



Cooperative Caching Strategy Based on Two-Layer Caching Model for Remote Sensing Satellite Networks

Rui Xu^{1,2,3}, Xiaoqiang Di^{1,3,4,*}, Hao Luo¹, Hui Qi^{1,3}, Xiongwen He⁵ and Wenping Lei⁶

¹School of Computer Science and Technology, Changchun University of Science and Technology, Changchun, 130022, China

²School of Information Engineering, Henan Institute of Science and Technology, Xinxiang, 453003, China

³Jilin Province Key Laboratory of Network and Information Security, Changchun University of Science and Technology, Changchun, 130022, China

⁴Information Center, Changchun University of Science and Technology, Changchun, 130022, China

⁵Beijing Institute of Spacecraft System Engineering, Beijing, 100094, China

⁶Beijing Institute of Space Mechanic and Electricity, Beijing, 100094, China

*Corresponding Author: Xiaoqiang Di. Email: dixiaoqiang@cust.edu.cn

Received: 21 October 2022; Accepted: 06 January 2023

Abstract: In Information Centric Networking (ICN) where content is the object of exchange, in-network caching is a unique functional feature with the ability to handle data storage and distribution in remote sensing satellite networks. Setting up cache space at any node enables users to access data nearby, thus relieving the processing pressure on the servers. However, the existing caching strategies still suffer from the lack of global planning of cache contents and low utilization of cache resources due to the lack of fine-grained division of cache contents. To address the issues mentioned, a cooperative caching strategy (CSTL) for remote sensing satellite networks based on a two-layer caching model is proposed. The two-layer caching model is constructed by setting up separate cache spaces in the satellite network and the ground station. Probabilistic caching of popular contents in the region at the ground station to reduce the access delay of users. A content classification method based on hierarchical division is proposed in the satellite network, and differential probabilistic caching is employed for different levels of content. The cached content is also dynamically adjusted by analyzing the subsequent changes in the popularity of the cached content. In the two-layer caching model, ground stations and satellite networks collaboratively cache to achieve global planning of cache contents, rationalize the utilization of cache resources, and reduce the propagation delay of remote sensing data. Simulation results show that the CSTL strategy not only has a high cache hit ratio compared with other caching strategies but also effectively reduces user request delay and server load, which satisfies the timeliness requirement of remote sensing data transmission.

Keywords: Information centric networking; caching strategy; two-layer caching model; hierarchical division



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1 Introduction

With their advantages of global coverage, low launch cost, and low propagation delay, low-orbit remote sensing satellite networks are highly favored in many fields such as weather forecasting, remote sensing mapping, disaster monitoring, and early warning, which provide global spatial data and gradually become the best choice for global data collection and dissemination in real time [1]. In recent years, there are some successful cases in the construction and operation of low-orbit remote sensing satellite networks in various countries. For example, the Russian Space Group is building a global remote sensing satellite network called Gosudarevo Oko [2]. Planet Labs has built a global high-resolution, high-frequency, full-coverage remote sensing satellite system [3]. Maxar has built and operated commercial remote sensing satellites such as GeoEye and WorldView [4]. Chinese Gaojing-1, Gaofen-1, and JiLin-1 are successfully formed into a network and officially operated [5]. Currently, the remote sensing satellite industry enters a period of rapid development, basically completing the construction of high-resolution earth observation systems, which would be widely applied in different industries such as meteorology, ocean, agriculture, forestry, mapping, and early warning.

With the continuous updating of the on-board hardware technology, new on-board processors with high performance and low power consumption have emerged, whose increasing performance enables the compression and slicing of remote sensing data on board [6]. It is foreseeable that remote sensing satellites will assume the role of network service providers in the satellite Internet in the future, where users could access remote sensing services anytime and anywhere. To satisfy the rapidly growing application demands of users, the quantity and type of remotely sensed data generated on the satellites are increasing, which makes the data acquisition capacity of remote sensing satellites exceed the load capacity of remote sensing satellite data transmission system, leading to the inability of this system to achieve real-time transmission of remotely sensed data and the users' requirements for timeliness of information acquisition cannot be satisfied [7]. Moreover, due to the high-speed movement of Low Earth Orbit (LEO) satellites, the inter-satellite link is often interrupted periodically, which results in excessive repetitive data transmission and increased propagation delay [8]. Two urgent challenges are to overcome the large spatial span of nodes and links in satellite networks and to satisfy users' demand for timeliness of data transmission, which requires a new network architecture with simplicity and efficiency. ICN fundamentally decouples the identity and physical address of nodes in the network, which is information-centric and well-suited for highly dynamic networks such as satellite networks, mobile ad hoc networks, and vehicular networks [9]. In-network caching is an important factor in improving network performance in ICN-based networks, where each ICN node has a caching function and stores data according to the established caching strategy. When the same content request occurs subsequently, the caching node can transmit data instead of the source server [10]. The proximity of data acquisition reduces the distance between users and content sources, reducing data propagation delay and bandwidth consumption during satellite communication, which is especially important in remote sensing satellite networks.

In remote sensing satellite networks, exploring the optimal caching strategy is an effective means to improve network performance and is an important research topic. Therefore, it is proposed a CSTL caching strategy for remote sensing satellite networks. Equipping the satellite side and the ground station with cache space separately constitutes a two-layer caching model. Among them, the ground station serves the users in the region and focuses on caching the data requested by users in the region with high frequency. The satellite side serves the entire coverage area and designs the caching scheme from a global perspective. The main contributions and innovative work of this paper are as follows.

- To overcome the lack of global planning of cached content, a two-layer caching model is proposed in this paper, where separate cache spaces are set up at the ground station and the satellite network. The independent caching scheme based on the awareness of local popularity is implemented at the ground station. The probabilistic caching scheme based on hierarchical division is accomplished in the satellite network. The ground station and the satellite network collaborate to realize the awareness of popular contents and the hierarchical division of contents, provide differentiated caching decisions for different levels of contents, and achieve the reasonable allocation of cache space.
- In the satellite network, the cache contents are placed in fixed positions which tends to reduce the utilization of cache resources. Therefore, the cached content is dynamically pushed to a reasonable location based on the subsequent change in popularity of the cached content to maximize the utilization of node space resources and the utility of the caching system.
- The Iridium constellation is selected as a simulation scenario to compare the performance of the CSTL strategy with six existing common caching strategies, and further analyze the effects of content request hotness, the number of contents, and the request rate on the caching strategy. The simulation experiments show that the CSTL strategy improves the cache hit rate, reduces the request delay and server load, and improves the performance of the satellite network compared with other strategies.

The rest of this article is structured as follows. The related works including on-path and off-path caching strategies are presented in Section 2. Section 3 constructs the scenario model in detail. Then, the CSTL caching strategy is proposed with a detailed analysis in Section 4. The experimental results are provided in Section 5 to demonstrate the CSTL strategy. Lastly, Section 6 concludes this work.

2 Related Work

The caching strategy is one of the research hotspots in ICN, which is the cornerstone to achieve fast content processing and distribution in the network. Designing an efficient caching strategy becomes a key issue to fully utilize the caching resources and increase the performance of the caching system. To this end, scholars have conducted a series of studies to classify caching strategies from the perspective of data transmission, including on-path strategies and off-path strategies [11].

