Computer Science and Application 计算机科学与应用, 2022, 12(4), 923-933
Published Online April 2022 in Hans. http://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2022.124094

Hans汉斯

基于智能合约的可信物联网SLA协议模型

谈玉胜，殷丹丽

广东工业大学，广东 广州

收稿日期：2022年3月8日；录用日期：2022年4月8日；发布日期：2022年4月15日

摘　要

在物联网服务质量评价体系中，用户难以找到物联网供应商在服务期间存在缺陷的证据。现行的云服务模型缺少一种可自动执行的机制来维护物联网用户与供应商双方的合法权益。现利用区块链智能合约技术，扩展现有的物联网服务评价模型，为检测和报告服务质量是否达到标准，引入了证人池机制，提出了一种无偏向随机选择算法保证了证人池中选择证人的随机性。同时，提出了证人审查算法用于检测潜在的懒惰证人与牺牲证人，通过引入博弈论证明证人机制的可靠性。最后，利用以太坊智能合约创建系统原型，分析智能合约接口工作量，最终实验结果表明该模型可行，性能符合设计要求。

关键词

区块链，物联网，服务质量，以太坊，智能合约

A Trusted IoT SLA Protocol Model Based on Smart Contracts

Yusheng Tan, Danli Yin

Guangdong University of Technology, Guangzhou Guangdong

Received: Mar. 8th, 2022; accepted: Apr. 8th, 2022; published: Apr. 15th, 2022

Abstract

In the IoT service quality evaluation system, it is difficult for users to find evidence of defects of IoT suppliers during service. The current cloud service model lacks an automatic mechanism to safeguard the legitimate rights and interests of both Internet of Things users and suppliers. In order to detect and report whether the service quality meets the standard, a witness pool mechanism is introduced and an unbiased random algorithm is proposed to ensure the randomness of the selection of witnesses in the witness pool. At the same time, a witness screening algorithm is proposed to detect potential lazy witnesses and sacrifice witnesses, and the reliability of witness me-

文章引用: 谈玉胜, 殷丹丽. 基于智能合约的可信物联网 SLA 协议模型[J]. 计算机科学与应用, 2022, 12(4): 923-933.
DOI: 10.12677/csa.2022.124094

chanism is proved by introducing Nash equilibrium point of game theory. Finally, the system prototype is created by using ethereum smart contract, and the workload of interface of smart contract is analyzed. The final experimental results show that the model is feasible and the performance meets the design requirements.

Keywords

Block Chain, The Internet of Things, Service Quality, Ethereum, Intelligent Contract

1. 引言

当前大多数物联网开发者用户通过使用物联网服务商提供的计算、存储和应用程序接口(API, Application Programming Interface)服务，共享服务器节点提供的计算资源。这种共享模式给用户带来极大的便利，使得小规模或初级用户也能以较低的成本获取便捷完善的物联网云服务体验。同时也带来了一些挑战，如物联网服务供应商在给用户提供服务期间出现宕机，用户若想从供应商索取相应的赔偿则存在一定的困难。业内普遍的处理方案是使用服务水平协议(SLA, Service Level Agreement)，这是一个商业概念，定义了用户与供应商之间的合同和财务协议。如果在服务期间内物联网云服务方出现了故障，用户可以向云服务方索取赔偿或抵消部分费用。如果在服务期间内，服务方没有出现故障，则用户正常支付全部费用。以此保障交易双方的合法权益。

现阶段，区块链技术与物联网技术进行结合，是一种发展趋势[1]。区块链技术的不可篡改特性，为物联网设备接入后的数据共享和协同工作提供了良好的保障。以太坊[2]中的智能合约提供了一种可行的方式来自动化执行服务水平协议。使用预言机[3] (Oracle)作为可信第三方，为智能合约提供可信的链外信息和数据，以验证触发智能合约的条件。

物联网 SLA 协议模型需要解决的问题有：如何保证物联网服务商与用户的合法权益；由哪一方检测和证明是否违反了 SLA 协议；如何实现支付流程的自动化执行，包括补偿流程的自动化。

本文在已有 SLA 协议的基础上，针对物联网云服务场景下用户权益的缺陷问题展开研究。主要贡献包括以下三个方面：

1) 引入了证人审查机制(WAM, Witness Audit Mechanism)，引入了证人池机制与证人委员会的监督工作，保障 SLA 协议的可靠性。

2) 结合区块链技术，制定了无偏向随机选择算法(URS, Unbiased Random Selection)和证人审查算法(TWR, The Witness Review)通过博弈论证明了证人为追求利益最大化必须诚实的原理。

