JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 40, 899-918 (2024)
DOI: 10.6688/JISE.202407_40(4).0013

Enhancing Content Moderation in Wireless Mobile Networks: A Decentralized Quality Management Approach*

YAN-HUA NIU[1,2,+], SHUAI GAO[1], HONG-KE ZHANG[1] AND YUAN-JIA GONG[1,2]
[1]*School of Electronic and Information Engineering*
Beijing Jiaotong University
[2]*Academy of Broadcasting Science*
Beijing, 100044 P.R. China
E-mail: {18111072+; shgao; hkzhang; 21111086}@bjtu.edu.cn

With the rapid growth of wireless mobile networks and the proliferation of user-generated content, ensuring efficient and effective content moderation has become imperative. To address the challenges posed by the increasing workload in content moderation, along with the lack of management in third-party moderation, this paper presents a decentralized quality management mechanism for content moderation in wireless mobile networks. Our approach leverages permissioned blockchain to establish a transparent and trustworthy infrastructure. Through the utilization of smart contracts, we automate content moderation management rules, thereby enhancing management efficiency. Our mechanism combines quality evaluation and monetary incentives based on historical authenticity data. This not only incentivizes participants to consistently provide high-quality services but also ensures fairness within the system. Experimental results demonstrate the effectiveness of our approach in encouraging high-quality contributions, deterring low-quality data from bad-behaved participants, and improving performance. Security analysis reveals that the cost of collusion outweighs the potential benefits.

***Keywords*:** decentralized quality management, content moderation, permissioned blockchain, smart contract, decentralized autonomous organization (DAO)

1. INTRODUCTION

Content moderation is the process of monitoring and filtering content across different platforms to ensure it aligns with community guidelines, terms of service, and legal regulations. Its aim is to remove or restrict harmful, abusive, illegal, or inappropriate content, thus fostering a safe and positive online environment. As content consumption on wireless mobile networks continues to rise, it has become increasingly important to tailor content moderation strategies to this specific context. The exponential growth of content, particularly user-generated content (UGC) such as online videos, presents challenges for media providers in effectively moderating the vast volume of videos available on these networks. The existing centralized moderation conducted by media providers is unable to cope with the increasing demand for moderation caused by the surge in content volume. Consequently, third-party moderation has gained popularity as a means of alleviating the workload. However, the lack of management in third-party moderation poses a significant challenge in ensuring content quality. Additional checks must be implemented before publishing content online due to variations in AI model maturity and moderator ability. Additionally, these institutions are not directly held accountable for any content violations, resulting

Received August 14, 2023; revised September 18, 2023; accepted September 28, 2023.
Communicated by Xiaohong Jiang.
* This work was supported by the Fundamental Research Fund for Academy of Broadcasting Science, China under project JBKY20230230.

in inconsistent moderation quality. To address this issue, media providers must conduct their own review, further delaying the moderation process. This highlights the need for a reliable quality management mechanism to evaluate moderation and assign appropriate rewards or penalties to moderation institutions.

The current management of moderation quality relies solely on the contractual constraints between media providers and third-party moderation institutions. In traditional centralized management models, a common approach is to have sample content evaluated by an authoritative organization. However, in the context of the massive development of media content, obtaining comprehensive evaluation results from a limited number of samples becomes challenging. Additionally, the centralized management model suffers from data asymmetry issues such as collusion and data falsification, which prevent the verification of data authenticity and may result in untrustworthy outcomes.

Blockchain is a decentralized technology that enhances the credibility, transparency, and verifiability of existing management models due to its inherent characteristics of decentralization, transparency, and immutability [1-3]. It has found applications in various domains including decentralized DNS [4], energy [5], and vehicular social networks [6], By addressing data security concerns and improving management challenges such as coordinating participants' activities and resolving disputes over benefits [7, 8], blockchain presents itself as a solution. Therefore, we believe that adopting blockchain technology for moderation quality management can ensure the impartiality and transparency of data. With impartial and transparent data, the rewarding of high-quality contributions and the penalization of low-quality ones can create a positive feedback loop for moderation quality, resulting in long-term improvements in performance.

In addition to reliable data, effective management methods play a crucial role in moderation quality management. Firstly, improving management efficiency is paramount. Smart contracts, deployed on blockchains, offer automatic and self-executing capabilities, reducing reliance on human factors and enhancing management efficiency [9-11]. Decentralized autonomous organizations (DAOs) present an innovative management model based on blockchain and smart contracts, aiming to streamline complex management tasks and enhance overall efficiency [12, 13], serving as a valuable reference. Moreover, it is vital to motivate users to consistently and actively participate in moderation activities. Content moderation entails significant human and computational costs, and institutions are often reluctant to share moderation information without proper incentives. Therefore, implementing an incentive mechanism becomes essential to encourage participants to contribute more effectively. We have not come across any existing literature that specifically addresses a decentralized quality management model for content moderation.

Based on the aforementioned challenges, we introduce a novel decentralized quality management mechanism for content moderation, inspired by the concept of DAOs. This study makes the following key contributions:

- We propose a decentralized management mechanism for content moderation by leveraging the concept of DAOs. This approach stands out by combining quality evaluation with monetary incentives, with a specific focus on ensuring the authenticity of historical data, thus fostering sustainable and long-term quality improvements.
- Our implementation includes the design and deployment of management smart contracts. By automating predefined management rules, there smart contacts alleviate the complex-

ities associated with human intervention, streamlining the overall process.

- To validate the efficacy of our proposed mechanism, we conduct rigorous experiments on a permissioned blockchain. Through simulations, we explicitly analyze and mitigate the potential threat of low-quality contributions, showcasing the effectiveness of our approach in deterring their prevalence.

The remainder of this paper is organized as follows: In Section 2, we provide an overview of related works in the field. Section 3 outlines the system model based on a permissioned blockchain. In Section 4, we introduce the decentralized quality management mechanism for content moderation. Section 5 discusses the performance evaluation of our proposed mechanism. Section 6 analyzes the security aspects such as data authenticity and collusion costs. Finally, in Section 7, we present the conclusions of this study and outline future research directions.

2. RELATED WORK

The concept of DAOs originated in 2016 with the introduction of a venture capital fund driven by investors, which serves as a specific example of the broader concept of DAOs [14]. DAOs represent an innovative approach to organizational design, focusing on computerized rules or smart contracts that replace traditional centralized organizational structures in favor of decentralized operations [15]. Any changes to the rules within a DAO require consensus among participants to modify the pre-programmed code that governs its functioning [16].

While DAOs offer a promising solution to existing organizational management challenges, their practical implementation benefits greatly from the development of blockchain technology. The combination of DAOs and smart contracts provides an incredible means of achieving effective management, utilizing various computational techniques to generate autonomous and automatic management decisions.

Smart contracts are computer protocols that facilitate, verify, or enforce the negotiation and performance of contracts. They are developed on blockchain-based platforms like Ethereum [17, 18], Hyperledger Fabric [19], Corda [20], EOS [21], NEM [22] , and more. These smart contracts enable automatic execution by encoding rules or terms on the blockchain. When predefined conditions are met, the agreement can be enforced without any third-party intervention [23, 24]. By providing an automatic and error-free management process, smart contracts greatly eliminate human error.