The on-path caching strategy focuses on improving the efficiency of the caching system by obtaining local information from the perspectives of network topology, mathematical statistics, and complex networks, and studying the reasonable selection of caching nodes and caching contents on the return path of data packets. The classical on-path caching strategies in ICN include LCE (leave copy everywhere) [12], LCD (leave copy down) [13], and Prob (copy with probability) [14]. Among them, the LCE strategy is the default caching strategy for ICN, which is characterized by the fact that all router nodes on the path cache all content copies indiscriminately during data backhaul, which tends to cause a large amount of content redundancy. The LCD strategy makes up for the shortcomings of the LCE strategy by storing content objects only at the next-hop of the node where they are located to bring the cached content gradually closer to the edge of the network. Prob strategy is to make caching decisions on the content on the transmission path with a fixed probability at the router nodes, which helps to achieve the diversity of cached content but causes uncontrollable caching probability. Ren et al. present a distributed caching strategy (MAGIC) that utilizes content popularity and hop count reduction to represent cache gain and caches content copies on the node with the largest cache gain, thereby reducing bandwidth consumption [15]. Gill et al. propose an auction caching model (BidCache) to improve the caching performance by comparing the bids of all nodes on the path and

placing the content in the highest bidder node on the transmission path [16]. With the existing on-path caching strategy, each node is limited to collaborating with only the upstream or downstream nodes on the transmission path due to the inability to obtain the overall picture of the network on time, which limits the network performance [17].

The placement of cached content in the off-path caching strategies is not restricted by the transmission path but is calculated according to the respective cache strategy. Chai et al. suggest a caching strategy (Betw) based on the centrality algorithm to cache data on nodes with large BC values, which allows users to access content quickly due to the fact that these nodes are mostly at hub locations in the network [18]. Wu et al. propose a probabilistic caching strategy (MBP) for maximizing revenue, which obtains the caching probability of content based on content popularity and content placement revenue, where content with high content popularity and high caching revenue is cached with a higher probability to optimize the performance of the caching system [19]. Zheng et al. propose a cache placement strategy (BEP) by combining content information and topology information to place popular content at nodes with high importance and design the filtering effect of cache on requests, thus reducing the redundancy of cached content in the network [20]. Amadeo et al. propose a novel caching strategy (PaCC) by accounting for the content popularity and proximity to the consumers. The PaCC strategy places the most popular content identified in the available edge nodes, which effectively limits the data retrieval delay and the exchanged data traffic [21]. The existing off-path strategies normally require a lot of communication overhead to collect the status information of related nodes such as node capacity, cache location, request distribution, etc., which limits the scalability of the network [22].

To achieve efficient transmission between nodes in satellite networks, scholars have also conducted a series of research works on the in-network caching technology in satellite networks for its advantages in bandwidth saving and delay reduction. Liu et al. propose a novel exchange-stable matching algorithm to achieve optimal placement of cache contents for the scenario of multiple satellites jointly serving multiple ground terminals [23]. Qiu et al. present an SDN-based satellite-ground network model for the dynamic management of network, cache, and computational resources, and a novel deep Q-learning learning method is implemented to solve the problem of optimal resource allocation [24]. Yang et al. propose a novel network caching scheme (TCSC) to improve the efficiency of file distribution in the network by filtering out candidate cache nodes with minimum time-evolving coverage set algorithm and obtaining file copies from selected cache nodes with minimum delay [25]. Li et al. suggest a satellite network model based on ICN/SDN to provide flexible management and efficient content retrieval while designing a simple and effective content retrieval scheme that further reduces traffic consumption with in-network caching resources [26]. The existing caching strategy in the satellite network enhances the efficiency of content distribution by rationalizing the placement of cached content so that users can access content from neighboring cache nodes [27,28].

For remote sensing satellite networks with higher timeliness requirements, the existing caching strategy still has some urgent problems to be solved. Firstly, cache content placement lacks global planning, with redundant content and cold content taking up valuable cache resources, resulting in low utilization of cache space. Secondly, existing research is devoted to improving the cache hit ratio, ignoring the user's demand for content access delay. Finally, existing research ignores the timeliness characteristics of content requests, leading to the impossibility of the caching system to dynamically adjust the cached content. Therefore, a collaborative two-layer caching model for the satellite network and the ground station is proposed to plan the caching resources of the satellite network and the ground station rationally. A probabilistic caching scheme oriented to the local region is designed at the ground station to provide more efficient services when content is requested. Rational placement and dynamic adjustment of cached contents are achieved in satellite networks through a probabilistic

caching scheme based on hierarchical division and a dynamic adjustment scheme of cached contents to maximize the effectiveness of the caching system.

3 Scene Description

Consider an application scenario of a remote sensing satellite network where satellite nodes provide satellite application services to terrestrial users. To improve readability, the notations used in this paper are listed in [Table 1](#).

Table 1: Definition of variables

Variable	Definition
gs_i	Ground stations
G	Number of ground stations
ut_i	Terrestrial user terminals
U	Number of terrestrial user terminals
sn_i	Satellite nodes
S	Number of satellite nodes
fl_i	Content stored in the source server
F	Number of contents stored in the source server
ts_i	Time slots
T	Number of time slots
$\rho_{cur}(c)$	Content popularity of content block c in time slot cur
$req_{cur}(c)$	Number of times that content block c is requested in time slot cur
θ	Weight function
β	Decay factor
$\delta_{cur}(c)$	Caching probability of content block c at the ground station in time slot cur
$hop_{max}(c)$	Maximum number of hops from the user terminal to the ground station for requesting content block c
hop_{max}	Maximum number of hops between the user terminal and the ground station
cc_i	Cache space of satellite node i
cc_{total}	The sum of the cache space of the nodes on the path from the requester to the server
k	Number of router nodes on the path from the requester to the server
R_i	Content based on hierarchical division
p	Hierarchical information about the content storable by this node
$\eta(c, p)$	Cache probability of content block c at satellite node sn_p
$\rho_{threshold}$	The predefined threshold for content popularity
$h_counts(i)$	Number of user requests hit by the cache nodes in time slot i
$H_counts(i)$	Number of content requests in the satellite network in time slot i
$Delay(i)$	Average time between sending requests and receiving data in time slot i
$SL(i)$	Number of user requests processed by the server in time slot i
$time(i)$	Running time of the time slot i
α	Zipf distribution parameter

A remote sensing satellite network architecture with a two-layer cache model is proposed, as shown in Fig. 1. The network model consists of a terrestrial gateway to the Internet, a satellite network with an inter-satellite link, multiple ground stations, and terrestrial user terminals. The satellites are interconnected through inter-satellite links to form a satellite network, which has a wide coverage area and provides backhaul transmission of data for the ground stations within the coverage area. Each ground station provides satellite application services for the user terminals in the region.

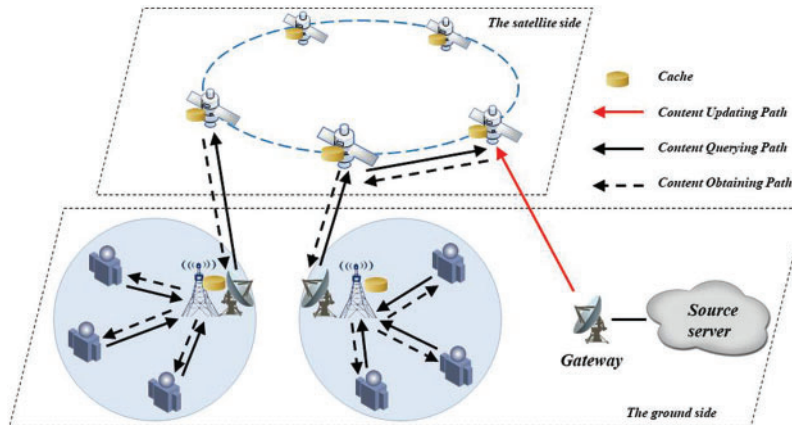


Figure 1: Remote sensing satellite networks architecture with a two-layer caching model

In the ICN-based communication model, all satellite nodes and ground stations are equipped with a content memory with the capability of content storage and forwarding, which collaborates to build a two-layer caching model. In the two-layer caching model, satellite nodes and ground stations decide what to cache and where to cache the content on the basis of the caching strategy. Specifically, the numbers of ground stations and user terminals on the ground are G and U , respectively, then the set of ground stations is denoted as $GS = \{gs_i | i = 1, 2, \dots, G\}$, and the set of users is denoted as $UT = \{ut_i | i = 1, 2, \dots, U\}$. There are S satellite nodes in the satellite network, then the set of satellite nodes is denoted as $SN = \{sn_i | i = 1, 2, \dots, S\}$. The set of files provided by the content source server is denoted as $FL = \{fl_i | i = 1, 2, \dots, F\}$. Moreover, due to the satellite nodes flying at high speed around a predetermined orbit, the connection relationship between the accessing satellite and the ground station is also changing over time. Therefore, the method of constructing time slots in the literature [29] is adopted to divide the operational cycle of the remote sensing satellite network into a series of time slots, denoted as $TS = \{ts_i | i = 1, 2, \dots, T\}$. In the time slot, the inter-satellite links in the network remain stable and unchanged, which is a common treatment in related studies [25].

