Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Optimal replication strategy for mitigating burst traffic in information-centric satellite networks: a focus on remote sensing image transmission*#

Ziyang XING[1,2], Xiaoqiang DI†‡[1,2,3], Hui QI[1,2], Jing CHEN[1,2], Jinhui CAO[1,2], Jinyao LIU[1,2], Xusheng LI[1,2], Zichu ZHANG[1,2], Yuchen ZHU[1,2], Lei CHEN[1,2], Kai HUANG[1,2], Xinghan HUO[1,2]

[1]*Jilin Key Laboratory of Network and Information Security, Changchun 130022, China*

[2]*School of Computer Science and Technology, Changchun University of Science and Technology, Changchun 130022, China*

[3]*Information Center, Changchun University of Science and Technology, Changchun 130022, China*

†E-mail: dixiaoqiang@cust.edu.cn

Received Jan. 11, 2024; Revision accepted Feb. 20, 2024; Crosschecked Mar. 22, 2024; Published online Apr. 18, 2024

Abstract: Information-centric satellite networks play a crucial role in remote sensing applications, particularly in the transmission of remote sensing images. However, the occurrence of burst traffic poses significant challenges in meeting the increased bandwidth demands. Traditional content delivery networks are ill-equipped to handle such bursts due to their pre-deployed content. In this paper, we propose an optimal replication strategy for mitigating burst traffic in information-centric satellite networks, specifically focusing on the transmission of remote sensing images. Our strategy involves selecting the most optimal replication delivery satellite node when multiple users subscribe to the same remote sensing content within a short time, effectively reducing network transmission data and preventing throughput degradation caused by burst traffic expansion. We formulate the content delivery process as a multi-objective optimization problem and apply Markov decision processes to determine the optimal value for burst traffic reduction. To address these challenges, we leverage federated reinforcement learning techniques. Additionally, we use bloom filters with subdivision and data identification methods to enable rapid retrieval and encoding of remote sensing images. Through software-based simulations using a low Earth orbit satellite constellation, we validate the effectiveness of our proposed strategy, achieving a significant 17% reduction in the average delivery delay. This paper offers valuable insights into efficient content delivery in satellite networks, specifically targeting the transmission of remote sensing images, and presents a promising approach to mitigate burst traffic challenges in information-centric environments.

Key words: Information-centric satellite network; Burst traffic; Content delivery; Federated reinforcement learning; Mixed-integer linear programming model; Bloom filter; Dynamic network

https://doi.org/10.1631/FITEE.2400025 **CLC number:** TP393

1 Introduction

Satellite remote sensing plays a crucial role in various domains within the space–ground network, such as information transmission for military data link applications (Wu F et al., 2022). Remote sensing satellites, including Meteosat and Jilin-1 commercial satellites, provide a God's eye view and serve as

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (No. U21A20451)

Electronic supplementary materials: The online version of this article (https://doi.org/10.1631/FITEE.2400025) contains supplementary materials, which are available to authorized users

ORCID: Xiaoqiang DI, https://orcid.org/0000-0001-9432-4564

essential tools for high-resolution observations of the Earth's surface. However, in the context of satellite networks, sudden increases in request, known as burst traffic, pose a challenge that additional bandwidth is required to handle the new demands. For instance, during earthquakes, there is a need to transmit a specific region's remote sensing image to numerous ground users who cannot communicate with each other. These simultaneous requests for the same remote sensing image within a short period lead to burst traffic occurrences. Thus, effectively mitigating or eliminating burst traffic in satellite networks becomes crucial. Traditional content delivery networks (CDNs), which can prevent burst traffic, consist of servers deployed in different regions, and are unable to handle burst traffic due to their predeployed content (Luglio et al., 2019). Other methods for reducing burst traffic cannot be implemented well due to the mobility characteristics of satellite networks (Liu S et al., 2021).

Based on this, the non-IP-based (IP is short for Internet protocol) information-centric networks (ICNs) have great advantages in content delivery (https://www.rfc-editor.org/rfc/rfc8793). ICNs decouple network location from content, and provide support for mobility and caching (Bilal and Kang, 2019), thus making them well-suited for large-scale content delivery and air–ground transmission. Fig. 1 illustrates the principle of content delivery. Consumer 1 aims to download image 1. When the router fails to hit image 1, the content provider supplies and caches it in router 2. If consumer 2 also needs image 1 (the same one that consumer 1 has), consumer 2 can simply redownload it from cached router 2, eliminating the need for content re-provisioning.

The term "information-centric satellite network" refers to a satellite network based on ICN, which enables efficient content delivery, bandwidth saving, content caching, and efficient transmission in terms of high coverage and low latency of the satellite network (Li J et al., 2020); i.e., remote sensing images have geographic and shooting time information, additionally, compared to normal images, making them more conducive to operations such as positioning, retrieval, and comparison. The research motivations are to fully use the spatiotemporal properties of remote sensing images in the information-centric satellite network and to solve the problem of content delivery in a burst traffic scenario.

However, as the number of users increases (as shown in Fig. 2), delivering the same remote sensing image to multiple users results in significantly redundant data, leading to increased bandwidth overhead and potential burst traffic formation. Therefore, this paper aims to balance and optimize the traffic required for new users, achieving the goals of maximizing the content delivery throughput and minimizing the transmission time. This approach avoids creating additional transmission data and excessive bandwidth consumption for the same remote sensing image, preventing a severe decline in the throughput. To achieve this, we apply a federated reinforcement learning approach to generate optimal replication

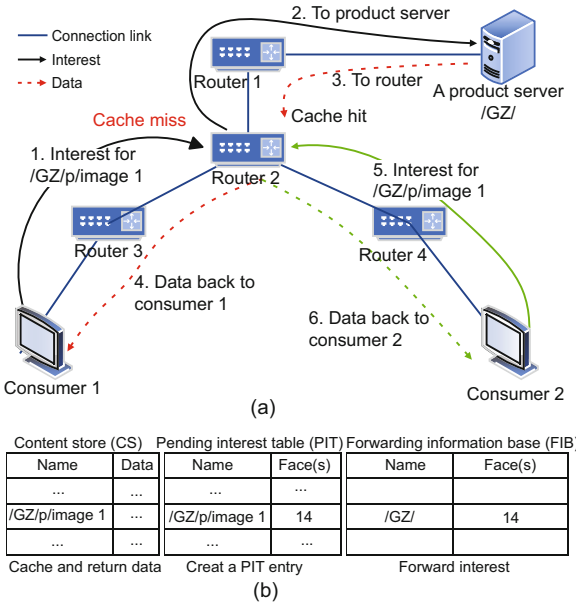

Fig. 1 Content delivery process in an information-centric network: (a) caching and hitting in content delivery; (b) tables in router 2

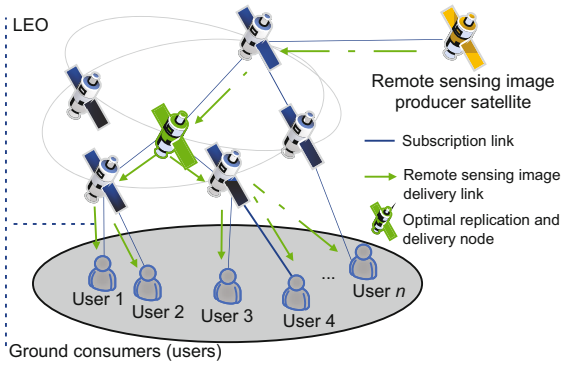

Fig. 2 Scheme of a satellite delivering remote sensing images to multiple users on the ground (LEO: low Earth orbit)

and transmission satellite nodes in remote sensing image delivery based on an information-centric satellite network.

The main contributions of this paper are as follows:

1. We propose an optimal replication strategy for mitigating burst traffic in information-centric satellite networks, specifically focusing on the transmission of remote sensing data. By selecting the most suitable replication delivery satellite node when multiple users subscribe to the same remote sensing content within a short time frame, we effectively reduce network transmission data and prevent throughput degradation caused by burst traffic expansion.

2. We formulate the content delivery process as a multi-objective optimization problem and apply Markov decision processes to determine the optimal value for burst traffic reduction. Leveraging federated reinforcement learning techniques, we address the challenges posed by burst traffic in information-centric satellite networks.

3. To enable rapid retrieval and encoding of remote sensing images, we use bloom filters with subdivision and data identification methods. This approach facilitates efficient content delivery in satellite networks, specifically targeting the transmission of remote sensing data.

4. Through software-based simulations using low Earth orbit (LEO) satellite constellation, we validate the effectiveness of our proposed strategy. Our results demonstrate a significant 17% reduction in the average delivery delay, highlighting the potential to mitigate burst traffic challenges in information-centric environments. This research provides valuable insights into efficient content delivery in satellite networks, particularly in the context of transmitting remote sensing data.

2 Related works

Content delivery refers to the process of delivering centralized content to some scattered nodes; content delivery in the context of satellites has the characteristics of low bandwidth, large user demand, and uneven delivery of nodes. CDNs are widely used in video-on-demand, large-scale website image acceleration and other applications. Narayanan et al. (2018) combined ICNs and CDNs in satellite networks. Satellite content delivery network (SCDN) (https://www.pioneeringminds.com/satellite-based-content-delivery-network-cdn-extraterrestrial-environment) is used for large-scale content delivery, but its disadvantage is that the content needs to be deployed in advance to the location near the user; this feature is not suitable for content delivery that requires high timeliness in the satellite network. Aung et al. (2023), who analyzed the serious impact of high mobility on content delivery in the Internet of Vehicles, proposed a traffic perception scheme based on deep reinforcement learning, which can significantly improve delivery efficiency. Liu JY et al. (2023) proposed a discrete-time CDN slicing system to address the issue of unstable distribution of user-requested content in dynamic networks; this system can delivery more content one time.