DAOs have already had a significant impact on emerging technological categories such as decentralized apps (DApps) and decentralized finance (DeFi) [25]. Extensive research has been conducted on decentralized management and its applications in various domains, including cryptocurrency [26, 27], decentralized complex queries [28], social networks [15, 29], and more. For example, SmartCon [30] presents a blockchain-based framework for smart contract and transaction management that supports DAOs using separate blockchains. Crypto management [31] illustrates a decentralized management model for decision-making based on blockchain, smart contracts, non-fungible tokens (NFTs), and federated data. This model is characterized by its collaboration mechanism both on and off the blockchain.

Incentive mechanism is one of the most important issues for decentralized management. Before blockchain was introduced, the well-known distributed P2P file-sharing system such as Gnutella, Kazaa, and BitTorrent were working without thriving incentive mechanism to continue sharing storage and bandwidth [32]. With the advent of blockchain, the cryptocurrencies became an incentive for participants to share resources such as storage [33, 34], which is a tokenized incentive. In the field of federate learning, there have been some studies on incentive mechanism. In addition to monetary incentives [35, 36], some studies have introduced non-monetary incentives, including reputation [37, 38] and endorsement [39]. Reference [40] proposes a hybrid incentive mechanism consisting of monetary incentive and reputation approach. The above solutions aim to accurately evaluate the probability of data being selected in model training.

In the field of media-related decentralized management system, Steemit is a social media and content-driven platform that rewards creators and curators with "Steem tokens" [29]. LBRY is a decentralized online content marketplace that utilizes blockchain technology to create a community-controlled platform for publishing, accessing, and monetizing content [41]. Sasikala [42] *et al.* proposes a content management system DApp, leveraging blockchain and smart contracts to overcome the security issues of centralized content management systems. Shahabi [43] *et al.* designs a resource management system that can exploit the resources of distributed continuous media server network to achieve higher utilization and better reliability. DISPERSE [44] proposes a decentralized architecture for content and service delivery that provides resilience against node failure through location-independent content storage and replication. Niu [45] *et al.* proposes a blockchain-based content moderation scheme to enhance trust among diverse platforms and enable media platforms share moderation data using a permissioned blockchain. In fact, we have not found any literature on a decentralized quality management model for content moderation, nor on an incentive mechanism.

3. SYSTEM MODEL AND THREAT MODEL

The decentralized management mechanism is facilitated through permissioned blockchain and management smart contracts, involving media providers and content moderation institutions as participants. Unlike a centralized management model, decentralized management does not rely on a single entity to govern the system. Instead, management rules are established through consensus among all participants. By utilizing smart contracts, predefined rules can be automatically processed, ensuring fair and unbiased management. This approach aims to encourage high-quality contributions and foster long-term quality improvement in the wireless mobile network industry.

3.1 System Model

The decentralized quality management system model for content moderation consists of application layer, management layer and blockchain layer as shown in Fig. 1.

- Application layer

The proposed quality management mechanism relies on the use of trusted historical data from content moderation. This historical data includes records from content modera-

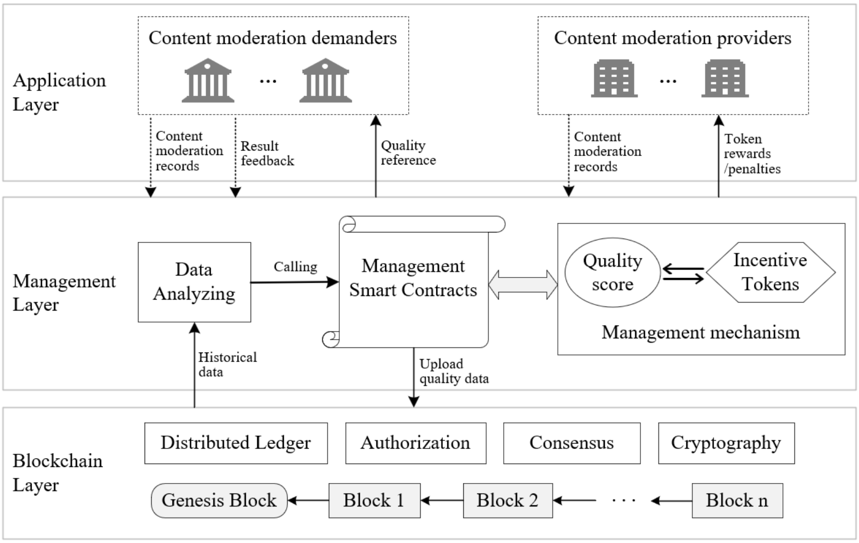


Fig. 1. Decentralized quality management system model for content moderation.

tion demanders (CMDs) and content moderation providers (CMPs), as well as feedback from CMDs. To ensure fairness, all this data is uploaded to the blockchain. By analyzing this data, the management layer can calculate the quality score for each CMP and determine appropriate token rewards or penalties. The quality score also serves as a reference for CMDs.

- Management layer

The management layer is facilitated by smart contracts deployed on blockchain nodes. These smart contracts operate independently in containers and can be triggered by application requests and execution triggers from the data analysis module. Designed for automated management and incentives, these smart contracts support various functions such as token creation, distribution, reward, deduction, and storage of scores and tokens. They provide a flexible interface to accommodate different application scenarios and cases. Any authorized node can access and retrieve information from the blockchain and trigger specific activities programmatically.

The data analysis module periodically collects historical performance data of all evaluated participants from the blockchain. When predefined conditions are met, the management smart contracts are executed. The management model encompasses the use of quality scores and incentive tokens. The quality score is employed to evaluate the performance and quality of CMPs, while the incentive tokens are utilized to encourage and reward high-quality contributions.

- Blockchain layer

The proposed quality management mechanism operates on a permissioned blockchain, involving participants such as CMDs, CMPs, and an administrative supervisor. To participate, all entities must undergo authentication and authorization processes. This ensures that only verified and authorized participants can access the network. With known and verified identities, the system becomes more secure and less prone to fraudulent activities. Through the implementation of authorization, consensus algorithms, and cryptographic techniques, the blockchain layer facilitates secure and efficient record-keeping. This, in turn, enables faster and more transparent transactions.

3.2 Threat Model

In addition to ensuring fair management, we must also address potential threats posed by malicious behavior. In traditional sharing economy models, participants are often motivated to share data, resulting in an influx of invalid or low-quality contributions. While permissioned blockchains can mitigate some of these issues by preventing junk data, the challenge of controlling low-quality contributions remains.

We specifically focus on the threat posed by participants engaging in malicious behavior, such as attempting to manipulate their scores or token rewards by sharing excessive amounts of low-quality moderation data as shown in Fig. 2.