In ICN-based networks, there are two data formats interest packets and data packets included. Interest packets are content requests from users (requesters), and data packets are contents that are reverse transmitted to users by content service nodes (cache nodes or content source servers) in response to requests. The ICN focuses on the information itself and drives the request and response behaviors of various information to operate in the network by coordinating the caching and distribution of related content. Moreover, in the proposed ICN-based remote sensing satellite network architecture, the ground stations all have caching capability to cache content with high popularity in the region to satisfy the needs of users in the region. And the ground station is connected to the access satellite by means of satellite-terrestrial links. To reduce the consumption of satellite bandwidth during satellite-terrestrial interconnection, user requests in the region are aggregated at the ground station. During a fixed period, if different user terminals send requests with the same content to the ground

station, the ground station merges the same user requests and forwards only one interest packet to the access satellite. After getting the response content, the access satellite transmits a data packet back to the ground station and then transmits it to the user terminal sending the request through broadcasting. The aggregation of user requests by the ground station can effectively reduce the duplicate transmission of the same content in the satellite-terrestrial links.

The service process of the two-layer cache model is as follows. Firstly, the ground station searches in the local cache for content requests sent by user terminals in the region. If the corresponding content is cached, the user request is responded to directly. If there is no cache, the same content requests are aggregated for a fixed period, and the requests are forwarded to the access satellite. Then, a local search is performed at the access satellite. If the corresponding content is cached, it is transmitted back to the ground station via the satellite-terrestrial link. Otherwise, the access satellite forwards the user request to other satellites in the network or servers on the ground, which respond to the request and return the data packets. Finally, the data packet is cached and forwarded on the original return path according to the caching strategy.

4 Methods

The ICN-based remote sensing satellite network mainly adopts a distributed caching technology, where the contents in the network are distributed on satellite nodes and ground stations according to a predefined caching strategy. As the duration and frequency of content requests from users change, the content on each cache node changes accordingly. In this section, the memory is set at the ground stations and satellite nodes respectively, which constitutes a two-layer caching model for the satellite and ground, thus proposing a cooperative caching strategy (CSTL) for remote sensing satellite networks based on two-layer caching. Among them, the ground station mainly caches the hotspot information in the region, and the satellite network adopts a hierarchical division scheme for on-path caching. Moreover, a cached content dynamic adjustment scheme is adopted in satellite networks to promote popular content closer to users and non-popular content closer to servers, which reduces the response delay of user requests and satisfies users' demand for satellite application service timeliness on the basis of ensuring caching system efficiency and stability. In this section, the probabilistic caching scheme for ground stations is first introduced. Then the caching scheme for the hierarchical division of satellite networks is introduced. Finally, the cached content dynamic adjustment scheme for satellite networks is introduced.

4.1 Probabilistic Caching Scheme for Ground Stations

In the two-layer caching model, the ground station has two principal functions. The first function is to aggregate the same content requests from user terminals within a fixed period, merge them into a single interest packet, and then forward them to the access satellite, thus reducing the bandwidth consumption of the satellite-terrestrial link. The second function is to cache the frequently requested contents in the region, which helps user terminals to get the requested contents quickly and reduce the response delay of user requests.

To effectively utilize the limited caching resources on the ground station, the content with high popularity needs to be screened for caching to satisfy the user terminal's demand for popular content. By analyzing the characteristics of user requests and the decay law of content popularity, the popularity of content block c is derived as shown below.

$$\rho_{cur}(c) = \theta \times req_{cur}(c) + (1 - \theta) \times \left(\sum_{i=1}^{cur-1} \rho_i(c) \times \beta^{cur-i} \right) \quad (1)$$

where $\rho_{cur}(c)$ denotes the content popularity of content block c in time slot cur and $req_{cur}(c)$ denotes the number of times content block c is requested in time slot cur . θ is a weight function that takes values from 0 to 1. β is a decay factor that takes values from 0 to 1.

In this paper, there is no strict distinction between popular and non-popular content, but the content with high popularity is selected for caching based on the ranking of content popularity and the cache space size of the ground station. At the same time, to increase the diversity of the cached content and to reduce the time delay of the user in accessing the content, the content block c is probabilistically cached at the ground station according to the following equation.

$$\delta_{cur}(c) = \frac{req_{cur-1}(c)}{req_{cur-1}} \times \frac{hop_max(c)}{hop_max} \quad (2)$$

where $\delta_{cur}(c)$ represents the probability that the content block c is cached at the ground station in time slot cur , req_{cur-1} indicates the volume of content requests received at the ground station in time slot $cur-1$, $req_{cur-1}(c)$ means the total number of times that content block c is requested in time slot $cur-1$, $hop_max(c)$ represents the largest hops from the user terminal to the ground station for requesting content block c , and hop_max indicates the maximum number of hops between the user terminal and the ground station. The equation takes full account of content popularity and access delay to ensure that popular content has a greater probability of being cached at the ground station while accounting for a lower delay in user access.

4.2 Probabilistic Caching Scheme for Satellite Networks Based on Hierarchical Division

The traditional probabilistic caching algorithm (Prob) is to cache the data with a constant probability on the path of data packet return [14]. Although this algorithm can reduce the redundancy of cached contents and improve the diversity of cached contents, using the same caching probability at cache nodes to cache contents with different degrees of importance easily causes the content of high importance to be missed in different degrees. Therefore, a hierarchical division-based probabilistic caching scheme is proposed to prioritize the content requested by users based on the path information such as hops statistics and the caching capacity of routers during the forwarding of interest packets in satellite networks. Thereby, contents at different levels can be cached with different probabilities during data packet backhaul to achieve a reasonable allocation of cache space.

The two types of data formats in ICN include interest packets and data packets to achieve data forwarding [30]. The requester sends a request for uniquely named content, and the ICN router uses the FIB table to forward the interest packet hop-by-hop until it hits the cache node or server. Thereafter, the cache node or server delivers the corresponding data packet to the requestor along the same path in reverse. In the satellite network, the process of sending interest packets from the ground station to request the corresponding content from the content source server is shown in Fig. 2. The interest packet passes through sn_1, sn_2, \dots, sn_k in turn along the way, totaling k satellite nodes. After hop-by-hop forwarding by the routers on each satellite node, the interest packets finally get a response at the content source server. Meanwhile, the interest packets count the number of hops experienced and accumulate the cache space of the routers along the way, while recording the hop count in the FIB entry of each router.

