Cooperative Caching in Satellite-Terrestrial Integrated Networks: A Region Features Aware Approach

Jin Tang, Jian Li, *Senior Member, IEEE*, Xianhao Chen, *Member, IEEE*, Kaiping Xue, *Senior Member, IEEE*, Lan Zhang, *Member, IEEE*, Qibin Sun, *Fellow, IEEE*, and Jun Lu

***Abstract*—As an essential part of the next-generation communication system, the satellite-terrestrial integrated network (STIN) can provide Internet access and content delivery services for terrestrial users in remote areas. However, more and more content-related traffic requested by users poses challenges to the transmission capability of STINs. Thus, how to provide high-quality content delivery service has become an urgent problem in STINs. In this paper, we leverage the caching capabilities of low earth orbit (LEO) satellites and devise a cooperative caching scheme to achieve low latency and bandwidth-efficient content delivery service. First, considering users' diverse request preferences in different geographical locations, we propose a region features prediction model based on the ridge regression method to update users' preference information. Second, according to the similarity of predicted region features, terrestrial regions covered by LEO satellites are further divided into multiple cooperative areas. Third, we propose a cooperative caching algorithm based on game theory to achieve distributed caching decisions in every single cooperative area. By implementing this cooperative caching scheme, terrestrial users can acquire high-quality content delivery service through STINs. Extensive evaluations show that the proposed caching scheme can significantly reduce content delivery delay while saving valuable bandwidth resources compared with the existing ones.**

***Index Terms*—Cooperative caching, content delivery service, region features prediction, satellite-terrestrial integrated network.**

Manuscript received 9 August 2023; revised 20 December 2023; accepted 13 February 2024. Date of publication 23 February 2024; date of current version 16 July 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62201540 and in part by the Open Research Fund of State Key Laboratory of Integrated Services Networks under Grant ISN24-11. The review of this article was coordinated by Dr. Zehui Xiong. *(Corresponding author: Jian Li.)*

Jin Tang is with the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei 230027, China, and also with the State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, China.

Jian Li is with the School of Cyber Science and Technology, University of Science and Technology of China, Hefei 230027, China, and also with the State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, China (e-mail: lijian9@ustc.edu.cn).

Xianhao Chen is with the Department of Electrical and Electronic Engineering, University of Hong Kong, Hong Kong.

Kaiping Xue and Qibin Sun are with the School of Cyber Science and Technology, University of Science and Technology of China, Hefei 230027, China.

Lan Zhang is with the Department of Electrical and Computer Engineering, Clemson University, Clemson, SC 29634 USA.

Jun Lu is with the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei 230027, China.

Digital Object Identifier 10.1109/TVT.2024.3369106

I. Introduction

In recent years, the satellite-terrestrial integrated network (STIN) has emerged as a crucial development direction for next-generation cellular systems, attracting significant attention from both academia and industry [1], [2], [3]. Considering its unique advantages of ubiquitous coverage, seamless access, and high-capacity communication, the STIN becomes an attractive complement to terrestrial cellular systems and is expected to provide reliable Internet access for terrestrial users located in remote areas, e.g., oceans and deserts, where the traditional communication infrastructure deployment is impractical [4], [5], [6].

According to the latest report, by 2028, global mobile traffic is expected to increase about three times compared to today [7]. The rapidly increasing content-related traffic poses a non-negligible challenge for STINs. Unfortunately, the increase of transmission capacity of STINs is severely hindered by the resource-limited environment of satellites, e.g., restricted spectrum and limited communication energy. In this case, the STIN is struggling to handle a large amount of content-related traffic requested by global users and suffering tremendous pressure from guaranteeing the quality of service [8]. As a result, how to provide high-quality content delivery service in STINs has become a critical but unsolved problem.

To alleviate the tremendous pressure from content-related traffic, existing studies have considered cache-enabled STINs as a promising solution. By utilizing on-board storage capacity and leveraging dedicated caching strategies, cache-enabled STINs have been proven that it can significantly save valuable on-board resources of satellites and reduce content delivery delay, especially in resource-limited environment [9], [10], [11]. Based on the dense-deployed small low earth orbit (LEO) satellites and heterogeneous satellite constellations, some related studies have further designed efficient content delivery schemes for STINs, including transmission optimization [12], [13], multi-layer caching design [14], [15], etc., and achieved significant performance improvement in terms of bandwidth saving and delay reduction.

Despite these proposed caching designs and remarkable theoretical results, in practice, two critical issues are not well addressed yet for cache implementation in STINs, i.e., pre-known content popularity and centralized control scheme. On the one

hand, most existing studies consider pre-known content popularity to characterize terrestrial users' request preferences [10], [11], [13], [14], [15]. Since users' request preferences vary in both spatial (dynamic coverage of geographical regions for satellites) and temporal (dynamic request preference for different time intervals) dimensions, a constant content popularity model can lead to a mismatch between theoretical and practical results. Therefore, an effective prediction method, which can characterize the spatial and temporal features based on real-time users' requests, is required to capture precise content popularity in STINs. On the other hand, most caching decisions in existing studies are updated through a centralized approach, e.g., medium earth orbit (MEO), geostationary earth orbit (GEO) satellite, or ground station is adopted as the centralized controller. Due to the highly dynamic connectivity and limited coverage of a single controller, the centralized approach can result in an unreliable update process and extra communication overhead. Hence, a distributed approach is preferred to update caching decisions in STINs.

To tackle the aforementioned two critical issues and provide more efficient content delivery services for the growing terrestrial users, in this paper, we design a cooperative caching scheme based on the region features prediction model for the cache-enabled STIN. At first, we characterize the features of each terrestrial region based on the users' real-time requests and propose a region prediction model that utilizes the ridge regression method to update users' preference information. Instead of adopting a pre-known content popularity model, the prediction results can help us to accurately deduce the popularity of different contents in different regions. After that, according to the similarity of predicted region features, terrestrial regions covered by LEO satellites are divided into multiple cooperative areas, and a distributed cooperative caching scheme is further devised. The proposed scheme updates caching decisions via a distributed approach and shares cached contents between satellites in the same cooperative area, which can fully leverage on-board storage resources to provide high-quality content delivery services. Finally, based on a real-world dataset, we conduct a series of evaluations compared with the existing caching schemes. Evaluation results demonstrate that the proposed cooperative caching scheme can significantly reduce the content delivery delay while saving valuable bandwidth resources.

The main contributions of this paper are organized as follows:

- We consider a unique characteristic in STINs, i.e., diverse users' request preferences across different geographical locations, and propose a prediction method to update region features based on real-time users' requests. According to users' requests for contents, the features of terrestrial regions can be characterized to support efficient and adaptive caching decisions.
- We propose a cooperative caching scheme based on predicted region features to further improve caching performance, and design a distributed caching algorithm for STINs. By dividing terrestrial regions with similar region features into the same cooperative area, on-board storage resources can be fully leveraged with cooperative caching decisions.
- We conduct a series of evaluations based on real-world datasets. Compared to the existing caching schemes, results show that the proposed cooperative caching scheme is significantly superior to traditional schemes in terms of content delivery delay and bandwidth consumption in STINs.

The rest of the paper is organized as follows. The related works are discussed in Section II. The system model is introduced in Section III, and the region features prediction model is discussed in Section IV. After that, the division of cooperative area and problem formulation of cooperative caching are proposed in Section V. Then the distributed cooperative caching algorithm based on the non-cooperative game is claimed and analyzed in Section VI. Finally, the performance evaluation and analysis are conducted in Section VII, and conclusions are drawn in Section VIII.

The following notations are used throughout this paper. Boldface letters indicate column vectors and matrices, respectively. $(\cdot)^{\top}$ and $(\cdot)^{-1}$ denote the transpose and inverse operations, respectively. diag$\{\mathbf{a}\}$ indicates the diagonal matrix whose diagonals are the elements of $\mathbf{a}$. Besides, we adopt $|\cdot|$ to represent the absolute value operation and $||\cdot||$ to represent the second-order norm of a vector or matrix.

II. Related Works

Existing studies mainly focus on improving the quality of content delivery services in STINs from two perspectives: network architecture design and edge caching technology application. In this section, we will introduce related works from the above two aspects.

In order to provide terrestrial users with low-latency content delivery services, existing studies [16], [17] firstly combined information-centric networking (ICN) architecture with the STIN and designed a unique architecture based on their characteristics. A large number of experiments have also proved the effectiveness of this architecture application in the STIN. Based on this structure, Li et al. [18] combined ICN architecture and software-defined networking (SDN) to design an efficient content transmission scheme for the STIN, where the MEO/GEO satellites act as controllers to globally control the caching strategies and content delivery process of LEO satellites. In addition, Yang et al. [8] focused on the content delivery process in STIN and proposed a chunk-level data retrieval scheme that can be combined with network coding methods. Under the ICN architecture with in-network caching capabilities, the proposed strategy effectively reduces the number of hops and delays in content delivery. Tang et al. [19] predicted the probability that the content requested by the user is cached on cache-enabled satellites. On this basis, the author designed a content-aware routing to obtain content from cache-enabled satellites in an opportunistic manner, thereby reducing content delivery delay and redundant traffic in STIN. Besides, Xu et al. [20] proposed a hybrid caching strategy for ICN-combined LEO satellite networks, considering node classification and popular content awareness, and dynamically divided satellite nodes into core nodes and edge nodes to improve content distribution efficiency

and overcome satellite node mobility and dynamic topology brought challenges.

On the other hand, except for designing an efficient content delivery scheme at the network level, other research works combined edge caching technology [9], [12], [15], [21] with the STIN to provide users with low-latency content delivery services. By caching popular contents on edge nodes, users can retrieve the required contents from cache-enabled nodes with fewer hops and lower delay rather than retrieve them from remote servers. In order to improve network utility in STIN, Han et al. [9] investigated the joint design of cache placement and cooperative multicast beamforming in STIN to provide content-centric services for mobile users. Specifically, users requesting the same content are scheduled into a multicast group and served by cache-enabled base stations (BS) and LEO satellites through cooperative beamforming. The author of [12] designed a novel in-network caching and file distribution method. The proposed method considers the uneven distribution of users in STIN and divides STIN into a series of blocks of different sizes. Based on the time-varying network model, the cached content update mechanism in STIN is derived, and the proposed solution achieves stable and sustainable user experience quality. By introducing cache-enabled LEO satellite networks as part of the radio access network (RAN), the author [15] proposed an integrated satellite/terrestrial cooperative transmission scheme from the perspective of achieving energy-saving RAN. Through the caching and broadcasting capabilities of satellites, the traffic of base stations is offloaded, and energy-efficient content delivery services are provided to terrestrial users. Considering that content delivery is interconnected and affected by network dynamics, He et al. [21] proposed a hierarchical deep Q-learning algorithm to learn cache placement and content delivery strategies in STIN. The proposed scheme reduces the latency of content delivery by leveraging two independent deep neural networks to learn cache placement and content delivery policies with small action spaces and low time complexity.

To fully leverage the storage resources of edge nodes and satellites, cooperative caching [22], [23], [24] can usually be deployed to cache more contents that users may be interested in. The cooperative approach can further improve the cache hit rate and reduce the content delivery delay. Ma et al. [25] considered the cooperation between edge nodes, investigated the problem between caching, computing, and communication, and abstracted the problem into a mixed integer nonlinear programming problem. Zhu et al. [14] proposed a multi-level collaborative caching architecture in STIN, in which base stations, satellites, and gateways cooperate to provide content delivery services to terrestrial users. The author formulated the problem of where to cache contents to minimize user content retrieval delay and solved it based on the proposed iterative algorithm, thereby providing low-latency content delivery services to terrestrial users.

