Incentive Mechanism Design for Blockchain Enabled Distributed Content Caching

Jia Guo*†, Wei Wang*†, Zuguang Li*†, Shan Sun*†
*College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, China
†Key Laboratory of Dynamic Cognitive System of Electromagnetic Spectrum Space, Nanjing, China 211106
Email: {guojia; wei_wang; zuguang_li; sunshan}@nuaa.edu.cn

2021 IEEE International Performance, Computing, and Communications Conference (IPCCC) | 978-1-6654-4331-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/IPCCC51483.2021.9679366

***Abstract*—By fully utilizing the caching space of nearby smart users, mobile user caching is a promising solution for efficient content delivery service. However, how to motivate mobile users to share their spare caching space and protect the privacy information during caching sharing is a challenging issue. In this paper, we consider a blockchain enabled mobile user caching on the basis of blockchain network, thus the security and privacy issues can be well addressed, and the contributions of each mobile user can be recorded in the blockchain. Specifically, the interactions between the content service provider and the mobile users are modelled as a Stackelberg game, and the equilibrium is analyzed. Simulation results demonstrate the effectiveness of the proposed scheme and it is shown that the optimal strategies of content service provider and caching users can be established with different cache cost and caching requirement.**

I. Introduction

With the development and commercialization of 5G, wireless data services are growing explosively, among which over 75% are video or multimedia content[1]. To better support the hot wireless content delivery services, edge caching where popular contents are cached in edge nodes that close to the end users is a promising solution. With edge caching, some contents can be pre-stored during off-peak hours, and thus the traffic burden of the core network can be reduced in peak hours. In addition, the transmission delay can be significantly reduced without fetching from the server. Therefore, edge caching has attracted extensive research interests over the past years [1]–[3].

A distributed proactive edge caching mechanism was proposed in [3] to optimize the cost of content retrieval and improve the experience of mobile users. In [4], combined with the request mode and user behavior habits, a geographic cooperative caching strategy for mobile video transmission was designed. Zhang et al. [5] investigated the cooperative edge cache scheme with optimal delay in large-scale user centric mobile networks, based on the information of network topology, traffic distribution, channel quality, and file popularity. Even though edge caching is effective for content-centric services, it is not cost effective due to the diversified user interests. Fortunately, with the emerging caching capability of end users, it is possible to distribute the content caching task to surrounding users without building edge caching servers [6]–[8]. But note that users are usually selfish and unwilling to contribute the caching space without proper reward. Thus, how to encourage end users to share their caching space as much as possible becomes a critical problem. Zou et al. [9] proposed a joint video pricing and cache placement strategy to maximize the profit of the video provider and the mobile network operator. In [10], an optimal auction mechanism was proposed to optimize the allocation of cache resources from the perspective of service providers.

However, the distributed content caching also faces many challenges, such as how to promote the interaction between the content service provider (CSP) and caching users (CUs) in the trustless environment, how to protect the privacy information of CUs, etc. As a new paradigm of data storage and information sharing, blockchain is a good candidate for enabling distributed caching, where the caching request and transactions can be recorded in the block without tamper-proof and thus the revenues of the caching contributors can be guaranteed [7], [11], [12]. In [7], a blockchain based edge cache service framework for cache resource and digital content sharing is designed with blockchain-based credentials and double-auction mechanism. To deal with the trust management problem in edge caching, a blockchain-based trustworthy edge caching scheme was proposed in [11], where the caching transactions between the edge nodes and mobile users was recorded with blockchain without modification, and a trust management mechanism was designed for mobile users.

Based on above research, in this paper, we propose an incentive mechanism for blockchain-enabled distributed content caching. The CSP and CUs are formulated as a Stackleberg game to optimize the caching reward and the optimal caching size of each CU. The contributions of this paper are summarized as follows:

- We propose a blockchain enabled distributed caching framework, where the CSP releases the caching requirements and rewards, and CUs on the blockchain response to the caching task accordingly. The caching transactions are recorded on the blockchain with tamper-proof and traceability, and the privacy of the CU can be protected.
- A Stackleberg game is formulated to obtain the optimal caching reward and the caching size of each CU. The game is analyzed in detail and optimal strategy is derived.
- Simulation results are given to evaluate the performance

[1]Cisco's Visual Networking Index forecast (VNI) https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.pdf

