计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.250200125

基于轻量级区块链的低压用户需求响应方案

昌宁远[1]　黄　挺[2]　张　煌[1]
1 长沙理工大学计算机学院　长沙 410076
2 永州职业技术学院信息学院　湖南 永州 425100
(22108041625@stu.csust.edu.cn)

摘　要　智能电网的迅速发展提升了电力公司与电力用户之间的通信能力,也使得低压用户需求响应成为了极富潜力的智能电网业务。近年来,利用区块链技术强化智能电网功能的研究逐渐吸引了更多学者的关注。然而,区块链引发的能源消耗问题、用户隐私问题也成为无法避开的话题。基于轻量级区块链,提出了能源消耗低并带有用户数据隐私性的低压用户需求响应方案。为了解决低能耗与轻量级区块链结构带来的区块链公平性问题和区块链主链的数据安全性问题,还提出了非法操控监测、分布式哈希表、片区认证算法以支持方案的正常运行。云计算在智能电网中的应用发展迅速,可以最大限度地整合资源,解决智能电网的海量数据分布式计算问题。

关键词:云计算;智能电网;低压用户;需求响应;轻量级区块链;数据隐私性;低能源消耗
中图分类号　TP309

Demand Response Scheme for Low Voltage Users Based on Light Weight Blockchains

CHANG Ningyuan[1],HUANG Ting[2] and ZHANG Huang[1]
1 School of Computer Science and Technology,Changsha University of Science and Technology,Changsha 410076,China
2 Information Institute,Yongzhou Vocational Technical College,Yongzhou,Hunan 425100,China

Abstract　The rapid development of smart grids has enhanced the communication between power companies and electricity users, making demand response for low-voltage users a highly promising smart grid service. In recent years,efforts to strengthen smart grid functionalities using blockchain technology have increasingly drawn attention. However,issues such as energy consumption and user privacy concerns introduced by blockchain have become unavoidable. This paper proposes a low-voltage user demand response solution based on a lightweight blockchain,which features low energy consumption and ensures user data privacy. To address the fairness issues in blockchain and data security concerns of the main chain arising from the low energy consumption and the lightweight blockchain structure,the paper further introduces illegal manipulation monitoring,distributed hash tables,and regional authentication algorithms to support the solution's normal operation. The application of cloud computing in smart grids has also developed rapidly,as it can maximize resource integration and address the distributed computing challenges posed by the massive data in smart grids.

Keywords　Cloud computing,Smart grid,Low voltage users,Demand response,Light weight blockchains,Data privacy,Low energy consumption

1　引言

随着智能电网的进一步发展,能源互补、分布式供电、智能表计等各种辅助技术不断涌现。这些技术需要庞大的算力支撑,目前电网所有的算力不能满足各种辅助技术的实时处理需求,因而必须引入云计算技术。但是因为云计算自身的技术也存在安全隐患,即云服务提供商的不可信问题,而区块链具有去中心化、不可篡改和可追踪的特点,因此用区块链来解决智能电网中云计算的不可信问题是一个可靠的解决方式。融合了云计算的智能电网为电力公司与电力用户之间引入了相互通信的能力,为需要多方配合实施的负载均衡、电力营销等策略提供了可能性。在这种背景下,低压用户需求响应成为当前具有巨大潜力的智能电网业务。低压用户用电的自主性强、负荷调节灵活,是协助电网调节电力分配的优秀主体。针对低压用户需求响应的研究已获得业界的关注,以分时电价或实时市场进行激励的手段已取得良好的成效。然而,鉴于实时市场情况,以传统方法实施分时电价存在不少局限性,其中包括分时电价的调控不够精确,低压用户参与实时市场交易的渠道不够通畅。由此,通过区域自治的分布式区块链系统来支持实时市场与精准电价调控成为了更加可行的解决策略。Tian 等[1]给出了基于区块链的社区电能安全交易方案,用于考虑区块链对社区电能交易安全的帮助。Qin 等[2]尝试在微电网中引入区块链技术,以解决集中式电能交易方法存在的交易效率低、维护成本大、隐私性低、信息安全

基金项目:湖南省自然科学基金(2023JJ40054)
This work was supported by the Natural Science Foundation of Hunan Province(2023JJ40054).
通信作者:张煌(zhanghuang@csust.edu.cn)

程度低、信息透明度低等问题。Li 等[3]提出了通过区块链监控智能电网避免电力数据泄露风险的方案。Guan 等[4]设计了一种基于智能合约技术的分布式电力交易方案，其中首次使用了分布式哈希表与布隆过滤器来记录、检索与访问文件。

与传统的融入了云计算的智能电网系统类似，智能电表获取实时用电数据有利于电网实施预测并进行针对性调整，但存在数据泄露的风险，从而影响低压用户的隐私。如果这些数据被公布在区块链上，上述风险可能进一步加重。Liu 等[5]从匿名密码货币出发，在工作中指出了区块链存在的隐私保护问题，并提出了 4 层的技术架构。另一方面，由于区块链系统自治需要严格的共识机制支持，直接采用工作量证明(Proof of Work，POW)机制进行算力竞争将导致过度的能源消耗。基于此，Guo 等[6]提出了以联盟链完善电力交易的方案。虽然此种做法可以降低能源消耗，但其降低了区块链系统的自治性。

由此可见，在降低计算存储开销的同时维持高自治性和公平性是低压用户响应方案进一步实用化的关键。为此，本文构造了基于轻量级区块链的低压用户响应方案。首先，本文方案依然采用了 POW 为区块链共识机制，但算力由电力公司统一均等分配。为了搭建轻量级区块链，避免低压用户个体进行恶意算力竞争，本文方案引入了智能电表监控策略。对于恶意加载(非电网公司认可)高算力设备的用户，监控系统将通过智能合约降低其信誉度并将信誉度信息公布于区块链系统。其次，本文方案采用分布式存储以及分布式哈希表策略来分别降低信誉度信息访问对主权结点产生的负载和轻量级区块链结点实际的数据存储量。此外，本文方案还采用布隆过滤器的信息检索方式，进一步提升了信息检索的速率。最后，针对分时电价及实时市场产生的用户电量与积分的更新数据，本文提出了低压用户需求响应数据的秘密通信策略，以保护用户隐私。

2 预备知识

2.1 符号标识

本文以大写粗体字母表示常见的代数系统，如 $\boldsymbol{N}$，$\boldsymbol{Z}$，$\boldsymbol{Z}^+$，$\boldsymbol{G}$，$\boldsymbol{F}$ 分别对应自然数集合、整数集合、正整数集合、某一循环群和域。如果 a，b 为整数，则$[a,b]$表示集合$\{x \mid a \leqslant x \leqslant b\}$。有序集合$(x_1, x_2, \cdots, x_n)$将被简化写成$(x_i)_{i=1}^n$。如果 S 是一个有限集合，则表示元素 a 随机均匀地选取自 S。如果算法 B 是算法 A 的子算法，使用 $A.B$ 来代指对 A 的子算法 B 的调用。采用传统的渐进符号 O，即 $f(n) \leqslant O(g(n))$表示存在正实数 c 与实数n_0，使得当 $n \geqslant n_0$时，有 $f(n) \leqslant c \cdot g(n)$。

