计算机科学
COMPUTER SCIENCE

ZKFERP:计算成本恒定的通用高效范围证明方案

李一聪，周宽久，王梓仲，徐琳

引用本文

李一聪，周宽久，王梓仲，徐琳. ZKFERP:计算成本恒定的通用高效范围证明方案[J]. 计算机科学, 2022, 49(10): 335-343.

LI Yi-cong, ZHOU Kuan-jiu, WANG Zi-zhong, XU Lin. ZKFERP:Universal and Efficient Range Proof Scheme with Constant Computational Cost[J]. Computer Science, 2022, 49(10): 335-343.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.210900044

ZKFERP:计算成本恒定的通用高效范围证明方案

李一聪　周宽久　王梓仲　徐　琳
大连理工大学软件学院　辽宁 大连 116024
(17640284112@mail.dlut.edu.cn)

摘　要　区块链去中心化的特性易导致交易层用户隐私泄露,引发信息安全问题。零知识范围证明的目的是在不透露交易数据的同时,机密验证数据属于合法正整数区间,有效解决了区块链隐私保护问题。现有的区块链范围证明方案在证明速度、验证速度及计算成本等方面仍有较大的优化空间;并且,现有方案无法处理浮点数问题,因此限制了范围证明的应用领域。基于此,提出了一种计算成本恒定且浮点数、整数通用的高效范围证明方案——ZKFERP。ZKFERP 在 Bulletproofs 的基础上改进零知识协议,优化证明结构,并设计了一种拉格朗日内积向量生成方法,使见证生成时间恒定,最后利用浮点数范围关系式构造承诺,实现浮点数范围证明。ZKFERP 仅依赖于离散对数假设,无需第三方可信。实验结果表明,ZKFERP 的通信成本和时间复杂度均恒定,且与已知最先进的范围证明方案相比,ZKFERP 的证明时间缩短了 40.0%,验证时间缩短了 29.8%。

关键词:区块链;隐私保护;零知识证明;范围证明;向量内积承诺
中图法分类号　TP399

ZKFERP:Universal and Efficient Range Proof Scheme with Constant Computational Cost

LI Yi-cong,ZHOU Kuan-jiu,WANG Zi-zhong and XU Lin
School of Software,Dalian University of Technology,Dalian,Liaoning 116024,China

Abstract　The decentralization of blockchain can easily lead to the leakage of users' private data at the transaction layer,which in turn leads to information security issues. The zero-knowledge range proof is designed to confidentially verify that the transaction data belongs to a legal positive integer range without revealing the transaction data. It effectively solves the problem of blockchain privacy leakage. The existing blockchain range proof scheme can still be further optimized in terms of proof speed,verification speed and calculation cost. In addition,the existing solutions cannot handle the floating-point number problem,thus limiting the application fields of range proofs. This paper proposes an efficient range proof scheme with constant computational cost and universal for floating-point numbers and integers,ZKFERP. It improves the zero-knowledge protocol based on Bulletproofs to optimize the proof structure,and a Lagrangian inner product vector generation method is designed to make the witness generation time constant and the commitment is constructed according to the floating-point number range relationship to implement floating-point range proof. ZKFERP only relies on the discrete logarithm assumption,and third-party credibility is not required. The communication cost and time complexity of ZKFERP are constant. Experimental results show that,compared with the most advanced known range proof scheme,ZKFERP's proof speed is increased by 40.0%,and the verification speed is increased by 29.8%.

Keywords　Blockchain,Privacy protection,Zero-knowledge proof,Range proof,Vector inner product commitment

1　引言

区块链是一种全局分布式数据库技术。其本身具有去中心化、全局节点共识的特点,能有效解决第三方信任委托问题,降低信用成本。但链上数据公开透明化的特点,使得任何个体用户都可以随时在分布式账本上验证其他用户交易的有效性,使用户数据受到了隐私泄露的威胁。虽然,比特币[1]等虚拟货币所使用的 UTXO 模型在交易时必须提供脚本签名供事物脚本验证有效性[2],但 UTXO 的交易金额仍是透明的;并且,当用户交易请求上链并交由系统验证时,用户隐私数据会以计算参数的形式被泄露给运行系统,降低了链上数据的安全保密性。因此,在保证数据去中心化性质的基础上,加强区块链数据的信息隐私安全性已成为区块链隐私保护的核心问题[3]。零知识证明[4]是解决数据机密验证的一种密码学手段,也是解决区块链数据隐私保护问题最有效的技术之一。其中,范围证明是零知识证明理论中一种十分重要的具体应用,其本质是证明提交的承诺值在指定的秘密数据区间内或属于公共集合中的成员,而不泄露承诺值内容。

到稿日期:2021-09-06　返修日期:2022-03-11
基金项目:科技部重点研发计划(2019YFD1101104)
This work was supported by the Key Research and Development Program Project of Ministry of Science and Technology(2019YFD1101104).
通信作者:周宽久(zhoukj@dlut.edu.cn)

目前，在现有区块链机密交易的范围证明方案中，当问题大小为 n 比特时，其证明时间复杂度最优为 $O(n)$；当问题的范围区间较大时，对于链上计算能力有限的区块链平台而言，其计算成本仍十分高昂。虽然个别方案实现了证明时间与验证时间的恒定，即时间开销与问题大小无关，但其并未考虑证据生成时间的恒定。因此，其总体计算成本依然是线性的，且运算速度仍有较大的提升空间。此外，目前已知的范围证明方案仅能处理整数，尚未深入研究能够同时兼容整数与浮点数的范围证明。针对上述问题，本文提出了一种计算成本恒定且浮点数、整数通用的高效范围证明方案——ZKFERP。ZKFERP 实现了见证生成时间、证明时间及验证时间的恒定，且运算速度更快。此外，ZKFERP 支持对任意灵活范围区间的整数和浮点数范围证明。

2 相关研究

2.1 机密交易与非交互式证明

针对链上隐私安全问题，Noether 提出了机密交易概念[5]，即通过构造交易承诺[6]来隐藏交易金额在一个数量范围内。为实现机密交易，Maxwell 等提出 Borromean 环签名技术[7]，Borromean 环签名后与 Pedersen 承诺[8]结合应用在门罗币(Monero)[9-10]中。

利用零知识证明理论，以太坊[11]平台为智能合约程序设计了一种非交互零知识证明(Non-interactive Zero-knowledge Proofs)工具，该工具为复杂智能合约程序提供了用户验证隐私保护[12]。但 NIZK 在链上的应用受限，原因是智能合约计算能力有限，区块链通信成本高昂，使得 NIZK 并不适合通过智能合约进行验证[13]。

zk-SNARK[14-15]作为一种简洁的非交互式零知识证明方案被提出。zk-SNARK 验证时间短、证据大小为常数，因而适合通过智能合约进行验证[16-17]。Zcash[18]零知识机密交易平台所应用的技术正是 zk-SNARK。zk-SNARK 利用数论假设有效地满足了机密交易概念，但其需要第三方可信机制(Trusted Setup)来托管运算所需的公共参考字符，这意味着基于 zk-SNARK 的证明系统可靠的前提是第三方信任机制完全可信，而这与区块链去中心化的特质在理论上有冲突。为解决可信机制问题，Wahby 等[19]针对 zk-SNARK 提出了 Hyrax 协议，使逻辑电路验证运算只依赖于离散对数假设而无需可信机制。Setty 等[20]也基于 zk-SNARK 提出了一种无需第三方可信的新构造方式。Ben-Sasson 等利用哈希函数碰撞提出 zk-STARK[21]，其主要通过哈希函数碰撞进行对称加密来替代 zk-SNARK 的数论假设，以消除可信机制。但 zk-STARK 生成的证据较长且生成方式不够简洁。2019 年，更多新的关于 zk-SNARK 的构建形式被提出，如 Maller 等提出的 Sonic[22]以及 Gabizon 等提出的 Plonk[23]。

2.2 范围证明

范围证明是零知识证明技术的研究热点。Lipmaa[24]利用拉格朗日四平方和定理，即任意非负整数都可以由 4 个平方数组成，实现了对非负整数的范围证明。Groth[25]对 Lipmaa 的研究工作进行平方数个数缩减，实现了基于三平方和的非负整数范围证明。但上述方法[24-25]的安全性依赖于强 RSA 假设，导致其承诺较大，不适合链上验证。

2018 年，Bunz 等[26]提出 Bulletproofs 简短证据范围证明，其在 Bootle 等[27]提出的多项式承诺方案基础上利用向量内积承诺，在不依靠第三方可信机制的情况下，将证据大小和时间复杂度缩减至对数级。Deng[28]等在 Bulletproofs 的基础上提出了 Cuproof，利用拉格朗日平方和定理构造维度恒定的内积向量，实现了证明时间与验证时间恒定的灵活整数区间范围证明。但 Cuproof 实现灵活范围证明的方式是执行两次证明的聚合，因此增加了时间开销。

