约隐私性保护的研究还不充分，特别是在智能合约交易数据以及参与方身份认隐私保护方面尤为突出。

由于合约被部署在开放透明的区块链上，这将导致合约交易中敏感数据的隐私泄露^[29]。作为存放合约的载体，交易以“键-值”对的形式存储合约参数，包括交易的发送和接收方地址、交易金额和数据等信息。区块链的开放性使得链上信息可供用户随时查询，这也使得交易参数也可被用户公开获取，这对于交易数据的隐私安全性带来极大的威胁。例如在竞拍合约中，若标底价信息被直接以交易的形式存放在区块链上，则竞拍人在出价之前就可获取标底价，这不符合开标之前标底价保密的约定，进而将影响竞拍的公平性。

除了交易数据隐私之外，在合约订立和交互执行过程中用户身份隐私信息同样值得关注。例如在投票合约中^[30]，选民在进行投票之前需向投票发起机构提供身份信息以验证其是否具有合法投票权；在金融交易过程中，银行在为用户办理相关业务之前需验证其是否在失信人名单中，验证过程需要交易用户提供身份或账户信息。上述合约实例中，用户身份验证过程均需提供个人敏感信息并通过合约平台验证身份的有效性，而区块链的公开透明性将导致验证过程中用户身份信息泄露的风险。

因此，针对合约交易数据隐私以及合约订立和交互执行过程中身份信息泄露问题，开展面向区块链智能合约的隐私保护技术研究是十分必要和迫切的，这也是本文的研究重点。

1.4 国内外研究现状

密码技术作为保障网络空间安全的核心，能够为区块链智能合约安全提供最有力的保障^[31]。近年来，为了适应区块链分布式网络的数据安全需求，面向区块链智能合约交易隐私以及特定领域合约验证隐私保护的密码技术得到了长足的发展。接下来，本节将对面向区块链智能合约隐私保护的相关密码技术国内外研究现状展开讨论。

1.4.1 面向智能合约交易隐私的广播加密机制

由于智能合约被部署在公开透明的区块链上，使得存放在区块链上的合约代码以及交易数据也是公开可获取的，这将导致合约隐私泄露问题^[32]。大多数合约交易数据涉及到交易双方的隐私信息，例如在竞拍合约中^[33]，标底

价作为秘密信息在开标之前应该是对所有竞标者严格保密的，这些信息一旦泄露将会影响竞拍流程的公平性。

为避免合约交易隐私数据被非授权用户获得，智能合约交易隐私的研究主要关注于对交易数据的加密保护上。Bünz 等人^[34]提出了一种名叫 Zether 的完全去中心化、保密的支付机制，该机制可通过密码学证明技术在账户余额加密状态下实现对账户的存款、转账和取款等操作。此外，该机制还能抵抗重放攻击并与以太坊平台相兼容。Steffen 等人提出的 Zkay 协议^{[35][36]}支持非密码专业的开发人员通过调用公钥基础设施（Public Key Infrastructure, PKI）体制下加密模式（例如 RSA）对交易数据进行加密来保护交易数据隐私。随后，Steffen 等人^[37]改进了上述协议，通过零知识证明和加法同态加密相结合实现合约对外部数据的安全操作。

为了实现智能合约交易数据隐私并在指定群体之间安全共享，上述密码技术还远远不够，这是因为 PKI 体制下的加密模式只支持“一对一”的加密，实现交易隐私数据在多个设备或个体之间安全共享需对数据反复加密并逐节点传输，这种方式是极其低效的。广播加密（Broadcast Encryption, BE）^{[38][39]}作为一种“一对多”的群组加密机制可实现一个发送者和多个接收者之间通过开放的网络安全地通信，其广播结构与区块链广播形式网络相吻合，因此广播加密是智能合约中实现敏感交易信息安全高效共享的首选加密技术。

Jin 等人^[40]探索了广播加密与智能合约相结合并实现了个人病例隐私数据在指定接收者集合内安全共享。该方案利用不同的密钥对不同敏感度医疗数据分别加密，其中用于加密低灵敏度数据的密钥可以从用于加密高敏感度数据的密钥中派生出来，这样就实现了对医疗数据的访问控制。虽然该方案具有一定访问控制能力，但其授权方式单一。最近，Deng 等人^[41]依据可撤销广播加密方案^[42]构造了一种基于区块链智能合约的物联网安全订阅推送服务模型并设计了一种新的广播加密方案，他们的方案支持对非法用户的撤销，被撤销用户将无法从密文中恢复出推送信息。此外，该方案与智能合约相结合管理物联网设备访问权限以保护推送消息的机密性。

上述面向智能合约交易隐私的广播加密机制分别对应于选择加密模式（Selective Encryption Mode, SEM）和排它加密模式（Exclusive Encryption Mode, EEM）。在选择加密模式中，广播发送方在选定的用户集 S 下加密隐私数据，并且只有在集合 S 中的用户才能从密文中恢复数据；在排它加密模式下，方案在排除用户集 R 下的加密私有数据，并且所有不在排除用户集中接收者均可以从密文中恢复数据。在某些特定领域中，智能合约对隐私的需求是多样的，例如在拍卖合约中，标底价仅在开标之后对参与竞拍者公开，同

时还要求只有不在黑名单中的竞标者才能了解拍卖物品详细信息。这就要求广播加密需要同时支持上述两种加密模式以满足交易数据的不同隐私需求。

现有广播加密方案多在单模式下对方案的功能和安全性进行研究，例如 Delerablée^[43]以及 Sakai 和 Furukawa^[44]分别独立提出了身份基广播加密方案（Identity-based Broadcast Encryption, IBBE），以广播接收者的身份（例如姓名或 IP）作为公钥用于消息的加密，仅支持选择加密模式。Boneh 等人^[45]利用多线性映射构造出一种低开销的 IBBE 方案，在该方案中密文、私钥和公钥尺寸均与用户总数成对数相关。为了保护广播接收者的身份隐私，Barth 等人^[46]首次将匿名性引入到广播加密中并构造了第一个匿名广播加密方案，该方案可以在不泄露接收者身份的前提下向多个接收者加密消息。

Susilo 等人^[42]首次提出了接收者可撤销的身份基广播加密概念以支持排它广播加密模式，该方案将密文发送给第三方，由第三方从密文中撤销部分接收者身份。此后，Lai^[47]构造了一种完全隐私保护的可撤销身份基广播加密方案，能够同时保护广播接收者以及被撤销用户的身份隐私，在随机预言机模型下，该方案被证明具有语义安全性。Jia 等人^[48]通过将 RSA 密码累加器与基于身份的加密系统相结合，提出了一种具有固定大小密文和私钥的可撤销广播加密方案，该方案利用双系统加密技术在标准假设下被证明是安全的。

目前面向智能合约交易隐私的广播加密研究多集中在单一的选择或排它模式。在同一个广播加密系统下，几乎没有方案可以同时满足双模式广播加密以满足交易数据的不同隐私要求。虽然部分广播加密方案能够实现指定和撤销机制^[49]，但未考虑广播接收者的身份隐私问题。通过对现有广播加密方案的分析和总结发现，目前还没有能够支持选择和排它的双模式广播加密方案来满足智能合约交易数据的多样化隐私需求。

1.4.2 面向交易数据安全共享的属性基加密机制

尽管广播加密机制能够实现智能合约交易数据的安全共享，但是与属性基加密（Attribute-based Encryption, ABE）相比授权方式单一，无法实现细粒度的访问授权。作为身份基加密（Identity-based Encryption, IBE）的推广，ABE 方案^{[50][51]}中属性是用户公钥，密文以及用户私钥与属性关联从而实现在用户群组之间进行安全数据共享，属性基加密机制不仅能够实现隐私数据在群组间共享，而且能够实现对加密数据的细粒度访问控制。

针对智能合约交易隐私数据安全共享问题，Xu 等人^[52]设计了一种新的分层属性基加密方案（Hierarchical Attribute-based Encryption, HABE），并设

计了基于区块链的细粒度访问控制数据安全共享平台，该平台通过将不同的用户属性分配给不同的授权中心，实现了灵活且细粒度的访问控制。此外，通过将部分高复杂度的解密计算转移至区块链智能合约执行，降低了用户端的解密开销。Alniamy 等人^[53]将超级账本技术和基于属性的加密方案结合来实现在去中心化环境下隐私数据共享的细粒度访问控制。在该方案中，数据所有者在由属性组成的访问控制策略下对数据进行加密以实现对数据的访问控制。此外，Wang 等人^[54]利用属性基密码系统与区块链技术实现了电子医疗数据的安全共享。

在密钥管理方面，Wu 等人^[55]基于区块链技术提出了一种高效的可追踪属性加密方案，该方案通过使用属性布隆过滤器对访问策略中的属性进行了隐藏，当密钥被滥用时，可以对滥用密钥的来源进行追踪。针对 ABE 方案中的用户撤销问题，Bramm 等人^[56]在区块链分布式属性基加密方案的基础上，提出了用户属性协同管理协议，并构造了密文策略属性基加密方案（Ciphertext-policy ABE, CP-ABE），它是通过添加一个共识驱动的基础设施实现了对属性密钥的分布式发布、存储和撤销。He 等人^[57]提出了一种基于区块链的物联网设备管理方案，该方案提供了高效的基于属性的访问控制，并支持密钥自动撤销。此外，Guo 等人^[58]还研究了面向交易数据安全共享的属性基加密机制的密钥更新问题。

然而，上述几乎所有方案都建立在单一中心之上，即用户私钥是由唯一可信中心颁发。为了适应多中心化环境，2007 年 Chase^[59]提出第一种多权威中心属性基加密方案（Multi-Authority ABE, MA-ABE），该方案允许任意多个独立权威中心管理属性和分发密钥。然而，为了抵抗用户属性共谋，该方案需要一个完全可信的中心参与用户私钥生成。为了去除可信中心，Lewko 和 Waters^[60]提出一种去中心化属性基加密（Decentralizing ABE, DABE）。在他们的方案中，可信中心节点被移除，每个权威中心管理一部分属性并独立地向用户颁发属性密钥，但系统初始化阶段需要所有属性权威协作完成。近年来，一些多中心 ABE 方案还研究了区块链隐私数据安全共享问题^{[61]-[63]}。

这些方案虽然可以实现用户私钥的去中心化生成，但容错性低，这意味着如果一个属性中心被破坏，整个系统将无法运行。此外，现有属性基加密方案中密文大多以数据形式存储，在密文解密过程中，用户首先检索属性子密文对应的密钥，然后再在属性之间进行逻辑匹配直到解密出消息，这种密文存储方式给解密操作带来了额外开销，不利于在计算受限的区块链节点上进行加解密运算。

1.4.3 面向特定场景下合约成员关系证明机制

一些特定场景下的智能合约需要验证参与方的身份，即判定交易参与方是否在某个指定范围之内。例如，在投票合约中，选民在投票之前需要在用户数据库中注册，投票发起机构在选民投票之前需要对用户身份的合法性进行检查，判断其是否在注册用户列表中。在 Baza 等人^[64]提出的区块链拼车合约中，乘客和司机需要互相验证对方的行程数据，以确认是否可以共享车辆。

解决上述问题的有效方法就是集合成员关系证明机制（Set Membership Proof, SMP），该机制本质是验证一个元素是否是集合成员，即 $e \in S$ 或 $e \notin S$ 。在上述投票合约中，利用 SMP 不仅可以避免非法用户对投票过程的干扰而且可以加速判定过程。因此，该机制是安全判断待测元素与集合所属关系的有效方式，也是智能合约动态群组成员关系证明的可靠途径。

现有密码学集合成员关系判定的方法包括布隆过滤器（Bloom Filter）^[65]、密码学累加器（Cryptographic Accumulator）^[66]等技术。布隆过滤器可以看作是一种有效地检查集合成员资格的数据结构^[67]。在基于布隆过滤器的集合关系证明中，首先将集合中的所有元素进行置乱^[68]，也即是通过一系列哈希函数将集合中所有元素的哈希值作为阵列的地址，然后，再将元素映射到长度为阵列上的一个点上。当该点对应的哈希值为 1 时，表示这个元素在集合中；否则，元素不属于这个集合。布隆过滤器将集合中的元素映射成一个二进制向量，这大大提高了集合的存储空间。此外，在元素的查询过程中，仅需要判断对应阵列位置上的数值是否为 1。综上，集合的哈希映射和元素的检查时间都是常数级别的，这大大提高了集合成员判定的效率。2009 年 Nojima 和 Kadobayashi^[69]采用盲签名（Blind Signature）和遗忘伪随机函数（Oblivious Pseudorandom Function, OPF）来增强阵列中比特信息检测的隐私性。此后，Ramezanian^[70]对上述工作进行了增强，采用 Goldwasser-Micali 同态加密^[71]和 RSA 盲签名^[72]给出了几种隐私保护的成员关系测试方案。

基于密码学累加器的集合关系证明也可用于判定一个候选元素是否为一个集合的成员，且不会在过程中暴露集合中的成员。Benaloh 等人^[73]构造了一个 RSA 累加器，利用 RSA 密码中加密指数交换性，给出了一种可用在属于关系判定的简单方案。在此基础之上，Papamanthou 等人^[74]采用认证哈希列表（Authenticated Hash Table, AHT）对集合的表示进行了优化。在此之后，该文作者对上述方案进行了修改^[75]，给出了基于椭圆曲线的双线性映射下的累加器构造以及更加严格的定义和证明。此外，一些学者也研究了基于累加器的不属于关系判定问题。例如 Camenisch 和 Lysyanskaya^{[76][77]}提出了具有

撤销机制的动态累加器概念，他们的方案能证明元素不属于指定的集合并能从累加器中以较低的开销删除该元素。

然而，现有集合成员关系证明机制仅考虑“正”(\in)或“负”(\notin)一种集合关系的证明，无法同时实现“正负”这种对偶关系的严格关系证明。其次，在集合关系证明过程中，虽然现有的关系证明机制可实现对集合一定程度的压缩表示，但更加彻底和安全的集合压缩还需要进一步研究，以满足智能合约大规模网络应用的需要。最后，为实现合约交易验证过程中的隐私保护，在构造支持对偶集合关系证明协议中还需考虑待测元素的隐私。

1.4.4 目前研究存在的问题

通过对面向区块链智能合约隐私保护相关密码技术的分析，结合智能合约隐私需求，总结了现有研究存在的三个主要问题如下：

问题 1：在面向智能合约交易隐私的广播机制中，构造同时支持选择和排它加密模式的广播加密方案以满足交易数据不同的隐私需求是一个挑战。

目前面向智能合约交易数据隐私保护的广播加密多集中在对选择接收者或排除接收者单独模式的研究。针对智能合约中敏感数据访问限定到指定人群和非冲突人群的两种隐私需求，如何在一个加密系统下设计同时满足选择加密和排它加密模式的双模式广播加密方案来分别满足上述隐私需求是需要考虑的问题。

问题 2：在面向交易数据安全共享的属性基加密机制中，构造密钥去中心化管理以及可编程的密文的属性基加密方案是当前面临的挑战。

当前面向交易数据安全共享的属性基加密大多部署在集中式环境中，针对区块链去中心化特征与属性基加密中心化密钥管理相冲突的问题，设计去中心化密钥管理的属性基加密方案是一项挑战；此外，现有属性基加密方案中密文大多以数据形式存储，如何将访问策略中复杂逻辑融入到密文中实现可编程密文表示形式也是需要关注的问题。

问题 3：在保证合约参与方身份隐私的前提下，构造零知识对偶集合成员关系证明协议，实现对正负集合成员关系的同时证明是需要突破的挑战。

现有面向合约成员身份认证的成员关系证明机制仅考虑正或负一种成员关系的证明，无法实现正负成员关系的同时验证；其次，区块链的公开透明性会导致验证过程中合约参与方的身份信息泄露。针对智能合约中动态群组成员的身份认证及判定过程中的用户隐私泄露问题，如何在满足正负集合成员关系同时判定的前提下实现对待测元素的隐私保护是当前面临的挑战。

1.5 研究目标及研究内容

智能合约交易隐私不仅涉及交易数据的机密性还包括合约订立和交互执行过程中用户身份隐私，密码技术作为保护数据隐私最有效的手段可为合约隐私保护提供强有力的理论和技术支撑。因此，针对区块链智能合约的结构特点，开展面向区块链智能合约的密码学隐私保护技术研究是十分必要的。

1.5.1 研究目标

本文将从智能合约安全角度，以设计安全且高效的密码协议为目标来确保对合约交易数据的隐私保护、安全共享以及智能合约成员关系安全验证，构造符合区块链智能合约结构特点的群组加密和集合成员关系证明方案。研究目标概括如下：

- 1) 探索面向智能合约交易隐私的双模式身份基广播加密。针对广播加密集中在“选择”或“排它”单一模式的研究不能满足合约数据多样化隐私需求的问题，探索在一个加密系统下同时支持上述两种不同加密模式的广播加密方案。此外，在智能法律合约语言下探究具有隐私保护功能的智能合约架构，用于保证敏感数据以合约条款形式声明隐私并由编译器将预定义的密码算法链接到智能合约程序中以实施对交易敏感数据的保护。
- 2) 探索面向属性基加密的去中心化密钥管理与脚本化密文机制。针对区块链去中心化特征与属性基加密中心化密钥管理相冲突的问题，设计符合区块链分布式网络特点并能够实现系统密钥去中心化管理和用户私钥的协同生成的属性基加密方案。其次，针对现有加密方案中数据形式密文对解密效率带来的影响，探索可编程密文表示形式将访问策略中的复杂逻辑嵌入到密文中，进而实现由区块链节点以程序代码形式进行加解密操作。
- 3) 探索面向合约成员身份认证的零知识对偶集合成员关系证明协议。针对智能合约中动态群组成员的身份认证及判定过程中的用户隐私泄露问题，探究聚合函数的构造实现将子集压缩到密码空间元素的表示方法。此外，提出安全聚合函数的概念并在子集压缩表示过程中将元素与集合间属于和不属于的判定问题转变为元素删除和插入下的聚合函数求解问题，在此基础上探究基于正负成员关系判定架构下的零知识对偶集合成员关系证明协议的构造。

1.5.2 研究内容及组织结构

本文主要针对面向区块链智能合约的密码隐私保护技术展开研究，对于合约数据隐私以及合约成员关系认证过程中的用户身份隐私，分别设计了符合区块链智能合约结构和需求的密码方案与协议。本文的组织结构如图 1-2 所示，以第二章描述的数学及密码学基础知识和区块链智能合约语言为基础展开研究，主要内容包括：首先，针对具体业务中交易数据隐私保护的需求，研究基于合约层的隐私保护机制（Contract-layer Privacy-protection），探索支持交易数据多样化隐私需求的广播加密设计；其次，考虑到更加复杂业务和多合约构建的综合业务系统的隐私需求，研究基于区块链层的隐私保护机制（Blockchain-based Privacy-protection），探索支持可编程密文和去中心化密钥管理的属性基加密设计；此外，针对智能合约中特有的动态当事人群组中的身份认证需求，研究合约成员关系认证过程中身份信息隐私保护机制，探索支持零知识对偶集合成员关系证明的协议设计等三个方面，它们分别对应于本文第三、第四和第五章，具体研究内容之间关系如图 1-2 所示。

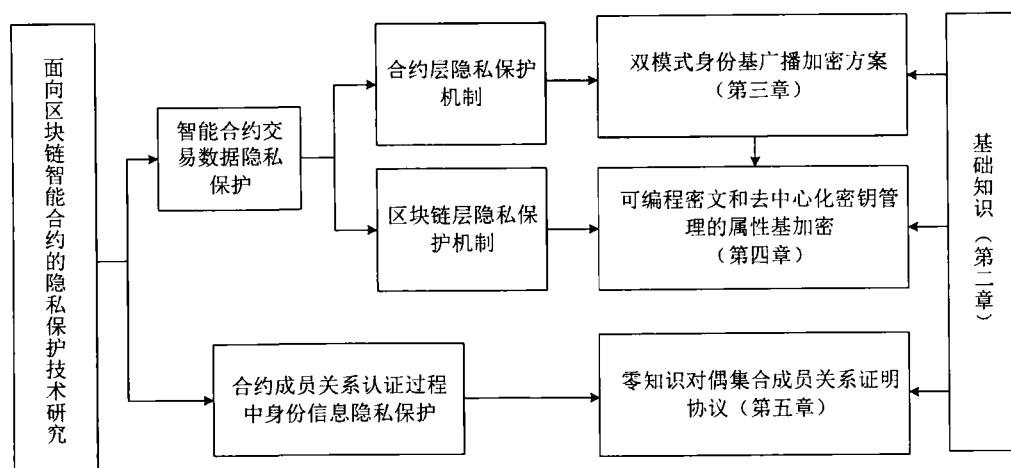


图 1-2 论文组织结构

1) 针对问题 1，第三章对交易数据隐私保护展开研究，主要包括：双模式身份基广播加密方案（Dual-mode Identity-based Broadcast Encryption, DM-IBBE）以及 DM-IBBE 用于交易数据隐私保护实例研究，并对该方案进行安全性证明和性能分析。

- ① 研究一种支持双模式身份基广播加密方案的构造方法，在同一个广播加密系统下同时实现“选择”和“排它”两种加密模式以满足交易数据不同的隐私需求；
- ② 以竞拍合约为例，研究一种具有隐私保护功能的智能合约架构，保证敏感数据以合约条款形式声明隐私并由编译器将预定义的 DM-IBBE

- 算法链接到智能合约程序中对数据隐私实施保护；
- ③ 研究双模式身份基广播加密方案的语义安全性以及在选择加密模式下的用户匿名性，并对上述方案的计算和存储开销进行分析。
- 2) 针对问题 2，第四章对交易数据安全共享过程中密钥管理以及密文存储形式展开研究，主要包括：去中心化密钥生成的密文策略属性基加密（Ciphertext-policy Decentralized-key Attribute-based Encryption, CP-DK-ABE）以及脚本化密文机制，并对上述方案进行安全性证明和性能分析。
- ① 研究一种去中心化密钥生成的密文策略属性基加密方案的构造方法，实现各方对主密钥的共同管理和用户私钥的协同生成以适应区块链去中心化的特征；
- ② 研究基于区块链脚本的可编程密文机制，通过密钥查询、密文逻辑等操作来扩展区块链脚本指令系统，实现对访问策略中复杂逻辑的密文脚本化，进而实现脚本解释器对脚本化密文的自动化解密；
- ③ 研究 CP-DK-ABE 方案中去中心化密钥生成算法的隐私性以及方案的语义安全性，并对上述方案的计算和存储开销进行分析。
- 3) 针对问题 3，第五章对合约成员身份认证过程中用户身份隐私保护展开研究，主要包括：安全聚合函数的构造以及子集安全表示方法，对偶集合成员关系零知识证明协议（Zero-knowledge Dual-membership Proof, ZKDMP），并对该协议进行安全性证明和性能分析。
- ① 研究安全聚合函数的构造实现子集压缩到密码空间元素的表示方法，并在子集压缩过程中将元素与集合间属于和不属于的判定问题转变为元素删除和插入下的聚合函数求解问题；
- ② 在子集安全表示的基础上研究对偶集合成员关系安全证明协议的构造方法，在满足正负集合成员关系同时判定的前提下实现对待测元素的隐私保护；
- ③ 研究对偶集合成员关系零知识证明协议的正/负完整性、完备性以及零知识性，并对该协议的计算和存储开销进行了分析。

1.6 研究意义

本文针对区块链智能合约中存在的两类隐私问题，即智能合约数据隐私保护以及合约成员关系认证过程中的用户身份隐私保护，构造了双模式广播加密方案、去中心化密钥管理的属性基加密方案以及零知识性对偶集合成员关系证明协议，研究结果对智能合约隐私保护以及相关密码方案构造具有一

定的理论与实践意义：

1) 在智能合约隐私保护研究方面,本研究提出的双模式广播加密方案能够有效的保护智能合约交易隐私数据的安全并满足合约交易中的不同隐私需求,通过将密码算法与合约条款相结合,在智能法律合约语言 SPESC 下给出一种具有隐私保护功能的智能合约架构,保证敏感数据以合约条款的形式声明隐私,并通过编译器将预定义的加密算法链接到智能合约程序中对敏感数据实施保护;其次,通过本文提出的去中心化密钥管理的属性基加密方案,实现区块链节点对主密钥的共同管理以及用户私钥的分布式密钥生成,为区块链智能合约中密钥管理提供技术参考;此外,提出的集合关系证明协议不仅能够满足合约成员身份认的严格验证,而且能够在验证过程中保护合约参与方的隐私。总体来说,本文提出的密码技术在区块链智能合约隐私保护方面就有积极的理论和实践意义。

2) 在密码理论研究方面,首先本文通过在拉格朗日插值曲线下对不同插值点与重构曲线上点的选取,设计了支持“选择”和“排它”两种加密模式的身份基广播加密方案,两种模式下生成的密文结构相同,这对具有丰富加密模式的群组密码系统的构造具有一定的借鉴作用;其次,通过融合两类加法同态的安全多方计算技术,构造了一种基于密文策略和去中心化密钥的属性基加密方案,实现各方对主密钥的共同管理和用户私钥的协同生成,解决了区块链去中心化特征与属性基加密中心化密钥管理相冲突的问题;此外,通过构造聚合函数实现子集压缩到密码空间元素的表示方法,将子集表示的规模降低到理论下限,并引入安全聚合函数(SAF)的概念,在子集压缩过程中将元素与集合间属于和不属于的判定问题转变为元素删除和插入下的聚合函数求解问题,这种安全的压缩映射可为密码学集合成员关系判定机制研究提供新的思路。

1.7 本章小结

本章首先对面向区块链智能合约的密码技术研究背景进行了介绍,并阐述了智能合约及其安全问题,在此基础上对实现智能合约交易隐私的相关密码机制进行了梳理和分析,并依据研究现状总结给出目前研究存在的问题,进而针对这些问题提出本文的研究目标并阐述了本文主要研究内容。最后,对本文的组织结构和具体研究进行了介绍,并对本文的研究意义进行总结。

2 预备知识

本章将列出一些数学基础知识和密码学以及区块链智能合约的相关概念，它们将被用于后续章节中密码算法的设计与分析中。

2.1 数学基础

定义 2-1 (双线性映射^{[81][82]}) 给定素数阶群 \mathbb{G} 中任意元素 g 和 h ，双线性映射将它们映射到 \mathbb{G}_T 中的元素，即 $e: \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ ，该映射满足以下条件：

- 1) 可计算性：对于 \mathbb{G} 中任意元素 g 和 h ，存在有效的算法计算 $e(g, h)$ ；
- 2) 双线性性：对于 $\forall a, b \in \mathbb{Z}_p^*$ 和 $\forall g, h \in \mathbb{G}$ ，则有 $e(g^a, h^b) = e(g, h)^{ab}$ 成立；
- 3) 非退化性：对于 $\forall g, h \in \mathbb{G}$ 有 $e(g, h) \neq 1$ ，其中 1 是群 \mathbb{G}_T 中的单位元。

在本文后续密码算法构造中，令 $\mathbb{S} = \{p, \mathbb{G}, \mathbb{G}_T, e(\cdot, \cdot), g, h\}$ 为一个双线性群实例，其中 \mathbb{G} 是素数 p 阶的加法循环群， g 和 h 是该群中的两个随机生成元， \mathbb{G}_T 是阶数为 p 的乘法循环群， $e(\cdot, \cdot)$ 为满足上述条件的双线性映射。

定义 2-2 (多项式零点与极点) 令 $H(x) = P(x)/Q(x)$ 为实多项式，若对于任意整数 z 使得 $P(z) = 0$ ，则称 z 为多项式 $H(x)$ 的零点；若对于任意整数 z 使得 $Q(z) = 0$ ，则称 z 为多项式 $H(x)$ 的极点。

定理 2-1 (全概率公式) 若事件 B_1, B_2, \dots, B_n 两两不相容，即对于任意的 $i \neq j, j \in [1, n]$ 总是有 $B_i B_j = \emptyset$ ，则称 $B_1 \cup B_2 \cup \dots \cup B_n = S$ 构成完备事件组。对于任意事件 A 且 $\Pr[B_i] > 0, i \in [1, n]$ ，则有

$$\begin{aligned} \Pr[A] &= \Pr[B_1] \Pr[A | B_1] + \Pr[B_2] \Pr[A | B_2] + \dots + \Pr[B_n] \Pr[A | B_n] \\ &= \sum_{i=1}^n \Pr[B_i] \Pr[A | B_i] \end{aligned} \tag{2-1}$$

证明：由于 $A = AS = A \left(\bigcup_{i \in [1, n]} B_i \right) = \bigcup_{i \in [1, n]} AB_i$ ，且对于 $i \neq j, j \in [1, n]$ 有 $AB_i \cap AB_j = \emptyset$ ，因此由可加性得

$$\Pr[A] = \Pr \left[\bigcup_{i \in [1, n]} AB_i \right] = \sum_{i=1}^n \Pr[AB_i] \tag{2-2}$$

又由于 $\Pr[AB_i] = \Pr[B_i] \Pr[A | B_i]$ ，则等式(2-2)可表示如下：

$$\Pr[A] = \sum_{i=1}^n \Pr[B_i] \Pr[A | B_i] \tag{2-3}$$

2.2 困难性假设

在介绍相关困难性假设之前，首先给出可忽略函数(Negligible Function)的定义，可忽略函数在现代密码学特别是在可证明安全领域具有重要作用，其具体定义如下：

定义 2-3 (可忽略函数^{[83][84]}) 设函数 $\epsilon(x): \mathbb{N} \rightarrow \mathbb{R}$ ，若对于任意多项式 $poly(x)$ ，存在自然数 N ，当 $x \geq N$ 时，总有 $\epsilon(x) \leq 1/poly(x)$ 成立，则称 $\epsilon(x)$ 为可忽略函数，为了方便起见，下文简写成 ϵ 。

定义 2-4 (强 Diffie-Hellman 假设^[85]) 从给定的群 \mathbb{G} 中的 $t+1$ 个元素 $(g, g^{\alpha}, \dots, g^{\alpha^t}) \in \mathbb{G}^{t+1}$ 中输出 $\langle c, g^{1/(\alpha+c)} \rangle$ ，其中 α 和 c 是 \mathbb{Z}_p^* 中的元素，则该问题称为 t -Strong Diffie-Hellman (t -SDH) 问题。定义一个概率多项式时间 (Probabilistic Polynomial Time, PPT) 敌手 \mathcal{A} 在解决上述问题的优势为

$$Adv_{SDH}(\mathcal{A}) = \Pr[\mathcal{A}(g, g^{\alpha}, \dots, g^{\alpha^t}) = \langle c, g^{1/(\alpha+c)} \rangle] \quad (2-4)$$

若对于任意的 PPT 敌手，它在解决 t -SDH 问题中的优势均小于可忽略的值 ϵ ，即 $Adv_{SDH}(\mathcal{A}) < \epsilon$ ，则称 t -SDH 假设成立。

定义 2-5 (判定性 Diffie-Hellman 假设^[86]) 令 \mathbb{G} 为素数 q 阶的循环群， g 为群 \mathbb{G} 中的一个随机生成元，对于任意的整数 $a, b \in \mathbb{Z}_p$ ， $Z \in \mathbb{G}$ 以及任意的 PPT 敌手 \mathcal{A} ，定义其正确区分出两个 4 元组 (g, g^a, g^b, g^{ab}) 和 (g, g^a, g^b, Z) 的优势如下：

$$Adv_{DDH}(\mathcal{A}) = \left| \Pr[\mathcal{A}(g, g^a, g^b, g^{ab}) = 1] - \Pr[\mathcal{A}(g, g^a, g^b, Z) = 1] \right| \quad (2-5)$$

若对于任意 PPT 敌手，它在正确区分上述两个 4 元组的优势都是小于可忽略的值 ϵ ，即 $Adv_{DDH}(\mathcal{A}) < \epsilon$ ，则称 Decision Diffie-Hellman (DDH) 假设成立。

定义 2-6 (判定性线性 Diffie-Hellman 假设^[87]) 令 $\{a_i, c_i\}_{i=1}^t$ ， b 以及 Z 为群 \mathbb{Z}_p^* 中的随机元素， g 为群 \mathbb{G} 的生成元。对于任意的 PPT 敌手 \mathcal{A} ，定义其正确区分多元组 $(g, g^b, \{g^{b(a_i+c_i)}, g^{-c_i}\}_{i=1}^t, g^{ab})$ 和 $(g, g^b, \{g^{b(a_i+c_i)}, g^{-c_i}\}_{i=1}^t, Z)$ 的优势如下：

$$Adv_{DLDH}(\mathcal{A}) = \left| \Pr[\mathcal{A}(g, g^b, \{g^{b(a_i+c_i)}, g^{-c_i}\}_{i=1}^t, g^{ab}) = 1] - \Pr[\mathcal{A}(g, g^b, \{g^{b(a_i+c_i)}, g^{-c_i}\}_{i=1}^t, Z) = 1] \right| \quad (2-6)$$

若对于任意 PPT 敌手 \mathcal{A} ，正确区分上述两个多元组的优势均小于可忽略的 ϵ ，即 $Adv_{DLDH}(\mathcal{A}) < \epsilon$ ，则称 Decision Linear Diffie-Hellman (DLDH) 假设成立。

定义 2-7 (判定性双线性 Diffie-Hellman 假设^[88]) 令 a, b, c, z 均为群 \mathbb{Z}_p^* 中的随机元素， g 为群 \mathbb{G} 的生成元。对于任意的 PPT 敌手 \mathcal{A} ，定义其正确区分 5 元组 $(g, g^a, g^b, g^c, e(g, g)^{abc})$ 和 $(g, g^a, g^b, g^c, e(g, g)^z)$ 的优势如下：

$$Adv_{DBDH}(\mathcal{A}) = \left| \frac{\Pr[\mathcal{A}(g, g^a, g^b, g^c, e(g, g)^{abc}) = 1]}{\Pr[\mathcal{A}(g, g^a, g^b, g^c, e(g, g)^z) = 1]} \right| \quad (2-7)$$

若对于任意 PPT 敌手 \mathcal{A} 正确区分上述两个 5 元组的优势都小于可忽略的 ϵ ，即 $Adv_{DBDH}(\mathcal{A}) < \epsilon$ ，则称 Decision Bilinear Diffie-Hellman (DBDH) 假设成立。

2.3 交互证明系统

交互证明系统 (Interactive Proof System, IPS) 是一个两方交互式游戏，一方 (证明者, Prover) 向另一方 (验证者, Verifier) 证明一个命题成立。其本质是证明者向验证者提供足够的信息，验证者可以以较小的错误概率验证证明者的断言，如图 2-1 所示。

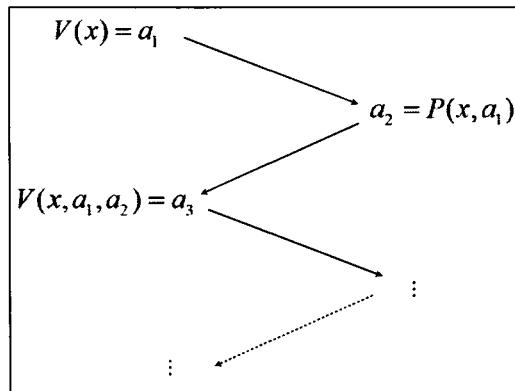


图 2-1 交互式证明系统一般表示

在游戏过程中，假设证明者计算能力无穷大，验证者具有概率多项式时间的计算能力，交互式证明系统形式化定义如下：

定义 2-8 (交互式证明系统^{[89][90][91]}) 对于 $\{0,1\}^*$ 上的语言 L ，一对交互式图灵机 $\langle P, V \rangle$ 称为 L 的一个交互证明系统，使得下面两个条件成立：

1) 完整性 (Completeness): 对于所有的 $x \in L$ ，

$$\Pr[(P, V)(x) = 1 | x \in L] \geq \epsilon \quad (2-8)$$

2) 完备性 (Soundness): 对于所有的 $x \notin L$ ，

$$\Pr[(P, V)(x) = 1 | x \notin L] \leq \delta \quad (2-9)$$

其中 ϵ 和 δ 是两个常数，满足 $\epsilon \in (1/2, 1]$ 以及 $\delta \in [0, 1/2]$ 。

2.4 秘密共享方案

本节将介绍 Shamir 在上世纪七十年代提出的 (t, n) 门限秘密共享方案^[92]，在介绍该方案之前，首先给出拉格朗日插值公式的相关概念。

定理 2-2 设 a_1, \dots, a_{n+1} 是 \mathbb{Z}_p 中的 $n+1$ 个不同元素， b_1, \dots, b_{n+1} 为 \mathbb{Z}_p 中 $n+1$ 个任意元素，那么多项式环 $\mathbb{Z}_p[x]$ 中存在唯一的一个次数不超过 n 的多项式 $f(x)$ ，使得对于任意的 $i \in [1, n+1]$ 有 $f(a_i) = b_i$ ，该多项式可表示如下：

$$\begin{aligned} f(x) &= \frac{(x-a_2)(x-a_3)\cdots(x-a_{n+1})}{(a_1-a_2)(a_1-a_3)\cdots(a_1-a_{n+1})} b_1 \\ &\quad + \frac{(x-a_1)(x-a_3)\cdots(x-a_{n+1})}{(a_2-a_1)(a_2-a_3)\cdots(a_2-a_{n+1})} b_2 + \dots \\ &\quad + \frac{(x-a_1)(x-a_2)\cdots(x-a_n)}{(a_{n+1}-a_1)(a_{n+1}-a_2)\cdots(a_{n+1}-a_n)} b_{n+1} \mod p \end{aligned} \quad (2-10)$$

下面介绍门限为 t ($t \leq n$) 的 Shamir 秘密共享方案，设一个秘密值 $s \in \mathbb{Z}_p$ 由 n 个用户 A_1, A_2, \dots, A_n 共享。 $\mathbb{Z}_p[x]$ 中一个 $t-1$ 阶多项式 $h(x)$ 定义如下：

$$h(x) = d_{t-1}x^{t-1} + d_{t-2}x^{t-2} + \dots + d_1x + d_0 \quad (2-11)$$

上式中，常数项 d_0 为秘密值 s ，即 $h(0) = s$ 。在 \mathbb{Z}_p^* 中随机选取元素 a_1, a_2, \dots, a_n 且 $p > n$ ，随后，对于 $i \in [1, n]$ ，计算 $b_i = h(a_i) \in \mathbb{Z}_p$ 作为秘密 s 的片段共享给用户 A_i 。则上述对秘密 s 的共享过程即为门限为 t 的秘密共享方案，下面对该方案的安全性进行分析。

1) 任意 t 个用户联合即可从秘密片段中恢复出原始秘密值 s

设 A_1, A_2, \dots, A_t 是要恢复秘密值的 t 个用户，这些用户联合一起便可得到 b_1, \dots, b_t 以及公开值 a_1, \dots, a_t 和关系 $h(a_i) = b_i$ ($1 \leq i \leq t$)。又因为 $h(x)$ 是多项式环 $\mathbb{Z}_p[x]$ 中的 $t-1$ 次多项式，由定理 2-2 可知， $h(x)$ 由 t 个不同的数值对 (a_i, b_i) 唯一确定，由拉格朗日插值公式可得，多项式 $h(x)$ 为：

$$\begin{aligned} h(x) &= b_1 \frac{(x-a_2)(x-a_3)\cdots(x-a_t)}{(a_1-a_2)(a_1-a_3)\cdots(a_1-a_t)} \\ &\quad + b_2 \frac{(x-a_1)(x-a_3)\cdots(x-a_t)}{(a_2-a_1)(a_2-a_3)\cdots(a_2-a_t)} + \dots \\ &\quad + b_{n+1} \frac{(x-a_1)(x-a_2)\cdots(x-a_n)}{(a_t-a_1)(a_t-a_2)\cdots(a_t-a_n)} \mod p \end{aligned} \quad (2-12)$$

进而计算出秘密值 s 如下：

$$s = h(0) = (-1)^{t-1} \left[\frac{b_1 a_2 a_3 \cdots a_t}{(a_1 - a_2) \cdots (a_1 - a_t)} + \cdots + \frac{b_t a_1 a_2 \cdots a_{t-1}}{(a_t - a_2) \cdots (a_t - a_{t-1})} \right] \quad (2-13)$$

2) 对于任意 $t-1$ 个用户合谋也不能重构出秘密值 s 的任何信息

假设 A_1, A_2, \dots, A_{t-1} 为 $t-1$ 个合谋试图重构秘密值 s 的用户，他们已知的信息有 $h(a_i) = b_i$ 以及公开值 $a_i \in \mathbb{Z}_p^*$ ($1 \leq i \leq t-1$)，因此对于任意的 $b \in \mathbb{Z}_p$ ，使得 $f(a_i) = b_i$ ($1 \leq i \leq t-1$) 和 $f(0) = b \in \mathbb{Z}_p$ 的次数不超过 $t-1$ 的多项式有且仅有一个，从这 $t-1$ 个合谋用户所知的信息来看， \mathbb{Z}_p 中任意元素为 $f(0)$ 的概率都是一样的，因此，任意 $t-1$ 个用户合谋也不能重构出秘密值 s 的任何信息。

2.5 区块链智能合约系统

区块链智能合约系统由开发者、合约当事人、合约执行环境（如虚拟机）和区块链网络等实体组成，如图 2-2 所示，其中合约执行环境和区块链网络共同构成了智能合约平台^{[93][94]}。上述实体具体功能介绍如下：

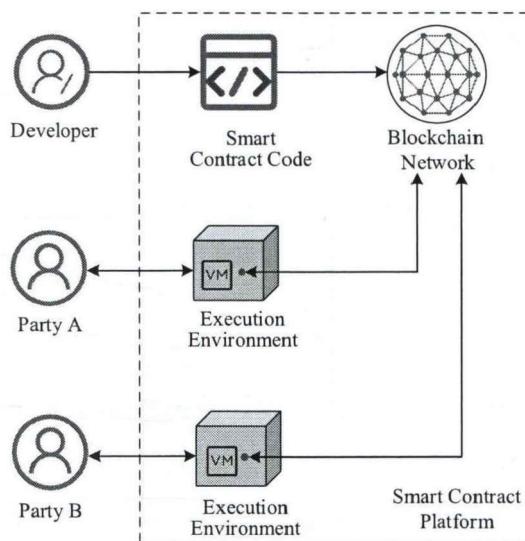


图 2-2 区块链智能合约系统

- 1) 合约开发者将现实业务按照商业规则编写智能合约程序，进而编译成智能合约代码，并将其部署至区块链网络。
- 2) 合约当事人是订立合约的两方或多边实体。它们通过智能合约平台完成合约的订立与交互式的执行，通常智能合约平台通过 Web 接口方式实现与当事人的交互，进而通过合约条款的执行来限制当事人之间的权利和义务关系。
- 3) 合约执行环境由执行机构和区块链节点组成。节点与区块链网络连接，可与区块链网络通信获取智能合约代码及程序当前运行状态，并将执

行后的结果及程序状态写回区块链；执行机构是执行智能合约代码的实体，通常采用虚拟环境（如虚拟机，Docker），并支持当事人身份认证、合约签名、激励机制、共识验证、条件触发等操作。

- 4) 区块链网络为智能合约提供了一个强有力的底层分布式通信能力和分布式账本形式的存储能力，它以交易形式记录合约代码、执行的中间状态及结果。

区块链智能合约通常支持面向合约软件开发，图 2-3 将介绍智能合约的开发、部署和执行过程中涉及到的三个实体，它们分别为编译器、公共账本和虚拟机。接下来，给出这些实体的具体介绍。