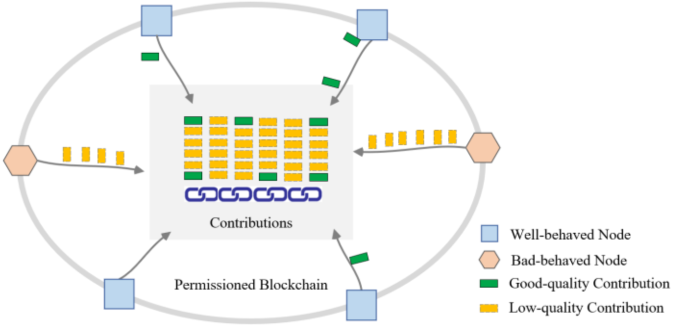


Fig. 2. Threat model.

When the bad-behaved nodes attempt to manipulate their score by uploading numerous low-quality contents, it can inundate the system with subpar contributions. This flood of low-quality contributions can negatively impact the experience of other participants and undermine the effectiveness of the quality management mechanism. Such behavior disrupts the fairness of the system. Therefore, this threat requires careful attention and countermeasures to maintain the integrity of the quality management mechanism.

4. DECENTRALIZED QUALITY MANAGEMENT MECHANISM

To ensure long-term and effective quality management, we propose a decentralized management mechanism that combines a quality score and incentive tokens. The quality score is determined by evaluating the CMP's contribution, adoption, error rate, experience value, and other factors, and it reflects the overall quality of the CMP. Incentive tokens serve as a monetary incentive method, offering a more direct and effective means of motivation [40]. We introduce incentive tokens as a necessary auxiliary method for moderation quality management, encouraging greater user participation and contributions to the system. This combination enables fair quality management and provides a basic model for long-term sustainable development. Additionally, defensive strategies have been incorporated into the mechanism to counter potential threats.

4.1 Quality Management Factors

To enhance the assessment of moderation quality of CMPs, we incorporate the following factors:

- Contribution value c_i: We denote $C_i = \{c_{i1}, c_{i2}, \ldots, c_{in}\}$ as the set of contribution items of participant node *i* within a given period *T*. The contribution value of node *i* is calculated by $c_{ij} = \sum_{j=1}^{n} \tau_{ij} c_{ij}$, here τ_{ij} represents the cost of contribution c_{ij} of different length of videos. We define $\tau_{ij} = \frac{d_{ij}}{d_s}$, $\tau_{ij} > 0$, here d_{ij} is the video duration of contribution c_{ij} and d_s is the reference duration, for example, a 30-minute-length video. Because the moderation cost of different length of videos varies greatly, τ_{ij} can nicely balance the benefits of different types of contributions. In other words, the contribution of a long-size video will be much more than that of a short video.

- Contribution adoption rate δ_i: we denote the contribution adoption rate of node *i* in the given period *T* as $\delta_i = \frac{|A_i|}{|C_i|}$, $0 \le \delta_i \le 1$, here A_i represents the set of the subscribed or adopted items by other participants, $A_i \subseteq C_i$. δ_i reflects the contribution quality of node *i*. When an item is adopted by any other participant, δ_i will increase. Therefore, there is a possibility of increasing δ_i through collusion with other participants.

- Contribution adoption factor ε_i: We denote the contribution adoption factor of node *i* in the given period *T* as $\varepsilon_i = \frac{\sum_{j=1}^{m} f_{ij}}{|A_i|}$, $\varepsilon_i \ge 1$, here f_{ij} represents the adoption frequency of a_{ij}. ε_i is introduced to increase the cost of collusion, since to collude with multiple participants is much more difficult. The combination of δ_i and ε_i better represent the quality and avoid fraud caused by collusion.

- Contribution error rate γ_i: We denote the contribution error rate that node *i* made in the given period *T* as $\gamma_i = \frac{|E_i|}{|C_i|}$, $0 \le \gamma_i \le 1$, here E_i represents the set of the items proven wrong by others, $E_i \subseteq C_i$. If the contribution item of node *i* is proved wrong by other participants, γ_i will increase.

- Experience factor α_i: we denote the experience value of node *i* as $\alpha_i = \sum_{j=1}^{l} \tau_{ij} c_{ij} - \sum_{j=1}^{k} \tau_{ij} e_{ij}$, here *l* represents the number of all the contribution made by node *i* and *k* represents the number of all error contribution made by node *i*. The experience factor indicates a CMP's business experience and basic competencies.

4.2 Quality Score

The quality score serves as a measure of moderation quality for CMPs. CMDs can identify the most suitable CMP for moderation services. Hence, the quality score holds significant importance for CMPs as it directly impacts their future business prospects, thereby playing a crucial role in effective management.

In order to fairly evaluate the contribution quality of each node, we denote quality factor vector for node i as $\vec{v} = [c_i^*, \delta_i^*, \varepsilon_i^*, \gamma_i^*, \alpha_i^*]$, here c_i^*, δ_i^*, and α_i^* are the normalization form of c_i, δ_i, and α_i by Eq. (1) respectively.

$$f(x) = \frac{x}{\sqrt{\sum_{i=1}^{n} x^2}}, 1 \le i \le |C_i| \tag{1}$$

Since the value range of ε_i varies greatly, we need to distinct gaps among different nodes with higher contribution adoption factor. For example, $\varepsilon_p = 30$, $1 \le p \le |C_i|$, and $\varepsilon_q = 50$, $1 \le q \le |C_i|$ can both indicate a high moderation quality of node i and node j. If we use the vector normalization method, we will get the result of $\varepsilon_i^* = 0.49$ and $\varepsilon_q^* = 0.82$, which will create a great contrast. Therefore, we choose Gompertz function [46], a commonly used sigmoid model, which is very suitable for our scenario, and adjust it to meet our requirement. ε_i^* can be calculated by

$$\varepsilon_i^* = e^{-e^{\lambda(\varepsilon_i - \phi)}}, \varepsilon_i \ge 1. \tag{2}$$

here λ is used to adjust the growth rate of curve and ϕ is a constant value used to adjust the function so that ε_i^* approaches zero infinitely when $\varepsilon_i = 1$, as shown in Fig. 3.

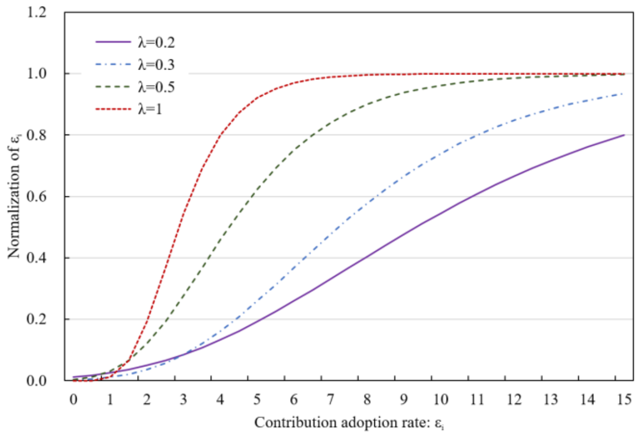


Fig. 3. Contribution adoption evaluated using Gompertz function under different growth rates.