In the process of content delivery from network satellites to ground users, when the number of subscribers suddenly increases in a short period of time, burst traffic will inevitably form, which will affect the data transmission performance of the entire network. Rastegar et al. (2020) proposed a flow table item caching scheme to avoid burst traffic in software-defined networks, which may cause excessive consumption of flow table items when numerous users are added. Wang W et al. (2020) proposed a burst traffic prediction scheme that combines neural networks and Gaussian process regression to address low accuracy of traditional traffic prediction. Zhang et al. (2023) improved fine-grained traffic perception and grouped congestion control notification labels to reduce queue backlog caused by burst traffic, addressing the issue of congestion control notification labels being prone to failures during transmission. Wang J et al. (2023) proposed a traffic differentiation load balance that can adaptively adjust long and short flows to avoid network congestion, addressing the problem of heterogeneous flows and avoiding the formation of burst traffic in a short period of time. Yu et al. (2023) proposed to coordinate parallel resources to handle burst and non-burst traffic separately, in response to the lack of mechanisms to cope with burst traffic in current fifth-generation (5G) networks, to achieve conflict-free resource allocation for burst traffic.

Therefore, it is necessary to comprehensively consider the impact of burst traffic generated by the rapid increase of the number of users in the satellite

network on content delivery.

Federated reinforcement learning began in 2019 and has been widely used in the Internet of Vehicles and the Internet of Things (Li X et al., 2022). Satellite networks and the Internet of Vehicles have similar mobility characteristics and diverse terminal requirements (Liu Y et al., 2023). The feature of federated learning is that only a small number of samples can be used for training, and different terminals can generate different models (Guo et al., 2023). Since the performance of each terminal is different (personal digital assistant, phone, computer, etc.), it is better to train a model that meets the real needs. Another feature is that the sample data of the terminal do not need to be collected in batches. Deep learning technology can summarize features or models from a large number of samples, but the storage and computing capabilities in the satellite network are weak, and the topology is dynamic and time-varying, which cannot provide enough learning samples (Miao et al., 2021; Li F et al., 2022; Li J et al., 2023).

Zhou et al. (2023) deployed a three-layer federated learning framework in a highly mobile-connected vehicle network to achieve data sharing between different network devices. Liu ZK et al. (2024) proposed a multi-agent federated reinforcement learning approach for wireless networks composed of cloud servers, drones, and mobile user devices, which calculates the optimal service route based on the user's location, direction, content, and other factors. Lin et al. (2023) applied deep reinforcement learning to solve communication problems in dynamic networks in an integrated network consisting of low orbit satellite networks and sixth-generation (6G) networks. In addition, remote sensing content delivery requires timely and high-performance attributes. When the user's location and receiving terminal change, it is necessary to quickly make a corresponding model. Based on this, this paper proposes a content delivery method with federated reinforcement learning in an information-centric satellite network.

Due to the unique identification of a five-tuple connection (source address, source port, destination address, destination port, and protocol identifier) between the source and destination addresses in a transmission control protocol/Internet protocol (TCP/IP), connection interruptions may occur in dynamic networks with high-speed satellite movement. In addition, TCP/IP does not support continuous transmission or nodule persistence, and can only re-establish connections, while the producer's location changes; thus, there is a high probability of final delivery failure. Although the dynamic routing protocol, namely, open shortest path first (OSPF) (Yang et al., 2023), recalculates and corrects the route after a link is disconnected, the content that needs to be delivered can be obtained only from the source satellite and still faces a delivery failure due to the missing of the link with the source satellite (without a content caching function), as shown in Fig. 3. The routing information protocol (RIP) faces the same issue (Biradar, 2020).

An ICN supports good mobility and has a content caching function, which can calculate the optimal delivery of satellite nodes based on the predictability of satellite movement, avoiding link interruption and delivery failure. The content caching function in the ICN allows multiple satellite nodes to

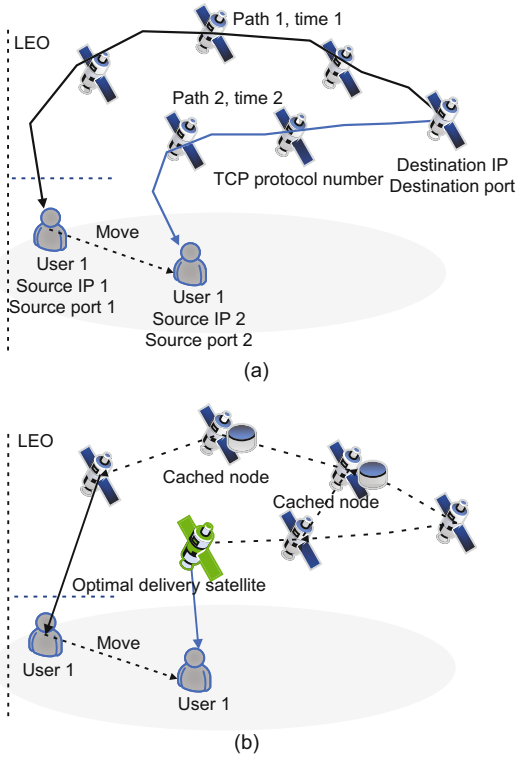

Fig. 3 Content delivery under TCP/IP and ICN approaches: (a) TCP/IP delivering content with a five-tuple connection; (b) ICN delivering content with cached nodes (TCP/IP: transmission control protocol/Internet protocol; ICN: information-centric network; LEO: low Earth orbit)

store the previous content, rendering the ICN more conducive to content delivery, as shown in Fig. 3.

The protocol independent multicast (PIM) approach does not require the maintenance of specialized unicast routing information, thus reducing complexity. However, this approach requires the PIM network be divided into multiple PIM domains for management, with the advantage of transmitting mainly streaming media such as video or audio conferences (Zha et al., 2022). Merling et al. (2023) studied the traffic saving ability of IP multicast (IPMC) and bit index explicit replication (BIER) under various conditions. The performance of traffic saving depends on the network topology, network size, and multicast group size. BIER needs to generate additional traffic. Therefore, the advantages of PIM and BIER in saving traffic are suitable for fixed and stable architecture networks. Lan et al. (2023) and Lu et al. (2023) studied the delivery performance of BIER under IPv6, with its advantages reflected mainly in the fixed delivery of data sources such as live streaming, ultra clear television, and online education.

To sum up, the current satellite network content delivery is still based on finding the optimal transmission path. The consumption of resources is not conducive to the rapid delivery of content, and the satellite network cannot deal with the disadvantages related to the increase of the number of ground users and the deterioration of delivery efficiency. We aim to optimize users' need and the amount of data, and to reduce duplicate data to reduce traffic burstiness.

The scenario is as follows: when an earthquake occurs in a certain area, multiple departments such as emergency (user 1), agriculture (user 2), and transportation (user 3) need relevant geographic images of the earthquake area, which may result in a short-term increase in the number of users. The processing mechanism is as follows: the producer delivers remote sensing images to user 1, and at this time, there are new users: user 2, user 3, $\cdots$, user n.

If the remote sensing images requested by user 2 and user 1 are the same, the optimal cached node is calculated and delivered to user 2.

If the remote sensing images requested by user 3 and user 1 are different but partially identical, the system first uses GeoSOT (GeoSOT is short for geographic coordinate subdividing grid with one dimension integral coding on 2^n-tree) and the bloom filter technology to calculate and query the same remote sensing images, and then copies and delivers the same remote sensing images to user 3 through the optimal cache node. Different remote sensing images are delivered separately by the producer.

Because this mechanism reduces duplicate data throughout the network, it avoids the formation of burst traffic. The advantages are mainly reflected in the following aspects:

In cases of emergencies, ground networks are unreliable and may face interruptions, while satellite networks are less susceptible to disruptions due to terrain irregularity, buildings, and other obstructions when transmitting data. They can achieve low-latency long-distance communication, such as direct connection of mobile phones to satellites.

Due to the lack of communication among ground users, the content is uniformly transmitted to the ground and then delivered, resulting in a large amount of duplicate traffic in the ground network, which may face the risks of congestion and burst traffic formation.

3 System model

The whole system consists of a content producer (providing remote sensing images to users), LEO satellites (name-based routing and caching), and multiple ground users (consumers). Users subscribe to the content related to a certain topic/area (remote sensing image) from the content producer, the content producer delivers the content (remote sensing image) to multiple ground users through the low orbit satellite network, and all these units together form an ICN network, as shown in Fig. 2.

Now, to model the system, the network of the content delivery process is assumed to be a graph consisting of nodes and edges, and the network is represented as a graph $G(\mathcal{N}, \mathcal{E})$, where $\mathcal{N}$ is the set of all nodes in the graph (nodes include routers, switches, communication equipment for ground users, etc.) and $\mathcal{E}$ is the edge set consisting of the interconnected nodes in the graph (Deng et al., 2021; Baldoni et al., 2023).

The content producer is represented as CP, the data flow in the entire network is set as $\mathcal{W}$, and the data flow under time slot s is set as w.

Ground consumers are set as $\mathrm{Ur}=(\mathrm{Ur}_1, \mathrm{Ur}_2, \cdots, \mathrm{Ur}_n)$, where n is the number of consumers.

Each node n is a unit with central processing unit (CPU) processing capability. The number of CPU cores is recorded as Z_n (the larger the number of cores, the stronger the processing capability), and the memory capacity of the node is recorded as M_n (the higher the memory capacity, the more the data that are processed).