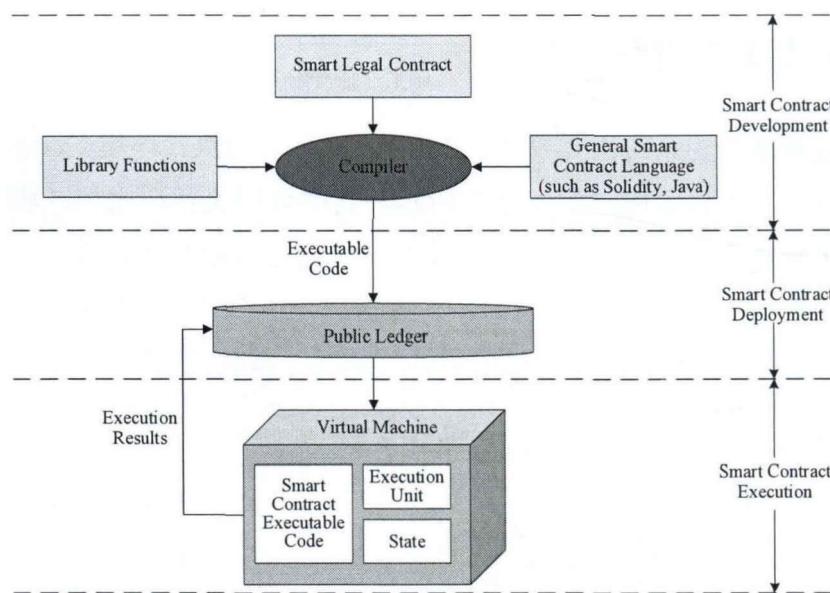


图 2-3 区块链智能合约开发、部署与执行流程

- 1) **编译器**是将高级智能合约编译成计算机可执行的程序及代码的工具。它的输入为高级智能合约，按照一定的转换规则（例如文献[95]介绍的规则），通过调用相关库函数，将高级智能合约转换为计算机可执行的程序代码（如 Java、Solidity 代码等）。
- 2) **公共账本**是用于部署智能合约可执行代码的实体，同时也能够支持存储当事人信息、标的物（资产）及程序状态等信息，它为智能合约提供了一个去中心化、不可篡改、公开透明的部署环境。
- 3) **虚拟机**是执行智能合约代码的工具。它从区块链平台获取合约可执行代码以及程序状态，通过执行模块，在满足合约中预定条件后，执行合约并将执行结果上传到区块链网络。

事实上，上述过程的起点是智能法律合约（Smart Legal Contract, SLC），它被用于智能合约的设计和开发。通过引入各式各样的库函数，大大降低开

发智能合约的难度。此外，很多成熟的商业化智能合约平台已经被提出，如以太坊和 Fabric，它们已经可以用于智能合约的部署和执行。

2.6 智能法律合约语言

表 2-1 基于 SPESC 智能法律合约语法介绍

合约模块名称		语法定义及语义规则	
合约框架		<i>Contract ::= Title{ Parties+ Assets+ Terms+ Additions+ Signs+ }</i>	
合约名称	@@合约标题：合约序号 <i>Title ::= contract Cname (: serial number Chash)?</i>		
当事人描述	@@当事人 群体？名称 {属性域+} <i>Parties ::= party group ? Pname {field+}</i>		
标的	@@资产 资产名称 {资产描述{属性域+} 资产权属{属性域+}} <i>Assets ::= asset Aname{ info{field+} right{field+}}</i>		
资产表达式	@@资产表达式：\$ (具体数量) ? (具体权属) ? 资产名称 <i>AssetExpressions ::= \$ (amount)? (right of)? Aname</i>		
资产操作	存入	@@存入 (满足某种价值关系的) ? 资产描述. <i>Deposits ::= deposit (value RelationOperator)? AssetExpression</i>	
	取回	@@取回 指定资产. <i>Withdraws ::= withdraw AssetExpression</i>	
	转移	@@转移指定资产到某当事人. <i>Transfers ::= transfer AssetExpression to target</i>	
合约条款	一般条款	@@条款名：当事人 (必须 可以 禁止) 行为 (属性域+) (执行所需的前置条件) ? (伴随的资产操作+) ? (执行后需满足的后置条件) ? <i>GeneralTerms ::= term Tname: Pname (must can cannot) action(field+) (when preCondition)? (while transactions+)? (where postCondition)?.</i>	
	违约条款	@@违约条款 条款名 (针对 条款名+) ? : 当事人 (必须 可以) 违约 处理 (属性域+) (执行所需的前置条件) ? (伴随的资产操作+) ? (执行后需满足的后置条件) ? <i>BreachTerms ::= breach term Bname (against Tname+)? : Pname (must can) action(field+) (when preCondition)? (while transactions+)? (where postCondition)?.</i>	
	仲裁条款	@@ (所声明之争议) ? 由某仲裁机构进行裁决. <i>ArbitrationTerms ::= arbitration term : (The statement of any controversy)? administered by institution : instName.</i>	
附加信息		@@ (属性域 +) 或 (附加信息 附加信息名 { 属性域 +}) <i>Additions ::= field + (addition Dname {field+})</i>	

智能法律合约语言是一种用于开发符合法律要求的智能合约的程序语言。通过遵守法律，它提供了正式的规范，包括词汇和语法规则，以确保智能法律合约具有与纸质合同类似的法律特征。作为实例，智能合约规范语言 [96][97][98] (SPEcification language of Smart Contract, SPESC) 一经提出就引起了学术界的广泛关注。SPESC 是基于传统合同的语法结构，以类似于自然语言的形式编写。如图 2-4 所示，它不仅明确规定了当事人的义务和权利，而且包含了具有约束力的合约的基本要素。因此，SPESC 可以促进法律专家和程序员之间的合作以设计并开发智能合约。

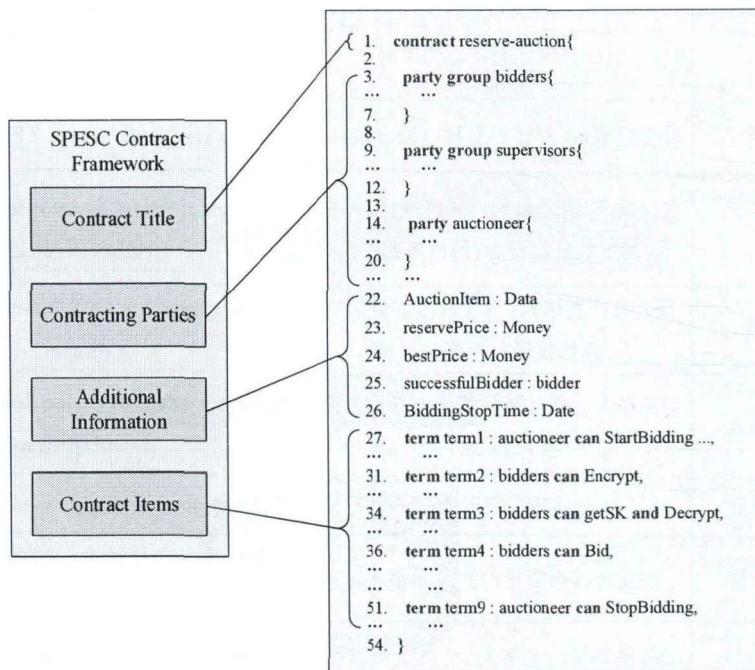


图 2-4 SPESC 合约结构

如图 2-4 所示，在一个以 SPESC 书写的智能法律合约中包含以下四个部分：合约名称、当事人、附加信息和合约条款。

- 1) 合约名称：一般来讲，合约的名称是对合同性质或目的的简单反映。
- 2) 合约当事人：当事人可以是个体，也可以是群组，当事人的属性和行为在这部分中被定义。
- 3) 附加信息：它是公开信息，合同各方均可获得用于支撑他们的行为。
- 4) 合约条款：条款是合约的主体，决定了合约中各方的权利和义务。

2.7 本章小结

本章介绍了在本文研究中所需要的基础预备知识，主要包括一些数学基础、困难性假设、密码学中的交互证明系统和秘密共享方案，此外，本章对区块链智能合约系统以及开发设计语言 SPESC 进行了描述。

3 支持交易数据多样化隐私需求的双模式广播加密

本章主要研究区块链智能法律合约中交易数据隐私保护问题。作为一种具有法律约束力的合同，智能法律合约近年来受到了广泛关注。然而，由于合约部署在公开透明的区块链网络中，所有交易数据都是公开可见的，这导致了交易数据隐私泄露的问题。针对这一问题，本章将广播加密机制引入到智能法律合约中，用来保护交易数据的隐私。在面向合约的编程思想的基础上，以合约条款形式对敏感数据的隐私进行声明并由编译器将广播加密算法链接到智能合约程序中实施保护。本章中将构造一种支持“选择”和“排它”双模式的基于身份的广播加密方案，来满足智能合约中敏感数据访问限定到指定人群和非冲突人群的隐私需求。

3.1 研究动机

面向合约的编程是现实世界纸质法律合同到可执行智能合约程序的有效途径，其中智能法律合约是它们之间过渡的桥梁。不同于智能合约，智能法律合约属于类自然语言描述的具有法律效力的协议，其中一些条款可转化为计算机可执行代码的形式。图 3-1 给出了合约转换的一般过程，首先，以自然语言描述的纸质法律合同可被翻译成由智能法律合约语言撰写的智能法律合约（如 SPESC）；进一步，它转化为由智能合约语言（如 Solidity, Serpent）编写的智能合约；最终，智能合约以交易的形式被上传至区块链网络中。在上述转换过程中，智能法律合约既是程序设计工具，又是开发工具。

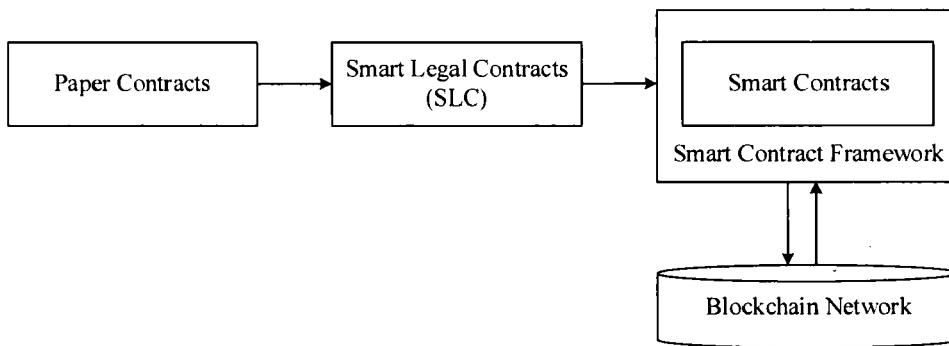


图 3-1 合约转换过程

在现实世界的法律合同中，通常使用保密条款来规定当事人对隐私信息的保护义务。例如在拍卖合同中，拍卖人应保证开标之前标底价格的机密性，并对标底价格泄露所造成的后果承担相应的法律责任。当采用区块链实现拍卖时，区块链的开放性对这种机密性带来了挑战。因此，智能法律合约应提供隐私保护机制，以确保在合约转换过程中保密条款得以实施。

加密技术是保护敏感数据隐私最有效的方法，本章希望找到一种能够同时满足交易中不同隐私要求的加密方案。本章将智能合约敏感数据隐私要求分为两类：

- 1) 敏感数据访问限定到指定人群(特定目的隐私)：交易中的私有数据只能由特定的用户组获得，这意味着只有在指定群组 S 中的用户才能获得相应的隐私数据；
- 2) 敏感数据访问限定到非冲突人群(一般目的隐私)：除了一些有冲突的用户，交易中的隐私数据可以被系统中的任何人获得，这意味着不在冲突群组 R 中的所有用户都可以获得相应的隐私数据。

然而到目前为止，几乎没有能够同时满足上述两种交易隐私要求的加密方案。为了保护智能法律合约中交易数据的隐私，包括特殊目的和一般目的隐私，需要提出一种新的加密方案，实现下面两种加密模式：

- 1) 选择加密模式：对于特定目的隐私，所提出的方案应该能够在一个特定的用户集 S 下加密私有数据，并且只有在集合 S 中的用户才能从密文中恢复数据；
- 2) 排它加密模式：对于一般目的隐私，提出的方案应该能够在特定用户集 R 下加密隐私数据，并且所有不在该 R 内的用户都均可从密文中成功恢复出原始数据。

作为一种密码原语，广播加密技术可以实现在开放网络中一个发送者和多个接收者的安全通信^[38]，其广播结构与区块链广播形式网络相吻合，有利于在发送者和特定接收者之间安全地共享敏感数据。因此，本文将基于广播加密设计一种新的加密方案来满足上述两种隐私要求。

自广播加密方案提出之后，具有不同功能和安全性的广播加密方案被相继研究。例如 Boneh 等人^[99]在合数阶双线性群下构造了一个抵抗任意数目用户合谋攻击的广播加密方案，在他们的方案中，任意数目的非授权用户联合也不能从密文中恢复出消息。此外，该方案的密文和私钥的规模均是常数级别。在此基础上，Delerablée^[100]构造一个新的抗完全合谋攻击的广播加密方案，该方案不仅具有常数规模大小的密文，而且可以在无需修改用户解密密钥的前提下实现新用户的动态加入。Goyal 等人^[101]采用证据加密（Witness Encryption, WE）构造了一种抗用广播加密，该方案中的密文、公钥和私钥大小均为常数尺寸与广播系统中用户总数无关。

2007 年，Delerablée^[43]以及 Sakai 和 Furukawa^[44]分别独立提出了身份基广播加密方案（Identity-based Broadcast Encryption, IBBE），在这些方案中，广播接收者的身份（例如姓名或 IP）作为公钥用于对消息加密。随后，Zhang

等人^[102]在合数阶群下构造了适应性安全的身份基广播加密，其私钥和密文长度均为常数级。Boneh 等人^[45]利用多线性映射构造出一个低开销的 IBBE 方案，在该方案中，密文、私钥和公钥尺寸均与用户总数呈对数相关。

为了保护广播接收者的身份隐私，Barth 等人^[46]首次将匿名性引入到广播加密中并构造了第一个匿名广播加密方案，该方案可以在不泄露接收者身份的前提下向多个接收者加密消息，其安全性证明依赖随机预言机模型。2013 年，Zhang 等人^[103]通过将匿名身份基加密与 IBBE 相结合，在标准模型下提出了匿名的基于身份的广播加密方案，他们的方案在合数阶群中的静态假设下被证明达到适应性安全。随后，He 等人^[104]在素数阶双线性群下提出了完全匿名的身份基广播加密方案，并证明其在适应性选择密文攻击下的安全性。

2016 年 Susilo 等人^[42]首次提出了接收者可撤销的身份基广播加密的概念，该方案首先将加密后的消息发送给第三方，第三方可从密文中撤销部分接收者身份。此后，Lai^[47]构造了一个完全隐私保护的可撤销身份基广播加密方案，能够同时保护广播接收者以及被撤销用户的身份隐私。最近，Zhu 等人^[49]在 Boneh 等人方案^[99]的基础上，提出了同时支持指定机制和撤销机制的广播加密，该方案可以通过授权集合的大小选择不同的广播模式，提高了广播加密方案效率。

然而，几乎没有一种广播加密方案能够同时满足选择和排它双加密模式。虽然文献[49]中的方案可以实现指定和撤销机制，但是该方案没有考虑广播接收者的隐私问题。因此，目前还没有一种具有双重加密模式的广播加密方案能够完美地满足智能合约中交易的不同隐私需求。

3.2 双模式身份基广播加密方案构造

本节中将介绍双模式基于身份的广播加密（Dual-Mode IBBE, DM-IBBE）方案的定义，并给出方案中具体算法的构造和分析。DM-IBBE 是在基于身份广播加密（Identity-Based Broadcast Encryption, IBBE）的基础上构造的，可以看作是一种具有灵活授权方式的身份基广播加密方案。特别地，该方案可以在同样的系统密钥情况下，同时支持“选择性加密模式”（Selective Encryption Mode, SEM）和“排它加密模式”（Exclusive Encryption Mode, EEM），以实现不同的授权需求。两种加密模式下的密文结构完全一致，在 SEM 中，任何属于指定接收集的用户都可以解密密文，而在 EEM 中，任何不在黑名单中的用户都可以从密文中恢复消息。

3.2.1 DM-IBBE 方案形式化定义

与基于身份广播加密方案类似，本章提出的 DM-IBBE 方案包括四个算法：系统建立算法 Setup ，私钥提取算法 Extract ，加密算法 Encryption 和解密算法 Decryption ，具体定义如下：

- 1) $\text{Setup}(1^\lambda) \rightarrow (PK, MSK)$ ：系统建立算法的输入为安全参数 1^λ ，输出系统的公钥 PK 以及主密钥 MSK 。
- 2) $\text{Extraction}(MSK, ID_i) \rightarrow (sk_i)$ ：用户私钥提取算法的输入为系统主密钥 MSK 以及身份标识 ID_i ，输出对应的用户私钥 sk_i 。
- 3) $\text{Encryption}(PK, M, S/R, Mode) \rightarrow CT$ ：加密算法的输入包括三部分，系统公钥 PK 和消息 M ，指定接收者集合 S （选择加密模式 SEM）或排除接收者集合 R （排它加密模式 EEM）以及加密模式 $Mode \in \{SEM, EEM\}$ ，最后加密算法输出密文 CT 。
- 4) $\text{Decryption}(CT, sk_i, Mode) \rightarrow M$ ：解密算法输入密文 CT ，用户私钥 sk_i 以及模式 $Mode \in \{SEM, EEM\}$ 。

在上述定义中，加密模式 $Mode$ 如下：

- 1) 在选择加密模式 SEM 中，对于任意的属于集合 S 中的用户，算法输出为消息 M ；对于任意其他用户，解密算法输出无效解密消息；
- 2) 在排它加密模式 EEM 中，对于任意的不属于集合 R 中的用户，算法输出为消息 M ；对于任意其他用户，解密算法输出无效解密消息。

具体而言，方案正确性要求任何授权用户当其身份标识属于授权集合 S ($ID_i \in S$) 或不属于非授权集合 R ($ID_i \notin R$) 的情况下都能从密文中成功解密出消息 M 。此外，上述定义还要求在不同的加密模式下密文结构完全相同。

3.2.2 DM-IBBE 构造方法

为了在同一个加密系统下实现 SEM 和 EEM 两种加密模式，并且保证不同模式下生成的密文结构相同，本文将采用多项式曲线形式将被选定的广播接收者集合 S 或被排除接收者集合 R 中的用户通过曲线进行表示，并选取曲线上的点作为该集合下的密文进行公开。用户通过自身私钥以及密文，重构多项式曲线并最终恢复出多项式中的秘密值，其基本思想是通过公开曲线上不同点的数据，将授权判定问题转换为多项式重构问题。具体构造步骤如下：

令全集 U 的两个非空子集 S 和 R 分别表示选定和排除的广播接受者集合，

其中 $|U|=m, |S|=l, |R|=k$ 。对于“选择加密模式”，首先将子集 S 中的元素映射成二维平面中的 l 个点，再选取空间中与上述映射点不同的一个点，通过上述 $l+1$ 个不同点利用拉格朗日插值重构出唯一的 l 次多项式 $h_s(x)$ 作为子集 S 的安全表示，随后，选择曲线 $h_s(x)$ 上的不同于集合 S 中的 l 个点作为选择加密模式下的密文进行公布。

解密阶段，在“选择加密模式”下，用户 ID_e 可利用自身私钥 $(x_e, h_s(x_e))$ 以及密文 $\{(x_i, h_s(x_i))\}_{i \in [1, l]}$ 通过拉格朗日插值重构出多项式 $h_s(x)$ ，

$$h_s(x) = \begin{cases} \sum_{i=[1, l], i=e} \left(\prod_{j \in [1, l], i \neq j} \frac{x - x_j}{x_i - x_j} \cdot h_s(x_i) \right), & ID_e \in S \\ \perp, & ID_e \notin S \end{cases} \quad (3-1)$$

在上式中，若 $ID_e \in S$ 则用户可正确重构出多项式 $h_s(x)$ ；若 $ID_e \notin S$ 则由于此时 $(x_e, h_s(x_e))$ 不在曲线 $h_s(x)$ 上则重构的多项式是无效的，用符号 \perp 表示。

与上述构造方法类似，对于“排它加密模式”，首先，将全集 U 中的元素映射成二维平面中的 m 个点，再选取空间中与上述映射点一个不同的点，通过上述 $m+1$ 个不同点利用拉格朗日插值重构出唯一的 m 次多项式 $h_R(x)$ 作为子集 R 的安全表示，随后，选组曲线 $h_R(x)$ 上的关于集合 R 的 k 个点以及不属于全集 U 的任意 $m-k$ 个点作为排它加密模式下的密文 CT 进行公布。

同样地，在“排它加密模式”下用户 ID_e 可利用自身私钥 $(x_e, h_s(x_e))$ 以及密文 $\{(x_i, h_R(x_i))\}_{i \in [1, m]}$ 通过拉格朗日插值重构出多项式 $h_R(x)$ ，

$$h_R(x) = \begin{cases} \sum_{i=[1, m], i=e} \left(\prod_{j \in [1, m], i \neq j} \frac{x - x_j}{x_i - x_j} \cdot h_R(x_i) \right), & ID_e \notin R \\ \perp, & ID_e \in R \end{cases} \quad (3-2)$$

在上式中，若 $ID_e \notin R$ ，则用户可正确重构出多项式 $h_R(x)$ ；若 $ID_e \in R$ 则由于

此时必有一项 $\frac{x - x_j}{x_i - x_j}$ 的分母为零（即对于给定的 x_e ，存在 x_j ，使得 $x_e = x_j$ ），

此时用户无法重构曲线，用符号 \perp 表示。

上述双模式广播加密方案的构造方法如表 3-1 所示，在不同的加密模式下，通过插值点选取的不同重构出不同的曲线表示形式。在密文选择方面，在选择加密模式下，令 $h(x)$ 表示 $h_s(x)$ ，则 $x_j \neq x_i$ 表示在曲线 $h_s(x)$ 上且不属于集合 S 中的点；在排它加密模式下，令 $h(x)$ 表示 $h_R(x)$ ，则 $x_j = x_i$ 表示在集合 R 中的点 $x_j \notin U$ 表示在曲线 $h_R(x)$ 上且不属于全集 U 中的点。在解密阶段，

选择加密模式下只有属于子集 S 的解密者可以重构出曲线 $h_S(x)$, 排它加密模式下只有在全集 U 且不属于冲突集 R 的解密者可以重构出曲线 $h_R(x)$ 。

表 3-1 不同加密模式下的构造方法

加密模式	拉格朗日插值曲线	插值点选取 ($x_i, h(x_i)$)	密文选取 ($x_j, h(x_j)$)	解密条件
选择加密模式	$h_S(x)$	$x_i \in S$	$x_j \neq x_i$	$x_e \in S$
排它加密模式	$h_R(x)$	$x_i \in U$	$x_j = x_i, \text{for } j \in [1, l]$ $x_j \notin U, \text{for } j \in [l+1, m]$	$x_e \notin R \text{ 且}$ $x_e \in U$

对于给定的全集 $U = \{ID_1, ID_2, \dots, ID_m\}$ 以及两个子集 $S, R \subseteq U$, 构造双模式广播加密方案的核心是基于安全重构函数 $Reconst$ 的多项式重构, 函数的输出是关于子集的多项式, 其具体定义如下。

令 PK 和 CT 分别是公钥和密文, sk_{ID} 为用户 ID 的私钥, 则重构函数 $Reconst$ 是一个确定性多项式时间算法, 使得 $Reconst(PK, CT, sk_{ID}) = h(x)$ 并满足以下条件,

$$\Pr \left[Reconst(PK, CT, sk_{ID}) = \begin{cases} h_S(x), & ID \in S \\ h_R(x), & ID \notin R, ID \in U \\ \perp, & ID \notin U \end{cases} \right] \geq 1 - \epsilon \quad (3-3)$$

其中 ϵ 为可忽略概率。

为了保证上述多项式输入 x 的隐私性以及多项式的不可预测性, 在双模式广播加密方案的具体构造中引入离散对数假设来进一步完善上述多项式的构造。通过 $(g^{x_i}, g^{h_S(x_i)})$ 来代替 $(x_i, h_S(x_i))$, 借助离散对数假设增强 x_i 的隐私性以及 $g^{h_S(x_i)}$ 的不可预测性, 甚至在 g^{x_i} 公开的情况下, 其安全性也能得到保证。

3.2.3 DM-IBBE 方案构造

在这一节中, 介绍 DM-IBBE 的具体构造, 根据上述关于双模式身份基广播加密定义, 描述定义中四个算法的具体设计。需要注意的是, 在构造的具体方案中, 用户数量除了不能超过陷门值 $m+1$ 之外再无其它任何限制, 即本章方案中的最大用户数为 m 个。具体来说, 在接下来构造的 DM-IBBE 方案中, 最大用户集可以描述为 $U = \{ID_1, ID_2, \dots, ID_m\}$ 。

Setup 算法输入安全参数 1^λ , 输出系统公钥 PK 以及主密钥 MSK 。该算法由系统管理员运行, 在输入安全参数后, 算法返回 p 阶循环群 \mathbb{G} 以及 \mathbb{G} 中

任意生成元 g 。令哈希函数 $Hash: \{0,1\} \rightarrow \mathbb{Z}_p^*$, 该算法随机选取 $\{a_i \in \mathbb{Z}_p\}_{i=0}^m$ 并令一个 m 阶多项式为 $f(x) = \sum_{i=0}^m a_i x^i \bmod p$, 算法输出系统公钥如下:

$$PK = \left\langle g, H, \left\{ g^{a_i} \right\}_{i \in [0, m]} \right\rangle \quad (3-4)$$

最后, 该算法秘密保存主私钥 $MSK = \{a_i\}_{i \in [0, m]}$ 。

Extraction 算法由系统管理员运行, 输入系统主私钥 MSK 和用户身份 ID_i , 输出用户对应的私钥 sk_i 。对于任意的用户 ID_i , 该算法首先计算 $H(ID_i) = x_i$ 并借助系统主私钥计算用户私钥 $sk_i = f(x_i)$ 。随后, 算法将私钥 sk_i 秘密地发送给用户 ID_i 。在用户私钥生成过程中, 应该注意请求私钥的用户数量除了要小于阈值外, 没有任何限制, 即方案中的最大用户数目是 m 。不失一般性, 本文假设最大用户集为 $U = \{ID_1, ID_2, \dots, ID_m\}$ 。

Encryption 算法输入系统公钥 PK , 消息 M , 加密模式 $Mode$ 以及指定的接收者集合 S 或黑名单集合 R , 最后算法输出密文 CT 。接下来将分别从选择加密模式和排它加密模式两个方面介绍密文生成过程。

选择加密模式: 在该模式下, 加密者首先选择指定接收消息的用户集合 $S = \{ID_{s1}, ID_{s2}, \dots, ID_{sl}\}$ 并通过以下步骤加密消息 M , 其中 $|S| = l \leq m$ 。

- 1) 对于任意的 $ID_i \in S$, 加密者计算 $H(ID_i) = x_i$ 和 $T_i = g^{f(x_i)} = \prod_{j=0}^m g^{a_j x_i^j}$ 。
- 2) 令 $x_0 = 0$, 加密者随机选取 $s \in \mathbb{Z}_p^*$ 并计算 g^s 。
- 3) 令 $h_s(x)$ 为 l 阶多项式并使得以下等式成立

$$g^{h_s(x)} = \begin{cases} T_i, & ID_i \in S \\ g^s, & x = 0 \end{cases} \quad (3-5)$$

在上述步骤 1) 和步骤 2) 中 $\{(x_i, T_i)\}_{ID_i \in S}$ 和 $(x_0 = 0, g^s)$ 的帮助下, 加密者能够通过拉格朗日插值公式重构出 $g^{h_s(x)} = g^{s \cdot L_0(x)} \cdot \prod_{ID_i \in S} T_i^{L_i(x)}$, 其中 $L_i(x)$ 为拉格朗日插值系数, 即 $L_i(x) = \prod_{j \neq i} \frac{x - x_j}{x_i - x_j}$ 。

- 4) 在该步骤中, 加密者在 \mathbb{Z}_p^* 中随机选取 α 和 $\{\tau_i\}_{i \in [1, l]}$, 其中对于任意的 $i \in [1, l]$ 和 $j \in [1, m]$ 有 $\tau_i \neq H(ID_j)$ 。随后, 加密者计算密文组件 $C_0 = g^\alpha$ 以及 $\{C_{i,1} = \tau_i, C_{i,2} = g^{h_s(\tau_i) \cdot \alpha}\}_{i \in [1, l]}$ 。
- 5) 最后加密者计算 $C_0 = g^{\alpha \cdot s} \cdot M$, 则在选择加密模式下密文可表示如下:

$$CT = \left\langle C_0, C_1, \left\{ C_{i,1}, C_{i,2} \right\}_{i \in [1, l]} \right\rangle \quad (3-6)$$

排它加密模式：在该加密模式下，加密者首先选择冲突用户组成的集合 $R = \{ID_{r_1}, ID_{r_2}, \dots, ID_{r_k}\}$ 并通过以下步骤加密消息 M ，其中 $|R| = k \leq m$ 。

- 1) 对于任意的 $ID_i \in U$ ，加密者计算 $H(ID_i) = x_i$ 和 $T_i = g^{f(x_i)} = \prod_{j=0}^m g^{a_j x_i^j}$ ；
- 2) 令 $x_0 = 0$ ，加密者随机选取 $s \in \mathbb{Z}_p^*$ 并计算 g^s ；
- 3) 令 $h_R(x)$ 为 m 阶多项式并使得以下等式成立

$$g^{h_R(x)} = \begin{cases} T_i, & ID_i \in U \\ g^s, & x = 0 \end{cases} \quad (3-7)$$

在步骤 1) 和步骤 2) 中 $\{(x_i, T_i)\}_{ID_i \in U}$ 和 $(x_0 = 0, g^s)$ 的帮助下，加密者能够通过拉格朗日插值公式重构出 $g^{h_R(x)} = g^{s \cdot L_0(x)} \cdot \prod_{ID_i \in U} T_i^{L_i(x)}$ ；

- 4) 加密者在 \mathbb{Z}_p^* 中随机选取 $\alpha \in \mathbb{Z}_p^*$ ，为了便于描述，对于任意冲突集合 R 中的用户，计算 $H(ID_j) = x_j = \delta_j \in \mathbb{Z}_p^*$ ，其中 $j \in [1, k]$ ；随后，在 \mathbb{Z}_p^* 中的随机元素 δ_j ，其中 $j \in [k+1, m]$ 且 $\delta_j \neq x_i$ ；最后，加密者计算密文 $C_1 = g^\alpha$ 和 $\{C_{j,1} = \delta_j, C_{j,2} = g^{h_R(\delta_j) \cdot \alpha}\}_{j \in [1, m]}$ ；
- 5) 最后加密者计算 $C_0 = g^{\alpha \cdot s} \cdot M$ ，则在排它加密模式下密文可表示如下：

$$CT = \left\langle C_0, C_1, \{C_{j,1}, C_{j,2}\}_{j \in [1, m]} \right\rangle \quad (3-8)$$

Decryption 算法输入密文 CT ，解密者私钥 sk_i 以及加密模式 $Mode$ ，输出明文消息 M 或者空字符 \perp ，该算法被解密用户执行用来恢复密文中的敏感信息。对于选择加密模式下生成的密文，若解密者 $ID_i \in S$ ，则可通过执行解密算法恢复消息 M ；若 $ID_i \notin S$ ，则算法返回 \perp 。在排它加密模式下，若解密者 $ID_i \notin R$ ，则可通过执行解密算法恢复消息 M ；若 $ID_i \in R$ ，则算法返回 \perp 。接下来，描述具体的解密过程。

- 1) 解密者 ID_i 通过其私钥 sk_i 计算 $H(ID_i) = x_i$ 以及 $(C_1)^{sk_i} = g^{\alpha \cdot f(x_i)}$ ；
- 2) 为了便于描述，令 $x_{i,1} = C_{0,1}$ ，解密者 ID_i 利用拉格朗日插值公式计算：

$$g^{\alpha \cdot s} = \begin{cases} C_1^{sk_i \cdot L_0} \cdot \prod_{i \in [1, l]} C_{i,2}^{L_i}, & Mode = SEM \\ C_1^{sk_i \cdot L_0} \cdot \prod_{i \in [1, m]} C_{i,2}^{L_i}, & Mode = EEM \end{cases} \quad (3-9)$$

上式中 L_i 为拉格朗日插值系数，即 $L_i = \prod_{j \neq i} \frac{C_{j,1}}{C_{j,1} - C_{i,1}}$ ；

- 3) 最后，解密者 ID_i 恢复敏感数据 $M = C_0 / g^{\alpha \cdot s}$ 。

正确性: 接下来验证上述 DM-IBBE 方案的正确性。在选择加密模式下, 对于任意解密者 $ID_i \in S$, 他可以通过其私钥 sk_i 从密文中恢复消息 M 如下:

$$\begin{aligned} \frac{C_0}{C_1^{sk_i \cdot L_0} \cdot \prod_{i \in [1, m]} C_{i, 2}^{L_i}} &= \frac{M \cdot g^{\alpha \cdot s}}{g^{\alpha \cdot f(x^*) \cdot L_0} \cdot \prod_{i \in [1, m]} g^{h_R(\delta_i) \cdot \alpha \cdot L_i}} \\ &= \frac{M \cdot g^{\alpha \cdot s}}{g^{\alpha \cdot \sum_{i=0}^m h_R(\delta_i) \cdot L_i}} = \frac{M \cdot g^{\alpha \cdot s}}{g^{\alpha \cdot s}} = M \end{aligned} \quad (3-10)$$

在排它加密模式下, 对于任意解密者 $ID_i \notin R$, 他可以通过其私钥 sk_i 从密文中恢复消息 M 如下:

$$\begin{aligned} \frac{C_0}{C_1^{sk_i \cdot L_0} \cdot \prod_{i \in [1, l]} C_{i, 2}^{L_i}} &= \frac{M \cdot g^{\alpha \cdot s}}{g^{\alpha \cdot f(x^*) \cdot L_0} \cdot \prod_{i \in [1, l]} g^{h_S(\tau_i) \cdot \alpha \cdot L_i}} \\ &= \frac{M \cdot g^{\alpha \cdot s}}{g^{\alpha \cdot \sum_{i=0}^l h_S(\tau_i) \cdot L_i}} = \frac{M \cdot g^{\alpha \cdot s}}{g^{\alpha \cdot s}} = M \end{aligned} \quad (3-11)$$

3.3 DM-IBBE 在智能合约交易隐私保护中的应用

随着智能合约在医疗、供应链、金融服务等领域的广泛应用, 交易中的隐私泄露问题变得越来越严重。隐私泄露一旦发生, 不仅会造成严重的商业损失, 还会侵犯个人利益。因此, 迫切需要寻找保护区块链智能合约交易隐私的方法, 这也是本章的研究重点之一。区块链对于记录交易非常可靠, 因为区块链账本的完整副本由所有节点维护。这意味着一个节点的离线不会影响账本对网络中其它所有参与者的可用性。然而, 考虑到区块链网络的开放性, 部署在其上的智能合约对所有节点都是公开可用的。这一事实将不可避免地导致合同交易中敏感数据的泄露。

作为第二代区块链核心技术, 智能合约的本质是存储在区块链上的计算机代码, 在触发预制条件下, 可被计算机自动执行, 也称为区块链智能合约。区块链智能合约支持面向合约编程, 它具有可编程性、自动执行、有效监督和法律化等特点, 因此受到广泛关注并被应用于金融、数字资产管理、物联网等诸多领域。然而, 由于区块链网络的开放性, 使得部署在其上的智能合约是公开透明的, 因此交易中的所有信息也都是公开可见的, 这势必导致交易中敏感信息泄露问题。例如, 在竞拍合约中, 开标之前标底价应是对所有投标人保密的, 若采用传统的开发模式设计竞拍合约, 交易中的隐私信息(标底价)将被直接存储在区块链上, 从而导致隐私泄露。因此, 对智能合约中交易隐私的保护是十分必要和迫切的。本节中, 将展示利用本章提出的双模

式身份基广播加密方案来实现对智能法律合约中交易隐私的保护。

3.3.1 具有隐私保护功能的智能法律合约实例

本节以标底价拍卖为例，介绍如何开发智能合约以满足合约中交易的隐私要求。标底价拍卖是指拍卖师在拍卖前设定一个底价，使出价最接近底价的竞买人获得拍卖物品。拍卖人开标后，监督者（公证员）可以通过比较中标的价和底价来监督本次拍卖的有效性。

需要注意的是，为保证拍卖的公平性，拍卖师在开标前底价必须对竞标者和监督人员保密。为了保护拍卖物品的隐私，在商业拍卖过程中，拍卖物品的详细信息只向符合条件的竞拍者开放。在竞拍前，拍卖人应当对所有意向竞拍者进行资格审查，将未通过审查的竞拍者列入黑名单。因此，在标底价拍卖合同中，存在两种隐私需求：

- 1) 特定目的隐私，即底价是敏感信息，只有在拍卖者开标之后，监督人员才能获得底价信息。
- 2) 一般目的隐私，强调拍卖物品是敏感信息，只有不在黑名单中的竞标者才能了解拍卖物品详细信息。

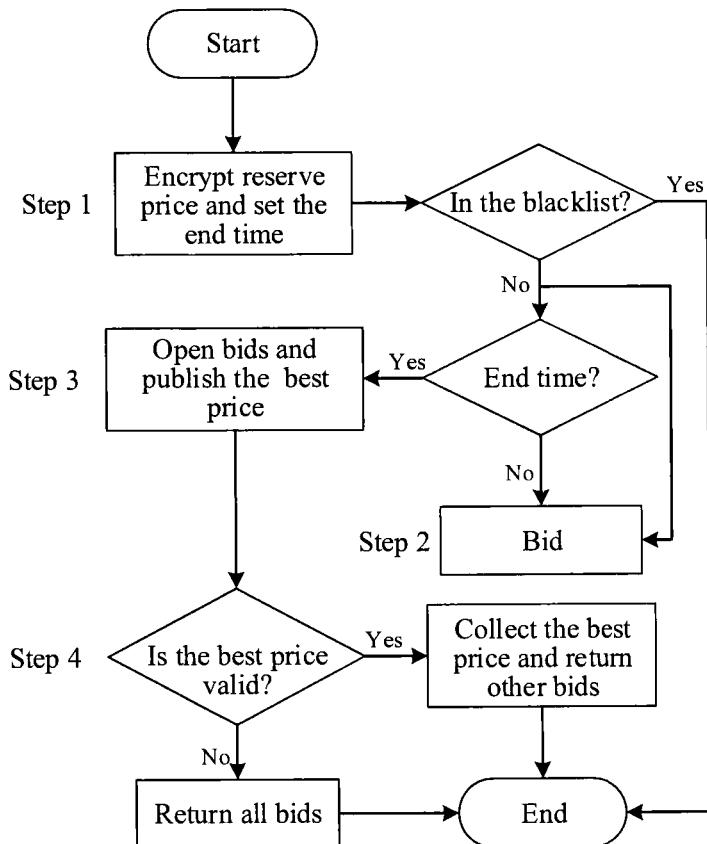


图 3-2 标底价竞拍执行流程

如图 3-2 所示，标底价竞拍具体流程包括以下四步，1)由拍卖者开启竞拍程序，并设置竞拍的结束时间和标底价格，为了确保标底价和拍卖物品的机密性，拍卖者需对标底价以及拍卖物的详细信息进行加密处理。2)在竞拍过程中，未列入黑名单的投标人可以随时投标，同时上交相应的押金至资金池。3)竞拍结束后，拍卖者应开标并选出中标人以及对应的出价（称为最优价），然后，拍卖师向所有竞标者和监督者公布最优价。4)最后，监督者从密文中解密标底价格，以判断其是否有效。若此次竞拍有效，拍卖师会收取中标价对应的押金，并退后所有未中标的竞拍者的押金；若无效，则所有参与竞标人员都将从资金池中提取他们的押金。

根据上述竞拍流程，利用高级智能合约语言 SPESC 可以编写标底价竞拍合约（reserve auction）智能合约如图 3-3 所示。在该图展示的 SPESC 合约中包括三类：投标人（bidder）、监督者（supervisior）和拍卖师（auctionner）三类当事人。

投标人为注册为参加投标的用户群组，他们可以执行以下操作：

- 1) amount 用于记录押金并撤回无效投标；
- 2) Bid 用于在拍卖过程中出价；
- 3) WithdrawOverbidMoney 用于撤回押金。

监督者为群组当事人，他们可以监督开标结果的有效性，执行以下操作：

- 1) getSK 用于获取自身私钥；
- 2) Decrypt 用于解密密文；

拍卖师是管理竞拍流程的个体当事人，可以执行以下操作：

- 1) StartBidding 用于开启竞拍程序；
- 2) getPK 用于获取本次竞拍活动中的公钥；
- 3) Encrypt 用于加密敏感信息；
- 4) OpenTheBid 用于开标，以找到中标者和最优价；
- 5) StopBidding 用于终止本次拍卖程序。

为了支持合约参与者上述操作，该合约中引入了以下四个附加信息：

- 1) AuctionItem 表示拍卖物品的详细信息；
- 2) reservePrice 表示标底价格；
- 3) successfulBidder 代表中标者；
- 4) bestPrice 表示中标价；
- 5) BiddingStopTime 代表拍卖终止时间。

```

1. contract reserve auction{
2.
3.   party auctioneer{
4.     getPK()           16.    }
5.     Encrypt()        17.
6.     OpenTheBid()      18.    party group supervisors{
7.     StartBidding(BiddingTime: Date 19.      getSK()
8.     StopBidding()       20.      Decrypt()
9.   }                   21.    }
10.                          22.
11.   party group bidders{ 23.    AuctionItem : Data
12.     amount : Money      24.    reservePrice : Money
13.     Bid()               25.    bestPrice : Money
14.     WithdrawOverbidMoney() 26.    successfulBidder : bidder
15.     Decrypt()          27.    BiddingStopTime : Date
16.
17.
18.
19.
20.
21.
22.
23.
24.
25.
26.
27.
28. term term1 : auctioneer can StartBidding and getPK,
29.   when before auctioneer did StartBidding
30.   while Encrypt reservePrice //对底价信息进行加密
31.   where BiddingStopTime = BiddingTime + now.
32. term term2 : auctioneer must Encrypt,
33.   when after auctioneer did StartBidding and before bidders did Bid,
34.   while Encrypt AuctionItem.
35. term term3 : bidders can getSK and Decrypt, //竞拍者获取私钥并进行解密
36.   when after auctioneer did Encrypt and before bidders did Bid.
37. term term4 : bidders can Bid,
38.   when after auctioneer did StartBidding and before BiddingStopTime.
39. term term5 : auctioneer must OpenTheBid,
40.   when after BiddingStopTime and before auctioneer did StopBidding
41.   while publish reservePrice, successfulBidder and bestPrice
42.   where bestPrice = the bidding price of successfulBidder.
43. term term6 : supervisors can getSK and Decrypt, //监督者获取私钥并进行解密
44.   when after auctioneer did OpenTheBid.
45. term term7 : bidders can WithdrawOverbidMoney,
46.   when bestPrice isn't very close to decrypted reservePrice
47.   while withdraw deposit $this bidder :: amount.
48. term term8 : bidders can WithdrawOverbidMoney,
49.   when bestPrice is very close to decrypted reservePrice and
50.     this bidder isn't successfulBidder //对竞拍者出价与底价进行对比
51.     while withdraw deposit $this bidder :: amount.
52. term term9 : auctioneer must StopBidding,
53.   when after BiddingStopTime and before auctioneer did StopBidding
54.   while withdraw $bestPrice.
55. }