3 本文的贡献

本文在向量内积承诺和多项式承诺思想的基础上，提出了一种新的计算成本恒定的通用高效范围证明方案——ZKFERP。ZKFERP 实现了见证生成时间、证明时间及验证时间的恒定，且运算速度更快。此外，ZKFERP 支持对整数和浮点数的范围证明，且支持任意的灵活范围区间。

ZKFERP 的主要创新性贡献如下：

(1)ZKFERP 通过设计新的承诺构造形式，提出了一种改进的零知识协议结构，使证明系统可以对多元二次算数表达式进行多项式承诺验证，而无须拆分表达式来执行聚合证明。在问题大小相同的情况下，ZKFERP 生成的承诺维度仅为 Cuproof 的 1/2。

(2)ZKFERP 对原始问题生成内积向量，提出内积向量生成方法，设计了一种最大平方拆分算法，将优化问题关系式内积向量搜索求解的时间复杂度降至 O(1)，使见证生成时间恒定，并对向量维度进行分化，缩减平均计算成本。

(3)ZKFERP 设计了一种浮点数范围关系表达式，并将其转化为新协议结构的多项式承诺，应用内积向量生成方法优化内积向量生成时间，实现计算成本恒定的通用高效范围证明。

本文将 ZKFERP 与其他先进的范围证明方法进行对比实验，结果表明，ZKFERP 的综合运算性能更加突出。

4 预备知识

本节首先引入符号的定义以及相关底层承诺工具的定义。

4.1 符号定义

本文令$\mathbb{G}$表示阶为素数 p 的循环群；$\mathbb{Z}_p$ 表示一个模为 p 的整数环；G^n 与 $\mathbb{Z}_p^n$ 分别表示在$\mathbb{G}$和 $\mathbb{Z}_p$ 上维度为 n 的向量空间；$\mathbb{Z}_p^*$ 表示 $\mathbb{Z}_p\backslash\{0\}$。$\mathbb{G}$的生成元表示为 $g,h\in\mathcal{G}$，由循环群元素构建的承诺用大写字母表示，例如，$A=g^a h^r\in\mathbb{G}$；而盲化因子则用小写字母 s 表示。

在向量的符号表示上，本文采用粗体小写字母来表示向量，例如 $\boldsymbol{a}\in\mathbb{Z}_p^n$ 表示 $a^1,\cdots,a^n\in\mathbb{Z}_p$；大写的粗体字母表示矩阵，例如 $\boldsymbol{A}\in\mathbb{Z}_p^{n\times m}$ 表示 n 行 m 列的矩阵。此外，$\langle\boldsymbol{a},\boldsymbol{b}\rangle=\sum_{i=1}^{n}a_i\cdot b_i$ 表示向量 $\boldsymbol{a},\boldsymbol{b}$ 的内积；而 $\boldsymbol{a}\circ\boldsymbol{b}=(a_1\cdot b_1,\cdots,a_n\cdot b_n)$表示向量 $\boldsymbol{a},\boldsymbol{b}$ 的点积；$c\cdot\boldsymbol{b}=(c\cdot b_1,\cdots,c\cdot b_n)$表示常数 c 与向量 $\boldsymbol{b}$ 的乘积。

本文同时将上述符号定义应用到群元素向量表示中。具体地，$\boldsymbol{g}=(g_1,\cdots,g_n)$表示循环群向量 $\mathcal{G}^n$ 的生成元向量；对于

向量 $\boldsymbol{a}$ 的绑定承诺可表示为 $C\ \boldsymbol{g}^{a}=\prod_{i=1}^{n}g_i^{a_i}$。具体的符号定义如表 1 所列。

表 1 符号定义

Table 1 Definition of notations

Notation	Description
P	证明端
V	验证端
$\mathbb{G}$	一个阶为素数 p 的循环群
$\mathbb{Z}_p$	一个模为 p 的整数环
$\mathcal{G}^n$	表示在 $\mathbb{G}$ 上维度为 n 的向量空间
$\mathbb{Z}_p^n$	表示在 $\mathbb{Z}_p$ 上维度为 n 的向量空间
$\mathbb{Z}_p^*$	$\mathbb{Z}_p\backslash\{0\}$
g,h	两个循环群 $\mathbb{G}$ 产生的点
$\boldsymbol{g},\boldsymbol{h}$	分别由 g 和 h 组成的多维向量
$\boldsymbol{v}_1,\boldsymbol{v}_2$	关于 v 的不等向量组
v	一个私有待证明消息
V	关于 v 的向量内积承诺
$\hat{V}$	关于 v 的差分承诺
$\boldsymbol{a},\boldsymbol{b}$	关于 v 的拉格朗日内积向量
a_i,b_i	向量 $\boldsymbol{a},\boldsymbol{b}$ 的元素
$\langle\boldsymbol{a},\boldsymbol{b}\rangle$	向量 $\boldsymbol{a},\boldsymbol{b}$ 的内积
A,B	关于向量 $\boldsymbol{a},\boldsymbol{b}$ 的 Pedersen 向量承诺
$\boldsymbol{s}_L,\boldsymbol{s}_R$	在整数环上选取的盲化因子向量
S	关于向量 $\boldsymbol{s}_L,\boldsymbol{s}_R$ 的 Pedersen 向量承诺
$\alpha,\rho,\varphi,\mu,\tau_1,\tau_2,x$	在模为 p 的整数环上选取的随机值
T_1,T_2	关于 τ_1,τ_2 的 Pedersen 承诺
z	在模为 p 的整数环上选取的非零随机值
X	多项式未知数
$l(X),r(X)$	由向量 $\boldsymbol{a},\boldsymbol{b}$ 与 X 构建的多项式
$l(x),r(x)$	将 x 带入 $l(X),r(X)$ 得到的结果
t	$l(x)$ 与 $r(x)$ 的内积
t_0,t_1,t_2	一元二次多项式系数
m	一个证明值
d_l,d_u	分别为 m 值的范围上、下界
f_m,f_{dl},f_{du}	分别为 m,d_l,d_u 的小数位数
$f_{\max}$	f_m,f_{dl},f_{du} 中的最大值

4.2 承诺定义

定义 1(承诺) 一个非交互式承诺方案由一组概率多项式时间(PPT)的算法(Setup,Com)组成。其中 Setup 算法为承诺方案生成公共参数 pp 和安全参数 λ。Com 算法则为消息空间 M_{pp} 和随机数空间 R_{pp} 定义与承诺空间 C_{pp} 的映射关系：$M_{pp}\times R_{pp}\to C_{pp}$。对于消息 $x\in M_{pp}$，承诺算法均匀随机生成随机数 $r\xleftarrow{\$}R_{PP}$，并计算承诺 $com=Com_{pp}(x,r)$。

定义 2(承诺同态性) 承诺的同态性指在交换群 M_{pp}，R_{pp}，C_{pp} 空间中，对于任意的 $x_1,x_2\in M_{pp}$，$r_1,r_2\in R_{pp}$ 存在如下关系：

$$Com(x_1,r_1)+Com(x_2,r_2)=Com(x_1+x_2,r_1+r_2)$$

定义 3(承诺隐藏性) 承诺的隐藏性指对于任意的概率多项式时间，如下关系成立：

$$\left|P\left[b=b'\mid pp\leftarrow Setup(1^{\lambda}),(x_0,x_1)\in M_{pp}^2\leftarrow A(pp),b\xleftarrow{\$}\{0,1\},r\xleftarrow{\$}R_{pp},com=Com(x_b,r),b'\leftarrow A(pp,com)\right]-\frac{1}{2}\right|\leqslant\eta(\lambda)$$

其中，$\eta(\lambda)$表示一个忽略不计的函数，当上述关系发生的概率 $\eta(\lambda)$趋近于 0 时，则承诺具有隐藏性，即承诺内容在计算上不可区分。

定义 4(承诺绑定性) 承诺的绑定性指对于任意的概率多项式时间，如下关系成立：

$$P\left[Com(x_0,r_0)=Com(x_1,r_1)\wedge x_0\neq x_1\mid pp\leftarrow Setup(1^{\lambda}),x_0,x_1,r_0,r_1\leftarrow A(pp)\right]\leqslant\eta(\lambda)$$