978-1-6654-4331-9/21/$31.00 ©2021 IEEE

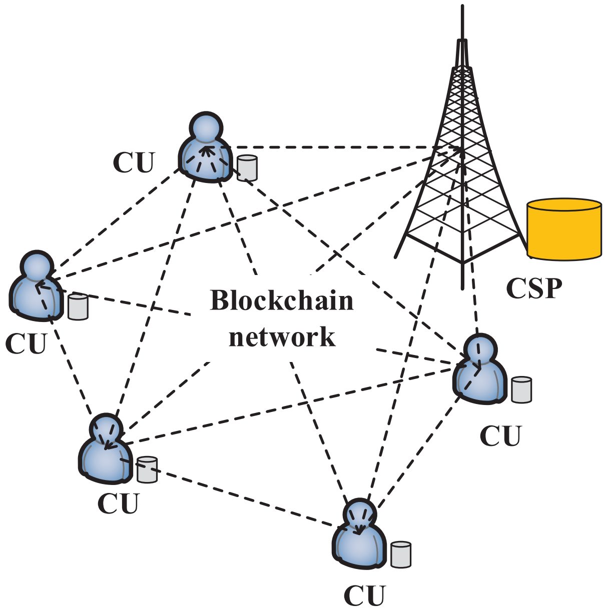

Fig. 1. The blockchain enabled distributed caching model.

of the proposed framework, it is shown that the optimal strategies of content service provider and caching users can be established with different caching cost and caching requirement.

The remainder of this paper is organized as follows. The system model and problem formulation are described in Section II. In Section III, we analyze the Stackelberg game and derive the optimal strategy for CSP and CUs. Simulation results are presented in Section IV, and conclusions are given Section V.

II. System Model and Problem Formulation

We consider a distributed content caching scenario, where a content service provider (CSP) aims to provide content subscription services for registrated users. To save the storage space and the maintenance cost, the CSP will incentive surrounding mobile users with caching capability to participate in the caching services, i.e., the traditional centralized caching can be distributed among caching users (CUs), to fully reap the benefits of distributed caching.

To protect the revenue of the participating caching users and prevent possible privacy information leakage, we consider a public blockchain enabled caching scenario. As shown in Fig. 1, the CSP and M CUs participate in the caching services based on the blockchain system. The CSP first publishes caching tasks and corresponding rewards by completing them on the blockchain network, then CUs on the blockchain receive the task and decide whether to join the caching task, based on the given reward and their own cost.

Suppose that the intended caching requirement of the CSP is Q, with R being the corresponding caching reward. Denote the CUs set as $\mathcal{M} = \{m_1, m_2, \ldots, m_M\}$, with M being the cardinality of $\mathcal{M}$, then the utility function of CSP can be given by

$$U_{CSP} = \alpha \sum_{i=1}^{M} \log_2 (1 + \beta\mu_i) - R, \tag{1}$$

where μ_i is the caching space provided by CU m_i, $\alpha, \beta > 0$, are constant values.

For CUs, they share the total rewards given by CSP based on their caching contribution. The more caching contents that CU contributes for CSP, the higher rewards will be given. We use p_i to represent the proportion of caching that CU m_i contributes, which is given by

$$p_i = \frac{\mu_i}{\sum_{j=1}^{M} \mu_j}. \tag{2}$$

Then the utility of m_i can be defined as:

$$U_i = p_i R - \lambda_i \mu_i, \tag{3}$$

where λ_i is the unit caching cost of m_i.

Because the CSP and CUs are smart and selfish, they will dynamically adjust their strategies to maximize their profits. Therefore, game theory is a suitable tool to analyze this problem. We can model the interaction between the CSP and the CUs as a two-stage Stackleberg game and analyze the optimal strategy of both sides.

Based on the Stackleberg game model, a leader and some followers compete with each other to maximize their respective profits. In this paper, we formulate the CSP as the leader and the CUs as the followers. In the upper stage, the CSP sets the reward to encourages CUs to share as much cache space as possible. The leader (i.e., CSP) needs to find the best reward R to maximize its profit with a minimum caching size constraint Q. In the lower stage, each follower (i.e., CU) decides the optimal amount of caching space μ_i they contribute by analysing the reward and its cost. The optimization problems can be formulated as follows.

Given the reward R and other CUs' caching strategies $\boldsymbol{\mu}_{-i} = \{\mu_1, \mu_2, \cdots, \mu_{i-1}, \mu_{i+1}, \cdots, \mu_M\}$, the CU m_i decides its own cache strategy μ_i to maximize its own profit. This subproblem can be formulated as follows:

Problem 1. Caching user's sub-game

$$\begin{aligned} \max_{\mu_i} \quad & U_i \left(\mu_i \mid \boldsymbol{\mu_{-i}}, R\right) \\ \text{s.t.} \quad & \mu_i \geq 0. \end{aligned} \tag{4}$$

For the upper stage of the game, the CSP dynamically adjusts the reward R to maximize its benefits with guaranteed caching requirements Q. This sub-problem can be formulated as follows:

Problem 2. Content service provider's sub-game

$$\begin{aligned} \max_{R} \quad & \mathrm{U}_{CSP}(R, \boldsymbol{\mu}) \\ \text{s.t.} \quad & \textstyle\sum_{i=1}^{M} \mu_i \geq Q. \end{aligned} \tag{5}$$

The object of the game is to find a Stackelberg Equilibrium (SE) point(s) from which neither the leader (CSP) nor the followers (CUs) can bilaterally change their strategies to increase its utility without affecting the opponents. For the proposed Stackelberg game, the SE can be defined as follows.