```

图 3-3 基于 SPESC 编写的标底价竞拍合约

此外，在该智能法律合约中包含九项条款，这些条款对应于图 3-3 中标底价拍卖过程中的四个步骤。接下来，本文将对合约中的九项条款与拍卖过程的四个步骤之间的对应关系进行详细描述。

条款 1 和条款 2 对应于有底价拍卖过程中的步骤 1)，它是指拍卖者开启

招标程序，并获取合约当事人的公钥信息。然后对拍卖品的底价和细节信息进行加密，以满足不同的隐私需求。条款 3 和条款 4 对应于有底价拍卖过程中的步骤 2)，这意味着未被列入黑名单的竞标者可以在拍卖师开始竞拍程序后且在拍卖停止之前，进行出价竞拍。条款 5 对应于有底价拍卖过程中的步骤 3)，它描述了拍卖师开标并找到中标者，然后将底价和最优价公布给监督者。条款 6 到 9 对应于有底价拍卖过程中的步骤 4)，其中条款 6 描述监督者解密底价对应的密文，条款 7、8 描述若最优价与解密出的底价相差很大，即它们之间的差值大于一个预定的值，那么本次拍卖无效，所有竞拍者从押金池中撤回他们的押金；否则，只有未中标者撤回他们的押金。条款 9 表示拍卖者收取中标者的押金并结束拍卖。

在 SPESC 书写的有底价竞拍智能法律合约中，交易中的隐私数据通过本章提出的 DM-IBBE 方案进行保护。接下来，给出该合约与本章提出的方案中的算法之间的对应关系。

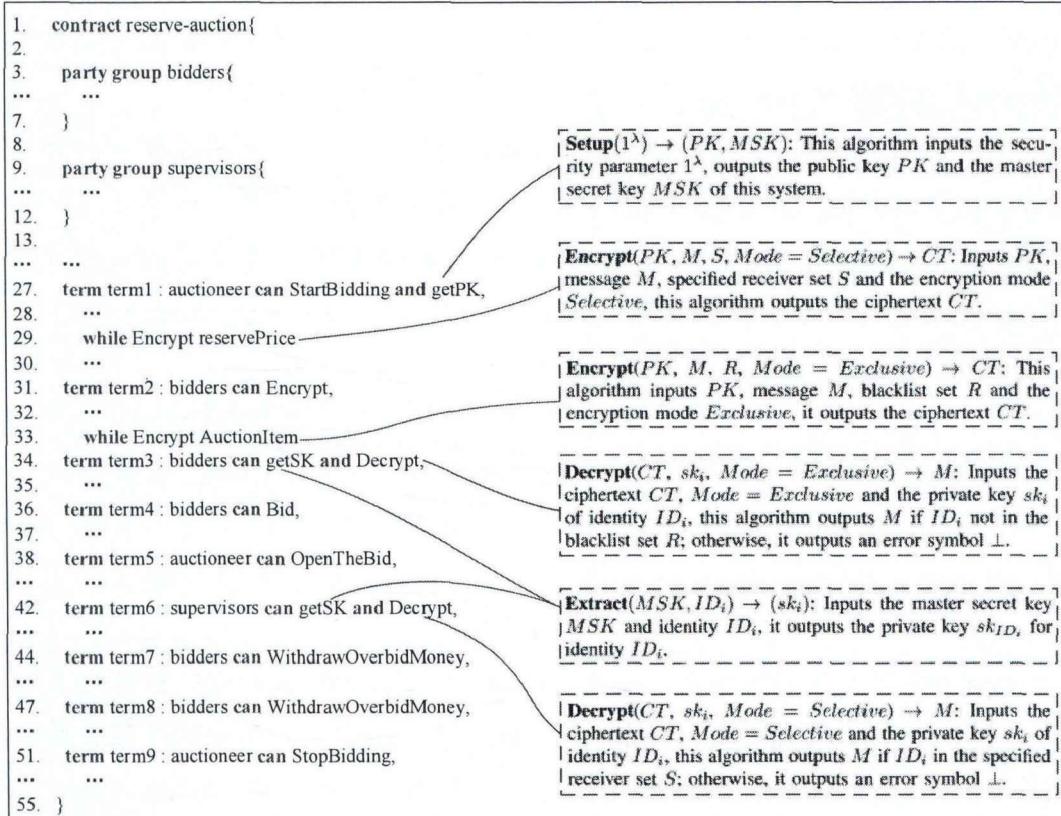


图 3-4 SPESC 竞拍合约与 DM-IBBE 算法中间关系

如图 3-4 所示，合约条款 1 中的 `getPK` 操作对应于 DM-IBBE 方案中的 `Setup` 算法，拍卖者可以通过执行该操作来获取系统公钥，其中包括监督者和竞拍者的公钥。该条款中的 `Encrypt` 操作对应于方案中“选择加密模式”下的 `Encrypt` 算法，该动作由拍卖者执行，用来加密底价信息。然而，条款 2 中的

Encrypt 操作对应于方案中“排它加密模式”下的 Encrypt 算法，用于加密拍卖物品的详细信息。在条款 3 和条款 6 中，getSK 操作对应于方案中的 Extract 算法，竞拍者和监督者分别利用该操作来获取他们的私钥。条款 3 和条款 6 中 Decrypt 操作分别对应于“选择加密模式”和“排它加密模式”下的 Decrypt 算法。通过执行此操作，竞拍者和监督者可以从密文中恢复出相关信息。

3.3.2 基于 DM-IBBE 竞拍合约的实现

基于 DM-IBBE 的竞拍合约的开发包括两部分：基于 SPESC 语言的合约程序和基于 Java 语言开发的密码函数库。前者通过标准化 SPESC 语言³进行开发，并通过编译器转化为 Java 语言智能合约程序框架，在此基础上将后者与所生成的框架进行融合。考虑到基于 Java 语言开发的密码函数库调用了 Java 下基于双线性对的密码库 JPBCv2.0⁴，尚不能直接部署到 Solidity 等商业区块链系统，因此合约实现采用了实验室自行开发的基于 Java 虚拟机智能合约平台 JuLiuSC v1.0⁵。该区块链平台支持 JAR 的软件部署，底层采用了基于 BitCoin 架构的区块链系统，支持基于交易的智能合约代码部署。

基于 DM-IBBE 的竞拍合约在 JuLiuSC 区块链系统上实现过程可以分为部署和执行两个阶段。如图 3-5 所示，部署阶段包括两个步骤，首先通过 Java 开发工具将 DM-IBBE 中的算法和 JPBC 密码库编译成 JAR 形式的智能合约代码，然后将代码上传到区块链网络。

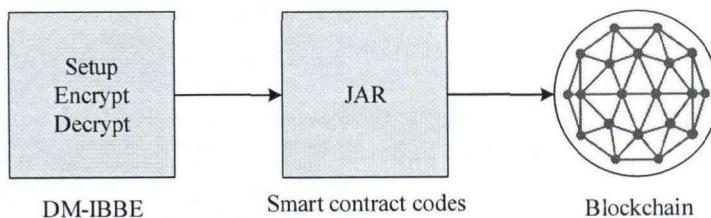


图 3-5 基于 DM-IBBE 的竞拍合约部署流程

DM-IBBE 方案被执行在一个名为 ContractChain 的 JuLiuSC 区块链子链上，该链以 JSON(JavaScript Object Notation) 格式提供面向事务的数据存储，它是一种轻量级的“键-值”对式数据并支持数组和集合结构。getPK、Encrypt 和 Decrypt 的执行结果分别如图 3-6、图 3-7 和图 3-8 所示。在这些结果中，txid 是交易的标识，funName 是操作名称，userResult 表示操作的执行结果。

³ SPESC 语言遵循中国电子学会《区块链智能合约形式化表达》标准

⁴ 密码库见 <http://gas.dia.unisa.it/projects/jpbc>

⁵ 该平台为一种面向法律的 JuLiuSC 智能合约平台，软著登记号 7047209

```

"txid" : "9563012b831ee57e428526291cd9dbed01f49f8bd22b5500f4cc9e9b9d49e3bf" ,
"ContractInput" : {
    "funName" : "getpk"
},
"Contractoutput" : {
    "userResult": "getPK success!
PK_0= 416518662179826911693097472979664274…01816832049863218298738674,
PK_1= 766431135900693096721276047594820191…07301287833658101014975848,
PK_2= 1812463681947881956435923469236856329…8615961709154543568297434,
...
PK_9=6912670725353010246521…71369259213814609294937810775342719316021"
}

```

图 3-6 getPK 动作执行结果

在图 3-6 中，拍卖者触发 getpk 动作以获取系统公钥，令 DM-IBBE 方案中用户总数为 10，则拍卖人最终获得用户公钥为 {PK_0,PK_1,…,PK_9}。

```

"txid" : "b0306a520639ce9d44ac5f88af1ef90b12af1f966722c505502105d69a06dfac" ,
"ContractInput" : {
    "funName" : "Encrypt"
},
"Contractoutput" : {
    "userResult": "Encrypt success!
C_0= 463759300117897456869454687697421…892063565967831436952920218473,
C_1= 56789686784115216525642731047887…7754688027341638938703578242598,
C_01= 119979425875216119517115057646931257121417,
C_02= 4494148483360498384206755439951…8469489686193418717953092423894,
...
C_31= 187430179395934536458724754369317378695716567,
C_32= 2542529351652780406523720…5541396546501787645923494259376587610"
}

```

图 3-7 Encrypt 动作执行结果

如图 3-7 所示，以选择加密模式为例来展示 Encrypt 动作的执行结果。在该操作中，令指定接收者的数量为 4，底价用 Message 表示，拍卖者触发 Encrypt 动作，最终生成密文 {C_0,C_1,{C_i,1,C_i,2}}，其中 i 属于 1 到 3。

最后，如果竞拍者在授权的接收者集合中，则其可以触发 Decrypt 动作并获得该 Message 的解密结果如图 3-8 所示。

```

"txid" : "7dc2e71ff32f0fae59c619f66e3e9f153e6fcf8109e17a6569c60af9afc5fc3d" ,
"ContractInput" : {
    "funName" : "Decrypt"
},
"Contractoutput" : {
    "userResult": "Decrypt success!
Message= 357373375195729738107610855864254849647348421628308469790404
591273524931259570486962…554261004145239765565396824639932419846673368
742860212290039407"
}

```

图 3-8 Decrypt 动作执行结果

3.4 安全性分析

本章提出的 DM-IBBE 方案，要求对于任何概率多项式时间攻击者，在没有有效私钥的情况下均不能从密文中获得消息，这被称为选择明文攻击下的不可区分性 (Indistinguishability under Chosen-plaintext Attacks, IND-CPA)。此外，还要求即使攻击者拥有有效的密钥，也无法在该方案的选择加密模式中获取接收者的任何身份信息，这称为选择明文攻击下的匿名性 (Anonymity under Chosen-plaintext Attacks, ANO-CPA)。接下来，将通过对手 \mathcal{A} 和挑战者 \mathcal{C} 之间游戏来定义 IND-CPA 和 ANO-CPA 安全，并给出详细证明过程。

3.4.1 DM-IBBE 方案选择明文攻击下的不可区分性

在证明本章所提方案安全性之前，首先介绍其安全性定义。该方案选择明文攻击下的不可区分性是通过以下游戏进行定义的，具体描述如下。

- 1) 初始阶段：对手 \mathcal{A} 首先选取全集 U 的一个子集 S^* 作为挑战集进行公布，被排除的接收者集合 R^* 可以表示为 $R^* = U \setminus S^*$ ，其中 U 为系统中所有用户组成的集合。
 - 2) 系统建立阶段：首先，挑战者 \mathcal{C} 执行原始方案中的 Setup 算法，获得系统公钥 PK 和主密钥 MSK 。随后，挑战者将系统公钥 PK 发送给对手 \mathcal{A} ，并秘密保存主密钥 MSK 。
 - 3) 学习阶段 I：在该阶段，对手将对用户 ID^* 进行私钥查询。若 $ID^* \notin S^*$ ，则挑战者运行原始方案的 Extract 算法，为对手返回私钥 sk^* ；否则，挑战者将空字符 \perp 返回给对手。
 - 4) 挑战阶段：对手首先向挑战者提交两个长度相等的消息 M_0 和 M_1 。随后挑战者在 $\{SEM, EEM\}$ 中选择加密模式并对消息 M_ρ 进行加密处理，其中 ρ 是从 $\{0,1\}$ 中随机选取的值。最后，挑战者 \mathcal{C} 将挑战密文 CT 发送给对手。
 - 5) 学习阶段 II：类似于阶段 I，对手 \mathcal{A} 在该阶段进行更多的私钥查询，但不能查询 S^* 中关于 ID^* 的密钥。
 - 6) 猜测阶段：对手 \mathcal{A} 输出对 ρ 的猜测 ρ' ，如果 $\rho' = \rho$ ，则赢得游戏。
- 令 $Adv_{IND-CPA}^{\mathcal{A}}$ 表示对手 \mathcal{A} 在 IND-CPA 游戏中的优势。则可以将该优势通过以下等式表示：

$$Adv_{IND-CPA}^{\mathcal{A}} = \left| \Pr[\rho' = \rho] - \frac{1}{2} \right| \quad (3-12)$$

定义 3-2 DM-IBBE 方案是 IND-CPA 安全的，如果对于任何概率多项式时间敌手 \mathcal{A} ，其优势 $Adv_{IND-CPA}^{\mathcal{A}}$ 均是可以忽略的。

定理 3-1 若 DDH 假设成立，则本章提出的双模式基于身份的广播加密方案，包括“选择加密模式”和“排它加密模式”，在选择明文攻击下具有语义安全性。

证明：若存在一个概率多项式时间敌手 \mathcal{A} 能够以不可忽略的概率 ϵ 赢得 IND-CPA 游戏，那么可以构造一个挑战者 \mathcal{C} ，该挑战者可以以 $\epsilon/2$ 的优势解决 DDH 问题，具体证明过程如下。

挑战者 \mathcal{C} 首先抛掷一枚均匀硬币 $\mu \in \{0,1\}$ ，若 $\mu = 0$ ，则将 $\{g, g^a, g^b, g^{ab}\}$ 设为 DDH 实例；若 $\mu = 1$ ，则将 DDH 实例设置为 $\{g, g^a, g^b, Z\}$ ，其中 g 是群 \mathbb{G}_p 中的一个随机生成元， a 和 b 均为 \mathbb{Z}_p 中的随机数， Z 是 \mathbb{G}_p 中随机元素。

1) 初始阶段：在该阶段，敌手首先要公布一个要挑战的接收者集合 $S^* = \{ID_{s1}^*, ID_{s2}^*, \dots, ID_{sl}^*\} \subseteq U$ ，在这种情况下，被排除的接收者集合 R^* 可以表示为 $R^* = \{ID_{r1}^*, ID_{r2}^*, \dots, ID_{rk}^*\} = U \setminus S^*$ 。

2) 系统建立阶段：挑战者随机选取 $(m+1)$ 个整数 $\{a_i \in \mathbb{Z}_p\}_{i=0}^m$ ，并令 m 阶多项式为 $f(x) = \sum_{i=0}^m a_i x^i \bmod p$ 。令 $H: \{0,1\}^* \rightarrow \mathbb{Z}_p^*$ 为哈希函数，对于任意用户 $ID_i \in U$ 计算 $H(ID_i) = x_i$ ，其中 $ID_i \in \{0,1\}^*$ 表示用户身份。最后，挑战者将系统公钥 PK 发送给敌手 \mathcal{A} 。

$$PK = \langle g, H, \{g^{a_i}\}_{i \in [0, m]} \rangle \quad (3-13)$$

3) 学习阶段 I：在该阶段，敌手将进行关于用户 ID^* 的私钥查询。若 $ID^* \notin S^*$ ，则挑战者 \mathcal{C} 计算 $H(ID^*) = x^*$ 并为敌手 \mathcal{A} 生成私钥 $sk^* = f(x^*)$ ；否则 \mathcal{C} 返回上。

4) 挑战阶段：敌手 \mathcal{A} 首先选择两个等长的消息 M_0 和 M_1 并发送给挑战者。随后，挑战者在 $\{SEM, EEM\}$ 中随机选取一种加密模式并加密消息 M_ρ ，其中 ρ 是 $\{0,1\}$ 中的随机值。

对于选择加密模式 SEM，挑战者计算密文组件 $C_0 = M \cdot Z$ 和 $C_1 = g^a$ ，其中 $a = \alpha$ 。令 $h_s(x)$ 为 l 阶多项式，使得下列等式成立。

$$g^{h_s(x)} = \begin{cases} T_i, & ID_i \in S^* \\ g^b, & x = 0 \end{cases} \quad (3-14)$$

在上述等式中，意味着 $s = b$ ，通过调用拉格朗日插值公式，挑战者 \mathcal{C} 可

以重构 $g^{h_{S^*}(x)} = g^{b \cdot L_0(x)} \cdot \prod_{ID_i \in S^*} T_i^{L_i(x)}$ 。随后，挑战者在 \mathbb{Z}_p^* 中随机选取 $\{\tau_i\}_{i \in [1, l]}$ ，其中对于任意的 $i \in [1, l]$ 和 $j \in [1, m]$ ，满足 $\tau_i \neq H(ID_j)$ 。最后，挑战者计算如下密文组件。

$$\left\{ C_{i,1} = \tau_i, C_{i,2} = g^{a \cdot h_{S^*}(\tau_i)} \right\}_{i \in [1, l]} \quad (3-15)$$

若 $\mu = 0$ ，则有 $Z = g^{ab}$ ，此时 $C_0 = g^{ab} \cdot M$ ；若 $\mu = 1$ ，则 Z 为 \mathbb{G}_p 中的随机元素，此时 $C_0 = Z \cdot M$ 也可以看作是 \mathbb{G}_p 中的随机元素。

对于“排它加密模式”EEM，挑战者计算密文组件 $C_0 = M \cdot Z$ 和 $C_1 = g^a$ ，其中 $\alpha = a$ 。令 $h_R(x)$ 为 l 阶多项式，使得下列等式成立。

$$g^{h_R(x)} = \begin{cases} T_i, & ID_i \in U \\ g^b, & x = 0 \end{cases} \quad (3-16)$$

在上述等式中，意味着 $s = b$ ，通过调用拉格朗日插值公式，挑战者 \mathcal{C} 可以重构 $g^{h_R(x)} = g^{b \cdot L_0(x)} \cdot \prod_{ID_i \in U} T_i^{L_i(x)}$ 。对于 $ID_{ri}^* \in R^*$ ，挑战者计算 $H(ID_{ri}^*) \rightarrow \delta_i$ ，其中 $i \in [1, k]$ 。随挑战者随机选取 $\{\delta_i \in \mathbb{Z}_p^*\}_{i \in [k+1, m]}$ 使得对于任意的 $ID_i \in U$ ，等式 $\delta_i \neq H(ID_i)$ 成立。最后，挑战者计算如下密文组件。

$$\left\{ C_{i,1} = \delta_i, C_{i,2} = g^{a \cdot h_R(\tau_i)} \right\}_{i \in [1, m]} \quad (3-17)$$

若 $\mu = 0$ ，则有 $Z = g^{ab}$ ，此时 $C_0 = g^{ab} \cdot M$ ；若 $\mu = 1$ ，则 Z 为 \mathbb{G}_p 中的随机元素，此时 $C_0 = Z \cdot M$ 也可以看作是 \mathbb{G}_p 中的随机元素。

5) 学习阶段 II：与阶段 I 类似，敌手 \mathcal{A} 在该阶段可以进行更多的密钥查询，但它不能询问 $ID^* \in S^*$ 的密钥。

6) 猜测阶段：敌手 \mathcal{A} 给出对 ρ 的猜测 ρ' ，若 $\rho' = \rho$ ，则挑战者输出 $\mu' = 0$ ，在这种情况下，挑战者给出一个正确的 DDH 实例；否则，挑战者输出 $\mu' = 1$ ，此时，挑战者给出一个随机 4 元组。

若 $\mu = 0$ ，则敌手 \mathcal{A} 将获得正确的密文，在这种情况下，由前述定义可知，敌手正确猜中加密消息的优势为 ϵ ，因此有 $\Pr[\rho' = \rho | \mu = 0] = 1/2 + \epsilon$ 。当 $\rho' = \rho$ 时，挑战者给出对 μ 的猜测为 $\mu' = 0$ ，因此有 $\Pr[\mu' = \mu | \mu = 0] = 1/2 + \epsilon$ 。

若 $\mu = 1$ ，则在消息 M_0 和 M_1 下生成的密文分布形式都是相同的，因此有等式 $\Pr[\rho' \neq \rho | \mu = 1] = 1/2$ 。当 $\rho' \neq \rho$ 时，挑战者给出对 μ 的猜测为 $\mu' = 1$ ，因此有 $\Pr[\mu' = \mu | \mu = 1] = 1/2$ 。

综上所述，挑战者在解决 DDH 问题中的整体优势可表示如下：

$$\begin{aligned} & \frac{1}{2} \Pr[\mu' = \mu | \mu = 0] + \frac{1}{2} \Pr[\mu' = \mu | \mu = 1] - \frac{1}{2} \\ &= \frac{1}{2} \times \left(\frac{1}{2} + \epsilon \right) + \frac{1}{2} \times \frac{1}{2} - \frac{1}{2} = \frac{\epsilon}{2} \end{aligned} \quad (3-18)$$

事实上，对于任意的概率多项式敌手，DDH 问题在群 \mathbb{G}_p 上都是困难的，因此上述假设不成立，即不存在这样一个对手，能够以不可忽略的概率攻破本章所提出方案的 IND-CPA 安全性。

3.4.2 SEM 中选择明文攻击下的匿名性

本文将对上述讨论中的选择加密模式，通过下列游戏定义选择明文攻击下的匿名性。

- 1) 初始阶段：敌手 \mathcal{A} 首先宣布两个接收者集合 S_0 和 S_1 。
- 2) 系统建立阶段：首先，挑战者 \mathcal{C} 执行原始方案中的 Setup 算法，获得系统公钥 PK 和主密钥 MSK 。随后，挑战者将系统公钥 PK 发送给敌手 \mathcal{A} ，并秘密保存主密钥 MSK 。
- 3) 学习阶段 I：敌手对用户 ID^* 进行私钥查询，若 $ID^* \notin (S_0 \cup S_1) \setminus (S_0 \cap S_1)$ ，则挑战者运行原始方案的 Extract 算法，为敌手返回私钥 sk^* ；否则，挑战者将空字符 λ 返回给敌手。
- 4) 挑战阶段：敌手首先向挑战者提交要挑战的消息 M 。随后挑战者随机选取 $\eta \in \{0,1\}$ 。最后，挑战者 \mathcal{C} 将在 S_η 下生成的挑战密文 CT 发送给敌手。
- 5) 学习阶段 II：类似于阶段 I，敌手 \mathcal{A} 在该阶段进行更多的私钥查询。
- 6) 猜测阶段：敌手 \mathcal{A} 输出对 η 的猜测 η' ，如果 $\eta' = \eta$ ，则赢得游戏。

令 $Adv_{\text{ANO-CPA}}^{\mathcal{A}}$ 表示敌手 \mathcal{A} 在 ANO-CPA 游戏中的优势。可以将该优势通过以下等式表示。

$$Adv_{\text{ANO-CPA}}^{\mathcal{A}} = \left| \Pr[\eta' = \eta] - \frac{1}{2} \right| \quad (3-19)$$

定义 3-3 DM-IBBE 方案中的选择加密模式是 ANO-CPA 安全的，若对于任何概率多项式时间敌手 \mathcal{A} ，其优势 $Adv_{\text{ANO-CPA}}^{\mathcal{A}}$ 均是可以忽略的。

定理 3-2 若 DDH 假设成立，则本章构造的 DM-IBBE 方案中的“选择加密模式”在选择明文攻击下具有匿名性。

证明：若存在一个概率多项式时间敌手 \mathcal{A} 能够以不可忽略的概率 ϵ 赢得

ANO-CPA 游戏，那么可以构造出一个挑战者 \mathcal{C} ，它可以以 $\epsilon/2$ 概率的优势解决 DDH 问题。

挑战者 \mathcal{C} 首先抛掷一枚质地均匀的硬币 $\mu \in \{0,1\}$ ，如果 $\mu=0$ ，则将四元组 $\{g, g^a, g^b, g^{ab}\}$ 设为 DDH 实例；如果 $\mu=1$ ，则将 $\{g, g^a, g^b, Z\}$ 设置为 DDH 实例，其中 g 是群 \mathbb{G}_p 中的一个随机生成元， a 和 b 是 \mathbb{Z}_p 中的随机数， Z 是 \mathbb{G}_p 中的随机元素。

1) 初始化阶段：在该阶段，敌手首先宣布两个要挑战的接收者集合 $S_0 = \{ID_{01}, ID_{02}, \dots, ID_{0l}\} \subset U$ 和 $S_1 = \{ID_{11}, ID_{12}, \dots, ID_{1l}\} \subset U$ 。

2) 系统建立阶段：挑战者随机选取 $(m+1)$ 个整数 $\{a_i \in \mathbb{Z}_p\}_{i=0}^m$ ，并令 m 阶多项式为 $f(x) = \sum_{i=0}^m a_i x^i \bmod p$ 。令 $H: \{0,1\}^* \rightarrow \mathbb{Z}_p^*$ 为哈希函数，对于任意用户 $ID_i \in U$ 计算 $H(ID_i) = x_i$ ，其中 $ID_i \in \{0,1\}^*$ 表示用户身份。最后，挑战者将系统公钥 PK 发送给敌手 \mathcal{A} 。

$$PK = \langle g, H, \{g^{a_i}\}_{i \in [0,m]} \rangle \quad (3-20)$$

3) 学习阶段 I：敌手将进行关于用户 ID^* 的私钥查询，若 $ID^* \notin (S_0 \cup S_1) \setminus (S_0 \cap S_1)$ ，则挑战者 \mathcal{C} 返回 \perp ；否则挑战者 \mathcal{C} 计算 $H(ID^*) = x^*$ 并将私钥 $sk^* = f(x^*)$ 发送给敌手 \mathcal{A} 。

4) 挑战阶段：敌手 \mathcal{A} 将消息 M 发送给挑战者，随后挑战者 \mathcal{C} 随机选择 $\eta \in \{0,1\}$ ，采用选择加密模式 SEM 在 S_η 下对消息 M 进行加密。挑战者计算密文组件 $C_0 = M \cdot Z$ 和 $C_1 = g^a$ ，其中 $\alpha = a$ 。令 $h_{S_\eta}(x)$ 为 l 阶多项式，使得下列等式成立。

$$g^{h_{S_\eta}(x)} = \begin{cases} T_i, & ID_i \in S_\eta \\ g^b, & x = 0 \end{cases} \quad (3-21)$$

上式意味着 $s = b$ ，通过调用拉格朗日插值公式，挑战者 \mathcal{C} 可以重构出多项式 $g^{h_{S_\eta}(x)} = g^{b \cdot L_0(x)} \cdot \prod_{ID_i \in S_\eta} T_i^{L_i(x)}$ 。随后，挑战者在 \mathbb{Z}_p^* 中随机选取 $\{\tau_i\}_{i \in [1,l]}$ ，对于任意的 $i \in [1,l]$ 以及 $j \in [1,m]$ ，使得 $\tau_i \neq H(ID_j)$ 成立。最后，挑战者计算密文组件如下所示：

$$\left\{ C_{i,1} = \tau_i, C_{i,2} = g^{a \cdot h_{S_\eta}(\tau_i)} \right\}_{i \in [1,l]} \quad (3-22)$$

若 $\mu=0$ ，则 $Z = g^{ab}$ ，此时 $C_0 = g^{ab} \cdot M$ ；若 $\mu=1$ ，则 Z 为 \mathbb{G}_p 中的随机元素，此时 $C_0 = Z \cdot M$ 也可以看作是 \mathbb{G}_p 中的随机元素。

- 5) 学习阶段 II: 与阶段 I 类似, 敌手 \mathcal{A} 进行更多的密钥查询。
- 6) 猜测阶段: 敌手 \mathcal{A} 给出对 η 的猜测 η' , 若 $\eta' = \eta$, 则挑战者输出 $\mu' = 0$, 在这种情况下, 挑战者给出一个正确的 DDH 实例; 否则, 挑战者输出 $\mu' = 1$, 在这种情况下, 挑战者给出一个随机 4 元组。

若 $\mu = 0$, 则敌手 \mathcal{A} 将获得正确的密文, 在这种情况下, 由前述定义可知, 敌手正确猜中加密消息的优势为 ϵ , 因此有 $\Pr[\eta' = \eta | \mu = 0] = 1/2 + \epsilon$ 。当 $\eta' = \eta$ 时, 挑战者给出对 μ 的猜测为 $\mu' = 0$, 因此有 $\Pr[\mu' = \mu | \mu = 0] = 1/2 + \epsilon$ 。

若 $\mu = 1$, 则在消息 M_0 和 M_1 下生成的密文分布形式都是相同的, 因此有等式 $\Pr[\eta' \neq \eta | \mu = 1] = 1/2$ 。当 $\eta' \neq \eta$ 时, 挑战者给出对 μ 的猜测为 $\mu' = 1$, 因此有 $\Pr[\mu' = \mu | \mu = 1] = 1/2$ 。

综上所述, 挑战者在解决 DDH 问题中的整体优势可表示如下:

$$\begin{aligned} & \frac{1}{2} \Pr[\mu' = \mu | \mu = 0] + \frac{1}{2} \Pr[\mu' = \mu | \mu = 1] - \frac{1}{2} \\ &= \frac{1}{2} \times \left(\frac{1}{2} + \epsilon \right) + \frac{1}{2} \times \frac{1}{2} - \frac{1}{2} = \frac{\epsilon}{2} \end{aligned} \quad (3-23)$$

事实上, 对于任意的概率多项式敌手来说, DDH 问题在群 \mathbb{G}_p 上都是困难的, 因此上述假设不成立, 即不存在这样一个对手, 能够以不可忽略的概率攻破本章所提出的 DM-IBBE 方案中选择加密模式下的 ANO-CPA 安全性。

3.5 性能分析

本节将对提出的双模式身份基广播加密方案进行性能评估, 首先对现有四个相关广播加密与本章提出 DM-IBBE 的功能特征进行对比, 随后对相关方案之间的计算和存储开销进行理论对比分析, 并对它们的私钥提取、加密和解密算法进行了仿真对比, 下面给出具体分析。

表 3-2 四个现有的广播加密方案与本章提出方案之间对比结果

方案	双模式加密	匿名性	无配对运算	困难假设	安全模型
文献[49]	✓	✗	✗	DBDHE	标准模型
文献[105]	✗	✗	✗	DBDHE BDHI	标准模型
文献[106]	✗	✓	✗	BDH	随机预言机 模型
文献[107]	✗	✓	✗	DDH	标准模型
本章方案	✓	✓	✓	DDH	标准模型

如表 3-2 所示，本节首先对现有四个广播加密方案与本章提出的方案在双模式加密、接收者匿名性、安全模型、困难假设以及双线性配对运算等方面进行比较。

从对比结果中不难发现，文献[49]和本章提出的广播加密方案能够实现两种不同的加密模式，从而实现多样化的隐私需求。此外，为了避免泄露密文中接收者的身份信息，文献[106]与本章方法考虑了广播接收者的隐私保护问题。进一步地，除了本章提出的方案外，该表中所有其它的广播加密方案在实施过程中都需要执行多次计算开销非常昂贵的配对运算。因此，与其它方案相比，本方案的计算复杂性较低。文献[105]中方案的安全性被规约到了判定性双线性 Diffie-Hellman 指数假设（Decisional Bilinear Diffie-Hellman Exponent, DBDHE）以及判定性 Diffie-Hellman 逆假设（Decisional Diffie-Hellman Inversion, DDHI），文献[106]的安全性在随机预言机模型下被规约到判定性 Diffie-Hellman 假设（Decisional Diffie-Hellman, DDH）。文献[19]中的广播加密方案在判定性双线性 Diffie-Hellman 假设（Decisional Bilinear Diffie-Hellman, DBDH）下被证明是安全的，文献[107]安全性被规约到 DDH 假设。

在计算和存储开销的理论对比上，令 m 表示系统中用户总数， l 和 k 分别表示指定和排除的广播接收者个数。 $E(\mathbb{G})$ ， $M(\mathbb{G})$ 和 $D(\mathbb{G})$ 分别表示群 \mathbb{G} 中的指数、乘法和除法运算， B 表示一次双线性配对运算。此外，令 $l(\mathbb{G})$ ， $l(\mathbb{G}_r)$ 和 $l(\mathbb{Z})$ 分别表示群 \mathbb{G} ， \mathbb{G}_r 以及 \mathbb{Z} 中一个元素的长度。由于群 \mathbb{Z} 中的运算以及哈希运算耗时较少，因此在理论对比中不予考虑。

表 3-3 相关广播加密方案计算开销对比

方案	加密阶段		解密阶段	
	选择加密模式	排除加密模式	选择加密模式	排除加密模式
文献 [49]	$2E(\mathbb{G}) + lM(\mathbb{G})$ $+E(\mathbb{G}_r) + B$	$2E(\mathbb{G}) + (k-l)M(\mathbb{G})$ $+E(\mathbb{G}_r) + D(\mathbb{G}) + B$	$(l-1)M(\mathbb{G})$ $+D(\mathbb{G}_r) + 2B$	$D(\mathbb{G}) + kM(\mathbb{G})$ $+D(\mathbb{G}_r) + 2B$
文献 [105]	$(2l+2)E(\mathbb{G}) + 2lB$ $+(3l+1)E(\mathbb{G}_r)$	—	$(l+2)E(\mathbb{G})$ $+3E(\mathbb{G}_r) + B$	—
文献 [106]	$(l-1)M(\mathbb{G}) + 2lE(\mathbb{G}_r)$ $+(2l+2)E(\mathbb{G}) + 2lB$	—	$(2l-1)E(\mathbb{G}) + 2D(\mathbb{G})$ $+(2l-2)M(\mathbb{G}) + 2B$	—
文献 [107]	$(l+5)E(\mathbb{G}) + E(\mathbb{G}_r)$ $+(l+2)M(\mathbb{G})$	—	$2lE(\mathbb{G}) + 2D(\mathbb{G}_r) + 3B$ $+(2l-2)M(\mathbb{G}) + M(\mathbb{G}_r)$	—
本章 方案	$(lm + 2l + 2)E(\mathbb{G})$ $+(lm + l)M(\mathbb{G})$	$(m^2 + 2m + 2)E(\mathbb{G})$ $+(m^2 + l)M(\mathbb{G})$	$(l+2)E(\mathbb{G})$ $+lM(\mathbb{G}) + D(\mathbb{G})$	$(m+2)E(\mathbb{G}) + D(\mathbb{G})$ $+lM(\mathbb{G}_r) + mM(\mathbb{G})$

表 3-3 展示了相关广播加密方案之间计算开销的理论对比。从计算开销对比结果可以看出，文献[49][105][106][107]中加密算法的计算消耗与广播集

合中被指定的接收者的数目线性相关，然而本章方案中的选择的加密模式下计算开销与指定接收者和系统中所有用户数目相关。在排它加密模式下，文献[49]中加密消耗与被排除接收者数目呈线性相关，而在本章方案中计算开销与系统中所有用户数目成平方关系。在解密阶段，所有方案的选择加密模式下计算开销均与指定接收者的数目线性相关，文献[49]中排它模式下解密消耗与被排除的接收者数目线性相关，而本章方案则与系统中所有用户的数目线性相关。此外，无论是加密还是解密阶段，本章方案均未涉及配对运算。

表 3-4 是相关广播加密之间公钥和密文存储开销的理论对比，从表中可以看出文献[106]的公钥尺寸是一个常量，其它方案中的公钥存储开销与系统中所有用户的数目线性相关。在文献[49]和[107]中，它们的密文长度也是定值，然而文献[105][106]以及本章方案中选择加密模式下的密文存储开销与指定广播接收者的数目相关，本章方案排它加密模式下密文存储开销则与系统中所有用户数目相关。

表 3-4 相关广播加密方案存储开销对比

方案	公钥	密文	
		选择加密模式	排除加密模式
文献[49]	$(3m+1) \cdot l(\mathbb{G})$	$2 \cdot l(\mathbb{G})$	$2 \cdot l(\mathbb{G})$
文献[105]	$(3m+4) \cdot l(\mathbb{G}) + l(\mathbb{G}_r)$	$(l+3) \cdot l(\mathbb{G}) + 2 \cdot l(\mathbb{Z})$	—
文献[106]	$2 \cdot l(\mathbb{G})$	$(2l+3) \cdot l(\mathbb{G})$	—
文献[107]	$(3m+14) \cdot l(\mathbb{G}) + l(\mathbb{G}_r)$	$4 \cdot l(\mathbb{G})$	—
本章方案	$(m+2) \cdot l(\mathbb{G})$	$(l+2) \cdot l(\mathbb{G}) + l(\mathbb{Z})$	$(m+2) \cdot l(\mathbb{G}) + m \cdot l(\mathbb{Z})$

为了直观地对比本章方案与文献[49][105][106][107]之间性能的差别，接下来基于 JPBC 对上述文方案进行仿真。本章实验运行在配置为 Intel(R) Core(TM) i5-4590S CPU@3.00GHz, 8G ROM 的 64 位 Windows 10 PC 上，实验选取 JPBC 库中阶数为 160 位的 Type A 配对 (Pairing)，该配对是基于域 \mathbb{F}_q 上椭圆曲线 $y^2 = x^3 + x$ 构造的，其中 q 为素数且满足 $q \equiv 3 \pmod{4}$ 且阶数 r 是 $q+1$ 的一个素因子。此外，设定最大用户总数 $m=100$ ，实验被单独重复 100 次，取算术平均值作为最后的结果。考虑到 Setup 算法仅在系统参数生成时被执行一次，与方案中的其它算法相比时间开销很小，因此仅对上述方案中的 Extraction, Encryption 和 Decryption 三个算法的计算开销进行对比分析。

从图 3-9 中可以明显看出，本章方案为用户生成私钥所需的时间几乎少于其它相关方案。当用户数大于 70 时，本方案生成用户私钥所消耗时间大于文献[106]中的方案。然而，与另外三个方案相比，本方案中密钥提取算法仍

然具有明显的优势。从图 3-10 中可知，当系统用户总数一定的情况下，加密算法的时间开销随着黑名单用户数目的增加而减小，然而在两种加密模式下的运行时间比其它方案要长，原因在于本方案选择加密和排它加密模式下的加密算法将分别重构多项式 $g^{hs(x)}$ 和 $g^{h_k(x)}$ ，而重构上述两个多项式需要运算多次群 G 中的指数运算。另一方面，图 3-11 结果表明本章方案中解密算法是高效的，特别是选择加密模式的解密时间比文献[105][106][107]中的都要少。

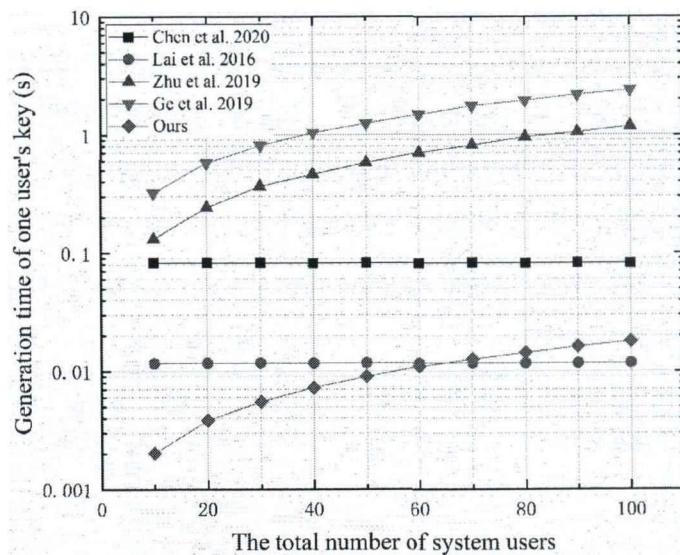


图 3-9 Extraction 算法时间消耗对比

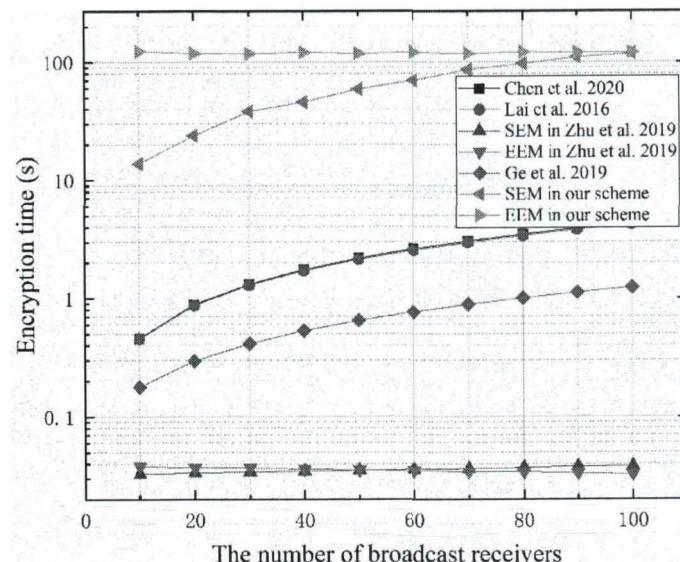


图 3-10 Encryption 算法时间消耗对比

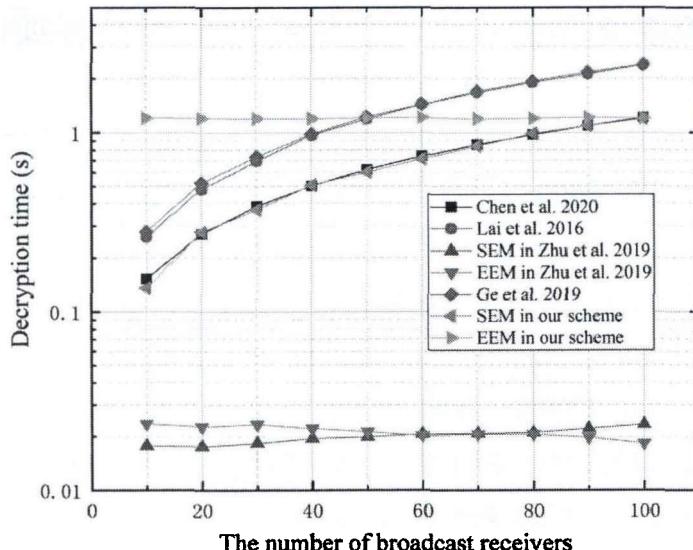


图 3-11 Decryption 算法时间消耗对比

3.6 本章小结

本章针对智能合约敏感数据访问限定到指定用户和非冲突用户的两种隐私需求，提出了一种面向智能合约交易隐私的 DM-IBBE 方案，通过在拉格朗日插值曲线下对指定集和冲突集上不同插值点与重构曲线上点的选取，设计支持“选择”和“排它”两种加密模式的 DM-IBBE 分别满足上述隐私需求。此外，基于 SPESC 语言给出一种具有隐私保护功能的智能合约架构，保证敏感数据以合约条款形式声明隐私并由编译器将预定义的 DM-IBBE 算法链接到智能合约程序中实施保护。在 DDH 假设下证明了该方案的语义安全性以及选择加密模式下接收者匿名性，与其它广播加密方案相比，DM-IBBE 具有更加丰富的加密模式以满足智能合约不同的隐私需求。