3) 利用了以太坊的智能合约实现了 SLA 协议模型的原型系统，分析各个接口工作时资源的消耗量，满足设计需求。

2. 相关理论研究

2.1. SLA 协议

SLA 是云服务服务提供者与用户之间的服务质量协议[4]，可以适用于物联网云服务背景下，在协议

被违反时保护用户的合法权益。SLA 是连接用户与物联网服务提供者的桥梁，包含参数表示、参数映射、违约惩罚三个方面。

一个典型的 SLA 生命周期由多个实施阶段组成，包含协商、建立、监控、违规报告和终止。研究工作主要集中在 SLA 参数的语法定义，便于系统的在线处理；优化资源分配算法，减少 SLA 冲突。

SLA 模型功能框架如图 1 所示。

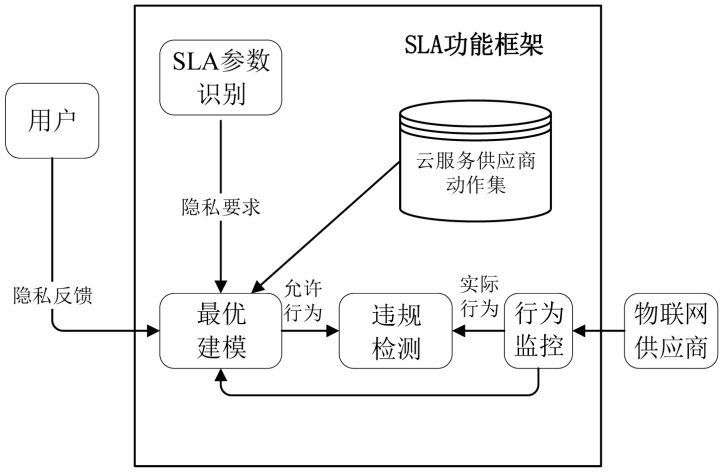

Figure 1. Service level agreement of functional framework
图 1. SLA 功能框架

根据云服务 SLA 的系统调查报告[5]，表明大部分的工作集中在监控和部署阶段，小部分工作集中在协商和建立阶段。目标是的供应商和用户的利益最大化。

2.2. 以太坊与智能合约

以太坊(Ethereum)是由比特币开发商 Vitalik Buterin 在 2013 年设计的[6]，用于促进区块链上分布式应用程序的开发。以太坊是一个开放的区块链平台可以建立和使用去中心化应用。以太坊是从 2015 年开始的第二代区块链，提出了以太坊虚拟机(Ethereum Virtual Machine)，这是一组表示虚拟机状态的字节值。在以太坊上运行的程序被称为智能合约(Smart Contract)。智能合约的工作流程如图 2 所示。

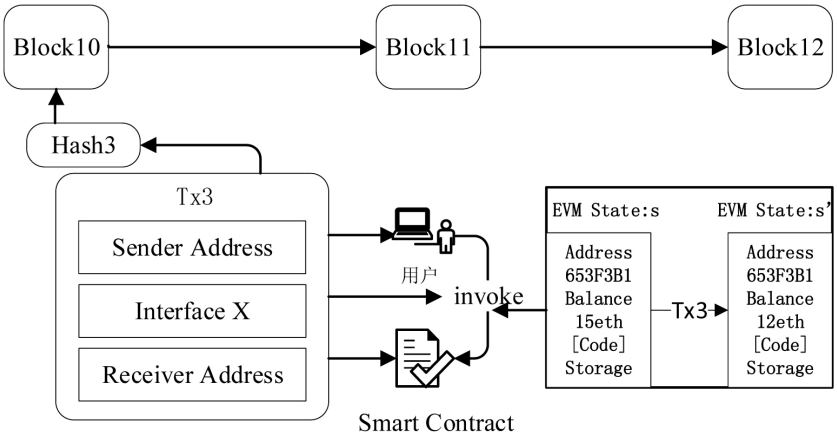

Figure 2. Smart contract workflow
图 2. 智能合约工作流程

在区块链中，利用加密算法实现价值转移，通过时间戳机制和散列链机制保证数据可追溯、不可篡改。依据共识算法实现区块数据的一致性，使得传统的基于信任的中心化机制所存在的安全性得到有效的解决。