In the STIN, due to the constellation settings, satellites provide services for users in different geographical locations, and the contents that users are interested in significantly vary across their geographical locations. Existing works do not fully consider users' preferences in different geographical locations and lack a scheme that can reflect the popularity of contents at different times and regions. Besides, the caching decision process usually requires MEO/GEO satellites, which may introduce additional communication overhead in a dynamic network environment.

In this paper, we consider the region features of different geographical locations, which can reflect users' preferences for contents in a specific region and infer the contents' popularity in that region. Furthermore, we also adopt a distributed algorithm for caching decisions, avoiding the additional overhead of using centralized schemes for caching decisions in dynamic networks.

III. System Model

A. Network Model

Consider an integrated satellite-terrestrial network composed of LEO satellites, terrestrial users, and cloud servers connected to the ground station. As shown in Fig. 1, each satellite has at most four collected satellites, including two adjacent satellites in the same orbital plane and two closest satellites in the adjacent orbital plane [26]. Besides, the satellite network consists of J cache-enabled satellites, which denoted as $\mathcal{S} = \{S_1, \dots S_j, \dots S_J\}$. Considering the mobility of LEO satellites relative to the ground, each satellite periodically serves a region to provide content delivery service for users in its covered region. Besides, some LEO satellites can connect to the ground station, which can retrieve any contents from cloud servers through terrestrial networks.

Users in the region covered by a satellite can directly send a request to the satellite that is serving them, called the serving satellite. If the serving satellite caches the requested content, the satellite will directly return the content to users, providing a low-latency content delivery service. If the content is not cached by the serving satellite, the serving satellite will forward the request to other satellites or cloud servers by forwarding it to the ground station first, which may bring a longer delay. For example, as shown in Fig. 1, a user in the region (i, j) sends a request for a certain content to the serving satellite S_2. Since satellite S_2 caches the requested content, it directly sends the content to the user. However, if the content requested by users in the region $(i, j-n)$ is not cached in the serving satellite, the serving satellite will forward the request to the ground station and then retrieve the content from the cloud server.

Denote the contents set as $\mathcal{M} = \{f_1, f_2, \dots f_m, \dots, f_M\}$. Without loss of generality, we assume that each content has same size B bytes, and each satellite has the same cache size c, i.e., it has $c \cdot B$ bytes of storage space to cache contents with quantity of c. Define a vector to describe the features of the requested content. For instance, the features of the content f_m are defined as $\mathbf{C}_m = \{C_{m,1}, \dots C_{m,d}, \dots C_{m,D}\}$, where $C_{m,d} \in \{0, 1\}$, represents the d-th feature of content f_m. Each content has D dimensions of features, where those features can be the content's classification, language, etc.

B. Region Features Model

For the sake of simplicity, assume that each region has one and only one satellite serving the region, and users in this region

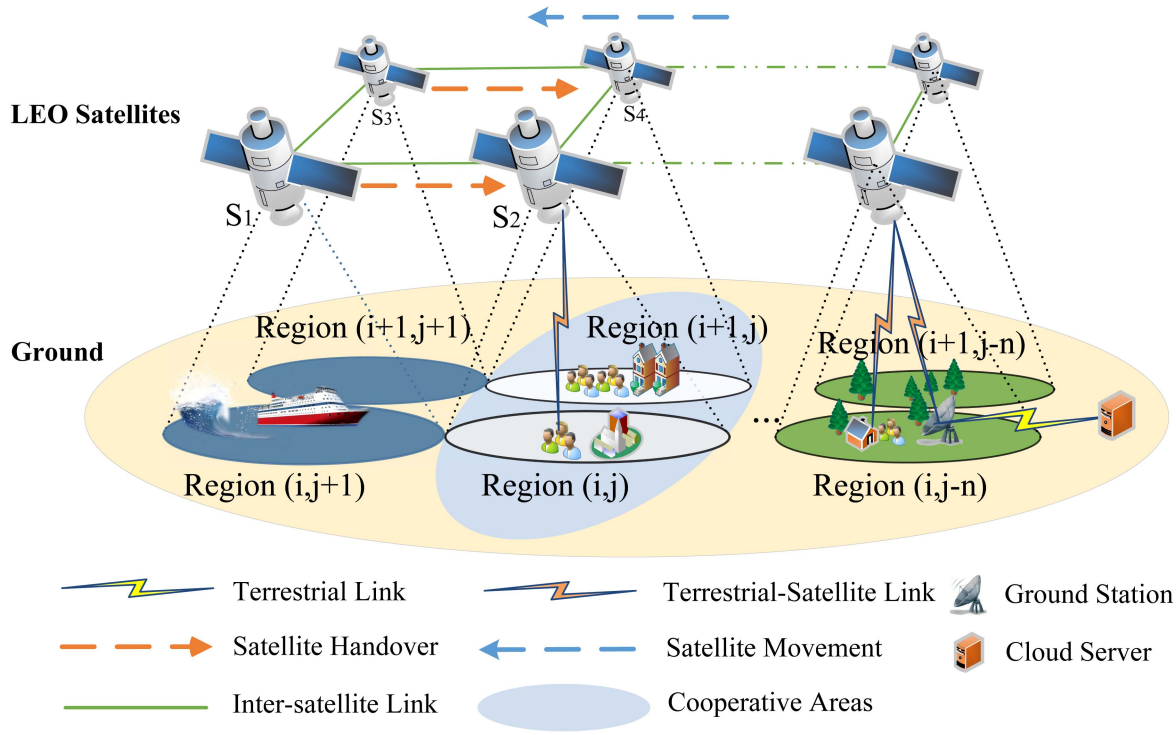


Fig. 1. System model: each region has one serving satellite which can provide content delivery service for terrestrial users.

can only directly access this satellite. Besides, the terrestrial regions are divided according to their latitude and longitude and the satellites' constellation. According to the previous definitions, there are J satellites in total, and each region has one and only one serving satellite. Therefore, we should divide the terrestrial regions covered by the satellite network into J regions. Each region is represented by a coordinate, denoted as $\mathcal{N} = \{(i,j) \mid i \times j \leq J, i > 0, j > 0\}$. Moreover, inspired by the idea of virtual nodes in the satellite network [27], the satellite serving region (i,j) should be located in the j-th logical position of the i-th orbital plane of the satellite constellation. When a satellite leaves the region due to constellation motion, the subsequent satellite in the same orbital plane as the satellite will serve the region.

Consider using users' preferences in a region for contents with different characteristics to describe the features of the region. Users' preferences in one region are represented by the characteristics of popular contents requested over time. Hence, users' preferences of region (i,j) or the features of region (i,j) can be expressed as $\mathbf{g}_{i,j,nT} = \{g_{i,j,nT,1}, \ldots g_{i,j,nT,d}, \ldots g_{i,j,nT,D}\}$, where $g_{i,j,nT,d} > 0$, represents the users' preference to the d-th feature of content during n-th period in region (i,j). Every T time, the popular contents of one region in the previous period should be used to describe the features of this region, which is also be updated as the predicted region features in the next period, denoted as $\mathbf{g}^*_{i,j,nT} = \mathbf{g}_{i,j,n(T-1)}$.

C. Caching Model

After predicting the features of one region, we next predict the popularity of different contents according to the region features. The popularity of content f_m in region (i,j) can be predicted as:

$$r^*_{m,i,j,nT} = \mathbf{g}^*_{i,j,nT}\mathbf{C}^\top_m = \sum_{d=1}^{D} g^*_{i,j,nT,d} C_{m,d}, \tag{1}$$

where $r^*_{m,i,j,nT}$ is the number of requests from users in region (i,j) for content f_m. The satellite in a certain region should calculate the popularity of different contents when caching content according to (1). Considering that satellite cache is limited, we should cache those contents that are more popular and can bring more benefits for us. Let $u_{i,j,m,nT}$ denote the benefits of caching content f_m on the satellite for serving region (i,j) during the n-th period, where $u_{i,j,m,nT} > 0$. The benefits of caching one certain content refer to the time saved because terrestrial users successfully retrieve the required content from the satellite rather than the remote cloud server. Hence, the higher the benefits of caching one content, the more it should be cached.

Considering the limited storage resources of satellites, we divide those regions with high similarity into one cooperative area, and users cannot only retrieve contents from the satellites serving their region but also from other satellites in the same cooperative area. For instance, in Fig. 1, region (i,j) and region $(i+1,j)$ are divided into one cooperative area because of their high similarity. So for users in the region (i,j), they cannot only retrieve the required content from satellite S_2 in service region (i,j) but also retrieve the required content from satellite S_4 in service region $(i+1,j)$. By enabling satellites in one cooperative area to use a cooperative caching method, they can cache more contents in which users may be interested and fully use limited storage resources.

When a satellite leaves a region due to orbital motion, subsequent satellites on the same orbital plane will arrive and continue to provide Internet access services and content delivery services to users in the region. This process is called inter-satellite handover. As shown in Fig. 1, when the satellite S_2 serving region (i,j) is about to leave the region (i,j) and enter the region $(i,j+1)$, the previous satellite serving this region, i.e., satellite S_1, should deliver its cache to S_2, and satellite S_2 also deliver its cache to the successor satellite [28].

In general, satellites have both inter-satellite links and satellite-ground links simultaneously. Therefore, the inter-satellite handover is independent of terrestrial users' access, which has no effect on content delivery service. Additionally, caches in adjacent regions often have similarities, so only a portion of the contents need to be delivered to subsequent satellites when inter-satellite handover happens.

IV. Prediction of Region Features Based on Ridge Regression

Considering that users' preferences for content vary in both spatial and temporal dimensions, we plan to design a distributed prediction method based on real-time user requests, which takes geographical location into consideration. In this section, we first predict the region features based on historical requests in different geographical locations. After that, we propose the ridge regression based prediction model to improve the accuracy of region features prediction. Based on the predicted region features, the users' preferences for contents in a specific region and the contents' popularity in that region can be deduced.

A. Prediction of Region Features

Edge caching technology allows contents to be cached on an edge node close to users according to a specific caching strategy so that users can obtain them with a shorter path and lower delay. In the STIN, popular content can also be cached on the LEO satellite close to the user. However, users' preferences for content are constantly changing, and popular content will also change accordingly. Therefore, it is necessary to predict the popularity of the content and adjust the caching strategy according to the predicted content popularity. Besides, in the STIN, the regions covered and served by satellites are different, and users in each region have different preferences for popular content. Therefore, when predicting the popularity of content in different regions, the characteristics of these regions should be considered.