4 支持可编程密文和去中心化密钥管理的属性基加密

针对区块链去中心化特征与属性基加密中心化密钥管理相冲突的问题，本章提出了一种面向属性基加密的去中心化密钥管理与脚本化密文机制。通过融合两类加法同态的安全多方计算技术，设计了一种基于密文策略和去中心化密钥的属性基加密方案（Ciphertext-policy Decentralized-key Attribute-based Encryption, CP-DK-ABE）实现区块链节点对主密钥的共同管理和用户私钥的协同生成。此外，通过解密过程中的密钥查询、密文逻辑等操作扩展区块链脚本指令系统，在加密过程中实现针对密文策略中复杂逻辑的密文脚本化以及脚本解释器对脚本化密文的自动化解密。

4.1 研究动机

针对区块链分布式广播网络特征，群组加密为区块链交易数据安全共享提供了一种更加有效的解决方案。作为一种“一对多”的加密形式，在群组加密方案中，授权者不再是单个用户而是由多个用户组成的一个集合，发送者首先选择授权者集合对消息进行加密并对密文进行广播，在授权集中的任意一方均可通过自己的私钥对密文进行解密操作。此外，对于新加入系统的用户，若其属于授权集，则在获得相关私钥后，仍然可对原有密文进行解密，而无需进行额外处理。

按照授权集合的选择方式不同，群组加密大致可以分为以下几种：基于标识的广播加密（Identity-based Broadcast Encryption, IBBE）、基于角色加密（Role-based Encryption, RBE）和基于属性加密（Attribute-based Encryption, ABE）。它们分别以“标识”、“角色”和“属性”作为最小单位来指定授权集合。特别地，在 CP-ABE 中数据拥有者被允许在由接收者的属性构成的策略下加密消息，例如个人健康记录的访问策略可以定义为“((Hospital = Rehabilitation AND Role = Clinician) OR Institution = Insurance company)”。这意味着除康复医院的临床医生外，保险公司的员工也允许查看健康记录。与 IBBE 和 RBE 相比，CP-ABE 中的授权用户描述方式更加细粒度和灵活。它适用于区块链中敏感数据的保护，具有更灵活的授权方式。

早期关于属性基加密的研究多集中在中心化环境中，即用户的私钥是由唯一可信中心颁发的。这不能满足分布式网络结构中的密钥去中心化管理的需求。为了适应多中心化环境，Chase^[59]提出第一个多权威中心属性基加密方案，该方案允许任意多项式个独立权威中心管理属性和分发密钥。然而，为

了抵抗用户属性共谋，该方案需要一个完全可信的中心参与用户私钥生成。随后，Lewko 和 Waters^[60]提出一个新的去中心化属性基加密（Decentralizing ABE, DABE）。在他们的方案中，可信中心节点被移除，每个权威中心管理一部分属性并独立地向用户颁发属性密钥，但系统初始化阶段需要所有属性权权威中心作完成。为了改进这一缺陷，Li 等人^[108]提出一个新的去中心 CP-ABE 方案，在该方案中每个属性权威中心独立工作且不与其他属性中心进行任何交互，但它仅适用于域管理，且在执行过程中需要一个混淆程序。在这些去中心化 ABE 方案中，中心权威被删除，每个权威独立地向用户发出属性子密钥。这些方案虽然可以实现用户私钥的去中心化生成，但容错性低。这意味着，如果一个属性中心被破坏，整个系统将无法运行。

尽管现有去中心化属性基加密方案能够实现多个中心参与用户私钥的生成，但是鲜有对符合区块链分布式结构的去中心化管理和分布式密钥生成的研究。此外，现有方案中多以数据形式存储密文，这给具有复杂访问策略的密文解密操作带来了不便，例如在密文解密过程中，用户首先搜索属性子密文对应的所有属性子密钥，然后在属性之间进行逻辑匹配操作，直到解密出消息。此外，现有 CP-ABE 方案中的加解密过程由客户端执行，这些基于客户端的操作增加了区块链设备程序设计的难度。因此，本章希望构造的能够进行区块链数据安全共享的 CP-ABE 方案需要满足以下要求。

- 1) 分布式密钥分发：CP-ABE 公钥是公开透明的，但需要对主密钥进行分布式管理，而不是由可信管理员单独地管理；
- 2) 去中心化密钥生成：用户的私钥需要由多个节点协同生成，私钥生成过程不涉及中心节点；
- 3) 可编程密文：密文的表示形式不仅紧凑，而且具有可编程性。此外，加密和解密过程可在区块链节点中通过程序代码执行。

4.2 区块链数据安全共享系统

区块链物联网（Blockchain Internet of Things, BIoT）对数据安全共享具有巨大的需求，这是由于链接到区块链网络中的设备计算和存储能力有限，需要通过设备之间的数据共享实现对数据的进一步处理和分析。然而，区块链物联网技术的大规模应用造成了一系列安全威胁，特别是数据隐私泄露问题。根据 2020 年 Palo Alto 的物联网威胁报告⁶，98% 的物联网设备未进行加密处理，这些暴露在网络上的数据很容易被攻击者监视和收集，进而在暗网上牟

⁶ <https://start.paloaltonetworks.com/unit-42-iot-threat-report>

利。尽管一些工作已经研究了物联网中的用户认证和隐私保护问题，但敏感数据的安全共享仍是需要被特别关注。接下来，将展示区块链隐私数据安全分享系统以实现敏感数据在指定群组之间安全共享。

4.2.1 系统框架

如图 4-1 所示，在区块链数据安全共享系统中，区块链包含两个实体：区块链网络和公共账本。接下来，将详细描述上述两个实体。

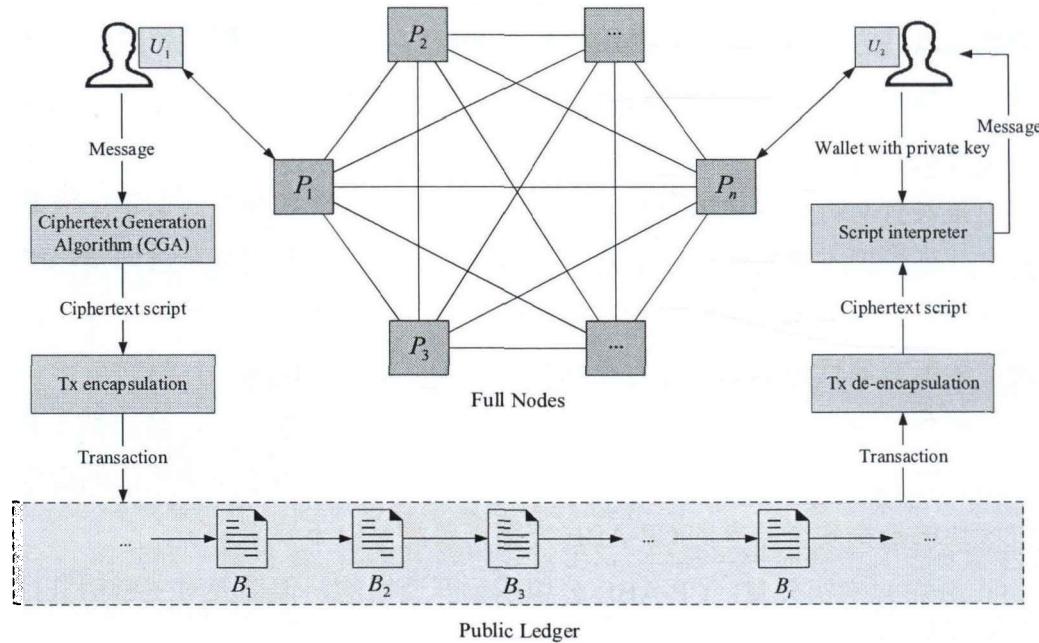


图 4-1 区块链隐私数据安全分享系统

区块链网络：在本节中，通过提出一种新的加密机制来保护区块链中交易的敏感内容，并通过授权控制对特定内容的接收者进行灵活选择。为此，本文将在一个联盟链系统（比如 Hyperledger）上设计新的方案。在该方案中，区块链包含两种主要类型的节点：完全节点和轻节点。

- 1) 完全节点 P_i 具有较强的运算能力来管理密钥，但并非完全可信的。此外，完全节点可用来存储区块链的历史副本，参与共识验证来保障系统数据的安全性和正确性，其中完全节点集 $P = \{P_i\}_{i \in [1, n]}$ ， n 为完全节点的个数；
- 2) 轻节点 U_k 表示用户，其数目不受限制 ($k \in [1, +\infty]$)，他们需要连接到完全节点上，以便与网络的当前状态同步，并能够提交和获取交易。此外，在该系统中，用户拥有钱包用来存储私钥和其他资产，用户既可以是数据拥有者也可以是数据消费者；

① 数据拥有者是希望将数据秘密地分享给特定用户的实体；

② 数据消费者是希望获取区块链账本上的分享数据的实体。

公共账本：在区块链系统中，公共账本是用来存储交易的实体。公共账本是由一系列区块 B_i 按照时间顺序组成的一个链状结构，如图 4-2 所示，这里将其表示为 $B = (B_1, \dots, B_i, \dots)$ ，其中对于任意时间 $t_i < t_j$ ，则有 $B_i \prec B_j$ 偏序关系存在。每个区块 B_i 包含一系列交易 $B_i = \{T_{i,j}\}_{j \in [1, m]}$ ，在每个交易中包含交易数据和一些其它参数（例如前一交易 ID，时间戳等），这些信息以“键-值”对的形式存储。特别地，该系统中的交易数据为密文，用脚本形式进行表示。

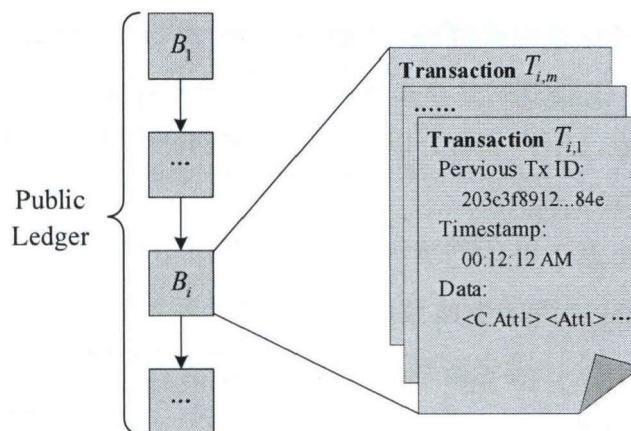


图 4-2 公共账本结构

在区块链数据安全分享系统中，用户以区块链为媒介对敏感数据（例如拍卖标的）在特定接收者之间进行秘密分享，分享过程采用公钥加密机制。系统中维护一个统一公钥 PK ，每个用户拥有自己的私钥，数据拥有者使用 PK 对隐私数据进行加密，数据消费者利用自身私钥对密文进行解密。为了减轻用户负担，加解密过程由区块链节点完成，称之为节点加解密。本工作难点不在于加解密过程，而在于密钥分发过程。传统意义上来说，用户私钥是由可信中心节点产生，但在区块链去中心化环境下，需要生成用户私钥的主密钥在所有完全节点 P_i 之间分散管理。因此，用户私钥是由用户与全部完全节点通过交互的形式产生，从而实现用户私钥生成的去中心化。

按照上述思想，基于区块链的隐私数据分享系统具体工作流程描述如下。

- 1) 完全节点私钥生成：首先，在区块链系统建立过程中，系统管理员生成系统公钥 PK 和主私钥 MSK ，并将 MSK 在完全节点集 P 中进行共享，作为其私钥 SK_i 。管理员将 PK 进行全网广播后永久退出系统；
- 2) 轻节点私钥生成：拥有属性集 L 的用户 U_k 与 P 中所有节点之间交互生成用户私钥 sk_k ，将私钥以及其他资产存储在用户的钱包中；

- 3) 数据加密: 数据拥有者在访问策略 T 下对消息 M 进行加密, 随后将密文以脚本形式写入交易 $T_{i,j}$ 中, 并将交易提交至公共账本;
- 4) 密文解密: 数据消费者 U_k 从公共账本中获取交易 $T_{i,j}$, 若该用户的属性集合 L 满足访问策略 T , 即 $L \models T$, 则可通过脚本系统对 $T_{i,j}$ 进行解密并从中恢复消息 M 。

4.2.2 区块链数据安全共享的合约化描述

按照上述对区块链数据安全共享系统的介绍, 接下来通过 SPESC 语言对数据分享过程进行合约化描述。如图 4-3 所示, 区块链数据安全共享合约包含三类当事人, 生成数据分享系统密钥的系统管理员 (administrator), 共享敏感数据的轻节点 (用户 user) 以及生成私钥的完全节点群体 (fullnodes), 他们可以分别执行以下操作。

系统管理员属于个体当事人, 可以执行以下操作:

- 1) GeneratePK 用于生成系统的公钥包括系统参数和完全节点的公钥;
- 2) GenerateSK_i 表示管理员为所有完全节点生成私钥, 包括系统主密钥片段以及各个完全节点私钥。

私钥请求用户属于群体当事人, 他们可以向完全节点请求自己的私钥, 执行以下操作:

- 1) IDparameter 用于生成与用户 ID 相关的参数;
- 2) ReconstructSK 用于用户从私钥片段中重构出自己的密钥;
- 3) Encrypt 用于数据分享用户在访问策略下对敏感数据进行加密计算;
- 4) Decrypt 用于数据接收用户通过自身私钥对密文进行解密计算。

完全节点是管理和生成用户私钥的群体当事人。他们可以执行以下操作:

- 1) SecretSharing 用于完全节点之间进行秘密片段的共享;
- 2) GeneratePP_i 用于生成完全节点的秘密参数;
- 3) GenerateSK_i 用于为用户 U_i 生成的私钥片段。

为了支持用当事人的的上述操作, 该合约中引入了以下四个附加信息:

- 1) Policy 表示加密算法输入的访问策略数, 由属性与逻辑节点组成;
- 2) Message 表示要分享的敏感数据;
- 3) Secret 代表完全节点的秘密值;
- 4) Attribute 表示用户属性。

```

1. contract DataSecuritySharing{
2.     party administrator{
3.         GeneratePK()
4.         GenerateSK_i()
5.     }
6.
7.     party group fullnodes{
8.         SecretSharing(Secret : Data)
9.         GeneratePP_i()
10.        GenerateD_i()
11.    }
12. }

23. term term1 : administrator can GeneratePK and GenerteSK_i,
24.     when before user did StartPrivateKeyRequester
25.     while send SK_i to fullnodes.           //完全节点私钥生成
26. term term2 : user can IDparameter,
27.     when after administrator did GeneratePK and GenerteSK_i
28.     while send attribute to fullnodes.      //轻节点身份参数生成
29. term term3 : fullnode must SecretSharing and GeneratePP_i(),
30.     when after user did IDparameter
31.     while sum the secret S_i.              //完全节点之间秘密共享
32. term term4 : fullnode can GenerateD_i,
33.     when after all fullnodes sumed the secret      //完全节点生成私钥片段
34. term term5 : user can ReconstructSK,
35.     when after fullnodes did GenerateD_i.        //轻节点生成私钥重构
36. term term6 : user can Encrypt,
37.     when after administrator did GeneratePK.    //节点加密
38. term term7 : user can Decrypt,
39.     when after user did ReconstructSK and attributes satisfy Policy. //节点解密
40. }

```

图 4-3 SPESC 编写的区块链数据安全分享智能合约

在图 4-3 中给出的合约中，条款 1 描述的是系统管理员为完全节点生成私钥，它是指轻节点在开始向完全节点请求个人私钥之前，系统管理员为所有完全节点生成各自的私钥以及系统主密钥的共享片段。条款 2 对应于轻节点私钥请求前对身份信息参数的生成，轻节点在管理员为完全节点生成私钥之后计算身份信息的承诺，并将该承诺以及属性集作为身份参数返回给完全节点。条款 3 和条款 4 对应于用户私钥请求，完全节点向用户返回私钥片段。条款 5 对应于轻节点的私钥重构，轻节点在获取足够多的密钥片段之后，从这些片段中恢复出密钥。条款 6 对应于数据加密，轻节点在管理员生成系统公钥之后，在选定的访问策略下对消息进行加密处理。条款 7 对应于数据解密，轻节点通过自身私钥对密文进行解密操作。

4.3 去中心化密钥生成密文策略属性基加密方案

通过前面对区块链数据安全共享系统的介绍以及对 SPESC 下设计的智能合约的描述，本节将在 Bethencourt, Sahai 和 Waters^[109]提出的密文策略属性基加密方案的基础上（简称 BSW 方案），构造一个新的去中心化密钥生成的 CP-ABE 方案来实现区块链隐私数据共享。BSW 方案被认为是最经典以及研究最广的属性基加密方案之一，与其相似，本章构造的方案同样包括 Setup, Key-generation, Encryption 和 Decryption 四个算法。特别地，本章方案基于门限秘密共享和多方安全计算重新设计了 Setup 和 Key-generation 两个算法，而 Encryption 和 Decryption 算法与 BSW 方案保持一致。

4.3.1 去中心化密钥生成构造方法

本文密钥生成算法的核心是系统私钥在完全节点之间共享并通过完全节点之间的多方安全计算为用户生成私钥。在给出具体算法之前，本节首先对用到的核心技术及构造思想进行介绍。

定义 4-1（安全多方计算） n 个参与方 P_1, P_2, \dots, P_n 共同完成一个计算任务，即

$$f(x_1, x_2, \dots, x_n) = (y_1, y_2, \dots, y_n) \quad (4-1)$$

在上述计算任务中，每个参与方 P_i 只能得到自己的输入 x_i 和自己的输出 y_i 并满足以下两个性质：

- 1) 终止性：计算任务能够在规定时间内结束；
- 2) 一致性：忠诚参与方在计算任务终止后获得相同结果 $y_1 = y_2 = \dots = y_n$ 。

在所有完全节点为用户计算私钥的过程中，需要将节点 P_i 的秘密值 θ_i 在所有节点之间共享，并由完全节点通过共享片段独自重构出所有秘密值片段的总和。这个共享过程是通过多项式实现的，满足多项式加法同态，即令 $[a]_i = f(x_i) = a + r_1 x_i + \dots + r_{n-1} x_i^{n-1}$ 为秘密值 a 通过多项式 $f(x)$ 的共享片段， $[b]_i = g(x_i) = b + t_1 x_i + \dots + t_{n-1} x_i^{n-1}$ 为秘密值 b 通过多项式 $g(x)$ 的共享片段，则多项式加法同态满足

$$[a]_i + [b]_i = f(x_i) + g(x_i) = (a + b) + \lambda_1 x_i + \dots + \lambda_{n-1} x_i^{n-1} = (f + g)(x_i) \quad (4-2)$$

在完全节点完成随机多项式秘密重构之后，完全节点 P_i 与用户 ID_k 通过指数同态加法计算用户的私钥片段，其中指数同态加法运算满足

$$h^{[a]} \otimes h^{[b]} = h^{f(x_i)} \otimes h^{g(x_i)} = h^{f(x_i)+g(x_i)} = h^{(f+g)(x_i)} = h^{[a+b]} \quad (4-3)$$

通过上面介绍的密码技术，本章提出的密钥生成算法的流程可描述如下：

- 1) **用户参数信息共享**: 用户 ID_k 将属性集 \mathbb{A} 以及私钥共享片段共享给所有完全节点 P_i ;
- 2) **随机多项式共享**: 完全节点之间通过随机多项式将各自的秘密值 θ_i 在所有完全节点之间共享;
- 3) **多项式加法同态**: 完全节点通过多项式加法同态重构出所有完全节点秘密值共享片段的总和;
- 4) **指数加法同态**: 完全节点通过指数加法同态性，计算用户私钥片段;
- 5) **用户私钥重构**: 用户 ID_k 在接收到足够数量的私钥片段之后，通过插值恢复出最终密钥。

4.3.2 CP-DK-ABE 方案构造

Setup 算法: 该算法被系统管理员执行，用来生成系统公钥和完全节点（管理属性私钥的节点）的私钥。在执行完该算法后，管理员将永久退出该系统。不失一般性，假设该方案中的完全节点有 n 个，表示为 $P = \{P_i\}_{i \in [1, n]}$ 。

该算法包含以下三步：

- 1) 系统管理员运行该算法生成系统主密钥和一些其他公共参数。首先，系统管理员执行该算法生成一个 p 阶双线性群 Φ 以及抗碰撞哈希函数 $H_1 : \{0,1\}^* \rightarrow \mathbb{G}$ 和 $H_2 : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$ 。最后，它随机选择 $\alpha, \beta \in \mathbb{Z}_p^*$ 作为该系统的密钥。
- 2) 管理员为每一个完全节点 P_i 随机选取整数 $p_i \in \mathbb{Z}_p^*$ 并计算 $Y = e(g, g)^\alpha$ 和 $h = g^\beta$ ，随后管理员公布系统公钥 PK 如下：

$$PK = \{\Phi, g, H_1, H_2, \{p_i\}_{i \in [1, n]}, Y, h\} \quad (4-4)$$

- 3) 为了分散区块链系统的权威，应该避免所有的主密钥由一个节点进行单独管理。在本步中，管理员通过 Shaimir 的 (t, n) 门限秘密共享方案，将主密钥 α 在所有的完全节点之间进行共享。在这种情况下，每个完全节点均可获取主密钥 α 的一个秘密片段，当且仅当至少 t 个完全节点进行合作，才能将主密钥 α 恢复出来。为了共享主密钥，管理员首先随机选取多项式 $f(x) = b_0 + b_1x + \dots + b_{t-1}x^{t-1} \pmod p$ ，使得 $\alpha = b_0$ ，其中 $\{b_i\}_{i \in [1, t-1]}$ 是 \mathbb{Z}_p^* 中的随机元素。其次，对于 $i \in [1, n]$ ，管理员计算 $\alpha_i = f(p_i)$ 并将 α_i 秘密地发送到完全节点 P_i 。最后，完全节点 P_i 的私

钥被定义如下：

$$SK_i = \{g^{1/\beta}, f(p_i)\} \in \mathbb{G} \times \mathbb{Z}_p^* \quad (4-5)$$

Key-generation 算法:在完全节点 P_i 的帮助下,该算法输入用户身份 ID_k 、属性集合 \mathbb{A} 以及系统公钥 PK , 输出用户 ID_k 的私钥 sk_k 。如图 4-4 所示,该算法是通过用户与完全节点之间的交互来实现的,它包含以下四个步骤。

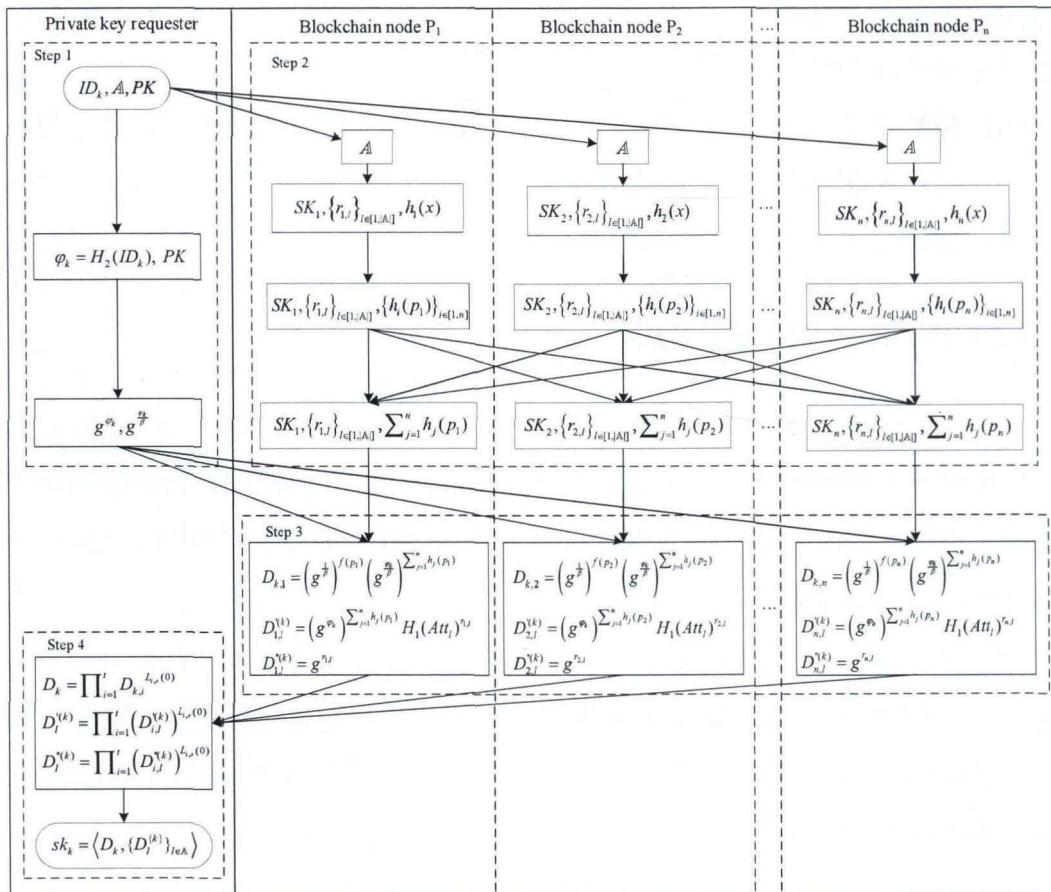


图 4-4 用户私钥生成过程

- 1) 对于具有属性集 $\mathbb{A} = \{Att_1^{(k)}, Att_2^{(k)}, \dots, Att_m^{(k)}\}$ 的用户 ID_k , 他首先向 P_i 发送 \mathbb{A} 并计算 $H_2(ID_k) = \varphi_k$ 。然后用户计算关于身份的参数 g^{φ_k} 和 $g^{\varphi_k/\beta}$, 并将它们发送给 P_i 。
- 2) 该步骤用来生成完全节点的参数。首先,对于任意属性 $Att_i^{(k)} \in \mathbb{A}$, 节点 P_i 随机选取 $r_{i,l} \in \mathbb{Z}_p^*$ 。随后, 该节点选取随机数 θ_i , 并通过 Shamir (t, n) 门限秘密共享方案将其在所有节点之间进行共享。对于 P_i , 随机选取 $t-1$ 阶多项式 $h_i(x) = \theta_i + a_{i,1}x + \dots + a_{i,t-1}x^{t-1} \pmod p$, 其中多项式系数 $a_{i,l}$ 是某个有限域中的随机元素。最后, P_i 计算 n 个共享片段 $h_i(p_1), \dots, h_i(p_n)$ 并将 $h_i(p_j)$ 秘密地发送给节点 P_j 。当 P_i 接收到所有其

他节点发送来的秘密片段 $h_j(p_i)$ 之后，计算 $\sum_{j=1}^n h_j(p_i)$ 。

- 3) 对于任意的节点 P_i ，利用其私钥 SK_i 以及前两步生成的参数，计算用户私钥片段如下：对于主私钥片段， P_i 计算如下：

$$D_{k,i} = \left(g^{\frac{1}{\beta}} \right)^{f(p_i)} \cdot \left(g^{\frac{\varphi_k}{\beta}} \right)^{\sum_{j=1}^n h_j(p_i)} \in \mathbb{G} \quad (4-6)$$

对于属性 $Att_i^{(k)} \in \mathbb{A}$ ， P_i 计算属性子密钥片段。

$$\begin{cases} D_{i,j}^{(k)} = \left(g^{\varphi_k} \right)^{\sum_{j=1}^n h_j(p_i)} \cdot H_1(Att_i^{(k)})^{r_j} \in \mathbb{G} \\ D_{i,l}^{(k)} = g^{r_l} \in \mathbb{G} \end{cases} \quad (4-7)$$

最后，节点 P_i 将 $D_{k,i}$ ， $D_{i,j}^{(k)}$ 以及 $D_{i,l}^{(k)}$ 发送给用户 ID_k 。

- 4) 在该步骤中，用户通过拉格朗日插值公式，从其接收到的私钥片段中重构主私钥和属性私钥。定义拉格朗日系数 $L_{i,S}(x) = \prod_{j \in S, j \neq i} \frac{x-p_j}{p_i-p_j}$ ，其中 $i \in \mathbb{Z}_p$ ， $S \subseteq \{1, 2, \dots, n\}$ 。对于主密钥，用户计算

$$D_k = \prod_{i=1}^t D_{k,i}^{L_{i,S}(0)} = g^{(\varphi_k \cdot \zeta + \alpha)/\beta} \in \mathbb{G} \quad (4-8)$$

对于属性密钥，用户计算

$$\begin{cases} D_l^{(k)} = \prod_{i=1}^t D_{i,l}^{(k)L_{i,S}(0)} = g^{\varphi_k \zeta} H_1(Att_l)^r \in \mathbb{G} \\ D_l''^{(k)} = \prod_{i=1}^t D_{i,l}''^{(k)L_{i,S}(0)} = g^r \in \mathbb{G} \end{cases} \quad (4-9)$$

最后，令 $D_l^{(k)} = \langle D_l^{(k)}, D_l''^{(k)} \rangle$ ，则用户获得其私钥 sk_k 可以表述如下：

$$sk_k = \{D_k, \{D_l^{(k)}\}_{l \in \mathbb{A}}\} \in \mathbb{G} \times \mathbb{G}^{m \times 2} \quad (4-10)$$

Encryption 算法：该算法与 BSW 方案中的加密算法是一致的，由数据用户执行用于加密敏感数据。为了方便起见，本文只考虑 $n=2$ 的情况下的 (t, n) 门限秘密共享，即该加密算法只关注两个输出的简单逻辑的“与”门和“或”门。在描述加密算法之前，首先给出访问树的一些大致介绍。

给定一个基于布尔函数的访问策略，其访问结构可以用二叉树 T 进行描述。在该树中，每个非叶节点表示一个逻辑“与”门或者“或”门。每个逻辑节点包含两个子节点，每个叶节点都与一个布尔谓词相关，用于属性匹配。此外，每个节点，无论是非叶节点还是叶节点均与一个秘密值相关联。令 v_x 表示节点 x 的秘密值， $v_{x,l}$ 和 $v_{x,r}$ 分别是其左右子节点的秘密值。为了在访问树 T 下加密消息 M ，加密算法按照以下步骤执行。

- 1) 首先, 将初始秘密值 $s \in \mathbb{Z}_p^*$ 赋值给访问树 T 的根节点, 即 $v_{root} = s$ 。随后, 加密算法通过“自上而下”的方式将该秘密值共享到树中的所有节点。该过程采用以下“与/或”秘密共享的方式, 将节点 \tilde{x} 的秘密值 $v_{\tilde{x}}$ 共享给其孩子节点: 对于“与”门, 节点选取一个新的随机数 $r \in \mathbb{Z}_p^*$, 令其左孩子节点秘密值为 $v_{\tilde{x},l} = r$, 则其右孩子节点处的秘密值为 $v_{\tilde{x},r} = s - r$; 对于“或”门而言, 节点令其左、右孩子节点处的秘密值均为秘密值 s , 即 $v_{\tilde{x},l} = v_{\tilde{x},r} = s$ 。最后, 直到每个叶子节点处都被分配一个秘密值, 结束上述分享过程。
- 2) 该步骤对应于叶子节点加密过程, 被称为 $EncNode()$ 。令 \mathcal{Y} 是访问树 T 中叶子节点组成的集合, 对于任意的 $\tilde{x} \in \mathcal{Y}$, 加密算法计算属性子密文 $C_{\tilde{x}} = \langle C'_{\tilde{x}}, C''_{\tilde{x}} \rangle \leftarrow EncNode(PK, \tilde{x}, v_{\tilde{x}})$, 其中 $C'_{\tilde{x}} = g^{v_{\tilde{x}}}$, $C''_{\tilde{x}} = H_1(att(\tilde{x}))^{v_{\tilde{x}}}$ 。
- 3) 对于访问树 T 以及消息 $M \in \mathbb{G}_T$, 加密算法利用初始秘密值 s 计算数据子密文 $\tilde{C} = M \cdot Y^s$ 与 $C_0 = h^s$ 。最终, 该算法输出消息 M 在访问树 T 下的密文如下:

$$CT_T = \{\tilde{C}, C_0, \{C_{\tilde{x}}\}_{\tilde{x} \in \mathcal{Y}}\} \in \mathbb{G}_T \times \mathbb{G} \times \mathbb{G}^{|\mathcal{Y}| \times 2} \quad (4-11)$$

Decryption 算法: 类似于 BSW 的 CP-ABE 方案中的解密算法, 本方案中的解密算法可以看作是上节加密算法的逆过程。具体来说, 该算法可分为三个步骤: 叶节点匹配、逻辑节点解密和数据子密文解密。

- 1) 叶节点匹配: 该步骤用于解密访问树 T 中的叶子节点, 对应的算法称为 $DecNode()$ 。对于任意的叶子节点 \tilde{x} , 令其对应的属性表示为 $att(\tilde{x})$, $C_{\tilde{x}}$ 是该叶子节点对应的属性子密文。若 $att(\tilde{x}) \in \mathbb{A}$, 则 $C_{\tilde{x}}$ 可由属性子密钥 $D_{att(\tilde{x})}^{(k)}$ 进行如下解密处理,

$$N_{\tilde{x}} = DecNode(C_{\tilde{x}}, D_{att(\tilde{x})}^{(\tilde{x})}) = \frac{e(D_{att(\tilde{x})}^{(k)}, C'_{\tilde{x}})}{e(D_{att(\tilde{x})}^{(k)}, C''_{\tilde{x}})} = e(g, g)^{\theta_k \cdot v_{\tilde{x}} \cdot \zeta} \in \mathbb{G}_T \quad (4-12)$$

其中, $\zeta = \sum_{i=1}^n \theta_i \pmod{p}$, 若 $att(\tilde{x}) \notin \mathbb{A}$, 则 $DecNode(C_{\tilde{x}}, D_{att(\tilde{x})}^{(k)}) = \perp$ 。

- 2) 逻辑节点解密: 对于逻辑节点 \tilde{x} , 令 $S_{\tilde{x}}$ 表示其孩子节点组成的集合。 $DecLogic()$ 表示逻辑节点解密算法, 它的输入为 \tilde{x} , $\{N_{\tilde{z}}\}_{\tilde{z} \in S_{\tilde{x}}}$ 和拉格朗日插值系数 $L_{i, S_{\tilde{x}}(0)}$, 输出节点 \tilde{x} 的解密结果为 $N_{\tilde{x}}$ 。

$$\begin{aligned}
 N_{\tilde{x}} &= DecLogic(\tilde{x}, \{N_{\tilde{z}}\}_{\tilde{z} \in S_{\tilde{x}}}, L_{i, S'_{\tilde{x}}(0)}) \\
 &= \prod_{\tilde{z} \in S_{\tilde{x}}} N_{\tilde{z}}^{L_{i, S'_{\tilde{x}}(0)}} = e(g, g)^{\varphi_k \cdot v_{\tilde{x}} \cdot \zeta} \in \mathbb{G}_T
 \end{aligned} \tag{4-13}$$

在上式中， i 是 \tilde{z} 在 $S_{\tilde{x}}$ 中的索引， $S'_{\tilde{x}}$ 表示索引集。最后，若属性集合 \mathbb{A} 满足访问树 T ，接下来仅需调用 $DecLogic()$ 算法来计算 T 中根节点的解密结果， $N_{root} = DecLogic(root, N_{\tilde{z}_{root}}, L_{i, S'_{root}}(0)) = e(g, g)^{\varphi_k \cdot s \cdot \zeta}$ 。

3) 数据子密文解密：数据子密文按照以下方式解密，

$$\frac{\tilde{C} \cdot N_{root}}{e(C_0, D_k)} = \frac{M \cdot e(g, g)^{\alpha \cdot s} \cdot e(g, g)^{\varphi_k \cdot s \cdot \zeta}}{e(g^{\beta \cdot s}, g^{(\varphi_k \cdot \zeta + \alpha)/\beta})} = M \in \mathbb{G}_T \tag{4-14}$$

4.4 脚本化可编程密文

脚本是一种可以自动解释任务的小型编程语言，其具有规则简单、轻量化等特征。基于脚本语言，本文引入了一个新的加解密框架，即脚本系统。得益于脚本语言自动解释任务的特点，在脚本系统中，用户加解密运算可由区块链节点通过脚本系统自动化执行，从而减轻用户端的计算负担。如图 4-5 所示，在本文的设计中，脚本系统包含交易、钱包、脚本运行栈和脚本解释器四个实体。

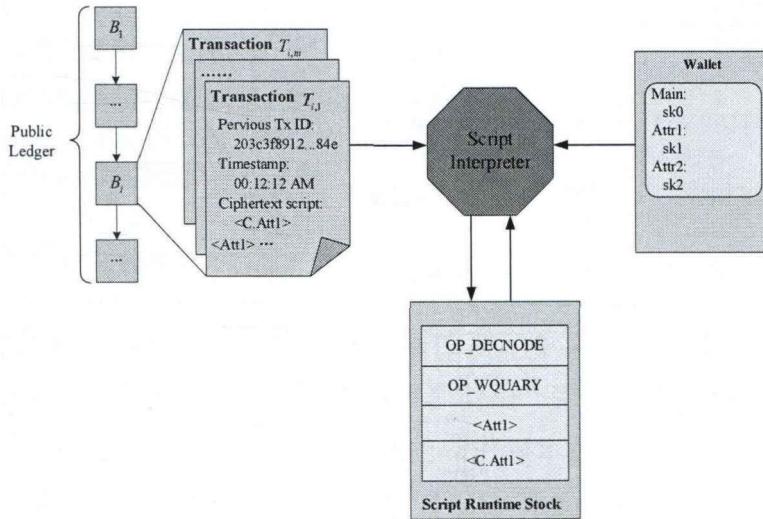


图 4-5 脚本系统框架图

- 1) 交易：每个交易包括交易数据和其他参数，比如前一交易的 ID。特别地，在系统中交易数据特指脚本化的密文。
- 2) 钱包：钱包是以“键-值”对的形式存储用户私钥和其他资产的实体。
- 3) 脚本运行栈：在该实体中，脚本数据和操作码可以根据栈规则进行操作，最后在栈顶返回运行结果。

- 4) 脚本解释器：脚本解释器是一个“虚拟处理器”，分别从交易和钱包中提取相关组件，然后通过脚本运行堆栈处理返回结果。

为了实现脚本化可编程密文，本章对第三章实验所采用的区块链智能合约平台 JuLiuSC v1.0 进行了二次开发。JuLiuSC 平台采用了基于 BitCoin 架构的区块链系统，该系统具有一套完整的脚本指令系统⁷，它包含了多达 256 种指令（opcode），并且允许通过添加指令解释程序的方式对其进行扩充。脚本语言是一种比智能合约更低级的语言形式，两者之间有如同高级语言（C、Java 等）与汇编语言相似的关系。

Bitcoin 脚本是一个类 Forth 且基于栈式结构的逆波兰式程序语言，它被用于描述区块链交易中特定算法的执行过程。Bitcoin 脚本语言是简单且非图灵完备的，它能够实现去中心化的验证。由于操作码的数量是可预测的，因此它的复杂性是有限的，因此它也被称之为最小的程序。这种对程序最小化的限定是加密货币设计的必要组成部分。更重要的，脚本是在区块链节点中的解释器中被执行，而不是在用户的设备中。任何人都不能覆盖其执行结果，同时也不能在执行过程中保存脚本的中间状态。因此，Bitcoin 脚本适用于加密和签名机制的实现，从而避免程序中 bug 和恶意代码。

表 4-1 Bitcoin 脚本中部分操作码

操作码	输入	输出	描述
OP_DUP	x	x, x	复制栈顶元素。
OP_HASH160	x	hash(x)	输入被两次哈希，第一次使用 SHA-256，第二次使用 RIPEMD-160。
OP_CHECKSIG	sig, pubkey	True/False	输入当前交易的签名和公钥验证签名有效性。
OP_EQUALVERIFY	a, b	Nothing/Fail	验证栈顶两个元素是否相等。

Bitcoin 脚本已经定义了丰富的操作码，用于实现各种高级操作。这些操作码由前缀“OP_”和后缀“操作名”组成，表 4-1 中介绍了四个常见操作码的输入，输出以及这些脚本操作码的基本描述。对于给定的操作码，尖括号中表示对应的操作数（例如， $\langle sig \rangle$ ），它将按照逆波兰法在操作码之前被推入栈中，以确保脚本解释器对栈中元素执行是按照操作码定义来实现的。

虽然 Bitcoin 脚本系统已经给出了一些操作码，但为了实现本章提出的方案，仍需要一些额外的操作码指令。如表 4-2 所示，通过替换保留操作码中 OP_NOP1 到 OP_NOP5，为本章构造的 CP-DK-ABE 方案定义了 5 个新操作码，这些操作码的详细描述如下。

⁷ 脚本指令系统详见 https://wiki.bitcoinsv.io/index.php/Opcodes_used_in_Bitcoin_Script

表 4-2 新添加操作码

操作码	输入	输出	描述
OP_WQUERY	x	out	从钱包中返回 x 对应的私钥。
OP_DEC_NODE	a, b	out	返回节点解密结果。
OP_DEC_AND	a, b	out	返回密文逻辑 AND 节点解密结果。
OP_DEC_OR	a, b	out	返回密文逻辑 OR 节点解密结果。
OP_DECRYPT	a, b, c, d	out	返回消息 m 。

OP_WQUERY 是钱包查询操作。区块链钱包是一个私钥集合，也可以是用于管理这些私钥的特殊程序。在 CP-DK-ABE 方案中，钱包用于存储用户 ID_k 的私钥 $sk_k = \{D_k, \{D_i^{(k)}\}_{i \in A}\}$ 以及属性集合 $A = \{Att_1^{(k)}, Att_2^{(k)}, \dots, Att_m^{(k)}\}$ 。为了清晰起见，私钥将由“键-值”对表示，即 $\langle Mainkey, D_k \rangle, \langle Att_i^{(k)}, D_{Att_i^{(k)}} \rangle$ 。

其中，用户 ID_k 的身份被定义为“Mainkey”。

算法 4-1 钱包查询 (OP_WQUERY)

输入：属性 x

输出：关于属性 x 的密钥 out

1. **if** $x \in \{Att_1^{(k)}, Att_2^{(k)}, \dots, Att_m^{(k)}\}$ **then**
 2. 从钱包中提取属性密钥 $D_x^{(k)}$ 并赋值给 out
 3. **else if** x 是 Mainkey **then**
 4. 从钱包中提取主密钥 D_k 并赋值给 out
 5. **else**
 6. $out \leftarrow \perp$
 7. **end if**
-

算法 4-2 节点解密 (OP_DEC_NODE)

输入：属性密文 $C_{\bar{x}} = \langle C'_{\bar{x}}, C''_{\bar{x}} \rangle$ 与属性子密钥 $D_x^{(k)} = \langle D'^{(k)}_x, D''^{(k)}_x \rangle$

输出：密文 $C_{\bar{x}}$ 的解密结果 $N_{\bar{x}}$

1. **if** $D_x^{(k)}$ 不是 \perp **then**
 2. $E \leftarrow e(D'^{(k)}_{att(\bar{x})}, C'_{\bar{x}})$
 3. $F \leftarrow e(D''^{(k)}_{att(\bar{x})}, C''_{\bar{x}})$
 4. $F' \leftarrow \text{inverse}(F)$
 5. $N_{\bar{x}} \leftarrow E \cdot F'$
 6. **else**
 7. $N_{\bar{x}} \leftarrow \perp$
 8. **end if**
-

算法 4-1 为钱包查询算法，该算法由脚本解释器运行该操作码来查询钱包中的私钥。它以属性 x 作为输入，并从钱包的“键-值”对中提取相应的属性子密钥 $D_x^{(k)}$ 。例如，以属性 $Att_i^{(k)}$ 为输入，算法输出自密钥 $D_{Att_i^{(k)}}^{(k)}$ 并将其置入栈中。此外，若属性 x 不在钱包中，则算法输出符号 \perp 。