Definition 1 (Stackelberg Equilibrium). *Let R^* be a solution for the CSP's sub-problem and $\boldsymbol{\mu}^*$ be a solution for the CU's sub-problem, where $\boldsymbol{\mu}^* = \{\mu_1^*, \mu_2^*, \cdots, \mu_M^*\}$. Then $(\boldsymbol{\mu}^*, R^*)$ is a SE for the proposed Stackelberg game if it satisfies the following two conditions:*

$$U_{CSP}(R^*, \boldsymbol{\mu}^*) \geq U_{CSP}(R, \boldsymbol{\mu}^*), \forall R \geq 0, \tag{6}$$

and

$$U_i({\mu_i}^* | \boldsymbol{\mu}^*_{-\boldsymbol{i}}, R^*) \geq U_i(\mu_i | \boldsymbol{\mu}^*_{-\boldsymbol{i}}, R^*), \forall \mu_i. \tag{7}$$

III. Game Analysis

In this section, we analyze the existence and the uniqueness of the Stackelberg equilibrium of the proposed Stackelberg game. First, we analyze the lower stage of the game to find the Nash equilibrium of the CU's sub-problem. On the basis, we analyze how to maximize the utility of CSP in the upper stage.

A. Analysis of CU's sub-game

Given a caching target Q and the corresponding reward R, each CU will adaptively determine their caching space to maximize their own profits. We first prove the existence of the Nash equilibrium in the CU's sub-problem.

Theorem 1. *The Nash equilibrium exists in the CU's sub-problem.*

Proof. From (3), we can know that $\mu_i \leq R/\lambda_i$ to ensure the utility function U_i larger than zero. So μ_i is continuous in $[0, R/\lambda_i]$, which is a nonempty, convex, and compact subset of the Euclidean space. The first and second derivatives of the utility function are given as follows:

$$\frac{\partial U_i}{\partial \mu_i} = R\frac{\sum_{j=1}^{M} \mu_j - \mu_i}{\left(\sum_{j=1}^{M} \mu_j\right)^2} - \lambda_i, \tag{8}$$

$$\frac{\partial^2 U_i}{\partial \mu_i^2} = -2R\frac{\sum_{j=1}^{M} \mu_j - \mu_i}{\left(\sum_{j=1}^{M} \mu_j\right)^3} \leq 0. \tag{9}$$

Therefore, the utility function U_i is a concave function, thus there exists a Nash equilibrium in the CU's sub-problem. ■

With Theorem 1, we can obtain that for a reward R and other CUs' strategies $\boldsymbol{\mu}_{-i}$ the optimal caching strategy of m_i can be calculated by setting the first derivative in (8) as zero, i.e.,

$${\mu_i}^* = \sqrt{\frac{R\sum_{j=1}^{M} \mu_j - \mu_i}{\lambda_i}} - \sum_{j \neq i}^{M} \mu_j. \tag{10}$$

As $\mu_i \geq 0$, the best strategy for m_i is given by

$$\mu_i^* = \begin{cases} 0, & \text{if } \frac{R}{\sum_{j\neq i}^{M} \mu_j} \leq \lambda_i, \\ \sqrt{\frac{R\sum\limits_{j=1}^{M} \mu_j - \mu_i}{\lambda_i}} - \sum\limits_{j \neq i}^{M} \mu_j, & \text{otherwise.} \end{cases} \tag{11}$$

Corollary 1. *Given the optimal caching strategies $\boldsymbol{\mu}^* = \{\mu_1^*, \mu_2^*, \cdots, \mu_M^*\}$, for any two CUs m_i and m_j, with their caching strategy μ_i^* and μ_j^*, if $\lambda_i \leq \lambda_j$, then $\mu_i^* \geq \mu_j^*$.*

Proof. This can be proved with contradiction. Suppose that for the two CUs m_i and m_j, $\lambda_i \leq \lambda_j$, $\mu_j^* > \mu_i^* \geqslant 0$. As $\mu_j^* \geq 0$, according to (8) and (11), we can know $\frac{\partial U_j}{\partial \mu_j}\left(\mu_j^* | \boldsymbol{\mu}^*_{-\boldsymbol{j}}\right) = 0$.