The ICN node set (a set with name-based routing and caching) is represented as $\mathcal{V}$, where one ICN node instance is v, and its service rate of requests is β_n^v; the total delay, including the processing delay and queuing delay, at node n is recorded as Y_n^v.

The link in the network is a two-way communication scheme: a link is denoted as $e \in \mathcal{E}$; the link communication from node n to node n' is denoted as $e_{n'}^n$; the data transmission rate in the communication link is denoted as U_e; the traffic into node n is denoted as $\mathcal{E}_n^{\text{plus}}$, the traffic out of the node is recorded as $\mathcal{E}_n^{\text{minus}}$, and the delay on link e includes the transmission delay and the propagation delay. The total delay is recorded as Y_e.

The variable g shows the content from either the same content producer or a different one:

$$g = \begin{cases} 1, & \text{from the same producer,} \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

A data flow, denoted as $w \in \mathcal{W}$, which includes all links in the network (also known as the sequence of data packets traversing the intermediate nodes), refers to all data packets generated at a rate α_w from a source node to a destination node.

Each flow w has its service chain of ICN, denoted as $\text{SC}_w = \{\text{SC}_{w_1 \to w_w} \mid \text{SC}_{w_s} \in \mathcal{V}\}$, and the b^{th} link service of the ICN represents the passage of an orderly sequence of data packets. The length of the SC flow link is denoted as $B_{w_s} = |\text{SC}_{w_s}|$.

The total end-to-end delay of each flow w_s from the source node s_w to the destination node d_w is recorded as Y_w. The end-to-end delay is composed of three parts: the total link delay Y_{li}, the total node delay Y_{no}, and the other delay Y_{ot}.

4 Description of the delivery problem

4.1 Mixed-integer linear programming problem

The content delivery problem in satellite networks can be represented as a mixed-integer linear programming (MILP) model (Qiao et al., 2020). To obtain the optimal solution, we first calculate and find its law in a small-scale network. We focus on calculating and analyzing the number of nodes, the number of generated transmission paths, and the location of nodes.

In addition, we calculate and analyze the solution rules under multiple nodes. We analyze the relationship between the number of nodes and the performance to improve delivery efficiency. Simultaneously, we analyze the flow rate and the relationship between different computing resources (Li L et al., 2020).

Thus, the content delivery problem can be transformed into the problem of selecting the optimal placement location among multiple identical ICN instances.

4.2 Optimization objective

According to the preceding description, the ground users subscribe to the content of a certain area from the content producer; then, the content producer delivers the content to the ground users. The content delivery problem can be converted into traffic corresponding to the network layer, that is, selecting the optimal delivery node for ground users. It is a typical Pareto optimal problem (Marler and Arora, 2010), i.e., how to determine the nodes to replicate the delivery. There are four goals in this approach:

(1) maximize the data flow in the network (deliver more content);

(2) minimize the number of service nodes (with fewer delivery nodes);

(3) minimize the number of CPU cores and memory (fewer computing resources);

(4) minimize link utilization (small number of transmission links).

The goals are to deliver as much content as possible to ground users with the least amount of resources.

An ICN node instance v, at node n, provides data flow w_s of the b^{th} content delivery chain, recorded as $fX_n^v(w_s, b, g)$.

Moreover, in link e, the data flow from node n_b to node n_{b+1} is w_s, and the traffic route from the b^{th} to $(b+1)^{\text{th}}$ link service is expressed as $fX_e(w_s, b, n_b, b+1, n_{b+1}, g)$; the same type of ICN may be on the same node.

Here, i is the number of ICN instances. An ICN instance on node n is recorded as i_n. The optimization goals are as follows:

(1) maximize

$$\sum_{\forall w\in\mathcal{W}}\frac{fX_n^v\left(w_s,b,g\right)}{B_w+\frac{1}{\alpha_w}}+\sum_{\forall w\in\mathcal{W}}B_w; \tag{2}$$

(2) minimize

$$\sum_{\forall n}\frac{i_n}{B_w+\frac{1}{\alpha_w}}; \tag{3}$$

(3) minimize

$$\sum_{\forall n}\frac{i_n}{Z_nM_n}; \tag{4}$$

(4) minimize

$$\sum_{\forall e,\forall w,\forall b}\frac{fX_e\left(w_s,b,n_b,b+1,n_{b+1},g\right)\alpha_w}{U_e+\frac{1}{\alpha_w}}+\frac{i_n}{\sum_{\forall w\in\mathcal{W}}B_w}. \tag{5}$$

4.3 Constraints

To solve the MILP problem, the following constraints must still be satisfied (Promwongsa et al., 2020, 2022):

1. Transmission capacity constraints. Capacity limitation, i.e., the total traffic on any link, does not exceed the maximum transmission capacity of the link:

$$\sum_{\forall w,\forall b}fX_e\left(w_s,b,n_b,b+1,n_{b+1},g\right)\alpha_w<U_e,\quad \forall e\in\mathcal{E},\ U_e>0. \tag{6}$$

The number of CPU cores allocated to ICN instances on any node does not exceed the number of cores, and the memory capacity does not exceed the total memory capacity of the node:

$$\sum_{\forall v}i_nz^v<Z_n,\quad \forall n\in\mathcal{N}, \tag{7}$$

$$\sum_{\forall v}i_nm^v<M_n,\quad \forall n\in\mathcal{N}. \tag{8}$$

Producers actually have more content than what users need:

$$\sum_{\forall v}i_nw<\mathrm{CP}_n,\quad \forall n\in\mathcal{N}. \tag{9}$$

2. ICN node processing constraints. The ICN chain allows the instance capacity required to process traffic not to exceed the total processing capacity of the instance:

$$\sum_{\forall w,\forall b}fX_n^v\left(w_s,b,g\right)\alpha_w<\beta_n^v,\ \forall v\in\mathcal{V},\ \forall n\in\mathcal{N}. \tag{10}$$

The data flow w has at least one type v of ICN instance $X_n^v(w,b)$ in node n:

$$i_n\geq fX_n^v\left(w_s,b,g\right),\ \forall n\in\mathcal{N},\ \forall v\in\mathcal{V},\ \forall w\in\mathcal{W},\ \forall b\in\{1,2,\cdots,B_w\}. \tag{11}$$

During the delivery process, there is only one ICN instance, and only one service is provided:

$$\sum_{\forall n,\forall v}fX_n^v\left(w_s,b,g\right)=1,\quad \forall w\in\mathcal{W},\ \forall b\in\{1,2,\cdots,B_w\}. \tag{12}$$

The links in a satellite dynamic network under each time slot are interconnected as follows:

$$\sum_{\forall n,\forall v}w_s>0,\ \forall w\in\mathcal{W}. \tag{13}$$

3. Delay constraints. Ensure that only the current end-to-end delay requirements cannot exceed the total transmission delay:

$$A_{\mathrm{li}}+A_{\mathrm{no}}+A_{\mathrm{ot}}<A_w,\ \forall w\in\mathcal{W}. \tag{14}$$

In flow w, the total delay of all links is as follows:

$$A_{\mathrm{li}}=\sum_{\forall e,\forall b,\forall n_b}fX_e(w_s,b,n_b,b+1,n_{b+1})A_e. \tag{15}$$

The total delay of all nodes is expressed as

$$A_{\mathrm{no}}=\sum_{\forall b,\forall n,\forall v}fX_n^v(w,b)A_n^v. \tag{16}$$

A_{ot} represents the other delays, such as algorithm calculations, except for all links and nodes.

4. Flow conservation constraint of the intermediate node. It specifies that the intermediate node of the data flow, the routing order specified by the assigned node, and the output link of the same node must be assigned the same order:

$$\sum_{e\in\mathcal{E}_n^{\mathrm{plus}}}fX_e\left(w_s,b,n_b,b+1,n_{b+1},g\right)=\sum_{e\in\mathcal{E}_n^{\mathrm{minus}}}fX_e\left(w_s,b,n_b,b+1,n_{b+1},g\right),\quad \forall w\in\mathcal{W},\ \forall b\in\{1,2,\cdots,B_w\},\ \forall n\in\mathcal{E}. \tag{17}$$

5. Content producer resource constraints. The content (remote sensing images) provided by content producers must be more than the CPU resources required for all services:

$$\frac{\mathrm{CP}}{Z_n M_n} > \sum_{\forall n} \frac{i_n}{Z_n M_n}. \tag{18}$$

The first and last nodes are both served, proving that the data flow reaches the starting (first) node from the destination node:

$$\begin{aligned} &\sum_{e \in \mathcal{E}_{n_b}^{\mathrm{plus}}} fX_e\left(w_s, b, n_b, b+1, n_{b+1}, g\right) \\ =&\sum_{e \in \mathcal{E}_{n_{b+1}}^{\mathrm{minus}}} fX_e\left(w_s, b, n_b, b+1, n_{b+1}, g\right), \\ &\forall e \in \mathcal{E},\ \forall w \in \mathcal{W},\ \forall n_b \in \mathcal{N}. \end{aligned} \tag{19}$$

In the data flow process, forward service $e_{n'}^{n}$ and reverse service $e_{n}^{n'}$ can be assigned only once to avoid forming a loop:

$$\begin{aligned} &fX_{e_{n'}^{n}}\left(w_s, b, n_b\right) = 1, \quad fX_{e_{n}^{n'}}\left(w_s, b, n_b\right) = 0, \\ &\text{or} \\ &fX_{e_{n'}^{n}}\left(w_s, b, n_b\right) = 0, \quad fX_{e_{n}^{n'}}\left(w_s, b, n_b\right) = 1, \\ &\forall e \in \mathcal{E},\ \forall w \in \mathcal{W},\ \forall b \in \{1, 2, \ldots, B_w\},\ \forall n_b \in \mathcal{N}. \end{aligned} \tag{20}$$

The content delivery for avoiding burst traffic involves mainly the following components. The relationship between the following steps is shown in Fig. 4.

