Coverage-Aware Cooperative Caching and Efficient Content Distribution Schemes in LEO Satellite Networks

Ruili Zhao
School of Communications and Information Engineering
Chongqing University of Posts and Telecommunications
Chongqing, China
D200101029@stu.cqupt.edu.cn

Jiangtao Luo, Yongyi Ran
Electronic Information and Networking Research Institute
Chongqing University of Posts and Telecommunications
Chongqing, China
{luojt,ranyy}@cqupt.edu.cn

ABSTRACT

Satellite networks provide the required content to users, specifically those in distant areas, by taking advantage of the wide coverage and low transmission latency of low earth orbit (LEO) constellations. With the improvement of storage and computing capabilities of satellite-based on-board devices, the utilizing of satellite caching to improve the efficiency of content distribution and reduce service delay is considered to be promising. However, traditional caching and distribution schemes are not suitable for satellite networks owing to the dynamic and time-varying topology of satellite transmission links. More specifically, the real-time dynamics of content popularity within different satellite coverage areas increases the difficulty of designing caching schemes. To solve these problems, a content popularity model based on streets of coverage (SOC) is first proposed based on the characteristics of satellite orbital operation. To maximize the utilization of limited storage space on individual satellites, a collaborative approach for content sharing among neighboring satellites is proposed. We address the collaborative caching problem by modeling it as a partially observable Markov decision process (POMDP), which enables us to jointly optimize content acquisition delays from access satellites, neighboring satellites, and ground gateways. To accommodate the dynamic nature of satellite networks, we adopt a multi-agent deep deterministic policy gradient (MADDPG) algorithm with centralized training and distributed execution. The results show that the average service delay of the proposed algorithm can be reduced by 5.3%~19.6%.

SatCom '23, October 6, 2023, Madrid, Spain

ACM ISBN 979-8-4007-0335-5/23/10.
https://doi.org/10.1145/3614454.3623000

CCS CONCEPTS

• **Networks → Location based services**.

KEYWORDS

LEO satellites network, cooperative caching, service delay, multi-agent reinforcement learning

1 INTRODUCTION

Terrestrial networks have undergone extensive development and expansion [1]. Nevertheless, terrestrial networks mostly cover the congested areas where humans live [2, 3]. Due to construction costs and geographical limitations, there are few traditional terrestrial network facilities in less economically developed and sparsely populated areas to meet the demand for network connectivity [4, 5]. In fact, 34% of the world's population will still not have the ability to connect to Internet by 2022. Extending network access to the unconnected population has become a necessity for the communication networks of the future to move forward. Satellite networks are often considered as a valuable complement to the existent terrestrial mobile communication networks owing to their global coverage and infrastructure-free nature [6]. Nevertheless, the nature of the satellite-terrestrial communication link results in high communication delays when users access the Internet via satellite. Delay optimization becomes an important and complex issue in future satellite networks.

The problem of long delay in satellite service can be effectively mitigated by satellite caching. However, owing to the rapid growing of data traffic and limited satellite cache space, coupled with the high speed motion of satellites and large constellations, where and what to cache becomes particularly important and difficult. Changes in coverage areas and user access patterns create fluctuations in content popularity and require frequent updates of cached content. Furthermore, the redundancy of cached content among satellites limits the effective utilization of cache space within the constellation. To address these issues, a dynamic cache policy is needed to determine which contents should be cached and where they should be stored, taking into account the changing cache space availability and content popularity dynamics.

At present, a large amount of work has been done to research satellite caching strategies in depth [7–12]. Optimized caching strategies on satellites, including the storage of popular user content, enhance satellite service performance in a variety of scenarios and facilitate more efficient content distribution within the satellite network [7–9], however, the movement of satellites often causes interruptions in content transmission, leading to higher service delays and packet loss rates. In study by Han et al. [8], they explore the optimization of video stream cache location based on quality of experience, taking into account factors such as desired transfer rate and social relationships among users. The cooperative caching of multi-satellites and base stations in satellite network is the primary research focus in [10–12]. In [11], Qiu et al. address the caching problem, allocation of resources and computational resources as a joint optimization problem.

However, whether for single satellite or multi-satellites, existing collaborative caching strategies mostly ignore the cooperation among satellites. To this end, a multi-satellite based fully cooperative caching scheme is proposed to reduce the content delivery delay and improve the hit ratio [13]. In this paper, the connection state between satellite and ground station (GS) is added to ensure the cooperative content transmission between satellite and GS. Our main contributions are summarized:

- Modeling the cooperative content sharing among multi-satellites and cooperative content distribution in satellites-ground gateways as a POMDP policy optimization problem, where each satellite requires only its own information, which greatly reduces the cost of high-dimensional state information transmission.
- To better capture the dynamics of the satellite network, the delay of the three service modes is optimized by using the centralized training and distributed execution of the MADDPG algorithm, which ensures the efficient distribution of satellite cache content.
- Results of the simulation show that the algorithm significantly reduces the service delay of the content while satisfying user service requests compared to the comparison algorithm.