以太坊是建立在区块链技术基础上的一个去中心化高鲁棒性的应用平台。网络架构完善，允许用户自定义操作。

2.3. 相关研究

国内外学者对 SLA 协议已有较多研究成果。这些方案都有其优势与不足。适用于不同的应用环境与网络模型。但对于物联网 SLA 协议的可信模型的研究还不够完善，还需要做进一步的深入研究。

文献[7]分别从硬件和传输协议角度保证了可信赖的执行环境。提出了一个认证的数据反馈系统：Town Crier，使用可信硬件后端，依靠智能合约提供源认证数据。该方案存在的问题是需要特殊的基础设施支持，实际应用环境中需要的是通用性与普适性的解决方案。文献[8]引入了一个服务器性能监视器的角色来检测违规并通知用户。提出了基于构建在区块链上的公共 SLA 合约的方法，实现了 SLA 协议平台原型，并将其应用到实例中。但是提议的解决方案缺乏对已识别违规行为的可信性分析。

基于区块链的共识协议系统，当事件发生在区块链之外时，如何可信地将随机事件上链。文献[3]的解决方案是使用预言机充当区块链的数据载体。比如 Oraclize 作为可信赖的第三方机构提供预言机服务，但是它可能会存在单点故障，这就偏离了区块链的去中心化思想。为了解决这个问题，文献[9]提出了在分布式预言机上工作，其工作方式是：只有在预言机之间达成协议时，事件的结果数据才会上链并触发事务，这个理念也存在一定的不足，对于预言机而言，如果没有足够的激励机制，预言机自身的可信问题并不能得到解决。

3. 基于智能合约的证人模型

本节将介绍基于智能合约的 SLA 协议模型，具体介绍证人的角色和功能以及如何检测和报告违规行为。介绍在区块链上如何使用智能合约来实施 SLA 协议的系统架构。

3.1. 引入证人池机制的 SLA 协议

证人是区块链的正常参与者，它们是自愿参与到系统中并且通过监控服务获取自身收入。证人角色默认是自私的，以获得自身收益最大化为目标[10]。

对于传统的物联网 SLA 协议，主要参与的两个角色是物联网云服务提供商 Provider 和物联网客户 Customer。使用一个基础示例来演示 SLA 协议的工作流程。假设一个物联网云服务提供商根据客户的请求，提供一个具备公网 IP 地址并定义了数据节点类型的服务器 Virtual Machine。在有效服务时间 T_service 内，客户 Customer 可以通过无线网卡设备透传连接到服务器[11]。服务质量协议 SLA 可以做出承诺：在有效服务时间 T_service 内，Provider 提供的服务器始终是可访问且稳定的，如果满足条件，在服务结束后 Customer 必须支付费用 F_service，否则在违反 SLA 的情况下，Customer 必须获得一笔补偿费用 F_compensation。这里我们假设 F_service > F_compensation，此外，对于 Customer 自身的网络问题引起的无法访问服务器不属于违反 SLA 协议。

为了确认是否真的违反了 SLA 协议，需要在传统 SLA 协议角色的基础上，再引入一个新的证人角色 Witness，利用以太坊交易平台创建一个可信的交易环境。这些 Witness 也是区块链的一般参与者，自愿参数 SLA 协议，通过提供监控服务获得收入。为了解决 SLA 协议中的信任问题，创建一个由 n 个 Witness 组建的证人委员会 $CW=\{w_1,w_2,\cdots,w_n\}$ 来参与 SLA 协议，它们一起监控 SLA 协议中的违规事件。并从

Provider 和 Customer 双方获得证人费作为回报。这里对证人角色做一个假设，证人都是自私的，都以自身收益最大化为目标。对于不符合假设的证人则视为懒惰证人，会通过下文算法将其排除在证人委员会之外。

3.2. 系统架构

系统由两类智能合约构成，第一类是证人池智能合约，是系统的基本智能合约；第二类是特定的物联网服务质量协议智能合约，用于 SLA 的自动执行。证人池智能合约主要有三个功能，分别是证人管理、具体 SLA 的生成和证人委员会的选举。区块链的任何用户都可以通过钱包地址在证人池中注册，成为证人。用户可以保持在线状态，等待被选中执行 SLA 协议。所有的证人参与这个系统的动机都是获得收益，系统中的证人越多，系统越可信[12]。系统整个架构如图 3 所示。