1. All users request remote sensing images from the producer satellite.

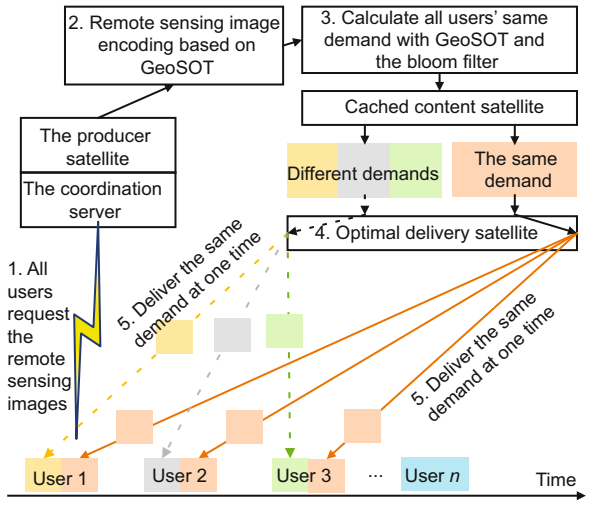

Fig. 4 Principle of avoiding burst traffic and its relationship with each section

2. The producer satellite performs GeoSOT encoding on the remote sensing images requested over a period of time.

3. The producer satellite calculates the same and different requirements from the cached satellite nodes based on GeoSOT and the bloom filter (see Section 1 in the supplementary materials for details).

4. The producer satellite calculates the optimal delivery satellite node and delivers the remote sensing images to all users through the optimal satellite delivery node.

5. The same demand requires only one copy, while different remote sensing images are delivered separately to users one by one. Due to the reduction of network traffic generated by the same demand throughout the network, it reduces burst traffic. Steps are detailed in Section 5 to implement the entire process.

5 Content delivery strategy

5.1 Content delivery process

According to Pfandzelter and Bermbach (2021), 93% of bandwidth can be saved by finding the nodes needed to copy the same content during content delivery. This paper implements content copying and delivery according to this principle. The content delivery process of the satellite network proposed in this paper is shown in Fig. 5. When there is no burst traffic, the process for ground users to obtain content is the same as that in the ICN. When the number of ground users increases sharply and burst traffic occurs, other nodes will be searched for and content delivery is enabled.

In the flowchart of data stream transmission, as shown in Fig. 6, whether the data stream passing

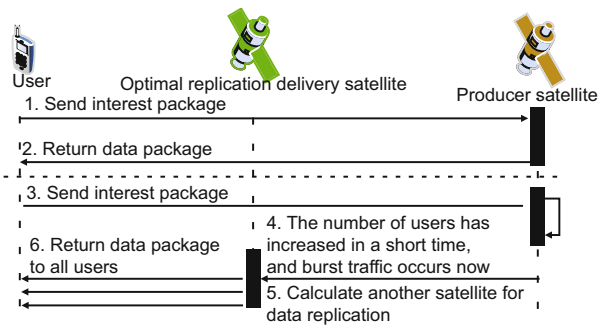

Fig. 5 The content delivery process of the satellite network (the number of ground users has increased in a short time)

through the node is overloaded is first checked, and when there is no overload, ICN default transmission is maintained. When an overload is detected, the scheme will be started, and the optimal replication and delivery node for users on the ground will be selected to replace the original node to accomplish replication and delivery of content.

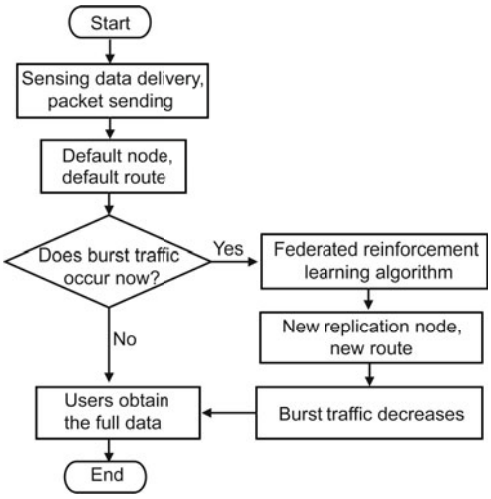

Fig. 6 Flowchart of data flow transmission

1. Initialization

The purposes of the flow cluster module are to find flows with similar paths together and group the flows.

The algorithm first sorts the nodes of all disjoint graphs by the weight, and if the vertices of the edges belong to two disjoint sets, they will be merged into one group.

2. Number of instances

$H(n, n')$ is the shortest service path between nodes n and n'.

The source node is s_{w_s}, and n_1 is the first service instance of the node; then, the service between them is recorded as $H(s, n_1)$.

$P\|n_b, d_{w_s}\|$ indicates the distance between node n_b and the first service flow of the b^{th} flow.

The content delivery problem can be expressed as follows:

$$P\|n_b, d_{w_s}\| = \min\left(H\left(n_b, n_{b+1}\right) + P\left(n_{b+1}, d_{w_s}\right)\right). \tag{21}$$

5.2 Optimal replication nodes

In Section 5.1, we discussed the optimization of Eq. (21) during the process of delivering content from the producers to ground users. This subsection transforms Eq. (21) into a Markov decision process (MDP) and applies federated reinforcement learning to solve it. Considering the above-mentioned MDP problem, and finally aiming at the disadvantages of weak computing power, rapid increase of the number of ground users in the satellite network, availability of only a few training samples, and difficult real-time decision-making, a solution comprising federated reinforcement learning is proposed.

The optimal replication delivery node refers to the node on which the content is replicated and delivered to users when burst traffic occurs. The goal is to reduce data caused by unsuitable nodes and to increase the consumption of resources. Finding the optimal replication nodes is the key. We then solve the multi-objective optimization in Section 5.1 through federated reinforcement learning to obtain the optimal replication delivery node.

Federated reinforcement learning is based on deep reinforcement learning. It consists of multiple agents, and the agents follow the Markov training algorithm (Xu et al., 2022). Each agent trains its own neural network model based on a part of the sample, and all the model data are uploaded to the coordination server. After comprehensive processing by the coordination server, the data are returned to each agent (Xie and Song, 2023). The sample of the whole process is always based on the ground user, avoiding information leakage caused by various reasons. The whole process requires no high computing or storage (Qi et al., 2021). Its model (Fig. 7) can be expressed as a tuple $(k, S_t, \pi_i, a_i, R_i, S_{i+1})$, where in each time slot, the following notations apply:

(1) k is the agent number;

(2) S_t is the state of an agent at time t;

(3) π_i is the control policy of the i^{th} agent at time t;

(4) a_i is an action generated by the i^{th} agent according to the control strategy π_i;

(5) R_i is the reward or punishment that the agent obtains after performing action a_i in state S_t (see Algorithm 1 for details);

(6) S_{i+1} is the new state of the agent at the next time slot after action a_i is executed.

1. State

In each time t, in the LEO satellite network, any ICN node is an agent, and the agents interact to select different states to represent the current

environment. Through continuous trial and error, according to the obtained feedback, information is iteratively optimized to finally solve the Markov problem, as shown in Fig. 8. In each time slot one state is $(\mathrm{Node}(t), \mathrm{User}(t), \mathrm{Data}(t))$, where

(1) $\mathrm{Node}(t)$ is the delivery satellite number in LEO at time t,

(2) $\mathrm{User}(t)$ is the ground user, and

(3) $\mathrm{Data}(t)$ is the size of content that needs to be delivered.

To sum up, we have

$$S(t)=\begin{bmatrix} \mathrm{Node}_1(t) & \mathrm{Node}_2(t) & \cdots & \mathrm{Node}_n(t) \\ \mathrm{User}_1(t) & \mathrm{User}_2(t) & \cdots & \mathrm{User}_n(t) \\ \mathrm{Data}_1(t) & \mathrm{Data}_2(t) & \cdots & \mathrm{Data}_n(t) \end{bmatrix}. \tag{22}$$

2. Action

The term action refers to the replication delivery node that an agent needs to select after comprehensive decision-making in time t, and the selection of the node is a 0–1 linear programming problem. The action A decision of agent n at time t is

$$A(t)=[\text{Node list in LEO}]. \tag{23}$$

"Node list in LEO" refers to a node that needs to be replicated and delivered in the LEO satellite network:

$$A(t)=[a_{\lambda_1}(t), a_{\lambda_2}(t), \cdots, a_{\lambda_n}(t)]. \tag{24}$$

$a_{\lambda_i}(t)=1$ indicates that the agent chooses this node for copying and delivery, and $a_{\lambda_i}(t)=0$ indicates that this node is not selected; the selection of the node directly affects the reward value.