当上述关系发生的概率 $\eta(\lambda)$趋近于 0 时，则承诺具有绑定性，即承诺内容不可抵赖。

定义 5(Pedersen 承诺) Pedersen 承诺旨在隐藏绑定消息空间 M_{pp} 下的秘密值 x，承诺构造形式如下：

$$M_{PP},R_{PP}\in\mathbb{Z}_p,C_{PP}\in\mathbb{G}$$

$$\text{Setup}:g,h\xleftarrow{\$}\mathbb{G}$$

$$Com(x,r)=(g^xh^r)$$

定义 6(Pedersen 向量承诺) Pedersen 向量承诺旨在隐藏绑定消息空间 M_{pp} 下的秘密向量 $\boldsymbol{x}$，承诺构造形式如下：

$$M_{pp}\in\mathbb{Z}_p^n,\ R_{pp}\in\mathbb{Z}_P,\ C_{PP}\in\mathbb{G}$$

$$\text{Setup}:g=(g_1,\cdots g_n),\ h\xleftarrow{\$}\mathbb{G}$$

$$Com(\boldsymbol{x},r)=(h^rg^x)=h^r\prod_i g_i^{x_i}\in\mathbb{G}$$

定义 7(多项式承诺) 假设证明端 P 有多项式 $t(X)=\sum_{i=0}^{n}t_iX^i$，多项式承诺旨在隐藏并证明证明端 P 掌握多项式 $t(X)$。关于 $t(X)$的多项式承诺，首先对多项式系数$(t_0,\cdots,t_n)$构造 Pedersen 承诺 $C_i=g^{t_i}h^{r_i}\in\mathbb{G}$。随后，验证端 V 选取随机数 x 发送给证明端 P，证明端 P 根据 x 计算 $t(x)$以及随机项 $\sum_{i=0}^{n}r_ix_i$，并发送给验证端 V，验证端 V 验证如下等式是否成立：

$$g^{t(x)}h^{\sum_{i=0}^{n}r_ix^i}=\prod_{i=0}^{n}C_i^{x^i}\in\mathbb{G}$$

证明系统可根据 Fiat-Shamir 启发式[29-31]的论断，在证明端利用哈希函数生成随机数 x，将承诺构造过程改为非交互式。

定义 8(向量内积承诺) 向量内积承诺旨在隐藏并证明证明端 P 掌握向量$\boldsymbol{a}_1,\boldsymbol{a}_2$ 使$\langle a_1,a_2\rangle=c$，承诺构造形式如下：

$$M_{PP}\in\mathbb{Z}_p^n,C_{PP}\in\mathbb{G}$$

$$\text{Setup}:\boldsymbol{g}=(g_1,\cdots g_n),\boldsymbol{h}=(h_1,\cdots h_n),u\xleftarrow{\$}\mathbb{G}$$

$$Com(\boldsymbol{a}_1^n,\boldsymbol{a}_2^n,\langle\boldsymbol{a}_1,\boldsymbol{a}_2\rangle)=(\boldsymbol{g}^{\boldsymbol{a}_1}\cdot\boldsymbol{h}^{\boldsymbol{a}_2}u^{\langle\boldsymbol{a}_1,\boldsymbol{a}_2\rangle})=\prod_i g_i^{a_{1i}}\cdot\prod_i g_i^{a_{2i}}u^{\langle\boldsymbol{a}_1,\boldsymbol{a}_2\rangle}\in\mathbb{G}$$

证明端 P 构造向量内积承诺后，设 $n'=n/2$，并进行如下计算：

$$c_L=\langle\boldsymbol{a}_{1[:n']},\boldsymbol{a}_{2[n':]}\rangle;c_R=\langle\boldsymbol{a}_{1[n':]},\boldsymbol{a}_{2[:n']}\rangle$$

$$L=\boldsymbol{g}_{[n':]}^{\boldsymbol{a}_{1[:n']}}\cdot\boldsymbol{h}_{[n':]}^{\boldsymbol{a}_{2[:n']}}u^{c_L};R=\boldsymbol{g}_{[n':]}^{\boldsymbol{a}_{1[n':]}}\cdot\boldsymbol{h}_{[n':]}^{\boldsymbol{a}_{2[:n']}}u^{c_R}$$

$$\boldsymbol{g}'=\boldsymbol{g}_{[:n']}^{x^{-1}}\circ\boldsymbol{g}_{[n':]}^{x}\in\mathbb{G};\boldsymbol{h}'=\boldsymbol{h}_{[:n']}^{x^{-1}}\circ\boldsymbol{h}_{[n':]}^{x}\in\mathbb{G}$$

$$P'=L^{x^2}PR^{x^{-2}}\in\mathbb{G}$$

$$\boldsymbol{a}'=\boldsymbol{a}_{[:n']}\cdot x+\boldsymbol{a}_{[n':]}\cdot x^{-1}\in\mathbb{Z}_p^{n'}$$

$$\boldsymbol{b}'=\boldsymbol{b}_{[:n']}\cdot x^{-1}+\boldsymbol{b}_{[n':]}\cdot x\in\mathbb{Z}_p^{n'}$$

随后，令 $n=n';\boldsymbol{a}=\boldsymbol{a}';\boldsymbol{b}=\boldsymbol{b}';\boldsymbol{g}=\boldsymbol{g}';\boldsymbol{h}=\boldsymbol{h}'$迭代计算，直至 $n=1$。验证端 V 计算式 $P=g^ah^bu^c$ 是否成立。

5 计算成本恒定的通用高效范围证明

本节提出一种计算成本恒定且浮点数、整数类型通用的高效范围证明方案——ZKFERP，并阐述 ZKFERP 的具体执行过程。在提出 ZKFERP 之前，5.1 节首先为 ZKFERP 设计了一种改进的零知识协议，以优化运算性能；5.2 节提出一种内积向量生成方法，设计最大平方拆分算法和向量维度分化算法，为 ZKFERP 分别解决计算成本恒定的拉格朗日平方和

搜索问题和内积向量维度压缩问题；5.3 节应用改进的零知识协议，并结合内积向量生成方法，提出 ZKFERP 范围证明的具体实施方案。

5.1 改进的零知识证明协议

本节在基于向量内积承诺、多项式承诺和拉格朗日四平方和定理思想的基础上，构造出了新的协议结构和验证方案，提出了一种改进的零知识协议。

协议的目的是向验证端 V 证明证明端 P 掌握了一个非负整数 v，而不揭示 v 的值。新协议可分为两部分，第一部分是设计一种差分承诺结构，构造关于多元二次算数表达式的向量内积承诺与差分承诺，以代替直接构造表达式值的 Pedersen 向量承诺，目的是承诺秘密值 v 的计算有效性和不可篡改性而不揭示 v；第二部分是利用拉格朗日平方和定理构造关于 $v\geqslant 0$ 的向量内积承诺，并转化为多项式承诺，目的是承诺秘密值的非负性而不揭示 v。最后，协议在验证端设计新的承诺验证方案。

协议首先是面向整数的，证明端 P 将整数 v 拆分为一组不等的向量，使这组向量做内积等于 v，拆分过程是随机且不唯一的。随后，生成采用关于向量组的向量内积承诺以绑定向量拆分结果。同时，利用关于 v 的向量组和随机值生成差分承诺。差分承诺与向量内积承诺的维度相同，唯一的不同之处在于承诺的矢量项，向量内积承诺矢量项绑定 v，而差分承诺矢量项绑定随机值，具体过程如下所示：

$$v=\langle \boldsymbol{v}_1,\boldsymbol{v}_2\rangle \wedge \boldsymbol{v}_1\neq \boldsymbol{v}_2 \tag{1}$$

$$V=h^v\cdot \boldsymbol{g}^{\boldsymbol{v}_1}\cdot \boldsymbol{h}^{\boldsymbol{v}_2}\wedge h\in \mathcal{G}_p\wedge \boldsymbol{g},\boldsymbol{h},V\in \mathcal{G}_p^4 \tag{2}$$

$$\varphi \xleftarrow{\$} \mathbb{Z}_p \tag{3}$$

$$\hat{V}=g^{\varphi}\cdot \boldsymbol{g}^{\boldsymbol{v}_1}\cdot \boldsymbol{h}^{\boldsymbol{v}_2}\wedge g\in \mathcal{G}_p\wedge \boldsymbol{g},\boldsymbol{h},V\in \mathcal{G}_p^4 \tag{4}$$

式(1)中的 $\boldsymbol{v}_1$ 和 $\boldsymbol{v}_2$ 表示关于 v 的不等向量组，式(2)中 V 表示关于 v 的向量内积承诺；式(3)中 $\hat{V}$ 表示关于 v 的差分承诺，差分承诺的构造形式与 Pedersen 向量承诺类似，其中 φ 表示在模为 p 的整数环中选取的随机值；式(4)中 g 和 h 表示两个循环群 $\mathbb{G}$ 产生的点，$\boldsymbol{g}$ 和 $\boldsymbol{h}$ 表示 g 和 h 张成的维度为 4 的向量。若将 $\boldsymbol{v}_1$ 和 $\boldsymbol{v}_2$ 的向量元素形式化为关于 v 的表达式元素，则通过向量内积承诺与差分承诺的构造，可以隐藏绑定关于 v 的任意多元二次表达式，而无须通过 v 的 Pedersen 承诺来构建关于 v 的表达式承诺。