2.2 Pedersen 承诺

Pedersen 承诺包含一对概率多项式时间算法 COM=(Gen，Com)。Gen 算法用于生成承诺方案的系统参数，用户使用 Com 算法隐藏并固定一个消息在承诺中。Pedersen 承诺方案各算法的描述如下。

$ck \leftarrow \mathrm{Gen}(1^\lambda)$：该算法以安全参数 λ 为输入，然后根据安全参数随机生成以素数 $p=O(2^\lambda)$为阶的循环群 $\boldsymbol{G}$。随机均匀且独立地选择群元素 $G, H \leftarrow \boldsymbol{G}$。最后该算法输出 $ck=(G, H, p)$作为方案的系统参数并公开。ck 将作为 Com 算法的隐含输入。

$C \leftarrow Com(m, r)$：该算法以被承诺的消息 $m \in \boldsymbol{Z}_p$和一个随机值 $r \leftarrow \boldsymbol{Z}_p$为输入，然后计算并输出承诺 $C=mH+rG$。

承诺 C 的生成者后续可以通过揭晓 m 和 r 来打开承诺，以使任何用户可以通过验证等式来确认 C 的确为消息 m 的承诺。Pedersen 已证明 C 不会透露任何关于 m 的信息(完美隐藏性)，并且仅当承诺者知晓 H 关于 G 的离散对数时，其可以将 C 打开为另一个消息m'(计算绑定性)[7]。

2.3 环签名与数字签名方案

环签名方案包含一组概率多项式算法 RS=(Setup，KGen，Sign，Verify)。本小节介绍的环签名方案为 Jeong 等提出的可链接环签名的非链接版本[8]。

$pp \leftarrow \mathrm{Setup}(1^\lambda)$：以安全参数 λ 为输入，该算法随机生成以素数 $p=O(2^\lambda)$为阶的循环群 $\boldsymbol{G}=\langle G \rangle$，其中 G 为群的生成元，然后选定密码学哈希函数 $h:\{0,1\}^* \rightarrow \boldsymbol{Z}_p$。$pp=\{G, p, h\}$作为系统参数公开，其为其他算法的隐含输入。

$(pk, sk) \leftarrow K\mathrm{Gen}(\cdot)$：该算法被调用后，随机选取 $x \leftarrow \boldsymbol{Z}_p$作为用户的私钥 sk，然后计算 $Y=xG$ 作为对应的公钥 pk。(sk, pk)返回给算法的调用者。

$\sigma \leftarrow Sign(\mu, L, n, sk_\pi)$：该算法以消息 μ、包含 n 个公钥的集合 L 与集合中某个公钥的私钥 $sk_\pi=x_\pi$为输入，按照如下方式生成签名。不失一般性，假设 $\pi \in [1,n]$，$L=\{Y_1, \cdots, Y_n\}$。

1)随机均匀地选取 $u \leftarrow \boldsymbol{Z}_p$，并计算$R_\pi=uG$；

2)对于 $i \in [1,n]$且 $i \neq \pi$，随机均匀且独立地选取$c_i, s_i \leftarrow \boldsymbol{Z}_p$，然后计算$R_i=s_iG+c_iY_i$；

3)计算哈希函数值 $c=h(\mu, R_1, \cdots, R_n)$；

4)计算 c_π使得其满足等式 $c=c_1+c_2+\cdots+c_n \bmod p$；

5)计算$s_\pi=u-x_\pi c_\pi$；

公布 $\sigma=(c_1, \cdots, c_n, s_1, \cdots, s_n)$作为环 L 对于消息 μ 的签名。

$b \leftarrow \mathrm{Verify}(\mu, L, n, \sigma)$：该算法以消息 μ、包含 n 个公钥的集合 L 以及签名 σ 为输入，按照下述方法判断签名的合法性。

1)解析签名为 $\sigma=(c_1, \cdots c_n, s_1, \cdots, s_n)$；

2)计算 $c=c_1+c_2+\cdots+c_n \bmod p$ 和$c'=h(\mu, s_1G+c_1Y_1, \cdots, s_nG+c_nY_n)$；

3)如果 $c=c'$，输出 $b=1$ 表示签名合法；否则输出 $b=0$ 表示签名无效。

该环签名方案基于 Rivest[9]中基本的环签名设计思想，被证明拥有无条件匿名性和选择消息下的签名不可伪造性。

数字签名方案同样由一组概率多项式时间算法组成，DS=(Setup，KGen，Sign，Verify)。由于篇幅原因，不单独列出某一种可用于本文的数字签名方案。提示读者，如果将上文的环签名集合大小设定为 $n=1$，则其实际上非常类似于 Schnorr 数字签名方案，并可在随机预言机模型下证明其满足选择消息下的签名不可伪造性[10]。

3 系统架构

本文提出的基于轻量级区块链的低压用户需求响应方案，通过模块化的功能组件解决用户交互数据的隐私性、用户匿名性、用户端轻量化和区块链公平性等问题，为系统提供了维护与升级的灵活性。所提方案的系统架构如图 1 所示，其可被划分为电力公司掌控的核心数据层、片区数据控制层和电表用户层。

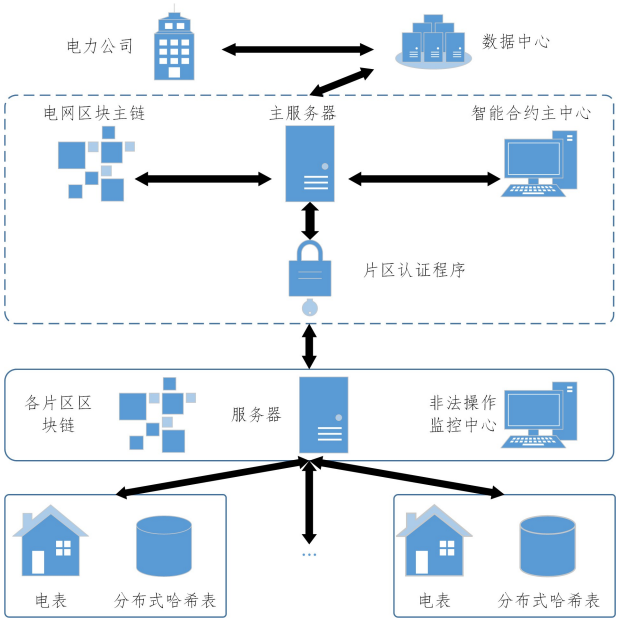

图 1　系统架构

Fig. 1　System architecture

1)核心数据层。系统架构的顶端为电力公司掌控的核心数据层，包含数据中心以及由数据中心管理的核心功能模块(包括电网区块主链、主服务器、智能合约中心、片区认证程序)。该层由电力公司建立维护，由电力公司控制该层亦能让其对系统具有一定的管理能力。