3. Reward

The reward is responsible for calculating the effectiveness benefits of the delivery node positions selected by the agents. There are both cooperative and competitive relationships between agents; thus, we apply a mixed reward mechanism to avoid the inertia caused by multiple agents using the same reward. The mixed reward function refers to the environment that after an agent performs an action, the algorithm will simultaneously give the agent a separate reward for all agent federated actions (Nguyen et al., 2021). Competition among multiple agents can also enhance the overall return of federated actions. We minimize the amount of computing resources and the number of paths by changing actions (which are the delivery nodes) to maximize the amount of content sent, as shown in Eq. (25):

$$Q_n(\pi_n)=E_{\pi_n}\left(\sum_{t=1}^{t+T}\gamma^t R_{n,t}\right). \tag{25}$$

Eq. (25) is for each agent. It uses the reward function at the construction time $R_{n,t}$, and obtains the desired value function through discount accumulation and summation. In Eq. (25), the mean value of all cumulative rewards produced by the strategy is E in time T. The discount coefficient $\gamma\in[0,1]$ at that time, with $\gamma=0$ indicating that the agent makes decisions based on the rewards obtained at the current time and $\gamma=1$ showing that the agent focuses on long-term returns. π_n is the actor network with parameter n, and Q_n is the critic network with parameter n.

Algorithm 1 uses federated reinforcement learning to solve the optimization problem described in Section 5.1 and finds the nodes that need to be delivered.

In lines 17–18, τ is a relatively small number, and ω is the parameter of the critic network.

Line 20 represents the gradient of the deterministic policy. The number of agents is NT, and the policy network is u (Wu Q et al., 2021). Each agent

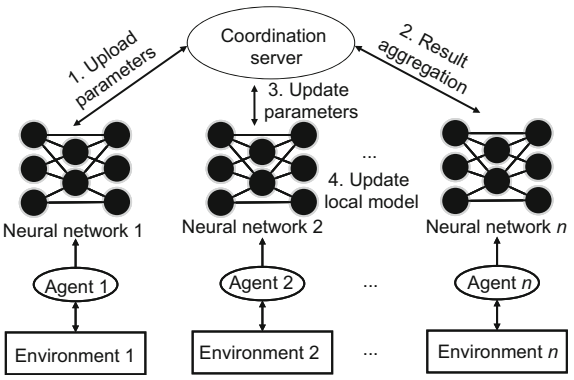

Fig. 7 Federated reinforcement learning model

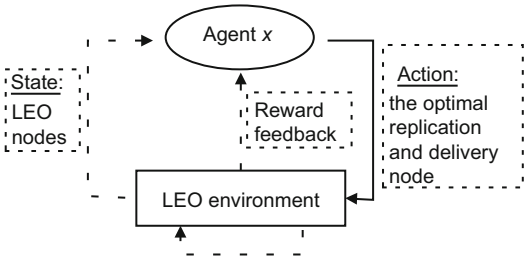

Fig. 8 Interaction between multiple agents and the environment in a satellite network (LEO: low Earth orbit)

Algorithm 1 Calculation of the optimal replication node based on DDPG

Input: network topology $G(\mathcal{N},\mathcal{E})$ and initialized experience replay buffer D.
Output: node n, $n\in\mathcal{N}$.
1: Initialize the actor network and critic network of each agent randomly and initialize the random process as the introduced noise.
2: Receive the initial state sx at episode T.
3: **for** each episode != NULL **do**
4: Initialize a random process N for action exploration.
5: Obtain the initial values of all agents x.
6: timeStart=now.
7: **for** each slot $t\in[1,T]$ & t != NULL **do**
8: **for** each agent != NULL **do**
9: Select an action with the current policy $a=N$.
10: Execute action a and obtain rewards and new observations x'.
11: Store (sx,a,x') in replay buffer D.
12: Update $x\leftarrow x'$.
13: Randomly select some data from D.
14: Train the critic network of each agent.
15: Train the actor network of each agent.
16: For each agent, update the target actor network and target critic network.
17: Update $\omega^{-}\leftarrow\tau\omega+(1-\tau)\,\omega^{-}$.
18: Update $\theta^{-}\leftarrow\tau\theta+(1-\tau)\,\theta^{-}$.
19: Update the actor according to the gradient of the policy strategy.
20: $\nabla_{\theta}J\approx\frac{1}{\mathrm{NT}}\sum_{i=1}^{\mathrm{NT}}\nabla_{\theta}u_{\theta}(s_i)\nabla_{a}Q_{\omega}\left(s_i,a\right)|_{a=\mu_{\theta}a(s_i)}$.
21: **return** optimal replication node n.
22: **end for**
23: **end for**
24: **if** now-timeStart<maximum deadline **then**
25: Return the maximum value of expectation in Eq. (25).
26: **end if**
27: **end for**

follows a separate deep deterministic policy gradient (DDPG) (Kwon et al., 2020).

Line 21 returns the optimal replication node as the last result.

Lines 24–26 calculate the maximum value of expectation in Eq. (25) as the reward value.

The complexity of Algorithm 1 is the optimal replication node calculated by federated reinforcement learning. According to Hazra et al. (2023), the complexity of federated reinforcement learning is related to the number of clients, i.e., $O(N^3)$.

The complexity of Algorithm S1 in the supplementary materials depends mainly on the number of users.

The complexity of Algorithm S2 in the supplementary materials depends mainly on the number of matrices.

In recent years, with the rapid development of aerospace technology, satellite computing capabilities have continued to increase. Conducting large-scale calculations on satellites is no longer a bottleneck problem (Schwaller et al., 2019). For example, in 2022, the computing power of the Chaohu 1 satellite in orbit launched by Changsha Tianyi Company has reached 4 tera operations per second (TOPS). Therefore, although the algorithm proposed in this paper is tested and verified on local computers, it can be deployed and applied to satellites based on its complexity. Here, 1 TOPS (10^{12}) represents the CPU's ability to process 1 trillion operations per second. Computing power of state-of-the-art CPU models is shown in Table 1.

6 Performance evaluation

In this section, the performance evaluation of the strategy proposed in this paper is carried out, and the experimental procedure is built according to Mamatas et al. (2023). It is a mixed constellation composed of Iridium NEXT (https://www.iridium.com/blog/iridium-next-review/) and a remote sensing satellite. The Iridium NEXT network with 66 satellites undertakes communication transmission tasks. The remote sensing satellite is responsible for taking images, i.e., the content provider. The detailed parameters are shown in Table 2 and the ground station is in Melbourne, Australia.

6.1 Experimental setup

According to Table 2, the constellation is set using the STK (short for Ansys Systems Tool Kit) software (https://www.ansys.com/products/missions/ansys-stk) and the satellite orbit parameters are derived; the network topology is configured in Mini-OF-CCNx (https://github.com/marcialf/mininet-of-ccnx) according to the parameters, multiple clients are connected to the constellation, forming an LEO satellite network transmission system and realizing the entire process of ground users subscribing to satellite remote sensing images, and the performance of the algorithm is evaluated according to parameters such as transmission delay and throughput.

1. Computer: Intel® Core™ i5 12400F CPU @2.50 GHz×6 processors, 16 GB memory.

Table 1 Satellite central processing unit (CPU) computing power

CPU model	Year	Country/Region	Frequency	Computing power	Architecture
DAHLIA	2019	Europe	1.6 GHz	7000 DMIPS	ARM
HPSC	2019	USA	800 MHz	7360 DMIPS	ARM
Yulong810	2021	China	1 GHz	12 TOPS	ARM
GRID-AICore	2022	China	–	6 W@4 TOPS	AI
Tianzhi-2D	2023	China	–	40 TOPS	SDN
WJ-1A	2023	China	–	80 TOPS	AI

DMIPS: dhrystone million instructions per second; TOPS: tera operations per second; ARM: advanced RISC machine; AI: artificial intelligence; SDN: software-defined network. DAHLIA: https://dahlia-h2020.eu/wp-content/uploads/2022/07/OBDP-2021-NG-Ultra-final.pdf; HPSC: https://ieeexplore.ieee.org/document/8742163; Yulong810: https://www.163.com/dy/article/G9FV52KM0512B07B.html; GRID-AICore: https://www.sohu.com/a/526919364_121124374; Tianzhi-2D: https://roll.sohu.com/a/630207350_121118997; WJ-1A: https://36kr.com/p/2449577694107528

Table 2 Satellite constellations and ICN parameters

Parameter	Value
Satellite–ground bandwidth (Mb/s)	60
Altitude (km)	780
Number of content producers	1
Inter-satellite link delay (ms)	40
Eccentricity	0
Inclination (°)	28.5
Argument of perigee	0
RAAN	0
True anomaly	0
Slot interval (s)	60
Zipf traffic	0.75
Content store strategy	Least used
Maximum pending interest table	30 000

RAAN: right ascension of the ascending nodes

2. Operating system: Ubuntu 22.10.

3. Analysis tool: Wireshark v3.6.3, used for network protocol analysis, routing analysis, data packet analysis, etc.

4. Network traffic simulation tool: Mahimahi is used to simulate network parameters in the experiments, and it can record and playback operations.

5. Deep learning framework: Golang auqlue v1.0.3 is used to implement federated reinforcement learning.

6. Satellite network simulation: SILLEO-SCNS (https://github.com/Ben-Kempton/SILLEO-SCNS).

Remote sensing images are generated in real time in the content producer satellite, and the content popularity conforms to the Zipf delivery (Cao et al., 2020). In addition, we construct a dynamic satellite network using algorithms (see Section 2 in the supplementary materials for details).

The experiment compares the performances of IP, basic ICN, social attributes based content delivery (SACD), ICN/SDN, GMR, multicast, and the proposed method. To eliminate the influence of other factors on the experimental results, the average value from 20 experiments is taken in the statistical results.