在完成第一部分的构造后，证明系统构造关于 $v\geqslant 0$ 的向量内积承诺。具体地，根据拉格朗日平方和定理，可以针对 v 构造出等价关系，如式(5)所示：

$$\exists \{a_i\mid i\in\{1,2,3,4\}\wedge a_i\in \mathbb{Z}\},\sum_{i=1}^{4}a_i{}^2=v\Leftrightarrow v\geqslant 0 \tag{5}$$

其中，a_i 分别表示为 4 个整数，当且仅当存在 4 个整数使其平方和等于 v 时，$v\geqslant 0$ 成立。虽然文献[25]阐述当 $v-1$ 是 4 的整数倍时，仅需 3 个整数平方和即可表示 v。但当且仅当向量维度为 2 的整数次幂时，向量内积承诺的验证协议(定义 8 呈现)才可以递归执行，因此本节采用的拉格朗日平方和维度仍为 4。

根据 v 的拉格朗日平方和表示法，构造关于 v 的内积，构造形式如式(6)、式(7)所示：

$$a_1^2+a_2^2+a_3^2+a_4^2=v \tag{6}$$

$$\begin{cases}\boldsymbol{a}=(a_1,a_2,a_3,a_4)\\ \langle \boldsymbol{a},\boldsymbol{a}\rangle=v\end{cases} \tag{7}$$

其中，$\boldsymbol{a}$ 表示关于 v 的拉格朗日内积向量，因此 $\boldsymbol{a}$ 有别于 $\boldsymbol{v}_1$ 和 $\boldsymbol{v}_2$。对于不同的 v，向量 $\boldsymbol{a}$ 的维度并不一致。例如，当 v 本身为平方数时，其内积向量 $\boldsymbol{a}$ 的维度仅为 1。因此，本文在 3.2 节中引入向量维度分化算法，使证明系统只取能够表示 v 的最小维度向量 $\boldsymbol{a}$。

计算得到 $\boldsymbol{a}$ 后，为了不泄露 v 值，证明系统不能直接将 $\boldsymbol{a}$ 发送给验证端 V。因此，证明端 P 需要构造关于向量 $\boldsymbol{a}$ 的 Pedersen 向量承诺。

$$\alpha \xleftarrow{\$} \mathbb{Z}_p \tag{8}$$

$$A=h^{\alpha}\boldsymbol{g}^{\boldsymbol{a}}\boldsymbol{h}^{\boldsymbol{a}}\wedge A\in \mathcal{G}_p \tag{9}$$

式(9)中，A 代表对向量 $\boldsymbol{a}$ 构造的 Pedersen 向量承诺，A 与差分承诺 $\hat{V}$ 的构造形式类似，但其使用 $\boldsymbol{g}$ 和 $\boldsymbol{h}$ 所绑定的两个向量是相同的。式(8)中，α 代表承诺 A 选取的随机数。

根据多项式承诺，证明端 P 构造一组关于向量 $\boldsymbol{a}$ 的一元一次多项式 $l(X)$ 和 $r(X)$，并计算关于 $l(X)$ 和 $r(X)$ 的内积多项式 $t(X)$。由于证明端 P 不能直接将多项式 $l(X)$ 和 $r(X)$ 发送至验证端 V，因此证明系统引入一组向量盲化因子使多项式盲化，并构造关于盲化因子的 Pedersen 向量承诺，具体过程如下：

$$\boldsymbol{s}_L,\boldsymbol{s}_R \xleftarrow{\$} \mathbb{Z}_P^4 \tag{10}$$

$$z \xleftarrow{\$} \mathbb{Z}_P^* \tag{11}$$

$$\begin{cases}l(X)=\boldsymbol{a}\cdot z+\boldsymbol{s}_L\cdot X\in \mathbb{Z}_P[X]\\ r(X)=\boldsymbol{a}\cdot z+\boldsymbol{s}_R\cdot X\in \mathbb{Z}_P[X]\end{cases} \tag{12}$$

$$\begin{aligned}t(X)&=\langle l(X),r(X)\rangle\\ &=\langle \boldsymbol{a},\boldsymbol{a}\rangle\cdot z^2+(z\cdot\langle \boldsymbol{a},\boldsymbol{s}_L\rangle+z\cdot\langle \boldsymbol{a},\boldsymbol{s}_R\rangle)\cdot X+\\ &\quad\langle \boldsymbol{s}_L,\boldsymbol{s}_R\rangle\cdot X^2\\ &=t_0+t_1\cdot X+t_2\cdot X^2\in \mathbb{Z}_P[X]\end{aligned} \tag{13}$$

$$\rho \xleftarrow{\$} \mathbb{Z}_P \tag{14}$$

$$S=h^{\rho}\boldsymbol{g}^{\boldsymbol{s}_L}\boldsymbol{h}^{\boldsymbol{s}_R}\wedge S\in \mathcal{G}_p \tag{15}$$

$$P\rightarrow V:A,S \tag{16}$$

式(10)中的 $\boldsymbol{s}_L$ 和 $\boldsymbol{s}_R$ 表示在整数环上选取的向量盲化因子，分别适配 $l(X)$ 与 $r(X)$；式(15)中的 S 表示向量盲化承诺；式(11)中的 z 表示在整数环上选取的非零随机数。$t(X)$ 是由随机数 z、向量 $\boldsymbol{a}$、盲化因子 $\boldsymbol{s}_L$ 和 $\boldsymbol{s}_R$ 以及未知量 X 组成的二元一次多项式，多项式系数分别为 t_0,t_1,t_2。由于 $l(X)$ 和 $r(X)$ 是特殊设计的，因此当且仅当 $l(X)$ 和 $r(X)$ 被正确计算时，$t(X)$ 常数项 t_0 的数值表达式可表示为：

$$t_0=\langle \boldsymbol{a},\boldsymbol{a}\rangle\cdot z^2=v\cdot z^2 \tag{17}$$

根据多项式承诺，证明端 P 构造关于非零次项系数的 Pedersen 承诺，并生成系数随机数多项式与承诺随机数多项式。随后在整数环上生成非零随机数 x 以计算出多项式 $l(X)$ 和 $r(X)$ 的数值结果。至此，证明端 P 完成全部证据的计算，并将证据集发送至验证端，具体过程如下：

$$\tau_1,\tau_2 \xleftarrow{\$} \mathbb{Z}_p \tag{18}$$

$$T_i=g^{\tau_i}h^{t_i},i=\{1,2\}\in \mathcal{G}_P \tag{19}$$

$$P\rightarrow V:T_1,T_2 \tag{20}$$

$$V:x \xleftarrow{\$} \mathbb{Z}_p^* \tag{21}$$

$$\tau_x=\tau_1\cdot x+\tau_2\cdot x+\varphi\cdot z^2\in\mathbb{Z}_p \tag{22}$$

$$\mu=\alpha\cdot z+\rho\cdot x\in\mathbb{Z}_p \tag{23}$$

$$\begin{cases}\boldsymbol{l}=l(x)=\boldsymbol{a}\cdot z+\boldsymbol{s}_L\cdot x\\ \boldsymbol{r}=r(x)=\boldsymbol{a}\cdot z+\boldsymbol{s}_R\cdot x\end{cases} \tag{24}$$

$$P\rightarrow V:\tau_x,\mu,\boldsymbol{l},\boldsymbol{r} \tag{25}$$

式(18)中的 τ_1 和 τ_2 为非零系数 t_1 和 t_2 的承诺随机数；式(19)中的 T_1 和 T_2 为关于 t_1 和 t_2 的 Pedersen 承诺；式(22)中的 τ_x 表示系数随机数多项式，由 τ_1、τ_2 差分承诺 $\hat{V}$ 对应的随机数 φ 和 x 计算得来；式(23)中的 μ 表示承诺随机数多项式，由承诺 A 对应的随机数 α 和承诺 S 对应的随机数 ρ 以及 z 和 x 计算得来；式(24)中 l 与 r 表示将 x 代入 $l(X)$ 和 $r(X)$ 分别计算得到的数值结果。

验证端 V 获取证据集并构造验证式。当且仅当式(26)和式(27)验证成立时，验证成功，即验证端 V 相信证明端 P 掌握非负整数 v；否则验证失败。

$$\hat{V}^{z^2}\cdot g^{\tau_x}h^t=V^{z^2}\cdot T_1^x\cdot T_2^{x^2} \tag{26}$$

$$h^{\mu}\cdot\boldsymbol{g}^{\boldsymbol{l}}\cdot\boldsymbol{h}^{\boldsymbol{r}}=A^z\cdot S^x\in\mathcal{G} \tag{27}$$