Since error rate is the most important factor of moderation quality, γ_i^* is calculated depending on different conditions. When there is no error occurred, the node will get a full score. However, when γ_i reaches a certain threshold Γ, it must be re-examined and re-authorized, and its quality score will be directly assigned to invalid. Otherwise, γ_i^* will be deducted. γ_i^* can be calculated by

$$\gamma_i^* = \begin{cases} 1, & \gamma_i = 0 \\ -\frac{\gamma_i}{\Gamma}, & 0 \le \gamma_i < \Gamma, 1 \le i \le |C_i|. \\ invalid, & \gamma_i \ge \Gamma. \end{cases} \tag{3}$$

We denote the evaluation weight vector as $\vec{\omega}$. The quality score of node i can be calculated via

$$\Phi_i = \vec{v}_i \cdot \vec{\omega}. \tag{4}$$

It can be seen that the quality score is calculated on the historical data, so the quality evaluation result is objective. Furthermore, all the data are stored on blockchain, which ensures the immutability, thus the quality score cannot be revised by simply tamper any data.

4.3 Incentive Tokens

To establish an effective incentive system, the token reward policy holds utmost importance. In the initial phase, the priority is to attract a larger user base and encourage information sharing. Therefore, users are incentivized with higher rewards based on the quantity of information they share. As the system evolves a stage of high-quality development, it becomes crucial to emphasize the quality of the shared information to enhance its usability. Consequently, incentives should be directed towards promoting the dissemination of high-quality content.

Let T_i be the token value of node i and we define $T_i = \sum(T_{si}, T_{ai}, T_{qi})$ where T_{si} is the reward for sharing information, T_{ai} is the reward for being adopted by other participants, and T_{qi} is the reward for good quality.

(1) T_{si}

To promote high-quality information sharing and prevent unfair advantages from uploading low-quality content, we have developed a three-stage algorithm, which incorporates two key factors: the proportion of information shared by a specific node compared to the total information shared by all nodes, and the adoption rate δ_i. We establish thresholds for these factors, denoted as φ and σ respectively.

When T_{si} is less than the average tokens of all nodes, or δ_i is greater than the set threshold σ, T_{si} increases linearly according to the number of shared messages. When T_{si} is greater than the average token and the proportion of T_{si} to the total tokens of all nodes, and δ_i is less than σ, the growth rate decreases to φ. When the proportion of T_{si} exceeds φ, and the adoption rate is less than σ, T_{si} will not increase. This algorithm can avoid participants uploading a large amount of low-quality information for more tokens,

$$T_{si} = \begin{cases} T_{si} + \tau_{ij} p_b, & 0 \le T_{si} \le \frac{1}{n}\sum_{i=1}^{n} T_{si} \text{ or } \delta_i \ge \sigma, \\ T_{si} + \tau_{ij} p_b \varphi, & \frac{1}{n}\sum_{i=1}^{n} T_{si} \le T_{si} < \varphi \sum_{i=1}^{n} T_{si},\ \delta_i < \sigma, \\ T_{si}, & T_{si} \ge \varphi \sum_{i=1}^{n} T_{si},\ \delta_i < \sigma. \end{cases} \tag{5}$$

(2) T_{ai}

For any adopted contribution, participants who share the data will be rewarded accordingly, which will be paid by the adopter. T_{ai} can be calculated by

$$T_{ai} = T_{ai} + \tau_{ij} p_b. \tag{6}$$

Where p_b represents the basic incentive token of a standard contribution unit.

(3) T_{qi}

When the quality score reaches a certain high value ϵ, such as 0.8, the node will be rewarded with p_b token.

$$T_{qi} = T_{qi} + p_b, \qquad \Phi_i > \epsilon. \tag{7}$$

4.4 Smart Contracts

According to the proposed mechanism, we have developed management smart contracts to facilitate the process. The basic process is depicted in Fig. 4, which showcases the call relationship between the participating entities and smart contracts. These entities comprise super administrators, administrators, and the data processing module.

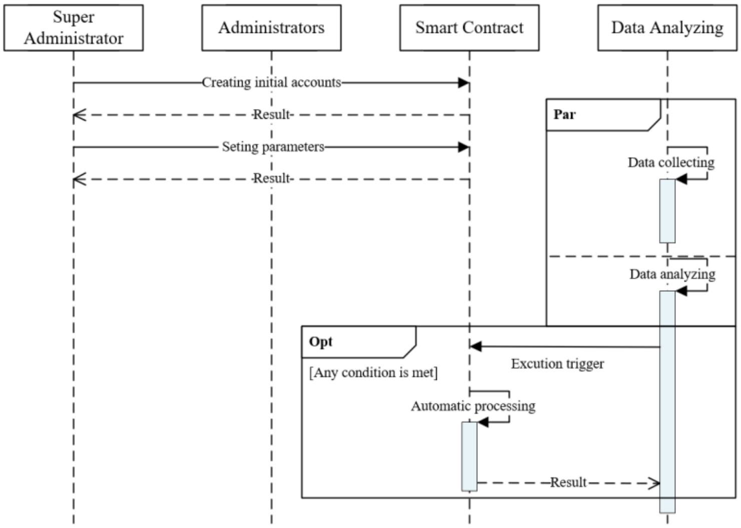


Fig. 4. Basic process of smart contract.

Super administrators and administrators are participant entities within smart contracts, each with different authorities. The role of the super administrator is to create the initial token data structure, while administrators are created by the super administrator through contract calls. The authorities of administrators can be configured based on different scenarios. Administrators have the ability to configure contract parameters and call the contract within their preset permissions. The data analyzing module collects and analyzes data in parallel. When certain preset conditions are met, the module triggers the smart contract to execute the corresponding processing.

Smart contracts function as chain code within the system, executing external interface requests, processing parameters, and ensuring functional operations and data storage while maintaining consistency and integrity. The management smart contract includes main functionalities such as account management, quality management, and smart contract invocation. Account management functions provide basic operations such as account and token creation. Quality management functions process quality scores and incentive tokens based on the aforementioned mechanisms. Smart contract invocation functions provide an interface for executing contracts to accommodate the needs of the management application, supporting both service-driven and administrator-driven invocations. This ensures the security and flexibility of the contract execution process. Additionally, several mechanisms are designed to enhance efficiency, including token rewards and timing processing.

(A) Automatic token rewards

We design the automatic token reward algorithm based on preset threshold as shown in Algorithm 1. When the quality score reaches the preset threshold (line 5 of Algorithm 1), the participant can automatically receive additional token rewards (line 6 of Algorithm 1).

Algorithm 1: Pseudocode for automatic token rewards.

```
1   base ← getBaseInfo(stub); user ← getUserInfo(stub, addr)
2   qs ← user.QualityScore
3   if base.EvaThreshold > 0
4           multiple ← qs/base.EvaThreshold
5               if multiple ≥ 1
6                   user.Token ← user.Token + base.AwardCount*multiple
7   saveUserInfo(stub, user)
```

(B) Timing trigger for quality evaluation

We design the time trigger algorithm for quality evaluation shown in Algorithm 2. It supports batch processing (line 3-4 of Algorithm 2) of the quality management. When the preset timing condition is met (line 5-7 of Algorithm 2), the evaluation is performed automatically.