算法 4-2 是叶子节点匹配运算，它对应于解密过程中第一步。脚本解释器运行该算法从栈顶弹出 $C_{\tilde{x}}$ 和 $D_{\tilde{x}}^{(k)}$ ，其中前者是属性子密文，后者为属性子密钥。若 $D_{\tilde{x}}^{(k)}$ 不为空字符 \perp ，则脚本解释器输出解密结果 $N_{\tilde{x}}$ ，否则解释器将空字符 \perp 推入栈中。

算法 4-3 “与”门解密 (OP_DEC_AND)

输入：“与”门节点 \tilde{x} 两个孩子节点的解密结果 $N_{\tilde{x}.left}$ 和 $N_{\tilde{x}.right}$

输出：节点 \tilde{x} 的解密结果 $N_{\tilde{x}}$

1. if $N_{\tilde{x}.left}$ 或 $N_{\tilde{x}.right}$ 为 \perp then
 2. $N_{\tilde{x}} \leftarrow N_{\tilde{x}.left} \cdot N_{\tilde{x}.right}$
 3. else
 4. $N_{\tilde{x}} \leftarrow \perp$
 5. end if
-

算法 4-3 是一个简化的仅有两个输入的拉格朗日插值操作码，用于解密带有“与”门的密文逻辑非叶子节点。令 \tilde{x} 为节点， $N_{\tilde{x}.left}$ 和 $N_{\tilde{x}.right}$ 分别表示 \tilde{x} 的左、右孩子节点。如算法所示，解释器提取栈顶两个元素 $N_{\tilde{x}.left}$ 和 $N_{\tilde{x}.right}$ 作为输入，若二者均非 \perp ，解释器返回 $N_{\tilde{x}}$ 的解密结果，否则将 \perp 推入栈中。

算法 4-4 “或”门解密 (OP_DEC_OR)

输入：“或”门节点 \tilde{x} 两个孩子节点的解密结果 $N_{\tilde{x}.left}$ 和 $N_{\tilde{x}.right}$

输出：节点 \tilde{x} 的解密结果 $N_{\tilde{x}}$

1. if $N_{\tilde{x}.left}$ 或 $N_{\tilde{x}.right}$ 为 \perp then
 2. $N_{\tilde{x}} \leftarrow \{N_{\tilde{x}.left}, N_{\tilde{x}.right}\}$
 3. else if $N_{\tilde{x}.left}$ 或 $N_{\tilde{x}.right}$ 其中一个为 \perp then
 4. if $N_{\tilde{x}.left}$ 为 \perp then
 5. $N_{\tilde{x}} \leftarrow N_{\tilde{x}.right}$
 6. else
 7. $N_{\tilde{x}} \leftarrow N_{\tilde{x}.left}$
 8. end if
 9. else
 10. $N_{\tilde{x}} \leftarrow \perp$
 11. end if
-

算法 4-4 是一个简化的仅有两输入的拉格朗日插值操作码，该算法被用于解密密文逻辑“或”门非叶节点。与算法 4-3 类似，解释器提取栈顶两个元素 $N_{\tilde{x}.left}$ 和 $N_{\tilde{x}.right}$ 作为输入，若二者均非 \perp ，解释器返回二者中的一个作为 $N_{\tilde{x}}$ 的解密结果；若仅有一个为 \perp ，解释器返回非 \perp 项作为 $N_{\tilde{x}}$ 的解密结果；若二者均为 \perp ，解释器返回 \perp 。

算法 4-5 数据子密文解密 (OP_DECRYPT)

输入：数据子密文 C_0 , \tilde{C} , 用户主私钥 D_k 以及根节点解密结果 N_{root}

输出：数据密文的解密结果 out

1. **if** D_k 或 N_{root} 为 \perp **then**
 2. $E \leftarrow \tilde{C} \cdot N_{root}$
 3. $F \leftarrow e(C_0, D_k)$
 4. $F' \leftarrow \text{inverse}(F)$
 5. $out \leftarrow E \cdot F'$
 6. **else**
 7. $out \leftarrow \perp$
 8. **end if**
-

算法 4-5 对应于解密数据子密文的操作码对应的算法。解释器提取栈顶的四个元素作为输入，即数据子密文 C_0 和 \tilde{C} ，用户的主私钥 D_k 以及根节点的解密结果 N_{root} 。若 D_k 和 N_{root} 均非 \perp ，解释器返回数据子密文的解密结果，否则解释器返回 \perp 。

4.4.1 脚本化密文生成

根据上述新增操作码，本节将详细描述在访问策略树下生成脚本化密文的过程。由于任意多分支树都可以转换为二叉树，为了便于描述，不失一般性，将本章构造的 CP-DK-ABE 方案中的访问树设置为二叉树。

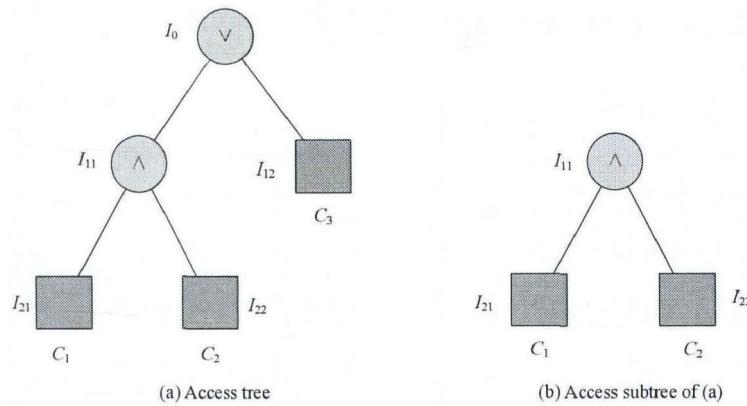


图 4-6 访问树实例

以访问策略 Policy:: = $(Att_1 \wedge Att_2) \vee Att_3 = (\text{计算机专业} \wedge \text{学生}) \vee \text{教授}$ 为例，首先将该策略转化成策略树 T ，如图 4-6 所示。该图中，叶子节点表示属性，具体来说 I_{21} 表示属性“计算机专业”， I_{22} 表示属性“学生”， I_{12} 表示“教授”。 I_0 为逻辑“或”节点， I_{11} 为逻辑“与”节点，这意味着一个“计算机专业的学生”或者任意“教授”可以满足访问策略树 T 。

算法 4-6 密文生成算法 (CGA)

输入：节点 \tilde{x} ，秘密值 s ，公钥 PK ，消息 M 和 out

输出：脚本化密文

1. **if** 节点 \tilde{x} 为叶子节点 **then**
 2. 获取节点 \tilde{x} 对应的属性 $Att(\tilde{x})$
 3. $C_{\tilde{x}} \leftarrow \langle C'_{\tilde{x}}, C''_{\tilde{x}} \rangle = \langle g^s, H_1(Att(\tilde{x}))^s \rangle$
 4. 将 $(C_{\tilde{x}} \langle Att(\tilde{x}) \rangle \text{ OP_WQUERY OP_DEC_NODE})$ 添加到 out 中
 5. **else**
 6. **if** 节点 \tilde{x} 为“与”门逻辑 **then**
 7. 随机选取 r
 8. 计算 CGA($\tilde{x}.\text{left}, r, out$)
 9. 计算 CGA($\tilde{x}.\text{right}, s-r, out$)
 10. 将 (OP_DEC_OR_AND) 添加到 out 中
 11. **else**
 12. CGA($\tilde{x}.\text{left}, s, out$)
 13. CGA($\tilde{x}.\text{right}, s, out$)
 14. 将 (OP_DEC_OR_OR) 添加到 out 中
 15. **else if**
 16. **if** 节点 \tilde{x} 为根节点 **then**
 17. $\tilde{C} = M \cdot e(g, g)^{as}, C_0 = g^{\beta s}$
 18. 将 $(\tilde{C} \ C_0 \ \langle \text{Mainkey} \rangle \text{ OP_WQUERY OP_DECRYPT})$ 添加到 out 中
 19. **end if**
 20. **end if**
-

为了在访问策略树 T 下对消息 M 进行加密，加密者随机选取一个秘密值 s 并运行密文生成算法 CGA（算法 4-6），生成相应密文 CT_T 。密文生成过程具体如下，CGA 算法的输入为节点 \tilde{x} ，秘密值 s ，公钥 PK ，消息 M 和输出 out 。若输入的 \tilde{x} 为叶子节点，则 CGA 算法计算并输出属性子密文 $C_{\tilde{x}} \leftarrow \langle C'_{\tilde{x}}, C''_{\tilde{x}} \rangle = \langle g^s, H_1(Att(\tilde{x}))^s \rangle$ ，并将生成的属性密文、属性名以及对应的操

作码添加到 out 中, 如 $(C_{\tilde{x}} \langle Att(x) \rangle OP_WQUERY OP_DEC_NODE)$ 。若 \tilde{x} 为“与”门逻辑节点, 则算法随机选取 r 并运行 $CGA(\tilde{x}.\text{left}, r, out)$ 和 $CGA(\tilde{x}.\text{right}, s-r, out)$, 将 $(OP_DEC_OR_AND)$ 添加到 out 中。若算法输入节点 \tilde{x} 为“或”门逻辑节点, 则算法运行 $CGA(\tilde{x}.\text{left}, s, out)$ 和 $CGA(\tilde{x}.\text{right}, s, out)$, 并将 $(OP_DEC_OR_OR)$ 添加到 out 中。此外, 若 \tilde{x} 为根节点, 则算法 CGA 计算数据子密文 $\tilde{C} = M \cdot e(g, g)^{\alpha s}$ 和 $C_0 = g^{\beta s}$, 并将 $(\tilde{C}, C_0, \langle Mainkey \rangle OP_WQUERY OP_DECRYPT)$ 添加到 out 中。注意, $\langle Mainkey \rangle$ 为数据子密文, 仅用于表示密文类型。以图 4-6(a)为例, 按照 CGA 算法, 策略树 T 下的密文可表示如下:

$$CT_T = \{ \langle C_1 \rangle \langle Att_1 \rangle OP_WQUERY OP_DEC_NODE \langle C_2 \rangle \langle Att_2 \rangle OP_WQUERY OP_DEC_NODE OP_DEC_AND \langle C_3 \rangle \langle Att_3 \rangle OP_WQUERY OP_DEC_NODE OP_DEC_OR \langle \tilde{C} \rangle \langle C_0 \rangle \langle Mainkey \rangle OP_WQUERY OP_DECRYPT \}$$

4.4.2 脚本驱动的解密

在本章方案的构造中, 敏感数据被加密后以脚本代码的形式提交到公共账本。想要获取受保护数据, 用户应在自己的钱包和脚本指令的帮助下, 运行脚本解释器来解密密文。接下来, 本小节将详细介绍脚本解密的过程。

如图 4-6(b)所示, 以访问子树 $T_{I_{11}}$ 为例, 介绍脚本解密过程。首先将策略子树 $T_{I_{11}}$ 下的密文分成属性子密文

$$\begin{aligned} & \langle C_1 \rangle \langle Att_1 \rangle OP_WQUERY OP_DEC_NODE \langle C_2 \rangle \langle Att_2 \rangle \\ & OP_WQUERY OP_DEC_NODE OP_DEC_AND \end{aligned}$$

和数据子密文

$$\langle \tilde{C} \rangle \langle C_0 \rangle \langle Mainkey \rangle OP_WQUERY OP_DECRYPT$$

与密文分类相对应, 解密过程也被分成两部分: 属性密文脚本解密和数据密文脚本解密。以访问子树 $T_{I_{11}}$ 下生成的密文为例, 详细介绍脚本化解密过程。

属性子密文的脚本化解密。当脚本解释器从密文脚本中读到属性子密文脚本后, 它将按照从左至右的顺序执行该密文。如

所示, 给出属性子密文的脚本解密过程描述。

- 1) 步骤 1, 对栈进行清空并初始化;
- 2) 步骤 2 和步骤 3, 对密文 $\langle C_1 \rangle$ 进行解密。脚本解释器首先从栈中按照先后顺序抽取 $\langle C_1 \rangle$ 和 $\langle Att_1 \rangle$ 。随后, 脚本解释器在步骤 4 中运行

OP_WQUERY 算法，用于获取属性 $\langle Att_1 \rangle$ 对应的私钥 $\langle sk_1 \rangle$ 。在步骤 5 中，脚本解释器抽取 $\langle C_1 \rangle$ 和 $\langle sk_1 \rangle$ 并运行 **OP_DEC_NODE** 算法，最后返回密文 $\langle C_1 \rangle$ 的解密结果 $\langle N_{Att_1} \rangle$ ；

- 3) 步骤 6 到步骤 9，对密文 $\langle C_2 \rangle$ 进行解密。与密文 $\langle C_1 \rangle$ 的解密过程一致，脚本解释器将密文 $\langle C_2 \rangle$ 的解密结果 $\langle N_{Att_2} \rangle$ 返回栈顶；
- 4) 步骤 10，根节点解密。脚本解释从栈中依次抽取 $\langle N_{Att_2} \rangle$ 和 $\langle N_{Att_1} \rangle$ 并执行 **OP_DEC_AND** 算法，最后返回根节点处的解密结果 I_{11} 。

表 4-3 属性子密文脚本解密过程

步骤	栈	密文脚本	相关说明
1	Empty	$\langle C_1 \rangle \langle Att_1 \rangle$ OP_WQUERY OP_DEC_NODE $\langle C_2 \rangle \langle Att_2 \rangle$ OP_WQUERY OP_DEC_NODE OP_DEC_AND	
2	$\langle C_1 \rangle$	$\langle Att_1 \rangle$ OP_WQUERY OP_DEC_NODE $\langle C_2 \rangle \langle Att_2 \rangle$ OP_WQUERY OP_DEC_NODE OP_DEC_AND	将 $\langle C_1 \rangle$ 添加到栈中。
3	$\langle C_1 \rangle \langle Att_1 \rangle$	OP_WQUERY OP_DEC_NODE $\langle C_2 \rangle \langle Att_2 \rangle$ OP_WQUERY OP_DEC_NODE OP_DEC_AND	属性 $\langle Att_1 \rangle$ 被添入栈。
4	$\langle C_1 \rangle \langle sk_1 \rangle$	OP_DEC_NODE $\langle C_2 \rangle \langle Att_2 \rangle$ OP_WQUERY OP_DEC_NODE OP_DEC_AND	查询与密文 $\langle C_1 \rangle$ 匹配的私钥 $\langle sk_1 \rangle$ 。
5	$\langle N_{Att_1} \rangle$	$\langle C_2 \rangle \langle Att_2 \rangle$ OP_WQUERY OP_DEC_NODE OP_DEC_AND	对密文 $\langle C_1 \rangle$ 进行解密。
6	$\langle N_{Att_1} \rangle \langle C_2 \rangle$	$\langle Att_2 \rangle$ OP_WQUERY OP_DEC_NODE OP_DEC_AND	将 $\langle C_2 \rangle$ 添加到栈中。
7	$\langle N_{Att_1} \rangle$ $\langle C_2 \rangle \langle Att_2 \rangle$	OP_WQUERY OP_DEC_NODE OP_DEC_AND	属性 $\langle Att_2 \rangle$ 被添入栈。
8	$\langle N_{Att_1} \rangle$ $\langle C_2 \rangle \langle sk_2 \rangle$	OP_DEC_NODE OP_DEC_AND	查询与密文 $\langle C_2 \rangle$ 匹配的私钥 $\langle sk_2 \rangle$ 。
9	$\langle N_{Att_1} \rangle \langle N_{Att_2} \rangle$	OP_DEC_AND	对密文 $\langle C_2 \rangle$ 进行解密。
10	$\langle N_{I_{11}} \rangle$	Empty	解密根节点 I_{11} 。

数据子密文的脚本化解密。在属性子密文被解密之后，脚本解释器可以

利用策略树根节点的解密结果 I_{11} ，从数据子密文中恢复出消息 M 。表 4-4 给出了数据子密文的脚本解密过程如下：

- 1) 步骤 1 对栈进行清空并初始化；
- 2) 步骤 2 到步骤 4 为钱包查询过程。在该过程中，脚本解释器首先从栈中抽取 $N_{I_{11}}$ 以及数据子密文 $\langle \tilde{C} \rangle$ 和 $\langle C_0 \rangle$ 。接下来，解释器执行 OP_WQUERY 算法用于从钱包中查询相关数据子密文对应的私钥。最后，脚本解释器将主私钥 D_k 存入栈中。
- 3) 步骤 5 是解密数据子密文过程。脚本解释器执行 OP_DECRYPT 算法来恢复消息 M 。

表 4-4 数据子密文脚本解密过程

步骤	栈	密文脚本	相关说明
1	Empty	$\langle N_{I_{11}} \rangle \langle \tilde{C} \rangle \langle C_0 \rangle \langle Mainkey \rangle$ OP_WQUERY OP_DECRYPT	
2	$\langle N_{I_{11}} \rangle \langle \tilde{C} \rangle \langle C_0 \rangle$	$\langle Mainkey \rangle$ OP_WQUERY OP_DECRYPT	$\langle N_{I_{11}} \rangle \langle \tilde{C} \rangle \langle C_0 \rangle$ 被添加入栈。
3	$\langle N_{I_{11}} \rangle \langle \tilde{C} \rangle \langle C_0 \rangle$ $\langle Mainkey \rangle$	OP_WQUERY OP_DECRYPT	$\langle Mainkey \rangle$ 被添加入栈。
4	$\langle N_{I_{11}} \rangle \langle \tilde{C} \rangle \langle C_0 \rangle$ $\langle D_k \rangle$	OP_DECRYPT	查询数据私钥 $\langle D_k \rangle$ ，并将其添加入栈。
5	M	Empty	解密数据密文获取消息。

4.5 安全性分析

本节将分别对用户私钥生成算法的隐私性以及本章提出的 CP-DK-ABE 方案的安全性进行详细分析。

4.5.1 Key-generation 算法的隐私性分析

本章方案中 Key-generation 算法的隐私性分析将从“被动攻击”和“主动攻击”两个方面进行讨论。被动攻击是指敌手仅在通信网络上进行监听攻击，主动攻击表示敌手有能力篡改信道中正在传输的消息。该算法密钥隐私意味着在用户私钥生成过程中，无论是主动还是被动敌手，都不能在用户私钥生成过程中，获取关于用户私钥以及完全节点持有的主密钥的任何信息。

定理 4-1 若 Shamir 门限秘密共享方案和判定性线性 Diffie-Hellman

(DLDH) 假设成立, 则本章提出的方案中 Key-generation 算法对于有限个数的完全节点的合谋攻击具有隐私性, 即在至多 $t-1$ 完全节点合谋的情况下, 敌手 \mathcal{A} 猜中算法输入值 $\zeta = \sum_{i=1}^n \theta_i$ 的概率是可忽略的。

证明: 接下来将分别从被动敌手攻击和主动敌手攻击两种场景对上述定理的正确性进行证明。

对于被动敌手攻击, 若存在一个 PPT 敌手 \mathcal{A} 可以以一个不可忽略的概率 ϵ 猜中 $\zeta' = \zeta$, 则可以构造一个挑战者以 $\epsilon/2$ 的概率解决 DLDH 问题。

因为证明的目标是 Key-generation 算法的隐私性, 所以在 Setup 算法中仅模拟那些只用于密钥生成的公钥。

1) 系统建立: 挑战者随机选取 $a, b \in \mathbb{Z}_p^*$, 此外, 挑战者选择一个随机数用来替代原方案中 Setup 算法中的第 2 步的输出结果。

$$\begin{cases} f(x) = a + \sum_{i=1}^{t-1} a_i x^i \pmod{p} \\ h(x) = c' + \sum_{i=1}^{t-1} c_i x^i \pmod{p} \end{cases} \quad (4-15)$$

其中上式中 $a_i, c_i \in \mathbb{Z}_p^*$ 。完全节点 P_i 随机选取 $p_i \in \mathbb{Z}_p^*$ 并令 $a = \alpha, b = 1/\beta$, $c' = \sum_{i=1}^n \theta_i$ 以及 $G = g$ 。随后, 挑战者选取抗碰撞哈希函数 $H_1 : \{0,1\}^* \rightarrow \mathbb{G}$ 和 $H_2 : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$ 并将公钥 PK 发送给敌手 \mathcal{A} 。

$$PK = \left\{ \{p_i\}_{i \in [1,n]}, G, H_1, H_2, G^b, G^{1/b} \right\} \quad (4-16)$$

2) 学习阶段: 在模拟方案中有 n 个完全节点, 假设最多 $t-1$ 完全节点被敌手腐化 ($t-1 < n$), 令被腐化完全节点构成的集合为 $\mathcal{I} = \{P_1, P_2, \dots, P_{t-1}\}$ 。接下来, 对于 \mathcal{I} 中的任意完全节点 P_i , 它随机选取 $r_{i,l} \in \mathbb{Z}_p^*$ 并计算私钥如下:

$$\begin{cases} D_{i,k} = (G^b)^a \left(\prod_{j=1}^{t-1} G^{ba_j p_j} \right) \cdot (G^{b\varphi_k})^{c'} \left(\prod_{j=1}^{t-1} G^{b\varphi_k c_j p_j} \right) \\ D_{i,l}^{(k)} = (G^{\varphi_k})^{c'} \left(\prod_{j=1}^{t-1} G^{\varphi c_j p_j} \right) H_1(Att_l^{(k)})^{r_{i,l}} \\ D_{i,l}''^{(k)} = G^{r_{i,l}} \end{cases} \quad (4-17)$$

3) 挑战阶段: 敌手给出将要挑战的用户身份 ID_k , 随后挑战者随机选取 $c \in \mathbb{Z}_p^*$, 令 $\varphi_k = c/c'$ 并计算 $G^{b\varphi_k}$, 挑战私钥可以表示如下:

$$\left\{ \begin{array}{l} D_{i,k} = (G^b)^a \left(\prod_{j=1}^{t-1} G^{ba_j p_i^j} \right) \cdot (G^{b\varphi_k})^{c'} \left(\prod_{j=1}^{t-1} G^{b\varphi_k c_j p_i^j} \right) \\ D_{i,l}'^{(k)} = (G^{\varphi_k})^{c'} \left(\prod_{j=1}^{t-1} G^{\varphi c_j p_i^j} \right) H_1(Att_l^{(k)})^{r_{i,l}} \\ D_{i,l}''^{(k)} = G^{r_{i,l}} \end{array} \right. \quad (4-18)$$

最终，挑战者重构如下私钥。

$$\left\{ \begin{array}{l} D_k = \prod_{i=1}^t D_{k,i}^{L_{i,s}(0)} = Z \cdot (G^{b\varphi_k})^{\zeta'} \\ D_l'^{(k)} = \prod_{i=1}^t D_{i,l}^{(k)L_{i,s}(0)} = G^c H_1(Att_l)^{(\sum_{i=1}^t r_{i,l}) \cdot L_{i,s}(0)} \\ D_l''^{(k)} = \prod_{i=1}^t D_{i,l}''^{(k)L_{i,s}(0)} = G^{\sum_{i=1}^t r_{i,l} \cdot L_{i,s}(0)} \end{array} \right. \quad (4-19)$$

4) 响应阶段：最后敌手 \mathcal{A} 给出关于完全节点输入的响应。

若 $Z = G^{ab}$ ，挑战者在真实攻击中是不可区分的，因此敌手正确猜中 $\zeta' = \zeta$ 的概率不低于 $1/2 + \epsilon$ ；若 Z 是群 \mathbb{G} 中的随机元素，则可以容易发现这时的用户私钥是“一次一密”，这是由于 $G^{b\zeta'}$ 是在 Z 中计算的。因此，敌手正确地猜中 c 的概率为 $1/2$ 。进一步地，敌手 \mathcal{A} 解决 DLDH 问题的优势为

$$\frac{1}{2} \times \left(\frac{1}{2} + \epsilon \right) + \frac{1}{2} \times \frac{1}{2} - \frac{1}{2} = \frac{\epsilon}{2} \quad (4-20)$$

对于主动敌手攻击，假定敌手在主动攻击情况下可能改变其输入。具体来说，敌手可能攻击 Key-generation 算法中的步骤 2 和步骤 3。接下来，将分析上述攻击是无效的。

1) 对于步骤 2 的攻击：攻击者主动攻击 Key-generation 算法的第 2 步，即改变完全节点生成的随机数。在原始方案的算法中，无法抵抗敌手对这一步的主动攻击。针对这一问题，可以使用可验证的秘密共享技术（Verifiable Secret Sharing, VSS），对原始算法进行微小的修改，从而达到抵抗主动攻击的目的。接下来，将使用 VSS 来描述修改后的算法如何对抗主动攻击。

完全节点 P_i 首先选取随机数 $\theta_i \in \mathbb{Z}_p^*$ ，令多项式 $h_i(x) = \theta_i + \sum_{j=1}^{t-1} c_{i,j} x^j$ 用于共享其秘密值。同时， P_i 对多项式系数计算承诺 $g^{c_{i,j}}$ ，并通过区块链网络以交易的形式广播给其它完全节点。最后，每个完全节点 P_i 可以通过接收到的秘密片段独立地生成联合的秘密片段 $\sum_{j=1}^n h_j(P_i)$ 。

为了确定是否存在恶意节点， P_i 可以使用承诺 $g^{c_{i,j}}$ 和联合的秘密片段

$\sum_{j=1}^n h_j(P_i)$ 进行如下判定。一旦有完全节点伪造了随机数，例如 P_i 随机选择 θ_i ，并令 $h_i(x) = \theta_i + \sum_{j=1}^{t-1} c_{i,j}x^j$ ，则 $g^{c_{i,j}} = g^{\sum_{j=1}^n h_j(p_i)}$ 不再成立。在这种情况下，至少有一个完全节点伪造了它的输入。

2) 对于步骤 3 的攻击：攻击者主动攻击 Key-generation 算法的第 3 步。假设完全节点 P_i 被腐化，它利用群 \mathbb{G} 中的随机元素 R_1 和 R_2 代替私钥 $D_{k,i}$ 和 $D_{i,j}^{(k)}$ 。接下来，将演示如何判定重建私钥中是否包含来自 P_i 的伪造信息。

随机选择 t 个完全节点来重建私钥。重复此这样的选择两次，且两次被选择的节点不完全相同。通过比较两次私钥重构的结果，可以判断所选节点是否包含恶意节点 P_i 。更具体地说，若两次私钥的重构结构一致，则这两次中均不包含恶意节点；若重构结果不同，则有两种可能：其中一次选择中包含恶意节点或两者都包含恶意节点。由于主私钥与属性子密钥的分析相同，接下来以主私钥 D_k 重构为例，进行具体分析。

1) 若两次选择的 t 节点均不包含 P_i ，即不包含恶意节点，则私钥的两次

$$\text{重构必定一致，即 } D_k = g^{\frac{1}{\beta}[\alpha + \varphi_k \sum_{j=1}^n \theta_j]}.$$

2) 若其中一次选择中包含恶意节点 P_i ，则两个主私钥分别重构如下：

$$\begin{cases} D_k^{(1)} = \prod_{i=1}^t D_{k,i}^{L_{i,S}(0)} = R_1 \cdot \prod_{i=1, i \neq i}^t D_{k,i}^{L_{i,S}(0)} \\ D_k^{(2)} = \prod_{i=1}^t D_{k,i}^{L_{i,S}(0)} = g^{\frac{1}{\beta}[\alpha + \varphi_k \sum_{j=1}^n \theta_j]} \end{cases} \quad (4-21)$$

在 $D_k^{(1)}$ 中， R_1 是 \mathbb{G} 中的一个随机元素。由 Shamir(t, n) 门限秘密共享方案，重构秘密值 α 和 $\sum_{i=1}^n \theta_i$ 是不可能的。因此上述私钥重构不相等。

3) 若两次选择中均包含恶意节点，根据前面的假设，两次选择中至少有一个完全节点是不同的。令 I_1 和 I_2 为 $\{1, 2, \dots, n\}$ 中两个不同的包含 t 个元素的子集，则私钥重构如下：

$$\begin{cases} D_k^{(1)} = \prod_{i \in I_1} D_{k,i}^{L_{i,S}(0)} = R_1 \cdot \prod_{i \in I_1, i \neq i} D_{k,i}^{L_{i,S}(0)} \\ D_k^{(2)} = \prod_{i \in I_2} D_{k,i}^{L_{i,S}(0)} = R_1 \cdot \prod_{i \in I_2, i \neq i} D_{k,i}^{L_{i,S}(0)} \end{cases} \quad (4-22)$$

根据 Shamir(t, n) 门限秘密共享方案，重构秘密值 α 和 $\sum_{i=1}^n \theta_i$ 是不可能的，因此 $D_k^{(1)}$ 和 $D_k^{(2)}$ 不相等。

4.5.2 CP-DK-ABE 语义安全性分析

定理 4-2 若判定性双线性 Diffie-Hellman (DBDH) 假设成立，则本章提出的 CP-DK-ABE 方案在选择明文攻击下是语义安全的。

证明：假设存在一个敌手 \mathcal{A} 可以以不可忽略的优势 ϵ 破坏 CP-DK-ABE 方案，则可以构建一个 PPT 挑战者 \mathcal{C} 可以以 $\epsilon/2$ 优势解决 DBDH 问题。挑战者 \mathcal{C} 首先在 $\{0,1\}$ 中随机选取 μ ，若 $\mu=0$ ，则挑战者 \mathcal{C} 令 DBDH 问题实例为 $[G, G^a, G^b, G^c, Z = e(G, G)^{abc}]$ ；否则，令实例置为 $[G, G^a, G^b, G^c, Z = e(G, G)^z]$ ，其中 a, b, c, z 为 \mathbb{Z}_p^* 中的随机数。

- 1) 初始话：在运行安全游戏之前，敌手 \mathcal{A} 宣布要挑战访问策略 W^* 。
- 2) 系统建立：挑战者设置 $Y = e(G_1, G_2) = e(G, G)^{ab}$ 和 G^d ，这意味着 $G_1 = G^a$ ， $G_2 = G^b$ 和 $\beta = d$ 。此外，挑战者控制两个随机哈希查询。
- 3) Hash 查询：在该阶段，敌手 \mathcal{A} 将进行哈希询问。令 q_{H_i} 是对随机预言机 H_i 的哈希查询的次数，其中 $i \in \{1, 2\}$ 。在查询的开始之前有两个空哈希列表，挑战者将所有查询和响应记录如下：

- ① 设 v_j 为要进行 H_1 哈希查询的属性。若 v_j 已在哈希列表中，挑战者将按照列表中存储内容返回查询结果。否则，挑战者在 \mathbb{Z}_p^* 中随机选取 t_j 并计算

$$H_1(v_j) = \begin{cases} g^{t_j}, & v_j \in W^* \\ g^{b t_j}, & v_j \notin W^* \end{cases} \quad (4-23)$$

最后，挑战者将计算结果 $H_1(v_j)$ 作为哈希查询结果返回给敌手，并将 $(v_j, H_1(v_j))$ 添加到哈希列表中。

- ② 设 φ_k 为进行 H_2 哈希查询的用户身份。若 φ_k 已在哈希列表中，挑战者将按照哈希列表中存储内容返回查询结果。否则，挑战者随机选取 $y_k \in \mathbb{Z}_p^*$ ，返回哈希查询结果 $H_2(\varphi_k) = y_k$ ，并将 $(\varphi_k, H_2(\varphi_k))$ 添加到哈希列表中。

- 4) 学习阶段 I：敌手对它的属性集合 \mathbb{A} 进行适应性的私钥查询，其中属性集合 \mathbb{A} 不满足要挑战的访问策略 W^* 。敌手首先计算 g^{φ_k} 和 $g^{\varphi_k/d}$ 并发送给挑战者，挑战者选取 $\theta^* \in \mathbb{Z}_p^*$ 并返回主私钥 $D_k = \prod_{i=1}^n D_{k,i}^{l_{i,s}(0)} = g^{(ab+\theta^*\varphi_k)/d}$ 。对于属性私钥，挑战者随机选取 $r'_j \in \mathbb{Z}_p^*$ ，令 $r_j = r'_j/t_j$ 并计算属性子私钥如下：

$$\begin{cases} D_j^{(k)} = g^{\varphi_k \theta^*} H_1(v_j)^{r_j} = g^{\varphi_k \theta^*} \cdot g^{r_j} \\ D_j''^{(k)} = g^{r_j} = g^{r_j / t_j} \end{cases} \quad (4-24)$$

5) 挑战阶段: 敌手 \mathcal{A} 输出两个相同长度的消息 $M_0, M_1 \in \mathbb{G}_T$ 中, 随后挑战者随机选取数 $c \in \mathbb{Z}_p^*$, 令 $c' = c/d$ 并在访问策略 W^* 下加密 M_ρ , 其中 $\rho \in \{0,1\}$ 。最后, 挑战者返回相应的密文 $\tilde{C} = M_\rho \cdot Z$ 和 $C = g^{dc'} = g^c$, 这意味着 $c' = s$ 。按照原始方案中的 Encryption 算法, 挑战者在访问策略树中将 c' 从根节点共享到叶节点。之后, 每个叶节点 \tilde{x} 可以获得一个分享 $q_{\tilde{x}}(0)$ 。最终, 挑战者计算 $C_{\tilde{x}} = g^{q_{\tilde{x}}(0)}$ 以及 $C'_{\tilde{x}} = H_1(\tilde{x})^{q_{\tilde{x}}(0)}$ 并将挑战密文 CT_{W^*} 发送给敌手, 其中挑战密文形式如下:

$$CT_{W^*} = \left\{ M_\rho \cdot Z, g^c, \left\{ g^{q_x(0)}, H_1(\tilde{x})^{q_x(0)} \right\}_{x \in W^*} \right\} \quad (4-25)$$

6) 学习阶段 II: 与阶段 I 类似, 敌手对它的属性集合 \mathbb{A} 进行适应性的私钥查询, 其中属性集合 \mathbb{A} 不满足要挑战的访问策略 W^* 。

7) 猜测阶段: 敌手输出它关于 ρ 的猜测 ρ' 。若 $\rho' = \rho$, 挑战者输出 $\mu = 0$, 在这种情况下, 挑战者给出一个正确的 DBDH 实例; 否则, 挑战者输出 $\mu = 1$, 在这种情况下, 敌手输出一个随机的 4 元组。

如果 $\mu = 0$, 则 \mathcal{A} 获得与策略 W^* 相关的有效密文。在这种情况下, 根据前述定义, 敌手的优势为 ϵ 。因此, 有 $\Pr[\rho' = \rho | \mu = 0] > 1/2 + \epsilon$ 。如果 $\mu = 1$, 则无论是对消息 M_0 还是对消息 M_1 加密, 挑战密文具有相同的分布情况。因此, 有概率 $\Pr[\rho' = \rho | \mu = 1] = 1/2$ 。因此, 挑战者在 DBDH 问题上的整体优势可以表示如下:

$$\begin{aligned} & \frac{1}{2} \Pr[\rho' = \rho | \mu = 0] + \frac{1}{2} \Pr[\rho' = \rho | \mu = 1] - \frac{1}{2} \\ &= \frac{1}{2} \times \left(\frac{1}{2} + \epsilon \right) + \frac{1}{2} \times \frac{1}{2} - \frac{1}{2} = \frac{1}{2} \epsilon \end{aligned} \quad (4-26)$$

4.6 性能分析

本节将对 CP-DK-ABE 方案的性能进行分析, 特别地, 对本章提出的属性基加密方案和一些已有的属性基加密方案在特征以及计算和存储开销方面进行详细比较。

如表 4-5 所示, 本章提出的 CP-DK-ABE 方案与现有的七个 ABE 方案从去中心化、访问结构、安全模型、困难假设、群的阶数以及密文的可编程性

等方面进行了详细对比。比较结果显示，文献[108][114]和本章中的属性基加密方案是去中心化的，其他的方案中均需要一个可信的中心节点参与主密钥的管理。此外，仅有文献[110]和本章提出的方案支持树状访问结构，与“AND 门”^[111]和 LSSS^{[108],[111]-[115]}访问结构相比，树状访问结构能够表达更加复杂的访问策略。进一步地，文献[112][114]中的 ABE 方案是基于合数阶群构造的，文献[113][115]中的方案安全性依赖于随机预言机模型。然而，本章提出的 CP-DK-ABE 方案是基于素数阶群构造的，它的安全性在标准模型下被规约到 DLDH 和 DBDH 假设之上。值得注意的是，与其它属性基加密方案相比，仅有本章提出的方案支持可编程密文表示形式。

表 4-5 现有方案与本章提出的 CP-DK-ABE 之间的比较

方案	去中心化	访问结构	安全模型	困难假设	群阶数	可编程密文
文献[108]	✓	AND	标准模型	q -PBDHE	素数	✗
文献[110]	✗	LSSS	随机预言机模型	CDH	素数	✗
文献[111]	✗	LSSS	标准模型	DBDH	素数	✗
文献[112]	✗	LSSS	标准模型	子群判定	合数	✗
文献[113]	✗	LSSS	随机预言机模型	CDH DBDH	素数	✗
文献[114]	✓	LSSS	标准模型	子群判定	合数	✗
文献[115]	✗	Tree	标准模型	q -BDHE l -SDH	素数	✗
本章方案	✓	Tree	标准模型	DLDH DBDH	素数	✓

令 $E(\mathbb{G})$ 和 $E(\mathbb{G}_T)$ 分别表示群 \mathbb{G} 和 \mathbb{G}_T 中的指数运算， $M(\mathbb{G}_T)$ 表示 \mathbb{G}_T 中的乘法运算， B 为双线性配对运算。令 $I(\mathbb{G})$ ， $I(\mathbb{G}_T)$ 和 $I(\mathbb{Z}_p^*)$ 分别表示 \mathbb{G} ， \mathbb{G}_T 和 \mathbb{Z}_p^* 中元素长度。此外， k 为系统中所有属性总数， n 是完全节点数目， m 为访问策略中的属性数目。注意，由于 \mathbb{Z}_p^* 中的指数和乘法运算、 \mathbb{G} 中的乘法运算以及哈希运算相比其他运算更为高效，本节在对性能进行理论分析时忽略上述运算的计算开销。

在上表 4-6 中，本节对文献[108][114][116]中的去中心化属性基加密方案与本章提出的 CP-DK-ABE 方案的计算开销进行了详细对比。如表所示，在本章方案中，不论是加密还是解密过程，计算开销均为 $O(m)$ ，这意味着方案加解密计算开销与访问策略中属性个数线性相关。文献[116]中的加密算法计算开销与 CP-DK-ABE 类似，但是解密计算为 $O(m+n)$ 大于本章方案相应计算

开销。文献[108]中的加密和解密计算开销也是 $O(m+n)$ ，文献[114]中的加密和解密计算开销则为 $O(m)$ ，但是要比本章提出的方案开销更大。

表 4-6 各方案之间计算开销对比

方案	加密	解密
文献[108]	$(n+3m) \cdot E(\mathbb{G}) + n \cdot E(\mathbb{G}_T)$	$(n+2m) \cdot B + (3+m) \cdot E(\mathbb{G}_T)$
文献[114]	$(4m+1) \cdot E(\mathbb{G}) + B$ $2m \cdot E(\mathbb{G}_T) + M(\mathbb{G}_T)$	$4m \cdot B + m \cdot E(\mathbb{G}_T) + (3m-1) \cdot M(\mathbb{G}_T)$
文献[116]	$(2m+1) \cdot E(\mathbb{G}) +$ $E(\mathbb{G}_T) + B + M(\mathbb{G}_T)$	$4m \cdot B + 2m \cdot E(\mathbb{G}_T) + (2n+2) \cdot E(\mathbb{G})$
本章方案	$(2m+2) \cdot E(\mathbb{G}) + M(\mathbb{G}_T)$	$(2m+1) \cdot B + (m-1) \cdot E(\mathbb{G}_T)$

接下来本节讨论上述 DABE 方案之间存储开销方面的对比。如表 4-7 所示，在本章提出的属性加加密方案中，公钥尺寸仅与完全节点个数相关，文献[114]中的方案存储开销与属性数目 k 线性相关。考虑在实际去中心化属性基加密方案，属性数目远大于完全节点数目，即 $k > n$ ，因此，本章方案存储开销优于文献[114]中的方案。此外，文献[108][116]中方案的公钥存储复杂度均为 $O(m+n)$ ，另一方面，文献[114]中密文存储开销与完全节点上数目线性相关，文献[116]与本章提出的方案的密文存储复杂度与访问策略中属性数目线性相关，文献[108]中的密文存储复杂度与访问策略中属性数目以及完全节点数目线性相关。综上，本章提出的 CP-DK-ABE 方案具有更低的存储开销。

表 4-7 各方案之间存储开销对比

方案	公钥	密文
文献[108]	$(2n+k) \cdot l(\mathbb{G})$	$(2m+n) \cdot l(\mathbb{G}) + 2 \cdot l(\mathbb{G}_T)$
文献[114]	$3k \cdot l(\mathbb{G}) + k \cdot l(\mathbb{G}_T)$	$(3n+1) \cdot l(\mathbb{G}) + (n+1) \cdot l(\mathbb{G}_T)$
文献[116]	$(3n+3k) \cdot l(\mathbb{G})$	$2m \cdot l(\mathbb{G}) + 1 \cdot l(\mathbb{G}_T) + m \cdot l(\mathbb{Z}_p^*)$
本章方案	$2 \cdot l(\mathbb{G}) + l(\mathbb{G}_T) + n \cdot l(\mathbb{Z}_p^*)$	$(2m+n) \cdot l(\mathbb{G}) + 1 \cdot l(\mathbb{G}_T)$

为了对本章中提出的 CP-DK-ABE 方案进行性能分析，接下来利用 JPBC 库来进行仿真实验，实验环境与 3.5 节中的描述一致，开发工具为 IntelliJ IDEA 2020.3.3。令访问策略中属性个数为 25，实验结果取算法被单独执行 100 次耗时的算数平均值。表 4-8 展示了本章提出的 CP-DK-ABE 中加密和解密算法运行时间，从表中不难发现，加解密运算时间均随访问策略中属性数目的增加而增长，这与表 4-6 中理论分析相一致。

表 4-8 CP-DK-ABE 中加解密算法运行时间 (单位: 毫秒)

属性数目	5	10	15	20	25
加密时间	128.43	259.38	352.27	497.18	616.06
解密时间	84.12	181.35	241.59	355.75	451.94