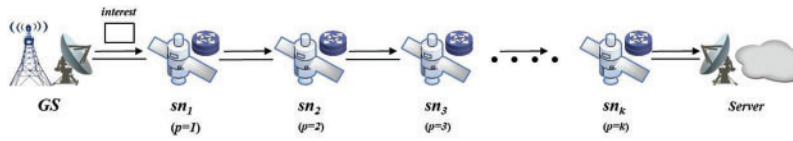


Figure 2: Diagram of interest packet forwarding

The cache space of the on-board routers that the interest packets pass through along the way may be different, which can be expressed as $CC = \{cc_i | i = 1, 2, \dots, k\}$. The interest packets cumulative the cache space of the routers on the transmission path, and the total cache capacity of the routers can be calculated as cc_total when the interest packets arrive at the server. The number of hops experienced by the interest packets along the path from the requester to the content source server is $k+1$. Then, the number of routing nodes on the forwarding path of the interest packets is adopted as the criterion for dividing the hierarchy, which divides the content into k layers. All the contents in the network are sorted in descending order according to Eq. (1), and the contents are divided into k levels according to the rules in the following table, as shown in Table 2.

Table 2: Content classification based on hierarchical division

Number	Range of rankings
R_1	$\left[1, \left\lfloor F \cdot \frac{cc_1}{cc_total} \right\rfloor \right]$
R_2	$\left(\left\lfloor F \cdot \frac{cc_1}{cc_total} \right\rfloor, \left\lfloor F \cdot \frac{cc_1 + cc_2}{cc_total} \right\rfloor \right]$
\vdots	\vdots
R_i	$\left(\left\lfloor F \cdot \frac{\sum_{j=1}^{i-1} cc_j}{cc_total} \right\rfloor, \left\lfloor F \cdot \frac{\sum_{j=1}^i cc_j}{cc_total} \right\rfloor \right]$
\vdots	\vdots
R_k	$\left(\left\lfloor F \cdot \frac{\sum_{j=1}^{k-1} cc_j}{cc_total} \right\rfloor, F \right]$

After a user sends a request in ICN-based networks if the requested content is cached at a location that is closer to the user, the more efficient it is for the user to access the content [31]. Also if the content with high popularity is more widely distributed in the network, the easier it is to respond to users' requests and the higher cache value in the network [9]. Therefore, after dividing the content into layers, setting different priorities for the content of different layers and adopting differentiated placement methods helps maximize the cache value of all cache nodes. After the server receives the user request, the data packets are transmitted back following the original path, and the routers along the way are available to cache the contents at different levels according to the value of p in their FIB entries. Therefore, the content R_i is placed on the router nodes with p ranging from $[i, k]$, as shown in Fig. 3.

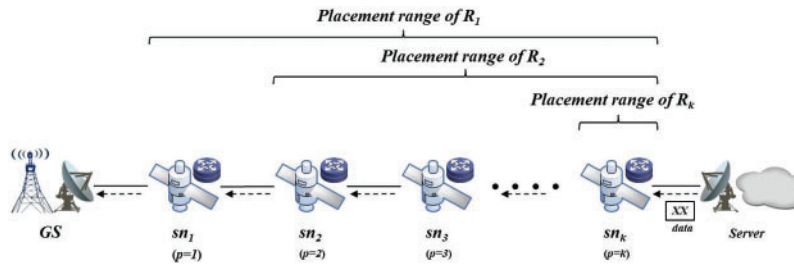


Figure 3: Diagram of data packet forwarding

Since the content with high priority is placed more widely, the traditional probabilistic caching method for it tends to cause redundancy in the cached content and reduce the utilization value of the cache space. Therefore, the probability of caching the content block c at the satellite node sn_p can be derived by the following equation.

$$\eta(c, p) = \frac{\rho_{cur}(c)}{cc_p} \times \frac{k-p}{k+1} \tag{3}$$

where $\eta(c, p)$ denotes the caching probability of content block c at satellite node sn_p , $\rho_{cur}(c)$ denotes the content popularity of content block c in time slot cur , cc_p denotes the cache space of satellite node p , $k+1$ denotes the number of hops between the requester and the server, and $k-p$ denotes the number of hops from the server to the current node, which is the number of hops saved when the content request gets a response at the cache node than the number of hops saved in getting a response from the server. The equation fully considers content popularity, access delay, and cache space to improve the probability of caching content with high popularity near the requestor, reduce the probability of caching that content near the server, and avoid redundancy of cached content.

To implement a probabilistic caching scheme based on hierarchical division during data transmission in satellite networks, the format of interest packets and data packets is fine-tuned, as shown in Fig. 4. The IntHop and CountCacheSpace fields are increased in the interest packets, where IntHop records the number of route hops forwarded by the interest packet, and CountCacheSpace accumulates the cache space of the routers on the path. DataHop and HierarchicalInformation fields are increased in the data packet, where DataHop records the route hops during data packet backhaul, and HierarchicalInformation marks the hierarchy where the content is located in the data packet.

<i>Interest Message</i>	<i>Data Message</i>
<i>Content Name</i>	<i>Content Name</i>
<i>Selector(Order Preference, Publisher Filter, Scope, ...)</i>	<i>Signature(Digest Algorithm, Witness, ...)</i>
<i>Nonce</i>	<i>Signed Info(Publisher ID, Key locator, Stale time, ...)</i>
<i>IntHop</i>	<i>Data</i>
<i>CountCacheSpace</i>	<i>DataHop</i>
	<i>HierarchicalInformation</i>

Figure 4: Extended packet format

In the ICN framework, routers communicate by maintaining content store (CS), pending interest table (PIT), and forwarding information base (FIB) [30]. Among them, CS is provided to cache the content, which exists in the storage of the router. PIT assists the data packets in following the original path back by recording the transmission path of the content request. FIB assists the content requests

in being forwarded hop by hop. When a router receives a content request, the location of the content is searched in the order of CS-PIT-FIB. In this paper, two perspectives on the processing of interest packets and data packets are presented to describe the process of implementing a hierarchical division-based probabilistic caching scheme at satellite nodes. Firstly, the interest packets record the cache space capacity and routing hops of each node along the path during the forwarding process from the requester to the server and assign the value of hop to the p -value of each satellite node. Next, the data packets carry hierarchical information R_i corresponding to their contents when they are returned along the original path. Each satellite node on the path decides whether to cache the data packets by checking the hierarchical information. The interest packet forwarding and data packet forwarding in the probabilistic caching scheme based on hierarchical division are illustrated as shown in Algorithms 1 and 2, respectively.

Algorithm 1: Interest package processing

Input:

cr content request, **IntPath** the set of all nodes on the shortest path from the user to the server

IntHop = 0;

CCS = 0; // CCS is the abbreviation of CountCacheSpace

for node j **in** the IntPath

if cr cache at node j **then**

 return a content copy of cr;

else

 IntHop = IntHop + 1;

 Set the p value of the node j to IntHop;

 CCS = CCS + cache space size of node j ;

 transmit the interest packet to the next-hop router node;

end if

end for

Algorithm 2: Data packet processing

Input:

rc requested content, R_i hierarchical information of rc, **DataPath** the set of all nodes on the shortest path from the server to the user

for node j **in** the DataPath

if the p value of the node $j \geq i$ **then**

if the cache space of $j \neq$ full **then**

 Probabilistic cache rc at node j ;

else

 cache rc at node j by LRU replacement strategy;

end if

else

 transmit the data packet to the next-hop router node;

end if

end for

4.3 *Cached Content Dynamic Adjustment Scheme for Satellite Networks*

The hierarchical division-based probabilistic caching scheme is to rationally allocate the content to storage areas in a certain time slot according to content popularity, where popular content has a greater chance of getting cached near the requester and non-popular content is more inclined to be cached near the server. However, during the working of the satellite network, the content popularity is always changing dynamically, while the location of the content is constant during a fixed period. Therefore, a dynamic adjustment scheme for cached contents is proposed, whose main goal is to dynamically adjust the placement of cached contents according to the change of the subsequent popularity of the cached contents by pushing them to a more reasonable location within the placement range, thus maximizing the utility of the caching system.

This dynamic adjustment scheme mainly includes Pushdown and Pushup processes.