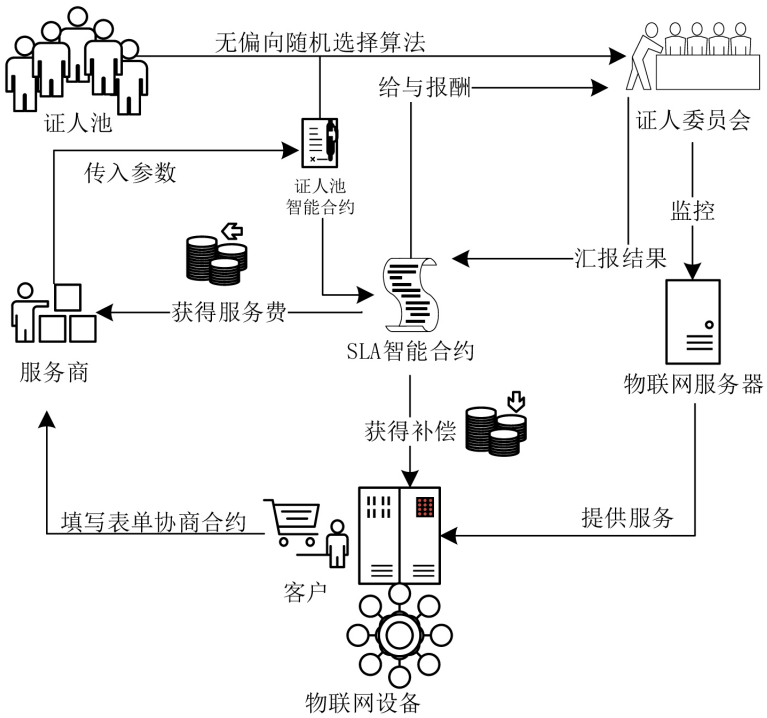

Figure 3. Diagram of the system structure
图 3. 系统架构图

物联网服务供应商 Provider 利用证人池智能合约为自己生成 SLA 智能合约，Provider 与 Customer 协商详细的 SLA 条款，服务时间 $T_{service}$、服务费用 $F_{service}$、补偿费用 $F_{compensation}$ 以及确定执行本 SLA 协议的证人 Witness 的数量。证人越多违规检测的结果越可信，同时 Provider 与 Customer 也要共同额外支出更多的费用。系统可以根据 Customer 表单提交的结果，Provider 传入这些参数在证人池智能合约中生成新的 SLA 智能合约。同时利用公正和随机的无偏向排序算法，从证人池中选举证人委员会。证人委员会中的成员是独立公正的无偏袒的，监控 SLA 协议的执行，于此同时，Provider 提供物联网服务器给 Customer 使用，并且在 SLA 中发布服务细节。接着，证人委员会开始监测服务，并将服务器公网 IP 通知给证人委员会成员。服务期间 Customer 可以正常使用服务器，证人委员会中的证人也可以不断 ping 服务器的 ip 地址，如果在 $T_{service}$ 内服务器出现故障，证人立即向智能合约报告违规事件[13]。

具体的 SLA 工作机制为：当系统收到证人委员会中的第一个举报时，开始计时，此后，SLA 智能合

约接收来自委员会中其它证人的举报。直到没有其它证人举报了为止[14]。如果 SLA 收到证人委员会中举报的人数 X，证人委员会的人数为 Y，当前仅当 $X/Y > 1/2$ 时，$T_{service}$ 期间内的违规行为将会被自动确认。例如，证人委员会中有三个成员参与监控 SLA 协议，只有其中至少两个证人报告该事件时，才能确认服务违规。同时，该 SLA 智能合约也定义了如果同一证人在 $T_{service}$ 时间内只有一次报告机会，预防恶意证人多次汇报扰乱机制。每一个证人都想要最大化自己的收益，他们都是诚实的参与者根据实际情况报告违规情况。最后 SLA 智能合约存在两种执行结果：一种是 $T_{service}$ 时间后，没有违反协议，Customer 正常支付的费用为 $F_{service}$；另一种是 $T_{service}$ 时间后，报告违反了协议，Customer 需要获得补偿，所需支付的费用为 $F_{service}$-$F_{compensation}$；不论哪种情况，证人委员会都会从智能合约中获取报酬。

4. 核心算法与理论支持

物联网服务质量可信认证模型所采用的关键技术是对 SLA 违规行为进行自动检测，得到使 Customer 和 Provider 的都信服的检测结果。使用无偏向排序算法，确保被选入证人委员会的证人都是随机且独立的。同时也使用了证人审计算法，分析了来自恶意证人的欺诈行为，确保参与者必须诚实才能利益最大化。