The distribution of requests for content is usually subject to the Zipf distribution [29], a small number of popular content has a high popularity and accounts for most requests. Therefore, we can use the features of the popular content in a region to describe the users' preferences for content in that region and take them as the features of the region [30]. We use the first L popular content of a region in a period to describe the region features, of which the first L popular contents are denoted as $\mathcal{M}^L_{i,j,(n-1)T} = \{M_1, \ldots M_l, \ldots, M_L\}$, where $\mathcal{M}^L_{i,j,(n-1)T} \subseteq \mathcal{M}$ and M_l represents the content which ranks No.l in the $(n-1)$-th period in the region (i,j). Moreover, the real popularity of content M_l is denoted as $r_{M_l,i,j,(n-1)T}$, which is the requested times of the content f_m in the period $(n-1)T$. Hence, the features of region (i,j) in period $(n-1)T$ can be obtained by minimizing the following mean squared error:

$$\begin{aligned}\mathbf{g}_{i,j,n(T-1)} &= \arg\min_{\mathbf{y}} \sum_{l=1}^{L} \left(\sum_{d=1}^{D} C_{M_l,d} y_d - r_{M_l,i,j,n(T-1)} \right)^2 \\ &= \arg\min_{\mathbf{y}} \left\| \mathbf{C}^L \mathbf{y} - \mathbf{R}^L \right\|^2, \end{aligned} \tag{2}$$

where content features matrix

$$\mathbf{C}^L = \begin{bmatrix} \mathbf{C_{M_1}} \\ \vdots \\ \mathbf{C}_{M_l} \\ \vdots \\ \mathbf{C}_{M_L} \end{bmatrix} = \begin{bmatrix} C_{M_{1,1}}, \ldots, C_{M_{1,d,}}, \ldots, C_{M_{1,D}} \\ \vdots \\ C_{M_{l,1}}, \ldots, C_{M_{l,d,}}, \ldots, C_{M_{l,D}} \\ \vdots \\ C_{M_{L,1}}, \ldots, C_{M_{L,d,}}, \ldots, C_{M_{L,D}} \end{bmatrix}. \tag{3}$$

The vector $\mathbf{R}^L = (r_{M_1,i,j,n(T-1)}, \ldots, r_{M_l,i,j,n(T-1)}, \cdots r_{M_L,i.j,n(T-1)})^\top$ denotes the real popularity of first popular L contents and vector $\mathbf{y} = (y_1, \ldots, y_d, \ldots y_D)^\top$ represents the desired coefficients for content popularity prediction.

We use the region features of one region in the last period $(n-1)T$ as prediction for the next period nT. Hence, the predicted region features of period nT can be represented as

$$\mathbf{g}^*_{i,j,nT} = \mathbf{g}_{i,j,n(T-1)}, \tag{4}$$

and it can be further deduced as

$$\mathbf{g}^*_{i,j,nT} = \left(\mathbf{C}^{L^\top} \mathbf{C}^L \right)^{-1} \mathbf{C}^{L^\top} \mathbf{R}^L. \tag{5}$$

B. Popularity Weight and Ridge Regression

To describe the region features, we solve it by minimizing the prediction error between the real popularity and the predicted popularity of the first L popular content according to (2). The prediction error of a certain content M_l in the first L popular content is recorded as:

$$e_{M_l} = \sum_{d=1}^{D} C_{M_l,d} y_d - r_{M_l,i,j,n(T-1)}, \tag{6}$$

and the total prediction error is $e = \sum_{l=1}^{L} e_{M_l}$.

Since the final total error, e, is the sum of the prediction errors of the first L content, the error of each content has an impact on the estimation of region features, and no matter how popular the content is, its effect on the prediction of region features is equal. However, even among the first L popular content, the popularity of different content is different. The prediction error caused by content M_1, which ranks NO.1 in popularity, and content M_l, which ranks NO.L in popularity, is different for the estimation of region features. Therefore, when characterizing region features, it is necessary to consider the popularity weight of these contents $\mathcal{M}^L_{i,j,nT}$.

Consider using popular content features in one region to describe the region features, hence, the more popular, the more

it can describe the region features. So we modify the prediction error in (6) of each content to:

$$e_{M_l} = \widehat{r}_{M_l,i,j,n(T-1)} \left(\sum_{d=1}^{D} C_{M_l,d} y_d - r_{M_l,i,j,n(T-1)} \right), \quad (7)$$

where $\widehat{r}_{M_l,i,j,n(T-1)} = r_{M_l,i,j,n(T-1)} / \sum_{l=1}^{L} r_{M_l,i,j,n(T-1)}$ represents the popularity proportion of content M_l in the first L popular contents $\mathcal{M}_{i,j,nT}^{L}$. Then the features of region (i,j) in (2) can be further denoted as:

$$\begin{aligned} \mathbf{g}_{i,j,n(T-1)} &= \arg\min_{\mathbf{y}} \sum_{l=1}^{L} e_{M_l}^2 \\ &= \arg\min_{\mathbf{y}} \left\| \widehat{\mathbf{R}}^L (\mathbf{C}^L \mathbf{y} - \mathbf{R}^L) \right\|^2, \end{aligned} \quad (8)$$

where the popularity weight matrix $\widehat{\mathbf{R}}^L$ is a diagonal matrix,

$$\widehat{\mathbf{R}}^L = \text{diag}\left\{ \widehat{r}_{M_1,i,j,n(T-1)}, \widehat{r}_{M_2,i,j,n(T-1)}, \ldots, \widehat{r}_{M_L,i,j,n(T-1)} \right\}. \quad (9)$$

By making the derivative of $\mathbf{g}_{i,j,n(T-1)}$ with respect to $\mathbf{y}$ equal to 0, and according to (4) and (5), we can denote the region features as:

$$\begin{aligned} \mathbf{g}_{i,j,nT}^{*} &= \left(\mathbf{C}^{L^\top} \widehat{\mathbf{R}}^{L^\top} \widehat{\mathbf{R}}^L \mathbf{C}^L \right)^{-1} \mathbf{C}^{L^\top} \widehat{\mathbf{R}}^{L^\top} \widehat{\mathbf{R}}^L \mathbf{R}^L \\ &= \left(\widehat{\mathbf{C}}^{L^\top} \widehat{\mathbf{C}}^L \right)^{-1} \widehat{\mathbf{C}}^{L^\top} \widehat{\mathbf{R}}^L \mathbf{R}^L, \end{aligned} \quad (10)$$

which is an unbiased estimation. However, there may exist related vectors in the matrix $\widehat{\mathbf{C}}^L$, which can cause matrix $\widehat{\mathbf{C}}^{L^\top} \widehat{\mathbf{C}}^L$ not invertible and make it difficult to determine the region features prediction $\mathbf{g}_{i,j,nT}^{*}$. Compared with the unbiased estimation mentioned in (10), ridge regression [30], [31] can improve the stability of the estimation by adding a punishment, that is, through unbiased loss estimation. Specifically, ridge regression aims to minimize prediction error:

$$\tilde{\mathbf{g}}_{i,j,nT} = \arg\min_{\mathbf{y}} \left(\left\| \widehat{\mathbf{R}}^L (\mathbf{C}^L \mathbf{y} - \mathbf{R}^L) \right\|^2 + \mu \left\| \mathbf{y} \right\|^2 \right), \quad (11)$$

and $\mu\|\mathbf{y}\|^2$ is the punishment for estimation $\mathbf{g}_{i,j,n(T-1)}^{*}$. As a result, the region features prediction is changed as follows:

$$\begin{aligned} \tilde{\mathbf{g}}_{i,j,nT} &= \left(\mathbf{C}^{L^\top} \widehat{\mathbf{R}}^{L^\top} \widehat{\mathbf{R}}^L \mathbf{C}^L + \mu \mathbf{I} \right)^{-1} \mathbf{C}^{L^\top} \widehat{\mathbf{R}}^{L^\top} \widehat{\mathbf{R}}^L \mathbf{R}^L \\ &= \left(\widehat{\mathbf{C}}^{L^\top} \widehat{\mathbf{C}}^L + \mu \mathbf{I} \right)^{-1} \widehat{\mathbf{C}}^{L^\top} \widehat{\mathbf{R}}^L \mathbf{R}^L, \end{aligned} \quad (12)$$

where matrix $\mathbf{I} \in \mathbb{R}^{M \times M}$. In addition, when using the region features predicted by ridge regression (12) and the features obtained by unbiased estimation (8) to predict the popularity of the content, if $|\mathbf{g}_{i,j,nT}^{*}| \leq \zeta$, for $\forall \delta > 0$ we have

$$|\tilde{\mathbf{g}}_{i,j,nT} \mathbf{C}_m^\top - \mathbf{g}_{i,j,nT}^{*} \mathbf{C}_m^\top| \leq (\delta + \zeta\mu) \left\| \mathbf{C}_m \right\|_{V^{-1}}, \quad (13)$$

with probability at least $1 - 2e^{-2\delta^2}$ from the lemma 1 in [30], where matrix $V = (\widehat{\mathbf{C}}^{L^\top} \widehat{\mathbf{C}}^L + \mu \mathbf{I})$.

By adopting the weight matrix $\widehat{\mathbf{R}}^L$, we can more accurately denote the region features through user preferences and predict future region features. On this basis, using ridge regression estimation can make the estimation of different region features more stable and the prediction error smaller as well.

V. Cooperative Caching Between Regions

Based on the proposed prediction model, in this section, we propose a cooperative caching method to provide cooperative caching decisions among multiple regions with similar region features. At first, we analyze the feasibility of cooperative caching in adjacent regions with similar features and design an algorithm for dividing similar regions into one cooperative area. After that, we proposed cache benefits to measure the gains of caching the content, that is, how much delay can be reduced for users to obtain the content. Finally, we formulate the caching decision problem for each region as a non-cooperative game problem and play the game by maximizing the cache benefits for each satellite.

A. Cooperative Area Division

After completing the prediction of the region features of each region, we divide the regions which are geographically adjacent and have similar region features into one cooperative area. In one cooperative area, we adopt a cooperative caching method to enable the satellites to cache more contents that users may request. Therefore, when providing content delivery service for users, the STIN can offload more redundant traffic from networks and reduce the content delivery delay by caching more popular contents on satellites.

Since the features of different regions are not normalized, to eliminate the influence of the feature vectors of regions having different sizes, the definition of similarity [32] between regions can be expressed as:

$$\beta_{(i,j)(i',j'),nT} = 1 - \frac{\left\| \tilde{\mathbf{g}}_{i,j,nT} - \tilde{\mathbf{g}}_{i',j',nT} \right\|^2}{\left\| \tilde{\mathbf{g}}_{i,j,nT} \right\| \left\| \tilde{\mathbf{g}}_{i',j',nT} \right\|}. \quad (14)$$

In this section, the main problem we focus on is the problem within the period nT, so the time subscript of the variable is not important. We simplify the above formula to:

$$\beta_{(i,j)(i',j')} = 1 - \frac{\left\| \tilde{\mathbf{g}}_{i,j} - \tilde{\mathbf{g}}_{i',j'} \right\|^2}{\left\| \tilde{\mathbf{g}}_{i,j} \right\| \left\| \tilde{\mathbf{g}}_{i',j'} \right\|}. \quad (15)$$

Besides, other variables will also be simplified, that is, the time subscripts of variables are ignored, and it does not make any difference to our next analysis.

If the similarity between the two regions satisfies the condition:

$$\beta_{(i,j)(i',j')} < \beta, \quad (16)$$

where $\beta \in [0,1]$ is the similarity threshold for judging whether two regions are similar. The larger the value of the threshold β is, the stricter conditions for judging two regions are similar.

The number of users or the number of requests made by users differs in each region, and we should prioritize regions with more users or requests. For example, as shown in Fig. 2, select the region with the largest number of requests sent by users during the previous period, which is region (1,2), to start the division of the cooperative area, and let it as the first cooperative area

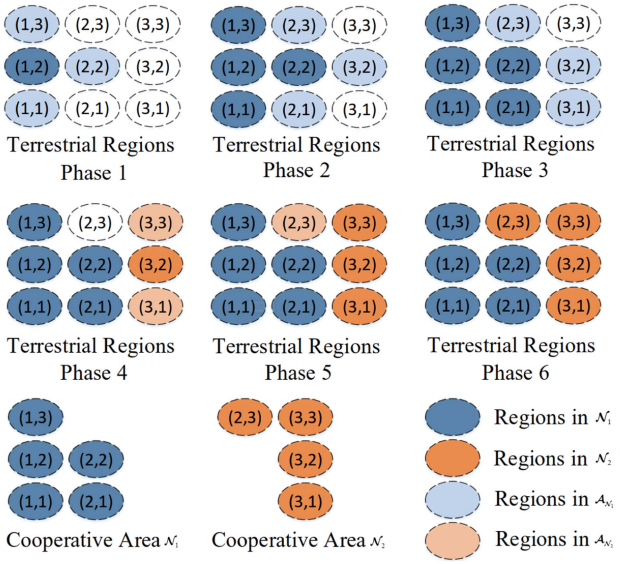


Fig. 2. Illustration of typical cooperative area division process for given 9 regions.