2 PROBLEM FORMULATION

2.1 SYSTEM MODEL

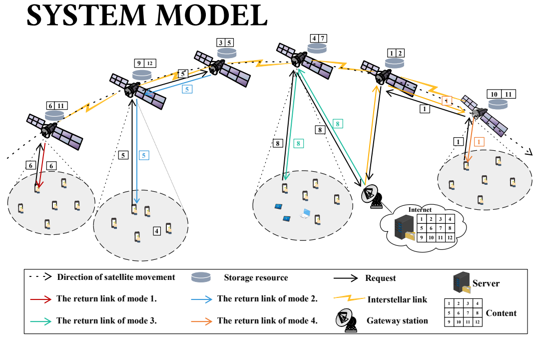


Figure 1: Cooperative caching system

As shown in Figure 1, this paper mainly considers the scenario in remote areas where users can only access the satellite network through a LEO satellite. This network consists of N LEO satellite and W GSs, satellites N provide services for the ground user terminals (UTs). The satellite set is represented by $S = \{S_1, \dots S_N\}$. The set of UTs' is represented by $U = \{U_1, \dots U_K\}$. Satellite networks guarantee wide area coverage and low access delay through inter-satellite links (ISLs). Therefore, it is crucial to design an efficient collaborative content caching and sharing strategy that minimizes the service latency of the requested content, which is the main research of this paper.

2.1.1 Coverage Model. Satellites move according to orbits, and satellites in the same orbit form SOC during their motion.

$A_n(t)$ is the total set of all points in the area covered by satellite n, can be represented as [14]:

$$A_n(t)=\{x(R,\theta,\Phi)\,|\,\cos\theta\cos\theta_n+\sin\theta\sin\theta_n\cos(\Phi-\Phi_n)\leq\cos\psi\} \tag{1}$$

In the given expression, R corresponds to the radius of earth, Φ_n represents the longitude of satellite n, $\theta_n = \left|\frac{\pi}{2} - n_{\text{latitude}}\right|$, and n_{latitude} is latitude of satellite n. The parameter ψ is half cone angle between covered area and Earth's core. Furthermore, the coordinate of any node is represented as $x(R,\theta,\Phi)$.

According to the given description, set $A_{\text{soc}}(t)$ represents all the points located within SOC at a given time t,

$$A_{\text{soc}}(t)=\{A_n(t)\,|\,n=1,2,\dots N\} \tag{2}$$

2.1.2 Request Model. UTs are distributed in SOC. UT U_k sends content request to access satellite n. The request state is denoted by $\boldsymbol{r}_n(t) = \left\{r_{n,k,1}(t), \cdots, r_{n,K,F}(t)\right\}$, where $k(R,\theta,\Phi) \in A_{soc}(t)$ is the k' coordinate. $r_{n,k,f}(t) \in \{0,1\}$, $r_{n,k,f}(t) = 1$ means that satellite n received a request from user k about content f; otherwise $r_{n,k,f}(t) = 0$.

2.1.3 Caching Model. Each content is encoded using rateless Fountain code into multi-segments, allowing UTs to recover the requested contents by collecting enough segments. $\mathcal{F}=\{1,\cdots,F\}$ is the set of contents requested by UTs. The different content is made up of a different number of segments, each of size b. The minimum number of different segments required to successfully recover content f is denoted as q_f. Every satellite has the equal cache capacity C, which is significantly smaller than the total storage requested for all contents. To simplify, we assume that no duplicate content segments are cached in the satellite. The content placement is periodically refreshed, and the time slots are indexed as $t = 0, 1, \cdots$. $c_{n,f}(t)$ denotes the number of segments with content f cached in satellite n. The constraint for fully cached segments in satellite n can be expressed:

$$c_{n,f}(t) \leq q_f, \forall n, f, t. \tag{3}$$

$$\sum_{f\in\mathcal{F}} c_{n,f}(t)b \leq C, \forall n, f, t. \tag{4}$$

$$C \ll \sum_{f=1}^{F} q_f b, \forall f. \tag{5}$$

The cache state of access satellite n is defined as

$$\boldsymbol{c}_n(t) = \left\{c_{n,1}(t), \cdots, c_{n,F}(t)\right\}. \tag{6}$$

2.1.4 Popularity Model. Owing to the rapid movement of satellites, popularity of content within the satellite coverage area could change geographically at different times. UTs in SOC are relatively constant, which facilitates statistical analysis and training of content popularity. The popularity of the o-th content within the coverage of satellite n is denoted by $p_{n,o}$, following Mandelbrot-Zipf (MZipf) distribution [15]. From this, we can derive the following relationship:

$$p_{n,o} = \frac{Q_n(o)^{-z_n}}{Q_{\mathrm{soc}}(o)^{-z_n}} \tag{7}$$

where z_n is a skewness factor, $Q_n(o)$ is popularity rank of the o-th content in the coverage of sateliite n, $Q_{\mathrm{soc}}(o)$ is the popularity rank of the o-th content in SOC coverage.