4.1. 无偏向随机选择算法

从证人池中选举证人委员会的方法使用的是无偏向随机选择算法，针对特定的 SLA 合同证人的选择必须是公正的，引入以太坊作为模型的可信方，在证人池智能合约中实现该算法，对于 Customer 或者 Provider 都不能有偏向。算法伪代码如算法 1 所示。

算法 1 无偏向随机选择算法

输入：

已注册的证人集合 set_registered;
在线证人数量 online_witness;
证人委员会中需要的人数 N;
第 b 个区块链的哈希值 B_b^{hash}
生成的新区块链长度 K_s
证人的在线状态 state
证人的信誉度：reputation
证人的地址 witness_address;

输出：

seed = 0; //初始化随机数种子
for(int i = 0; i < K_s; i++):
seed += B_{b+1+i}^{hash} ;//将 K_s 个区块哈希值累加
end for//结束循环获取随机种子
for(int j = 0; j < N; j++)://选择证人
index = seed % len(set_registered)+1;
//判断证人在线状态与信誉度
check(set_registered[index].state)&&check(set_registered[index].reputation)
set_registered.add(witness_address)//将证人添加到证人委员会集合中
end for//证人满足数量结束循环

return set_registered;//返回证人委员会集合

在证人池智能合约中，开放了提供注册接口，区块链用户都可以注册到证人池中，同时区块链用户可以将自身状态切换为“在线”或“离线”，表明当前是否愿意加入证人池选举活动。证人池中存储的地址列表记录了区块链用户注册的顺序。为了从证人池中选择 Y 个证人委员会成员，证人池智能合约定义了两个接口：请求和排序。利用区块的哈希值生成随机数种子，新的种子是根据前一个种子的哈希值生成的，这个过程重复进行，直到选定出 Y 个证人委员会成员[15]。这样既保证了算法的输出，又保证了算法的随机性，无论是对于 Customer 还是 Provider 都不能随意操纵排序结果，从而控制证人委员会的选举结果。

4.2. 证人审查选择算法

无偏向排序算法保证了选举的证人的公平独立，但是仍然需要一种审计机制来保证系统能够检测到恶意或者懒惰证人，并将其从证人池中开除[16]。由于智能合约中的交易是公开并永久存储在链上的，因此可以通过目击者的行为历史来审查证人的声誉，而不是根据其它人的反馈来评估证人的信誉度。

算法 2 证人审查算法

输入：

恶意证人集合 set_malicious;
默认声誉值 reputation;
证人委员会 set_registered;

输出：

```
flag = event.isValid;//获取 SLA 结果
setYes = set_registered.getYes();//获取证人委员会中所有投赞成票的集合
setNo = set_registered.getNo();//获取证人委员会中所有投反对票的集合
for(String address : set_registered)://对证人委员会的每一个地址进行遍历
    if(flag==false)://  如果最终结果为违规
        for(String address:setYes)://所有赞成票的地址扣除声誉
address.reputation--;
        else://否则所有反对票的地址扣除声誉
            for(String address:setNo):
address.reputation--;
if(address.reputaion < 0)://如果证人声誉小于 0
    set_malicious.add(address);//将证人地址加入恶意地址集
```

一般情况下，恶意证人主要分为两类：懒惰证人[17]和牺牲证人。懒惰证人是指不愿意举报违规行为的证人，或者举报违规行为的收益不足以激励证人工作，他们选择不真正监视的策略总是保持沉默，从不报告违规行为，使用这种策略后。即使 SLA 最终状态被违反，懒惰证人也不会付出代价。同时，整个系统中由于总体报告违规的次数应小于正常的次数导致懒惰证人仍然能够通过参与多轮服务总体获得收益。牺牲证人是指总是在特定时间点报告的证人，例如证人 W_s 总是在 SLA 启动 1 分钟内报告违规，虽然一开始 W_s 可能由于自己的恶意行为受到惩罚，但是 W_s 可以在以后的历史中向他人展示自己的行为模式，从而获得最大利益。为了审查这些恶意证人，系统需要设置声誉值参数，当违规行为被证实时，没有做出举报行为的证人声誉扣除 1 点，举报违规行为，最后系统并没有违规的证人同样扣除 1 点声誉值。当证人的声誉值降至为 0 时，证人的选举流程也被终止，提出证人池，系统恶意地址表中记录恶意证人

的地址，恶意证人无法再次加入证人池。这种算法的设计能有效减少懒惰证人与牺牲证人的参与，提高了 SLA 系统的稳定性与安全性。

4.3. 博弈论

博弈论是利用数学方法预测战略情景中的行为，每一个参与者的收益取决于自身和他人的决策。在经济学、生物学、计算机科学、政治科学中有着广泛的应用[18]。物联网服务质量协议的证人博弈机制定义如下。