1. IP is the traditional integrated delivery system of IPv6 and CDN (Wang YT et al., 2024).

2. Basic ICN is the delivery system of unmodified ICN (Chaudhary et al., 2023).

3. SACD is the delivery system in the vehicular content-centric network (Wang XN and Chen, 2023).

4. ICN/SDN is the content delivery system of the software-defined ICN in the satellite network (Mamatas et al., 2023).

5. GMR is a graph neural network (GNN) enabled multipath routing (GMR) process in LEO satellite networks. This algorithm comprehensively considers available transmission paths and uneven traffic density (Huang et al., 2023).

6. Multicast is a 6G multicast video transmission system (Chukhno et al., 2023).

The burst traffic is constructed according to the method in Wang YT et al. (2024), and the dataset is WiGLE (https://wigle.net).

6.2 Analysis of experimental results

6.2.1 The selected path

The selected path refers to the path that is in the working state after the ground user subscribes to the image of a certain area. According to the previous analysis, when the number of selected paths is too large, the resource consumption on the satellite will be increased, and when the number of selected paths is too small, the delivery of content cannot be satisfied.

When IP or multicast is used, transmission fails due to the influence of satellite dynamics, as shown

in Fig. 9a, and the ground users finally fail to obtain the content.

Basic ICN is also unable to cope with the impact of satellite dynamics, and the transmission fails, as shown in Fig. 9b.

SACD/GMR and ICN/SDN can better cope with the impact of satellite dynamics, but there are many repeated data, and the ground users can successfully obtain the content, as shown in Figs. 9c and 9d.

The proposed method can better cope with the impact of satellite dynamics, without repeated data, and the ground users can successfully acquire content, as shown in Fig. 9e.

The proposed method reduces the transmission of duplicate data by replicating nodes, which is less prone to network congestion, resulting in high efficiency. The ICN/SDN architecture has high duplicate data and is highly prone to congestion.

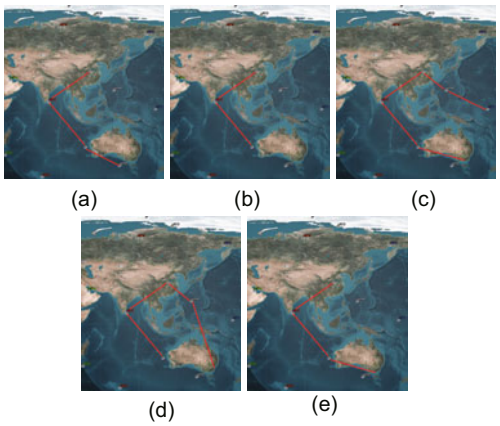

Fig. 9 Selected path(s) under different algorithms: (a) IP/multicast; (b) basic ICN; (c) SACD/GMR; (d) ICN/SDN; (e) proposed method

6.2.2 Average delivery delay

The average delivery delay (ADD) refers to the ratio of the number of data packets delivered successfully (n_{success}) to the total number of data packets (n_{total}) in the content delivery:

$$\text{ADD} = \frac{n_{\text{success}}}{n_{\text{total}}}. \tag{26}$$

As shown in Fig. 10, under IP, multicast, basic ICN, and ICN/SDN approaches, it is impossible to cope with the impact of network dynamics on delivery, and the average delivery delay fluctuates and is extremely unstable; the average delivery delay is significantly higher than that of GMR and the proposed method. Because SACD, GMR, and the proposed method optimize and balance the relationship between the traffic demand and the amount of data, the average delivery delay is stable.

Fig. 11 shows that as the packet loss rate increases, the average delivery delay increases as well. Under different packet loss rates, the impact on IP is the greatest, showing extreme instability, followed by the multicast.

Furthermore, we compare the delivery performance under different remote sensing images, analyzing the greater role played by GeoSOT encoding. The normalization function is defined as follows

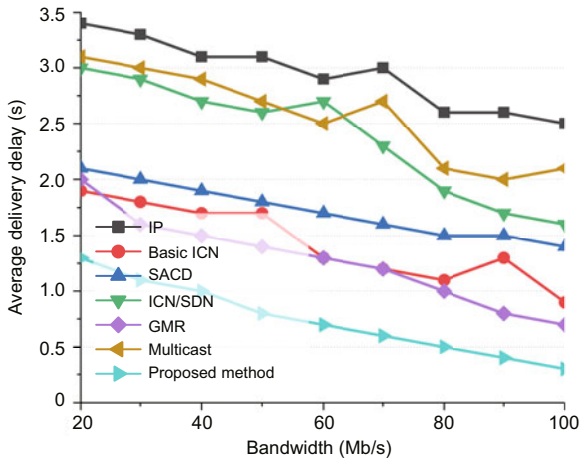

Fig. 10 Average delivery delay under different bandwidths

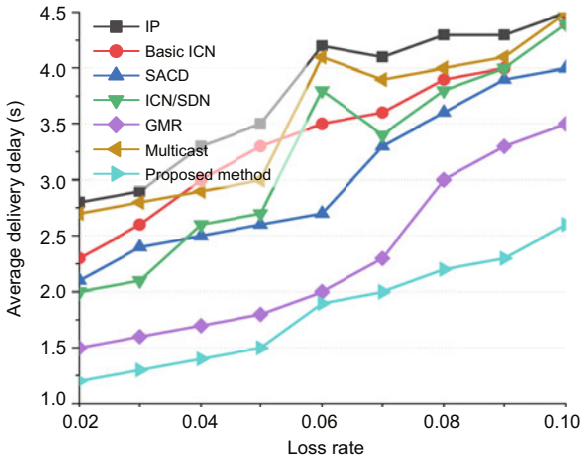

Fig. 11 Average delivery delay under different packet loss rates

(Deng et al., 2021):

$$\rho_{\text{number}} = \frac{\text{delay}_{\text{current}} - \text{delay}_{\text{min}}}{\text{delay}_{\text{max}} - \text{delay}_{\text{min}}}. \quad (27)$$

In Eq. (27), different numbers of remote sensing images (number $\in \{200, 400, 600, \cdots, 1000\}$) are used to simulate the various remote sensing images of different users, such as some users having the same request. Here, $\text{delay}_{\text{current}}$ is the current delay, and $\text{delay}_{\text{max}}$ and $\text{delay}_{\text{min}}$ are the maximum and minimum delay respectively. As shown in Fig. 12, the normalized delivery delay decreases with the increase in the number of remote sensing images. The delay of the proposed method is the highest, indicating that it involves the largest number of data packets transmitted and successfully handles merge requests.

6.2.3 Average completion time

The average completion time refers to the time required for IP, basic ICN, and the proposed method to deliver the same content under the same network conditions. As shown in Fig. 13, the average completion time of the algorithms increases with the increase in the size of the delivery content, and the average completion time of IP is the longest. The proposed method can cope with the impact of network dynamics on the delivery, so the average completion time is lower than that with the two other schemes. When the delivery content exceeds 700M, the average completion time of IP is longer than that of basic ICN (before 700M, the average completion time of basic ICN is the longest in most cases), and the average completion time of the proposed method is the shortest throughout the process.

6.2.4 Number of packets

When evaluating content delivery, there is a possibility of retransmission, and the number of various data packets is counted. As shown in Fig. 14, the probability of retransmission in the proposed method is low.

6.2.5 Success rate of content acquisition

As shown in Fig. 15, the success rate of content acquisition refers to the state in which the content acquisition needs of subscribers can be finally met. The failure of content delivery due to the influence of network dynamics and retransmissions is considered

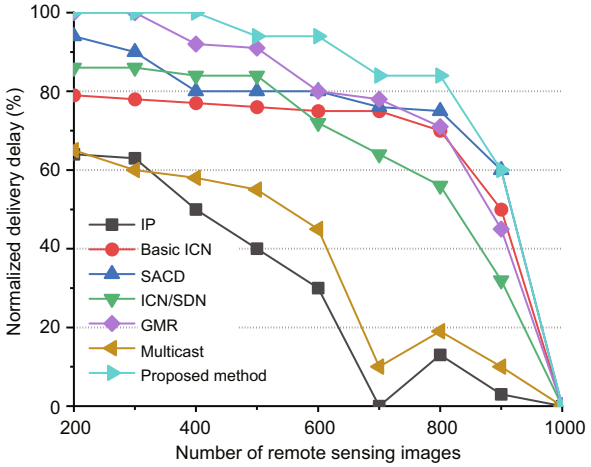

Fig. 12 Normalized delivery delay of different numbers of remote sensing images

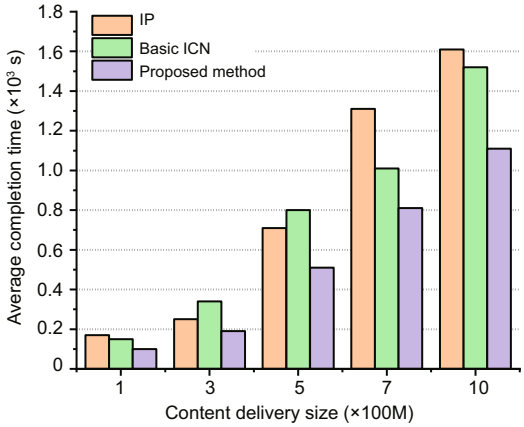

Fig. 13 Comparison of average completion time (References to color refer to the online version of this figure)

the failure of user content acquisition. Considering various factors, the content acquisition success rate of the proposed method is the highest.

The higher the success rate of content acquisition, the greater the probability of obtaining complete remote sensing images. Other algorithms may fail to obtain remote sensing images due to the dynamic nature of satellite networks and network congestion.