2.1.5 Satellite Access Model. Due to the limited number of GSs, satellites are not always connected to them. Let $I_{n,w}(t)$ denote the connection state between the satellite n and the GS w, $I_{n,w}(t) \in \{0, 1\}$. Specifically, $I_{n,w}(t) = 1$ means that the satellite n is connected to GS w within the duration of $[t, t+1)$. Therefore, the connection state of satellite n to GS could be expressed as $\boldsymbol{I}_n(t) = \left\{I_{n,1}(t), \cdots, I_{n,W}(t)\right\}$.

2.1.6 Service Model. In our collaborative caching system, requests of UT can be provided by GSs, access satellite, or neighboring satellite, depending on the content placements shown in Figure 2.

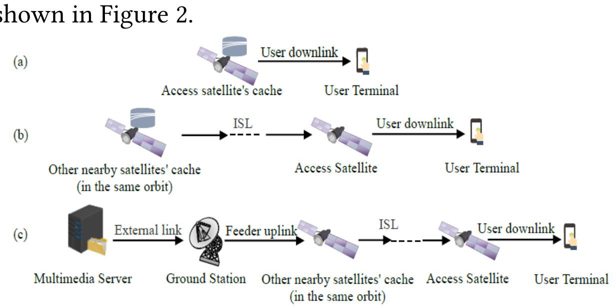


Figure 2: Service paths for different cache locations.

Access service mode ($m^{ac} = 1$) : If requested content is cached in access satellite, it could be delivered to UT directly, as illustrated in Figure 2 (a). Content service latency in this model is denoted as $D_n^{ac}(t)$ and is calculated as follows:

$$D_n^{ac}(t) = D_n^{tr}(t) + D_n^{p}(t), \tag{8}$$

where $D_n^{tr}(t)$ indicates the content transmission delay, $D_n^{p}(t)$ indicates the content propagation delay.

$$D_n^{tr}(t) = \frac{c_{n,f}(t) r_{n,k,f}(t) b}{R_{n,k}}, D_n^{p}(t) = \frac{c_{n,f}(t) r_{n,k,f}(t) d_{n,k}(t)}{c}. \tag{9}$$

c stands for the speed of light. For simplicity, channel model of satellite-user link is modeled as a free space propagation model. Let $h_{n,k}(t)$ represent the channel gain of satellite n and UT U_k in the time slot t, $h_{n,k}(t)$ can be expressed as:

$$h_{n,k}(t) = G_n G_k(t) L_{n,k}(t) L_p \tag{10}$$

where G_n indicates the satellite transmitting antenna gain; $G_k(t)$ indicates the UT k receiving antenna gain; L_{p} denotes the rain-decay factor; $L_{n,k}(t)$ denotes the free space loss at time slot t satellite n to UT k. Let $R_{n,k}(t)$ denote the transmission rate from satellite n to UT k, it can be represented:

$$R_{n,k}(t) = B_{n,k} \log_2\left(1 + \frac{P_n h_{n,k}(t)}{\sigma^2}\right) \tag{11}$$

where $B_{n,k}$, $h_{n,k}(t)$ denote the channel bandwidth, channel gain of the satellite n to UT U_k, respectively. σ^2 denotes the background noise power, P_n is transmit power of satellite n.

Cooperative service mode ($m^{co} = 1$) : In our collaborative caching system, content segments that are part of the same content can be distributed across multi-satellites. If the access satellite does not cache the user's requested content but caches it on nearby satellites, the UT's request could be fulfilled through sharing content among these satellites. When requested content needs to be obtained from a nearby satellite, it is first send to access satellite of the requested content through ISL, and then sent to UT by access satellite.

For simplicity, the channel model of ISL is modeled as random fading channel. Let $R_{n,n-1}(t)$ denote the transmission rate rom satellite n to satellite $n-1$,

$$R_{n,n-1}(t) = \frac{G_{\mathrm{r}} G_{\mathrm{t}} P_n L_{n,n-1}(t) \xi}{k_{\mathrm{s}} T_{\mathrm{s}} \left(E_{\mathrm{b}}/N_0\right)}, L_{n,n-1}(t) = \left(\frac{\lambda}{4\pi d_{n,n-1}(t)}\right)^2 \tag{12}$$