Algorithm 2: Pseudocode of time trigger for quality evaluation.

```
1   base ← getBaseInfo(stub)
2   if base.EvaThreshold > 0 then
3           user ← getAllUserInfo(stub)
4           for user in range (users) do
5             if user.EvaStart.Equal(time.Time{})
6               durn ← time.Now().Hour() – user.EvaStartTime.Hour()
7               if unit64(durn) ≥ base.EvaluateTime
8                   user.QualityScore ← user.QualityEvaluate()
9                   user.EvaStartTime ← time.Now()
10            else
11              User.EvaStartTime ← time.Now()
12            saveUserInfo(stub, user)
```

5. PERFORMANCE EVALUATION

5.1 Simulation Setting

(A) Data setting

To better evaluate the robustness of our model, we conducted a simulation to test the impact of bad-behaved (those who continually share low-quality data) participants on our model. The simulation consists of 25 well-behaved participants who adhere to the normal behavior of not contributing low-quality data, as well as 5 bad-behaved participants.

We generated 5 factor matrices based on the distributions specified in Table 1. To enhance the realism of the simulation, the values of c_i for bad-behaved participants were set to be much higher compared to well-behaved participants. Both types of participants were assigned α_i values of 1 to eliminate the influence of experience value. The weight vector $\vec{\omega}$ was set as [0.25, 0.2, 0.25, 0.2, 0.1].

Table 1. Data setting of the scenario to test the impact of bad-behaved participants.

Type	c_i	δ_i	ε_i	γ_i	α_i
Well-behaved participants	c_i~$N(\mu, \sigma^2)$, $\mu = 50$, $\sigma^2 = 20$	δ_i~$N(\mu, \sigma^2)$ μ increases gradually from 0.1 to 0.5. σ increases gradually from 0.1 to 0.6.	ε_i increases ϵ gradually from 1. ϵ~$N(\mu, \sigma^2)$ $\mu = 0$, $\sigma^2 = 0.2$	γ_i increases ϵ gradually from a random value between (0, 0.5). ϵ~$N(\mu, \sigma^2)$	1
Bad-behaved participants	c_i~$U(a, b)$, $a = 200$, $b = 300$	δ_i~$N(\mu, \sigma^2)$ μ increases gradually from 0.1 to 0.3. σ increases gradually from 0.1 to 0.3.	$\varepsilon_i = 1$	Same as above	1

(B) Environment setting

We have developed a trial system consisting of five nodes deployed on a private cloud in our laboratory. The nodes are running the Ubuntu 18.04.5 LTS operating system and are equipped with Intel(R) Xeon(R) CPU E5-2650 v2 @ 2.60GHz, 8 cores, 32 gigabytes of memory, and a 500 GB hard disk. Our trial system is built on the ACME blockchain, which is an optimized version of PalletOne [47].

The management smart contracts were developed using Go, Java, and Node.js. They were then compiled into independent applications that run within isolated Docker containers. The Docker image of the contract is automatically generated during the deployment of the chain code. This approach ensures the security of the contract environment and mitigates the risk of system failure caused by malicious contract attacks.

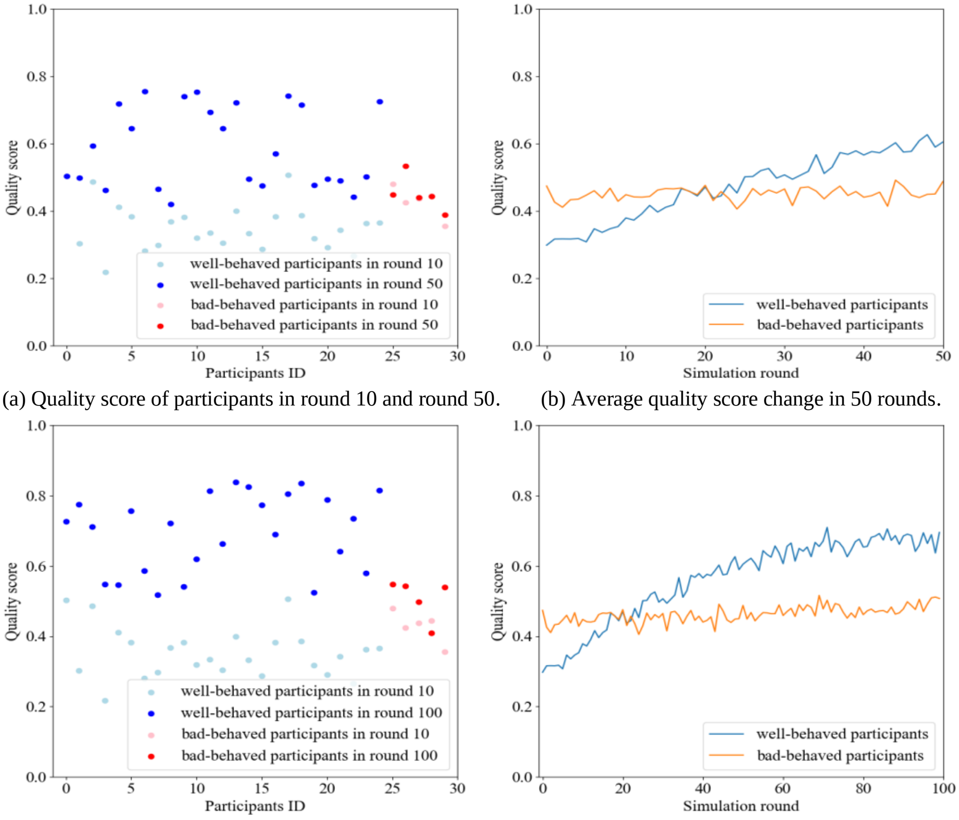
We conducted functionality and performance testing of the trial system over a remote connection, utilizing tools like Postman and JMeter. The data requests were encapsulated in JSON format to ensure a standardized interface. This paper presents some of the significant findings and key results from our testing efforts.

5.2 Experiment Result

We evaluate the performance of the proposed mechanism from quality score and smart contracts utility.

(A) Quality score

Quality score is an important metric in content moderation quality management. The participants who accumulate higher quality scores gain a competitive advantage. As shown in Fig. xxx, we compare the quality scores of different participants, which comprises 25 well-behaved participants and 5 bad-behaved ones.



(a) Quality score of participants in round 10 and round 50. (b) Average quality score change in 50 rounds.

(c) Quality score of participants in round 10 and round 500. (d) Average quality score change in 500 rounds.

Fig. 5. Comparison of quality scores between well-behaved participants and bad-behaved ones.

As discussed in Section 3.2, bad-behaved participants are continuing to share large amount of low-quality data for more token reward, we can see the quality scores of bad-behaved participants (shown in light red) are higher than that of well-behaved participants (shown in light blue) in round 10 as shown in Fig. 5 (a). However, in round 50 the quality scores of well-behaved participants (shown in blue) have improved significantly, while the change of bad-behaved participants (shown in red) has been small, and some are even lower than that of round 10. It can be seen that although the red ones have certain advantages at the beginning, they cannot get high quality scores afterwards, so even if they share a lot of data, they cannot get high scores. Moreover, it is worth mentioning that all the scores surpass 0.2. This outcome is attributed to the deliberate exclusion of error rate influences during the simulation process, which was to maintain the clarity of the algorithm's effect. Additionally, we conducted additional simulations incorporating random error rates, and the observed trend remained consistent.