证人机制定义：

通过三元组(CW，Δ，Γ)定义的 n 个角色的博弈，其中：

$$\mathrm{CW}=\{w_1,w_2,\cdots,w_n\}$$

表示 n 个成员，组成了证人委员会。

$\Delta=\Delta_1*\Delta_2*\Delta_3*\cdots*\Delta_n$ 表示一组策略集，其中 Δ_k 表示证人 w_k 所采取的策略集。这里，w_k 可以选择 w_k 中的任意一个策略 $\delta^*\in\Delta_k$，这里 $\delta^*=\left(\delta_1^*,\delta_2^*,\delta_3^*,\cdots,\delta_n^*\right)$。

$\Gamma=\{\gamma_1,\gamma_2,\gamma_3,\cdots,\gamma_n\}$ 表示一组支付函数，γ_k 表示在证人 w_k 在特定策略下获得收益的支付函数。

此外，$\delta_{-k}=\{\delta_1,\delta_2,\cdots,\delta_{k-1},\delta_{k+1},\cdots,\delta_n\}$ 表示没有证人 k 参与的策略配置。完整的策略可以表示成 $\delta=\{\delta_{-k},\delta_k\}$。实际上在证人博弈中只有两个动作，$\Delta_k=\left\{\delta_k^r,\delta_k^s\right\}$，其中 δ_k^r 将违反 SLA 协议的情况报告到智能合约中，δ_k^s 表示不向系统报告违规情况。在 Y 个证人的博弈中，我们定义一个报告 W_{report} 用于存储主动汇报的证人，定义一个报告 W_{slience} 用于存储沉默证人的信息。

违规机制定义：

$\mathrm{SLA}_{\text{status}}$ 有两种状态 $\mathrm{SLA}_{\text{status}}=\text{completed}$ 表示物联网服务质量协议完成了，$\mathrm{SLA}_{\text{status}}=\text{violated}$ 表示物联网服务质量协议被违反了。

根据 Y 个证人的配置文件，如果 $W_{\text{report}}\geq X$，其中 $1<Y/2<X<Y,X\in\mathrm{N}$ 表示违反了 SLA 协议，否则表示没有违反。

支付函数定义：

支付函数根据 SLA 状态设置对应参数的值，当 $\mathrm{SLA}_{\text{status}}=\text{violated}$，有 $\forall w_k\in W_{\text{report}},\gamma_k\left(\delta_k^r,\delta_{-k}\right)=5$；$\forall w_k\in W_{\text{slience}},\gamma_k\left(\delta_k^s,\delta_{-k}\right)=0$。

当 $\mathrm{SLA}_{\text{status}}=\text{completed}$，有 $\forall w_k\in W_{\text{report}},\gamma_k\left(\delta_k^r,\delta_{-k}\right)=-1$；$\forall w_k\in W_{\text{slience}},\gamma_k\left(\delta_k^r,\delta_{-k}\right)=1$。

支付函数设计为如果系统判定违规了，积极汇报的证人奖励 5 点收益，沉默的证人无收益；如果系统判定没有违规，汇报的证人扣除一点收益，沉默的证人奖励 1 点收益。总体收益分析如表 1 所示。

Table 1. Payoff function three-witness game profit analysis
表 1. 支付函数三证人博弈收益分析

w_1	w_3			
	w_2		w_2	
	δ_3^r		δ_3^s	
	δ_2^r	δ_2^s	δ_2^r	δ_2^s
δ_1^r	(5, 5, 5)	(5, 0, 5)	(5, 5, 0)	(−1, 1, 1)
δ_1^s	(0, 5, 5)	(1, 1, −1)	(1, −1, 1)	(1, 1, 1)

在 Y 个证人的博弈中，如果其中一个证人知道其它证人的行为会影响到自己的最终收益，当前证人会选择一个最佳策略来最大化自己的收益，证人的选择被称为纳什均衡点[19]。这里假设所有的证人都是客观理性，以赚取收益为目的。所以证人必须诚实，以实现收益最大化。