Pushdown: When a content request hits a cache node, it determines whether the popularity of the content reaches a predefined threshold $\rho_{\text{threshold}}$ for content popularity. If satisfied, the content is stored in the cache space of the next hop within the placement range corresponding to it, and it is removed from the CS of the current cache node. Thus, the popular content is brought closer to the requester by this process.

Pushup: Limited by the cache node's storage space, it is not possible to cache all popular contents, so it is necessary to remove the contents with low popularity from the CS of the current node. To ensure the integrity of the data in the cache system, the removed content is stored in the cache space of the previous hop. This process is repeated until the content is transferred to the server and then completely removed.

By evaluating both Pushdown and Pushup processes, it is discovered that the predefined threshold of content popularity plays a pivotal role in cache performance. When $\rho_{\text{threshold}}$ is too small, a large number of eligible content requests are responded to in the network, then the contents are stored in the next-hop router. A large amount of content being pushed down affects the network overhead, causing frequent cache replacement in downstream routers, which in turn reduces the utilization of cached content. When $\rho_{\text{threshold}}$ is too large, some of the cached contents are continuously requested but do not qualify for Pushdown, thus failing to Pushdown, which in turn slows down the speed of pushing up the cached contents in downstream routers to the server. Moreover, those cached contents in downstream routers that are no longer popular occupy high-quality network resources for a long time, which is a great waste of the limited cache space. Therefore, $\rho_{\text{threshold}}$ should be rationally established by considering the topological characteristics of the satellite network and the request rate of the content in each time slot.

5 Simulation Results and Analysis

5.1 *Simulation Setup*

In the simulation scenario, the Iridium constellation is referenced as the satellite network configuration, which is an LEO satellite network supporting interstellar links and consists of six orbital planes with an altitude of 780 km and an orbital inclination of 86.4° , with 11 satellites evenly distributed on each orbital plane, for a total of 66 satellites [32]. Each satellite is interconnected with four neighboring satellites through inter-satellite links, including two neighboring satellites in the same orbital plane and two neighboring satellites at the nearest distance on the different orbital planes. 22 ground stations are selected in the satellite coverage area, and the document popularity in the area where each ground station is located is independent of each other. Moreover, there are 40 user terminals in the coverage

area of each earth station. Assume that there are 1000 different contents of size 1024 KB in the network. The user's request hotness follows the Zipf distribution and the request process follows the Poisson distribution. The cache capacity of each satellite node is kept consistent and much lower than the total amount of content. The total duration of the simulation is 50 s, including the first 10 s of warm-up time and the last 40 s of sampling time. Moreover, the other parameters of the simulation experiments are shown in [Table 3](#).

Table 3: Experimental parameters

Parameters	Default	Range of variation
Operation period/s	6000	—
Cache replacement strategy	LRU	—
Number of contents	1000	500~3500
Zipf distribution parameter α	1.0	0.7~1.3
Request rate/(req/s)	200	50~350
Node cache size	30	—
Link delay/ms	10	—
Link bandwidth/Mbps	100	—

The caching strategies of LCE [12], LCD [13], Prob (0.5) [14], Betw [18], BEP [20], and PaCC [21] are implemented with the open-source simulation tool ndnSIM [33]. The caching performance of each strategy is quantitatively analyzed in terms of performance indicators such as average cache hit rate, average request delay, and average server load. Moreover, the effects of Zipf parameters [34], the number of contents, and the request rate on the cache performance are explored.

5.2 Performance Evaluation Indicators

The advantages and disadvantages of a caching strategy are judged comprehensively from the following three aspects.

(1) Average cache hit ratio (ACHR)

The average cache hit ratio reflects how efficiently the cached content is utilized. The higher the cache hit ratio, the more effective the caching system is.

$$ACHR = \frac{1}{T} \sum_{i=1}^T \frac{h_counts(i)}{H_counts(i)} \quad (4)$$

where T indicates the number of time slots, $h_counts(i)$ indicates the number of user requests hit by the cache nodes in time slot i , and $H_counts(i)$ indicates the total number of content requests in the satellite network in time slot i .

(2) Average request delay (ARD)

The average request delay reflects the satisfaction level of user experience. The smaller the request delay, the faster the response time of the caching service.

$$ARD = \frac{1}{T} \sum_{i=1}^T Delay(i) \quad (5)$$

where T indicates the number of time slots, and $Delay(i)$ indicates the average time between sending requests and receiving data in time slot i .

(3) Average server load (ASL)

The average server load is a measure of the pressure on the server. If the average server load is too high, the processing capacity of the server is insufficient, which in turn affects the stable operation of the network.

$$ASL = \frac{1}{T} \sum_{i=1}^T \frac{SL(i)}{time(i)} \quad (6)$$

where T indicates the number of time slots, $SL(i)$ indicates the number of user requests processed by the server in time slot i , and $time(i)$ indicates the running time of time slot i .

5.3 Simulation Results

5.3.1 Effect of Zipf Parameter

The effect of the Zipf distribution parameter α on the cache performance is explored in this section, as shown in Fig. 5.

Fig. 5a shows that the caching effectiveness of all caching strategies improves as the Zipf parameter α grows from 0.7 to 1.3. The reason is that the user's request preference is more concentrated during the Zipf parameter α increase, resulting in a widening gap between popular and non-popular content. The CSTL, BEP, and PaCC strategies outperform the other strategies in terms of the cache hit ratio by sensing the popular content in different ways and optimizing the placement of cached content. Moreover, the caching performance of CSTL strategy improves faster than other strategies. The probabilistic caching approach of Prob strategy assists in ensuring the diversity of cached contents with better caching results compared with LCE and LCD strategies in the same experimental conditions. Fig. 5b shows that the request delay decreases for all strategies due to the increase in the percentage of user requests hitting the cache nodes. The hierarchical division-based probabilistic caching scheme in the CSTL strategy avoids frequent replacement of cached content, which therefore outperforms other strategies in the whole range. Based on the Betw strategy, the BEP and PaCC strategies select the most popular contents to be placed on the cache nodes, reducing the frequency of replacing cached contents. Compared with the LCE strategy, the Prob and LCD strategies improve the caching everywhere and place the cached content reasonably nearer to the user, which provides better caching effectiveness. Since the Betw strategy only caches content at critical nodes, it is easy to cause content overload at cache nodes and the replacement of some popular content, leading to an increase in the request delay. Fig. 5c shows that the server load of all strategies decreases significantly when the Zipf parameter α is increased. The CSTL strategy allocates popular content to cache nodes rationally and adopts a dynamic adjustment scheme for cached content, which improves the response ratio of user requests at cache nodes and outperforms other strategies in terms of caching effectiveness. The BEP and PaCC strategies place popular content on critical nodes, which promotes the caching system to function and reduces the server load. Prob, LCD, and LCE strategies do not differentiate the cached content, which leads to part of the cache space being occupied by non-popular content, which affects the caching effect. The Betw strategy has poor utilization of cache space at non-critical nodes.