Fig. 5 (b) shows more clearly the trend in average quality scores of different types of participants in 50 rounds. In the early rounds, the bad-behaved participants achieve higher average quality scores due to their substantial contributions. However, after approximately 20 rounds, the well-behaved participants consistently outperform them with higher average quality scores.

Fig. 5 (c) illustrates the comparison of quality scores between round 10 and round 100. In round 100, the quality scores of bad-behaved participants (shown in red) remain low as round 50. However, the well-behaved participants (shown in blue) clearly show improvement in their quality scores in round 100 compared to round 50, although a few experiences a decline. This discrepancy can be attributed to variations in the adoption rate and adoption factor of contributions among the participants.

Fig. 5 (d) illustrates the average quality scores of various participant types throughout the simulation. Upon analyzing the trends depicted by the two curves, it is apparent that the average quality score of bad-behaved participants remains relatively constant, whereas the score of well-behaved participants exhibits a continuous upward trend. The slower growth rate observed in the scores of well-behaved participants can be attributed to the influence of the contribution adoption factor ε_i. As depicted in Fig. 3, once ε_i reaches a certain threshold, the scores reach a plateau and cease to increase further.

Hence, these serves as evidence that the proposed mechanism efficiently addresses and reduces the impact of low-quality contributions.

(B) Performance of smart contract

We evaluate the performance of the smart contract from the perspective of time cost and throughput. We chose 5 main functions, namely accountCreate(), tokenAlloc(), pointsReward(), thresholdConf() and getInfo() and each function was tested 50 times. Fig. 6 shows the average response time and the average throughput of each function respectively. It can be observed that the time consumption of each function call is below 30ms. Specifically, the getInfo() function takes the least average time (around 24.8ms) and the account-

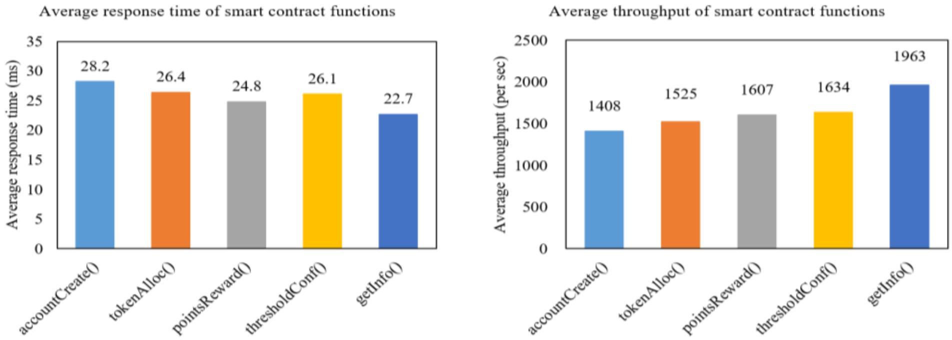


Fig. 6. Average response time and average throughout of smart contract functions.

Create() function takes the most average time (around 28.2ms). The average time cost and the average throughput of each function are within the acceptable range, which meets the management requirements for the permissioned blockchain participants.

We also found that when testing the same function multiple times, the test results may vary. This variability is mainly influenced by two factors: the size of the parameters used in each test and the state of the network during the tests. For example, if we test the tokenAlloc() function with different token allocation sizes, we may observe differences in the average response time and throughput. Similarly, if the network is experiencing high traffic or congestion during certain tests, it can affect the performance of the smart contract and lead to variations in the test results.

Based on the smart contracts, the numerical calculation and logical processing are automatically completed during the execution process, eliminating the need for manual intervention and supervision. Through automation, significant time and effort are saved for management personnel, resulting in a reduction in workload. Extensive manual handling and data queries may have been required, but now these operations can be accomplished automatically through smart contracts, alleviating the burden on management personnel. At the same time, since all operations are performed based on the contract, the security and reliability of the system are greatly improved, and risks caused by application operations are avoided.

6. SECURITY ANALYSIS

6.1 Data Authenticity

To guarantee the authenticity of moderation data, the proposed mechanism utilizes a permissioned blockchain with a centralized identity management system that includes a CA system. This generates a continuously updated digital identification to verify the authenticity of participants behavior, including CMDs and CMPs. Furthermore, to ensure the accuracy and high standards of content moderation, all participating CMPs must undergo qualification verification to confirm their professionalism and the quality of their content moderation service. The mechanism also enhances transparency and accountability by enabling the tracking of moderator actions. Moderation data is based on a real moderation service and stored on the blockchain, enabling any moderation data to be traced back to a corresponding transaction, thus ensuring the authenticity and reliability of the data. Overall, this mechanism fosters trust among all participants of this permissioned blockchain and creates a secure and dependable environment for data sharing.

6.2 Cost of Collusion

Some participants may collude to submit false data, thereby artificially inflating their quality scores and receiving higher token rewards. Due to the authenticity requirements for verifying service transaction, the cost of collusion is greatly increased, including the cost of time and expense. Malicious participants must construct the transactions in advance and pay the corresponding fee to the colluding party. Therefore, the cost of a collusion is equal to the base content moderation fee.

To evaluate the cost of collusion, we design a scenario with 25 well-behaved participants and 5 malicious participants. The malicious participants collude continuously without increasing their contribution values ($c_i \sim N(50, 16)$, $0 < c_i < 100$). We use the simulation data in round 1, round 20 and round 50 in Section 5.2 to represent the low-scoring malicious participants, the medium-scoring malicious participants and the high-scoring malicious participants, respectively. To clearly observe the influence of collusion on the quality scores, we intentionally disregarded the impacts of error rates and experience values in the simulation.

As shown in Fig. 7, selecting the points with faster growth, the low-scoring participants (shown in red) gain 0.23 points after 101 collusions with an average of about 0.0023 points per collusion. The medium-scoring participants (shown in green) gain 0.19 points after 101 collusions with an average of about 0.0019 points per collusion. The high-scoring participants (shown in blue) obtain 0.05 points after 101 collusions with an average of about 0.0005 points per collusion. After about 300 collusions, the quality score increases very little with the number of collusions. We can see that the cost of collusion is much higher than the benefit of score increases. At the same time, since the quality score is evaluated dynamically, the quality score will decrease if the contribution value remains unchanged for a certain period of time. This can therefore discourage the collusion to improve the quality score.

Similarly, collusion for token rewards is also discouraged. As seen in Fig. 7, the malicious participants need more than 500 collusions to achieve 0.8, which is a really high cost.