上述零知识证明方案为交互式方案。为将方案改造为非交互式证明方案，证明系统根据 Fiat-Shamir 启发式，即以本地哈希函数生成随机数的方式代替验证端向证明端交互发送随机数，以更好地应用于区块链中。

5.2 内积向量生成方法

在实际链下计算中，依靠 5.1 节提出的零知识协议并不能完全实现证明时间和计算成本的恒定，因为对式(6)中关于 v 的拉格朗日内积向量 $\boldsymbol{a}$ 求解十分困难。传统的关于 v 的拉格朗日平方和求解算法需要使用枚举法遍历数值域 $[1,\sqrt{v}]$，因此其时间复杂度为 $O(n^2)$，即使根据文献[25]对 v 进行分解优化，其时间复杂度仍为 $O(n)$。当 v 的位数过大时，采用枚举法计算拉格朗日平方和所产生的时间开销将非常大。因此，证明系统需要一种计算成本恒定的拉格朗日内积向量生成方法，即内积向量搜索时间复杂度为 $O(1)$。

本节设计了一种最大平方拆分算法以解决恒定时间内的拉格朗日平方和搜索问题。具体地，对于给定的非负整数 v，计算小于且最接近于 v 的平方数 a，即求解拉格朗日平方和内积向量 $\boldsymbol{a}$。令 $v=v_0$，令向量 $\boldsymbol{a}$ 的元素 $a_i\in\{a_i\mid\max(v_i-a_i)\wedge\sqrt{a_i}\in\mathbb{N}^*\}$。随后执行式(28)并迭代计算直至 $v=0$ 结束。

$$v_{i+1}=v_i-a_i,i\in\{0\cdots,n\} \tag{28}$$

其中，n 表示迭代计算次数，即拉格朗日内积向量维度。每一次迭代计算中求解的 a_i 组成关于 v 的拉格朗日内积向量 $\boldsymbol{a}=(a_1,\cdots,a_n)$。因为迭代计算的次数不固定，所以仅依靠 4 次迭代运算无法完全将内积向量 $\boldsymbol{a}$ 的维度限制为 4。

若将上述迭代过程记作函数 $x_{n+1}=f(x_n)\wedge f(x)=\sqrt{x}$，且令 $x_0=v_0$，则函数 $f(x)$ 为二阶收敛，当 $f(x)$ 经过 4 次迭代后，非零的 v_3 会远小于 v_0，因此 v_3 的拉格朗日内积向量求解难度会远小于 v_0。当迭代次数大于 4 时，最大平方拆分算法单独对 v_3 求解，从而限制了向量维度。具体地，针对 v_3 构建不可拆分字典，不可拆分字典的键主要存储无法在 4 次迭代运算内解出拉格朗日内积向量的 v 值，而字典的值则存储每个 v 对应的拉格朗日内积向量。v_3 首先根据不可拆分字典索引键查找对应的内积向量，若查询成功，则直接返回字典值；若查询失败，则继续利用最大平方拆分求解。若 v_3 在不可拆分字典中查询失败，则利用最大平方拆分法计算一定有解，且解的维度小于或等于 4。在实际运算中，可根据 v 值来扩充不可拆分字典的维度。最后，得到关于 v_3 的拉格朗日内积向量表示为 $\boldsymbol{b}=(b_1,\cdots,b_m)$，$\boldsymbol{b}$ 的维度应与 $\boldsymbol{a}$ 相同，若不一致，则补充 0 元素。至此，v 的拉格朗日内积关系如式(29)所示：

$$v=\langle\boldsymbol{a},\boldsymbol{a}\rangle+\langle\boldsymbol{b},\boldsymbol{b}\rangle=\sum_{i=1}^{n}a_i^2+\sum_{j=1}^{n}b_j^2 \tag{29}$$

完整的最大平方拆分算法如算法 1 所示。

算法 1　最大平方拆分算法

```
输入：v
输出：a,b
1. tp_list ← []
2. tp_v← v
3. ns_dic← {Non-split dictionary}
4. / * 初始化不可拆分字典 * /
5. for i=0 to 3 do / * 求解内积向量 a * /
6.    sqrt_tp_v ← int(sqrt(tp_v))
7.    tp_list[i] ← sqrt_tp_v
8.    if power(sqrt_tp_v)==tp_v then
9.       a← tp_list
10.         b←Ø
11.      return a,b
12. end if
13. if i==3 then / * 判断内积向量 a 维度是否已达上限 * /
14.    a← tp_list
15.    tp_v=tp_v - power(sqrt_tp_v)
16.    if tp_v in ns_dic then
17.       b← ns_dic[tp_v]
18.       return a,b
19.    for j=0 to 3 do / * 求解内积向量 b * /
20.       sqrt_tp_v ← int(sqrt(tp_v))
21.       tp_list[j] ← sqrt_tp_v
22.       if power(sqrt_tp_v)==tp_v then
23.          b← tp_list
24.          return a,b
25.          end if
26.       tp_v=tp_v−power(sqrt_tp_v)
27.    end for
28. end if
29. tp_v=tp_v-power(sqrt_tp_v)
30. end for
```

为进一步缩减平均向量维度，压缩平均运算时间，证明系统对内积向量 $\boldsymbol{a},\boldsymbol{b}$ 进行维度分化，通过删除向量 0 元素，在保证 $\boldsymbol{a}$ 与 $\boldsymbol{b}$ 维度相同的同时，使 $\boldsymbol{a},\boldsymbol{b}$ 的维度 n 从 $n=4$ 压缩分化为 $n\in\{1,2,4\}$。其中，由于向量内积承诺的验证过程是令维度 $n=n/2$ 递归执行的，因此 a,b 压缩后维度不可为 3。向量维度分化算法的具体流程如算法 2 所示。

算法 2　向量维度分化算法

```
输入：a,b
输出：a,b
1. vec_len← 0
2. if b==Ø then / * 判断向量 b 是否为空 * /
3.    if len(a)==3 / * 当向量 a 维度为 3 时，整体内积向量维度应为
      4 * /
```

```
4.      vec_len ← 4
5.   else
6.      vec_len ← len(a)
7.   end if
8. else
9.      vec_len ← 4
10. end if
11. for i=0 to 4-vec_len do
12.  a.remove() /* 删除向量 a 最后一个元素 */
13.  b.remove() /* 删除向量 b 最后一个元素 */
14. end for
15. return a,b
```

在算法 2 中，当 $b\neq\emptyset$ 时，有 $a_i\neq 0, i\in\{1,2,3,4\}$，即 $\boldsymbol{a}$ 和 $\boldsymbol{b}$ 维度不可压缩。

5.3 ZKFERP 方案具体描述

本节阐述 ZKFERP 方案的整体执行过程和核心技术细节。ZKFERP 首先设计一种数据类型通用的范围关系式，并应用改进的零知识证明协议构造关于范围关系式的向量内积承诺与差分承诺，之后应用内积向量生成方法构造计算成本恒定且维度分化的拉格朗日内积向量，并针对内积向量构造 Pedersen 向量承诺以及多项式承诺，从而在计算成本和时间开销恒定的情况下，实现对浮点数、整数通用的范围证明。

具体地，任意浮点数灵活非负范围区间关系可表示为：

$$m\in[d_l,d_u]\wedge d_l>0\wedge d_u>0 \tag{30}$$

$$m-d_l>0\wedge d_u-m>0\wedge d_l>0\wedge d_u>0 \tag{31}$$

其中，m 表示浮点数证明值，d_l 和 d_u 分别代表浮点数区间上下界，非负范围区间当且仅当 $m-d_l>0$ 时，$d_u-m<0$ 成立；相反，当 $m-d_l<0$ 时，$d_u-m>0$ 成立。因此，上述范围关系可转化为：

$$m\in[d_l,d_u]\wedge d_l>0\wedge d_u>0\Leftrightarrow (m-d_l)(d_u-m)>0\wedge d_l>0\wedge d_u>0 \tag{32}$$

对于浮点数范围关系，取区间上下界 d_l 和 d_u 与证明值 m 的浮点数小数位数最大值作为 10 的幂数，并同时乘以区间界值与证明值，范围关系依然成立。

$$w\in\{10^{f_{\max}}\mid f_{\max}=\max(f_m,f_{dl},f_{du})\} \tag{33}$$

$$m\in[d_l,d_u]\wedge d_l>0\wedge d_u>0\Leftrightarrow(w\cdot m-w\cdot d_l)(w\cdot d_u-w\cdot m)>0\wedge d_l>0\wedge d_u>0\Leftrightarrow w^2\cdot(m\cdot d_u-m^2-d_l\cdot d_u+d_l\cdot m)>0\wedge d_l>0\wedge d_u>0 \tag{34}$$