$\mathcal{N}_1 = (1,2)$. Besides, the adjacent regions of cooperative area $\mathcal{N}_1$ are denoted as $\mathcal{A}_{\mathcal{N}_1}$. Each region in $\mathcal{A}_{\mathcal{N}_1}$ is adjacent to at least one region in cooperative area $\mathcal{N}_1$. Then for each adjacent region $(i',j'), (i',j') \in \mathcal{A}_{\mathcal{N}_1}$, we calculate the similarity between it and every region in the cooperative area $\mathcal{N}_1$ according to (15). If the adjacent region (i',j') satisfied the condition:

$$\beta_{(i,j)(i',j')} < \beta, \forall (i,j) \in \mathcal{N}_1, \tag{17}$$

the adjacent region (i',j') should be added to the cooperative area $\mathcal{N}_1$. For instance, as shown in Fig. 2, the region in $\mathcal{A}_{\mathcal{N}_1} = \{(1,3),(1,1),(2,2)\}$ are similar to every region in $\mathcal{N}_1 = \{(1,2)\}$, so the cooperative area should be updated as $\mathcal{N}_1 = \{(1,1),(1,2),(1,3),(2,2)\}$. At the same time, the adjacent area should also be updated, which currently is $\mathcal{A}_{\mathcal{N}_1} = \{(2,1),(2,3),(3,2)\}$. After that, calculate the similarity between regions according to the above steps. The division of this cooperative area is accomplished until all adjacent regions of the current cooperative area do not satisfy the similarity condition. For example, in the third phase of Fig. 2, all adjacent regions in $\mathcal{A}_{\mathcal{N}_1} = \{(2,3),(3,1),(3,2)\}$ do not satisfy the condition in (17), so the division of first cooperative area is accomplished. The subsequent step is to divide the next cooperative area. In the regions that are not divided into any cooperative areas, the region with the largest number of requests is selected as the next cooperative area, and the division steps are the same as the method of dividing the first cooperative area $\mathcal{N}_1$. Note that each region belongs to and only belongs to one cooperative area.

Algorithm 1 is the process of dividing cooperative areas. When there are $\mathcal{N}$ regions, Algorithm 1 requires at most $|\mathcal{N}_k| \cdot |\mathcal{A}_{\mathcal{N}_k}| \cdot |\mathcal{N}|$ calculations to calculate and divide all cooperative regions, where $\mathcal{A}_{\mathcal{N}_k}$ is the adjacent region of the cooperative area $\mathcal{N}_k$. For our definition of adjacent areas, $\mathcal{A}_{\mathcal{N}_k}$ satisfies $|\mathcal{A}_{\mathcal{N}_k}| \geq 2 \cdot |\mathcal{N}_k| + 2$ and $|\mathcal{N}_k| \leq |\mathcal{N}|$. Hence, Algorithm 1 requires a total of $\frac{|\mathcal{N}|^3}{3}$ calculations. Therefore, the complexity of Algorithm 1 is $O(|\mathcal{N}|^3)$, and it only needs to be calculated once in one period T, which is completely acceptable.

Algorithm 1: Cooperative Area Division Algorithm.

Input: Similarity threshold β and features of all regions $\mathcal{N}$;
Output: cooperative area $\mathcal{N}_1$,$\mathcal{N}_2$,...$\mathcal{N}_k$...;

Step1 Initialize :
Denote $\mathcal{N}^d$ as the regions which are divided into some cooperative areas;
$\mathcal{N}^d = \emptyset$;
Step2 Divide cooperative areas:
for $k = 1, 2, 3, \ldots$ **do**
  Select one region with most request times in regions $\{\mathcal{N} \backslash \mathcal{N}^d\}$, denoted as (i,j);
  Let $\mathcal{N}_k = \{(i,j)\}$;
  for *each region* $(i',j') \in \mathcal{A}_{\mathcal{N}_k}$ **do**
    flag=1;
    for $(i'',j'') \in \mathcal{N}_k$ **do**
      if $\beta_{(i',j'),(i'',j'')} \geq \beta$ **then**
        flag=0;
        break;
      end
    end
    if *flag==1* **then**
      $\mathcal{N}_k \leftarrow \mathcal{N}_k + (i',j')$;
      Update $\mathcal{A}_{\mathcal{N}_k}$
    end
  end
  Update divided cooperative areas:
  $\mathcal{N}^d \leftarrow \mathcal{N}^d + \mathcal{N}_k$;
end
return *cooperative areas* $\mathcal{N}_1$,$\mathcal{N}_2$,...,$\mathcal{N}_k$....

Since the division of similar regions is executed periodically every T time, the region's features can be collected by either the ground station or the GEO satellite. Subsequently, the cooperative areas division can be executed following Algorithm 1.

B. Cache Benefits

Considering the limited satellite cache space, the greater the benefits of caching a certain content, the higher the priority that the content should be cached. During the n-th time period, the cache benefits of the satellite for caching content f_m in region (i,j) are defined as $u_{i,j,m,nT}$, simplified to $u_{i,j,m}$, which contains two parts. The first part is the time saved by users in the region (i,j) to obtain the content f_m on the serving satellite in service region (i,j). The second part is the time saved by users in other regions in the same cooperative area as region (i,j) to obtain content f_m on satellite in service region (i,j).

Users in a region can directly retrieve the required content from the serving satellite of the region or retrieve the content from satellites serving other regions in the same cooperative areas. If none of these satellites cache the content, the user can only retrieve it from the remote cloud servers. Define

$x_{m,i,j} \in \{0,1\}$ as the content f_m whether cached by satellite in service region (i,j), and if content f_m is cached, $x_{m,i,j}=1$, otherwise, $x_{m,i,j}=0$. Besides, denote the delay of terrestrial users retrieving contents from the cloud server through the satellite and the ground station as τ^{TP}, the delay of retrieving contents from the satellite as τ^{TS}, the inter-satellite retrieval delay as τ^{SS}. Therefore, the cache benefits of the first part of $u_{i,j,m}$ can be denoted as:

$$u_{i,j,m}^{i,j} = r_{m,i,j}^{*}\left(\tau^{\mathrm{TP}} - \tau^{\mathrm{TS}}\right). \tag{18}$$

Considering one cooperative area, the saved time of users in regions with a distance of 1 hop from region (i,j) retrieving content at the satellite in service region (i,j) is:

$$u_{i,j,m}^{N_{i,j}^{1}} = \sum_{(i',j')\in N_{i,j}^{1}} \left(1 - x_{m,i',j'}\right) r_{m,i',j'}^{*}\left(\tau^{\mathrm{TP}} - \tau^{\mathrm{TS}} - \tau^{\mathrm{SS}}\right), \tag{19}$$

where $N_{i,j}^{1}$ represents the regions set, that each region of it is in the same cooperative area as region (i,j), as well as the distance from (i',j') is one hop. For instance, as shown in Fig. 2, $N_{3,2}^{1} = \{(3,1),(3,3)\}$. In the same way, we can conclude that the time saved by the users of $N_{i,j}^{K}$ due to the satellite in service region (i,j) caching content f_m is:

$$u_{i,j,m}^{N_{i,j}^{K}} = \sum_{(i',j')\in N_{i,j}^{K}} \left(1 - x_{m,i',j'}\right) \prod_{(i'',j'')\in \mathcal{J}_{i',j'}^{K-1}} \left(1 - x_{m,i'',j''}\right) r_{m,i',j'}^{*}\left(\tau^{\mathrm{TP}} - \tau^{\mathrm{TS}} - K\tau^{\mathrm{SS}}\right), \tag{20}$$

where $\mathcal{J}_{i',j'}^{K-1} = \{N_{i',j'}^{1} \cup N_{i',j'}^{2} \ldots \cup N_{i',j'}^{K-1}\}$. And K represents the maximum inter-regional distance that users in the region (i,j) can retrieve content from other regions (i',j') in the same cooperative area, which should satisfy:

$$\tau^{\mathrm{TP}} - \tau^{\mathrm{TS}} - K\tau^{\mathrm{SS}} \geq 0, \tau^{\mathrm{TP}} - \tau^{\mathrm{TS}} - (K+1)\tau^{\mathrm{SS}} \leq 0. \tag{21}$$

If the distance exceeds K, users in the region (i,j) will not go to the region (i',j') to retrieve the content. This is because retrieving content from satellites in a region with a distance of $K+1$ takes longer time than retrieving content from the cloud server.

According to the derivation above, the second part of cache benefits $u_{i,j,m}$ is:

$$u_{i,j,m}^{\mathcal{J}_{i,j}^{K}} = u_{i,j,m}^{N_{i,j}^{1}} + u_{i,j,m}^{N_{i,j}^{2}} \ldots + u_{i,j,m}^{N_{i,j}^{K}}, \tag{22}$$

where $u_{i,j,m}^{N_{i,j}^{K}}$ represent the cache benefits brought to users in regions $N_{i,j}^{K}$. Note that $\mathcal{J}_{i,j}^{K}$ and the cooperative area $\mathcal{N}_k$, where the region (i,j) belongs, are not the same regions set. For example, in the fourth phase of dividing cooperative areas as shown in Fig. 2, if K equals to 1, for region (1,1), $\mathcal{J}_{1,1}^{K} = \{(1,2),(2,1)\}$, while the cooperative area that region (1,1) belongs to is $\mathcal{N}_1 = \{(1,1),(1,2),(1,3),(2,1),(2,2)\}$, so $\mathcal{J}_{1,1}^{K} \neq \mathcal{N}_1$.

Therefore, the total cache benefits of satellite in region (i,j) caching content f_m can be expressed as

$$u_{i,j,m} = \left(u_{i,j,m}^{i,j} + u_{i,j,m}^{\mathcal{J}_{i,j}^{K}}\right) x_{m,i,j}. \tag{23}$$

The cache benefits of all contents on the satellite serving region (i,j) can be derived as:

$$\begin{aligned} u_{i,j} &= \sum_{m=1}^{M} u_{i,j,m} x_{m,i,j} \\ &= \sum_{m=1}^{M} \left(u_{i,j,m}^{i,j} + u_{i,j,m}^{\mathcal{J}_{i,j}^{K}}\right) x_{m,i,j} \\ &= u_{i,j}^{i,j} + u_{i,j}^{\mathcal{J}_{i,j}^{K}}. \end{aligned} \tag{24}$$

C. Problem Formulation

Let $\mathbf{X}_k$ represent the set of all possible caching strategies in the cooperative area $\mathcal{N}_k$. The cache benefits brought by satellite serving region (i,j) can be denoted as $u_{i,j}(\mathbf{x}_{(i,j)}, \mathbf{x}_{-(i,j)}), \mathbf{x}_{(i,j)} \in \mathbf{X}_k, \mathbf{x}_{-(i,j)} \subseteq \mathbf{X}_k$, where $\mathbf{x}_{(i,j)} = (x_{0,i,j}, x_{1,i,j}, \cdots x_{M,i,j})$ represents the caching decision of region (i,j) and $\mathbf{x}_{-(i,j)}$ represents caching decisions of regions other than region (i,j).

However, according to the definition of cache benefits in (24), that the cache benefits $u_{i,j}(\mathbf{x}_{(i,j)}, \mathbf{x}_{-(i,j)})$ is affected by other satellite caches. Consequently, one satellite caching decision could be influenced by other satellites. For example, in the sixth phase of dividing cooperative areas as shown in Fig. 2, if satellite serving region (3,1) caching a content f_m, then users in the region (3,1) would not send the request to satellite serving region (3,2). As a result, the cache benefits $u_{m,3,2}$ is changed according to (23), and the caching decision on satellite serving region (3,2) may also be changed accordingly.