具体而言，电力公司架设数据中心统筹管理低压用户响应方案。数据中心通过核心功能模块与下层进行通信，将核心功能模块部署于数据中心，但对于体量庞大的低压用户需求响应业务(如横跨多省)，核心功能模块可以以省或市为单位部署于就近的电力子公司的服务器上，由数据中心远程管理(图 1 中的虚线框表示核心功能模块的这项特点)。

电网区块主链由电力公司生成与维护(在存在多个核心功能模块就近部署的情况下，可由各子公司公平生成与维护)，用于存放所有系统底层产生的片区区块链数据。区块主链的核心目的是维持一个完整统一的低压用户需求响应的数据副本。基于此，低层服务器与用户端无需时刻维持完整的区块链副本，仅在需要时向顶层检索数据(但需保存检索信息)。主链保存的数据副本令电力公司对数据有一定的追踪调查能力。

电力公司在需要发布某些业务活动时，可以通过智能合约主中心向下发布需求内容与奖励措施(智能合约)。这些智能合约将在智能合约主中心留下一份副本作为存证，然后通过片区区块链发布到各个片区。电网区块主链并不直接部署这些智能合约，仅当智能合约在片区被各用户履行后，主区块链从片区区块链上载数据进行保存。

电网区块主链负责保存完整数据，因此必须保持其数据的正确性。换言之，当区块主链上载片区区块链数据时，其需要对数据来源进行考证。如果不是旗下设置的片区服务器提供的数据，其一律不将这些数据上载至主区块链(即仅有合法的下层对上层有写入能力)。为此，核心功能模块在与片区服务器通信时，需要片区服务器对自身身份进行认证。该任务通过片区认证程序完成，其主要采用 4.4 节的片区认证算法。该算法为用户匿名性提供了支持，即片区服务器可以向主服务器证明自身的合法性，但不透露自身所属的片区。具体而言，片区服务器在建立之初将为片区认证程序注册一个公私钥对(Y_i, x_i)。在片区服务器与上层通信时，其从同层的片区中选取$u-1$个片区服务器公钥与自身公钥组成一个群组 L(随机打乱公钥的顺序)，将自身隐藏在群组中。鉴于片区服务器一般有较高的安全系数(如已被敌手控制但未被诚实方发现)，片区认证程序可以大范围防范不法数据侵害电力公司区块主链或主服务器的行为。

2)片区数据控制层。逻辑上被划分一个自治单位的社区共享一个片区的数据控制层，社区可包含一个或多个智能电表用户小区。片区用户通过片区数据控制层与顶层交互，并利用片区数据控制层提供的服务自治片区的区块链系统。片区数据控制层对片区用户具有一定的管理与约束能力。

片区区块链由所有电表用户公平竞争生成，但完整的区块链数据仅保存在片区服务器中，以减小用户的存储开销。区块链数据包含电力公司下发的需求(智能合约)和电表用户电力交易信息等(可根据新增业务灵活扩展)，仅有成功竞争区块生成权的用户对片区区块链拥有写入权力。本系统的片区区块链竞争机制仍采用工作量证明，但为了避免无意义的算力竞争，计算设备由电力公司统一提供(如电表本身)，以在用户诚实的情况下各自拥有均等的获胜概率。电力公司提供的计算设备在确保片区区块正常生成速度的情况下，需尽可能地降低计算能力(即降低能耗)[1)]。

由于电力公司提供的智能电表与某些设备需要安装在接近用户的地理位置上，这为恶意用户对电表和设备进行恶意操作提供了可能，因此需要对恶意用户的行为进行限制。为此，片区数据控制层引入非法操控监测中心，以限制恶意用户对电力公司提供设备的任意非法操作。限制通过非法操作监控中心部署的监控算法自动完成(见 4.3 节)，其根据需要定义了多个功能代码，用于接收设备反馈给监控中心的各类状态信息。本文设计的方案并未限定异常行为的种类与其对应的惩罚措施，实际使用时可由系统建立者与使用者协商新增内容。作为对详细定义惩罚内容的替代，文中的非法操作监控中心为每个片区用户维护一个公开的信誉度信息，任意恶意行为将降低信誉度。低下的信誉度将导致用户在使用低压用户需求响应系统时受到限制(即未定义的惩罚)。

由于系统业务越复杂，将导致信誉度信息访问越频繁，为了减小用户频繁访问非法操作监控中心造成的交互负荷，除了在中心留存完整的信誉度副本，本文方案还采用了分布式哈希表分布式存储信誉度信息，用户可以从逻辑临近的用户结点获取相应的信息(见 4.2 节)。

3)电表用户层。其由某一片区各家庭的电表用户设备构成，其与所属片区的数据控制层交互，以使用低压用户响应方案的功能。

电表用户层的实体合力自治片区区块链系统，仅有通过共识机制筛选出的实体拥有对当前区块链下一区块的写入权。共识机制采用弱化的 POW，以避免无意义的算力竞争。如前文所述，“弱化”主要体现在算力设备由电力公司均等分配，以及对非法操作进行监测两点。此种做法的主要目的是

1) 实际上，区块生成速度由片区的总算力与 POW 问题求解难度共同决定。由于其需要在实际应用中进行权衡，故本文未进行讨论。

在尽可能降低能耗的同时，维护片区区块链的公平性。

电表用户存在破坏或篡改电表及其他设备的可能，从而影响抵押用户需求响应方案的正常运行。以电力公司提供的用于 POW 的低算力设备为例，如果恶意用户接入高算力设备（如用户家中的个人计算机）以辅助争夺区块链写入权，则片区区块链的公平性被随之削弱。为此，系统可要求 POW 证明必须带有电表的数字签名，但签名私钥集成在电表中，且恶意接入的高算力设备无法在不破坏电表的情况下获取私钥，即无法完成签名。因此，对恶意算力提供的 POW 证明可以依据数字签名的合法性予以监测。

本文方案的电表用户层未考虑片区用户的匿名性，仅考虑了用户区块链数据的机密性，即用户在进行各种业务时，电表和设备的 ID 或公钥会直接被记录（尽管这可以理解为假名系统，但假名系统不能提供完善的匿名性），但用户在区块链上的业务数据不是以明文形式（如积分和电量的增减）出现的。不考虑匿名性是因为当前信誉度信息需要精确的用户身份标识。业务数据的机密性通过 4.1 节的秘密通信方案实现，其在隐藏业务数据内容的同时，使全片区用户能验证其一般代数运算的合法性。

4 具体构造

本章主要介绍基于轻量级区块链的带监控的低压用户响应方案各模块的实现细节，其中包括电量及积分数据的秘密通信、信誉度信息的分布式存储与检索、非法操作的监控，以及片区认证算法。

4.1 电量及积分数据的密码通信

为区块链系统用户提供机密性数据时，加密方案无法胜任所有的情形。例如，智能电网低压用户相应响应节电任务时产生的电量数据与完成任务后系统发放的积分奖励数据等。由于区块链上的这些数据前后变化的合法性需要被全网用户验证，采取加密方式对它们进行隐藏是不切实际的。为了在不透露这部分数据的情况下，维持全网用户的验证能力，本节采取了 RingCT 等秘密交易技术的思想[11]，给出了最朴素的一种解决方案。具体而言，数据将被抽象化为整数并隐藏在承诺中，然后秘密证明它们满足某种等式关系。