5. 实验与分析

为了测试当前物联网服务质量模型的可用性，在以太坊区块链测试网[19]“Rinkeby”上部署实现 SLA 智能合约。以太币是以太坊的加密货币，将用户的 MetaMask 账户地址发布到社交媒体上，并将该链接提交至 Rinkeby 便可以领取以太币，用于调试智能合约。在 Rinkeby 上生成若干账户来模拟不同的 SLA 模型中的角色，Provider、Customer、Witness。在每个模拟账户上使用以太币来执行接口。根据模型预付不同类型的费用。在实验前，首先部署证人池智能合约，让所有证人账户注册到证人池。Provider 生成物联网 SLA 智能合约以启动 SLA 生命周期。测试多种场景，以验证不同接口的功能。结果表明，当前系统模型满足设计要求，无偏向排序算法和证人审查算法保证了系统的可信度。

实验主要分析该模型的性能信息，测试智能合约中接口的复杂性。由于需要执行接口中定义的程序，需要消耗电能。接口定义的工作逻辑越复杂，调用时所需要的事务费用就越多。在以太坊中，矿工在工作时，衡量工作量的单位是 gas。因此我们可以通过实验记录每一个接口中的 gas 消耗量[20]来判断各个接口法复杂度。

Provider 与 Customer 相比，在整个生命周期中，Customer 和 Witness 接口的消耗量更少。这符合我们模型的设计需求和现实需要，因为在大多数情况下，Provider 在提供服务时赚取最多的收益，他们具有在 SLA 生命周期中提供更多资源的动机。图 4 显示了实验的研究结果。

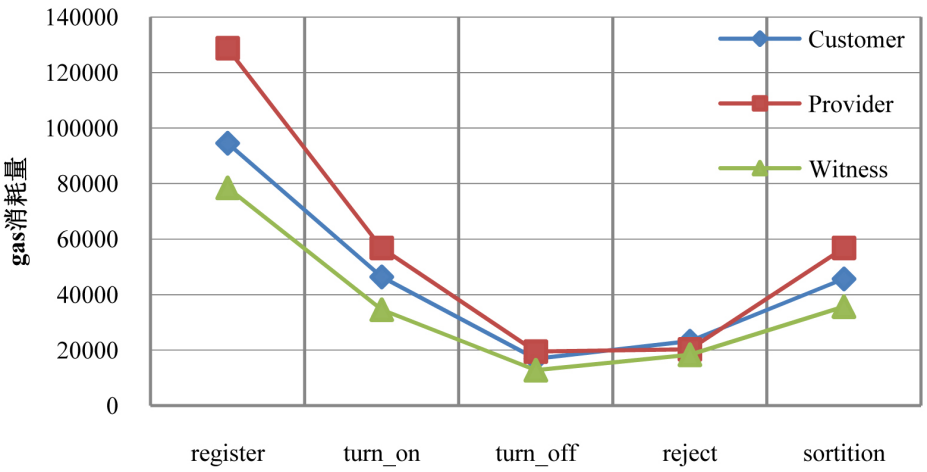

Figure 4. Interface workload of each role in SLA
图 4. SLA 各角色接口工作量

Customer 和 Witness 的消耗量是在基础的实验结果上得出的，还有进一步的优化空间，总体实验结果满足设计需求。

6. 结束语

本文提出了一种物联网 SLA 协议模型，确定了证人管理和无偏随机选择的机制；利用算法分析得出证人必须提供诚实的监督服务，以实现自身收益的最大化，同时提出了证人审查算法。最后利用以太坊的智能合约实现了一个原型系统。

实验研究证明了模型的可行性，将信任问题转化为经济问题，证人机制的合理性使得证人由于经济原则不得不诚实。对于物联网用户与服务供应商之间交易的公平与安全有着显著的意义。对于未来的工作，文中提出的可信 SLA 协议模型可以结合车联网、移动计算等领域提供可信的服务质量协议框架，具有广阔的应用前景。

基金项目

广州市科技计划项目(No.202007040005)。

参考文献

[1] Ali, M.S., Vecchio, M., Pincheira, M., *et al*. (2018) Applications of Blockchains in the Internet of Things: A Comprehensive Survey. *IEEE Communications Surveys & Tutorials*, **21**, 1676-1717. https://doi.org/10.1109/COMST.2018.2886932

[2] Buterin, V. (2014) A Next-Generation Smart Contract and Decentralized Application Platform. *White Paper*, **3**, No. 37.

[3] (2014) Ethereum and Oracles. https://blog.ethereum.org/2014/07/22/ethereum-and-oracles/

[4] Alhamad, M., Dillon, T. and Chang, E. (2010) Conceptual SLA Framework for Cloud Computing. 4*th IEEE International Conference on Digital Ecosystems and Technologies*, Dubai, 13-16 April 2010, 606-610.