Similarly, if $\mu_i^* > 0$, we have $\frac{\partial U_i}{\partial \mu_i}\left(\mu_i^* | \boldsymbol{\mu}^*_{-\boldsymbol{i}}\right) = 0$. If $\mu_i^* = 0$, according to (11), we have $\frac{R}{\sum_{j\neq i}^{M} \mu_j} \leqslant \lambda_i$, substituting to (8),we have

$$\begin{aligned} \frac{\partial U_i}{\partial \mu_i}\left(\mu_i^* | \boldsymbol{\mu}^*_{-\boldsymbol{i}}\right) &= R\frac{\sum_{j=1}^{M} \mu_j - \mu_i}{\left(\sum_{j=1}^{M} \mu_j\right)^2} - \lambda_i \\ &\leq R\frac{\sum_{j=1}^{M} \mu_j - \mu_i}{\left(\sum_{j=1}^{M} \mu_j\right)^2} - \frac{R}{\sum_{j\neq i}^{M} \mu_j} \\ &= R\frac{\sum_{j\neq i}^{M} \mu_j}{\left(\sum_{j=1}^{M} \mu_j\right)^2} - R\frac{\sum_{j\neq i}^{M} \mu_j}{\left(\sum_{j\neq i}^{M} \mu_j\right)^2} \\ &< 0. \end{aligned} \tag{12}$$

Therefore, if $\mu_i^* \geqslant 0$, we have $\frac{\partial U_i}{\partial \mu_i}\left(\mu_i^* | \boldsymbol{\mu}^*_{-\boldsymbol{i}}\right) \leq 0$.

As $\lambda_i \leqslant \lambda_j$ and $\mu_j^* > \mu_i^* \geq 0$, we have

$$\begin{aligned} \frac{\partial U_j}{\partial \mu_j}\left(\mu_j^* | \boldsymbol{\mu}^*_{-\boldsymbol{j}}\right) &= R\frac{\sum_{k=1}^{M} \mu_k - \mu_j^*}{\left(\sum_{k=1}^{M} \mu_k\right)^2} - \lambda_j \\ &< R\frac{\sum_{k=1}^{M} \mu_k - \mu_i^*}{\left(\sum_{k=1}^{M} \mu_k\right)^2} - \lambda_i \\ &= \frac{\partial U_i}{\partial \mu_i}\left(\mu_i^* | \boldsymbol{\mu}^*_{-\boldsymbol{i}}\right) \leq 0. \end{aligned} \tag{13}$$

which is contradictory to $\frac{\partial U_j}{\partial \mu_j}\left(\mu_j^* | \boldsymbol{\mu}^*_{-\boldsymbol{j}}\right) = 0$. Therefore, the corollary holds. ■

Based on Corollary 1, we rank all CUs with their caching cost in ascending order, i.e., $\lambda_1 \leqslant \lambda_2 \leqslant \cdots \leqslant \lambda_M$, then we have $\mu_1 \geqslant \mu_2 \geqslant \cdots \geqslant \mu_M$.

We assume that the optimal caching policy for the first S CUs is non-zero, i.e., $\mu_S > 0$, $\mu_{S+1} = 0$. Denote $\mathcal{M}_S$ as the effective CU set with S being the cardinality of $\mathcal{M}_S$. Summing up with (8) for $i = 1, 2, ..., S$, we have

$$\frac{R(S-1)}{\sum_{j=1}^{S} \mu_j} - \sum_{i=1}^{S} \lambda_i = 0. \tag{14}$$

Then, we can obtain

$$\sum_{j=1}^{S} \mu_j = \frac{R\,(S-1)}{\sum_{i=1}^{S} \lambda_i}. \tag{15}$$

Substituting (15) into (8), we have

$$\mu_i = \frac{R\,(S-1)}{\sum_{j=1}^{S} \lambda_j}\left(1 - \frac{(S-1)\,\lambda_i}{\sum_{j=1}^{S} \lambda_j}\right). \tag{16}$$

Algorithm 1: Nash equilibrium calculation algorithm for CUs.

Reorder CUs in ascending order of λ_i, i.e., $\lambda_1 \leqslant \lambda_2 \leqslant \cdots \leqslant \lambda_M$.
$\mathcal{M}_S = \{m_1, m_2\}$, $S = 2$
while $S < M$ and $\lambda_{S+1} < \frac{\sum_{i=1}^{S} \lambda_i}{S-1}$ **do**
  $\mathcal{M}_S \leftarrow \mathcal{M}_S \cup \{m_{S+1}\}$;
  $S \leftarrow S + 1$;
end while
for $i = 1$; $i \leqslant M$; $i++$ **do**
  if $m_i \in \mathcal{M}_S$ **then**
    $\mu_i^* = \frac{R(S-1)}{\sum_{j=1}^{S} \lambda_j} \left(1 - \frac{(S-1)\lambda_i}{\sum_{j=1}^{S} \lambda_j}\right)$;
  else
    $\mu_i^* = 0$;
  end if
end for
return $\boldsymbol{\mu}^* = \{\mu_1{}^*, \mu_2{}^*, \cdots, \mu_M{}^*\}$.