4.1.1 承诺的和证明

不失一般性地考虑如下场景：1）低压用户响应节电任务后，电网核心结点需为其分配对应的积分奖励；2）低压用户希望消耗已赚取的积分从某方兑换物品。在情况 1）中，用户获得的积分额应等于电网核心结点分配给该用户的积分额；在情况 2）中，用户账户减少的积分额应等于兑换物品需要消耗的积分额。这类相等关系需要得到区块链全网用户的验证，但又不应透露具体的积分数量。针对这类需求，我们可以将积分额放入承诺方案中进行隐藏和绑定（参见 2.2 节），然后调用对应的知识证明方案秘密地证明上述相等关系。

具体而言，将积分发出方各账户应减少的积分额转译成整数 a_i[1)]，然后对其进行承诺，即对于 $i\in[1,n]$，令 $C_i=a_iH+z_iG$ 是消息 a_i 的承诺，其中 a_i 随机值为 z_i。同样地，积分接收方各账户应增加的积分额被转译成整数 b_i，并对其进行承诺，即对于 $j\in[1,m]$，令 $D_j=b_jH+y_jG$ 是消息 b_j 的承诺，其中 y_j 为随机值。欲向全网用户（验证者）说明输出的积分与输入的积分相等，即要在全网用户仅观察到 C_i，D_j 时，由积分输出方（证明者）说明 $\sum_{i=1}^{n}a_i=\sum_{j=1}^{m}b_i$。提醒读者 C_i，D_j 均由证明者生成，因此其知晓所有承诺的打开方式。

用于 Pedersen 承诺值之间上述等式关系的证明系统包含 3 个概率多项式时间算法，即 POS=(Setup, Proof, Verify)。

$pp\leftarrow$Setup(1^λ)：输入安全参数 λ，该算法生成证明系统的系统参数。令 E 是一条定义在有限域 $\boldsymbol{F}_q$ 上的椭圆曲线。令 $\boldsymbol{G}\in E$ 为一个阶为素数 p 的点，其中 $p=O(2^\lambda)$。令 $\boldsymbol{G}$ 为由 $\boldsymbol{G}$ 生成的 $\boldsymbol{E}$ 的素数阶子群，并选取 $\boldsymbol{H}\leftarrow\boldsymbol{G}$。公开系统参数 $pp=(\boldsymbol{G},G,H,p,q)$，该参数将作为其他算法的隐含输入。

$\pi\leftarrow$ Proof $((C_i)_{i=1}^n,(a_i)_{i=1}^n,(z_i)_{i=1}^n,(D_i)_{i=1}^n,(b_i)_{i=1}^n,(y_i)_{i=1}^n)$：该算法以所有的承诺与打开方式为输入，按照如下方式计算并输出对整数和关系的证明 π。

1）计算 $pk=\sum_{i=1}^{n}C_i-\sum_{j=1}^{m}D_j$。注意如果 $\sum_{i=1}^{n}a_i=\sum_{j=1}^{m}b_i$，则有：

$$pk=\sum_{i=1}^{n}C_i-\sum_{j=1}^{m}D_j=(\sum_{i=1}^{n}a_i-\sum_{j=1}^{m}b_j)H+(\sum_{i=1}^{n}z_i-\sum_{j=1}^{m}y_j)G=(\sum_{i=1}^{n}z_i-\sum_{j=1}^{m}y_j)G。$$

2）计算 $sk=\sum_{i=1}^{n}z_i-\sum_{j=1}^{m}y_j$，然后以计算出的 pk 为公钥，以 sk 为私钥，以及以任意字符串 μ 为消息，调用数字签名方案的签名算法（参见 1.3 节）生成签名 $\sigma\leftarrow$DS.Sign(μ,pk,sk)。令 $\pi=(\sigma,\mu)$ 为证明记录并予以公布。

$c\leftarrow$Verify($(C_i)_{i=1}^n,(D_i)_{i=1}^n,\pi$)：输入所有的承诺和对应的证明 π，验证算法计算 $pk=\sum_{i=1}^{n}C_i+\sum_{j=1}^{m}D_j$，并调用数字签名方案的验证算法 $DS.Verify(\mu,pk,\sigma)$，如果签名验证算法判断签名合法，则本文算法输出 $c=1$ 以表示验证通过；若验证算法判定签名无效，则本文算法输出 $c=0$ 表示验证不通过。

4.1.2 承诺的范围证明

上一小节对等式关系的证明存在数值溢出的问题，即如果 $\sum_{i=1}^{n}a_i=\sum_{j=1}^{m}b_i+kp$，其依然可以产生合法的证明来通过验证算法的验证，其中 k 为任意整数。因此必须限定被承诺消息 a_i 所代表的整数或浮点数的范围（如对于某一正整数 n 和任意的 a_i，有 $na_i<p$）。范围证明即为证明被承诺整数 $a_i\in[0,2^l-1]$，以实现这一目标的证明系统。

范围证明系统由一组概率多项式算法 RP=(Setup, Proof, Verify) 构成，其详细描述如下。

$pp\leftarrow$Setup(1^λ)：输入安全参数，生成证明系统的系统参数。承诺的范围证明与前一节承诺的和证明可共用系统参数建立算法。公开系统参数为 $pp=(\boldsymbol{G},G,H,p,q)$，且系统参数将作为其他算法的隐含输入。

$\varphi\leftarrow$Proof(C,a,z)：证明算法以承诺 C 及其打开方式 a，z 为输入，按照如下步骤计算输出证明 ϕ。

1）算法首先将消息 a 转译成二进制形式 $(a_0,a_1,\cdots,a_l)$，即 $a=\sum_{i=0}^{l}a_i2^i$。

2）调用 Pedersen 承诺方案对 a_i2^i 生成承诺 $A_i=a_i2^iH+z_iG$，其中 $z_i\leftarrow\boldsymbol{Z}_p$。

3）对于 $i\in[1,l]$，令 $pk_{i1}=A_i$，$pk_{i2}=A_i-2^iH$，$sk_i=z_i$，

1) 若为浮点数，则可以将小数点右移系统规定的固定量完成整数转译。

然后以 pk_{i1} 和 pk_{i2} 为群组成员，以 sk_i 为私钥和以任意字符串 μ 为消息，调用环签名的签名算法生成签名 $\sigma_i \leftarrow$ S. RSign(μ, (pk_{i1}, pk_{i2}), sk_i)。

4）$\varphi=((\sigma_1,\sigma_2,\cdots,\sigma_l),(A_1,\cdots,A_l),\mu)$ 将作为证明记录公布。注意如果 $a_i\in\{0,1\}$，则 pk_{i1} 与 pk_{i2} 中的一个为 z_iG，因此 z_i 实则为 1.3 节所述环签名方案的私钥。