To maximize the cache benefits of the satellite cache in each region of one cooperative area $\mathcal{N}_k$, as well as consider the influence of adjacent satellites' cache, we formulate the caching decisions problem in one cooperative area $\mathcal{N}_k$ as a non-cooperative game, denoted as $\mathcal{G}_k$. By adopting a non-cooperative game to make caching decisions, each satellite can use a distributed algorithm to calculate the most appropriate cache, avoiding the huge overhead brought by the centralized way. Note that $\mathcal{G}_k = \{\mathcal{N}_k, \mathbf{X}_k, U_{i,j}\}$, where $\mathcal{N}_k$ is the players in the game and $\mathbf{X}_k$ is the action space, and $U_{i,j}$ is the utility function. The utility function here is defined as the sum of the cache benefits of the satellites in the region (i,j) and $\mathcal{J}_{(i,j)}^{K}$,

$$U_{i,j} = u_{i,j} + \sum_{(i',j')\in \mathcal{J}_{(i,j)}^{K}} u_{i',j'}. \tag{25}$$

Therefore, the game can be denoted as follows:

$$\mathcal{G}_k : \max U_{i,j}\left(\mathbf{x}_{(i,j)}, \mathbf{x}_{-(i,j)}\right) \quad \forall (i,j) \in \mathcal{N}_k. \tag{26}$$

In summary, in the above game, the players are all satellites in the cooperative area $\mathcal{N}_k$, and the game strategy of each satellite is the satellite's caching strategy $\mathbf{x}_{(i,j)} = (x_{0,i,j}, x_{1,i,j}, \cdots x_{M,i,j})$ in $\mathcal{N}_k$, and the utility function is the sum of the cache benefits $U_{i,j}(\mathbf{x}_{(i,j)}, \mathbf{x}_{-(i,j)})$.

VI. Non-Cooperative Game Caching Strategy

In this section, we use non-cooperative game theory [33], [34] to solve the problem in the previous section. The non-cooperative game is a mathematical framework that describes how multiple decision-makers compete with each other in order to pursue their interests so that the whole can reach a local or global optimal solution. In the STIN, each satellite is an independent decision-maker committed to pursuing its own optimal caching strategy to improve content delivery efficiency. However, when cooperative caching is employed, one satellite's caching decisions may directly impact the performance of other satellites in providing content delivery services. By using non-cooperative game theory, we can simulate and analyze the competition and cooperation relationships between satellites, thereby formulating strategies for the entire system to achieve a collaborative optimal solution. This method considers the satellite's own pursuit of interests and uses a game framework to promote satellite cooperation in mutual competition to achieve overall performance improvement. Through non-cooperative games, we can use distributed algorithms to make the caching benefits of satellites in the cooperation area reach a local maximum, thereby determining the caching strategy for each satellite.

First, we prove that the proposed non-cooperative game has Nash equilibrium, i.e., the optimal solution of the proposed problem exists. Second, we design a caching decision algorithm to reach Nash equilibrium for each cooperative area, which can achieve Nash equilibrium with polynomial complexity through the distributed caching decision process.

A. Analysis of Nash Equilibrium

Before we provide the solution to the proposed non-cooperative game problem, we first define the Nash equilibrium under this non-cooperative game and prove the existence of the Nash equilibrium.

Definition 1 (Nash Equilibrium): If there exists a set of caching strategies $\left(\mathbf{x}^*_{(i,j)}, \mathbf{x}^*_{-(i,j)}\right)$ satisfy: none of the satellites in the cooperative area can improve its cache benefits by unilaterally changing its own caching strategy, such a set of strategies is called the pure strategy Nash equilibrium of the non-cooperative game $\mathcal{G}_k$. Hence, the Nash equilibrium of the game can be described as follows:

$$U_{i,j}\left(\mathbf{x}^*_{(i,j)}, \mathbf{x}^*_{-(i,j)}\right) \geq U_{i,j}\left(\mathbf{x}'_{(i,j)}, \mathbf{x}^*_{-(i,j)}\right), \quad \forall (i,j) \in \mathcal{N}_k, \mathbf{x}'_{(i,j)} \in \mathbf{X}_k, \mathbf{x}'_{(i,j)} \neq \mathbf{x}^*_{(i,j)} \tag{27}$$

Definition 2 (Potential Game): If there is a function Φ, which satisfies

$$\begin{aligned} &\Phi\left(\mathbf{x}'_{(i,j)}, \mathbf{x}_{-(i,j)}\right) - \Phi\left(\mathbf{x}_{(i,j)}, \mathbf{x}_{-(i,j)}\right) \\ &= U_{i,j}\left(\mathbf{x}'_{(i,j)}, \mathbf{x}_{-(i,j)}\right) - U_{i,j}\left(\mathbf{x}_{(i,j)}, \mathbf{x}_{-(i,j)}\right), \\ &\forall (i,j) \in \mathcal{N}_{\mathrm{k}}, \mathbf{x}'_{(i,j)} \in \mathbf{X}_k, \mathbf{x}'_{(i,j)} \neq \mathbf{x}_{(i,j)}, \end{aligned} \tag{28}$$

we call the game $\mathcal{G}_k$ as a potential game, and the function Φ is the potential function of the game.

Algorithm 2: Cooperative Caching Algorithm.

Step1 Initialize caching strategy:
Set max iteration times P;
for *each region* $(i,j) \in \mathcal{N}_k$ **do**
  for *content* $m \in \mathcal{M}$ **do**
    Calculate $u^0_{i,j,m}$ according to (29), $x_{m,i,j}$=0;
  end
  Find the top c pieces of contents of cache benefits:
  $\mathcal{M}^c_{i,j} \leftarrow \mathbf{Top}^c\left\{u_{i,j,1}, u_{i,j,2}, ..., u_{i,j,M}\right\}$;
  for *content* $m \in \mathcal{M}^c_{i,j}$ **do**
    $x_{m,i,j}$=1;
  end
  The initial caching strategy of region (i,j) is:
  $\mathbf{x}_{(i,j)}$=$\{x_{1,i,j}, x_{2,i,j}, ..., x_{M,i,j}\}$;
end
Step2 Check satellite cache benefits:
ischanged=true, p=1;
while *ischanged and* $p \leq P$ **do**
  for *each region* $(i,j) \in \mathcal{N}_k$ **do**
    ischanged=false;
    for *content* $m \in \mathcal{M}^c_{i,j}$ **do**
      Calculate correct $u_{i,j,m}$ according to (23),
      $u^p_{i,j,m}$=$u_{i,j,m}$;
      if $u^p_{i,j,m} \neq u^{p-1}_{i,j,m}$ **then**
        ischanged=true;
      end
    end
    $\mathcal{M}^c_{i,j} \leftarrow \mathbf{Top}^c\left\{u_{i,j,1}, u_{i,j,2}, ..., u_{i,j,M}\right\}$
    for *content* $m \in \mathcal{M}^c_{i,j}$ **do**
      $x_{m,i,j}$=1;
    end
    Update caching strategy of region (i,j):
    $\mathbf{x}'_{(i,j)}$=$\{x_{1,i,j}, x_{2,i,j}, ..., x_{M,i,j}\}$, $\mathbf{x}_{(i,j)} = \mathbf{x}'_{(i,j)}$;
  end
  $p \leftarrow p+1$;
end
Step3 Make caching decision:
for *each region* $(i,j) \in \mathcal{N}_k$ **do**
  $\mathbf{x}^*_{(i,j)}$=$\mathbf{x}'_{(i,j)}$;
end

We analyze the existence of Nash equilibrium for game $\mathcal{G}_k$ by the following theorem and its proof.

Theorem 1: Game $\mathcal{G}_k$ is a potential game that has at least one pure-strategy Nash equilibrium.

Proof: Please see Appendix A for details.

The potential game must have a Nash equilibrium solution according to [33], therefore, the game $\mathcal{G}_k$ exists at least one Nash equilibrium.

B. Cooperative Caching Algorithm

In this subsection, we solve the game $\mathcal{G}_k$ according to another important property of the potential game, that the strategies combination that makes the potential function reach a local or global maximum value is a set of pure strategy Nash equilibrium [35]. Therefore, after we prove that the game $\mathcal{G}_k$ is a potential game,

TABLE I
EVALUATION PARAMETERS

Parameters	Values
Height of Orbits	1200km
Number of LEO	16
Number of Terrestrial Regions	16
Delay between Users and Cloud Servers	500ms
Delay between Users and Satellites	300ms
Delay of Inter-satellites	40ms
Number of Contents' Dimensions	19
Number of Contents	1682
Number of Total Requests	100000

the critical step for us to find the Nash equilibrium solution is to find a set of caching strategies for satellites that can achieve local maximum cache benefits. The process of solving the Nash equilibrium of the game is shown in Algorithm 2.

The first step of Algorithm 2 is initialization. Assuming that the satellite serving region (i, j) in the cooperative area thinks that other satellites do not cache any contents, which is the "local only" initial state. Then the satellite calculates the initial cache benefits of each content denoted as $u^0_{i,j,m}$ according to

$$u^0_{i,j,m} = r^*_{m,i,j}\left(\tau^{\mathrm{TP}} - \tau^{\mathrm{TS}}\right) + \sum_{N^k_{i,j}\in\mathcal{J}^K_{i,j}} \sum_{(i',j')\in N^k_{i,j}} r^*_{m,i',j'}\left(\tau^{\mathrm{TP}} - \tau^{\mathrm{TS}} - k\tau^{\mathrm{SS}}\right). \tag{29}$$

The second step is to check each satellite's cache benefits in one cooperative area. Because the initial benefits of content cached by satellite are calculated under an ideal "local only" state, we should calculate the correct cache benefits under current caching strategies $(\mathbf{x}_{(i,j)}, \mathbf{x}_{-(i,j)})$. In the subsequent p-th iteration, check whether the cache benefits are correctly calculated from the cache strategy formulated after $(p-1)$-th iteration. Lines 20–24 are to check the benefits of the cached content f_m on satellite serving region (i, j) is calculated correctly under current caching strategies $(\mathbf{x}_{(i,j)}, \mathbf{x}_{-(i,j)})$. If the cache benefits are not calculated correctly, the Nash equilibrium has not been reached, and further iteration is still needed. Lines 27–32 are to update the caching strategy based on the corrected cache benefits, change it from $\mathbf{x}_{(i,j)}$ to $\mathbf{x}'_{(i,j)}$. Since we choose to cache content with more considerable cache benefits, the updated caching strategy satisfies $u_{i,j}(\mathbf{x}'_{(i,j)}, \mathbf{x}_{-(i,j)}) - u_{i,j}(\mathbf{x}_{(i,j)}, \mathbf{x}_{-(i,j)}) \geq 0$. Therefore, according to (31), the potential function also satisfies

$$\Phi\left(\mathbf{x}'_{(i,j)}, \mathbf{x}_{-(i,j)}\right) - \Phi\left(\mathbf{x}_{(i,j)}, \mathbf{x}_{-(i,j)}\right) \geq 0. \tag{30}$$

Each update and iteration make the potential function larger and converge gradually. When accomplishing checking the satellites' cache and finding that all satellite's cache benefits do not need to be changed, the potential function at this time converges to a local maximum, that is, the Nash equilibrium of $\mathcal{G}_k$ is reached. The proposed Algorithm 2 relies on a simple signaling interactions process of satellites in a cooperative area. Moreover, the complexity of the process is $O(|\mathcal{N}_k|^2 \cdot |P| \cdot c^2)$, where $|\mathcal{N}_k|$ and $|P|$ represent the scale of cooperative area $\mathcal{N}_k$ and the maximum number of iterations.