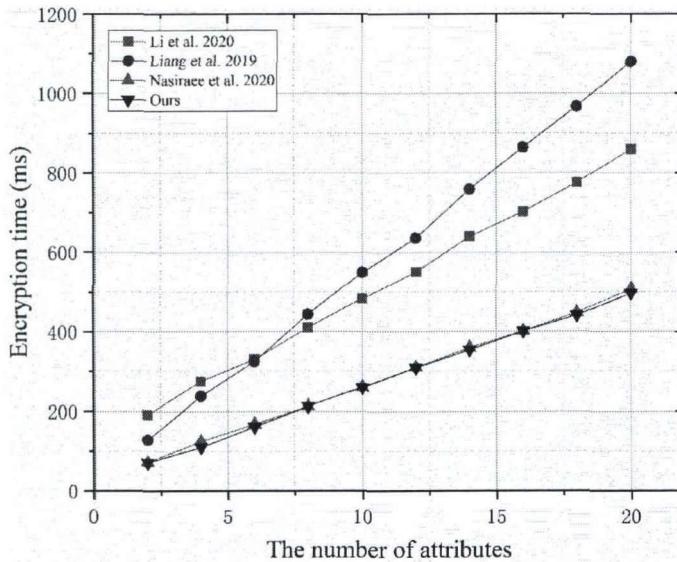


图 4-7 加密算法运行时间对比

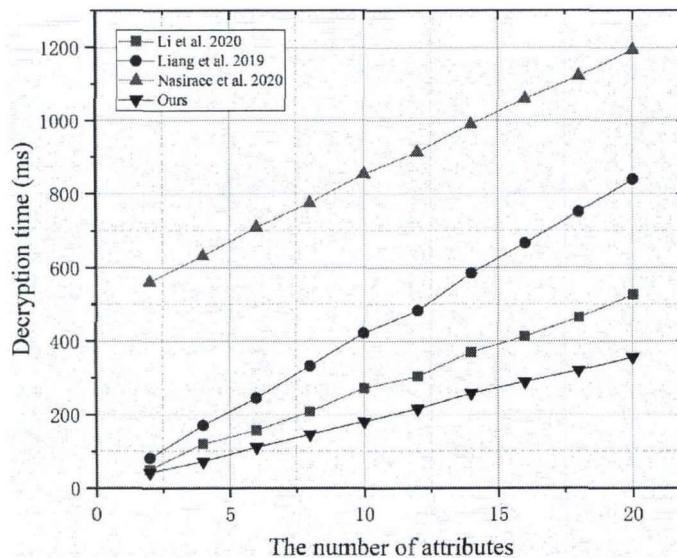


图 4-8 解密算法运行时间对比

进一步地，图 4-7 和图 4-8 分别给出文献[108][114][116]与本章提出方案在加密和解密算法计算开销的对比。为了表述方便，在评估加解密开销时，对各方案中完全节点个数进行固定，不失一般性，实验中设定完全节点数目为 $n=10$ 。图 4-6 表明，所有 DABE 方案的加密算法运行时间均随着访问策略中属性个数的增加而增长，特别地，文献[108]中的去中心化属性基加密方案的加密时间与本章提出的 CP-DK-ABE 方案相近，与其他相关方案加密运

算相比，上述两个 DABE 方案加密算法耗时更短。图 4-8 展示了上述 DABE 方案之间解密算法时间开销，从该图中不难发现，本章提出的 CP-DK-ABE 方案的解密时间开销明显优于其他方案。总体上讲，实验结果表明，本章提出属性基加密方案具有高效的加密和解密算法。

4.7 本章小结

本章提出了一种面向属性基加密的去中心化密钥管理与脚本化密文机制，通过融合两类加法同态的安全多方计算技术，设计了 CP-DK-ABE 及智能合约化的方案构造。同时，通过解密过程中的密钥查询、密文逻辑等操作扩展区块链脚本指令系统，在加密过程中实现针对密文策略中复杂逻辑的密文脚本化，实现了各方对主密钥的共同管理和用户私钥的协同生成、以及脚本解释器对脚本化密文的自动化解密。此外，证明了设计的密钥生成算法能够保护完整节点持有的密钥不被泄露，并且在 DBDH 假设下证明了所提出的 CP-DK-ABE 方案的语义安全性。

5 零知识对偶集合成员关系证明协议

针对智能合约中动态群组成员身份认证及判定过程中用户隐私泄露问题，本章将研究面向合约成员身份认证的零知识对偶集合成员关系证明（Zero-knowledge Dual-membership Proof, ZKDMF）协议设计。通过构造聚合函数实现将子集压缩到密码空间元素的表示方法，通过并引入安全聚合函数（Security Aggregation Function, SAF）的概念，在子集压缩过程中将元素与集合间属于和不属于的判定问题转变为元素删除和插入下的聚合函数求解问题，进而设计基于正负成员关系判定架构下的 ZKDMF 协议，在保证待测元素隐私的前提下实现对正集合关系（ \in ）和负集合成员关系（ \notin ）的同时判定。

5.1 研究动机

随着互联网开放性的增强，密码技术被越来越广泛的应用到互联网中，用于保障核心数据的隐私和业务安全，已经成为一些领域的核心价值。同时，为了提高应用业务系统的可靠性和安全性，密码学中的一些数学基础问题也越来越受到重视。集合论作为研究集合的数学理论，在当今动态性、多样化需求的互联网环境下具有重要的研究价值。

判定一个元素与集合之间的所属关系问题被称为集合关系判定，它是集合论的基础问题，也是互联网中一个常见问题，被广泛应用于电子竞拍、黑白名单机制、匿名证书系统等许多领域^{[117]-[119]}。例如，作为一种常见的计算机技术，黑名单机制要求除了明确规定实体之外，允许系统中其他任何用户进入。其本质在于判定用户是否属于黑名单集合。

解决集合成员关系判定问题最朴素的方法是拿待测元素与集合中的所有元素进行逐个对比，然而，这种方式十分低效，不符合互联网中大规模集合成员关系判定的需求。为了提高效率，布隆滤波器和密码学累加器两类主要技术被应用到集合成员关系判定中。其基本思路为：首先，对集合进行随机化压缩表示；随后，通过元素与集合表示之间的关系，判定元素是否属于集合。具体来讲，基于布隆滤波器的集合关系判定方案首先通过 k 个不同的哈希函数，将集合中的每个元素映射到数组中的 k 个位置，并将这些位置的值设置为 1，其他位设置为 0，该二进制数组称为集合的表示。验证阶段，首先将待测元素映射到数组的 k 个点，通过这些点处的 0/1 情况判定元素与集合的关系，若有一个为 0，则该元素一定不在集合中；若均为 1，则待测元素很可能在集合中。基于累加器的集合关系判定过程被描述为：累加器将集合中

的所有元素累加到一个累加值进行表示，并为元素生成对应的证据。判定阶段，一个验证者可以通过证据和公开的累加值之间的代数关系，判定元素与集合的所属关系。

通过上述技术，集合成员关系判定从待测元素与集合中所有元素的逐个对比转化为了元素与集合的表示之间的一次判定，大大提高了判定效率。此外，为了在集合关系判定过程中保护用户隐私，零知识证明技术^{[120]-[122]}被考虑进来^{[77][123]}。具有零知识特征的集合成员关系判定方案允许在待测元素信息不泄露的情况下证明其属于集合^[124]。具有该特征的集合关系判定方案具有广阔的应用前景，例如，在金融监管合约中，用户通过智能合约可以向银行证明自己属于一个开放用户集，而不需要透露自己的具体身份信息，这样就可以降低用户身份隐私的泄露风险。

尽管关于集合成员关系判定及相关问题研究已经取得了众多令人瞩目的成果，但是与复杂应用需求以及大规模、动态的网络所需安全性相比，现有研究仍然存在一些挑战。例如，虽然现有集合表示已经具有一定的压缩功能，但更加细致且安全的集合紧凑表示问题仍需进一步研究，以适应大规模的智能合约应用需求。此外，为了进一步保护隐私和提高判定效率，支持保护隐私且高效判定的协议构造研究仍是一种挑战。针对上述挑战，本章将构造一种新的安全集合表示形式，并在此基础之上设计同时支持正集合关系和负集合成员关系判定的零知识证明协议。

5.2 子集的安全表示

本章重点在于构造一个新的零知识集合成员关系证明协议，用于对偶集合成员关系的判定。构造上述协议的基础在于实现子集的安全表示，这种子集表示形式可被用于正集合关系(\in)和负集合成员关系(\notin)的判定。集合成员关系判定的基本思想是通过聚合函数对子集进行密码学安全表示。

定义 5-1 (聚合函数) 令 $PK \subseteq \mathbb{G}$ 为公钥空间，集合 $U = \{e_1, e_2, \dots, e_n\}$ ， $\mathcal{P}(U)$ 表示集合 U 的幂集，聚合函数 *Aggregator* 是确定性多项式时间算法，满足以下等式，

$$\text{Aggregator}(\text{mpk}, S) = R_S \quad (5-1)$$

其中，公钥 $\text{mpk} \in PK$ ， S 是集合 U 的一个子集， R_S 为群 \mathbb{G} 中的一个避免猜测的足够随机元素。

在上述定义中，由于公钥 mpk 被用于函数的输入，因此聚合函数是公开可计算的且未对集合 U 以及子集 S 的大小做任何额外的限制。此外，在上述

定义中，仅对聚合函数的功能需求进行了描述，而与安全性无关。为了进一步描述聚合函数的安全性，接下来对聚合函数的安全特性进行定义。

定义 5-2（安全聚合函数） 给定子 $S \subseteq U$ ，一个安全的聚合函数可以将子集 S 压缩成一个随机元素并满足以下两个安全性质：

1) 对于正确输入，聚合函数能够轻易计算聚合值：

- ① 对于一个元素 $e \notin S$ ，将其添加到集合 S 中；
- ② 对于一个元素 $e \in S$ ，将其从集合 S 中删除。

2) 对于异常输入，聚合函数难以计算聚合值：

- ① 对于一个元素 $e \notin S$ ，将其从集合 S 中删除；
- ② 对于一个元素 $e \in S$ ，将其添加到集合 S 中。

毫无疑问，设计一个安全的聚合函数是一项重要挑战，特别是将任意大小的子集压缩到密码空间中的一个随机元素。接下来将通过以下四个步骤，展示本文是如何解决这个具有挑战性的问题，为了便于解释，本文以图 5-1 为例展示该方法的示意图。此外，该方法还将用于后续两个聚合函数构造，具体设计思路如下：

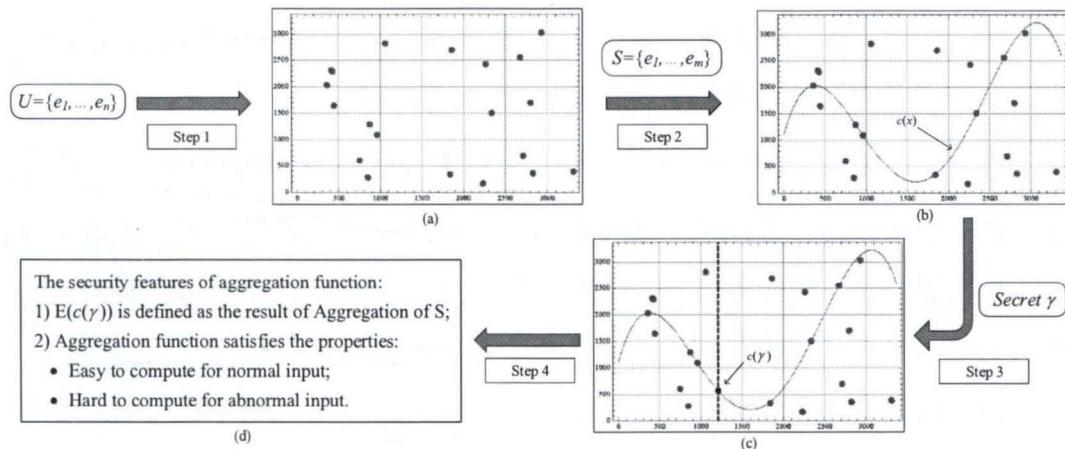


图 5-1 安全的聚合函数设计路线

步骤一：对于任意的集合 $U = \{e_1, \dots, e_n\}$ ，该步骤首先将集合 U 中的元素 e_i 映射成密码学空间中的点 v_i 进行表示（其中 $i \in [1, n]$ ），并将该点的部分信息作为公钥进行公开。如图 5-1(a)所示，红色圆点即为集合 $U = \{e_1, \dots, e_n\}$ 中所有元素的映射结果。

步骤二：给定集合 U 的一个子集 $S \subseteq U$ ，构造过子集 S 中所有元素映射的随机点 v_i 的曲线 $c(x)$ ，如图 5-1(b)中蓝色曲线所示。

步骤三：该步骤首先选取一个秘密值 γ 并定义聚合函数的输出作为曲线上点 $c(\gamma)$ 的封装，表示为 $Aggregator(mpk, S) \rightarrow E(c(\gamma))$ ，其中 mpk 是公钥密

码参数， $E(\cdot)$ 表示密码学封装函数，详情参见图 5-1(c)中的蓝色虚线与曲线 $c(x)$ 的交点。

步骤四：该步骤将定义聚合函数的两个安全特征来保证聚合函数在恶意攻击下的安全性：首先，定义 $E(c(\gamma))$ 为集合 S 聚合结果；其次，聚合函数满足以下两个条件，1)对于正常输入，聚合函数输出聚合结果是容易的；2)对于无效出入，聚合函数输出聚合结果是困难的，具体描述见图 5-1(d)。

最后，本文希望使用上述聚合思想来构造集合成员关系判定协议，聚合点 $E(c(\gamma))$ 的隐私性和随机性将为判定协议的安全性提供保证。此外，聚合函数的压缩特性可以保证聚合结果的规模保持在常数大小范围内。基于上述思路，本章将设计两个聚合函数，即 ZerosAggr 和 PolesAggr，分别实现对正集合成员 ($e_i \in S$) 和负集合成员 ($e_i \notin S$) 关系的判定。接下来对聚合函数构造、实现以及安全性给出具体介绍。

5.2.1 基于零点聚合的子集安全表示

对于给定的子集 $S = \{e_1, \dots, e_m\} \subseteq U$ ，接下来的目标就是寻求一种方式将集合 S 中所有元素的哈希值 $\text{hash}(e_i) \rightarrow x_i$ 转换成二维空间中的点 (x_i, y_i) 。为了实现上述目标，本章首先定义一个 $m+1$ 阶多项式如下：

$$f_S(x) = x(x + x_1) \cdots (x + x_m) = x \prod_{e_i \in S} (x + x_i) \pmod{p} \quad (5-2)$$

从上式中不难发现，不同的子集 S 对应于不同的多项式表达形式。通过上述多项式，子集 S 中的任意元素 e_i 可被转换成 $(x_i, y_i) = (x_i, f_S(x_i))$ 进行表示，并且当 x_i 未知时，任意敌手均不能获取关于其映射点的信息。

为了保证上述多项式输入 x 的隐私性以及 $f_S(x)$ 的不可预测性，接下来将引入离散对数问题完善上述多项式的构造。首先，给定一个乘法循环群 \mathbb{G} ，其中 $g \in \mathbb{G}$ 为该群中的任意一个生成元，本文将 $(g^{x_i}, g^{f_S(x_i)})$ 来代替 $(x_i, f_S(x_i))$ 从而借助离散对数假设增强 x_i 的隐私性以及 $g^{f_S(x_i)}$ 的不可预测性，甚至在 g^{x_i} 公开的情况下，其安全性也能得到保证。

对于秘密值 γ 以及子集 $S \subseteq U$ ，接下来介绍聚合函数的定义，该聚合函数能将子集 S 通过多项式 $f_S(x)$ 转化成一个固定尺寸的值 $g^{f_S(\gamma)}$ 进行表示。由于子集 S 中所有元素的哈希值对应于多项式 $f_S(x)$ 中的零点，因此称之为基于零点的聚合函数，算法具体定义如下。

定义 5-3（基于零点的聚合函数） 给定一个子集 $S = \{e_1, e_2, \dots, e_m\} \subseteq U$ 以

及乘法循环群 \mathbb{G} ，若存在多项式时间算法 $ZerosAggr(\cdot)$ 使得输出如下，则该算法称为基于零点的聚合函数

$$G_S = ZerosAggr(mpk, S) = g^{f_S(\gamma)} = g^{\gamma \prod_{e_i \in S} (\gamma + x_i)} \quad (5-3)$$

在上式中，公共参数 $mpk = \{g_i = g^{r^i}\}_{i \in [1, U]}$ ， g 是群 \mathbb{G} 中的生成元， $x_i = \text{hash}(e_i)$ 。

为了快速实现基于零点的聚合，对于子集 $S \subseteq U$ ，其中 $|S| = t \leq m = |U|$ ，首先从公钥中提取相关信息 $\{g_i = g^{r^i}\}_{i \in [1, m]}$ ，接下来介绍一个快速的递归方式来实现基于零点的聚合函数 $ZerosAggr$ 。

对于给定的 $\{x_i = \text{hash}(e_i)\}_{e_i \in S}$ ，定义关于 x 的 t 阶多项式如下：

$$f_S(x) = x \cdot \prod_{e_i \in S} (x + x_i) = \sum_{k \in [0, t]} a_k x^{k+1} \pmod{p} \quad (5-4)$$

对于任意的 $k \in [0, t]$ ，可按照以下递归步骤计算上述多项式系数 $a_k \in \mathbb{Z}_p$ 。对于任意的 $j = 1, \dots, t$ 以及 $i = 0, 1, \dots, j$ ，令 $a_i^{(j)}$ 表示系数 a_i 在第 j 轮循环中的值，在每次循环中添加一个新的数值 x_k ，初始化系数被定义为 $a_0^{(0)} = 1$ ，则多项式系数可以被计算如下：

$$\begin{cases} a_k^{(k)} = a_{k-1}^{(k-1)} & (k \geq 1) \\ a_{k-1}^{(k)} = a_{k-2}^{(k-1)} + a_{k-1}^{(k-1)} \cdot x_k & (k \geq 2) \\ \dots & \dots \\ a_1^{(k)} = a_0^{(k-1)} + a_1^{(k-1)} \cdot x_k & (k \geq 2) \\ a_0^{(k)} = a_0^{(k-1)} \cdot x_k & (k \geq 1) \end{cases} \quad (5-5)$$

经过 t 轮计算，可以输出 $a_0 = a_0^{(t)}, \dots, a_t = a_t^{(t)}$ ，基于零点的聚合函数 $ZerosAggr$ 的输出为 $G_S = g^{f_S(\gamma)}$ 。事实上，尽管 γ 是未知密钥，然而依然可以按照以下方式通过公钥 (g_1, \dots, g_m) 计算出聚合值 G_S

$$G_S = g^{f_S(\gamma)} = g^{\gamma \prod_{e_i \in S} (\gamma + x_i)} = g^{\sum_{k=1}^{t+1} a_{k-1} \cdot \gamma^k} = \prod_{k=1}^{t+1} g_k^{a_{k-1}} \quad (5-6)$$

值得注意的是，在上式中 $t < m$ ，当集合 $S = \emptyset$ 且 $t = 0$ 时，基于零点的聚合算法的输出为 $ZerosAggr(mpk, S) = g_1$ 。

基于上述递归过程，接下来给出基于零点聚合的快速算法。在算法 5-1 中，计算多项式 $f_S(x)$ 的系数 (a_0, \dots, a_t) 过程被描述成一个两重循环结构，其中 $A[1] = a_0, \dots, A[t+1] = a_t$ 。最后，算法输出是通过 $g_k^{a_{k-1}} = \text{pow}(g_k, A[k])$ 的累加计算得到的，其中 $k = 1, \dots, n+1$ 。在上述过程中，函数 $\text{pow}(\cdot)$ 被调用来计算群 \mathbb{G} 中元素的幂，也就是 $\text{pow}(g_k, A[k]) = g_k^{A[k]} = g_k^{a_{k-1}}$ 。

算法 5-1 基于零点快速聚合算法 (ZerosAggr)

输入: 公钥 mpk , 子集 S

输出: 子集 S 的零点聚合值

1. $A[1]=1$
 2. **for** $i=1$ to n **do**
 3. $A[i+1]=A[i]$
 4. **for** $j=i$ to 2 **do**
 5. $A[j]=A[j-1]+A[j]\cdot x_i$
 6. **end for**
 7. $A[1]=A[1]\cdot x_i$
 8. **end for**
 9. $sum=1$
 10. **for** $k=1$ to $n+1$ **do**
 11. $sum=sum\cdot pow(g_k, A[k])$
 12. **end for**
 13. **Return** sum
-

结合上述基于零点聚合函数的定义, 接下来将对其安全性给出进一步的描述。给定一个多项式 $f(x)$, 使用简单的多项式除法以便从该多项式中删除某个零点 x_i , 即 $f(x)/(x+x_i)$ 。此时会出现两种情况: 第一种情况 x_i 不是 $f(x)$ 的根, 因此定义 $f(x)=(x+x_i)q(x)+r(x)$ 和 $\frac{f(x)}{x+x_i}=q(x)+\frac{r(x)}{x+x_i}$, 其中 $q(x)$ 和 $r(x)$ 是两个多项式, 余数 $r(x)$ 不等于 0; 第二种情况 x_i 是 $f(x)$ 的根, 则余数等于 0。因此, 可将集合成员之间关系的判定问题转化为判定余数 $r(x)$ 是否等于 0 问题上。

基于上述讨论, 本文将集合 S 与元素 e_i 的减法表示为 $S_-=S\setminus\{e_i\}$, 相应的聚合函数减法被描述如下:

$$G_{S_-} = G_{S\setminus\{e_i\}} = g^{\prod_{e_k \in S \setminus \{e_i\}} (r+x_k)} = g^{\frac{f_S(x)}{r+x_i}} \quad (5-7)$$

在上式中, 令 $\frac{f_S(x)}{x+x_i}=q(x)+\frac{r(x)}{x+x_i}$, 则对于 $e_i \in S$ 和 $e_i \notin S$ 有以下两种情况:

- 1) 当 $e_i \in S$ 时, 则有 $(x+x_i)|f_S(x)$ 且 $r(x)=0$, 因此有 $G_{S_-} = g^{q(r)}$;
- 2) 当 $e_i \notin S$ 时, 则 x_i 不是多项式 $f_S(x)$ 的根, 且 $r(x) \neq 0$, 因此则有

$$G_{S_-} = g^{q(r)} \cdot g^{\frac{r(x)}{r+x_i}}.$$

通过基于零点的聚合定义可知, 对于 $e_i \in S$, $S_-=S\setminus\{e_i\}$ 对应的聚合函数

G_{S_-} 可以轻松计算出来。若不通过基于零点的聚合函数来计算 G_{S_-} ，则必须能通过公钥 $mpk = \{g_i = g^{r^i}\}_{i \in [1, |U|]}$ 计算出 $g^{f_S(\gamma)/(r+x_i)}$ 。因此，可以继续将集合判定问题进一步地转化成一个计算性问题： G_{S_-} 能否在多项式时间内被计算出来。在以上讨论的基础上，通过多项式与单项式的带余除法可定义基于零点聚合函数的安全性。

定义 5-4（基于零点聚合函数的安全性） 给定一个元素 $e_i \in U$ ，一个关于集合 S 的基于零点聚合函数 $ZerosAggr(\cdot)$ 是安全的，当且仅当满足以下两个安全条件：

- 1) 对于任意元素 $e_i \in S$ ，计算 G_{S_-} 是容易的，其中 $S_- = S \setminus \{e_i\}$ ，也即是 $G_{S_-} = g^{f_{S_-}(\gamma)/(r+x_i)}$ 可在多项式时间内通过基于零点聚合函数计算，即有以下概率公式成立：

$$\Pr[ZerosAggr(mpk, S_-) = G_{S_-} \mid e_i \in S] = 1 \quad (5-8)$$

- 2) 对于任意元素 $e_i \notin S$ ，计算 G_{S_-} 是困难的。对于任意的概率多项式敌手 \mathcal{A} 而言，其成功计算出 $G_{S_-} = g^{f_{S_-}(\gamma)/(r+x_i)}$ 的概率都不超过一个可忽略的概率 ϵ ，即，

$$\Pr[\mathcal{A}(mpk, e_i, S) = G_{S_-} \mid e_i \notin S] < \epsilon \quad (5-9)$$

定理 5-1 在 t -SDH 假设下，基于零点的聚合函数 $ZerosAggr(\cdot)$ 满足定义 5-4 中的安全性。

证明：在聚合函数 $ZerosAggr(\cdot)$ 中，令 $g = G$ 和 $\gamma = \alpha$ ，使得 $g_i = g^{r^i} = G^{\alpha^i}$ 以及公钥 $mpk = \{g_i\}_{i \in [1, t]}$ ，其中 α 是一个秘密值， $t \geq |U|$ 。对于任意的元素 $e_i \in U$ 以及子集 $S \subseteq U$ ($|S| = m$)，有以下两种情形：

对于任意的元素 $e_i \in S$ ，首先通过哈希函数 $hash$ 将 e_i 映射到 \mathbb{Z}_p 中， $hash(e_i) \rightarrow x_i \in \mathbb{Z}_p$ ，由多项式 $f_S(x)$ 的定义可知 $(x + x_i)$ 能够整除多项式 $f_S(x)$ ，即 $(x + x_i) \mid f_S(x)$ 。因此可以定义 $f_{S_-}(x) = x \cdot \prod_{e_k \in S, e_k \neq e_i} (x + x_k) = \sum_{i=0}^{m-1} c_i x^{i+1}$ ，此时 G_{S_-} 可被计算如下：

$$G_{S_-} = g^{f_{S_-}(\gamma)} = g^{\sum_{i=1}^{m-1} c_i \gamma^{i+1}} = \prod_{i=1}^m (G^{\alpha^i})^{c_i-1} \quad (5-10)$$

对于任意的元素 $e_i \notin S$ ，若存在一个多项式时间敌手 \mathcal{A} 可以在 $(x + x_i) \nmid f_{S_-}(x)$ 的情况下计算出 $G_{S_-} = g^{f_{S_-}(\gamma)/(r+x_i)}$ ，则可以构造一个概率多项式

算法来解决 t -SDH 问题，具体构造如下：首先令 $\frac{f_s(x)}{x+x_i} = q(x) + \frac{r(x)}{x+x_i}$ ，其中

$q(x) = \sum_{i=0}^m d_i x^i$ 和 $r(x) = r$ 是两个已知多项式。由于 $G^{f_s(r)/(r+x_i)} = G^{q(r)} \cdot G^{r/(r+x_i)}$

以及 $G^{q(r)} = \prod_{i=0}^m (G^{\alpha^i})^{d_i}$ 也是已知多项式，因此可以计算

$$G^{1/(\alpha+x_i)} = (G_{s_i} / \prod_{i=0}^m (G^{\alpha^i})^{d_i})^{1/r} \quad (5-11)$$

这意味着可以从公钥中伪造出一个有效的 $(x_i, G^{1/(\alpha+x_i)})$ ，然而这与 t -SDH 假设相矛盾，因此在 $e_i \notin S$ 的情况下敌手计算 G_{s_i} 是困难的。

综上，聚合函数 $ZerosAggr(\cdot)$ 在 t -SDH 假设下是安全的，定理 5-1 得证。

5.2.2 基于极点聚合的子集安全表示

上节已经介绍了通过零点聚合的方式来进行正集合关系的判定 (\in)，但是上述方式对负集合关系判定 (\notin) 是无效的。为解决负集合成员关系判定问题，本文希望构造一个与零点聚合相反的聚合函数，该聚合函数对于 $e_i \notin S$ 计算是容易的，而对于计算是困难的，满足上述性质的聚合函数称为基于极点的聚合函数。

与基于零点聚合函数类似，接下来本节将给出基于极点聚合函数的详细描述。对于给定的子集 $R = \{e_1, \dots, e_m\} \subseteq U$ ，接下来的目标是将子集中的元素映射到二维空间中的点进行表示。首先，利用哈希函数将 R 中元素映射到 \mathbb{Z}_p 中的值，即 $hash(e_i) \rightarrow x_i$ 。随后通过定义一个多项式 $g_R(x)$ ，计算 $y_i = g_R(x_i)$ 从而将元素 e_i 表示成二维空间中的点 (x_i, y_i) 。其中多项式 $g_R(x)$ 阶为 $m+1$ ，表示如下：

$$g_R(x) = \frac{1}{(x+x_1) \cdots (x+x_m)} = \frac{1}{\prod_{e_i \in R} (x+x_i)} \pmod{p} \quad (5-12)$$

对于一个未知且随机的输入 $x \in \mathbb{Z}_p$ ，则多项式 $g_R(x)$ 的输出是随机不可预测的。此外，当 p 足够大时，对于给定的随机值 x ，它与密码学哈希 $\{hash(e_i)\}_{i \in [1, m]}$ 产生碰撞的概率是可以忽略的，这意味着 $x+x_i = 0$ ，即多项式 $g_R(x)$ 分母为零的概率也是可以忽略的。基于多项式 $g_R(x)$ ，接下来将给出基于极点聚合函数的具体定义。

定义 5-5（基于极点的聚合函数） 给定一个子集 $R = \{e_1, e_2, \dots, e_m\} \subseteq U$ 以及乘法循环群 \mathbb{G} ， h 是群 \mathbb{G} 中的一个生成元。若存在多项式时间算法

*PolesAggr(·)*使得输出如下，则该算法称为基于零的聚合函数

$$H_R = PolesAggr(mpk, R) = h^{h_R(\gamma)} = h^{\prod_{e_i \in R} (\gamma + x_i)} \quad (5-13)$$

在上式中， γ 为秘密值， $x_i = \text{hash}(e_i)$ ，公共参数 $mpk = \{h_i = h^{\gamma/(r+x_i)}\}_{i \in [1, |U|]}$ 。

接下来介绍快速实现基于极点聚合算法的流程，首先对于子集 $R \subseteq U$ 从公钥中提取相关信息 $\{(x_i, h_i = h^{\gamma/(r+x_i)})\}_{e_i \in R}$ ，对于给定的基于极点的聚合值 H_i 和 H_j ，可以轻松地计算出它们的聚合结果

$$(H_j / H_i)^{\frac{1}{x_i - x_j}} = \left(H^{\gamma/(r+x_j)} / H^{\gamma/(r+x_i)} \right)^{\frac{1}{x_i - x_j}} = H^{\frac{1}{(r+x_i)(r+x_j)}}. \quad (5-14)$$

上式中的前提条件是 $x_i \neq x_j$ ，其中 $1/(x_i - x_j)$ 可通过扩展欧几里得算法获得。

接下来将上述方程扩展到多个值聚合的情况，首先定义一个新的符号 $B_{i,j}$ 用于表示 (H_i, \dots, H_j) 的聚合值，其中 $1 \leq i < j \leq t$ ，且 $B_{i,j} = h^{\gamma/\left(\prod_{k=i}^j (r+x_k)\right)}$ 。通过等式(5-14)可计算

$$\begin{aligned} B_{i,j+1} &= \left(B_{i,j} / B_{i+1,j+1} \right)^{\frac{1}{x_{j+1}-x_i}} = \left(H^{\gamma/\left(\prod_{k=i}^j (r+x_k)\right)} / H^{\gamma/\left(\prod_{k=i+1}^{j+1} (r+x_k)\right)} \right)^{\frac{1}{x_{j+1}-x_i}} \\ &= \left(H^{\gamma/(r+x_i)} / H^{\gamma/(r+x_{j+1})} \right)^{\frac{1}{x_{j+1}-x_i}} \prod_{k=i+1}^j \frac{1}{(r+x_k)} \\ &= H^{\frac{1}{(r+x_i)(r+x_{j+1})} \frac{1}{\prod_{k=i+1}^j (r+x_k)}} = H^{\frac{1}{\prod_{k=i}^j (r+x_k)}} \end{aligned} \quad (5-15)$$

最后的输出值 $H_R = B_{1,t}$ 可通过一系列的 $B_{i,j}$ 通过下面的递归过程计算出来，其中 $i \in [1, t-1]$ 以及 $j \in [1, t-i]$ 。

$$\begin{cases} B_{i,i} = H_i, & \forall i \in [1, t] \\ B_{i,j+1} = \left(B_{i,j} / B_{i+1,j+1} \right)^{\frac{1}{x_{j+1}-x_i}}, & i \in [1, t-1], j \in [1, t-i] \end{cases} \quad (5-16)$$

在上式中，对于任意的 $r \in [1, t]$ ， $B_{r,r}$ 表示 H_r 的初始输入。

接下来在算法 5-2 中上述递归过程进行描述，基于极点的快速聚合算法来源于公式(5-16)，在算法运行中，对于任意的 $i \in [1, t-1]$ 以及 $j \in [1, t-i]$ ， $B[j] = B_{j,j+i}$ 是通过两次循环结构计算而来的。在第一个循环中，输出若为 0，则表示存在两个相等的值 x_{j+i} 和 x_j ，从而算法报错，这意味着两个用户身份完全一致（或者发生了哈希碰撞，但这种情况不太可能发生，因为根据密码哈希函数的性质，对于任意的 $e_{j+i} \neq e_j$ 有 $\text{hash}(e_{j+i})$ 与 $\text{hash}(e_j)$ 相等的概率是可以忽略不计）。在该算法中，函数 *invert(·)* 和 *pow(·)* 被分别用于计算群 \mathbb{G} 中元素的乘法逆元以及元素的幂，也即是 $\text{temp}_2 = \text{invert}(\text{temp}_1) = 1/\text{temp}_1 \pmod p$ 以

及 $B[j] = \text{pow}(\text{temp}_3, \text{temp}_2) = \text{temp}_3^{\text{temp}_2}$ 。

算法 5-2 基于极点快速聚合算法 (*PolesAggr*)

输入：公钥 mpk ，子集 R

输出：子集 R 的极点聚合值

1. **for** $k=1$ to n **do**
2. $B[k] = H_k$
3. **end for**
4. **for** $i=1$ to $n-1$ **do**
5. **for** $j=i$ to $n-i$ **do**
6. **if** $x_{j+i} = x_j$ **then**
7. Return 0
8. **end if**
9. $\text{temp}_1 = x_j - x_{j+i}$
10. $\text{temp}_2 = \text{invert}(\text{temp}_1, p)$
11. $\text{temp}_3 = B[j+1]/B[j]$
12. $B[j] = \text{pow}(\text{temp}_3, \text{temp}_2)$
13. **end for**
14. **end for**
15. Return sum

结合基于极点聚合函数的定义，本文将集合 R 与元素 e_i 的加法表示为 $R_+ = R \cup \{e_i\}$ ，相应的聚合函数加法描述为

$$H_{R_+} = H_{R \cup \{e_i\}} = h^{g_R(\gamma) \cdot \frac{1}{\gamma+x_i}} = h^{\frac{1}{(\gamma+x_i) \prod_{e_k \in R} (\gamma+x_k)}} \quad (5-17)$$

在上式中有以下两种情况：

1) 当 $e_i \notin R$ 时，对于任意的 $e_k \in R$ 以及一个抗碰撞哈希函数 $\text{hash}(\cdot)$ ，则有

$$\text{hash}(e_i) \neq \text{hash}(e_k)，因此有 H_{R_+} = h^{g_{R_+}(\gamma)} = h^{\frac{1}{(\gamma+x_i) \prod_{e_k \in R} (\gamma+x_k)}}；$$

2) 当 $e_i \in S$ 时， $\text{hash}(e_i) = x_i$ 为多项式 $g_{R_+}(x)$ 的双重极点，即

$$H_{R_+} = h^{g_{R_+}(\gamma) \cdot \frac{1}{(\gamma+x_i)^2}} = h^{\frac{1}{(\gamma+x_i)^2 \prod_{e_k \in R \setminus \{e_i\}} (\gamma+x_k)}}。$$

通过基于极点聚合函数的定义可知，对于 $e_i \notin R$ ， $R_+ = R \cup \{e_i\}$ 对应的聚合函数值 H_{R_+} 可以轻松计算出来。若不通过基于极点的聚合函数来计算 H_{R_+} ，

则必须能通过公钥 $mpk = \{h_i = h^{1/(r+x_i)}\}_{i \in [1, |U|]}$ 计算出 $h^{\frac{g_{R_+}(x)-1}{(r+x_i)^2}}$ 。然而，计算该值是不可能的，这是由于此时分母上出现了零。在以上讨论的基础上，通过对多项式与单项式进行带余除法可定义基于极点聚合函数的安全性。

定义 5-6（基于极点聚合函数的安全性） 给定一个元素 $e_i \in U$ ，一个关于集合 $R \subseteq U$ 的基于极点聚合函数 $PolesAggr(\cdot)$ 是安全的，当且仅当满足以下两个条件：

- 1) 对于任意元素 $e_i \notin R$ ，计算 H_{R_+} 是容易的，其中 $R_+ = R \cup \{e_i\}$ ，也即是 $H_{R_+} = h^{g_R(r)/(r+x_i)}$ 可以在多项式时间内，通过基于极点聚合函数以压倒性的概率被计算出来，即，

$$\Pr[PolesAggr(mpk, R_+) = H_{R_+} | e_i \notin R] > 1 - \epsilon \quad (5-18)$$

- 2) 对于任意元素 $e_i \in R$ ，计算 H_{R_+} 是困难的。即对于任意概率多项式敌手 \mathcal{A} ，其成功计算 $H_{R_+} = h^{g_R(r)/(r+x_i)}$ 的概率都不超过一个可忽略的概率 ϵ ，

$$\Pr[\mathcal{A}(mpk, e_i, R) = H_{R_+} | e_i \in R] < \epsilon \quad (5-19)$$

定理 5-2 在 t -Strong Diffie-Hellman 假设下，聚合函数 $PolesAggr(\cdot)$ 满足定义 5-7 中的安全性。

证明：令 $(G, G^\alpha, G^{\alpha^2}, \dots, G^{\alpha^n}) \rightarrow (c, G^{1/(\alpha+c)})$ 为 t -SDH 问题的一个实例，接下来展示如何将上述实例转化到基于极点的聚合函数中。令集合 $U = \{e_1, \dots, e_n\}$ ，多项式 $f(x) = \prod_{i=1}^n (x + x_i)$ ，其中 $n \leq t$ 。随后，定义 $r = \alpha$ ，
 $h = G^{f(r)}$ 以及 $h_i = h^{1/(r+x_i)} = G^{f_i(r)}$ ，此时公钥可以表示为 $mpk = \{h_i\}_{i \in [1, n]}$ 。对于任意的 $e_i \in U$ ，则有 $f_i(x) = f(x) / (x + x_i) = \sum_{i=0}^{n-1} d_i x^i$ ，
 $G^{f_i(r)} = \prod_{i=0}^{n-1} (G^{\alpha^i})^{d_i}$ 。对于任意的元素 $e_i \in U$ 以及子集 $R \subseteq U$ ($|R| = m$)，有以下两种情形：

对于给定的子集 $R \subseteq U$ 以及任意元素 $e_i \notin R$ ，根据 $(x + x_i) \mid f(x)$ 定义

$$f(x) \cdot g_{R_+}(x) = \frac{f(x)}{\prod_{e_k \in R_+} (x + x_k)} = \prod_{e_k \in U \setminus (R \cup \{e_i\})} (x + x_k) = \sum_{i=0}^{n-m-1} c_i x^i \quad (5-20)$$

可以通过以下多项式计算出 H_{R_+} 的值，

$$H_{R_+} = h^{g_{R_+}(r)} = G^{f(x) \cdot g_{R_+}(x)} = G^{\sum_{i=0}^{n-m-1} c_i r^i} = \prod_{i=0}^{n-m-1} (G^{\alpha^i})^{c_i} \quad (5-21)$$

对于元素 $e_i \in R$ ，若存在一个敌手 \mathcal{A} 可以在 $f(x) \cdot g_R(x) = \prod_{e_k \in U \setminus R} (x + x_k)$ 的情况下计算出 $H_{R_+} = h^{g_{R_+}(r)} = G^{\frac{f(x) \cdot g_R(r)}{r+x_i}}$ ，其中 $f(x) \cdot g_R(x) = \prod_{e_k \in U \setminus R} (x + x_k)$ ，

则可以构造一个概率多项式时间算法来解决 t -SDH 问题，具体构造如下：令 $\frac{f(x) \cdot g_R(x)}{x + x_i} = q(x) + \frac{r(x)}{x + x_i}$ ，其中 $q(x) = \sum_{i=0}^{n-m-1} d_i x^i$ 和 $r(x) = r$ 是两个已知多项式。由于已知 $G^{\frac{f(\gamma)g_R(\gamma)}{\gamma+x_i}} = G^{q(\gamma)} \cdot G^{\frac{r}{\gamma+x_i}}$ 和 $G^{q(\gamma)} = \prod_{i=0}^{n-m-1} (G^{\alpha^i})^{d_i}$ ，因此可以计算

$$G^{1/(x+x_i)} = (H_{R_i} / \prod_{i=0}^{n-m-1} (G^{\alpha^i})^{d_i})^{1/r} \quad (5-22)$$

这意味着可以从公钥中伪造出一个有效的 $(x_i, G^{1/(x+x_i)})$ ，然而这与 t -SDH 假设相矛盾，因此对于 $e_i \in R$ 计算 H_{R_i} 是困难的。

综上，聚合函数 $PolesAggr(\cdot)$ 在 t -SDH 假设下是安全的，定理 5-2 得证。

5.3 集合关系安全证明

基于零点和极点的子集安全表示已经表现出了一定的集合成员关系判定能力，本节将进一步展示集合成员关系证明的更加简洁的形式，即当子集的两种安全表示方法同时使用时，不仅能够实现正集合关系判定而且支持负集合关系判定。接下来，本节首先给出一个集合关系谓词的简单定义。

定义 5-7（集合成员关系谓词） 对于 $U = \{e_1, e_2, \dots, e_N\}$ ，集合关系判定谓词是一个二值函数 $P: e_i \times S \rightarrow \{0, 1\}$ ，它的结果表示条件 $e_i \in S$ ($P_e(e_i, S) = 1$) 或 $e_i \notin S$ ($P_e(e_i, S) = 0$) 的正确与否，其中 e_i 是 U 中的元素， S 是 U 的子集。

当判定条件是 $e_i \in S$ 时， P_e 被称为正集合关系判定谓词，类似地，当判定条件是 $e_i \notin S$ 时， P_e 被称为负集合关系判定谓词。通过对集合成员关系谓词的描述，接下来本节将给出密码学正负集合成员关系判定问题的定义。

定义 5-8（集合成员关系安全判定） 令 $P(e, S)$ 表示集合成员关系判定谓词（正或负集合成员关系），给定元素 e 和子集 S ，一个概率多项式时间算法 $Verify_P$ 被称为安全的集合成员关系判定，当且仅当满足以下两个条件：

完整性：若 $P(e, S) = 1$ 成立，则验证者接受证明，即，

$$\Pr[Verify_P(R_e, C_S) = 1 | P(e, S) = 1] = 1 \quad (5-23)$$

完备性：若 $P(e, S) = 0$ 成立，则验证者接受证明的概率是可忽略的 ϵ ，即，

$$\Pr[Verify_P(R_e, C_S) = 1 | P(e, S) = 0] < \epsilon \quad (5-24)$$

其中， R_e 表示元素 e 的密码学表示， C_S 表示子集 S 的一个密码学化表示形式。

在讨论集合关系安全判定问题之前，首先观察这样一个事实：当且仅当有理多项式函数中的“零点”和“极点”相等时，二者可以互相抵消。正是

这个原理为 ZerosAggr 和 PolesAggr 聚合函数的结合提供了可能，这样就可以验证这两个聚合函数中是否存在相同的“零点”和“极点”。具体来说，通过双线性映射($e: \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$)使得在群 \mathbb{G} 中执行的两个聚合函数映射到 \mathbb{G}_T 中的一个元素，这样的结合将进一步应用到集合成员关系的判定中。

5.3.1 正集合成员安全判定

在正集合成员关系判定中包括 $Setup$, $Extract$ 以及 $Verify_e$ 三个算法，接下来将给这些算法的出具体描述。

$Setup(\mathbb{S}, U, S) \rightarrow (mpk, H_S)$: 该算法为概率性算法，用于生成公钥 mpk 和子集 S 的安全表示 H_S ，其中 $\mathbb{S} = \{p, \mathbb{G}, \mathbb{G}_T, e(\cdot, \cdot), g, h\}$ 是双线性群。令 $sk = \gamma \in \mathbb{Z}_p^*$ 为随机选取的秘密值， $n \in \mathbb{Z}^+$ 为正整数，则公钥可以表示为 $mpk = (g^\gamma, \dots, g^{\gamma^n})$ 。对于给定的任意集合 $S = \{e_1, \dots, e_m\}$ ，可通过基于极点的聚合函数计算其安全表示：

$$H_S \leftarrow PolesAggr(sk, S) = h^{g_S(\gamma)} \quad (5-25)$$

在上式中， $m < n$ ， $g_S(x) = 1 / \prod_{e_k \in S} (x + x_k)$ 以及 $x_k = \text{hash}(e_k)$ 。