$b\leftarrow$Verify(C,φ)：验证算法以承诺 C 和证明 φ 为输入，按照如下步骤判断证明的合法性。

1）将证明解析为 $\phi=((\sigma_1,\sigma_2,\cdots,\sigma_l),(A_1,\cdots,A_l),\mu)$。

2）对于 $i\in[1,l]$，首先计算 $pk_{i1}=A_i$，$pk_{i2}=A_i-2^iH$，其次调用环签名验证算法 RS. Verify(μ, (pk_{i1}, pk_{i2}), σ_i) 验证签名 σ_i 的合法性。

3）如果签名验证算法判定签名为合法，则该算法输出 $b=1$，表示验证通过；否则该算法输出 $b=0$，表示验证不通过。

4.1.3　完整的部署

电量及积分数据的秘密通信方案由一组概率多项式时间算法 SC=(Setup,Com,Proof,Verify) 组成。假设发送者需要将 n 个账户中的电量或积分数据分别转出 $(a_1,a_2,\cdots,a_n)$ 单位，然后将转出总量分别转入 $(b_1,b_2,\cdots,b_m)$ 单位到 m 个接收账户中。为了使转移结果是合法且公平的，显然有 $\sum_{i=1}^{n}a_i=\sum_{j=1}^{m}b_j$。发送者使用 Com，Proof 算法，用于生成秘密通信数据提交给区块链，而全网用户包括矿工结点使用 Verify 算法进行合法性验证。具体描述如下。

$pp\leftarrow$Setup(1^λ)：可由低压用户响应系统调用，生成秘密通信方案的系统参数，其中包括承诺方案、数字签名方案、整数的范围证明系统、整数范围证明系统的系统参数。用户可以调用各个子方案的系统建立算法支持总系统的建立，但观察这些方案与系统的参数建立算法，不难看出其中有相当多的公共部分，因此可以去掉冗余的部分，针对性地建立系统。简而言之，调用 4.1.1 小节的 Setup(1^λ) 生成 $(\boldsymbol{G},G,H,p,q)$，并根据安全参数挑选一个合适的密码哈希函数 $h:\{0,1\}^*\rightarrow\boldsymbol{Z}_p$ 和一个适当的二进制整数长度上界 l。令 $pp=(\boldsymbol{G},G,H,p,q,h,l)$ 为公开的系统参数，且 pp 将作为其他算法的隐含输入。

$C\leftarrow$Com(a,z)：用于隐藏和绑定发送者需要传输的消息 a。对于消息 $a\in\boldsymbol{Z}_p$，该算法直接调用承诺方案的承诺算法生成承诺 $C\leftarrow$COM. Com(a,z)，其中 $z\leftarrow\boldsymbol{Z}_p$。对于所有需要传输的数据，发送者均会以其为输入调用该算法一次。

$\eta\leftarrow$ Proof $((C_i)_{i=1}^n,(a_i)_{i=1}^n,(z_i)_{i=1}^n,(D_i)_{i=1}^n,(b_i)_{i=1}^n,(y_i)_{i=1}^n)$：用于证明 $C_1,\cdots,C_n,D_1,\cdots,D_m$ 内隐藏着合法的整数（即 $\sum_{i=1}^{n}a_i=\sum_{j=1}^{m}b_i$），并且等式关系不存在越界问题（即 $a_1,\cdots,a_n,b_1,\cdots,b_m<2^l$）。为此，该算法进行以下操作。

1）对于 $i\in[1,n]$，调用范围证明算法生成证明 $\varphi_i\leftarrow$RP. Proof(C_i,a_i,z_i)。

2）对于 $j\in[1,m]$，调用范围证明算法生成证明 $\bar{\varphi}_j\leftarrow$RP. Proof($D_i,b_i,y_i$)。

3）调用整数和关系证明算法生成证明。

$\pi\leftarrow$ POS. Proof $((C_i)_{i=1}^n,(a_i)_{i=1}^n,(z_i)_{i=1}^n,(D_i)_{i=1}^n,(b_i)_{i=1}^n,(y_i)_{i=1}^n)$

4）令 $\eta=(\varphi_1,\cdots\varphi_n,\bar{\varphi}_1,\cdots\bar{\varphi}_m,\pi)$ 为所有秘密传输的消息的合法性证明。证明将与秘密消息一同被发送。

$c\leftarrow$Verify($(C_1,\cdots,C_n),(D_1,\cdots,D_n),\eta$)：验证者观察到秘密传送的消息 $(C_1,\cdots,C_n),(D_1,\cdots,D_n)$ 和对应的合法性证明 η 后，调用该算法予以合法性验证。为此该算法进行以下操作。

1）将 η 解析为：

$\eta=(\phi_1,\cdots\phi_n,\bar{\phi}_1,\cdots\bar{\phi}_m,\pi)$。

2）对于 $i\in[1,n]$，调用范围证明验证算法得到 $b_i\leftarrow$RP. Verify(C_i,ϕ_i)。如果 $b_i\neq1$，则算法停止并返回 0，表示秘密传送的数据不具有合法性。

3）否则对于 $j\in[1,m]$，调用范围证明验证算法 $c_j\leftarrow$RS. Verify($D_j,\bar{\phi}_j$)。如果 $c_j=0$，则算法停止并返回 $c=0$，表示秘密传送的数据不合法。

4）否则调用整数等式关系验证算法。

5）POS. Verify($(C_1,\cdots,C_n),(D_1,\cdots,D_n),\pi$)。

如果验证不通过，则返回 $c=0$，表示秘密传送的数据不合法；否则返回 $c\neq1$，表示秘密传送的数据合法。

4.2　信誉度信息的分布式存储与检索

4.2.1　基于布隆过滤器的信息检索方案

布隆过滤器是一种利用密码哈希函数快速检索某一元素是否位于指定集合的技术，用于快速检索用户的信誉度信息。本文采用了基于布隆过滤器的高效信息检索方案。

基于布隆过滤器的信息检索方案包含一组概率多项式时间算法 IR=(Setup,Save,Retrieve)，其具体描述如下。

$pp\leftarrow$Setup(n,m,q,F)：输入自然数 n,m,q 与带密钥的哈希函数族 $F=\{F_n\}_{n\in\boldsymbol{N}}$，其中 $F_n=\{f_k\mid f_k:\{0,1\}^*\rightarrow\{0,1\}^n\wedge k\in\boldsymbol{K}\}$，$\boldsymbol{K}$ 为哈希函数的密钥空间，该算法选取 $k_1,k_2,\cdots,k_m\leftarrow\boldsymbol{K}$ 定下 m 个哈希函数实体 $f_1,f_2,\cdots,f_m$，然后初始化一个长度为 q 的布尔型数组 I（值全为 0 的数组）。该算法输出系统参数 $pp=(n,m,l,F,k_1,k_2,\cdots,k_m,I)$。系统参数将作为其他算法的隐含输入，需要在各区块链结点中维持统一的版本[1)]。

$b\leftarrow$Save($file$)：对于记录信誉度的文件 $file$，该算法调用 m 个哈希函数实体生成其布隆过滤器索引值，即对于 $i\in[1,m]$，$d_i=f_i(file)\bmod q$，然后将数组 I 中下标为 d_i 的元素设置为 1（即 $I[d_i]=1$）。所有操作进行完毕后，算法输出 $b=1$，表示算法正常结束；否则输出 $b=0$。