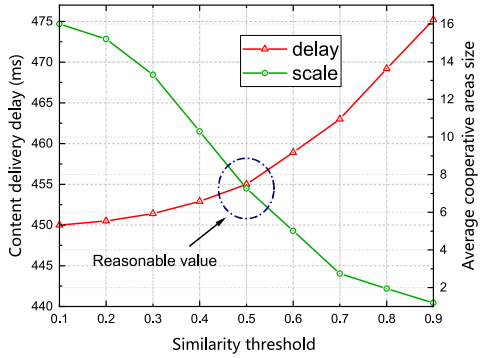


Fig. 3. Performance under different similarity threshold β.

VII. PERFORMANCE EVALUATION

In this section, we conduct extensive evaluations to compare the performance of the proposed cooperative caching method with existing methods. The numerical evaluation results show that our proposed scheme outperforms other cooperative caching schemes in terms of cache hit rate and content delivery delay. All experiments were performed on a PC with four 3.6 GHz CPUs, 16 GB RAM, and Windows 10 OS.

A. Real-World Dataset

In order to more realistically reflect the characteristics of users' requests for content, we use the real-world dataset MovieLens [36], which is widely used in research related to recommender systems and content caching [37], [38], [39]. The dataset records 100,000 requests for 1,682 items of content by 1,000 users from different regions, each request identifying when the request occurred, the geographic location (represented by zip code), and the requested content's name. We generate requests to satellites for content based on the actual time and location of each request. At the same time, the content requested by the user is a movie, and the category of the movie is used to represent the features of the content, and each content has 19-dimensional features respectively, i.e., D=19. In our scheme, the region features reflect the preference of users in the region for the 19-dimensional features in the content items, which determines that the region features are also 19-dimensional. The more feature dimensions of content, the corresponding increase in the dimensions of region features, and the more accurately they can reflect the preferences of regional users. Consequently, we can more accurately estimate the popularity of contents in different regions and execute more reasonable caching strategies.

B. Evaluation Setup

Consider the satellite constellation using OneWeb's polar-orbiting satellite constellation [40], which has nearly 800 LEO satellites. The area where users are distributed is about 10 million square kilometers, accounting for about 1/50 of the earth's surface area. Therefore, the number of LEO satellites serving those regions is 16, distributed on four orbital planes. In order to ensure that each region has at least one satellite in service, we also divide the regions where users are located into 16

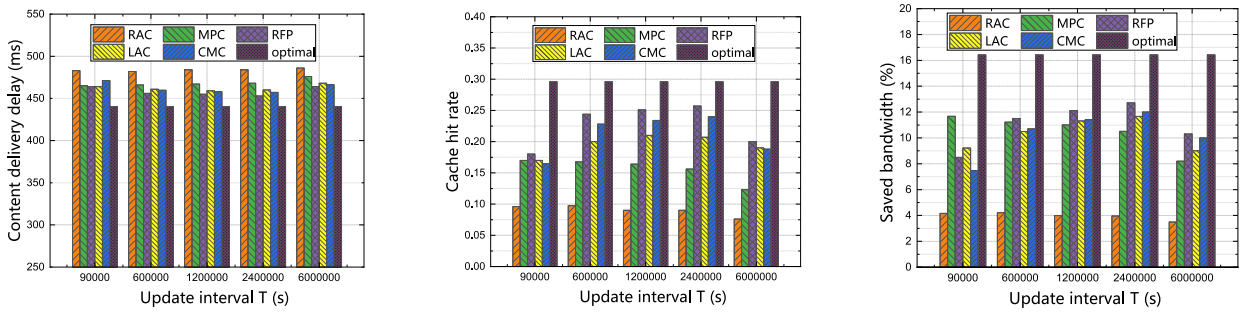


Fig. 4. Performance under different update interval T. (a) Content delivery delay verse different update interval T. (b) Cache hit rate verse different update interval T. (c) Saved bandwidth verse different update interval T.

regions, i.e., $\mathcal{N} = \{(i, j) \mid i^* j \leq 16, 0 < i \leq 4, 0 < j \leq 4\}$. In addition, considering that the number of requests in each period is different in different regions, L cannot be defined as a fixed value and should be set to the top $x\%$ of the number of requests in each period, for example, set L as the top 10% of the number of requests. The evaluation parameters are shown in Table I.

The caching strategies implemented for comparison are listed as follows:

- *RAC (Random cache):* This strategy allows satellites to cache contents randomly.
- *MPC (Most popular cache):* This strategy allows satellites to cache contents that are most popular in a greedy manner [41].
- *LAC (Location-aware cache):* This strategy is derived from location-aware content caching proposed in [30]. Satellites cache contents according to the content popularity based on location but without executing a non-cooperative game.
- *CMC (Cooperative multi-layer cache):* This strategy is derived from a cooperative multi-layer edge caching in the STIN [14].
- *RFP (Region features prediction cache):* This strategy is proposed in this paper. Satellites serving similar regions implement a cooperative caching method.
- *Optimal (Theoretically optimal cache):* This strategy is the theoretical optimal result of executing cooperative caching in cooperative areas.

C. Performance Analysis

Fig. 3 represents the effect of the similarity threshold when dividing the cooperative area on our scheme. We can find that with the increment of the similarity threshold, the scale of the average cooperative area is gradually reduced, and the delay for users to retrieve content is accordingly increased. Considering that the complexity brought by the process that satellites in the cooperative area execute Algorithm 2 is related to the size of the cooperative area, i.e., $O(|\mathcal{N}_k|^2)$. Hence it is necessary to set a reasonable similarity threshold to control the overhead within an acceptable range and ensure that users can obtain content with a low delay. It can be seen from the experimental results that when the similarity threshold value is 0.5, it cannot only ensure a low content acquisition delay but also avoid high overhead. In the following experiments, in order to ensure that the scheme can have a low delay within a reasonable range, the similarity threshold β is designed to be 0.5 by default.

Fig. 4 shows the performance of different schemes under different update intervals T. We validate our solution from three indicators: content delivery delay, cache hit rate, and saved bandwidth. Among them, the content delivery delay has a great impact on the users' experience, while the cache hit rate and bandwidth saved are measured from the perspective of the entire network. We can observe that with the increase of update interval T, the performance of the proposed scheme RFP first improves and then deteriorates. If the update period is short, the content requests aggregated by users in a region in one period may not precisely represent the more popular content, so the prediction scheme cannot achieve the best results. At the same time, if the update period is too large, some content will become popular at the beginning of this interval and then become unpopular. This part of the contents occupies a more significant proportion in the current period when we predict the features of the region, while they cannot reflect the latest user preferences. Therefore, it also biases our prediction of region features based on the users' preferences. When the update period is 2.4×10^6 seconds, our scheme significantly outperforms others. Therefore, we can conclude that under the update period 2.4×10^6 seconds, users' requests in one interval can well reflect their preferences in their region. Based on this, the region features can be predicted more precisely, and cooperative caching between similar regions can be carried out better. In addition, the scheme RFP we proposed is closer to the theoretically optimal solution in terms of content delivery delay, cache hit rate and saved bandwidth.

Fig. 5 shows the performance comparison of five caching strategies under different cache spaces. As the cache capacity gradually increases, the five schemes' performance improves. This is because the satellites' cache space becomes more extensive, which can cache more contents that users may need. When the cache space is large enough, such as cache capacity $c = 160$ and $c = 320$, the performance of schemes MPC and LAC are very close, while scheme RFP can further improve the cache performance. The reason is that our solution is a cooperative caching method, and the goal is to improve the gain of the entire cooperative area, not a single region. Hence users in one cooperative area can get better services when satellites

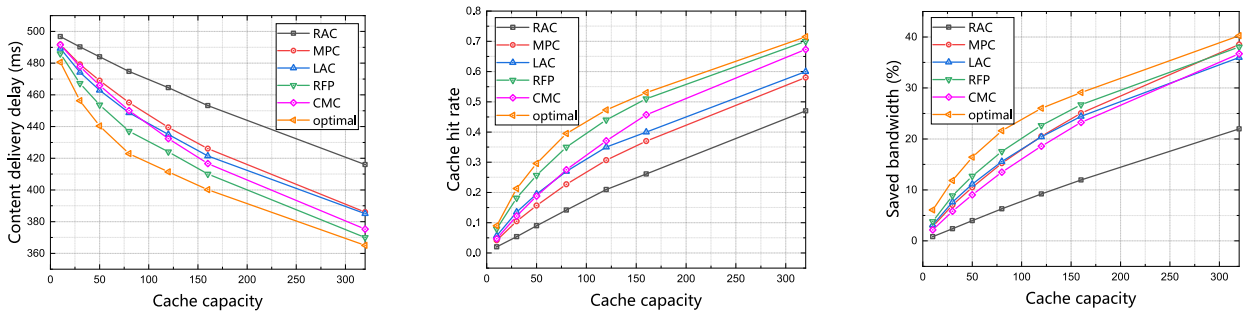


Fig. 5. Performance under different cache capacity c. (a) Content delivery delay verse different cache capacity c. (b) Cache hit rate verse different cache capacity c. (c) Saved bandwidth verse different cache capacity c.

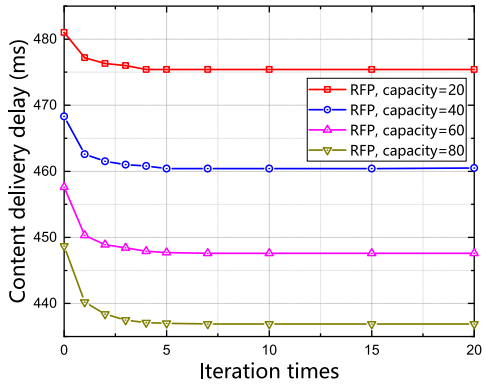


Fig. 6. Content delivery delay under different iteration times P.

execute a cooperative caching method. Moreover, when the cache space is larger, the disparity between the RFP and the theoretically optimal solution is reduced compared to the scenario with a smaller cache space. This phenomenon is primarily attributed to the fact that with a larger cache space, the cached contents closely align with the actual popular contents, resulting in a closer resemblance to the theoretically optimal outcome.

Fig. 6 represents the performance impact of the number of iterations P in Algorithm 2 on our cooperative caching algorithm. We compare the performance of our schemes with different iterations times under different cache capacity c. It can be found that with the increment of the number of iterations, the performance of the proposed scheme gradually converges and eventually reach stability. Besides, if the cache capacity is more extensive, it will need more iterations to achieve convergence. This is because when the cache capacity becomes extensive, more contents should be checked to correctly calculate the cache benefits of each update and iteration. As shown in step 2 in Algorithm 2, checking more contents means that it needs more iterations to achieve convergence is more at the same time. Fortunately, no matter the size of the cache capacity, our solution can achieve a relatively ideal performance when iteration times $P = 5$.