where G_r indicates the satellite transmitting antenna gain; P_n denotes the transmitting power of satellite n; E_b denotes the transmitting power of satellite n; T_s is the system noise temperature; N_0 is the power spectral density; k_s denotes the bolmanz constant; ξ is the random fading factor; At time slot t, $L_{n,n-1}(t)$ denotes the free space loss from satellite n to satellite $n-1$, λ is the wavelength, $d_{n,n-1}(t)$ is the distance between satellite n and $n-1$ in slot t. $D_n^{co}(t)$ indicates content service latency for this model, and the service path is shown in Figure 2 (b),

$$D_n^{co}(t) = D_n^{ac}(t) + D_{n-h,n}^{h}(t), \tag{13}$$

where $D_{n-h,n}^{h}(t) = D_{n-h,n}^{p}(t) + D_{n-h,n}^{tr}(t)$, $D_{n-h,n}^{p}(t)$ indicates the content propagation delay through the h hop ISL, $D_{n-h,n}^{p}(t) = \frac{c_{n-h,f}(t) r_{n,k,f}(t) d_{n-h,n}(t)}{c}$, $d_{n-h,n}(t) = d_{n-h,n-h+1}(t) + \ldots + d_{n-1,n}(t)$. $D_{n-h,n}^{tr}(t)$ is the transmission delay of the satellite $n-h$ transmitted to access satellite n through h hop,

$$D_{n-h,n}^{tr}(t) = \frac{c_{n-h,f}(t) r_{n,k,f}(t) b}{R_{n-h,n-h+1}(t)} + \ldots + \frac{c_{n-h,f}(t) r_{n,k,f}(t) b}{R_{n-1,n}(t)}. \tag{14}$$

$$c_{n,f}(t) + c_{n-1,f}(t) + \ldots + c_{n-h+1,f}(t) = 0. \tag{15}$$

Source service mode ($m^{so} = 1$) : If the content requested by the user is not cached in the satellite constellation, it will be retrieved from GS as illustrated in Figure 2 (c). In the uplink channel of GS, large-scale fading is usually considered. $I_{n-h,w}(t) = 1$ and $I_{n-h+1,w}(t), \ldots, I_{n,w}(t) = 0$ indicates the GS w is connected to satellite $n-h$ in satellite $n, \ldots, n-h$. $g_{w,n-h}(t)$ is the channel gain from GS w to satellite $n-h$. Thus, the channel gain could be expressed as:

$$g_{w,n-h}(t) = G d_{w,n-h}^{-\alpha}(t), \tag{16}$$

where $d_{w,n-h}$ denotes the distance from GS w to the satellite $n-h$. α, G denote the path loss index and constant power gain factor, respectively. Assuming that GS w communicates with one satellite at a time with a transmit power of P_w. Therefore, the uplink transmission rate between GS w and satellite n can be expressed as:

$$R_{w,n-h}(t) = B_s \log_2\left(1 + \frac{P_g g_{w,n-h}(t)}{\sigma^2_{g,n-h}(t)}\right), \tag{17}$$

where $\sigma^2_{w,n-h}$, B_w denote channel noise power and uplink transmission bandwidth, respectively. $D^p_{w,n-h}(t)$ is the propagation delay. $D^{so}_n(t)$ is content service delay for this mode,

$$D^{so}_n(t) = D^{ac}_n(t) + D^h_{n,n-h}(t) + D^w_{w,n-h}(t), \tag{18}$$

$$c_{n,f}(t) + c_{n-1,f}(t) + \ldots + c_{n-h,f}(t) = 0. \tag{19}$$

$$D^w_{w,n-h}(t) = D^p_{w,n-h}(t) + D^{tr}_{w,n-h}(t), D^p_{w,n-h}(t) = \frac{d_{w,n-h}(t)}{c}. \tag{20}$$

$D^{tr}_{w,n-h}(t) = \frac{I_{n-h,w}(t)q^{so}_f(t)r_{n,k,f}(t)b}{R_{w,n-h}(t)}$, $q^{so}_f(t)$ denotes the number of segments with content f provided by GS w.

2.2 Problem Formulation

Define D^{avg} as the average service delay for UTs to request content to be served within time slot T, calculated as follows:

$$\begin{aligned} D^{avg} &= \frac{1}{NTKF}\sum_t\sum_n\sum_k\sum_f \frac{\left[c_{n,f}(t)m^{ac}D^{ac}_n(t)\right]}{\left|r_{n,k,f}(t)\right|} + \frac{1}{NTKF} \\ &\cdot \sum_t\sum_n\sum_k\sum_f \frac{\left[c_{n-1,f}(t) + \ldots + c_{n-h,f}(t)\right] m^{co}D^{co}_n(t)}{\left|r_{n,k,f}(t)\right|} \\ &+ \frac{1}{NTKF}\sum_t\sum_n\sum_k\sum_f \frac{I_{n-h,w}(t)q^{so}_f(t)r_{n,k,f}(t)m^{so}D^{so}_n(t)}{\left|r_{n,k,f}(t)\right|} \end{aligned} \tag{21}$$