$b\leftarrow$Retrieve($file$)：当输入想要检索的文件时，算法调用 m 个哈希函数实体生成其布隆过滤器索引值，即对于 $i\in[1,m]$，$d_i=f_i(file)\bmod q$，然后取出数组 I 对应位置的元素 $I[d_i]$。如果所有值的与运算 $\bigwedge_{i=1}^{m}I[d_i]=1$，则算法输出 $b=1$，表示被检索的文件存在于系统中；否则输出 $b=0$，表示检索的文件不存在。

4.2.2　基于分布式哈希表信息存储方案

信誉度信息经过 4.2.1 小节方案的检索后，若被判定为存在于低压用户需求响应系统中，则用户应可以快速追踪和访问该数据。为了减轻片区服务器被集中访问的负担，信誉度信息将通过分布式哈希表存储。本文的分布式哈希表采用

1) 系统建立算法也可选取 m 个不同的输出长度均为 n 的哈希函数作为系统参数。

了简化的 Kademlia 方案[12]。

对 Kademlia 方案进行简化主要出于对电网实际情况以及片区用户轻量化的考虑。原版方案的用户 ID 与文件摘要 ID 建议使用 160 甚至更长的二进制字符串，其中后者可由输出为 160 位的密码学哈希函数用于文件生成。然而，低压用户需求响应方案中某一片区的用户通常仅包含数个小区，用户 ID 数量较难超过 10 万（即 17 bits），且每一个用户对应唯一的信誉度文件，这与一些 P2P 文件分享系统的情况大为不同。此外，过长的 ID 将增加分布式哈希表中检索文件与确定存储结点时的计算开销，由此本节希望简化参数与步骤，降低分布式哈希表方案中 RPC（Remote-Procedure-Call）类程序的执行时间。

具体而言，用户 ID 与文件 ID 采取相同的二进制字符串，以 ID 表示（即文件与用户 1-1 对应）。ID 长度尽可能短，例如设置 $n=32$ 位，以减小结点和文件定位时的计算开销（即 $ID\in\{0,1\}^{32}$，为 8 位 16 进制数）。方案采用的距离测度为异或形成的整数值，该整数值越大则表示距离越大。例如令 $ID_1=\{a_{31},a_{30},\cdots,a_0\}$，$ID_2=\{b_{31},b_{30},\cdots,b_0\}$，则两个二进制字符串之间的距离为 $(a_i\oplus b_i)$。如果想直观地查看其代表的十进制整数，则仅需计算 $\sum_{i=0}^{31}(a_i\oplus b_i)\cdot 2^i$，其中，$\oplus$ 为异或操作，$\sum$ 为普通整数加法下的求和，乘法为普通整数乘法。

本文的分布式哈希方案包含（SSetup，NSetup，Ping，Store，FNode，FValue）6 个程序，其中后三者为 RPC 类程序。

$pp\leftarrow$SSetup(·)：系统建立程序为分布式哈希表方案生成系统参数 $n,k,a\in\boldsymbol{Z}^+$。令 $pp=(n,k,a)$，该系统参数将引导分布式系统中的各结点正确进行初始化。其中，n 为系统中用户与文件 ID 的长度，k 为记录某一距离的通信录中保留的最大记录数，a 与定位结点时的操作有关。

$b\leftarrow$NSetup(pp，ID)：各结点收到系统参数后，初始化本地数据库的数据结构。其生成 n 张可以存储 k 条记录的表（用于记录结点位置），分别以整数 $i\in[1,n]$ 进行编号。编号为 i 的表可称为 k-bucket i，表中的记录以键值对（ID，$addr$）的形式存储（即用户号与用户的网络地址）。系统参数中的 a 将被存储以作为其他程序的输入，$ID\in\{0,1\}^n$ 为系统分配给结点的唯一标识。如果结点初始化成功，则程序输出 $b=1$；否则输出 $b=0$。

$b\leftarrow$Ping($addr$)：向网络地址 addr 的结点发出请求以确定对方是否在线。若在线，则返回 $b=1$；否则返回 $b=0$。

$b\leftarrow$Store((ID，$value$)，$addr$)：存储程序被调用以向地址为 $addr$ 的结点要求存储关键值数据对（ID，$value$）。若存储成功，则返回 $b=1$；否则返回 $b=0$。

$(ID_j,addr_j)_{j=1}^{k}\leftarrow$FNode($ID$，$addr'$)：结点寻找程序被调用，以向地址为 $addr'$ 的结点请求提供其已知的与 ID 距离最近的结点标识与结点地址（ID_j，$addr_j$），要求尽可能返回 k 个标识地址对（如果被记录的最近结点数不够，则选择次近的，依此类推；仅当所有被记录结点少于 k 个时，返回所有被记录结点的标识与地址信息）。

想要在全网寻找与某一 ID 最近的结点，仅调用一次 FNode 通常是不能实现的。因此需要迭代进行。该过程被称为 Lookup 的本地程序管理。Lookup 调用 FNode 寻找目标结点的过程如算法 1 所示。

算法 1 Lookup

Input：当前结点 ID_c，目标结点 ID_t，参数 α

Output：目标结点地址 $addr_t$ 或非法操作监控中心的答复

1. 初始化待查队列 Q=null
2. 初始化已查列表 L=null
3. 计算距离 $l=ID_c\oplus ID_t$
4. 从自身 k-bucket l 中取出 α 个（ID，addr）加入 Q（如果当前 k-bucket 取出的元素不够 α 个则从下一个 k-bucket l-1 中选取，直至 k-bucket 1 中没有新元素可取）。
5. While(Q≠Ø)
6. 取出 Q 中第一个 ID 和 addr
7. If ID∉L Then
8. 将 ID 加入 L
9. 运行 $(ID_j,addr_j)_{j=1}^{k}\leftarrow$FNode(ID，addr)
10. If $ID_t\in(ID_j)_{j=1}^{k}$ Then
11. Return $addr_t$
12. End if
13. For j∈[1，k] do
14. If $ID_j\notin$L
15. 将 $(ID_j,addr_j)$ 移入 Q 的队尾
16. End If
17. End do
18. End If
19. End While
20. 通知非法操作监控中心以获取目标地址或报错
21. Return 非法操作监控中心答复

通过上述迭代过程，当前结点将逐渐接近与目标结点距离最近结点的地址，最终获得目标结点地址。如果对所有可能结点的 FNode 请求均未获得目标结点地址，则目标结点信息未被任何结点保存，或者分布式存储系统遇到了不可预想的错误。此时，Lookup 程序将直接联系非法操作监控中心获取目标结点地址，或者从中心收到错误报告。