式(33)中，f_m，f_{dl}，f_{du} 分别表示 m，d_l，d_u 的小数位数。$f_{\max}$ 表示 f_m，f_{dl}，f_{du} 中的最大值，w 表示 10 的 $f_{\max}$ 次方。式(34)中，w^2 为平方数，因此式(34)中的等价关系成立，即推导得到的浮点数范围关系式。令 $f_{\max}=0$，则式(34)可表示为整数范围关系式，此时 $w=1$。证明系统通过计算 $f_{\max}$，以实现浮点数与整数范围证明之间的切换。

证明端 P 利用范围关系式构造向量内积承诺与差分承诺，构造方式如下：

$$\begin{cases}\boldsymbol{v}_1=(w\cdot m,-w\cdot m,-w\cdot d_l,w\cdot d_l)\\ \boldsymbol{v}_2=(w\cdot d_u,w\cdot m,w\cdot d_u,w\cdot m)\end{cases} \tag{35}$$

$$v=\langle\boldsymbol{v}_1,\boldsymbol{v}_2\rangle \tag{36}$$

$$V=h^v\boldsymbol{g}^{\boldsymbol{v}_1}\boldsymbol{h}^{\boldsymbol{v}_2}\wedge\boldsymbol{g},\boldsymbol{h},V\in\mathcal{G}_p^4 \tag{37}$$

$$\hat{V}=g^{\varphi}\cdot\boldsymbol{g}^{\boldsymbol{v}_1}\boldsymbol{h}^{\boldsymbol{v}_2}\wedge\varphi\overset{\$}{\leftarrow}\mathcal{Z}_p\wedge\boldsymbol{g},\boldsymbol{h},V\in\mathcal{G}_p^4 \tag{38}$$

$$P\rightarrow V:V,\hat{V} \tag{39}$$

证明系统将关系式拆分为两组由 $w.m$，$w.a$，$w.b$ 构造的向量 $\boldsymbol{v}_1$ 和 $\boldsymbol{v}_2$，并构造关于 $\boldsymbol{v}_1$ 和 $\boldsymbol{v}_2$ 的向量内积承诺 V 与 Pedersen 向量承诺 $\hat{V}$，并将 $\hat{V}$ 作为差分承诺。

范围关系式转化为拉格朗日平方和可表示为：

$$w^2(m-d_l)(d_u-m)=\langle\boldsymbol{a},\boldsymbol{a}\rangle+\langle\boldsymbol{b},\boldsymbol{b}\rangle=\sum_{i=1}^{n}a_i^2+\sum_{j=1}^{n}b_{ii}^2,n\in\{1,2,4\} \tag{40}$$

证明系统利用内积向量生成方法求得关于浮点数范围关系式且维度分化的拉格朗日内积向量 $\boldsymbol{a}=(a_1,\cdots,a_n)$ 与 $\boldsymbol{b}=(b_1,\cdots,b_n)$。

随后，证明端 P 构造内积向量 $\boldsymbol{a}$ 与 $\boldsymbol{b}$ 的 Pedersen 向量承诺，并选取盲化因子构造对应的承诺，具体过程如下：

$$\alpha\overset{\$}{\leftarrow}\mathcal{Z}_p \tag{41}$$

$$A=h^{\alpha}\boldsymbol{g}^{\boldsymbol{a}}\boldsymbol{h}^{\boldsymbol{a}}\wedge A\in\mathcal{G}_P \tag{42}$$

$$\varepsilon\overset{\$}{\leftarrow}\mathcal{Z}_p \tag{43}$$

$$B=h^{\varepsilon}\boldsymbol{g}^{\boldsymbol{b}}\boldsymbol{h}^{\boldsymbol{b}}\wedge B\in\mathcal{G}_P \tag{44}$$

$$\rho\overset{\$}{\leftarrow}\mathcal{Z}_p \tag{45}$$

$$\boldsymbol{s}_L,\boldsymbol{s}_R\overset{\$}{\leftarrow}\mathcal{Z}_P^n \tag{46}$$

$$S=h^{\rho}\boldsymbol{g}^{\boldsymbol{s}_L}\boldsymbol{h}^{\boldsymbol{s}_R}\wedge S\in\mathcal{G}_p \tag{47}$$

$$P\rightarrow V:A,B,S \tag{48}$$

其中，α,ε,ρ 表示在模为 p 的整数环上生成的承诺随机点，因此 A 和 B 的空间域也在整数环上。由于向量 $\boldsymbol{a}$ 和 $\boldsymbol{b}$ 经过维度分化，因此盲化向量 $\boldsymbol{s}_L,\boldsymbol{s}_R$ 维度是可变的。

完成 $\boldsymbol{a},\boldsymbol{b}$ 的承诺构造后，根据多项式承诺构造方法，证明端 P 构造多项式 $l(X)$ 和 $r(X)$ 并计算 $\boldsymbol{l},\boldsymbol{r}$ 以及 t，构造及计算过程如下：

$$\boldsymbol{l}=l(x)=\boldsymbol{a}\cdot z+\boldsymbol{b}\cdot z+\boldsymbol{s}_L\cdot x\in\mathcal{Z}_P^n \tag{49}$$

$$\boldsymbol{r}=r(x)=\boldsymbol{a}\cdot z+\boldsymbol{b}\cdot z+\boldsymbol{s}_R\cdot x\in\mathcal{Z}_P^n \tag{50}$$

$$\begin{aligned}t&=\langle\boldsymbol{l},\boldsymbol{r}\rangle\\&=(\langle\boldsymbol{a},\boldsymbol{a}\rangle+\langle\boldsymbol{b},\boldsymbol{b}\rangle+2\cdot\langle\boldsymbol{a},\boldsymbol{b}\rangle)\cdot z^2+(\langle\boldsymbol{a}\cdot z+\boldsymbol{b}\cdot z,\boldsymbol{s}_L\rangle+\langle\boldsymbol{a}\cdot z+\boldsymbol{b}\cdot z,\boldsymbol{s}_R\rangle)\cdot x+\langle\boldsymbol{s}_L,\boldsymbol{s}_R\rangle\cdot x^2\\&=(\langle\boldsymbol{a},\boldsymbol{a}\rangle+\langle\boldsymbol{b},\boldsymbol{b}\rangle)\cdot z^2+2\cdot\langle\boldsymbol{a}\cdot z,\boldsymbol{a}\cdot z\rangle+(\langle z\cdot(\boldsymbol{a}+\boldsymbol{b}),\boldsymbol{s}_L\rangle+\langle z\cdot(\boldsymbol{a}+\boldsymbol{b}),\boldsymbol{s}_R\rangle)\cdot x+\langle\boldsymbol{s}_L,\boldsymbol{s}_R\rangle\cdot x^2\end{aligned} \tag{51}$$

$$\begin{aligned}t_0&=(\langle\boldsymbol{a},\boldsymbol{a}\rangle+\langle\boldsymbol{b},\boldsymbol{b}\rangle)\cdot z^2+2\cdot\langle\boldsymbol{a}\cdot z,\boldsymbol{b}\cdot z\rangle\\&=v\cdot z^2+2\cdot\langle\boldsymbol{a}\cdot z,\boldsymbol{b}\cdot z\rangle\end{aligned} \tag{52}$$

其中，t 为 $\boldsymbol{l}$ 和 $\boldsymbol{r}$ 作内积得到的一元二次多项式。证明端 P 可根据 Fiat-Shamir 启发式，利用承诺 A,B 与 S 的哈希函数 $H(A,B,S)$ 生成多项式随机数 z，从而取代验证端生成随机数的方式，实现非交互式证明的目的。

随后，证明端 P 计算如下：

$$\lambda,\tau_1,\tau_2\overset{\$}{\leftarrow}\mathcal{Z}_p \tag{53}$$

$$T_i=g^{\tau_i}h^{t_i}i=\{1,2\}\in\mathcal{G}_p \tag{54}$$

$$\tau_x=\tau_1\cdot x+\tau_2\cdot x^2-z^2\cdot\varphi+2z^2\cdot\lambda\in\mathcal{Z}_p \tag{55}$$

$$\mu=\alpha\cdot z+\varepsilon\cdot z+\rho\cdot x\in\mathcal{Z}_p \tag{56}$$

$$K=g^{\lambda}\cdot h^{\langle\boldsymbol{a},\boldsymbol{b}\rangle}\in\mathcal{G}_p \tag{57}$$

$$P\rightarrow V:T_1,T_2,\mu,\tau_x,K,\boldsymbol{l},\boldsymbol{r} \tag{58}$$

其中，在完成 T_i,τ_x,μ 的构造后，证明系统构造关于 $\langle\boldsymbol{a},\boldsymbol{b}\rangle$ 的 Pedersen 承诺 K，以保证 $\boldsymbol{a}$ 和 $\boldsymbol{b}$ 在验证端的零知识性，