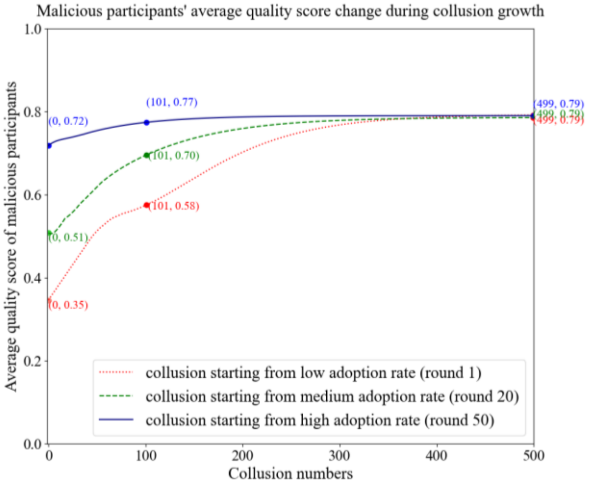


Fig. 7. Changes in average quality score of malicious participants during successive collusion growth without increase in contribution values.

7. CONCLUSION AND FUTURE WORK

In this paper, we propose a decentralized quality management mechanism for content moderation in wireless mobile networks, leveraging a permissioned blockchain. The use of a permissioned blockchain ensures secure participant management and guarantees the authenticity of moderation data. Our approach incorporates management smart contracts to automate management tasks, while employing a token and points-based management mechanism as a core economic strategy. We conducted experiments to evaluate the effecttiveness of our quality management mechanism and its performance. The results demonstrate that our mechanism successfully mitigates a significant number of low-quality contributions. Furthermore, the smart contracts effectively execute predefined management rules with acceptable performance, simplifying management complexity and reducing the need for human intervention. Security analysis confirms that our proposed mechanism ensures data authenticity through the underlying permissioned blockchain. Additionally, the deterrent effect of collusion suppresses the cheating for higher quality scores.

In future research, we will investigate additional threat scenarios, with a specific focus on detecting collusion among participants. Although limitations in real-world data prevented us from including this aspect in the current study, we plan to conduct further experiments and analyze the characteristics of different types of malicious behavior using acquired data. Our goal is to enhance the security of our proposed mechanism by incorporating more comprehensive threat models.

REFERENCES

1. A. Extance, "The future of cryptocurrencies: Bitcoin and beyond," *Nature*, Vol. 526, 2015, pp. 21-23.
2. W. Zhang, G. Sun, L. Xu, Q. Lu, H. Ning, P. Zhang, and S. Yang, "A trustworthy safety inspection framework using performance-security balanced blockchain," *IEEE Internet of Things Journal*, Vol. 9, 2022, pp. 8178-8190.
3. S. Filipčić, "Web3 & DAOs: an overview of the development and possibilities for the implementation in research and education," in *Proceedings of the 45th Jubilee International Convention on Information*, 2022, pp. 1278-1283.
4. M. Ali, J. Nelson, R. Shea, and M. J. Freedman, "Blockstack: A global naming and storage system secured by blockchains," in *Proceedings of USENIX Conference on Usenix Annual Technical Conference*, 2016, pp. 181-194.
5. Z. Li, J. Kang, R. Yu, D. Ye, Q. Deng, and Y. Zhang, "Consortium blockchain for secure energy trading in industrial internet of things," *IEEE Transactions on Industrial Informatics*, Vol. 14, 2018, pp. 3690-3700.
6. K. Fan, Q. Pan, K. Zhang, Y. Bai, S. Sun, H. Li, and Y. Yang, "A secure and verifiable data sharing scheme based on blockchain in vehicular social networks," *IEEE Transactions on Vehicular Technology*, Vol. 69, 2020, pp. 5826-5835.
7. J. Zhu, J. Cao, D. Saxena, S. Jiang, and H. Ferradi, "Blockchain-empowered federated learning: Challenges, solutions, and future directions," *ACM Computing Surveys*, Vol. 55, 2023, pp. 1-31.
8. Y. Qu, M. P. Uddin, C. Gan, Y. Xiang, L. Gao, and J. Yearwood, "Blockchain-enabled federated learning: A survey," *ACM Computing Surveys*, Vol. 55, 2022, pp. 1-35.
9. M. Kaleem, K. Kasichainula, R. Karanjai, L. Xu, Z. Gao, L. Chen, and W. Shi, "An event driven framework for smart contract execution," in *Proceedings of the 15th ACM International Conference on Distributed and Event-based Systems*, 2021, pp. 78-89.
10. K. Wüst, S. Matetic, S. Egli, K. Kostiainen, and S. Capkun, "ACE: Asynchronous and concurrent execution of complex smart contracts," in *Proceedings of ACM SIGSAC Conference on Computer and Communications Security*, 2020, pp. 587-600.
11. A. Mense and M. Flatscher, "Security vulnerabilities in ethereum smart contracts," in *Proceedings of the 20th International Conference on Information Integration and*

Web-based Applications and Services, 2018, pp. 375-380.

12. R. Chotkan, J. Decouchant, and J. Pouwelse, "Unstoppable DAOs for web3 disruption," in *Proceedings of the 3rd International Workshop on Distributed Infrastructure for the Common Good*, 2022, pp. 37-42.
13. B. Nissen, K. Symons, E. Tallyn, C. Speed, D. Maxwell, and J. Vines. "New value transactions: Understanding and designing for distributed autonomous organisations," in *Proceedings of ACM Conference Companion Publication on Designing Interactive Systems*, 2017, pp. 352-355.
14. U. W. Chohan, "The decentralized autonomous organization and governance issues," *SSRN Electronic Journal*, 2017, pp. 1-6.
15. C. Chao, I. Ting, Y. Tseng, B. Wang, S. Wang, Y. Wang, and M. Chen, "The study of decentralized autonomous organization (DAO) in social network," in *Proceedings of the 9th Multidisciplinary International Social Networks Conference*, 2022, pp. 59-65.
16. V. Dwivedi, V. Pattanaik, V. Deval, A. Dixit, A. Norta, and D. Draheim, "Legally enforceable smart-contract languages: A systematic literature review," *ACM Computing Surveys*, Vol. 54, 2021, pp. 1-34.
17. Ethereum Whitepaper, https://ethereum.org/en/whitepaper/, 2023.
18. Home | Ethereum.org, https://ethereum.org/en/, 2023.
19. Y. Manevich, A. Barger and Y. Tock, "Endorsement in hyperledger fabric via service discovery," *IBM Journal of Research and Development*, Vol. 63, 2019, pp. 2:1-2:9.
20. R. G. Brown, *The Corda Platform: An Introduction*, Whitepaper, 2018.
21. Home Eosio Blockchain Software & Services, https://eos.io/, 2023.
22. Nem Ecosystem Blockchain – Because Together, Everything is Possible, http://nem.io/.
23. Z. Zheng, S. Xie, H. Dai, W. Chen, X. Chen, J. Weng, and M. A. Imran, "An overview on smart contracts: Challenges, advances and platforms," *Future Generation Computer Systems*, Vol. 105, 2020, pp. 475-491.
24. S. Wang, Y. Yuan, X. Wang, J. Li, R. Qin and F.-Y. Wang, "An overview of smart contract: Architecture, applications, and future trends," in *Proceedings of IEEE Intelligent Vehicles Symposium*, 2018, pp. 108-113.
25. R. Ushida and J. Angel, "Regulatory considerations on centralized aspects of DeFi managed by DAOs," *Financial Cryptography and Data Security*, 2021, pp. 21-36.
26. Y. Hsieh, J. Vergne, P. Anderson, K. Lakhani, and M. Reitzig, "Bitcoin and the rise of decentralized autonomous organizations," *Journal of Organization Design*, 2018, pp. 1-15.
27. L. Mosley, H. Pham, X. Guo, Y. Bansal, E. Hare, and N. Antony, "Towards a systematic understanding of blockchain governance in proposal voting: A dash case study," *Blockchain: Research and Applications*, Vol. 3, 2022, pp. 1-11.
28. M. Zichichi, L. Serena, S. Ferretti, and G. D'Angelo, "Governing decentralized complex queries through a DAO," in *Proceedings of Conference on Information Technology for Social Good*, 2021, pp. 121-126.
29. B. Guidi, A. Michienzi, and L. Ricci, "Analysis of witnesses in the steem blockchain," *Mobile Networks and Applications*, Vol. 26, 2021, pp. 2099-2110.
30. M. Muneeb, Z. Raza, I. U. Haq, and O. Shafiq, "SmartCon: A blockchain-based framework for smart contracts and transaction management," *IEEE Access*, Vol. 10, 2022, pp. 23687-23699.
31. G. Wang, J. Li, X. Wang, J. Li, Y. Yuan, and F.-Y. Wang, "Blockchain-based crypto management for reliable real-time decision-making," *IEEE Transactions on Computa-*