Fig. 7 represents the performance of the five schemes under different ground-satellite and inter-satellite delays [14]. The delay of terrestrial users connecting to LEO satellites is affected

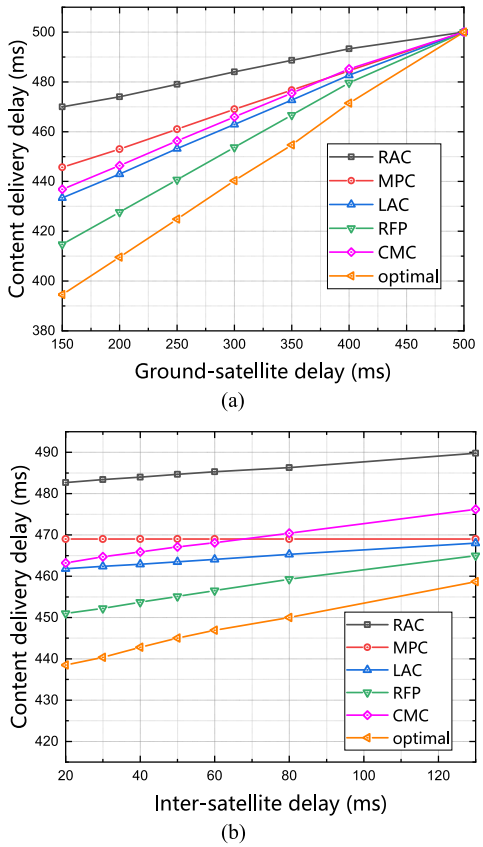


Fig. 7. Performance under different ground and satellite delay τ^{TS} and different inter-satellite delay τ^{SS}. (a) Content delivery delay verse different τ^{TS}. (b) Content delivery delay verse different τ^{SS}.

by other factors, such as satellites' height and weather, so it is necessary to observe the performance of our scheme under different ground-satellite delays. As shown in Fig. 7(a), we can find that our scheme significantly outperforms other schemes when there is a shorter ground-satellite delay. At $\tau^{\mathrm{TS}} = 150$ ms, the gain brought by our scheme is much higher than other caching strategies. Similarly, considering the uncertainty of the

inter-satellite link, we observed the performance under different inter-satellite delays. The evaluation results are shown in Fig. 7(b). We can find that the lower the delay of the inter-satellite link, the higher the benefit of using our scheme. For example, when the inter-satellite link delay is 20 ms, the benefit of using our scheme is much higher than the delay is 130 ms. However, regardless of the inter-satellite link delay, compared with the other four strategies, our scheme is closer to the theoretically optimal scheme and can provide users with lower latency and more stable content delivery services.

VIII. Conclusion

In this paper, we investigated the problem of satellites providing content delivery services for users in areas that terrestrial networks cannot cover. We first described and predicted the features of each region based on its historical request information during a period. Based on the predicted regional characteristics, we divided regions with similar characteristics and geographical proximity into one cooperative area. Then, to fully use the limited satellite cache, we adopted a cooperative caching method to set up the cache for each satellite in one cooperative area. We formulated the caching decision problem of each satellite in one cooperative area into a non-cooperative game. Then, we proved the existence of the Nash equilibrium of the proposed game and designed an algorithm that can achieve Nash equilibrium with polynomial complexity. Finally, we conducted extensive evaluations which adopted real-world datasets and considered user requests from different geographical locations. Compared with the existing non-cooperative and cooperative caching methods, our scheme performs better in terms of content delivery delay, bandwidth consumption, and cache hit ratio.

Appendix A Proof of Theorem 1

We define the potential function

$$\Phi\left(\mathbf{x}_{(i,j)},\mathbf{x}_{-(i,j)}\right)=\sum_{(i,j)\in\mathcal{N}_k}u_{i,j}\left(\mathbf{x}_{(i,j)},\mathbf{x}_{-(i,j)}\right), \tag{31}$$

which represents the benefits that all satellites bring to users located in the regions they are serving under caching strategies $(\mathbf{x}_{(i,j)},\mathbf{x}_{-(i,j)})$. Considering the satellite caching strategy of region (i,j) unilaterally change from $\mathbf{x}_{(i,j)}$ to $\mathbf{x}'_{(i,j)}$, and the other regions caching strategies remain unchanged, which still can be denoted as $\mathbf{x}_{-(i,j)}$. The cooperative area $\mathcal{N}_k$ where the region (i,j) belongs to can be disassembled into the following three parts: the first part is the region (i,j) itself; the second part is the regions whose distance from the region (i,j) is not greater than K hops, denoted as $\mathcal{J}^K_{i,j}$; the third part is the regions whose distance from region (i,j) exceeds than K hops, denoted as $\{\mathcal{N}_k\backslash\mathcal{J}^K_{i,j}\backslash(i,j)\}$. Hence, the potential function can be further derived as follows:

$$\begin{aligned}\Phi\left(\mathbf{x}_{(i,j)},\mathbf{x}_{-(i,j)}\right)&=u_{i,j}\left(\mathbf{x}_{(i,j)},\mathbf{x}_{-(i,j)}\right)\\&+\sum_{(i',j')\in\mathcal{J}^k_{i,j}}u_{i',j'}\left(\mathbf{x}_{(i',j')},\mathbf{x}_{-(i',j')}\right)\\&+\sum_{(i',j')\in\left\{\mathcal{N}_k\backslash\mathcal{J}^K_{i,j}\backslash(i,j)\right\}}u_{i',j'}\left(\mathbf{x}_{(i',j')},\mathbf{x}_{-(i',j')}\right).\end{aligned} \tag{32}$$

If we change the caching strategy of satellite in region (i,j) from $\mathbf{x}_{(i,j)}$ to $\mathbf{x}'_{(i,j)}$, the change of the potential function is

$$\begin{aligned}&\Phi\left(\mathbf{x}'_{(i,j)},\mathbf{x}_{-(i,j)}\right)-\Phi\left(\mathbf{x}_{(i,j)},\mathbf{x}_{-(i,j)}\right)\\&=\left\{u_{i,j}\left(\mathbf{x}'_{(i,j)},\mathbf{x}_{-(i,j)}\right)-u_{i,j}\left(\mathbf{x}_{(i,j)},\mathbf{x}_{-(i,j)}\right)\right\}\\&+\sum_{(i',j')\in\mathcal{J}^k_{i,j}}u_{i',j'}\left(\mathbf{x}_{(i',j')},\mathbf{x}'_{-(i',j')}\right)\\&-\sum_{(i',j')\in\mathcal{J}^k_{i,j}}u_{i',j'}\left(\mathbf{x}_{(i',j')},\mathbf{x}_{-(i',j')}\right)\\&+\sum_{(i',j')\in\left\{\mathcal{N}_k\backslash\mathcal{J}^K_{i,j}\backslash(i,j)\right\}}u_{i',j'}\left(\mathbf{x}_{(i',j')},\mathbf{x}'_{-(i',j')}\right)\\&-\sum_{(i',j')\in\left\{\mathcal{N}_k\backslash\mathcal{J}^K_{i,j}\backslash(i,j)\right\}}u_{i',j'}\left(\mathbf{x}_{(i',j')},\mathbf{x}_{-(i',j')}\right).\end{aligned} \tag{33}$$

By observing the formula, we find that the last two items on the right side of the above formula are the changes in cache benefits of the regions which are exceeding K hops away from the region (i,j) caused by changing the caching strategy of the satellite serving region (i,j), denoted as:

$$\begin{aligned}\Delta=&\sum_{(i',j')\in\left\{\mathcal{N}_k\backslash\mathcal{J}^K_{i,j}\backslash(i,j)\right\}}u_{i',j'}\left(\mathbf{x}_{(i',j')},\mathbf{x}'_{-(i',j')}\right)\\&-\sum_{(i',j')\in\left\{\mathcal{N}_k\backslash\mathcal{J}^K_{i,j}\backslash(i,j)\right\}}u_{i',j'}\left(\mathbf{x}_{(i',j')},\mathbf{x}_{-(i',j')}\right).\end{aligned} \tag{34}$$

However, the cache benefits of satellites in these regions do not change with the change of the satellite cache in region (i,j), because users in regions $\{\mathcal{N}_k\backslash\mathcal{J}^K_{i,j}\backslash(i,j)\}$ will not go to the satellites in the region (i,j) to get any contents. Otherwise, for these users, retrieving content from the satellite in region (i,j) consumes more time than the cloud server. Hence, we have $\Delta=0$ and the cache benefits change in (33) can be derived as:

$$\begin{aligned}&\Phi\left(\mathbf{x}'_{(i,j)},\mathbf{x}_{-(i,j)}\right)-\Phi\left(\mathbf{x}_{(i,j)},\mathbf{x}_{-(i,j)}\right)\\&=\left\{u_{i,j}\left(\mathbf{x}'_{(i,j)},\mathbf{x}_{-(i,j)}\right)-u_{i,j}\left(\mathbf{x}_{(i,j)},\mathbf{x}_{-(i,j)}\right)\right\}\\&+\sum_{(i',j')\in\mathcal{J}^k_{i,j}}u_{i',j'}\left(\mathbf{x}_{(i',j')},\mathbf{x}'_{-(i',j')}\right)\\&-\sum_{(i',j')\in\mathcal{J}^k_{i,j}}u_{i',j'}\left(\mathbf{x}_{(i',j')},\mathbf{x}_{-(i',j')}\right).\end{aligned} \tag{35}$$

At the same time, when the caching strategy of the satellite in the region (i,j) changes from $\mathbf{x}_{(i,j)}$ to $\mathbf{x}'_{(i,j)}$, the change of its utility function is:

$$\begin{aligned} &U_{i,j}\left(\mathbf{x}'_{(i,j)},\mathbf{x}_{-(i,j)}\right)-U_{i,j}\left(\mathbf{x}_{(i,j)},\mathbf{x}_{-(i,j)}\right)\\ &=u_{i,j}\left(\mathbf{x}'_{(i,j)},\mathbf{x}_{-(i,j)}\right)+\sum_{(i',j')\in\mathcal{J}^k_{i,j}}u_{i',j'}\left(\mathbf{x}_{(i',j')},\mathbf{x}'_{-(i',j')}\right)\\ &-u_{i,j}\left(\mathbf{x}_{(i,j)},\mathbf{x}_{-(i,j)}\right)-\sum_{(i',j')\in\mathcal{J}^k_{i,j}}u_{i',j'}\left(\mathbf{x}_{(i',j')},\mathbf{x}_{-(i',j')}\right), \end{aligned} \tag{36}$$

which is equal to the change of potential function denoted as (35).

Based on Definition 2, the game $\mathcal{G}_k$ can be judged as a potential game, and the proof is completed.

REFERENCES

[1] B. Di, L. Song, Y. Li, and H. V. Poor, "Ultra-dense LEO: Integration of satellite access networks into 5G and beyond," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 62–69, Apr. 2019.

[2] H. Guo, J. Li, J. Liu, N. Tian, and N. Kato, "A survey on space-air-ground-sea integrated network security in 6G," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 1, pp. 53–87, Firstquarter 2022.

[3] 3GPP, "NR NTN (non-terrestrial networks) enhancements," Technical Report TR 38.801; 3GPP: Sophia Antipolis, France, 2023.

[4] S. Zhang, D. Zhu, and Y. Wang, "A survey on space-aerial-terrestrial integrated 5G networks," *Comput. Netw.*, vol. 174, 2020, Art. no. 107212.

[5] J. Li, H. Lu, K. Xue, and Y. Zhang, "Temporal netgrid model-based dynamic routing in large-scale small satellite networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 6009–6021, Jun. 2019.

[6] C. Zhou et al., "Deep reinforcement learning for delay-oriented IoT task scheduling in SAGIN," *IEEE Trans. Wireless Commun.*, vol. 20, no. 2, pp. 911–925, Feb. 2021.

[7] "The mobile economy 2023," 2023. [Online]. Available: https://www.gsma.com/mobileeconomy/wp-content/uploads/2023/03/270223-The-Mobile-Economy-2023.pdf

[8] Y. Yang, T. Song, W. Yuan, and J. An, "Towards reliable and efficient data retrieving in ICN-based satellite networks," *J. Netw. Comput. Appl.*, vol. 179, 2021, Art. no. 102982.