6.2.6 Content locating rate

Content locating rate refers to the ratio of the retrieved content size to the time. With the continuous development of time, more and more remote sensing images generated by satellites in orbit will form massive data. How to quickly locate the images that ground users need has become a key problem. This paper adds time and spatial location attributes

to achieve efficient transmission of satellite routing. This experiment compares the content locating rate with that of the basic ICN, as shown in Fig. 16. A higher content locating rate indicates a shorter locating time. The encoded locating rate is significantly higher than that of the basic ICN.

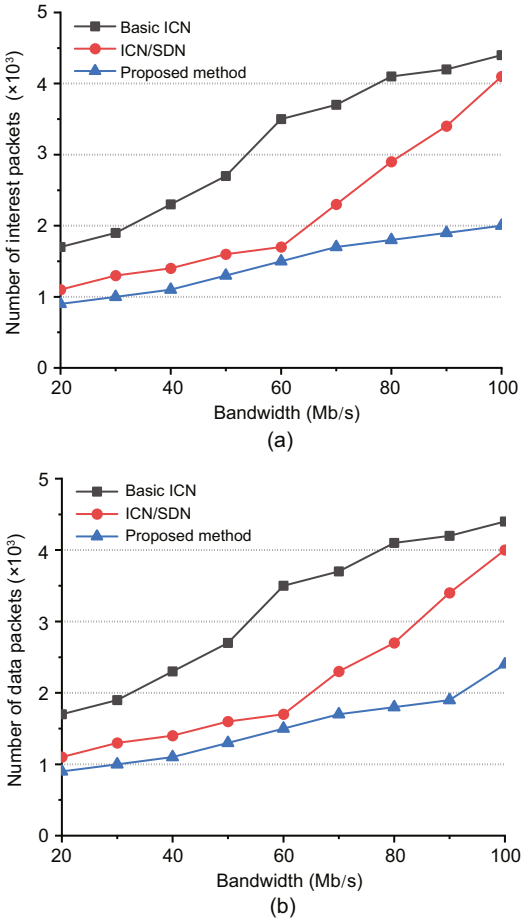

Fig. 14 Statistics on the numbers of interest packets (a) and data packets (b)

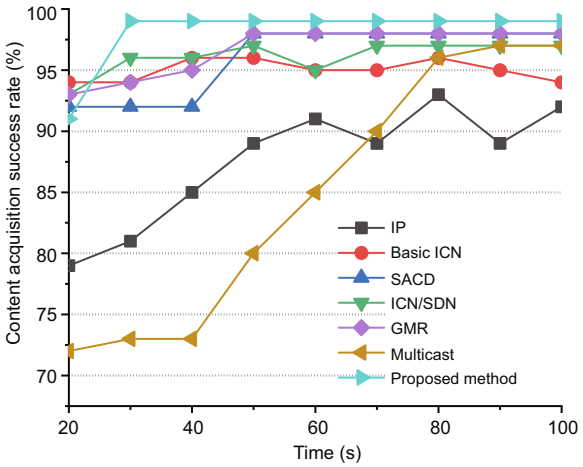

Fig. 15 Content acquisition success rate

6.2.7 Load balancing

There are many paths in the whole LEO satellite network. Due to the limited satellite computing and storage resources, it is necessary to make the working path loads balanced as much as possible from the perspective of saving bandwidth resources. This experiment counts the flow delivery of inter satellite links over a period of 550 s, and the flow delivery thermodynamic diagram of each working path is shown in Fig. 17, where x-axis is the time 50–550 s, and y-axis is the number of working paths of x-axis at a certain time. The color in the heat map changes from light to dark, indicating that the value of traffic in the path changes from small to large. The traffic in the selected path increases steadily without significant fluctuations. In addition, the traffic in the working path is roughly the same, indicating that the routing algorithm in this paper has stable transmission. Within 500 s, the maximum traffic difference

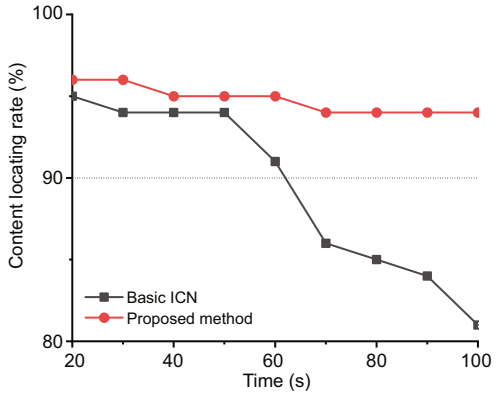

Fig. 16 Content locating rate under different schemes

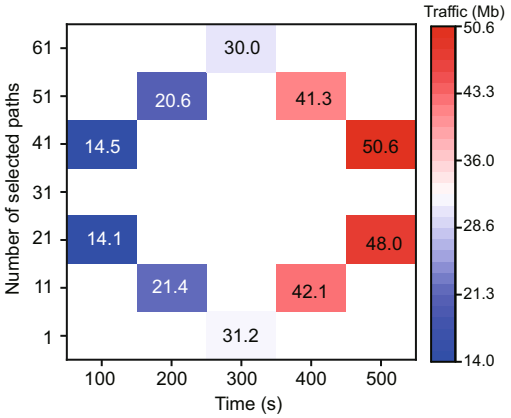

Fig. 17 Traffic on the selected paths

between the two paths selected by the algorithm is about 2 Mb. The proposed method has achieved load balancing (the delivery of path traffic in the working state is equivalent).

7 Conclusions and future work

ICN is an important direction in future network architectures, and efficient transmission is achieved by decoupling location and content. Aiming at the drawback of performance degradation with the increase in the number of users during content delivery for traditional satellite networks, we use federated reinforcement learning to generate an optimal replication delivery satellite node, which effectively reduces network transmission data and avoids burst traffic. It can easily cope with the impact of a sudden surge of users on the delivery of remote sensing images. In future work, other strategies for content delivery under dynamic networks will be studied.

Contributors

Ziyang XING and Xiaoqiang DI designed the research. Hui QI and Jing CHEN processed the data. Ziyang XING drafted the paper. Jinhui CAO, Jinyao LIU, Xusheng LI, Zichu ZHANG, Yuchen ZHU, Lei CHEN, Kai HUANG, and Xinghan HUO helped organize the paper. Ziyang XING and Xiaoqiang DI revised and finalized the paper.

Conflict of interest

All the authors declare that they have no conflict of interest.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

Aung N, Dhelim S, Chen LM, et al., 2023. VeSoNet: traffic-aware content caching for vehicular social networks using deep reinforcement learning. *IEEE Trans Intell Transp Syst*, 24(8):8638-8649. https://doi.org/10.1109/TITS.2023.3250320

Baldoni G, Quevedo J, Guimaraes C, et al., 2023. Data-centric service-based architecture for edge-native 6G network. *IEEE Commun Mag*, early access. https://doi.org/10.1109/MCOM.001.2300178

Bilal M, Kang SG, 2019. Network-coding approach for information-centric networking. *IEEE Syst J*, 13(2):1376-1385. https://doi.org/10.1109/JSYST.2018.2862913

Biradar AG, 2020. A comparative study on routing protocols: RIP, OSPF and EIGRP and their analysis using GNS-3. Proc 5th IEEE Int Conf on Recent Advances and Innovations in Engineering, p.1-5. https://doi.org/10.1109/ICRAIE51050.2020.9358327

Cao Y, Zhu YF, Lv JF, et al., 2020. Research on in-network caching mechanisms for space-integrated-ground content delivery service. *Space-Integr-Ground Inform Netw*, 1(2):48-56. https://doi.org/10.11959/j.issn.2096-8930.20200207

Chaudhary P, Hubballi N, Kulkarni SG, 2023. eNCache: improving content delivery with cooperative caching in named data networking. *Comput Netw*, 237:110104. https://doi.org/10.1016/j.comnet.2023.110104

Chukhno N, Chukhno O, Pizzi S, et al., 2023. Approaching 6G use case requirements with multicasting. *IEEE Commun Mag*, 61(5):144-150. https://doi.org/10.1109/MCOM.001.2200659

Deng SG, Zhang C, Li C, et al., 2021. Burst load evacuation based on dispatching and scheduling in distributed edge networks. *IEEE Trans Parall Distrib Syst*, 32(8):1918-1932. https://doi.org/10.1109/TPDS.2021.3052236

Guo Q, Tang FX, Kato N, 2023. Federated reinforcement learning-based resource allocation in D2D-enabled 6G. *IEEE Netw*, 37(5):89-95. https://doi.org/10.1109/MNET.122.2200102

Hazra A, Donta PK, Amgoth T, et al., 2023. Cooperative transmission scheduling and computation offloading with collaboration of fog and cloud for industrial IoT applications. *IEEE Int Things J*, 10(5):3944-3953. https://doi.org/10.1109/JIOT.2022.3150070

Huang YX, Yang D, Feng BH, et al., 2023. A GNN-enabled multipath routing algorithm for spatial-temporal varying LEO satellite networks. *IEEE Trans Veh Technol*, early access. https://doi.org/10.1109/TVT.2023.3333848

Kwon D, Jeon J, Park S, et al., 2020. Multiagent DDPG-based deep learning for smart ocean federated learning IoT networks. *IEEE Int Things J*, 7(10):9895-9903. https://doi.org/10.1109/JIOT.2020.2988033

Lan SF, Ma PY, Yang GM, et al., 2023. Research and verification of new multicast BIER IPv6 technology in IP network. Proc IEEE Int Conf on Sensors, Electronics and Computer Engineering, p.1320-1325. https://doi.org/10.1109/ICSECE58870.2023.10263464