tional Social Systems, Vol. 10, 2022, pp. 3333-3342.

32. V. H. Lakhani, "Token-based incentive schemes in decentralized P2P storage networks," in *Proceedings of the 22nd International Middleware Conference Doctoral Symposium*, 2021, pp. 23-24.
33. A. Miller, A. Juels, E. Shi, B. Parno, and J. Katz, "Permacoin: Repurposing bitcoin work for data preservation," in *Proceedings of IEEE Symposium on Security and Privacy*, 2014, pp. 475-490.
34. H. Kopp, C. Bösch, and F. Kargl, "Koppercoin – a distributed file storage with financial incentives," *Information Security Practice and Experience*, Vol. 10060, 2016, pp. 79-93.
35. A. Ghorbani and J. Zou, "Data shapley: Equitable valuation of data for machine learning," in *Proceedings of International Conference on Machine Learning*, 2019, pp. 2242-2251.
36. R. Jia, D. Dao, B. Wang, F. A. Hubis, N. Hynes, N. M. Gurel, B. Li, C. Zhang, D. Song, and C. Spanos, "Towards efficient data valuation based on the shapley value," in *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*, Vol. 89, 2019, pp. 1167-1176.
37. Y. Zhao, J. Zhao, L. Jiang, R. Tan, D. Niyato, Z. Li, L. Lyu, and Y. Liu, "Privacy-preserving blockchain-based federated learning for IoT devices," *IEEE Internet of Things Journal*, Vol. 8, pp. 1817-1829.
38. M. H. ur Rehman, K. Salah, E. Damiani, and D. Svetinovic, "Towards blockchain-based reputationaware federated learning," in *Proceedings of IEEE Conference on Computer Communications Workshops*, 2020, pp. 183-188.
39. T. Luo, S. S. Kanhere, and H. Tan, "SEW-ing a simple endorsement web to incentivize trustworthy participatory sensing," in *Proceedings of the 11th Annual IEEE International Conference on Sensing, Communication, and Networking*, 2014, pp. 636-644.
40. M. Qi, Z. Wang, S. Chen, and Y. Xiang, "A hybrid incentive mechanism for decentralized federated learning," *Distributed Ledger Technologies: Research and Practice*, Vol. 1, 2022, pp. 1-15.
41. J. Li, A. Grintsvayg, J. Kauffman, and C. Fleming, "LBRY: A blockchain-based decentralized digital content marketplace," in *Proceedings of IEEE International Conference on Decentralized Applications and Infrastructures*, 2020, pp. 42-51.
42. N. Sasikala, B. M. Sundaram, S. Biswas, A. Sai Nikhil, and V. S. Rohith, "Survey of latest technologies on decentralized applications using blockchain," in *Proceedings of the 2nd International Conference on Artificial Intelligence and Smart Energy*, 2022, pp. 1432-1436.
43. C. Shahabi and F. Banaei-Kashani, "Decentralized resource management for a distributed continuous media server," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 13, 2002, pp. 710-727.
44. S. Anand, D. Ding, P. Gasti, M. O'Neal, M. Conti, and K. S. Balagani, "DISPERSE: A decentralized architecture for content replication resilient to node failures," *IEEE Transactions on Network and Service Management*, Vol. 17, 2010, pp. 201-212.
45. Y. Niu, S. Gao, and H. Zhang, "A trustworthy content moderation scheme based on permissioned blockchain," in *Proceedings of International Conference on Emerging Networking Architecture and Technologies*, Vol. 1696, 2022, pp. 131-145.
46. P. Waliszewski and J. Konarski, "A mystery of the Gompertz function," *Fractals in Biology and Medicine*, 2005, pp. 277-286.

47. W. Ou, S. Huang, J. Zheng, Q. Zhang, G. Zeng, and -W. Han, "An overview on cross-chain: Mechanism, platforms, challenges and advances," *Computer Networks*, Vol. 218, 2022, pp. 1-21.

Yan-Hua Niu (牛妍华) received the BS and MS degrees in Communication and Information Systems from Beijing Jiaotong University in 2002 and 2005 respectively. She is currently a Senior Engineer in Academy of Broadcasting Science, China, and is pursuing the Ph.D. degree at Beijing Jiaotong University. She is mainly engaged in the research of trustworthy media content management and blockchain. She has published three international standards and authored more than ten patents.

Shuai Gao (郜帅) received the BS, MS, and Ph.D. degrees in Communication and Information Systems from Beijing Jiaotong University in 2001, 2004, and 2010, respectively. He is currently a Full Professor with the School of Electronic and Information Engineering, Beijing Jiaotong University. His research interests are in the areas of Internet architecture, computing aware networks, and mobile Internet. He has published more than 30 high-quality papers and seven international standards.

Hong-Ke Zhang (张宏科) is a Fellow of the Chinese Academy of Engineering, a Professor of Beijing Jiaotong University mainly specialized in network technologies, the Director of Chinese National Engineering Center on Mobile Specialized Network. He has been engaged in research of architecture and protocol design for the future Internet and specialized networks. He has won two prizes of the national technological invention award of China and authored more than ten books and a hundred patents.

Yuan-Jia Gong (龚媛嘉) received the BS degree in Communication and Information Systems from Beijing Jiaotong University in 2016, and got the MS degree in KTH Royal Institute of Technology, Sweden. She is currently employed in Academy of Broadcasting Science, China as an Engineer, and is pursuing the Ph.D. degree at Beijing Jiaotong University. She is mainly engaged in the research of network communication systems.