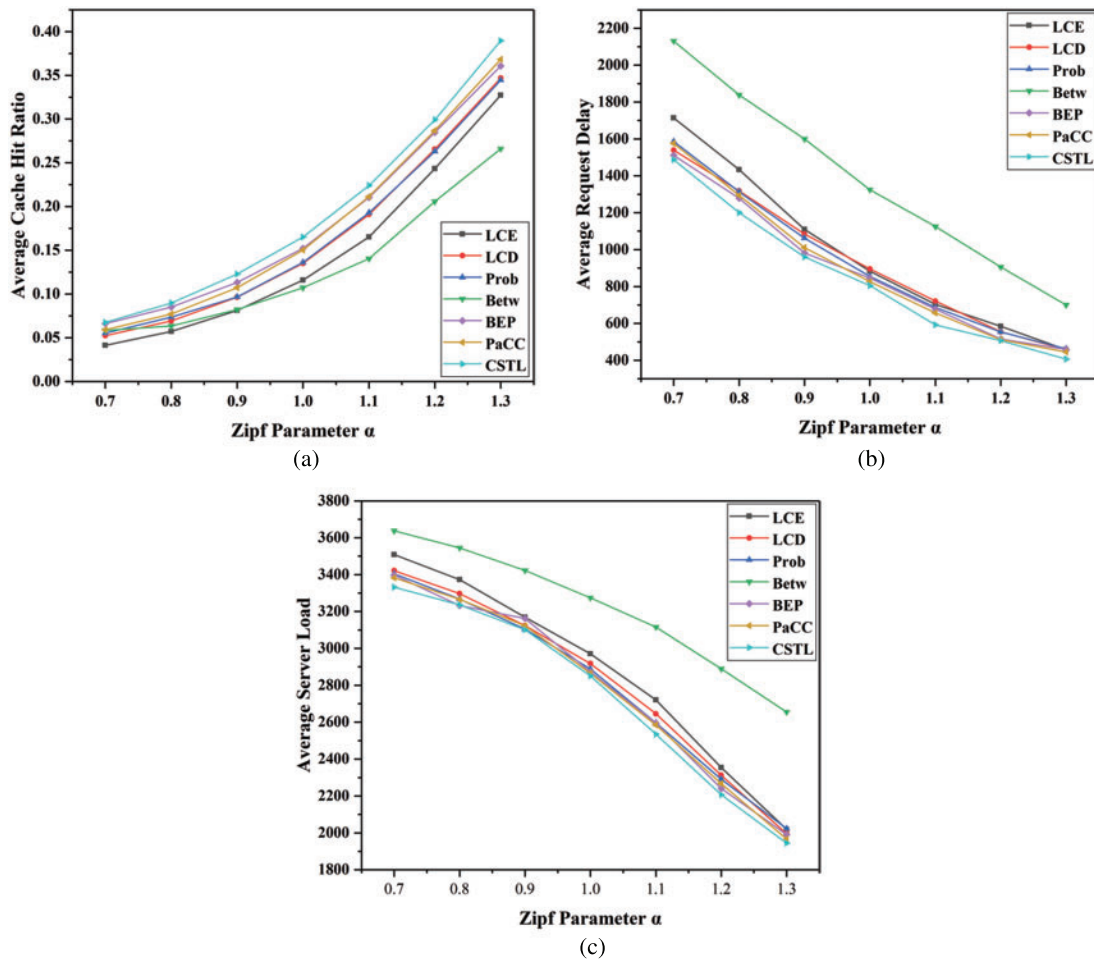


Figure 5: Effect of Zipf parameter α (a) ACHR with Zipf parameter α (b) ARD with Zipf parameter α (c) ASL with Zipf parameter α

5.3.2 Effect of Number of Contents

The practical performance of each caching strategy under the different number of contents is investigated in this section, as shown in Fig. 6.

Fig. 6a shows that the caching effectiveness of all caching strategies decreases as the number of contents grows from 500 to 3500. As the number of user requests increases and the types become more dispersed, the limited storage space of the cache nodes cannot accommodate more content copies, resulting in a decrease in the percentage of user requests hit by cache nodes. The CSTL strategy hierarchizes content based on content popularity and enhances the diversity of cached content through probabilistic caching to facilitate user requests to be responded to at cache nodes, which in turn reduces the effect. The BEP and PaCC strategies place popular content on critical nodes to avoid data redundancy on the transmission path, enabling better caching than the other caching strategies. Fig. 6b shows the variation of the average request delay for each strategy, where the request delay rises rapidly in the early stage and rises slowly in the later stages. Due to the reasonable allocation of cache space and the dynamic adjustment scheme of cached content by the CSTL strategy, which

promotes the placement of cached content at the edge of the network, the CSTL strategy provides better caching services. Cached content is overloaded in the Betw strategy, and some of the popular content is answered by the server, causing a higher delay in response to user requests. Fig. 6c shows that the server load of all strategies increases significantly when the number of contents increases. The CSTL strategy adopts a hierarchical division based on content popularity and a probabilistic caching scheme to promote cache nodes to hit user requests, thus reducing the server load and outperforming other caching strategies. The BEP and PaCC strategies take into account the topological characteristics of the cache nodes and the dynamic popularity of the content, improving the probability of the cache nodes hitting user requests and reducing the ratio of user requests forwarded to the server. The LCE, LCD, and Prob strategies have a large number of redundancies in the transmission path, resulting in an increasing percentage of user requests being forwarded to the server. The Betw strategy only exploits the cache space of critical nodes and ignores the cache space of non-critical nodes, which causes a large number of user requests to hit the server and puts tremendous pressure on the server.

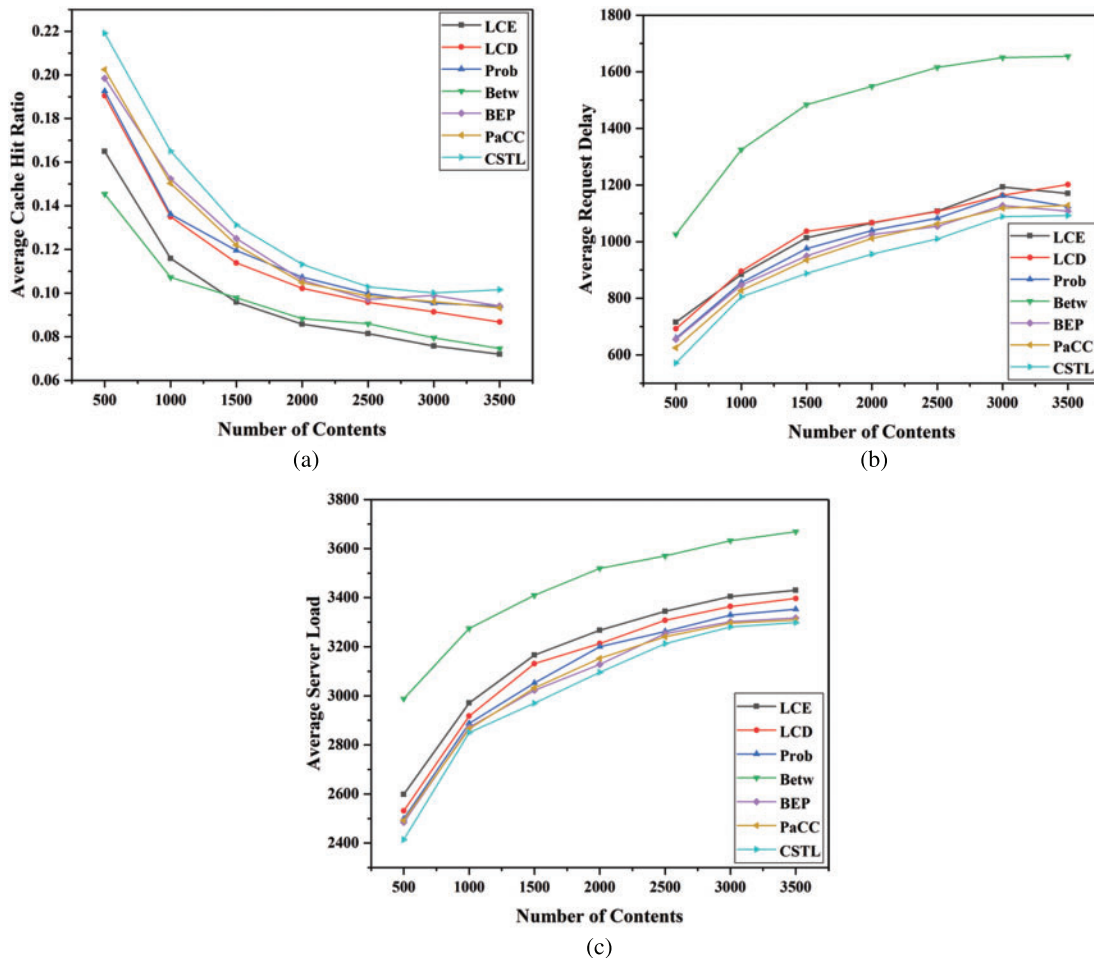


Figure 6: Effect of number of contents (a) ACHR with the number of contents (b) ARD with the number of contents (c) ASL with the number of contents

5.3.3 Effect of Request Rate

The performance of each caching strategy is compared in this section with varying request rates, as shown in Fig. 7.