Li F, Shen BW, Guo JL, et al., 2022. Dynamic spectrum access for Internet-of-Things based on federated deep reinforcement learning. *IEEE Trans Veh Technol*, 71(7):7952-7956. https://doi.org/10.1109/TVT.2022.3166535

Li J, Xue KP, Liu JQ, et al., 2020. An ICN/SDN-based network architecture and efficient content retrieval for future satellite-terrestrial integrated networks. *IEEE Netw*, 34(1):188-195. https://doi.org/10.1109/MNET.2019.1900138

Li J, Yang ZP, Wang XW, et al., 2023. Task offloading mechanism based on federated reinforcement learning in mobile edge computing. *Dig Commun Netw*, 9(42):492-504. https://doi.org/10.1016/j.dcan.2022.04.006

Li L, Shi D, Hou RH, et al., 2020. Energy-efficient proactive caching for adaptive video streaming via data-driven optimization. *IEEE Int Things J*, 7(6):5549-5561. https://doi.org/10.1109/JIOT.2020.2981250

Li X, Lu LY, Ni W, et al., 2022. Federated multi-agent deep reinforcement learning for resource allocation of vehicle-to-vehicle communications. *IEEE Trans Veh Technol*, 71(8):8810-8824. https://doi.org/10.1109/TVT.2022.3173057

Lin SC, Lin CH, Chu LC, et al., 2023. Enabling resilient access equality for 6G LEO satellite swarm networks. *IEEE Int Things Mag*, 6(3):38-43. https://doi.org/10.1109/IOTM.001.2200272

Liu JY, Yao WB, Wang C, et al., 2023. Provisioning network slice for mobile content delivery in uncertain MEC environment. *Comput Netw*, 224:109629. https://doi.org/10.1016/j.comnet.2023.109629

Liu S, Huang JW, Jiang WC, et al., 2021. Reducing traffic burstiness for MPTCP in data center networks. *J Netw Comput Appl*, 192:103169. https://doi.org/10.1016/j.jnca.2021.103169

Liu Y, Jiang L, Qi Q, et al., 2023. Energy-efficient space–air–ground integrated edge computing for Internet of Remote Things: a federated DRL approach. *IEEE Int Things J*, 10(6):4845-4856. https://doi.org/10.1109/JIOT.2022.3220677

Liu ZK, Garg N, Ratnarajah T, 2024. Multi-agent federated reinforcement learning strategy for mobile virtual reality delivery networks. *IEEE Trans Netw Sci Eng*, 11(1):100-114. https://doi.org/10.1109/TNSE.2023.3292570

Lu L, Li Q, Zhao D, et al., 2023. Hawkeye: a dynamic and stateless multicast mechanism with deep reinforcement learning. Proc IEEE Conf on Computer Communications, p.1-10. https://doi.org/10.1109/INFOCOM53939.2023.10228869

Luglio M, Romano SP, Roseti C, et al., 2019. Service delivery models for converged satellite-terrestrial 5G network deployment: a satellite-assisted CDN use-case. *IEEE Netw*, 33(1):142-150. https://doi.org/10.1109/MNET.2018.1800020

Mamatas L, Demiroglou V, Kalafatidis S, et al., 2023. Protocol-adaptive strategies for wireless mesh smart city networks. *IEEE Netw*, 37(2):136-143. https://doi.org/10.1109/MNET.002.2200347

Marler RT, Arora JS, 2010. The weighted sum method for multi-objective optimization: new insights. *Struct Multid Optim*, 41(6):853-862. https://doi.org/10.1007/s00158-009-0460-7

Merling D, Stüber T, Menth M, 2023. Efficiency of BIER multicast in large networks. *IEEE Trans Netw Serv Manag*, 20(4):4013-4027. https://doi.org/10.1109/TNSM.2023.3262294

Miao QY, Lin H, Wang XD, et al., 2021. Federated deep reinforcement learning based secure data sharing for Internet of Things. *Comput Netw*, 197:108327. https://doi.org/10.1016/j.comnet.2021.108327

Narayanan A, Ramadan E, Zhang ZL, 2018. OpenCDN: an ICN-based open content distribution system using distributed actor model. Proc IEEE Conf on Computer Communications Workshops, p.268-273. https://doi.org/10.1109/INFCOMW.2018.8406937

Nguyen TG, Phan TV, Hoang DT, et al., 2021. Federated deep reinforcement learning for traffic monitoring in SDN-based IoT networks. *IEEE Trans Cogn Commun Netw*, 7(4):1048-1065. https://doi.org/10.1109/TCCN.2021.3102971

Pfandzelter T, Bermbach D, 2021. Edge (of the Earth) replication: optimizing content delivery in large LEO satellite communication networks. Proc IEEE/ACM 21st Int Symp on Cluster, Cloud and Internet Computing, p.565-575. https://doi.org/10.1109/CCGrid51090.2021.00066

Promwongsa N, Abu-Lebdeh M, Kianpisheh S, et al., 2020. Ensuring reliability and low cost when using a parallel VNF processing approach to embed delay-constrained slices. *IEEE Trans Netw Serv Manag*, 17(4):2226-2241. https://doi.org/10.1109/TNSM.2020.3029108

Promwongsa N, Ebrahimzadeh A, Glitho RH, et al., 2022. Joint VNF placement and scheduling for latency-sensitive services. *IEEE Trans Netw Sci Eng*, 9(4):2432-2449. https://doi.org/10.1109/TNSE.2022.3163927

Qi JJ, Zhou QH, Lei L, et al., 2021. Federated reinforcement learning: techniques, applications, and open challenges. *Intell Robot*, 1(1):18-57. https://doi.org/10.20517/ir.2021.02

Qiao GH, Leng SP, Maharjan S, et al., 2020. Deep reinforcement learning for cooperative content caching in vehicular edge computing and networks. *IEEE Int Things J*, 7(1):247-257. https://doi.org/10.1109/JIOT.2019.2945640

Rastegar SH, Abbasfar A, Shah-Mansouri V, 2020. Rule caching in SDN-enabled base stations supporting massive IoT devices with bursty traffic. *IEEE Int Things J*, 7(9):8917-8931. https://doi.org/10.1109/JIOT.2020.3000393

Schwaller B, Holtzman S, George AD, 2019. Emulation-based performance studies on the HPSC space processor. Proc IEEE Aerospace Conf, p.1-11. https://doi.org/10.1109/AERO.2019.8742163

Wang J, Rao SY, Liu Y, et al., 2023. Load balancing for heterogeneous traffic in datacenter networks. *J Netw Comput Appl*, 217:103692. https://doi.org/10.1016/j.jnca.2023.103692

Wang W, Zhou CH, He HL, et al., 2020. Cellular traffic load prediction with LSTM and Gaussian process regression. Proc IEEE Int Conf on Communications, p.1-6. https://doi.org/10.1109/ICC40277.2020.9148738

Wang XN, Chen XL, 2023. Social attributes-based content delivery for sparse vehicular content-centric network. *IEEE Trans Intell Transp Syst*, 24(12):14406-14414. https://doi.org/10.1109/TITS.2023.3292216

Wang YT, Han XF, Jin SF, 2024. Performance analysis of a VM-PM repair strategy in MEC-enabled wireless systems with bursty traffic. *IEEE Trans Veh Technol*, 73(1):1146-1161. https://doi.org/10.1109/TVT.2023.3300975

Wu F, Liu XL, Wang J, et al., 2022. Research on application of space rapid response launch system based on data link. Int Conf on Neural Networks, Information, and Communication Engineering, p.353-360. https://doi.org/10.1117/12.2639282

Wu Q, Chen X, Zhou Z, et al., 2021. Deep reinforcement learning with spatio-temporal traffic forecasting for data-driven base station sleep control. *IEEE/ACM Trans Netw*, 29(2):935-948. https://doi.org/10.1109/TNET.2021.3053771

Xie ZJ, Song SH, 2023. FedKL: tackling data heterogeneity in federated reinforcement learning by penalizing KL divergence. *IEEE J Sel Areas Commun*, 41(4):1227-1242. https://doi.org/10.1109/JSAC.2023.3242734

Xu MR, Peng JL, Gupta B, et al., 2022. Multiagent federated reinforcement learning for secure incentive mechanism in intelligent cyber–physical systems. *IEEE Int Things J*, 9(22):22095-22108. https://doi.org/10.1109/JIOT.2021.3081626

Yang H, Guo BL, Xue XW, et al., 2023. Interruption tolerance strategy for LEO constellation with optical inter-satellite link. *IEEE Trans Netw Serv Manag*, 20(4):4815-4830. https://doi.org/10.1109/TNSM.2023.3274638

Yu MX, Pi YB, Tang AM, et al., 2023. Coordinated parallel resource allocation for integrated access and backhaul networks. *Comput Netw*, 222:109533. https://doi.org/10.1016/j.comnet.2022.109533

Zha YL, Cui PS, Hu YX, et al., 2022. A scalable bitwise multicast technology in named data networking. *IEICE Trans Inform Syst*, E105-D(12):2104-2111. https://doi.org/10.1587/transinf.2022EDP7057

Zhang JH, Shen D, Dong F, et al., 2023. Micro-burst aware ECN in multi-queue data centers: algorithm and implementation. *IEEE Trans Netw Sci Eng*, early access. https://doi.org/10.1109/TNSE.2023.3271869

Zhou XK, Zheng XZ, Cui XS, et al., 2023. Digital twin enhanced federated reinforcement learning with lightweight knowledge distillation in mobile networks. *IEEE J Sel Areas Commun*, 41(10):3191-3211. https://doi.org/10.1109/JSAC.2023.3310046

List of supplementary materials