构造形式如式(56)所示。

证明端 P 将证据集一并发送至验证端 V。验证端 V 验证式(53)、式(54)是否成立：

$$\hat{V}^{z^2} \cdot g^{\tau_z} h^t = V^{z^2} \cdot K^{z \cdot z^2} \cdot T_1^x \cdot T_2^{x^2} \tag{59}$$

$$h^{\mu} \cdot \boldsymbol{g}^{\boldsymbol{l}} \cdot \boldsymbol{h}^{\boldsymbol{r}} = A^z \cdot B^{z^2} \cdot S^x \in \mathcal{G} \tag{60}$$

当且仅当式(53)和式(54)成立时，验证成功；否则失败。

通过向量内积承诺以及多项式承诺的概念可以推论出 ZKFERP 具有完备性，即对于任何诚实的证明端 P，P 伪造证据使验证端 V 验证通过的概率小到可以忽略不计；ZKFERP 具有正确性，即对于任何验证端 V，P 提供有效证据而验证不通过的概率小到可以忽略不计；ZKFERP 具有零知识性，即验证端 V 在验证过程中无法获取任何关于 m，d_l，d_u 的额外知识。

6 实验结果与分析

本节评估了 ZKFERP 的理论效果与实验性能。为确保评估的针对性和有效性，本节将 ZKFERP 与近年来提出的基于向量内积承诺的其他先进范围证明方案进行了对比。

6.1 方案理论效果

本节讨论了 ZKFERP 与其他范围证明方案的通信成本对比与理论时间复杂度对比。

表 2 列出了 Bulletproofs，Cuproof 和 ZKFERP 使用的循环群元素和整数环元素数量，本节将问题数量设置为 1。ZKFERP 与 Bulletproofs 使用了一个阶为 p 的循环群，而 Cuproof 分别使用了阶为 p 和 q 的两个循环群。表 2 中，n 表示 Bulletproofs 处理的二进制位数，由于 Bulletproofs 按二进制来构造多项式，因此其循环群元素使用数量与二进制位数有关。而 ZKFERP 与 Cuproof 根据拉格朗日定理构造固定维度的内积向量，因此其通信成本是恒定的。ZKFERP 与 Cuproof 的循环群元素使用数量均为 2。由于 ZKFERP 构造差分承诺并利用最大平方拆分算法生成内积向量，因此其在整数环上的随机数取值比 Bulletproofs 多 2。

表 2 通信成本对比

Table 2 Communication cost comparison

Scheme	$\mathcal{G}_p$	$\mathbb{Z}_p$	$\mathcal{G}_q$	$\mathbb{Z}_q$
Bulletproofs	$2\log(n)+4$	5	0	0
Cuproof	2	4	2	3
ZKFERP	**2**	**7**	**0**	**0**

表 3 列出了 Bulletproofs，Cuproof 和 ZKFERP 的理论运算时间复杂度。其中 Bulletproofs 的证明时间、验证时间与生成见证时间复杂度是线性的；Cuproof 的证明时间、验证时间复杂度虽然为 O(1)，但生成见证时间是线性的，因此其整体运算时间复杂度依然是线性的；ZKFERP 的证明时间、验证时间与生成见证时间复杂度均为 O(1)。因此在理论上，ZKFERP 的整体运算时间复杂度最低，且整体计算成本恒定。

表 3 运算时间复杂度对比

Table 3 Computational time complexity comparison

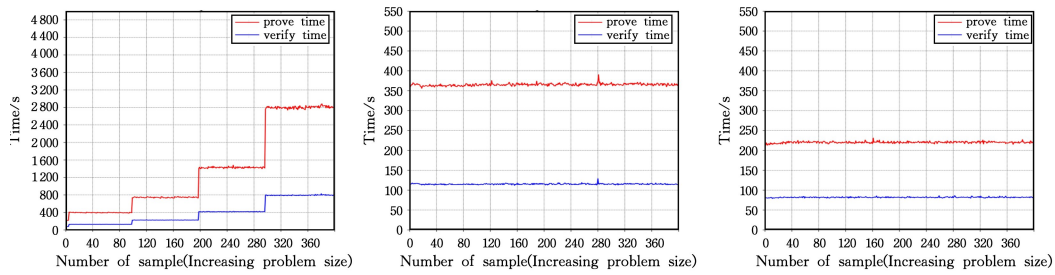
Scheme	Proof and verification time	Witness generation time	Total time
Bulletproofs	$O(n)$	$O(n)$	$O(n)$
Cuproof	O(1)	$O(n)$	$O(n)$
ZKFERP	**O(1)**	**O(1)**	**O(1)**

6.2 方案实验性能

本节将 ZKFERP 与其他先进的范围证明方案进行性能上的测试对比。为确保可参照性和有效性，实验采用 Python 语言对 ZKFERP，Bulletproofs 和 Cuproof 分别进行代码实现，并将源码上传到 Github 平台[1)]。实验环境采用由多节点组成的区块链分布式仿真系统，实验将证明端 P 部署于本机，验证端 V 部署于服务器节点；节点硬件条件采用 AMD 4800H 8 核心 16 线程处理器，16GB 运行内存；实验程序采用单线程运行。

在对比实验中，采用 secp256k12 椭圆曲线作为循环群；采用问题大小由 8bit～128bit 递增的 400 组 0-2 的整数次幂范围区间样本、400 组灵活整数范围区间样本、400 组灵活浮点数、整数混合范围区间样本，分别测试 Bulletproofs，Cuproof 和 ZKFERP 的性能。

Bulletproofs，Cuproof 和 ZKFERP 在 400 个区间测试样本中的运算时间与样本问题大小的关系如图 1 所示。图像中的蓝色线条表示证明时间；红色线条表示验证时间，横轴表示样本序号，样本问题大小由 8bit 随样本序号的递增而提高至 128bit。由于 Bulletproofs 采用二进制单比特构造多项式承诺，因此其承诺维度和运算时间随问题二进制比特位数的扩大而阶梯式增加。而 Cuproof 和 ZKFERP 基于拉格朗日平方和定理构造恒定维度的多项式承诺，在不考虑内积向量生成时间的情况下，其运算时间也是恒定的。ZKFERP 采用改进的协议结构来优化承诺构造形式，进一步压缩运算成本，证明时间和验证时间更优。



(a)Bulletproofs (b) Cuproof (c)ZKFERP

图 1 证明验证时间与问题大小关系(电子版为彩图)

Fig. 1 Relationship between proof and verification time and the size of problem

1) https://github.com/liyicong763/Ferproof for Python

实验分别对不同方案在不同问题大小下的性能测试结果进行采样，Bulletproofs 性能和证据大小如表 4 所列。由于 Bulletproofs 将样本问题处理为二进制单比特向量，因此其证据长度和运算时间随问题比特数的递增而增加，其平均证明时间为 1 191.85 ms，平均验证时间为 362.20 ms，平均证据大小为 622 bytes。Cuproof 性能和证据大小如表 5 所列，由于 Cuproof 采用拉格朗日平方和定理将样本问题处理为恒定维度的向量，因此其证据长度和运算时间与问题比特数无关，但 Cuproof 需要将样本问题拆解为两个证明任务并聚合，因此增加了时间开销，其平均证明时间为 365.20 ms，平均验证时间为 115.44 ms，平均证据大小为 2 090 bytes。ZKFERP 性能和证据大小如表 6 所列，ZKFERP 应用改进的协议结构，无须拆解样本问题即可构造证据，因此其平均证明时间为 219.27 ms，相比 Cuproof 和 Bulletproofs 分别缩短了 40.0% 和81.6%；平均验证时间为 81.01 ms，相比 Cuproof 和 Bulletproofs 分别缩短了 29.8% 和 77.6%；ZKFERP 平均证据大小为 739 bytes。实验结果表明，在证明时间与验证时间上，本文提出的 ZKFERP 均为最优。

表 4 Bulletproofs 性能和证据大小

Table 4 Performance and proof size of Bulletproofs

Problem size	Gate	ProofSize/bytes	Timing/ms	
			prove	verify
8 bit	8	490	375.59	136.03
16 bit	16	556	408.07	138.04
32 bit	32	622	758.23	229.03
64 bit	64	688	1437.23	416.09
128 bit	128	754	2980.12	891.83
Average		622	1191.85	362.20

表 5 Cuproof 性能和证据大小

Table 5 Performance and proof size of Cuproof

Problem size	Gate	ProofSize/bytes	Timing/ms	
			prove	verify
8 bit	6	2068	364.08	117.03
16 bit	6	2078	365.66	116.02
32 bit	6	2088	364.08	116.12
64 bit	6	2099	369.08	114.02
128 bit	6	2119	363.09	114.02
Average		2090	365.20	115.44