$Extract(sk, e_i) = e(g, h)^{\frac{\gamma}{r+x_i}} \rightarrow W_i$: 该算法被用于生成元素 e_i 的证据，它的输入是元素 e_i 以及私钥 sk ，输出对应元素的证据 W_i 。

$Verify_e(W_i, H_S) \rightarrow (e_i \in S \text{ or } e_i \notin S)$: 该算法用于测试元素 e_i 是否属于子集 S ，其中 W_i 是待测元素 e_i 的证据， H_S 为子集 S 的安全表示。若 $e_i \in S$ ，则可计算 $S_- = S \setminus \{e_i\}$ 以及 S_- 对应的零点聚合值

$$G_{S_-} \leftarrow ZerosAggr(mpk, S_-) = g^{f_{S_-}(\gamma)} = g^{\gamma \prod_{e_k \in S_-} (\gamma + x_k) / (r + x_i)} \quad (5-26)$$

随后验证证据 W_i 是否与 $e(G_{S_-}, H_S)$ 相等，即，

$$W_i = ? e(G_{S_-}, H_S) \quad (5-27)$$

若以上两个等式均成立，则算法输出 $e_i \in S$ ；否则输出 $e_i \notin S$ 。

图 5-2 描述了正集合成员关系判定过程。首先对于给定的子集 S 通过基于极点的聚合函数表示成 $H_S \leftarrow PolesAggr(S)$ ，随后对于给定的元素 e_i 利用 $Extract(sk, e_i) \rightarrow W_i$ 生成对应的证据。若 $e_i \in S$ ，则可通过基于零点聚合函数计算 $S_- = S \setminus \{e_i\}$ 的安全表示 G_{S_-} ，随后通过在 H_S 和 G_{S_-} 之间执行双线性映射，

从而消去集合 S 和 S_- 中的相同元素。最后，将上述双线性映射结果与证据 W_i 之间进行匹配，实现最终的集合成员关系的判定。

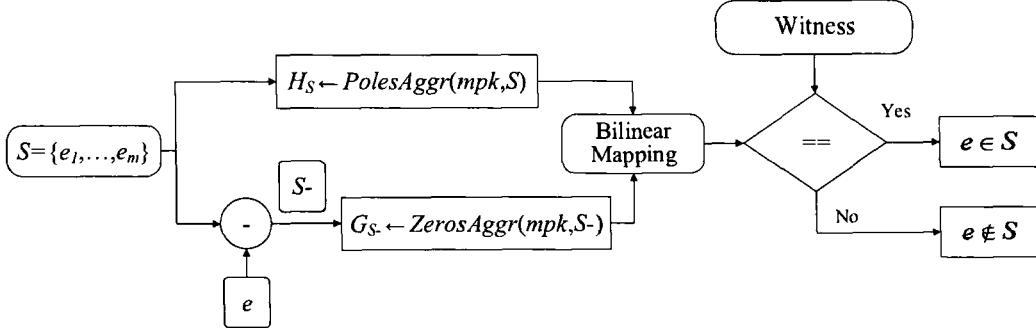


图 5-2 正集合成员关系判定示意图

定理 5-3 当基于零点的聚合函数满足定义的安全需求时，上述正集合成员关系判定构造是安全的。

证明：根据集合成员关系安全判定的定义，接下来证明上述正集合成员关系判定构造满足完整性和完备性。

完整性：当 $e_i \in S$ 时，对于任意的有效值 G_{S_-} ，下面等式总是成立。

$$e(G_{S_-}, H_S) = e(g^{f_{S_-}(\gamma)}, h^{g_S(\gamma)}) = e(g^{\frac{\prod_{e_k \in S} (\gamma + x_k)}{\gamma + x_i}}, h^{\frac{1}{\prod_{e_k \in S} (\gamma + x_k)}}) = e(g, h)^{\frac{\gamma}{\gamma + x_i}} = W_i \quad (5-28)$$

完备性：由该定理的前提条件，假设 $ZerosAggr$ 是一个安全的基于零点聚合，这意味着对于任意的概率多项式时间敌手 \mathcal{A} 以及给定的元素 $e_i^* \notin S$ ，则有

$$\Pr[\mathcal{A}(mpk, e_i^*, S) = G_{S_-} | e_i^* \notin S] < \epsilon \quad (5-29)$$

接下来证明验证者接受验证的概率是可忽略的，即， $\Pr[Verify_\epsilon(W_i, H_S) = 1] < \epsilon$ ，

$W_i = e(g, h)^{\frac{\gamma}{\gamma + x_i}}$ 和 $H_S = h^{\frac{1}{\prod_{e_k \in S} (\gamma + x_k)}}$ 是两个确定且不变的值， $x_i^* = \text{hash}(e_i^*)$ 。这意味着敌手 \mathcal{A} 可以伪造一个有效的 G^* 使得下列等式成立。

$$\Pr[Verify_\epsilon(W_i, H_S) = 1] = \Pr[e(g, h)^{\gamma/(y+x_i^*)} = e(G^*, h^{\prod_{e_k \in S} (\gamma + x_k)})] \quad (5-30)$$

不失一般性，令 $G^* = g^z$ ，则有

$$\begin{aligned}
 \Pr[Verify_\epsilon(W_i, H_S) = 1] &= \Pr[e(g, h)^{\frac{\gamma}{\gamma + x_i^*}} = e(G^*, h^{\frac{1}{\prod_{e_k \in S} (\gamma + x_k)}})] \\
 &= \Pr \left[\begin{array}{l} e(g, h) = e(g, h)^{\frac{z(\gamma + x_i^*)}{\gamma \prod_{e_k \in S} (\gamma + x_k)}} \\ g^z \leftarrow \mathcal{A}(mpk, e_i^*, S) \end{array} \right] \cdot \Pr[g^z \leftarrow \mathcal{A}(mpk, e_i^*, S)] \\
 &= \Pr \left[\mathcal{A}(mpk, e_i^*, S) \rightarrow g^z = g^{\frac{\prod_{e_k \in S} (\gamma + x_k)}{\gamma + x_i^*}} \right] \\
 &= \Pr[\mathcal{A}(mpk, e_i^*, S) = G_{S_-} | e_i^* \notin S] < \epsilon
 \end{aligned} \quad (5-31)$$

在上式中要求关系 $\frac{z(\gamma + x_i^*)}{\gamma \prod_{e_k \in S} (\gamma + x_k)} = 1$ 成立，因此有

$$\Pr \left[e(g, h) = e(g, h)^{\frac{z(\gamma + x_i^*)}{\gamma \prod_{e_k \in S} (\gamma + x_k)}} \right] = 1 \quad (5-32)$$

在这种情况下， $z = \gamma \frac{\prod_{e_k \in S} (\gamma + x_k)}{\gamma + x_i^*}$ 以及 $G_S = g^z$ 。这意味着敌手 \mathcal{A} 攻破安全的正集合成员关系判定的优势与攻破安全的基于零点聚合函数的优势相等，因此，在基于零点聚合函数安全的前提下，正集合成员关系判定也是安全的，该定理得证。

5.3.2 负集合成员安全判定

与正集合成员关系判定构造类似，负集合成员关系判定过程同样包含 $Setup$ ， $Extract$ 以及 $Verify_\epsilon$ 三个算法，接下来将对这些算法给出具体描述。

$Setup(\mathbb{S}, U, S) \rightarrow (mpk, G_S)$ ：该算法为概率性算法，用于生成公钥 mpk 和子集 S 的安全表示 G_S ，其中 $\mathbb{S} = \{p, \mathbb{G}, \mathbb{G}_T, e(\cdot, \cdot), g, h\}$ 是双线性群阶数 p 为素数。令 $sk = \gamma \in \mathbb{Z}_p^*$ 为秘密值， $n \in \mathbb{Z}^+$ 为正整数，则公钥可以表示为 $mpk = (h^{V(\gamma+x_1)}, \dots, h^{V(\gamma+x_n)})$ 。对于给定的任意集合 $S = \{e_1, \dots, e_m\}$ ，可通过基于零点聚合函数计算其安全表示：

$$G_S \leftarrow ZerosAggr(sk, S) = g^{f_S(\gamma)} \quad (5-33)$$

在上式中， $m < n$ ， $f_S(\gamma) = \gamma \cdot \prod_{e_k \in S} (x + x_k)$ 以及 $x_k = \text{hash}(e_k)$ 。

$Extract(sk, e_i) = e(g, h)^{\frac{\gamma}{x_i}} \rightarrow W_i$ ：该算法被用于生成元素 e_i 的证据，它的输入是元素 e_i 以及私钥 sk ，输出对应元素的证据 W_i 。

$Verify_\epsilon(W_i, H_s) \rightarrow (e_i \in S \text{ or } e_i \notin S)$ ：该算法用于测试元素 e_i 是否属于子集 S ，其中 W_i 是待测元素 e_i 的证据， G_S 为子集 S 的基于零点的聚合值。若 $e_i \notin S$ ，则可计算 $S_+ = S \cup \{e_i\}$ 以及 S_+ 对应的极点聚合值

$$H_{S_+} \leftarrow PolesAggr(mpk, S_+) = h^{g_{S_+}(\gamma)} = h^{\frac{1}{\gamma \prod_{e_k \in S_+} (x + x_k)}} \quad (5-34)$$

随后验证证据 W_i 是否与 $e(H_{S_+}, G_S)$ 相等，即，

$$W_i = e(H_{S_+}, G_S) \quad (5-35)$$

若以上两个等式均成立，则算法输出 $e_i \notin S$ ；否则输出 $e_i \in S$ 。

图 5-3 描述了负集合成员关系判定过程。首先对于给定的子集 S 通过基于零点的聚合函数表示成 $G_S \leftarrow ZerosAggr(S)$ ，随后对于给定的元素 e_i 利用 $Extract(sk, e_i) \rightarrow W_i$ 生成对应的证据。若 $e_i \notin S$ ，则可通过基于极点聚合函数计算 $S_+ = S \cup \{e_i\}$ 的安全表示 H_{S_+} ，随后通过在 G_S 和 H_{S_+} 之间执行双线性映射，从而消去集合 S 和 S_+ 中的相同元素。最后，将上述双线性映射结果与证据 W_i 之间进行匹配，实现最终的集合成员关系的判定。

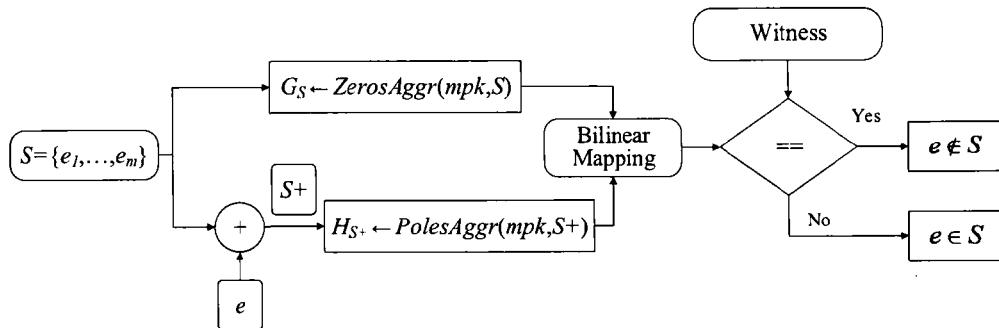


图 5-3 负集合成员关系判定示意图

定理 5-4 当基于极点的集合函数满足定义的安全需求时，上述负集合成员关系判定构造是安全的。

证明：根据集合成员关系安全判定的定义，接下来证明上述负集合成员关系判定构造满足完整性和完备性。

当 $e_i \notin S$ 时，对于任意的有效值 H_{S_+} ，下面等式总是成立。

$$\begin{aligned} e(G_S, H_{S_+}) &= e(g^{f_S(\gamma)}, h^{g_{S_+}(\gamma)}) \\ &= e(g^{\gamma \prod_{e_k \in S} (\gamma + x_k)}, h^{\frac{1}{(\gamma + x_i) \prod_{e_k \in S} (\gamma + x_k)}}) = e(g, h)^{\frac{\gamma}{\gamma + x_i}} = W_i \end{aligned} \quad (5-36)$$

根据该定理的前提条件，假设 $ZerosAggr$ 是一个安全的基于零点聚合函数，这意味着对于任意的概率多项式时间敌手 \mathcal{A} 以及给定的元素 $e_i^* \in S$ ，则有

$$\Pr[\mathcal{A}(mpk, e_i^*, S) = H_{S_+} | e_i^* \in S] < \epsilon \quad (5-37)$$

接下来证明验证者接受验证的概率是可忽略的，即 $\Pr[Verify_\epsilon(W_i, G_S) = 1] < \epsilon$ ，

$W_i = e(g, h)^{\frac{\gamma}{\gamma + x_i}}$ 和 $G_S = g^{\gamma \prod_{e_k \in S} (\gamma + x_k)}$ 是两个确定且不变的值， $x_i^* = \text{hash}(e_i^*)$ 。这意味着敌手 \mathcal{A} 可以伪造一个有效的 H^* 使得下列等式成立。

$$\Pr[Verify_\epsilon(W_i, G_S) = 1] = \Pr[e(g, h)^{\frac{\gamma}{\gamma + x_i}} = e(g^{\gamma \prod_{e_k \in S} (\gamma + x_k)}, H^*)] \quad (5-38)$$

不失一般性，令 $H^* = h^x$ ，则有

$$\begin{aligned}
& \Pr[\text{Verify}_\epsilon(W_i, G_S) = 1] = \Pr[e(g, h)^{\frac{\gamma}{r+x_i}} = e(g)^{z\gamma \prod_{e_k \in S} (r+x_k)}, H^*] \\
&= \Pr \left[\begin{array}{l} e(g, h)^{\frac{\gamma}{r+x_i}} = e(g, h)^{z\gamma \prod_{e_k \in S} (r+x_k)} \\ h^* \leftarrow \mathcal{A}(mpk, e_i^*, S) \end{array} \right] \cdot \Pr[h^* \leftarrow \mathcal{A}(mpk, e_i^*, S)] \\
&= \Pr \left[\mathcal{A}(mpk, e_i^*, S) \rightarrow h^* = h^{\frac{1}{(r+x_i) \prod_{e_k \in S} (r+x_k)}} \right] \\
&= \Pr[\mathcal{A}(mpk, e_i^*, S) = H_{S_+} | e_i^* \in S] < \epsilon
\end{aligned} \tag{5-39}$$

在式中，要求关系 $\frac{\gamma}{r+x_i} = z\gamma \prod_{e_k \in S} (r+x_k)$ 成立，因此有

$$\Pr \left[e(g, h)^{\frac{\gamma}{r+x_i}} = e(g, h)^{z\gamma \prod_{e_k \in S} (r+x_k)} \right] = 1 \tag{5-40}$$

在这种情况下 $z = \frac{1}{(\gamma + x_i^*) \prod_{e_k \in S} (r+x_k)} = \frac{1}{(\gamma + x_i^*)^2 \prod_{e_k \in S \setminus \{e_i^*\}} (r+x_k)}$ 以及 $H_{S_+} = h^*$ 。

这意味着敌手 \mathcal{A} 攻破安全的负集合成员关系判定的优势与攻破安全的基于极点聚合函数的优势相等，因此，在基于极点聚合函数安全的前提下，负集合成员关系判定也是安全的，该定理得证。

通过上述讨论可以看出，上述两个正负成员关系判定构造将判定问题转化成了计算 G_{S_-} 和 H_{S_+} 的问题。更重要的，从图 5-2 和图 5-3 中可以发现，这两种解决方案具有几乎相同的结构，因此可以很容易将谓词 \in 和 \notin 组合到一个密码系统中。下一节将应用这种方法来构造对偶集合成员证明协议。

5.4 零知识对偶集合成员关系证明协议

本章提出的零知识对偶集合成员关系证明协议（Zero-knowledge Dual-membership Proof, ZKDMMP）是通过证明者和验证者之间的交互来实现的。假设验证者持有一个公共集合 S ，验证者拥有一个秘密元素 e ，证明者想通过二者之间的交互，向验证者证明元素 $e \in S$ 或者 $e \notin S$ 而不暴露元素本身信息。

定义 5-9 对于给定的证明者 P 和验证者 V ，在全集 U ，子集 $S \subseteq U$ 以及某待测元素 e 之间的零知识对偶集合成员关系证明协议是安全的，当且仅当满足下列性质：

正完整性 (Positive Completeness): 对于诚实的证明者以及正集合成员关系 $e \in S$ ，验证者在与证明者交互之后输出 True 的概率不小于 $1 - \epsilon$ ，其中 ϵ 是可忽略的概率，即，

$$\Pr[\langle P(e), V(S) \rangle = \text{True} | e \in S] \geq 1 - \epsilon \tag{5-41}$$

负完整性 (Negative Completeness): 对于诚实的证明者以及正集合关系 $e \in S$ ($e \notin S \wedge e \in U$)，验证者在与证明者交互之后输出 False 的概率不小于 $1 - \epsilon'$ ，其中 ϵ' 是一个可忽略的概率，即

$$\Pr[\langle P(e), V(S) \rangle = \text{False} | e \in \bar{S}] \geq 1 - \epsilon' \quad (5-42)$$

完备性 (Soundness): 对于任意无效的 $e^* \notin U$ 以及证明者 P^* ，验证者在与证明者交互之后输出 True 或 False 的概率至多为 ϵ'' ，其中 ϵ'' 是一个可忽略的概率，即，

$$\Pr[\langle P^*(e^*), V(S) \rangle = \text{True} \vee \text{False} | e^* \notin U] < \epsilon'' \quad (5-43)$$

令 \perp 为 $e^* \notin U$ 是交互证明系统的输出，根据完备性可将上式改写成

$$\Pr[\langle P^*(e^*), V(S) \rangle = \perp | e^* \notin U] \geq 1 - \epsilon'' \quad (5-44)$$

零知识性 (Zero-knowledge): 在集合成员关系判定的交互证明系统中，证明者 P 具有零知识性是指，对于任意的验证者 V^* ，则存在一个多项式时间模拟器 M ，使得下列概率分布统计不可区分：

- 1) $\{(P, V^*)(e)\}_{e \in U}$ 表示当 P 和 V^* 的公共输入 $e \in S$ 时，二者交互过程中验证者所看到的信息，即 $\text{View}_{V^*}(e) = \{(P, V^*)(e)\}_{e \in U}$ ；
- 2) $\{M^*(e)\}_{e \in U}$ 表示模拟器 M^* 对公共输入 $e \in U$ 模拟协议运行的输出。

其中， M^* 表示模拟器 M 针对验证者 V^* 构造的多项式算法。上述两种分布的统计不可区分性可以表示为 $\{(P, V^*)(e)\}_{e \in U} \cong \{M^*(e)\}_{e \in U}$ 。

上述 ZKDMMP 协议的输出包括三种情况：True 表示正集合成员关系 ($e \in S$)；False 表示负集合成员关系 ($e \notin S$)； \perp 表示无效的集合成员关系 ($e \notin U$)。包含上述三种关系的判定协议被称为严格的集合成员关系判定。

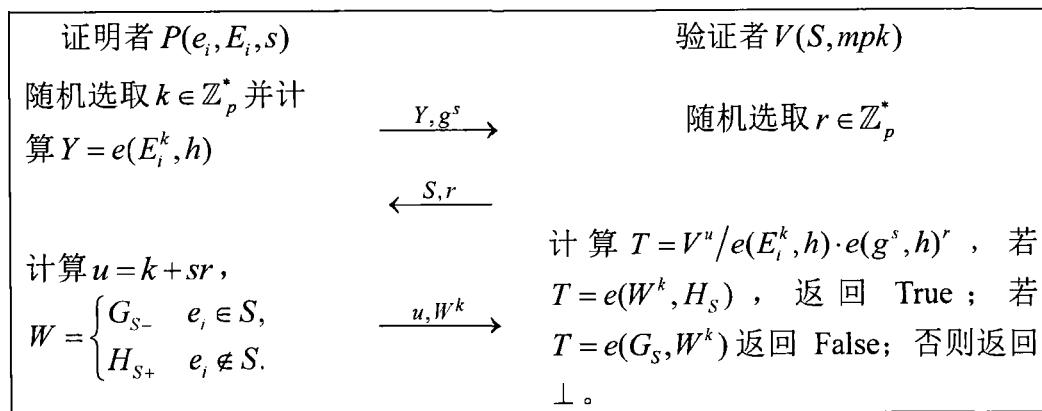


图 5-4 对偶集合成员关系零知识证明协议

接下来将给出集合成员关系判定交互协议的具体构造。如图 5-4 所示，

选取素数阶 p 的双线性群 $\mathbb{S} = (\mathbb{G}, \mathbb{G}_T, e(\cdot, \cdot))$ 。此外，令 $hash : \{0,1\} \rightarrow \mathbb{Z}_p^*$ 为抗碰撞哈希函数，它将任意的二进制字符串组成的身份标识 ID_i 映射到一个随机元素 $x_i \in \mathbb{Z}_p^*$ ，即 $x_i = hash(ID_i)$ ，其中抗碰撞性保证任意不同的元素被映射到同一个整数值，即对于任意的 $e_i \neq e_j$ 总有 $hash(e_i) \neq hash(e_j)$ 。接下来，本节将介绍对偶集合成员关系零知识证明协议具体描述。

$Setup(\mathbb{S}, U) \rightarrow (mpk, msk)$ ：令 g 是群 \mathbb{G} 中的一个随机生成元， $Setup$ 算法随机选取 $\delta \in \mathbb{Z}_p^*$ 并令 $h = g^\delta$ 以及 $V = e(g, h)$ 。该算法随机选取 $\gamma \in \mathbb{Z}_p^*$ ，对于任意的 $j \in [1, |U|]$ ，计算 $g_j = g^{\gamma^j}$ 。对于全集 U 中的任意元素 $e_i \in U$ ，算法计算 $x_i = hash(e_i)$ 与 $h_i = h^{\gamma^{x_i}}$ 并输出主密钥 $msk = \{\delta, \gamma, g\}$ 。最后，算法输出公共密钥 $mpk = \{\mathbb{S}, U, \{g_j\}_{j \in [1, |U|]}, \{h_i\}_{e_i \in U}, h, V\}$ 。

完整的对偶集合成员关系零知识证明协议包括以下三个阶段：

1) **初始化阶段：**证明者随机选取 $e_i \in U$ 和 $s \in \mathbb{Z}_p^*$ ，随后管理员通过系统主密钥计算证据 $E_i = g^{x_i / (r + x_i)}$ 和一个承诺 g^s ，并将它们分别发送给证明者和验证者。同时，验证者选取全集 U 的一个子集 $S \subseteq U$ 。

2) **交互证明阶段：**根据 Σ 协议中的三步结构，协议可以表示如下：

承诺：证明者随机选取 $k \in \mathbb{Z}_p^*$ ，并将 $e(E_i^k, h)$ 发送给验证者；

挑战：验证者随机选取 $r \in \mathbb{Z}_p^*$ ，并将子集 S 和 r 发送给证明者；

响应：证明者计算相应 u 和 W^k ，并将它们发给验证者，其中 $u = k + sr$ ，

$$W = \begin{cases} G_{S-} = ZerosAggr(mpk, S \setminus \{e_i\}), & e_i \in S \\ H_{S+} = PolesAggr(mpk, S \cup \{e_i\}), & e_i \notin S \end{cases} \quad (5-45)$$

3) **验证阶段：**验证者首先计算 $T = V^u / e(E_i^k, h) \cdot e(g^s, h)^r$ ，若 $T = e(W^k, H_S)$ ，则验证者返回 True；若等式 $T = e(G_S, W^k)$ 成立，则返回 False；否则返回 False，其中 $G_S = ZerosAggr(mpk, S)$ ， $H_S = PolesAggr(mpk, S)$ 。

5.5 基于 ZKDMR 的智能合约投票系统

基于 ZKDMR 协议设计的智能合约投票系统模型如图 5-5 所示，整个投票系统是基于支持虚拟机的区块链系统进行设计开发的，取代了以往第三方投票机构，从而实现投票系统的去中心化。在投票过程中，选票内容在开票之前应该对所有人保密以保证投票过程的公平性。通过将 ZKDMR 协议引入

到智能合约投票系统，不仅实现对投票内容有效性的验证从而避免无效选票对系统的影响，而且在验证过程中保护投票内容的隐私。

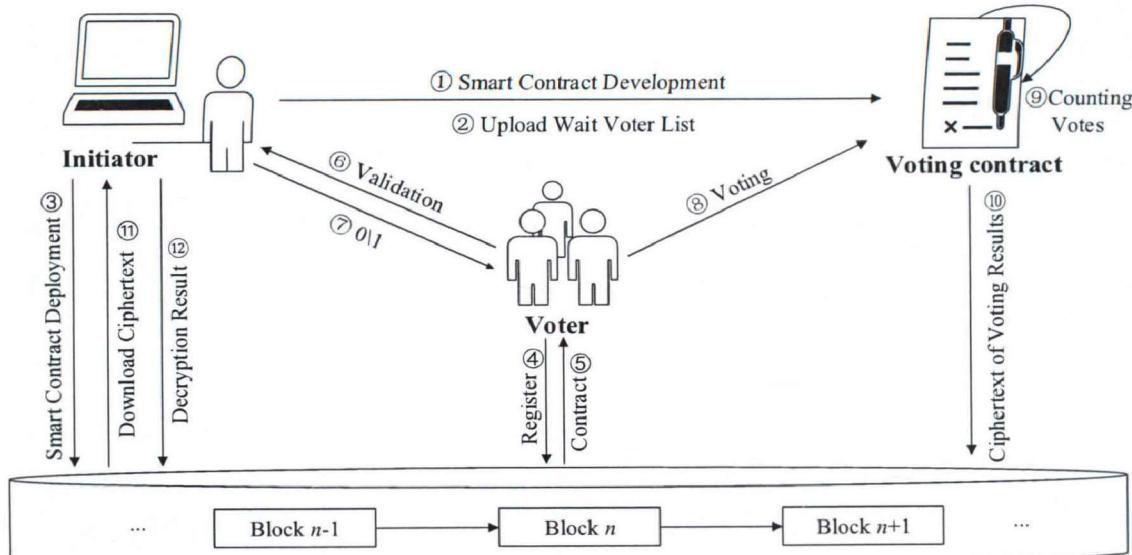


图 5-5 智能合约投票系统模型

智能合约投票系统执行流程如下：首先，投票发起者创建投票合约，并在合约中指定候选者名单，之后将合约部署到区块链中；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通过 ZKDMF 协议向投票发起者进行合法性验证，验证通过后将投票结果加密并按合约规定上传选票；当所有投票者完成投票后，合约开始计票并将密文状态的计票结果上传至区块链；计票结束后，投票发起者将从链上获取密文状态下的计票结果，利用自身私钥解密后，将计票结果公布到区块链上。

投票系统是通过智能合约控制实现的，投票合约在开始执行后，整个投票流程会被自动触发执行，智能合约投票的每一步都会被日志记录。投票系统可分为初始化、注册、投票和计票四个阶段。

1) 初始阶段：系统初始化阶段首先借助 JPBC 库完成对系统参数的生成，随后投票发起者制定投票智能合约并指定候选者名单，随后将投票合约以 JAR 包的形式部署到区块链。

2) 注册阶段：发起者在区块链上完成合约部署之后，投票者需完成系统注册，注册完成后，投票者名单会被上传至区块链。

3) 投票阶段：在投票之初，发起者需对投票者的投票内容进行有效性验证，只有验证通过才能继续投票操作。最后，投票者利用同态加密技术对验证通过的选票进行加密，按合约规定以交易的形式上传密文状态的投票结果。

4) 计票阶段：所有投票者完成投票后，投票合约进行计票并将结果上传至区块链。投票发起者可从区块链中获取密文状态的计票结果，利用自身私

钥进行解密后以交易的形式公布到区块链上。

接下来，本节将采用 SPESC 合约语言，对投票系统的初始化、注册、投票和计票四个阶段的触发条件以及执行过程进行合约化描述。

```

1. contract voting{
2.
3.     party initiator{
4.         waitVoterForm()
5.         initParam()
6.         Decryp()
7.         voterResult()
8.         membershipProof()
9.     }
10.    party voter{
11.        voterRegist()
12.        voting()
13.        Encrypt()
14.        membershipProof()
15.    }
16.    term term1 : initiator can waitVoterForm,
17.        when after initiator did initParam. //投票发起者上传候选人名单
18.    term term2 : voter can voterRegist,
19.        when after initiator did initParam. //选民身份注册
20.    term term3 : initiator and voter must do membershipProof,
21.        when before voter did voting. //验证选民身份合法性
22.    term term4 : voter can Encrypt,
23.        when after initiator and voter did membershipProof,
24.        while membershipProof outputs 1. //身份验证通过后进行投票
25.    term term5 : initiator can voteResult and Decrypt,
26.        when after voter did Encrypt. // 投票发起者解密选票
27. }
```

图 5-6 SPESC 语言编写的投票合约

如图 5-6 所示，具有零知识性的智能投票合约包括发起者（initiator）和投票者（voter）两个参与方。该合约包含 5 个条款，条款 1 表示在投票发起者进行参数初始化之后可指定候选者名单；条款 2 表示在发起者指定候选者名单之后，投票者进行注册；条款 3 描述的是在投票者注册完成之后，必须与发起者进行零知识集合关系证明，即 ZKDMP 来验证选票内容确为候选者名单中的一员；条款 4 表示当选票内容有效性验证通过，即交互协议返回为“1”时投票者利用 Boneh, Goh 和 Nissim 提出的同态加密算法^[125]对投票内容进行加密；条款 5 表示发起人在所有投票者完成投票之后通过合约进行计票，最终对计票结果进行解密获取投票最终结果。

5.6 安全性分析

按照定义 5-9 对对偶集合成员关系零知识证明交互协议安全性的定义，接下来证明本章设计的隐私集合成员判定协议满足正完整性、负完整性、完备性以及零知识性。

定理 5-5 本章所提出的 ZKDMP 协议满足定义 5-9 中描述的正完整性以及负完整性。

证明：当 $e_i \in S$ 时，有以下等式成立，

$$\begin{aligned} e(E_i^k, h) \cdot e(g^s, h)^r \cdot e(W^k, H_S) &= e(g^{\frac{s}{r+x}}, h) \cdot e(g^{sr}, h) \cdot e(G_{S_+}^k, H_S) \\ &= e(g^{\frac{s}{r+x}}, h) \cdot e(g^{sr}, h) \cdot e(g^{\frac{f_S(r)k}{r+x}}, h^{\frac{1}{rs(r)}}) = e(g, h)^{k+sr} = V^u \end{aligned} \quad (5-46)$$

因此，此时验证者输出 True 的概率可以表示如下：

$$\begin{aligned} \Pr[\langle P(e), V(S) \rangle = \text{True} | e \in S] \\ = \Pr[e(E_i^k, h) \cdot e(g^s, h)^r \cdot e(W^k, H_S) = V^u | e \in S] = 1 \end{aligned} \quad (5-47)$$

按照定义 5-9 的描述，协议的正完整性得证。

当 $e_i \in \bar{S}$ 时，有以下等式成立，

$$\begin{aligned} e(E_i^k, h) \cdot e(g^s, h)^r \cdot e(G_S, W^k) &= e(g^{\frac{s}{r+x}}, h) \cdot e(g^{sr}, h) \cdot e(G_S, H_{S_+}^k) \\ &= e(g^{\frac{s}{r+x}}, h) \cdot e(g^{sr}, h) \cdot e(g^{r f_S(r)}, h^{\frac{k}{r f_S(r)(r+x)}}) = e(g, h)^{k+sr} = V^u \end{aligned} \quad (5-48)$$

因此，此时验证者输出 False 的概率可以表示如下：

$$\begin{aligned} \Pr[\langle P(e), V(S) \rangle = \text{False} | e \in \bar{S}] \\ = \Pr[e(E_i^k, h) \cdot e(g^s, h)^r \cdot e(G_S, W^k) | e \in \bar{S}] = 1 \end{aligned} \quad (5-49)$$

按照定义 5-9 的描述，协议的负完整性得证。

定理 5-6 在定理 5-3 和定理 5-4 成立的条件下，本章所提出的 ZKDMP 协议满足定义 5-9 描述的完备性。

证明：在对协议完备性证明之前首先定义以下六个事件：

- 1) 事件 A_1 表示 $\langle P^*(e^*), V(S) \rangle = \text{True}$ ；
- 2) 事件 A_2 表示 $\langle P^*(e^*), V(S) \rangle = \text{False}$ ；
- 3) 事件 A 表示 $A_1 \cup A_2$ ；
- 4) 事件 B_1 表示 $e^* \notin S$ ；
- 5) 事件 B_2 表示 $e^* \notin \bar{S}$ ，即 $(e^* \in S) \cup (e^* \notin U)$ ；
- 6) 事件 B 表示 $e^* \notin U$ ，即 B 是 B_1 和 B_2 的联合事件， $B = B_1 B_2$ 。

这种情况下，事件 A_1 和事件 A_2 互斥，即 $A_1 A_2 = \emptyset$ ，则以下概率成立：

$$\begin{aligned}
& \Pr[\langle P^*(e^*), V(S) \rangle = \text{True} \vee \text{False} | e^* \notin U] \\
&= \Pr[A|B] = \Pr[(A_1 \cup A_2)|B] \\
&= \frac{\Pr[(A_1 \cup A_2)B]}{\Pr[B]} = \frac{\Pr[(A_1B) \cup (A_2B)]}{\Pr[B]} \\
&= \frac{\Pr[A_1B] + \Pr[A_2B]}{\Pr[B]}
\end{aligned} \tag{5-50}$$

对于任意的元素 $e^* \notin U$, 则有 $\Pr[B]=1$, 因此上式可表示为

$$\begin{aligned}
& \Pr[AB] = \Pr[(A_1 \cup A_2)|B] \\
&= \frac{\Pr[A_1B] + \Pr[A_2B]}{\Pr[B]} = \Pr[A_1B] + \Pr[A_2B] \\
&= \Pr[A_1B_1B_2] + \Pr[A_2B_1B_2] \leq \Pr[A_1B_1] + \Pr[A_2B_2]
\end{aligned} \tag{5-51}$$

若 $\Pr[A_1B_1] = \epsilon_1$ 和 $\Pr[A_2B_2] = \epsilon_2$ 是两个可忽略概率, 则 $\Pr[A|B]$ 也是可忽略的。在上述情况下, 协议的完备性就得到了证明。接下来, 只需证明上述假设成立, 即 $\Pr[A_1A_2] = \epsilon_1$ 和 $\Pr[B_1B_2] = \epsilon_2$ 均为可忽略概率。

对于任意的 $e^* \notin U$, 则有 $\Pr[B_1]=1$ 以及 $\Pr[A_1B_1] = \frac{\Pr[A_1|B_1]}{\Pr[B_1]} = \Pr[A_1|B_1]$ 成立。因此有,

$$\begin{aligned}
& \Pr[A_1B_1] = \Pr[A_1|B_1] \\
&= \Pr[\langle P(e^*), V(S) \rangle = \text{True} | e^* \notin S] \\
&= \Pr[T = e(W^k, H_s) | e^* \notin S] = \epsilon_1
\end{aligned} \tag{5-52}$$

接下来证明对于给定的 $e^* \notin S$, 验证者接受正集合成员关系验证的概率为 ϵ_1 。这意味着存在一个概率多项式时间敌手 \mathcal{A} 可以伪造出 W , 使得验证者成功验证 $T = e(W^k, H_s)$ 的概率为 ϵ_1 。由于 $H_s = h^{1/f_s(r)}$ 是一个确定且不变的值, 这意味着敌手 \mathcal{A} 能够伪造出 W 使得以下等式成立。

$$\Pr[T = e(W^k, H_s)] = \Pr\left[\frac{V^u}{e(E^k, h)e(g^s, h)^r} = e(W^k, H_s)\right] \tag{5-53}$$

不失一般性, 令 $W = g^z$, 因此有以下概率:

$$\begin{aligned}
& \Pr[T = e(W^k, H_S)] = \Pr[V^u / e(E_s^k, h) e(g^s, h)^r = e(W^k, H_S)] \\
&= \Pr[V^u = e(W^k, H_S) \cdot e(E_s^k, h) \cdot e(g^s, h)^r] \\
&= \Pr[e(g, h)^{k+sr} = e(W^k, h^{\frac{1}{f_S(r)}}) \cdot e(g^{\frac{x^*k}{r+x^*} + sr}, h) \cdot e(g^{sr}, h)] \\
&= \Pr_{\substack{W=g^z \\ g^z \leftarrow \mathcal{A}(mpk, e^*, S)}} \left[e(g, h)^{k+sr} = e(g, h)^{\frac{zk}{f_S(r)} + \frac{x^*k}{r+x^*} + sr} \right] \cdot \Pr[g^z \leftarrow \mathcal{A}(mpk, e^*, S)] \quad (5-54) \\
&= \Pr \left[\mathcal{A}(mpk, e^*, S) \rightarrow g^z = g^{\frac{zf_S(r)}{r+x^*}} \right] \\
&= \Pr[\mathcal{A}(mpk, e^*, S) = G_{S_1} \mid e^* \notin S] = \epsilon_1
\end{aligned}$$

在上式中，要求 $\frac{zk}{f_S(r)} + \frac{x^*k}{r+x^*} = k$ ，从而 $\Pr \left[e(g, h)^u = e(g, h)^{\frac{zk}{f_S(r)} + \frac{x^*k}{r+x^*} + sr} \right] = 1$ 成立，进而有 $z = \frac{rf_S(r)}{r+x^*}$ 以及 $G_{S_1} = g^z = W$ 。若概率 ϵ_1 是不可忽略的，则说明敌手 \mathcal{A} 可以以不可忽略的优势攻破基于零点的聚合表示，这与定理 5-3 相矛盾。

因此 ϵ_1 为可忽略的概率。

对于任意的 $e^* \notin U$ ，则有 $\Pr[B_2] = 1$ 以及 $\Pr[A_2 B_2] = \frac{\Pr[A_2 \mid B_2]}{\Pr[B_2]} = \Pr[A_2 \mid B_2]$

成立。因此有，

$$\begin{aligned}
\Pr[A_2 B_2] &= \Pr[A_2 \mid B_2] \\
&= \Pr[\langle P(e^*), V(S) \rangle = \text{False} \mid e^* \notin \bar{S}] \quad (5-55) \\
&= \Pr[T = e(G_S, W^k) \mid e^* \notin \bar{S}] = \epsilon_2
\end{aligned}$$

接下来，将证明对于给定的元素 $e^* \notin \bar{S}$ ，即 $e^* \in S$ 或 $e^* \notin U$ ，验证者接受负集合成员关系验证的概率为 ϵ_2 。这意味着存在一个概率多项式时间算法敌手 \mathcal{A} 可以伪造 W ，使得验证者成功验证 $T = e(G_S, W^k)$ 的概率为 ϵ_2 。由于 $G_S = g^{rf_S(r)}$ 是一个确定且不变的值，这意味着敌手能够伪造出 W 使得以下等式成立。

$$\Pr[T = e(G_S, W^k)] = \Pr \left[\frac{V^u}{e(E_s^k, h) e(g^s, h)^r} = e(G_S, W^k) \right] \quad (5-56)$$

不失一般性，令 $W = h^z$ ，因此有以下概率：

$$\begin{aligned}
& \Pr[T = e(G_s, W^k)] = \Pr[V^u / e(E_s^k, h) e(g^s, h)^r = e(G_s, W^k)] \\
&= \Pr[V^u = e(G_s, W^k) \cdot e(E_s^k, h) \cdot e(g^s, h)^r] \\
&= \Pr[e(g, h)^{k+sr} = e(g^{\gamma f_s(\gamma)}, W^k) \cdot e(g^{\frac{x^* k}{\gamma+x^*} + sr}, h) \cdot e(g^{sr}, h)] \\
&\stackrel{W=h^z}{=} \Pr \left[\begin{array}{l} e(g, h)^{k+sr} = e(g, h)^{\gamma f_s(\gamma)zk + \frac{x^* k}{\gamma+x^*} + sr} \\ g^z \leftarrow \mathcal{A}(mpk, e^*, S) \end{array} \right] \cdot \Pr[g^z \leftarrow \mathcal{A}(mpk, e^*, S)] \quad (5-57) \\
&= \Pr \left[\mathcal{A}(mpk, e^*, S) \rightarrow h^z = g^{\frac{1}{(\gamma+x^*)f_s(\gamma)}} \right] \\
&= \Pr[\mathcal{A}(mpk, e^*, S) = H_{S_1} \mid e^* \notin S] = \epsilon_2
\end{aligned}$$

在上述等式中，要求有等式关系 $\gamma f_s(\gamma)zk + \frac{x^* k}{\gamma+x^*} = k$ 成立，从而使得概率等式 $\Pr \left[e(g, h)^k = e(g, h)^{\frac{zk\gamma f_s(\gamma) + x^* k}{\gamma+x^*}} \right] = 1$ 成立，进而可知 $z = \frac{1}{(\gamma+x^*)f_s(\gamma)}$ 以及 $H_{S_1} = h^z = W$ 。若概率 ϵ_2 是不可忽略的，这说明敌手可以以不可忽略的优势攻破基于极点聚合的子集表示，这与定理 5-4 相矛盾。因此， ϵ_2 为可忽略的。

综上所述， ϵ_1 和 ϵ_2 均为可忽略的概率，因此可知概率 $\Pr[A \mid B] \leq \epsilon_1 + \epsilon_2$ 也是可忽略的，这意味着概率 $\Pr[(P^*(e^*), V(S)) = \text{True} \vee \text{False} \mid e^* \notin U] \leq \epsilon$ 也是可忽略的，其中 $\epsilon = \epsilon_1 + \epsilon_2$ 。至此，协议的完备性得证。

定理 5-7 本章所提出的 ZKDMF 具有统计零知识性。

证明：当公共输入为 e_i 时，协议正确执行过程中验证者能够获取的信息 $View_{V^*}(e_i)$ 的分布如下：

$$View_{V^*}(e_i) = \{(P, V^*)(e_i)\}_{e_i \in U} = (Y, S, r, W^k, u) \in_R \{\mathbb{G}_T, \mathcal{P}(U), \mathbb{Z}_p^*, \mathbb{G}, \mathbb{Z}_p^*\} \quad (5-58)$$

其中， $Y = e(E_i^k, h)$ ， $\mathcal{P}(U)$ 为集合 U 的幂集。接下来，构造一个多项式时间的协议模拟器 M^* 如下：

- 1) 随机选取 $r \in \mathbb{Z}_p^*$ ；
- 2) 随机选取 $W \in \mathbb{G}$ ， $t \in \mathbb{Z}_p^*$ 并计算 $u = t + sr$ ；
- 3) 按照如下方式计算 Y ：

$$Y = \begin{cases} \frac{V^u}{e(g^s, h)^r \cdot e(W^t, H_S)}, & e_i \in S \\ \frac{V^u}{e(g^s, h)^r \cdot e(G_s, W^t)} & e_i \notin S \end{cases} \quad (5-59)$$

通过上述计算过程可知，无论在 $e_i \in S$, $e_i \in \bar{S}$ 还是 $e_i \notin U$ 任何情况下， Y 始终是群 \mathbb{G}_T 中的随机元素。因此有 $M^*(e_i) \in_k \{\mathbb{G}_T, \mathcal{P}(U), \mathbb{Z}_p^*, \mathbb{G}, \mathbb{Z}_p^*\}$ ，进而 $\{(P, V^*)(e_i)\} \cong \{M^*(e_i)\}$ 成立，因此问题得证。