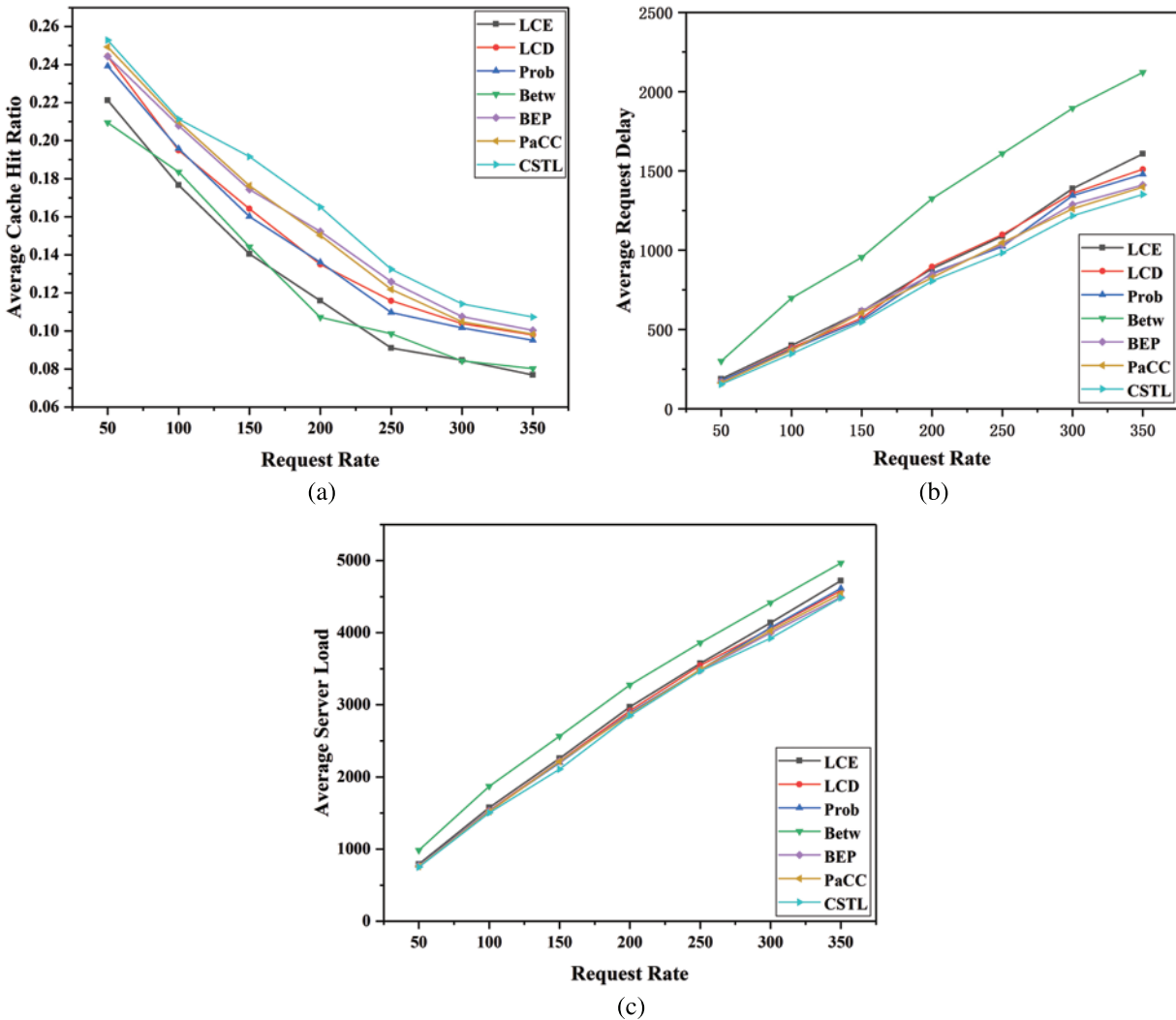


Figure 7: Effect of the request rate (a) ACHR with the request rate (b) ARD with the request rate (c) ASL with the request rate

Fig. 7a shows that the caching effectiveness of all caching strategies is decreasing as the request rate grows from 50 to 350. The main reason is that the request rate is rising too fast causing the overflow of cached content and the replacement of popular content. The CSTL strategy has a higher average cache hit ratio compared to other strategies. The BEP and PaCC strategies dynamically cache popular content at critical nodes to help reduce the chance of popular content being substituted. Fig. 7b illustrates the effect of the request rate on the request delay for each caching strategy. The increase in content requests from users per unit of time increases the queuing delay of content requests in the queuing system, which in turn increases the average request delay. Compared to other caching schemes, the CSTL strategy is least affected by the growth of request rate and provides a more stable

caching service continuously. Fig. 7c shows that the server load for all strategies rises noticeably when the request rate rises. The CSTL strategy performs better over the entire range. The BEP and PaCC strategies dynamically sense popular content and place it on critical nodes, which better satisfies user requests. Compared with LCE and LCD strategies, Prob strategy enriches the type of cached content by probabilistic caching and better satisfies the content requests of users, thus reducing the percentage of interest packets forwarded to the server. Some of the cached content in the Betw strategy can only be answered by the server, resulting in a heavier server load than other strategies.

The experimental results show that the CSTL strategy has advantages over other caching strategies in terms of ACHR, ARD, and ASL. This is due to the CSTL strategy promoting effective utilization of cache resources and avoiding frequent replacement of cache contents through hierarchical division of contents, differentiated probabilistic caching, and dynamic adjustment of cache contents. However, the CSTL strategy only adopts the default LRU strategy and does not design a special cache replacement strategy to ensure that high-value content is stored in the cache space for a long time, which is a direction for subsequent improvement of the CSTL strategy.

6 Conclusion

Remote sensing satellite networks are characterized by time-varying topology, long delay, frequent link switching, etc., which are unable to provide a long time connected end-to-end path. To guarantee the performance of content processing and forwarding in remote sensing satellite networks, a cooperative caching strategy for remote sensing satellite networks based on a two-layer caching model is proposed in this paper. By constructing a two-layer caching model, rational planning of caching resources in satellite networks and ground stations is realized. Specifically, a probabilistic caching scheme for ground stations is devised to realize the proximity service for users in the region. In addition, differentiated caching strategy and cached content dynamic adjustment scheme are established for different levels of content in the satellite networks to achieve a reasonable allocation of cache resources. Simulation experiments show that the CSTL strategy has advantages over the existing schemes with respect to the cache hit ratio, the request delay, and the server load, which improves the timeliness of remote sensing data transmission on the basis of ensuring the stable performance of the caching system. Taking into full consideration the geographical and time-domain distribution characteristics of user requests in remote sensing satellite networks in future work, more refined placement schemes of cached contents are investigated to better accommodate the dynamic changes of contents over time in the real network. In addition, it is scheduled to further validate the caching performance of the CSTL strategy under more complex network size and simulation parameters.

Funding Statement: This research was funded by the National Natural Science Foundation of China (No. U21A20451), the Science and Technology Planning Project of Jilin Province (No. 20200401105GX), and the China University Industry University Research Innovation Fund (No. 2021FNA01003).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] J. Li, Y. Q. Pei, S. H. Zhao, R. L. Xiao and C. Y. Zhang, "A review of remote sensing for environmental monitoring in China," *Remote Sensing*, vol. 12, no. 7, pp. 1130, 2020.