Lookup 程序能够成功并以较快的期望速度返回目标结点地址，必须有完备的通信录与文件存储策略。由于本文方案考虑的文件记录是用户信誉度信息，因此非法操作监控中心通信维护了一份全局的通信录与文件副本。但为了避免各用户结点访问文件时频繁与中心交付导致中心的通信与计算负担增加，文件与通信目录还需分布式地存储于各个结点。如前文所述，由于片区用户的数量远低于常见 P2P 文件分享系统的用户和文件数量，因此本文方案并未采用 Kademlia 自适应式地通信录与文件存储策略，而是改由中心决策分配给结点存储。原则上，仅需将结点地址与结点文件交由距离最近的少量结点（包括本身）存储即可。

4.3 非法操作监控

第 3 章详细阐述了区块链 POW 证明计算设备将由电力公司统一提供，且将会极力压低其能源损耗量。在这一前提下，很可能导致恶意用户引入未受认可的额外算力抢夺片区区块链控制权的行为。由此，需由非法操作监控中心对恶意行为进行监测，并配合一定的惩罚与警告手段对该行为加以抑制。

非法操作监控算法将实时地接收智能电表、POW 算力设备或其他影响低压用户需求相应方案正常运行设备的反馈消息，以达到恶意监测的目的。该算法可以以智能合约的方式部署于片区区块链，由片区用户自治，也可放置于非法操作监控中心的服务器，为片区提供服务。本文方案推荐后者，以达

到简化系统设计的目的。

非法算力监测算法主要依靠算力设备内置的签名私钥进行监控。算力设备的工作量证明 μ 必须跟上算力设备内置私钥对其的数字签名。数字签名方案可以灵活选择,为了方便描述,直接采用 2.3 节介绍的环签名方案,并将成员个数设定为 $n=1$,环成员设定为仅包含算力设备的公钥 $L=\{pk\}$。如果用户发出的工作量证明与其签名不能通过签名验证算法的验证,则说明用户行为可疑,可能绕开了算力设备进行了区块链生成权的竞争,监控中心需要做出反应。

算法 2 checkComp

输入:工作量证明 μ,签名 σ,用户 ID

输出:b∈{0,1}

1. 查找用户 ID 对应的公钥 pk
2. 调用签名验证算法获得 b=Verify(μ,pk,1,σ)
3. Return b

非法操作监控中心并非单一地处理恶意算力问题,方案的实际部署者可根据需求,设计更多类似于算法 checkComp 的功能以监测不同的非法操作。基于此,电表或算力设备需要在每次向非法操作监控中心汇报时,表明汇报内容归属于哪一类,使监控中心能准确调用对应的功能予以处理。本文方案以 dtcCheck 算法统筹处理这一问题。以非法算力监测为例,dtcCheck 描述如算法 3 所示。

算法 3 dtcCheck

输入:功能代码 func,功能需要的数据集 S

输出:警告字符串 warnString

1. Switch(func)
2. Case:'1 000'
3. 解析 S 为 S=(μ,σ,ID)
4. If checkComp(μ,σ,ID)==0 then
5. warmString="算力竞争非法嫌疑"
6. 公布用户 ID 的恶意嫌疑
7. Else
8. warmString="正常"
9. End IF
10. Return warnString
11. End Switch
12. Return warnString

电表或算力设备向非法操作监控中心汇报后,统筹算法 dtcCheck 根据功能代码选择合适的处理算法进行处理。上文假设了非法算力的监测功能代码为 func='1 000'。若统筹算法进入非法算力监测算法,则根据其判断结果进行对应操作。本文仅简单地将嫌疑结论反馈给用户的智能电表予以警告,除此之外,电力公司还可以根据结论前往现场证实或证伪用户的非法操作,以便进行后续的惩罚或澄清步骤。其他需要引入的监测功能仅需通过指定唯一功能代码,并在 dtcCheck 算法中将其作为 Case 引入即可。

4.4 片区认证算法

将片区认证算法用于片区认证程序,该程序在片区区块链向电网主区块链上传数据时向主服务器说明数据来源的可信性,即主服务器仅接受电力公司旗下片区的区块链数据,以避免攻击者向电网区块主链上传恶意数据从而影响低压用户响应系统。

片区认证算法可采用常用的身份认证协议[10],但为了赋予片区进一步强化片区用户的匿名性与用户数据的隐私性,本文系统选用了在 2015 年 Groth 等提出的"多中取一"身份认证协议。对于没有匿名性和隐私性需求的片区,其仍然可以采用简单的身份认证协议,或是在减少协议数量的考虑下将"多中取一"协议的成员数选取为 2,并复制一份成员运行协议。

该身份认证协议包含 MP=(Setup,KGen,Prove,Verify) 3 个有效的算法,本文的描述采用了 Zhang 等的描述方法[13]。

$pp\leftarrow$Setup(1^λ):令 E 是一条定义在有限域 $\boldsymbol{F}_q$ 上的椭圆曲线。令 $G\in E$ 为一个阶为素数 p 的点,其中 $p=O(2^\lambda)$。令 $\boldsymbol{G}$ 为由 G 生成的 E 的素数阶子群,并随机均匀地选取 $H\leftarrow\boldsymbol{G}\backslash\{0\}$。公开系统参数 $pp=(\boldsymbol{G},G,H,p,q)$,该参数将作为其他算法的隐含输入。

$(pk,sk)\leftarrow$KGen(pp):用于为单个用户生成身份认证的公钥 pk 与私钥 sk。其选择私钥 $x\leftarrow\boldsymbol{Z}_p$,然后计算出公钥 $Y\leftarrow xG$。公私钥对$(pk,sk)=(Y,x)$返回给用户保存。

证明与验证算法 Prove=(P_1,P_2),$Verify=(V_1,V_2)$为交互式概率多项式时间算法,认证过程由两方交互 3 轮完成。Prove.P_1发送初始消息 cmt 给验证者;验证者 Verify.V_1收到 cmt 后提交挑战值 e 给证明者;证明者 Prove.P_2根据挑战值 cmt 做出答复,并将答复值 rsp 发送给验证者;验证者 Vefiry.V_2根据所有收到的消息对证明做出判断,如果认为证明有效,则输出 1,否则输出 0。各个子算法的细节如算法 4 所示。

算法 4 Prove.P_1

输入:公钥集合 L=(Y_0,Y_1,…,Y_{u-1}),真实证明者在 L 中的编号 m∈[0,u−1],$u=2^n$。

输出:初始消息 cmt

Require:要求 n∈Z^+。(如果 L 的公钥数不足2^n,可复制其中某些数据以达到条件)

1. 计算整数 m 的二进制表示(m_1,m_2,…,m_n)
2. For j∈[1,n]do
3. 随机均匀独立地选择,r_j,a_j,s_j,$t_j\leftarrow Z_p$
4. 计算
5. $C_{mj}=m_jH+r_jG$
6. $C_{ai}=a_jH+s_jG$
7. $C_{bj}=a_im_iH+t_iG$
8. End do
9. For k∈[1,n−1]do
10. 随机均匀地选取 $\rho_k\leftarrow Z_P$
11. $C_{dk}=(\sum_{i=1}^{u-1}P_{i,k}Y_i)+\rho_kG$
12. End do
13. Return cmt=$((C_{mj})_{j=1}^n,(C_{aj})_{j=1}^n,ID_t\in(C_{bj})_{j=1}^n,(C_{dj})_{j=1}^n)$