5.7 性能分析

接下来本节将对提出的安全聚合函数和集合成员关系证明协议进行性能评估。首先通过实验数据对两个聚合函数以及 ZKDMP 协议中的算法计算开销进行评估，随后从理论和仿真实验角度对比分析现有四个集合成员关系证明协议与本章提出的协议之间的差别。本章实验采用了第三章中相同的 JuLiuSC 区块链实验平台，密码系统由 JPBC 密码库实现，程序运行在 64 位 Windows10 PC 上，实验选取 SHA-256 密码函数作为仿真实验中的哈希函数。

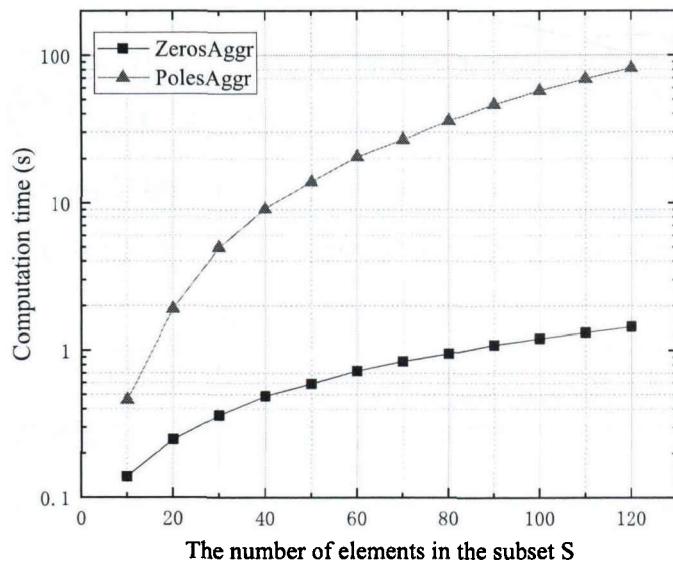


图 5-7 聚合函数计算开销

如图 5-7 所示，基于零点和极点的聚合函数计算开销都随着子集 S 中元素个数的增加而增加，具体来说基于零点聚合函数时间开销与元素个数呈线性关系，而基于极点聚合函数时间开销与元素个数呈指数关系。在元素个数相同的情况下，基于极点的聚合函数比基于零的聚合函数花费的时间更长，这是由于，基于极点聚合函数要比基于零点聚合函数执行更多次的指数运算，而且这种运算与其它运算相比更耗时。

本章提出的 ZKDMP 协议是在通过证明者和验证者之间通过三步交互执行的，按照 5.4 节中的描述，接下来将通过实验数据评估协议中的 Setup 算法以及交互过程中的三个阶段（初始化、交互证明、验证阶段）的时间开销。如图 5-8 所示，Setup 算法时间开销不随子集中元素个数变化而变化，且计算

时间维持在3到4秒之间，初始化阶段以及验证阶段的时间消耗是也是常量，时间开销均在0.05秒之内。

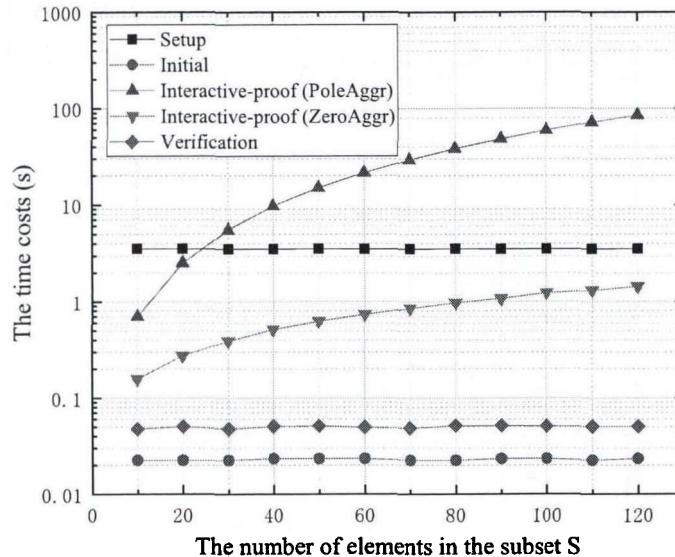


图 5-8 ZKDMF 协议中各个算法时间开销

为了展示 ZKDMF 协议与现有四个集合成员关系判定零知识证明协议在计算开销上的对比结果，这里首先对相关符号进行说明，其中 n 表示子集 S 中元素个数， $E(\mathbb{G})$ 和 $E(\mathbb{G}_T)$ 分别表示群 \mathbb{G} 和 \mathbb{G}_T 中的指数运算。 $M(\mathbb{G})$ 和 $D(\mathbb{G})$ 分别是群 \mathbb{G} 中的乘法和除法运算； $M(\mathbb{G}_T)$ 和 B 分别为群 \mathbb{G}_T 中的乘法和双线性运算。此外，性能对比忽略了哈希、群 \mathbb{G} 中的乘法以及 \mathbb{Z}_p^* 中的所有运算，这是因为与协议中的其它运算相比，它们是非常高效的。

表 5-1 证明者和验证者计算开销对比

方案	Prover	Verifier
文献[79]	$3E(\mathbb{G}) + M(\mathbb{G}) + 2E(\mathbb{G}_T) + M(\mathbb{G}_T) + 2B$	$(n+4)E(\mathbb{G}) + 2M(\mathbb{G}) + 3E(\mathbb{G}_T) + 2M(\mathbb{G}_T) + 3B$
文献[80]	$nE(\mathbb{G})$	$2nE(\mathbb{G}) + nM(\mathbb{G})$
文献[128]	$9E(\mathbb{G}) + 4M(\mathbb{G})$	$(n+7)E(\mathbb{G}) + 4M(\mathbb{G})$
文献[129]	$4E(\mathbb{G}) + M(\mathbb{G}) + 2E(\mathbb{G}_T) + M(\mathbb{G}_T) + 2B$	$(n+4)E(\mathbb{G}) + 2M(\mathbb{G}) + 3E(\mathbb{G}_T) + 2M(\mathbb{G}_T) + 3B$
本章协议	$\left(\frac{n^2+n}{2}+2\right)E(\mathbb{G}) + \frac{n^2+n}{2}D(\mathbb{G}) + B \quad \text{for } e_i \in S$ $(n+1)E(\mathbb{G}) + (n-1)M(\mathbb{G}) + B \quad \text{for } e_i \notin S$	$E(\mathbb{G}_T) + M(\mathbb{G}_T) + 3B$

从表 5-1 可知，文献[79][128][129]中的证明者时间消耗是常数级别的，这些协议中的验证者时间消耗与子集中的元素个数线性相关，而文献[80]中

证明者和验证者计算时间开销均与元素个数线性相关。在 ZKDMP 协议中，正集合成员关系判定中证明者的计算开销与子集 S 中的元素数量呈指数关系；在负集合成员关系判定中，验证者的计算开销与子集 S 中的元素数量线性相关。然而，无论是正的还是负集合成员关系判定，ZKDMP 协议中验证者的计算开销都是恒定的，与子集中元素数目无关。

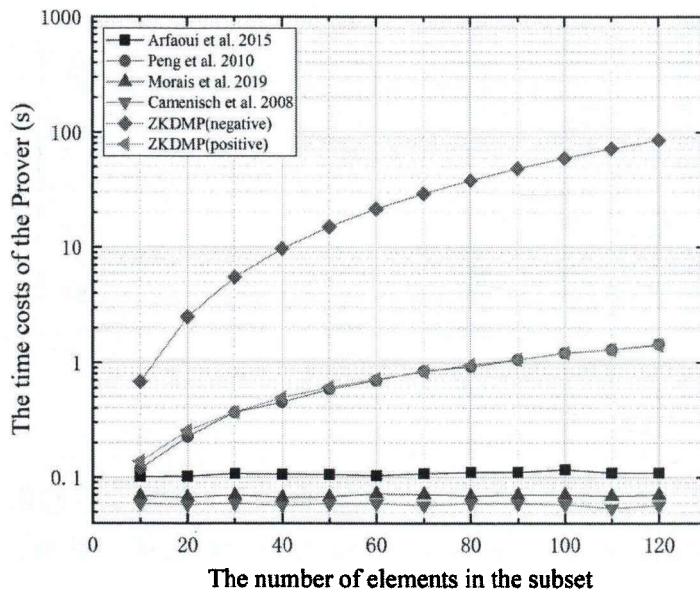


图 5-9 证明者时间开销对比

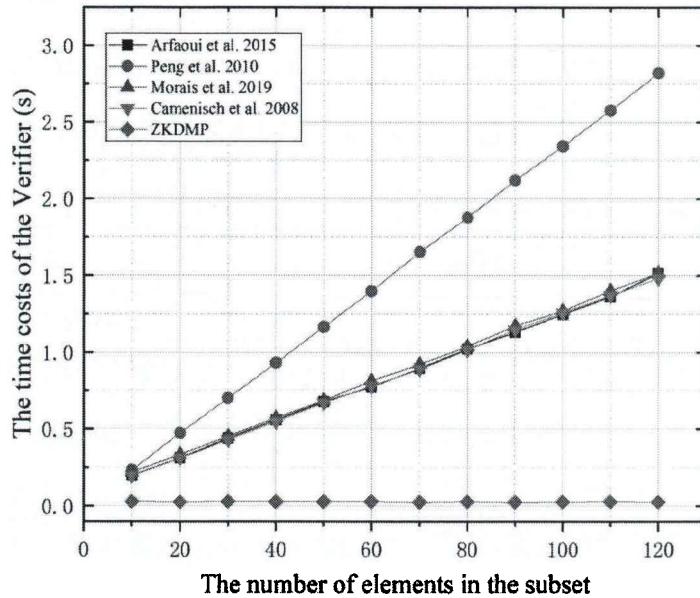


图 5-10 验证者时间开销对比

图 5-9 是相关零知识集合成员关系判定协议之间证明者时间开销对比结果，从图中不难发现，ZKDMP 协议中证明者在负集合成员关系判定过程中花费时间比其它协议中的成员关系判定要高，这是因为证明者需在交互证明过程中计算基于极点的聚合函数，而该聚合函数需要计算大量的群 G 中的指

数运算。ZKDMP 协议中验证者正集合成员关系判定的时间开销与文献[127]基本一致，随子集元素个数的增加而线性增长，这要比文献[126][128][129]中验证者计算时间开销要高。另一方面，如图 5-10 所示，仅有 ZKDMP 协议中的验证者时间开销保持常量，时间开销低于其它四个对比协议，这与前面的理论分析保持一致。

5.8 本章小结

本章提出了一种面向合约成员身份认证的零知识对偶集合成员关系证明（ZKDMP）协议。针对智能合约中动态群组成员的身份认证及判定过程中的用户隐私泄露问题，通过构造聚合函数实现子集压缩到密码空间元素的表示方法，将子集表示的规模降低到理论下限。本章引入安全聚合函数的概念，在子集压缩到随机元素过程中将元素与集合间属于和不属于的判定问题转变为元素删除和插入下的聚合函数求解问题，进而给出了基于正负成员关系判定架构下的 ZKDMP 协议及投票合约案例。此外，在 SDH 假设下证明了协议的正/负完整性、完备性以及零知识性。

6 总结

随着对智能合约技术研究的深入，智能合约隐私保护问题日益受到关注。本文通过对相关密码技术（包括广播加密、属性基加密和集合成员关系证明）研究现状进行了梳理和分析后发现：现在还没有能够与智能合约相结合且同时满足双模式广播加密以满足数据的不同隐私要求的广播加密方案；缺乏符合区块链分布式网络结构的去中心化密钥生成的属性基加密方案，且鲜有对密文存储结构的研究；缺乏同时支持正负成员关系严格判定的零知识证明协议以保护集合所属关系过程中的待测元素隐私。根据上述问题，本文针对智能合约交易数据隐私以及交易验证过程中的用户隐私两个方面展开研究，根据区块链去中心化网络特点，以面向智能合约隐私保护的密码协议构造为重点，分别对双模式广播加密、去中心化密钥管理和可编程密文的属性基加密以及零知识对偶集合成员关系证明协议展开研究，主要研究结果总结如下：

1) 支持双模式身份基广播加密（DM-IBBE）方案。针对智能合约敏感数据访问限定到指定用户和非冲突用户的两种隐私需求，提出了一种面向智能合约交易隐私的 DM-IBBE 方案，通过在拉格朗日插值曲线下对指定集和冲突集上不同插值点与重构曲线上点的选取，设计支持“选择”和“排它”两种加密模式的 DM-IBBE 分别满足上述隐私需求。此外，基于 SPESC 语言给出一种具有隐私保护功能的智能合约架构，保证敏感数据以合约条款形式声明隐私并由编译器将预定义的 DM-IBBE 算法链接到智能合约程序中实施保护。在 DDH 假设下证明了该方案的语义安全性以及选择加密模式下接收者匿名性，与其它广播加密方案相比，DM-IBBE 具有更加丰富的加密模式以满足智能合约不同的隐私需求。

2) 去中心化密钥管理的密文策略属性基加密（CP-DK-ABE）方案。针对区块链去中心化特征与属性基加密中心化密钥管理相冲突的问题，融合两类同态的安全多方计算技术，提出了一种面向属性基加密的去中心化密钥管理与脚本化密文机制以及智能合约化的方案构造，进而通过解密过程中的密钥查询、密文逻辑等操作扩展区块链脚本指令系统，在加密过程中实现针对密文策略中复杂逻辑的密文脚本化。最后，分别在 DBDH 和 DLDH 假设下证明了 CP-DK-ABE 方案的语义安全性以及用户私钥生成算法的隐私性。该方案实现了各方对主密钥的共同管理和用户私钥的协同生成、以及脚本解释器对脚本化密文的自动化解密。

3) 零知识对偶集合关系证明（ZKDMP）协议。针对智能合约中动态群组成员的身份认证及判定过程中的用户隐私泄露问题，通过构造聚合函数实

现子集压缩到密码空间元素的表示方法，将子集表示的规模降低到理论下限，并引入安全聚合函数（SAF）的概念，在子集压缩到随机元素过程中将元素与集合间属于和不属于的判定问题转变为元素删除和插入下的聚合函数求解问题。另外，在子集安全表示的基础上给出了基于正负成员关系判定架构下的 ZKDMP 协议及投票合约案例，证明了该协议在 SDH 假设下的正/负完整性、完备性以及零知识性。该协议具有支持动态元素添加与删除、集合元素数目不受限制以及更加严格的集合关系证明等优势。

本文研究结果表明，双模式身份基广播加密是实现智能合约交易隐私的有效途径并能满足交易数据不同隐私保护需求，主密钥的共同管理和用户私钥的协同生成符合区块链去中心化结构特点并提高了密钥管理的安全性，脚本化密文存储形式进一步简化了解密操作，零知识对偶集合关系证明协议在对正负集合成员关系证明的同时也保护了待测元素的隐私。这些工作将为面向智能合约隐私保护中更加安全高效的密码技术研究提供理论与实践支撑。

本文在面向智能合约隐私保护的密码协议构造方面取得了一定的成果，但有些内容仍需后续进一步研究，这些内容主要包括：在现有身份基广播加密的基础上，进一步研究排它加密模式下的接收者匿名性问题，探索更加安全的双模式广播加密方案的构造；尝试通过其它数学工具，例如中国剩余定理，进行密钥协同生成算法的研究，探索符合区块链结构特点的密钥管理新方法；优化聚合函数算法复杂度，在此基础上构造更加高效且安全的零知识对偶集合关系证明协议。

参考文献

- [1] Tolmach P, Li Y, Lin S W, et al. A survey of smart contract formal specification and verification[J]. ACM Computing Surveys, 2021, 54(7): 1-38.
- [2] Szabo N. Smart contracts: building blocks for digital markets[J]. The Journal of Transhumanist Thought, 1996, 18(2): 2-20.
- [3] Hewa T, Ylianttila M, Liyanage M. Survey on blockchain based smart contracts: Applications, opportunities and challenges[J]. Journal of Network and Computer Applications, 2021, 177: 1-55.
- [4] Buterin V. A next-generation smart contract and decentralized application platform[J]. White Paper, 2014, 3(37): 1-36.
- [5] 朱岩, 王巧石, 秦博涵, 王中豪. 区块链技术及其研究进展[J]. 工程科学学报, 2019, 41(11): 1361-1373.
- [6] Savelyev A. Contract law 2.0: ‘Smart’ contracts as the beginning of the end of classic contract law[J]. Information & Communications Technology Law, 2017, 26(2): 116-134.
- [7] Kolgart M, Poola M, Rull A. Smart contracts[M]. The Future of Law and Etechnologies. Springer, Cham, 2016: 133-147.
- [8] Drummer D, Neumann D. Is code law? current legal and technical adoption issues and remedies for blockchain-enabled smart contracts[J]. Journal of Information Technology, 2020, 35(4): 337-360.
- [9] 朱岩, 王迪. 智能法律合约: 面向合约的软件开发语言、技术及应用[M]. 北京: 清华大学出版社, 2022.
- [10] Wang S, Yuan Y, Wang X, et al. An overview of smart contract: architecture, applications, and future trends[C]/Proceedings of the 2018 Intelligent Vehicles Symposium. IEEE, 2018: 108-113.
- [11] 范吉立, 李晓华, 聂铁铮, 于戈. 区块链系统中智能合约技术综述[J]. 计算机科学, 2019, 46(11): 1-10.
- [12] 蔡维德, 郁莲, 王荣, 刘娜, 邓恩艳. 基于区块链的应用系统开发方法研究[J]. 软件学报, 2017, 28(6): 1474-1487.
- [13] Karamitsos I, Papadaki M, Al Barghuthi N B. Design of the blockchain smart contract: A use case for real estate[J]. Journal of Information Security, 2018, 9(03): 177-190.
- [14] 郭少飞. 区块链智能合约的合同法分析[J]. 东方法学, 2019, 3(8): 4-17.
- [15] 邵奇峰, 金澈清, 张召, 钱卫宁, 周傲英. 区块链技术: 架构及进展[J].

计算机学报, 2018, 41(5): 969-988.

- [16] Wang S, Ouyang L, Yuan Y, et al. Blockchain-enabled smart contracts: architecture, applications, and future trends[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2019, 49(11): 2266-2277.
- [17] Tso R, Liu Z Y, Hsiao J H. Distributed e-voting and E-bidding systems based on smart contract[J]. Electronics, 2019, 8(4): 1-22.
- [18] Negara E S, Hidayanto A N, Andryani R, et al. Survey of smart contract framework and its application[J]. Information, 2021, 12(7): 1-10.
- [19] Rouhani S, Deters R. Security, performance, and applications of smart contracts: a systematic survey[J]. IEEE Access, 2019, 7: 50759-50779.
- [20] Kushwaha S S, Joshi S, Singh D, et al. Systematic review of security vulnerabilities in ethereum blockchain smart contract[J]. IEEE Access, 2022.
- [21] 朱健, 胡凯, 张伯钧. 智能合约的形式化验证方法研究综述[J]. 电子学报, 2021, 49(4): 792-804.
- [22] Praitheeshan P, Pan L, Yu J, et al. Security analysis methods on ethereum smart contract vulnerabilities: a survey[J]. arXiv preprint arXiv:1908.08605, 2019, 1-21.
- [23] Zhang F, Cecchetti E, Croman K, et al. Town crier: an authenticated data feed for smart contracts[C]//Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. 2016: 270-282.
- [24] Sergey I, Hobor A. A concurrent perspective on smart contracts[C]//International Conference on Financial Cryptography and Data Security. Springer, Cham, 2017: 478-493.
- [25] Luu L, Chu D H, Olickel H, et al. Making smart contracts smarter[C]//Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. 2016: 254-269.
- [26] Permenev A, Dimitrov D, Tsankov P, et al. Verx: Safety verification of smart contracts[C]//IEEE Symposium on Security and Privacy. IEEE, 2020: 1661-1677.
- [27] Nehai Z, Piriou P Y, Daumas F. Model-checking of smart contracts[C]//International Conference on Internet of Things and IEEE Green Computing and Communications and IEEE Cyber, Physical and Social Computing and IEEE Smart Data. IEEE, 2018: 980-987.
- [28] Bigi G, Bracciali A, Meacci G, et al. Validation of decentralised smart contracts through game theory and formal methods[M]. Programming Languages with Applications to Biology and Security. Springer, Cham, 2015: 142-161.
- [29] Gupta R, Tanwar S, Al-Turjman F, et al. Smart contract privacy protection

- using AI in cyber-physical systems: tools, techniques and challenges[J]. IEEE Access, 2020, 8: 24746-24772.
- [30] Alvi S T, Uddin M N, Islam L. Digital voting: A blockchain-based e-voting system using biohash and smart contract[C]//Proceedings of the 3rd International Conference on Smart Systems and Inventive Technology. IEEE, 2020: 228-233.
- [31] 韩璇, 袁勇, 王飞跃. 区块链安全问题: 研究现状与展望[J]. 自动化学报, 2019, 45(1): 206-225.
- [32] 胡甜媛, 李泽成, 李必信, 包骐豪. 智能合约的合约安全和隐私安全研究综述[J]. 计算机学报, 2021, 44(12): 2485-2514.
- [33] Galal H S, Youssef A M. Succinctly verifiable sealed-bid auction smart contract[M]. Data Privacy Management, Cryptocurrencies and Blockchain Technology. Springer, Cham, 2018: 3-19.
- [34] Bünz B, Agrawal S, Zamani M, et al. Zether: Towards privacy in a smart contract world[C]//International Conference on Financial Cryptography and Data Security. Springer, Cham, 2020: 423-443.
- [35] Steffen S, Bichsel B, Gersbach M, et al. zkay: Specifying and enforcing data privacy in smart contracts[C]//Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security. 2019: 1759-1776.
- [36] Baumann N, Steffen S, Bichsel B, et al. zkay v0. 2: Practical data privacy for smart contracts[J]. arXiv preprint arXiv:2009.01020, 2020, 1-44.
- [37] Steffen S, Bichsel B, Baumgartner R, et al. ZeeStar: Private smart contracts by homomorphic encryption and zero-knowledge proofs[C]//IEEE Symposium on Security and Privacy. IEEE Computer Society, 2022: 1543-1543.
- [38] Fiat A, Naor M. Broadcast encryption[C]//Annual International Cryptology Conference. Springer, Berlin, Heidelberg, 1993: 480-491.
- [39] Dodis Y, Fazio N. Public key broadcast encryption for stateless receivers[C]//ACM Workshop on Digital Rights Management. Springer, Berlin, Heidelberg, 2002: 61-80.
- [40] Jin H, Xu C, Luo Y, et al. Blockchain-based secure and privacy-preserving clinical data sharing and integration[C]//International Conference on Algorithms and Architectures for Parallel Processing. Springer, Cham, 2020: 93-109.
- [41] Deng Y, Wang S, Zhang Q, et al. A Secure Subscription-Push Service Scheme Based on Blockchain and Edge Computing for IoT[J]. KSII Transactions on Internet and Information Systems, 2022, 16(2): 445-466.
- [42] Susilo W, Chen R, Guo F, et al. Recipient revocable identity-based broadcast

- encryption: How to revoke some recipients in IBBE without knowledge of the plaintext[C]//Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security. 2016: 201-210.
- [43] Delerablée C. Identity-based broadcast encryption with constant size ciphertexts and private keys[C]//International Conference on the Theory and Application of Cryptology and Information Security. Springer, Berlin, Heidelberg, 2007: 200-215.
- [44] Sakai R, Furukawa J. Identity-based broadcast encryption[J]. Cryptology ePrint Archive, 2007, 217: 1-14.
- [45] Boneh D, Waters B, Zhandry M. Low overhead broadcast encryption from multilinear maps[C]//Annual Cryptology Conference. Springer, Berlin, Heidelberg, 2014: 206-223.
- [46] Barth A, Boneh D, Waters B. Privacy in encrypted content distribution using private broadcast encryption[C]//International Conference on Financial Cryptography and Data Security. Springer, Berlin, Heidelberg, 2006: 52-64.
- [47] Lai J, Mu Y, Guo F, et al. Fully privacy-preserving and revocable ID-based broadcast encryption for data access control in smart city[J]. Personal and Ubiquitous Computing, 2017, 21(5): 855-868.
- [48] Jia H, Chen Y, Yang K, et al. Revocable broadcast encryption with constant ciphertext and private key size[J]. Chinese Journal of Electronics, 2019, 28(4): 690-697.
- [49] Zhu Y, Yu R, Chen E, et al. An efficient broadcast encryption supporting designation and revocation mechanisms[J]. Chinese Journal of Electronics, 2019, 28(3): 445-456.
- [50] Sahai A, Waters B. Fuzzy identity-based encryption[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer, Berlin, Heidelberg, 2005: 457-473.
- [51] Goyal V, Pandey O, Sahai A, et al. Attribute-based encryption for fine-grained access control of encrypted data[C]//Proceedings of the 13th ACM Conference on Computer and Communications Security. 2006: 89-98.
- [52] Xu H, He Q, Li X, et al. BDSS-FA: a blockchain-based data security sharing platform with fine-grained access control[J]. IEEE Access, 2020, 8: 87552-87561.
- [53] Alniamy A, Taylor B D. Attribute-based access control of data sharing based on hyperledger blockchain[C]//Proceedings of the 2nd International Conference on Blockchain Technology. 2020: 135-139.
- [54] Wang H, Song Y. Secure cloud-based EHR system using attribute-based cryptosystem and blockchain[J]. Journal of Medical Systems, 2018, 42(8): 1-

9.

- [55] Wu A, Zhang Y, Zheng X, et al. Efficient and privacy-preserving traceable attribute-based encryption in blockchain[J]. Annals of Telecommunications, 2019, 74(7): 401-411.
- [56] Bramm G, Gall M, Schütte J. Blockchain-based distributed attribute based encryption[C]//International Conference on Security and Cryptography. IEEE, Piscataway, 2018: 265-276.
- [57] He Q, Xu Y, Liu Z, et al. A privacy-preserving Internet of Things device management scheme based on blockchain[J]. International Journal of Distributed Sensor Networks, 2018, 14(11): 1-12.
- [58] Guo R, Shi H, Zheng D, et al. Flexible and efficient blockchain-based ABE scheme with multi-authority for medical on demand in telemedicine system[J]. IEEE Access, 2019, 7: 88012-88025.
- [59] Chase M. Multi-authority attribute based encryption[C]//Theory of Cryptography Conference. Springer, Berlin, Heidelberg, 2007: 515-534.
- [60] Lewko A, Waters B. Decentralizing attribute-based encryption[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer, Berlin, Heidelberg, 2011: 568-588.
- [61] Kirupanithi D N, Antonidoss A. Efficient Data Sharing using Multi-authority Attribute Based Encryption in Blockchain[C]//Proceedings of the 5th International Conference on Electronics, Communication and Aerospace Technology. IEEE, 2021: 642-646.
- [62] Banerjee S, Bera B, Das A K, et al. Private blockchain-envisioned multi-authority CP-ABE-based user access control scheme in IIoT[J]. Computer Communications, 2021, 169: 99-113.
- [63] Zhang Y, Zhang L, Wu Q, et al. Blockchain-enabled efficient distributed attribute-based access control framework with privacy-preserving in IoV[J]. Journal of King Saud University-Computer and Information Sciences, Available Online 13 September 2022.
- [64] Baza M, Lasla N, Mahmoud M M E A, et al. B-ride: Ride sharing with privacy-preservation, trust and fair payment atop public blockchain[J]. IEEE Transactions on Network Science and Engineering, 2019, 8(2): 1214-1229.
- [65] Bloom B H. Space/time trade-offs in hash coding with allowable errors[J]. Communications of the ACM, 1970, 13(7): 422-426.
- [66] Ozcelik I, Medury S, Broaddus J, et al. An overview of cryptographic accumulators[J]. arXiv preprint arXiv:2103.04330, 2021, 1-11.
- [67] Broder A, Mitzenmacher M. Network applications of bloom filters: a survey[J]. Internet Mathematics, 2004, 1(4): 485-509.

- [68] Egert R, Fischlin M, Gens D, et al. Privately computing set-union and set-intersection cardinality via bloom filters[C]//Australasian Conference on Information Security and Privacy. Springer, Cham, 2015: 413-430.
- [69] Nojima R, Kadobayashi Y. Cryptographically secure Bloom-filters[J]. Transactions on Data Privacy, 2009, 2(2): 131-139.
- [70] Ramezanian S. A study of privacy preserving queries with bloom filters[D]. Turku: University of Turku, 2016.
- [71] Goldwasser S, Micali S. Probabilistic encryption & how to play mental poker keeping secret all partial information[C]//Proceedings of the 14th Annual ACM Symposium on Theory of Computing. ACM, 1982: 365-377.
- [72] Chaum D. Blind signatures for untraceable payments[C]//Advances in Cryptology. Springer, Boston, MA, 1983: 199-203.
- [73] Benaloh J, De Mare M. One-way accumulators: A decentralized alternative to digital signatures[C]//Workshop on the Theory and Application of Cryptographic Techniques. Springer, Berlin, Heidelberg, 1993: 274-285.
- [74] Papamanthou C, Tamassia R, Triandopoulos N. Authenticated hash tables[C]//Proceedings of the 15th ACM Conference on Computer and Communications Security. ACM, 2008: 437-448.
- [75] Papamanthou C, Tamassia R, Triandopoulos N. Authenticated hash tables based on cryptographic accumulators[J]. Algorithmica, 2016, 74(2): 664-712.
- [76] Camenisch J, Lysyanskaya A. An efficient system for non-transferable anonymous credentials with optional anonymity revocation[C]//International Conference on the Theory and Applications of Cryptographic Techniques. Springer, Berlin, Heidelberg, 2001: 93-118.
- [77] Camenisch J, Lysyanskaya A. Dynamic accumulators and application to efficient revocation of anonymous credentials[C]//Annual International Cryptology Conference. Springer, Berlin, Heidelberg, 2002: 61-76.
- [78] Camenisch J, Lysyanskaya A. A signature scheme with efficient protocols[C]//International Conference on Security in Communication Networks. Springer, Berlin, Heidelberg, 2002: 268-289.
- [79] Derler D, Hanser C, Slamanig D. Revisiting cryptographic accumulators, additional properties and relations to other primitives[C]//Cryptographers' Track at the RSA Conference. Springer, Cham, 2015: 127-144.
- [80] Ghosh E, Ohrimenko O, Papadopoulos D, et al. Zero-knowledge accumulators and set operations[J]. IACR Cryptology ePrint Archive, 2015, 404: 1-46.
- [81] Zhang F, Safavi-Naini R, Susilo W. An efficient signature scheme from bilinear pairings and its applications[C]//IACR International Workshop on

- Public Key Cryptography. Springer, Berlin, Heidelberg, 2004: 277-290.
- [82] Koblitz N, Menezes A. Pairing-based cryptography at high security levels[C]//IMA International Conference on Cryptography and Coding. Springer, Berlin, Heidelberg, 2005: 13-36.
- [83] Oded G. Foundations of cryptography: Volume 1, Basic Tools[M]. Cambridge University Press, 2006.
- [84] Bellare M. A note on negligible functions[J]. Journal of Cryptology, 2002, 15(4): 271–284.
- [85] Boneh D, Boyen X. Short signatures without random oracles and the SDH assumption in bilinear groups[J]. Journal of Cryptology, 2008, 21(2): 149-177.
- [86] Boneh D. The decision diffie-hellman problem[C]//International Algorithmic Number Theory Symposium. Springer, Berlin, Heidelberg, 1998: 48-63.
- [87] Boneh D, Boyen X, Shacham H. Short group signatures[C]//Annual International Cryptology Conference. Springer, Berlin, Heidelberg, 2004: 41-55.
- [88] Boneh D, Boyen X. Efficient selective-ID secure identity-based encryption without random oracles[C]//International Conference on the Theory and Applications of Cryptographic Techniques. Springer, Berlin, Heidelberg, 2004: 223-238.
- [89] Condon A. The complexity of space boundes interactive proof systems[C]//Complexity Theory: Current Research. 1992: 147-189.
- [90] Mao W. Modern cryptography: theory and practice[M]. Pearson Education India, 2003.
- [91] 朱岩, 陈娥. 数字认证技术[M]. 北京: 清华大学出版社, 2022.
- [92] Shamir A. How to share a secret[J]. Communications of the ACM, 1979, 22(11): 612-613.
- [93] Zhu Y, Guo Q, Yin H, et al. Blockchain-based software architecture development for service requirements with smart contracts[J]. Computer, 2021, 54(12): 72-80.
- [94] 郭倩, 朱岩, 殷红建, 陈娥, 王迪, 刘国伟. 基于要约 - 承诺的智能法律合约订立方法与实现[J]. 工程科学学报, 2022, 44(12): 2138-2153.
- [95] Chen E, Qin B, Zhu Y, et al. SPESC-translator: towards automatically smart legal contract conversion for blockchain-based auction services[J]. IEEE Transactions on Services Computing, 2021, 15(5): 3061-3076.
- [96] He X, Qin B, Zhu Y, et al. Spesc: A specification language for smart contracts[C]//Proceedings of the 42nd Annual Computer Software and Applications Conference. IEEE, 2018: 132-137.

- [97] Zhu Y, Song W, Wang D, et al. TA-SPESC: toward asset-driven smart contract language supporting ownership transaction and rule-based generation on blockchain[J]. IEEE Transactions on Reliability, 2021, 70(3): 1255-1270.
- [98] 王迪, 秦博涵, 宋伟静, 朱岩. SPESC: 面向法律的智能合约设计与实践[J]. 网络空间安全, 2020, 11(9): 39-46.
- [99] Boneh D, Gentry C, Waters B. Collusion resistant broadcast encryption with short ciphertexts and private keys[C]//Annual International Cryptology Conference. Springer, Berlin, Heidelberg, 2005: 258-275.
- [100] Delerablée C, Paillier P, Pointcheval D. Fully collusion secure dynamic broadcast encryption with constant-size ciphertexts or decryption keys[C]//International Conference on Pairing-Based Cryptography. Springer, Berlin, Heidelberg, 2007: 39-59.
- [101] Goyal R, Vusirikala S, Waters B. Collusion resistant broadcast and trace from positional witness encryption[C]//IACR International Workshop on Public Key Cryptography. Springer, Cham, 2019: 3-33.
- [102] Zhang L, Hu Y, Wu Q. Adaptively secure identity-based broadcast encryption with constant size private keys and ciphertexts from the subgroups[J]. Mathematical and Computer Modelling, 2012, 55(1): 12-18.
- [103] Zhang L, Wu Q, Mu Y. Anonymous identity-based broadcast encryption with adaptive security[C]//International Symposium on Cyberspace Safety and Security. Springer, Cham, 2013: 258-271.
- [104] He K, Weng J, Liu J N, et al. Anonymous identity-based broadcast encryption with chosen-ciphertext security[C]//Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security. ACM, 2016: 247-255.
- [105] Chen L, Li J, Lu Y, et al. Adaptively secure certificate-based broadcast encryption and its application to cloud storage service[J]. Information Sciences, 2020, 538: 273-289.
- [106] Lai J, Mu Y, Guo F, et al. Anonymous identity-based broadcast encryption with revocation for file sharing[C]//Australasian Conference on Information Security and Privacy. Springer, Cham, 2016: 223-239.
- [107] Ge A, Wei P. Identity-based broadcast encryption with efficient revocation[C]//IACR International Workshop on Public Key Cryptography. Springer, Cham, 2019: 405-435.
- [108] Li J, Hu S, Zhang Y, et al. A decentralized multi-authority ciphertext-policy attribute-based encryption with mediated obfuscation[J]. Soft Computing, 2020, 24(3): 1869-1882.
- [109] Bethencourt J, Sahai A, Waters B. Ciphertext-policy attribute-based

- encryption[C]//IEEE Symposium on Security and Privacy. IEEE, 2007: 321-334.
- [110] Islam M A, Madria S. Attribute-based encryption scheme for secure multi-group data sharing in cloud[J]. IEEE Transactions on Services Computing, 2022, 15(4): 2158-2172.
- [111] Jiang Y, Susilo W, Mu Y, et al. Ciphertext-policy attribute-based encryption against key-delegation abuse in fog computing[J]. Future Generation Computer Systems, 2018, 78: 720-729.
- [112] Li Q, Ma J, Li R, et al. Secure, efficient and revocable multi-authority access control system in cloud storage[J]. Computers & Security, 2016, 59: 45-59.
- [113] Belguith S, Kaaniche N, Laurent M, et al. Phoabe: securely outsourcing multi-authority attribute based encryption with policy hidden for cloud assisted IoT[J]. Computer Networks, 2018, 133: 141-156.
- [114] Liang P, Zhang L, Kang L, et al. Privacy-preserving decentralized ABE for secure sharing of personal health records in cloud storage[J]. Journal of Information Security and Applications, 2019, 47: 258-266.
- [115] Han D, Pan N, Li K C. A traceable and revocable ciphertext-policy attribute-based encryption scheme based on privacy protection[J]. IEEE Transactions on Dependable and Secure Computing, 2022, 19(1): 316-327.
- [116] Nasiraei H, Ashouri-Talouki M. Anonymous decentralized attribute-based access control for cloud-assisted IoT[J]. Future Generation Computer Systems, 2020, 110: 45-56.
- [117] Peng K, Bao F. Efficiency improvement of homomorphic e-auction[C]//International Conference on Trust, Privacy and Security in Digital Business. Springer, Berlin, Heidelberg, 2010: 238-249.
- [118] Acar T, Nguyen L. Revocation for delegatable anonymous credentials[C]//IACR International Workshop on Public Key Cryptography. Springer, Berlin, Heidelberg, 2011: 423-440.
- [119] Bayer S, Groth J. Zero-knowledge argument for polynomial evaluation with application to blacklists[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer, Berlin, Heidelberg, 2013: 646-663.
- [120] Goldreich O, Micali S, Wigderson A. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems[J]. Journal of the ACM, 1991, 38(3): 690-728.
- [121] Dwivedi A D, Singh R, Ghosh U, et al. Privacy preserving authentication system based on non-interactive zero knowledge proof suitable for Internet of Things[J]. Journal of Ambient Intelligence and Humanized Computing,

2021, 13: 4639-4649.

- [122] Robert L, Miyahara D, Lafourcade P, et al. Physical zero-knowledge proof and NP-completeness proof of Suguru puzzle[J]. Information and Computation, 2022, 285: 1-14.
- [123] Micali S, Rabin M, Kilian J. Zero-knowledge sets[C]//Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science. IEEE, 2003: 80-91.
- [124] Benarroch D, Campanelli M, Fiore D, et al. Zero-knowledge proofs for set membership: efficient, succinct, modular[C]//International Conference on Financial Cryptography and Data Security. Springer, Berlin, Heidelberg, 2021: 393-414.
- [125] Boneh D, Goh E J, Nissim K. Evaluating 2-DNF formulas on ciphertexts[C]//Theory of Cryptography Conference. Springer, Berlin, Heidelberg, 2005: 325-341.
- [126] Camenisch J, Chaabouni R. Efficient protocols for set membership and range proofs[C]//International Conference on the Theory and Application of Cryptology and Information Security. Springer, Berlin, Heidelberg, 2008: 234-252.
- [127] Peng K, Bao F. Improving applicability, efficiency and security of non-membership proof[C]//Proceedings of the 2nd International Symposium on Data, Privacy, and E-Commerce. IEEE, 2010: 39-44.
- [128] Arfaoui G, Lalande J F, Traoré J, et al. A practical set-membership proof for privacy-preserving NFC mobile ticketing[J]. arXiv preprint arXiv:1505.03048, 2015, 1-25.
- [129] Morais E, Koens T, Van Wijk C, et al. A survey on zero knowledge range proofs and applications[J]. SN Applied Sciences, 2019, 1(8): 1-17.

作者简历及在学研究成果

一、 作者入学前简历

起止年月	学习或工作单位	备注
2010 年 09 月至 2014 年 06 月	在盐城师范学院数学与应用数学专业攻读学士学位	
2015 年 09 月至 2018 年 06 月	在西安电子科技大学应用数学专业攻读硕士学位	
2018 年 09 月至 2023 年 01 月	在北京科技大学计算机科学与技术专业攻读博士学位	

二、 在学期间从事的科研工作

- [1] 国家自然科学基金: 集合成员关系的密码学构造方法及其应用研究 (Cryptographic Construction for Set Membership and its applications, National Natural Foundation of China, Grant No. 61972032), 参与人.
- [2] 国家重点研发计划课题: 基于区块链技术的智能合约服务(National Key Technologies Research and Development Programs of China, Grant No. 2018YFB1402702), 参与人.

三、 在学期间论文情况

- [1] **Yin H**, Chen E, Zhu Y, Zhao C, Feng R, Yau S. Attribute-based private data sharing with script-driven programmable ciphertext and decentralized key management in blockchain internet of things[J]. IEEE Internet of Things Journal, 2022, 9(13): 10625-10639. SCI 检索. 检索号: 000812536000026.
- [2] **Yin H**, Chen E, Zhu Y, Feng R, Yau S. An efficient zero-knowledge dual membership proof supporting pos-and-neg membership decision[J]. Mathematics, 2022, 10(17): 3217. SCI 检索. 检索号: 000851818900001.
- [3] 殷红建, 朱岩, 王静, 郭光来, 陈娥. 基于零知识证明的智能合约投票系统设计与实现[J]. 工程科学学报, 2022. 在线发表. EI 刊源.
- [4] Zhu Y, Guo Q, **Yin H**, Liang K, Yau S. Blockchain-based software architecture development for service requirements with smart contracts[J].

- Computer, 2021, 54(12): 72-80. SCI 检索. 检索号: 000720517400009.
- [5] Chen E, Zhu Y, Liang K, Yin H. Secure remote cloud file sharing with attribute-based access control and performance optimization[J]. IEEE Transactions on Cloud Computing, 2021. 在线发表. SCI 刊源.
- [6] 郭倩, 朱岩, 殷红建, 陈娥, 王迪, 刘国伟. 基于要约 - 承诺的智能法律合约订立方法与实现[J]. 工程科学学报, 2022, 44(12): 2138-2153. 在线发表. EI 刊源.

独创性说明

本人郑重声明：所呈交的论文是我个人在导师指导下进行的研究工作及取得研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写的研究成果，也不包含为获得北京科技大学或其他教育机构的学位或证书所使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中做了明确的说明并表示了谢意。

签名: 陈红建 日期: 2022.11.24

关于论文使用授权的说明

本人完全了解北京科技大学有关保留、使用学位论文的规定，即：学校有权保留送交论文的复印件，允许论文被查阅和借阅；学校可以公布论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存论文。

(保密的论文在解密后应遵循此规定)

签名: 陈红建 导师签名: 李岩 日期: 2022.11.24

学位论文数据集

关键词*	密级*	中图分类号*	UDC	论文资助
智能合约，隐私保护，广播加密，属性基加密，零知识证明	公开	TP309.7		
学位授予单位名称*		学位授予单位代码*	学位类别*	学位级别*
北京科技大学		10008	工学	博士
论文题名*		并列题名		论文语种*
面向区块链智能合约的隐私保护技术研究				中文
作者姓名*	殷红建		学号*	B20180328
培养单位名称*		培养单位代码*	培养单位地址	邮编
北京科技大学		10008	北京市海淀区 学院路 30 号	100083
学科专业*		研究方向*	学制*	学位授予年*
计算机科学与技术		信息安全	4	2023 年
论文提交日期 * 2022 年 11 月 24 日				
导师姓名*	朱岩		职称*	教授
评阅人	答辩委员会主席*		答辩委员会成员	
	冯荣权 教授			
电子版论文提交格式 文本() 图像() 视频() 音频() 多媒体() 其他()				
推荐格式: application/msword; application/pdf				
电子版论文出版(发布)者		电子版论文出版(发布)地		权限声明
论文总页数*	117			
共 33 项, 其中带*为必填数据, 为 22 项。				