As $\mu_S > 0$, and $S \geqslant 2$, we have $1 - \frac{(S-1)\lambda_S}{\sum_{j=1}^{S} \lambda_j} > 0$, i.e., $\lambda_S < \frac{\sum_{j=1}^{S-1} \lambda_j}{S-2}$.

Based on the above analysis, we design an algorithm to find the Nash equilibrium point of the CU's sub-problem, which is given in Algorithm 1.

Next we prove that the strategy proposed by Algorithm 1 is a Nash equilibrium point of CU's sub-problem, and then prove the uniqueness of this equilibrium point.

Theorem 2. *The strategy obtained with Algorithm 1 is a Nash equilibrium point of CU's sub-problem.*

Proof. For the CUs in $\mathcal{M}_S$, their optimal strategy is obtained by solving the first-order derivative of $U_i(\cdot)$ to zero, and this result is the best strategy. In order to prove the theorem, we only need to prove that the optimal policy of CUs not in $\mathcal{M}_S$ is 0, i.e., $\mu_j^* = 0, \forall m_j \in \mathcal{M} \backslash \mathcal{M}_S$. Base on Algorithm 1, we have

$$\lambda_j \geqslant \frac{\sum_{m_i \in \mathcal{M}_S} \lambda_i}{|\mathcal{M}_S| - 1}, \forall m_j \in \mathcal{M} \backslash \mathcal{M}_S. \tag{17}$$

From (15), we can obtain $\sum_{m_i \in \mathcal{M}_S} \lambda_i = R\left(|\mathcal{M}_S| - 1\right) / \sum_{m_i \in \mathcal{M}_S} \mu_i$, substituting it into (17), we have

$$\lambda_j \geqslant \frac{R}{\sum_{m_i \in \mathcal{M}_S} \mu_i}, \forall m_j \in \mathcal{M} \backslash \mathcal{M}_S. \tag{18}$$

Since $m_j \notin \mathcal{M}_S$, we have $\sum_{m_i \in \mathcal{M}_S} \mu_i = \sum_{m_i \in \mathcal{M} \backslash \{m_j\}} \mu_i$. Thus we can obtain

$$\lambda_j \geqslant \frac{R}{\sum_{m_i \in \mathcal{M} \backslash \{m_j\}} \mu_i}, \forall m_j \in \mathcal{M} \backslash \mathcal{M}_S. \tag{19}$$

According to (11), we know that the strategy for any CU $m_j \in \mathcal{M} \backslash \mathcal{M}_S$ is 0. This proves the theorem. ■

Theorem 3. *CU's sub-problem has a unique Nash equilibrium point.*

Proof. For the CUs in $\mathcal{M} \backslash \mathcal{M}_S$, their optimal strategy is $\mu_i = 0$ in a Nash equilibrium. Therefore, the all CUs' sub-problem can be seen as a game among the CUs in $\mathcal{M}_S$, we only need to prove that a unique Nash equilibrium point exists in the sub-game of CUs in $\mathcal{M}_S$.

For the CUs in $\mathcal{M}_S$, $\mu_i > 0$. The utility function of CUs is a concave function for any CU in $\mathcal{M}_S$, according to (9) , their optimal strategy is obtained by solving the first-order derivative of $U_i(\cdot)$ to zero. As shown in (14)-(16), we can obtain the unique solution by solving the first-order derivative of $U_i(\cdot)$ to zero for each edge device in $\mathcal{M}_S$. Thus we can know that a unique Nash equilibrium point exists in the sub-game of CUs in $\mathcal{M}_S$, therefore Theorem 3 holds. ■

B. Analysis of the CSP's sub-game

Based on above analysis, there is always a Nash equilibrium point for CUs for any value of the reward given by the CSP. When the reward R is given, the CSP will have a corresponding utility, so we can set an optimal R to maximize its utility.

According to Problem 2, we can know that only when the total caching space is larger than Q, i.e., $\sum_{i=1}^{M} \mu_i \geqslant Q$, can the CUs obtain rewards. From (15), we have $\sum_{i=1}^{S} \mu_i = R(S-1) \sum_{i=1}^{S} / \lambda_i$. Meanwhile, we know, for $m_i \in \mathcal{M} \backslash \mathcal{M}_S$, the best strategy is 0, i.e., $\mu_i = 0$, thus we can obtain $\sum_{i=1}^{M} \mu_i = \sum_{i=1}^{S} \mu_i$. Therefore, $\frac{R(S-1)}{\sum_{i=1}^{S} \lambda_i} \geqslant Q$, that is $R \geqslant \frac{Q \cdot \sum_{i=1}^{S} \lambda_i}{(S-1)}$.