注:算法第 11 行的 $P_{i,k}$将在后文介绍。

算法 5 Verify.V_1

输入:公钥集合 L=(Y_0,Y_1,…,Y_{u-1}),初始消息 cmt

输出:挑战值 e

1. 随机均匀选择 $e\leftarrow z_p$
2. Return e

算法 6 Prove.P_2

输入:公钥集合 L=(Y_0,Y_1,…,Y_{u-1}),真实签名者在 L 中的编号 m∈[0,u−1],私钥 x_m,初始消息 cmt,挑战值 e

输出:答复值 rsp

Require:x_m为真实证明者的私钥,即 $Y_m=x_mG$

1. For j∈[1,n]do
2. 计算
3. $f_j = e\ m_j + a_j \bmod P$
4. $Z_{ai} = er_j + s_j \bmod P$
5. $Z_{bi} = (e - f_j)r_j + t_i \bmod P$
6. End do
7. 计算 $Z_d = e^n x_m - \sum_{k=0}^{n-1} e^k \rho_k$
8. Return
$rsp = ((f_j)_{j=1}^n, (Z_{ai})_{j=1}^n, (Z_{bi})_{j=1}^n, Z_d)$

算法 7 Verify. V_2

输入：公钥集合 $L = (Y_0, Y_1, \cdots, Y_{u-1})$，初始消息 cmt，挑战值 e，答复值 rsp

输出：b∈{0,1}

1. For j∈[1,n]do
2. IF $e\ C_{mi} + C_{ai} \neq f_j H + Z_{ai} G$ or $(e - f_j)C_{lj} + C_{bi} \neq Z_{bi} G$ then
3. Return 0
4. End If
5. End do
6. if $\sum_{i=0}^{u-1} (\prod_{i=j}^{n} f_{j,i_j}) Y_i + \sum_{k=0}^{n-1} (-e^k) C_{d_k} \neq z_d G$ then
7. Return 0
8. Else
9. Return 1

算法 *Prove*. P_1 中的 $p_{i,k}$ 为其他变量通过计算后产生的结果。令 δ_{ij} 为克诺内克符号（即 $\delta_{ij} = 1$，当且仅当 $i = j$），考虑算法 Prove. P_2 中的变量 f_j。定义 $f_{j,1} = f_j = m_j e + a_j = \delta_{1m_j} e + a_j$，$f_{j,0} = e - f_j = (1 - m_j)e + a_j = \delta_{0m_j} e - a_j$。因此对于所有的 $i \in [0, u-1]$，乘积 $\prod_{j=1}^{n} f_{j,i_j}$ 是一个关于 e 的多项式：

$$p_i(e) = \prod^{n} (\delta_{i_j m_j} e) + \sum^{n-1} p_{i,k^k} = \delta_{im} e^n + \sum^{n-1} p_{i,k^k}$$

其中，$p_{i,k}$ 为多项式 $p_i(e)$ 的 k 次项，且当 a_j，i 和 m 已知时可以预先计算（无需 Prove. V_2 阶段获取的挑战值 e）。

此认证协议通过 Fiat-Shamir 变换能转化为签名长度与群组成员数呈对数关系的环签名方案。

结束语 区块链技术因其具备去中心化和数据安全可追踪的特点，正在被逐步应用于融入了云计算的智能电网的建设中。本文基于轻量级区块链设计了一个低压用户需求相应方案。首先采取了 RingCT 等秘密交易技术的思想，针对区块链上最为重要的数据通信部分提出了隐私保护方案并给出了部署方法；其次采用布隆过滤器和分布式哈希表对数据进行存储和检索；然后设计了一个非法操作监控算法，对全网结点中可能存在的非法操作实施监控和警示；最后给出了片区区块链与区块主链通信时的身份认证方法。

通过上述手段，本文的低压用户响应方案初步具有了用户数据隐私性、低能源消耗性、区块链自治公平性和区块链数据安全性等特点。然而，将轻量级区块链更好地应用于融入了云计算的智能电网低压用户需求响应业务仍有大量的工作可以进行。今后的工作将从高匿名性的角度进一步提升方案的性能，为基于轻量级区块链的低压用户需求方案投入实际应用做铺垫。

参 考 文 献

[1] SOUSA E L, MARQUES L A A, LIMA I S F, et al. Development a Low-Cost Wireless Smart Meter with Power Quality Measurement for Smart Grid Applications[J]. Sensors, 2023, 23(16): 7210.

[2] SOUSA E L, MARQUES L A A, LIMA I S F, et al. Development a Low-Cost Wireless Smart Meter with Power Quality Measurement for Smart Grid Applications[J]. Sensors, 2023, 23(16): 7210.

[3] ZHAI F, YANG T, ZHAO B, et al. Privacy-preserving outsourcing algorithms for multidimensional data encryption in smart grids[J]. Sensors, 2022, 22(12): 4365.

[4] KHAN A, UMAR A I, SHIRAZI S H, et al. QoS-Aware Cost Minimization Strategy for AMI Applications in Smart Grid Using Cloud Computing[J]. Sensors, 2022, 22(13): 4969.

[5] TIAN X X, CHEN X, TIAN F L. Community distributed power security transaction scheme based on blockchain [J]. Netinfo Security, 2019, 1: 51-58.

[6] QIN J L, SUN W Q, ZHU Y C, et al. Energy transaction method of microgrid based on blockchain [J]. Electric Power Automation Equipment, 2020, 40(11): 130-137.

[7] LI W X, MA B, LI H F, et al. Smart grid monitoring based on blockchain[J]. Information Technology, 2020, 44(1): 144-149.

[8] LIU Z R, WANG D, WANG B. Privacy preserving technology in blockchain [J]. Computer Engineering and Design, 2019(6): 1567-1573.

[9] JAKOBSSON M, JUELS A. Proofs of work and bread pudding protocols[C]// Secure Information Networks: Communications and Multimedia Security IFIP TC6/TC11 Joint Working Conference on Communications and Multimedia Security (CMS'99). Springer US, 1999: 258-272.

[10] GUO Y, YU L, ZHANG H W, et al. Trading platform of power data asset based on consortium blockchain [J]. Frontiers of Data and Computing, 2021, 3(1): 48-59.

[11] ZHAI F, YANG T, ZHAO B, et al. Privacy-preserving outsourcing algorithms for multidimensional data encryption in smart grids[J]. Sensors, 2022, 22(12): 4365.

[12] KIM J W, KIM J, LEE J. An Adaptive Network Design for Advanced Metering Infrastructure in a Smart Grid[J]. Sensors, 2022, 22(22): 8625.

[13] QIN Z, ZHANG X, FENG K, et al. An efficient identity-based key management scheme for wireless sensor networks using the bloom filter[J]. Sensors, 2014, 14(10): 17937-17951.

CHANG Ningyuan, born in 2000, postgraduate. Her main research interest is cryptography.

ZHANG Huang, born in 1988, Ph.D, lecturer. His main research interests include lattice-based cryptography, and zero-knowledge.