Within this work, we solved problem of minimizing average content service delay by optimizing cache placement of satellites. We express the collaborative caching problem as:

$$\begin{aligned} \min_{c_{n,f}} \quad & D^{avg} \\ \text{s.t.} \quad & C_1 : r_{n,k,f}(t), m^{ac}, m^{co}, m^{so} \in \{0,1\} \\ & C_2 : m^{ac} + m^{co} + m^{so} = 1 \\ & C_3 : H \ll N \\ & C_4 : (3), (4), (5) \end{aligned} \tag{22}$$

where C_2 ensures UTs' content requests can always be served by one of service models; C_3 ensure the maximum number of hops of ISLs does not exceed the number of satellites.

3 THE PROPOSED CACC ALGORITHM

3.1 POMDP

Status, action and reward are given by following definitions:

- **State Space:** At the time slot t, state of the whole system including the request state, the cache state, the content popularity state and the connection state between satellites and GSs can be expressed as:

$$\boldsymbol{\chi_n}(t) = (\boldsymbol{r}_n(t), \boldsymbol{c}_n(t), \boldsymbol{p}_n(t), \boldsymbol{I}_n(t)) \tag{23}$$

$\boldsymbol{r}_n(t)$, $\boldsymbol{c}_n(t)$ are the received content request state of satellite n at time slot t and its own content cache state, respectively. $\boldsymbol{p}_n(t)$ is content popularity state, $\boldsymbol{p}_n(t) = \{p_{n,1}(t), \ldots, p_{n,F}(t)\}$, $p_{n,f}(t)$ denotes content popularity of content f in the coverage of satellite n. Finally, $\boldsymbol{I}_n(t)$ is connection state of satellites n and GSs.

- **Action Space:** To adapt to the dynamics in content popularity and satellites, three types of actions are defined, denoted as: $\boldsymbol{a}^{ac}_n(t)$, $\boldsymbol{a}^{co}_n(t)$, and $\boldsymbol{a}^{so}_n(t)$. Thus, at time t, action taken by satellite n could be denoted as:

$$\Phi(\boldsymbol{\chi}_n(t)) = \left\{\boldsymbol{a}^{ac}_n(t), \boldsymbol{a}^{co}_n(t), \boldsymbol{a}^{so}_n(t)\right\}. \tag{24}$$

where $\boldsymbol{a}^{ac}_n(t) = \left\{a^{ac}_{n,1}(t), \ldots, a^{ac}_{n,F}(t)\right\}$ is access actions, $a^{ac}_{n,f}(t)$ represent the cache size (measured in segments) allocated for the corresponding content f to be stored locally by satellite n at the current time, as indicated by the caching action taken. At the time slot t, let $c_{n,f}(t)$ indicate the segment number of content f cached in the satellite n at present. While taking action $a^{ac}_{n,f}(t)$, it means that at time slot t, the number of fragments of satellite n cache content f is $a^{ac}_{n,f}(t)$. For the sake of simplicity, we suppose that cache replacement updates do not take time. Therefore, at time instant $t+1$, satellite n will have $a^{ac}_{n,f}(t)$ segments for content f, i.e. $\boldsymbol{c}_n(t+1) = \boldsymbol{a}^{ac}_n(t)$. To represent the change in cache contents after action $a^{ac}_{n,f}(t)$, we make a comparison between $a^{ac}_{n,f}$ and $c_{n,f}$. More specifically, if $a^{ac}_{n,f} > c_{n,f}$, then content f will have $\left|a^{ac}_{n,f} - c_{n,f}\right|$ segments added to satellite n. If $a^{ac}_{n,f} < c_{n,f}$, then content f will have $\left|a^{ac}_{n,f} - c_{n,f}\right|$ segments removed from the cache of satellite n. Otherwise, there is no adjustment of the content f. $\boldsymbol{a}^{co}_n(t) = \left\{\ldots, a^{ac}_{n,n-h,f}(t), \ldots\right\}$ is cooperative actions, $\boldsymbol{a}^{so}_n(t) = \left\{\ldots, a^{ac}_{n,w,f}(t), \ldots\right\}$ is GS actions. Similarly, cooperative actions $a^{co}_{n,n-h,f}(t)$, GS actions $a^{so}_{n,w,f}(t)$ indicates the number of segments for content f that neighboring satellite $n-h$ and GS w need to provide to the access satellite n, respectively.
- **Reward Function:** After the system takes action **A** in state **S**, it receives feedback rewards. To maximize the system reward and minimize the average content service latency, we adopt a negative exponential function to normalize the reward. Therefore, the reward could be expressed as:

$$\boldsymbol{R}_n\left(\boldsymbol{\chi}(t), \Phi\left(\chi(t)\right)\right) = \begin{cases} e^{-D^{ac}_n(t)}, & \text{Access} \\ e^{-D^{co}_n(t)}, & \text{Cooperation} \\ e^{-D^{so}_n(t)}, & \text{Source} \\ 0, & \text{Else} \end{cases} \tag{25}$$

where $e^{-D_n^{ac}(t)}$, $e^{-D_n^{co}(t)}$, $e^{-D_n^{so}(t)}$ denote the rewards for the access satellite to provide the requested content directly, the rewards for the requested content provided by neighboring satellites, and the rewards for the requested content provided by GSs, respectively. Otherwise, the reward is 0.