Then Problem 2 can be express as

$$\max_{R \geq \varphi} \ \alpha \sum\nolimits_{i=1}^{S} \log_2\left(1 + \beta \mu_i\right) - R. \tag{20}$$

where $\varphi = \frac{Q \cdot \sum_{i=1}^{S} \lambda_i}{(S-1)}$, $S \geqslant 2$.

Substituting (16) into (20), we can obtain

$$\begin{aligned} U_{CSP} &= \alpha \sum_{i=1}^{S} \log_2\left(1 + \beta \mu_i^*\right) - R \\ &= \alpha \sum_{i=1}^{S} \log_2\left(1 + \beta \chi_i R\right) - R. \end{aligned} \tag{21}$$

where $R \geqslant \frac{Q \cdot \sum_{i=1}^{S} \lambda_i}{(S-1)}$, $S \geqslant 2$, and

$$\chi_i = \frac{(S-1)}{\sum_{i=1}^{S} \lambda_i} \left(1 - \frac{(S-1)\lambda_i}{\sum_{i=1}^{S} \lambda_i}\right). \tag{22}$$

From (16) and algorithm 1, we can know $\chi_i > 0$.

Theorem 4. *The proposed Stackelberg game has a unique Stackelberg equilibrium* $(\boldsymbol{\mu}^*, R^*)$ *, where* $\boldsymbol{\mu}^*$ *and* R^* *are optimal strategies for CUs and CSP.*

Proof. The first and second derivatives of the utility function $U_{CSP}(\cdot)$ are given by

$$\frac{\partial U_{CSP}}{\partial R} = \sum_{i=1}^{S} \frac{\alpha \beta \chi_i}{(1 + \beta \chi_i R) \ln 2} - 1, \tag{23}$$

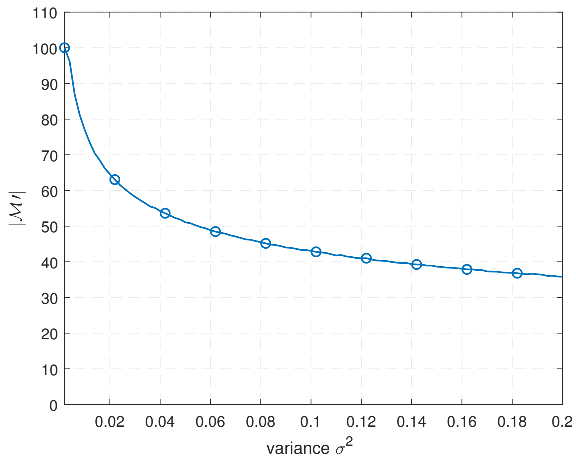

Fig. 2. Impact of σ^2 on $|\mathcal{M}|$.

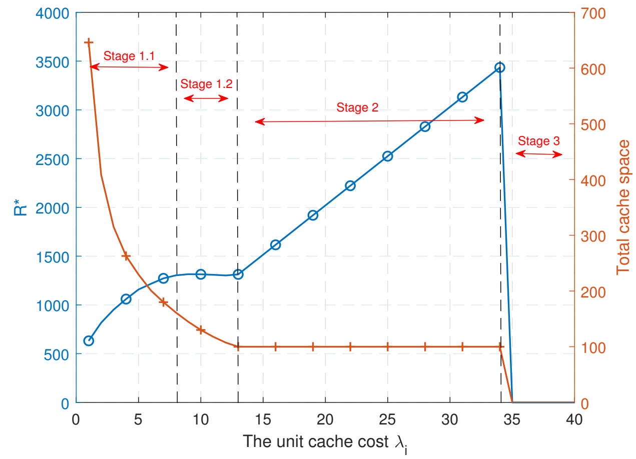

Fig. 3. Impact of λ_i on the caching strategies.

$$\frac{\partial^2 U_{CSP}}{\partial R^2} = -\sum_{i=1}^{S} \frac{\alpha(\beta\chi_i)^2}{(1+\beta\chi_i R)^2 \ln 2} < 0. \tag{24}$$

Therefore, the utility function $U_{CSP}(\cdot)$ is concave with R, which means that there exists a unique R^* to maximize the utility. Combined with Theorem 3, Theorem 4 can be proved. ∎

When $R = 0$, we have $U_{CSP} = 0$, and the utility function is a concave function, which first increases and then decreases. Then, the optimal reward can be given by

$$R^* = \begin{cases} 0, & \text{if } \varphi \geqslant \xi, \\ \max(\varphi, R'), & \text{otherwise.} \end{cases} \tag{25}$$

where $\varphi = \frac{Q \cdot \sum_{i=1}^{S} \lambda_i}{(S-1)}$, ξ and R' are the solutions of the utility function and the first derivative, respectively, i.e., $U_{CSP}(\xi) = 0$, $\xi \neq 0$, $\partial U_{CSP}/\partial R(R') = 0$.