表 6 ZKFERP 性能和证据大小

Table 6 Performance and proof size of ZKFERP

Problem size	Gate	ProofSize/bytes	Timing/ms	
			prove	verify
8 bit	6	740	220.04	80.01
16 bit	6	739	219.04	82.01
32 bit	6	738	222.03	80.01
64 bit	6	740	216.13	82.02
128 bit	6	738	219.11	81.01
Average		739	219.27	81.01

为测试内积向量生成方法在计算成本上的效果，实验对比了文献[25]中的内积向量求解方法和本文的最大平方拆分方法随 v 值位数增加时的平方和求解搜索时间，实验对比结果如表 7 所列。最大平方拆分算法的迭代次数与样本问题大小无关，因此其迭代搜索时间是恒定的，而文献[25]方法的迭代次数随样本问题的扩大而增加，因此其搜索时间随 v 值位数的增加而增长。实验结果表明，应用内积向量生成方法的 ZKFERP 方案可以实现见证生成时间以及总体计算成本的恒定。

表 7 拉格朗日平方和求解搜索时间对比

Table 7 Comparison of search time for Lagrangian sum of squares solution

v digits (decimal)	Timing/ms	
	Enumeration	Maximum square split
10^3	0.99	0.00
10^4	7.14	0.00
10^5	35.58	0.00
10^6	206.77	0.00
10^7	923.71	0.00

结束语 本文提出了一种计算成本恒定且适用于浮点数和整数任意灵活范围区间的高效范围证明方案——ZKFERP。ZKFERP 利用最大平方拆分算法解决了恒定时间内的拉格朗日平方和搜索问题，实现了范围证明方案整体计算成本的恒定；并通过向量维度分化算法以及构造新的零知识协议结构使证明和验证速度更优。实验结果表明，ZKFERP 在实现计算成本恒定且适配浮点数的基础上，相比现有的范围证明方案，其证明时间和验证时间分别缩短了 40.0%和 29.8%。虽然 ZKFERP 在证据生成速度、验证速度和证据长度等方面性能突出，但在通信成本上仍有优化空间。在未来的研究中，将考虑优化协议结构以减少整数环随机数的选取。此外，应用 ZKFERP 实现聚合证明也将是下一步的研究重点。

参 考 文 献

[1] NAKAMOTO S. Bitcoin: A Peer-to-Peer Electronic Cash System[J]. Decentralized Business Review, 2008, 1: 2012.

[2] BONNEAU J, MILLER A, CLARK J, et al. Research Perspectives and Challenges for Bitcoin and Cryptocurrencies[C]// 2015 IEEE Symposium on Security and Privacy. San Jose: IEEE, 2015: 104-121.

[3] LIU M D, CHEN Z N, SHI Y J, et al. Research progress of blockchain in data security[J]. Chinese Journal of Computer, 2021, 44(1): 1-27.

[4] GOLDREICH O, OREN Y. Definitions and properties of zero-knowledge proof system[J]. Journal of Cryptology, 1994, 7(1): 1-32.

[5] NOETHER S, MACKENZIE A. Ring confidential transactions [J]. Ledger, 2016, 1: 1-18.

[6] BLUM M. Coin flipping by telephone a protocol for solving impossible problems [J]. ACM SIGACT News, 1983, 15(1): 23-27.

[7] MAXWELL G, POELSTRA A. Borromean ring signatures[J]. Accessed, 2015, 8: 2019.

[8] PEDERSEN T P. Non-interactive and information-theoretic secure verifiable secret sharing[C]// Annual International Cryptology Conference. Springer: Berlin, 1991: 129-140.

[9] NOETHER S. Ring signature confidential transactions for Monero[J]. IACR Cryptology ePrint Achieve, 2016, 1: 1-18.

[10] SUN S F, AU M H, LIU J K, et al. RingCT 2.0: A Compact Ac-

cumulator-Based(Linkable Ring Signature) Protocol for Blockchain Cryptocurrency Monero[C]// European Symposium on Research in Computer Security. Cham:Springer,2017:456-474.

[11] WOOD G. Ethereum:A secure decentralized transaction ledger [EB/OL]. http://gavwood.com/paper.pdf.

[12] MCCORRY P,SHAHANDASHTI S F,HAO F. A Smart Contract for Boardroom Voting with Maximum Voter Privacy [C]// International Conference on Financial Cryptography and Data Security. Cham:Springer,2017:357-375.

[13] CAMPANELLI M,GENNARO R,GOLDFEDER S,et al. Zero-knowledge contingent payments revisited:Attacks and payments for servicesy[C]// Conference on Computer and Communications Security. Dallas,Texas,USA:ACM,2017:229-243.

[14] PARNO B, HOWELL J, GENTRY C, et al. Pinocchio: Nearly practical verifiable computation[C]// Proceedings of the 34th IEEE Symposium on Security and Privacy. Berkeley,CA,USA: IEEE,2013:238-252.

[15] BEN-SASSON E,CHIESA A,GENKIN D,et al. SNARKs for C:Verifying program executions succinctly and in zero knowledge[C]// Annual Cryptology Conference. Berlin: Springer, 2013:90-108.

[16] PARNO B, HOWELL J, GENTRY C, et al. Pinocchio: Nearly practical verifiable computation[C]// 2013 IEEE Symposium on Security and Privacy. Berkeley,CA,USA:IEEE,2013:238-252.

[17] GROTH J. On the Size of Pairing-Based Non-interactive Arguments[C]// Annual International Conference on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 2016:305-326.

[18] QUESNELLE J. On the linkability of Zcash transactions[J]. arXiv:1712.01210,2017.

[19] WAHBY R S,TZIALLA I,SHELAT A,et al. Doubly-efficient zk SNARKs without trusted setup[C]// Proceedings of 2018 IEEE Symposium on Security and Privacy(SP). San Francisco, CA,USA:IEEE,2018:2375-1207.

[20] SETTY S,ANGEL S,LEE J. Verifiable state machines:Proofs that untrusted services operate correctly[J]. ACM SIGOPS Operating Systems Review,2020,54(1):40-46.

[21] BEN-SASSON E,BENTOV I,HORESH Y,et al. Scalable, transparent, and post-quantum secure computational integrity [J]. IACR Cryptology ePrint Archive,2018,46:1-83.

[22] MALLER M,BOWE S,KOHLWEISS M,et al. Sonic:Zero-knowledge SNARKs from linear-size universal and updatable structured reference strings[C]// Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security. New York:ACM,2019:2111-2128.

[23] GABIZON A, WILLIAMSON Z J, CIOBOTARU O. PLONK: Permutations over Lagrange-bases for Oecumenical Noninteractive arguments of Knowledge[J]. IACR Cryptol. ePrint Arch., 2019,953:1-35.

[24] LIPMAA H. On diophantine complexity and statistical zero-knowledge arguments[C]// International Conference on the Theory and Application of Cryptology and Information Security. Berlin:Springer,2003:398-415.

[25] GROTH J. Non-interactive zero-knowledge arguments for voting[C]// International Conference on Applied Cryptography and Network Security. Berlin, Heidelberg: Springer, 2005: 467-482.

[26] BUNZ B,BOOTLE J,BONEH D,et al. Bulletproofs:Short Proofs for Confidential Transactions and More[C]// 2018 IEEE Symposium on Security and Privacy. San Francisco,CA,USA: IEEE,2018:315-334.

[27] BOOTLE J,CERULLI A,CHAIDOS P,et al. Efficient zero-knowledge arguments for arithmetic circuits in the discrete log setting[C]// Annual International Conference on the Theory and Applications of Cryptographic Techniques. Berlin:Springer, 2016:327-357.

[28] DENG C,TANG X,YOU L,et al. Cuproof:A Novel Range Proof with Constant Size[J]. IACR Cryptol. ePrint Arch., 2021,13(2):127-137.

[29] FEIGE U,FIAT A,SHAMIR A. Zero knowledge proofs of identify[J]. Journal of Cryptology,1988,1(2):77-94.

[30] RIVEST R,SHAMIR A, ADLEMAN L. A method for obtaining digital signatures and public-key cryptosystems[J]. Communications of the ACM,1978,21(2):120-126.

[31] ALMUHAMMADI S,SUI N T,MCLEOD D. Better privacy and security in e-commerce: using elliptic curve-based zero knowledge proofs[C]// Proceedings IEEE International Conference on e-Commerce Technology. San Diego,CA,USA:IEEE, 2004:299-302.

LI Yi-cong, born in 1997, postgraduate. His main research interests include blockchain system security and cryptography theory and application.

ZHOU Kuan-jiu, born in 1966, Ph. D, professor, Ph. D supervisor, is a senior member of China Computer Federation. His main research interests include blockchain theory and technology and trusted software.

(责任编辑:何杨)