3.2 CACC

As shown in Figure 3, each satellite is considered an agent. The estimation network and the target network of the agent are $Q_i\left(s_i, a_i; \theta_i\right)$, $Q_i'\left(s_i, a_i; \theta_i'\right)$ respectively. The parameters of which are θ_i and θ_i' respectively. Each agent's experiential playback buffer is also independently initialized to restore its environmental transformations and to update network parameters using random sampling.

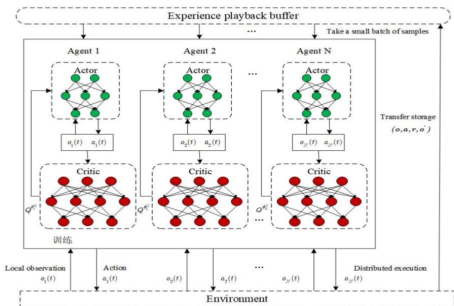


Figure 3: Architecture of cooperative caching.

At each decision moment t, the agent i will observe and share local information. Through integrating the present state $\boldsymbol{\chi}_i(t)$, the agent i selects an action from the action space $\boldsymbol{\Phi}(\boldsymbol{\chi}_i(t))$ based on the $\varepsilon - greedy$ strategy, i.e. Probability selects an action randomly from the action space $\boldsymbol{\Phi}(\boldsymbol{\chi}_i(t))$, or selects an action that has the highest Q value with probability $1-\varepsilon$, as shown in formula (26) :

$$\Phi\left(\mathcal{X}_i(t)\right) = \begin{cases} \text{Random selection action}, & \varepsilon \\ \arg\max_{\Phi} Q_i\left(\mathcal{X}_i(t), \Phi\left(\mathcal{X}_i(t)\right); \theta_i(t)\right), & 1-\varepsilon \end{cases} \tag{26}$$

Based on the selected action, agent i will make a cache replacement and compute rewards $\boldsymbol{R}\left(\boldsymbol{\chi}_i(t), \boldsymbol{\Phi}\left(\chi_i(t)\right)\right)$. The present state $\boldsymbol{\chi}_i(t)$ of the agent i will transform to next state $\boldsymbol{\chi}_i(t+1)$. After agent i receives this feedback, i records transformation $\left(\boldsymbol{\chi}_i(t), \boldsymbol{\Phi}\left(\chi_i(t)\right), \boldsymbol{R}\left(\boldsymbol{\chi}_i(t), \boldsymbol{\Phi}\left(\chi_i(t)\right)\right), \boldsymbol{\chi}_i(t+1)\right)$ into the experiential playback buffer $\mathcal{D}$, and then takes a random set of samples for learning, thus disrupting the dependency between the training data. We make the parameters of $Q_i\left(\theta_i(t)\right)$ by minimizing the average square error between the target Q value and the present Q network output. The loss $L_i(t)$ is provided by equation (27).

$$L_i(t) = \left(y_i(t) - Q_i\left(\mathcal{X}_i(t), \Phi\left(\mathcal{X}_i(t)\right); \theta_i(t)\right)\right)^2. \tag{27}$$

The $y_i(t)$ indicates the target value could be defined as the formula (28).

$$\begin{aligned} y_i(t) &= R\left(\mathcal{X}_i(t), \Phi\left(\mathcal{X}_i(t)\right)\right) \\ &+ \gamma \max_{\Phi(\mathcal{X}_i(t+1))} Q_i'\left(\mathcal{X}_i(t+1), \Phi\left(\mathcal{X}_i(t+1)\right); \theta_i(t')\right), \end{aligned} \tag{28}$$

where γ denotes the discount factor; $\theta_i(t)$, $\theta_i(t')$ are the parameters of the Q-estimation network and the parameters of the Q-target network, respectively. The parameter $\theta_i(t')$ is updated by $\theta_i(t)$, which is updated every C-step and remains constant between each update. Therefore, we can update parameter $\theta_i(t)$ as shown in equation (29):

$$\theta_i(t) \leftarrow \theta_i(t) + \alpha \nabla_{\theta_i(t)}\left(y_i(t) - Q_i\left(\mathcal{X}_i(t), \Phi\left(\mathcal{X}_i(t)\right); \theta_i(t)\right)\right)^2, \tag{29}$$

where α is the learning rate.