- [2] C. Toth and G. Jozkow, "Remote sensing platforms and sensors: A survey," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 115, pp. 22–36, 2016.
- [3] A. E. Frazier and B. L. Hemingway, "A technical review of planet smallsat data: Practical considerations for processing and using planetscope imagery," *Remote Sensing*, vol. 13, no. 19, pp. 3930, 2021.
- [4] L. B. Zhang, J. Zhang, J. Ma and X. P. Jia, "SC-PNN: Saliency cascade convolutional neural network for pansharpening," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 11, pp. 9697–9715, 2021.
- [5] D. R. Li, M. Wang and J. Jiang, "China's high-resolution optical remote sensing satellites and their mapping applications," *Geo-Spatial Information Science*, vol. 24, no. 1, pp. 85–94, 2021.
- [6] A. Tsigkanos, N. Kranitis, G. Theodorou and A. Paschalis, "A 3.3 Gbps CCSDS 123.0-B-1 multispectral & hyperspectral image compression hardware accelerator on a space-grade SRAM FPGA," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 1, pp. 90–103, 2018.
- [7] J. Yang, D. Z. Li, X. F. Jiang, S. W. Chen and L. Hanzo, "Enhancing the resilience of low earth orbit remote sensing satellite networks," *IEEE Network*, vol. 34, no. 4, pp. 304–311, 2020.
- [8] Y. T. Yang, T. Song, W. J. Yuan and J. P. An, "Towards reliable and efficient data retrieving in ICN-based satellite networks," *Journal of Network and Computer Applications*, vol. 179, pp. 102982, 2021.
- [9] R. Zhang, J. Liu, R. C. Xie, T. Huang and F. R. Yu, "Service-aware optimal caching placement for named data networking," *Computer Networks*, vol. 174, pp. 107193, 2020.
- [10] I. U. Din, S. Hassan, A. Almogren, F. Ayub and M. Guizani, "PUC: Packet update caching for energy efficient IoT-based information-centric networking," *Future Generation Computer Systems*, vol. 111, pp. 634–643, 2020.
- [11] B. Chen, L. Liu and H. D. Ma, "HAC: Enable high efficient access control for information-centric internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 10347–10360, 2020.
- [12] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs *et al.*, "Networking named content," *Communications of the ACM*, vol. 55, no. 1, pp. 117–124, 2012.
- [13] N. Laoutaris, H. Che and I. Stavrakakis, "The LCD interconnection of LRU caches and its analysis," *Performance Evaluation*, vol. 63, no. 7, pp. 609–634, 2006.
- [14] M. Zhang, H. B. Luo and H. K. Zhang, "A survey of caching mechanisms in information-centric networking," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1473–1499, 2015.
- [15] J. Ren, W. Qi, C. Westphal, J. P. Wang, K. J. Lu *et al.*, "MAGIC: A distributed max-gain in-network caching strategy in information-centric networks," in *Proc. INFOCOM*, Toronto, ON, Canada, pp. 470–475, 2014.
- [16] A. S. Gill, L. D'Acunto, K. Trichias and R. V. Brandenburg, "BidCache: Auction-based in-network caching in ICN," in *Proc. Globecom*, Washington, DC, USA, pp. 1–6, 2016.
- [17] D. P. Man, Q. Lu, H. B. Wang, J. F. Guo, W. Yang *et al.*, "On-path caching based on content relevance in information-centric networking," *Computer Communications*, vol. 176, pp. 272–281, 2021.
- [18] W. K. Chai, D. L. He, I. Psaras and G. Pavlou, "Cache "less for more" in information-centric networks (extended version)," *Computer Communications*, vol. 36, no. 7, pp. 758–770, 2013.
- [19] H. B. Wu, J. Li and J. Zhi, "MBP: A max-benefit probability-based caching strategy in information-centric networking," in *Proc. ICC*, London, UK, pp. 5646–5651, 2015.
- [20] Q. Zheng, Y. Z. Kan, J. B. Chen, S. Wang and H. L. Tian, "A cache replication strategy based on betweenness and edge popularity in named data networking," in *Proc. ICC*, Shanghai, China, pp. 1–6, 2019.
- [21] M. Amadeo, C. Campolo, G. Ruggeri and A. Molinaro, "Popularity-aware closeness based caching in NDN edge networks," *Sensors*, vol. 22, no. 9, pp. 3460, 2022.
- [22] O. Serhane, K. Yahyaoui, B. Nour and H. Mounsla, "A survey of ICN content naming and in-network caching in 5G and beyond networks," *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 4081–4104, 2021.
- [23] S. J. Liu, X. Hu, Y. P. Wang, G. F. Cui and W. D. Wang, "Distributed caching based on matching game in LEO satellite constellation networks," *IEEE Communications Letters*, vol. 22, no. 2, pp. 300–303, 2018.

- [24] C. Qiu, H. P. Yao, F. R. Yu, F. M. Xu and C. L. Zhao, "Deep Q-learning aided networking, caching, and computing resources allocation in software-defined satellite-terrestrial networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 6, pp. 5871–5883, 2019.
- [25] Z. H. Yang, Y. Li, P. Yuan and Q. Y. Zhang, "TCSC: A novel file distribution strategy in integrated LEO satellite-terrestrial networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 5, pp. 5426–5441, 2020.
- [26] J. Li, K. P. Xue, J. Q. Liu, Y. D. Zhang and Y. G. Fang, "An ICN/SDN-based network architecture and efficient content retrieval for future satellite-terrestrial integrated networks," *IEEE Network*, vol. 34, no. 1, pp. 188–195, 2020.
- [27] X. M. Zhu, C. X. Jiang, L. L. Kuang and Z. F. Zhao, "Cooperative multilayer edge caching in integrated satellite-terrestrial networks," *IEEE Transactions on Wireless Communications*, vol. 21, no. 5, pp. 2924–2937, 2022.
- [28] Q. T. Ngo, K. T. Phan, W. Xiang, A. Mahmood and J. Slay, "Two-tier cache-aided full-duplex hybrid satellite-terrestrial communication networks," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 58, no. 3, pp. 1753–1765, 2022.
- [29] R. Xu, X. Q. Di, J. Chen, H. W. Wang, H. Luo *et al.*, "A hybrid caching strategy for information-centric satellite networks based on node classification and popular content awareness," *Computer Communications*, vol. 197, pp. 186–198, 2023.
- [30] B. Nour, H. Khelifi, H. Moungra, R. Hussain and N. Guizani, "A distributed cache placement scheme for large-scale information-centric networking," *IEEE Network*, vol. 34, no. 6, pp. 126–132, 2020.
- [31] H. T. Wu, H. H. Cho, S. J. Wang and F. H. Tseng, "Intelligent data cache based on content popularity and user location for content centric networks," *Human-centric Computing and Information Sciences*, vol. 9, no. 1, pp. 44, 2019.
- [32] P. Chini, G. Giambene and S. Kota, "A survey on mobile satellite systems," *International Journal of Satellite Communications & Networking*, vol. 28, no. 1, pp. 29–57, 2010.
- [33] S. Mastorakis, A. Afanasyev and L. X. Zhang, "On the evolution of ndnSIM: An open-source simulator for NDN experimentation," *ACM SIGCOMM Computer Communication Review*, vol. 47, no. 3, pp. 19–33, 2017.
- [34] L. Breslau, P. Cao, L. Fan, G. Phillips and S. Shenker, "Web caching and Zipf-like distributions: Evidence and implications," in *Proc. INFOCOM*, New York, NY, USA, pp. 126–134, 1999.