[9] D. Han, W. Liao, H. Peng, H. Wu, W. Wu, and X. Shen, "Joint cache placement and cooperative multicast beamforming in integrated satellite-terrestrial networks," *IEEE Trans. Veh. Technol.*, vol. 71, no. 3, pp. 3131–3143, Mar. 2022.

[10] C. Jiang and Z. Li, "Decreasing Big Data application latency in satellite link by caching and peer selection," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 4, pp. 2555–2565, Oct.–Dec. 2020.

[11] K. An, Y. Li, X. Yan, and T. Liang, "On the performance of cache-enabled hybrid satellite-terrestrial relay networks," *IEEE Wireless Commun. Lett.*, vol. 8, no. 5, pp. 1506–1509, Oct. 2019.

[12] D. Jiang et al., "QoE-aware efficient content distribution scheme for satellite-terrestrial networks," *IEEE Trans. Mobile Comput.*, vol. 22, no. 1, pp. 443–458, Jan. 2023.

[13] H. Zhang, J. Xu, X. Liu, K. Long, and V. C. Leung, "Joint optimization of caching placement and power allocation in virtualized satellite-terrestrial network," *IEEE Trans. Wireless Commun.*, vol. 22, no. 11, pp. 7932–7943, Nov. 2023.

[14] X. Zhu, C. Jiang, L. Kuang, and Z. Zhao, "Cooperative multilayer edge caching in integrated satellite-terrestrial networks," *IEEE Trans. Wireless Commun.*, vol. 21, no. 5, pp. 2924–2937, May 2022.

[15] J. Li, K. Xue, D. S. Wei, J. Liu, and Y. Zhang, "Energy efficiency and traffic offloading optimization in integrated satellite/terrestrial radio access networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2367–2381, Apr. 2020.

[16] L. Galluccio, G. Morabito, and S. Palazzo, "Caching in information-centric satellite networks," in *Proc. IEEE Int. Conf. Commun.*, 2012, pp. 3306–3310.

[17] T. De Cola and A. Blanco, "ICN-based protocol architectures for next-generation backhauling over satellite," in *Proc. IEEE Int. Conf. Commun.*, 2017, pp. 1–6.

[18] J. Li, K. Xue, J. Liu, Y. Zhang, and Y. Fang, "An ICN/SDN-based network architecture and efficient content retrieval for future satellite-terrestrial integrated networks," *IEEE Netw.*, vol. 34, no. 1, pp. 188–195, Jan./Feb. 2020.

[19] J. Tang, J. Li, L. Zhang, K. Xue, Q. Sun, and J. Lu, "Content-aware routing based on cached content prediction in satellite networks," in *Proc. IEEE Glob. Commun. Conf.*, 2022, pp. 6541–6546.

[20] R. Xu et al., "A hybrid caching strategy for information-centric satellite networks based on node classification and popular content awareness," *Comput. Commun.*, vol. 197, pp. 186–198, 2023.

[21] M. He, C. Zhou, H. Wu, and X. S. Shen, "Learning-based cache placement and content delivery for satellite-terrestrial integrated networks," in *Proc. IEEE Glob. Commun. Conf.*, 2021, pp. 1–6.

[22] Y. Ma and A. Jamalipour, "A cooperative cache-based content delivery framework for intermittently connected mobile ad hoc networks," *IEEE Trans. Wireless Commun.*, vol. 9, no. 1, pp. 366–373, Jan. 2010.

[23] W. Jiang, G. Feng, and S. Qin, "Optimal cooperative content caching and delivery policy for heterogeneous cellular networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 5, pp. 1382–1393, May 2017.

[24] 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. Technol.*, vol. 68, no. 4, pp. 3890–3904, Apr. 2019.

[25] X. Ma, A. Zhou, S. Zhang, and S. Wang, "Cooperative service caching and workload scheduling in mobile edge computing," in *Proc. IEEE Conf. Comput. Commun.*, 2020, pp. 2076–2085.

[26] R. Liu, M. Sheng, K.-S. Lui, X. Wang, D. Zhou, and Y. Wang, "Capacity of two-layered satellite networks," *Wireless Netw.*, vol. 23, no. 8, pp. 2651–2669, 2017.

[27] E. Ekici, I. F. Akyildiz, and M. D. Bender, "A distributed routing algorithm for datagram traffic in LEO satellite networks," *IEEE/ACM Trans. Netw.*, vol. 9, no. 2, pp. 137–147, Apr. 2001.

[28] J. Li, K. Xue, J. Liu, and Y. Zhang, "A user-centric handover scheme for ultra-dense LEO satellite networks," *IEEE Wireless Commun. Lett.*, vol. 9, no. 11, pp. 1904–1908, Nov. 2020.

[29] M. E. Newman, "Power laws, pareto distributions and Zipf's law," *Contemporary Phys.*, vol. 46, no. 5, pp. 323–351, 2005.

[30] P. Yang, N. Zhang, S. Zhang, L. Yu, J. Zhang, and X. Shen, "Content popularity prediction towards location-aware mobile edge caching," *IEEE Trans. Multimedia*, vol. 21, no. 4, pp. 915–929, Apr. 2019.

[31] G. Smith and F. Campbell, "A critique of some ridge regression methods," *J. Amer. Stat. Assoc.*, vol. 75, no. 369, pp. 74–81, 1980.

[32] J. Vegelius, S. Janson, and F. Johansson, "Measures of similarity between distributions," *Qual. Quantity*, vol. 20, no. 4, pp. 437–441, 1986.

[33] Y. Zhang, Y. Xu, Q. Wu, X. Liu, K. Yao, and A. Anpalagan, "A game-theoretic approach for optimal distributed cooperative hybrid caching in D2D networks," *IEEE Wireless Commun. Lett.*, vol. 7, no. 3, pp. 324–327, Jun. 2018.

[34] Z. Hu, Z. Zheng, T. Wang, L. Song, and X. Li, "Game theoretic approaches for wireless proactive caching," *IEEE Commun. Mag.*, vol. 54, no. 8, pp. 37–43, Aug. 2016.

[35] S. Chien and A. Sinclair, "Convergence to approximate Nash equilibria in congestion games," *Games Econ. Behav.*, vol. 71, no. 2, pp. 315–327, 2011.

[36] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *ACM Trans. Interactive Intell. Syst.*, vol. 5, no. 4, pp. 1–19, 2015.

[37] N. Garg, M. Sellathurai, V. Bhatia, B. Bharath, and T. Ratnarajah, "Online content popularity prediction and learning in wireless edge caching," *IEEE Trans. Commun.*, vol. 68, no. 2, pp. 1087–1100, Feb. 2020.

[38] B. Chen and C. Yang, "Caching policy for cache-enabled D2D communications by learning user preference," *IEEE Trans. Commun.*, vol. 66, no. 12, pp. 6586–6601, Dec. 2018.

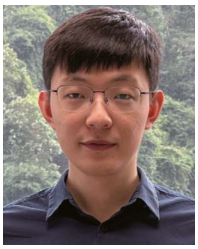
[39] D. Yu, T. Wu, C. Liu, and D. Wang, "Joint content caching and recommendation in opportunistic mobile networks through deep reinforcement learning and broad learning," *IEEE Trans. Serv. Comput.*, vol. 16, no. 4, pp. 2727–2741, Jul./Aug. 2023.

[40] J. Radtke, C. Kebschull, and E. Stoll, "Interactions of the space debris environment with mega constellations–using the example of the oneweb constellation," *Acta Astronautica*, vol. 131, pp. 55–68, 2017.

[41] C. Bernardini, T. Silverston, and O. Festor, "MPC: Popularity-based caching strategy for content centric networks," in *Proc. IEEE Int. Conf. Commun.*, 2013, pp. 3619–3623.

Jin Tang received the bachelor's degree in communication engineering from Beijing Jiao Tong University, Beijing, China, in 2021. He is currently working toward the graduation degree with the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei, China. His research interests include satellite-terrestrial integrated network, content delivery, and edge caching.

Jian Li (Senior Member, IEEE) received the bachelor's degree from the Department of Electronics and Information Engineering, Anhui University, Hefei, China, in 2015, and the doctor's degree from the Department of Electronic Engineering and Information Science, University of Science and Technology of China (USTC), Hefei, in 2020. From 2019 to 2020, he was a Visiting Scholar with the Department of Electronic and Computer Engineering, University of Florida, Gainesville, FL, USA. From 2020 to 2022, he was a Postdoctoral Researcher with the School of Cyber Science and Technology, USTC. He is currently an Associate Researcher with the School of Cyber Science and Technology, USTC. His research interests include quantum networks wireless networks, and next-generation Internet. He is the Editor of *China Communications*.



Xianhao Chen (Member, IEEE) received the B.Eng. degree from Southwest Jiaotong University, Chengdu, China, in 2017, and the Ph.D. degree from the University of Florida, Gainesville, FL, USA, in 2022. He is currently an Assistant Professor with the Department of Electrical and Electronic Engineering, University of Hong Kong, Hong Kong. His research interests include wireless networking and machine learning.

Kaiping Xue (Senior Member, IEEE) received the bachelor's degree from the Department of Information Security, University of Science and Technology of China (USTC), Hefei, China, in 2003 and the doctor's degree from the Department of Electronic Engineering and Information Science (EEIS), USTC, in 2007. From 2012 to 2013, he was a Postdoctoral Researcher with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL, USA. He is currently a Professor with the School of Cyber Science and Technology, USTC. He is also the Director of Network and Information Center, USTC. His research interests include next-generation Internet architecture design, transmission optimization, and network security. His work won best paper awards in IEEE MSN 2017 and IEEE HotICN 2019, Best Paper Honorable Mention in ACM CCS 2022, Best Paper Runner-Up Award in IEEE MASS 2018, and best track paper in MSN 2020. He serves on the Editorial Board of several journals, including IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, and IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT. He was also the (Lead) Guest Editor of many reputed journals/magazines, including IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, *IEEE Communications Magazine*, and IEEE NETWORK. He is an IET Fellow.

Lan Zhang (Member, IEEE) received the B.E. and M.S. degrees from the University of Electronic Science and Technology of China, Chengdu, China, in 2013 and 2016, respectively, and the Ph.D. degree from the University of Florida, Gainesville, FL, USA, in 2020. Since 2024, she has been a tenure-track Assistant Professor with the Department of Electrical and Computer Engineering, Clemson University, Clemson, SC, USA. From 2020 to 2023, she was an Assistant Professor with the Department of Electrical and Computer Engineering, Michigan Technological University, Houghton, MI, USA. Her research interests include wireless communications, distributed machine learning, and cybersecurity for various Internet-of-Things applications.

Qibin Sun (Fellow, IEEE) received the Ph.D. degree from the Department of Electronic Engineering and Information Science, University of Science and Technology of China (USTC), Hefei, China, in 1997. From 1996 to 2007, he was with the Institute for Infocomm Research, Singapore, where he was responsible for industrial as well as academic research projects in the area of media security, and image and video analysis. He was the Head of Delegates of Singapore in ISO/IEC SC29 WG1(JPEG). He was with Columbia University, New York, NY, USA, during 2000–2001 as a Research Scientist. He is currently a Professor with the School of Cyber Security, USTC. His research interests include multimedia security, and network intelligence and security. He led the effort to successfully bring the robust image authentication technology into ISO JPEG2000 standard Part 8 (Security). He has authored or coauthored more than 120 papers in international journals and conferences.

Jun Lu received the bachelor's degree from Southeast University, Nanjing, China, in 1985, and the master's degree from the Department of Electronic Engineering and Information Science (EEIS), University of Science and Technology of China (USTC), Hefei, China, in 1988. He is currently a Professor with the Department of EEIS, USTC. His research interests include theoretical research and system development in the field of integrated electronic information systems. He is an Academician of the Chinese Academy of Engineering.