4 PERFORMANCE EVALUATION

Throughout this section, simulations are given to verify the performance of the proposed collaborative caching algorithm CACC in LEO satellite networks. We first discuss the simulation parameter settings. To construct a simulation scenario of the satellite network, we utilized STK, Python 3.7.0 and Pytorch 1.9.0, using Starlink as the representative reference. This network is made of 66 satellites, which are uniformly distributed in one orbit [16]. Additional parameters are provided in Table I. To facilitate comparison, we consider the following five caching algorithms.

Table 1: Some Typical Commands

Parameter	Value	Parameter	Value
Number of Satellites	66	Content size	0.1MB
LEO orbit height	550	Discount factor	0.9
Update rate	0.01	Learning rate η	0.001
Capacity C	30MB	Epochs	200
Number of content	3000	Mini-batch size	8
Speed of light c	3×10^8	Discount factor	0.95
Decay coefficient	0.0001	Replay buffer	50000
Orbit inclination	53°	Fading factor	0.9

- ESM (Exchange-Stable Matching): An exchange-stable matching algorithm based on game theory [10].
- FIFO (First Input First Output): The system prioritizes replacing the oldest content.
- LRU (Least Recently Used): The system replaces the content that has been used the least recently.
- LFU (Least Frequently Used): The system replaces the content that has been accessed the least frequently.

4.1 Results & Performance

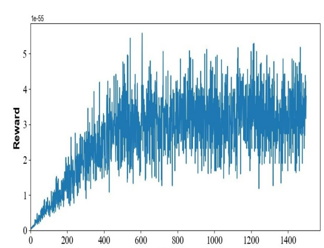


Figure 4: Learning process.

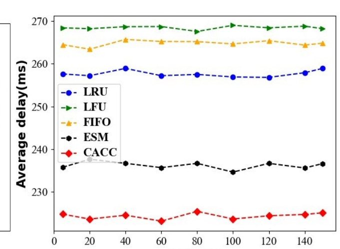


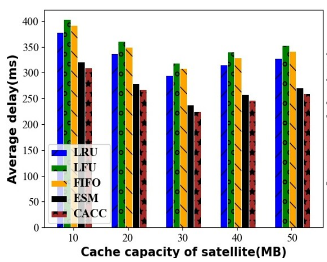
Figure 5: Average delay.

In Figure 4, the reward initially grows rapidly within approximately 500 episodes and then stabilizes, indicating convergence of the learning algorithm after around 500 training episodes. Once the network is sufficiently trained, it can accurately estimate the value function.

In Figures 5, CACC achieves the lowest average latency with 224.4ms, outperforming ESM, LRU, LFU, and FIFO by 5.3%, 14.8%, 19.6%, and 18% respectively. CACC effectively shares cached content among satellites, eliminating redundancy and saving storage space.

The experimental results in Figures 6 ($F = 3000, S = 11, W = 26$) show that the average delay of all algorithms decreases as the satellite storage capacity increases. This is because a larger storage capacity allows for caching of more diverse contents, leading to a higher content request fulfillment rate and consequently lower content service delay.

In Figure 7 ($C = 30MB, S = 11, W = 26$), the average service delay of the system increases with the increase of the number of content requests. This is mainly due to the fact that when the more popular content is cached in the satellite, as the amount of content increases, the newly requested content is not replaced by the cache into the satellite, so the average service delay of the content increases.



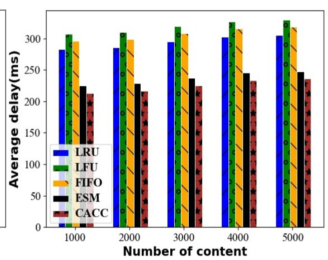
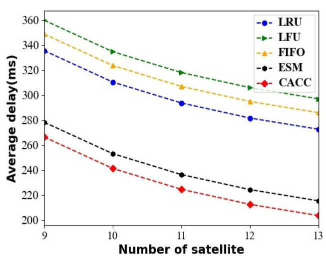


Figure 6: Delay-Cache capacities. **Figure 7: Delay-Content number.**



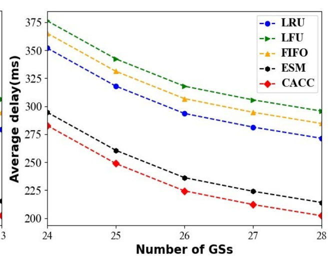


Figure 8: Delay-Satellite number. **Figure 9: Delay-GS number.**

Similarly, from Figure 8 ($C = 30MB, F = 3000, W = 26$), it can be seen that as the number of satellites increases, the average content service delay of the five algorithms decreases significantly. More satellites means that more popular content is cached on satellite so that it can be sent directly to the user, thus reducing service delay.