IV. Simulation Results and Discussions

In this section, we conduct extensive simulations to evaluate the performance of the proposed incentive mechanism. We assume that there are $M = 100$ users with caching capability around the CSP who may share their cache space. The caching cost of different CUs are uniformly distributed with a mean of 10 and variance σ^2. For the CSP, the minimum caching size Q is set to 100, α and β are set to be 10, unless otherwise specified.

Each CU estimates their own revenue by analyzing other CUs' caching space , their own caching cost and the caching requirement Q. The impact of unit caching cost on the number of effective CUs is evaluated in Fig. 2. We can observe from Fig. 2 that with the increase of the variance of caching cost, the total number of CUs participating in caching task gradually decreases. This is straightforward, since as the cost variance increases, more CUs cannot meet the conditions. When the cost variance is 0.054, nearly half of the CUs no longer shares their caches.

The effect of the unit caching cost λ_i on the caching strategies is show in Fig. 3. We can observe that when the total cache space provided by the CUs is larger than Q, the caching space contributed by the CUs decreases with λ_i. In stage1.1, the utility of the CSP improves as R^*. This is because when the caching cost increases, the CSP needs to improve R^* to encourage more CUs to share caching space. Due to the increase of λ_i, the caching space shared by CUs decreases greatly. Therefore, even R^* increases slightly, the average utility of the CU is also decreasing. In stage 1.2, with the increase of λ_i, R^* remains unchanged, the total caching space decreases, and the utilities of the CSP and CUs decrease. This is because the number of users has reached the maximum, so the CSP will no longer increase its reward R^*. For the CSP, the increase of λ_i leads to the decrease of caching space and the utility of the CSP; For the CU, due to the increase of λ_i, the total caching space and the average utility decrease. For stage 2 that the total caching space provided by CUs is equal to Q, we can observe that R^* increases continuously with the caching cost. The utility of the CSP decreases and the average utility of the CUs increases. This is because the CSP needs to increase the reward R^* to remain the caching requirement Q. Whereas, in stage 3, the caching cost is too high that the minimum caching constraint cannot be satisfied, thus the reward reduces to zero, so the utilities of the CSP and CUs are.

Similarly, the impact of caching requirement Q on the strategies and the utilities are evaluated in Figs. 5 and 6, respectively. In stage 1, when the target value of Q is relatively small, the CSP can complete tasks easily, the reward R^* remains unchanged. In stage 2, the CSP will continuously increase R^* to encourage more CUs to complete the cache target Q. The total caching space keep increasing, resulting in a decrease of the CSP's utility and increase of CU's utility. When the caching requirement Q is relatively too large in stage 3, the utility of the CSP become zero, and thus the reward reduces to zero, and all the CUs will not participate in the process.

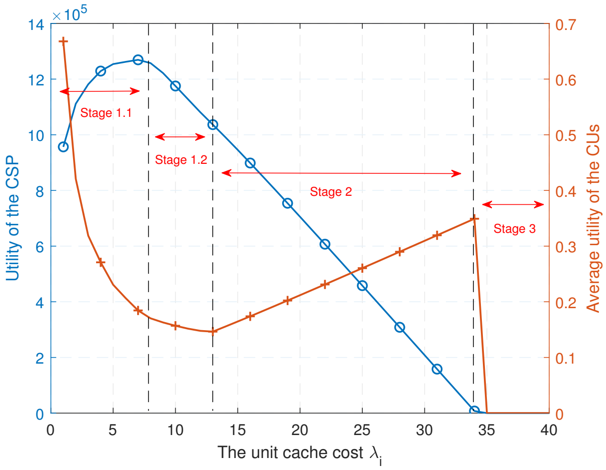

Fig. 4. Impact of λ_i on the utilities.

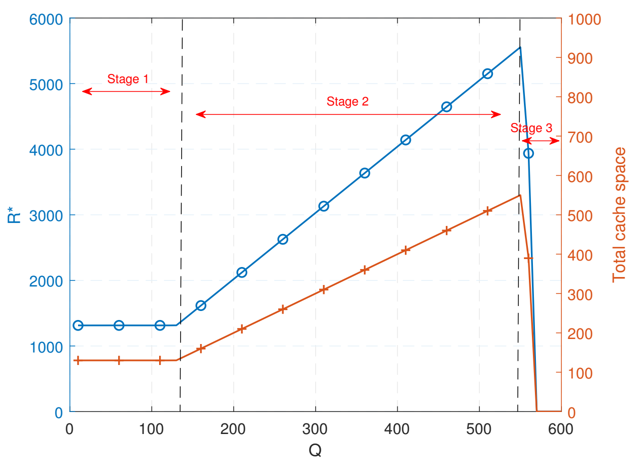

Fig. 5. Impact of the caching target Q on the strategies.