[5] Faniyi, F. and Bahsoon, R. (2015) A Systematic Review of Service Level Management in the Cloud. *ACM Computing Surveys* (*CSUR*), **48**, Article No. 43. https://doi.org/10.1145/2843890

[6] Wood, G. (2014) Ethereum: A Secure Decentralised Generalised Transaction Ledger. *Ethereum Project Yellow Paper*, **151**, 1-32.

[7] Zhang, F., *et al*. (2016) Town Crier: An Authenticated Data Feed for Smart Contracts. *Proceedings of the* 2016 *ACM SIGSAC Conference on Computer and Communications Security*, Vienna, 24-28 October 2016, 270-282. https://doi.org/10.1145/2976749.2978326

[8] Nakashima, H. and Aoyama, M. (2017) An Automation Method of SLA Contract of Web APIs and Its Platform Based on Blockchain Concept. 2017 *IEEE International Conference on Cognitive Computing* (*ICCC*), Honolulu, 25-30 June 2017, 32-39. https://doi.org/10.1109/IEEE.ICCC.2017.12

[9] Ellis, S., Juels, A. and Nazarov, S. (2017) Chainlink: A Decentralized Oracle Network. *Retrieved March*, **11**, 2018.

[10] Scheid, E.J., Rodrigues, B.B., Granville, L.Z., *et al*. (2019) Enabling Dynamic SLA Compensation Using Blockchain-Based Smart Contracts. 2019 *IFIP/IEEE Symposium on Integrated Network and Service Management* (*IM*), Washington DC, 8-12 April 2019, 53-61.

[11] Tian, X., Li, M. and Tian, X. (2005) Design and Implementation of GPRS-Based Long-Distance Radio Unvarnished Transmission Terminal System. *Modern Electronic Technique*, **4**.

[12] Maarouf, A., Mifrah, Y., Marzouk, A., *et al*. (2018) An Autonomic SLA Monitoring Framework Managed by Trusted Third Party in the Cloud Computing. *International Journal of Cloud Applications and Computing* (*IJCAC*), **8**, 66-95.

[13] Ghosh, N. and Ghosh, S.K. (2012) An Approach to Identify and Monitor SLA Parameters for Storage-as-a-Service Cloud Delivery Model. 2012 *IEEE Globecom Workshops*, Anaheim, 3-7 December 2012, 724-729. https://doi.org/10.1109/GLOCOMW.2012.6477664

[14] De Carvalho, C.A.B., de Castro Andrade, R.M., de Castro, M.F., *et al*. (2017) State of the Art and Challenges of Security SLA for Cloud Computing. *Computers & Electrical Engineering*, **59**, 141-152. https://doi.org/10.1016/j.compeleceng.2016.12.030

[15] Venticinque, S., Aversa, R., Di Martino, B., *et al*. (2010) A Cloud Agency for SLA Negotiation and Management. *European Conference on Parallel Processing*, Ischia, 31 August-3 September 2010, 587-594. https://doi.org/10.1007/978-3-642-21878-1_72

[16] Zhou, H., Ouyang, X., Ren, Z., *et al*. (2019) A Blockchain Based Witness Model for Trustworthy Cloud Service Level Agreement Enforcement. *IEEE INFOCOM* 2019-*IEEE Conference on Computer Communications*, Paris, 29 April-2 May 2019, 1567-1575. https://doi.org/10.1109/INFOCOM.2019.8737580

[17] Arafat, N.A., Basu, D. and Bressan, S. (2019) Topological Data Analysis with Epsilon Net Induced Lazy Witness Complex. *International Conference on Database and Expert Systems Applications*, Linz, 26-29 August 2019, 376-392. https://doi.org/10.1007/978-3-030-27618-8_28

[18] Sartipi, F. (2020) Organizational Structure of Construction Entities Based on the Cooperative Game Theory. *Journal of Construction Materials*, **1**, 10 p.

[19] Saltari, E., Semmler, W. and Di Bartolomeo, G. (2021) A Nash Equilibrium for Differential Games with Moving-Horizon Strategies. *Computational Economics*, 1-14. https://doi.org/10.1007/s10614-021-10177-8

[20] Ranganthan, V.P., Dantu, R., Paul, A., *et al*. (2018) A Decentralized Marketplace Application on the Ethereum Blockchain. 2018 *IEEE 4th International Conference on Collaboration and Internet Computing* (*CIC*), Philadelphia, 18-20 October 2018, 90-97. https://doi.org/10.1109/CIC.2018.00023