Finally, from Figure 9 ($C = 30MB, S = 11, F = 3000$), it can be seen that as the number of GSs increases, the average content service delay of the five algorithms decreases significantly. This is because when the requested content is served by the GS, more GSs can reduce the hop number of ISL transmission, thereby reducing the content service delay.

5 CONCLUSION

We approached the collaborative caching problem as a POMDP-based multi-agent decision problem, aiming to optimize content service delay. To address this, we introduced the CACC algorithm, which effectively enabled satellite collaboration and reduced content retrieval delays.

6 ACKNOWLEDGMENTS

This work was supported by National Natural Science Foundation of China (No.62171072, 62172064, 62003067, 62101525), Natural Science Foundation of Chongqing (cstc2021jcyj-msxmX0586) and Chongqing Postgraduate Research and Innovation Project (CYB21204).

REFERENCES

[1] Jun Du, Chunxiao Jiang, Haijun Zhang, Yong Ren, and Mohsen Guizani. Auction design and analysis for sdn-based traffic offloading in hybrid satellite-terrestrial networks. *IEEE Journal on Selected Areas in Communications*, 36(10):2202–2217, 2018.

[2] Zhenjiang Zhang and Wenyu et al Zhang. Satellite mobile edge computing: Improving qos of high-speed satellite-terrestrial networks using edge computing techniques. *IEEE Network*, 33(1):70–76, 2019.

[3] Giovanni Giambene, Sastri Kota, and Prashant Pillai. Satellite-5g integration: A network perspective. *IEEE Network*, 32(5):25–31, 2018.

[4] Linling Kuang, Xi Chen, Chunxiao Jiang, Haijun Zhang, and Sheng Wu. Radio resource management in future terrestrial-satellite communication networks. *IEEE Wireless Communications*, 24(5):81–87, 2017.

[5] Xiangming Zhu, Chunxiao Jiang, Linling Kuang, Zhifeng Zhao, and Song Guo. Two-layer game based resource allocation in cloud based integrated terrestrial-satellite networks. *IEEE Transactions on Cognitive Communications and Networking*, 6(2):509–522, 2020.

[6] Xiangming Zhu, Chunxiao Jiang, Liuguo Yin, Linling Kuang, Ning Ge, and Jianhua Lu. Cooperative multigroup multicast transmission in integrated terrestrial-satellite networks. *IEEE Journal on Selected Areas in Communications*, 36(5):981–992, 2018.

[7] Guiting Zhong, Jian Yan, and Linling Kuang. Qoe-driven social aware caching placement for terrestrial-satellite networks. *China Communications*, 15(10):60–72, 2018.

[8] Dairu Han, Haixia Peng, Huaqing Wu, Wenhe Liao, and Xuemin Sherman Shen. Joint cache placement and content delivery in satellite-terrestrial integrated c-rans. In *ICC 2021 - IEEE International Conference on Communications*, pages 1–6, 2021.

[9] Kang An, Yusheng Li, Xiaojuan Yan, and Tao Liang. On the performance of cache-enabled hybrid satellite-terrestrial relay networks. *IEEE Wireless Communications Letters*, 8(5):1506–1509, 2019.

[10] Shuaijun Liu, Xin Hu, Yipeng Wang, Gaofeng Cui, and Weidong Wang. Distributed caching based on matching game in leo satellite constellation networks. *IEEE Communications Letters*, 22(2):300–303, 2018.

[11] Chao Qiu, Haipeng Yao, F. Richard Yu, Fangmin Xu, and Chenglin Zhao. Deep q-learning aided networking, caching, and computing resources allocation in software-defined satellite-terrestrial networks. *IEEE Transactions on Vehicular Technology*, 68(6):5871–5883, 2019.

[12] Ashkan Kalantari, Marilena Fittipaldi, Symeon Chatzinotas, Thang X. Vu, and Bjoern Ottersten. Cache-assisted hybrid satellite-terrestrial backhauling for 5g cellular networks. In *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, pages 1–6, 2017.

[13] Ruili Zhao, Yongyi Ran, Jiangtao Luo, and Shuangwu Chen. Towards coverage-aware cooperative video caching in leo satellite networks. In *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*, pages 1893–1898, 2022.

[14] Haotian Zhou, Liang Liu, and Huadong Ma. Coverage and capacity analysis of leo satellite network supporting internet of things. In *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, pages 1–6, 2019.

[15] Hao Wu, Jian Li, Hancheng Lu, and Peilin Hong. A two-layer caching model for content delivery services in satellite-terrestrial networks. In *2016 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, 2016.

[16] Aizaz U. Chaudhry and Halim Yanikomeroglu. Laser intersatellite links in a starlink constellation: A classification and analysis. *IEEE Vehicular Technology Magazine*, 16(2):48–56, 2021.