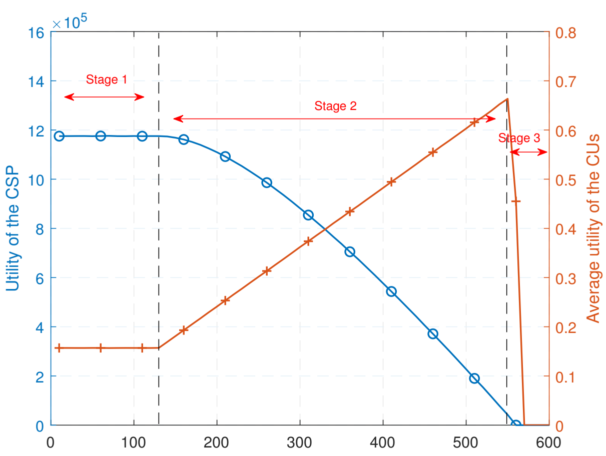

Fig. 6. Impact of the caching target Q on the utilities.

V. Conclusions

In this paper, we have developed a blockchain enabled mobile user caching framework that motivates nearby users to contribute their caching space to reduce the cost of traditional caching scheme. The participation and contributions that each caching users made is recorded on the blockchain, with guaranteed security and privacy. On this basis, we have also developed the game theoretical framework for the incentive mechanism design. Simulations have shown the effectiveness of the proposed scheme. In the future, we will blockchain consensus and smart contract enabled caching transactions.

Acknowledgment

This work was supported in part by the National Key R&D Program of China under Grant 2020YFB1005900, the National Natural Science Foundation of China No. 62001220, the Natural Science Foundation of Jiangsu Province BK20200440, the Fundamental Research Funds for the Central Universities No.1004-YAH20016, No.NT2020009.

References

[1] W. Sun, S. Li, and Y. Zhang, "Edge caching in blockchain empowered 6G," *China Commun.*, vol. 18, no. 1, pp. 1–17, 2021.

[2] W. Wu, N. Zhang, N. Cheng, Y. Tang, K. Aldubaikhy, and X. Shen, "Beef up mmwave dense cellular networks with D2D-assisted cooperative edge caching," *IEEE Trans. Veh. Tech.*, vol. 68, no. 4, pp. 3890–3904, 2019.

[3] S. Khanal, K. Thar, and E.-N. Huh, "Dcol: Distributed collaborative learning for proactive content caching at edge networks," *IEEE Access*, vol. 9, pp. 73 495–73 505, 2021.

[4] G. Ma, Z. Wang, M. Zhang, J. Ye, M. Chen, and W. Zhu, "Understanding performance of edge content caching for mobile video streaming," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 5, pp. 1076–1089, 2017.

[5] S. Zhang, P. He, K. Suto, P. Yang, L. Zhao, and X. Shen, "Cooperative edge caching in user-centric clustered mobile networks," *IEEE Trans. Mobile Comput.*, vol. 17, no. 8, pp. 1791–1805, 2017.

[6] W. Wang, N. Cheng, Y. Liu, H. Zhou, X. Lin, and X. Shen, "Content delivery analysis in cellular networks with aerial caching and mmwave backhaul," *IEEE Trans. Veh. Tech.*, vol. 70, no. 5, pp. 4809–4822, May 2021.

[7] J. Liu, S. Guo, Y. Shi, L. Feng, and C. Wang, "Decentralized caching framework toward edge network based on blockchain," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 9158–9174, 2020.

[8] W. Sun, J. Liu, Y. Yue, and Y. Jiang, "Social-aware incentive mechanisms for D2D resource sharing in IIoT," *IEEE Trans. Ind. Informat.*, vol. 16, no. 8, pp. 5517–5526, 2019.

[9] J. Zou, C. Li, C. Zhai, H. Xiong, and E. Steinbach, "Joint pricing and cache placement for video caching: A game theoretic approach," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 7, pp. 1566–1583, 2019.

[10] X. Cao, J. Zhang, and H. V. Poor, "An optimal auction mechanism for mobile edge caching," in *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2018, pp. 388–399.

[11] R. Zhang, F. R. Yu, J. Liu, R. Xie, and T. Huang, "Blockchain-incentivized D2D and mobile edge caching: A deep reinforcement learning approach," *IEEE Netw.*, vol. 34, no. 4, pp. 150–157, 2020.

[12] Q. Xu, Z. Su, and Q. Yang, "Blockchain-based trustworthy edge caching scheme for mobile cyber-physical system," *IEEE Internet Things J.*, vol. 7, no. 2, pp. 1098–1110, 2019.