IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 1 August 2022, accepted 5 September 2022, date of publication 16 September 2022, date of current version 28 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3207293

RESEARCH ARTICLE

Load-Balancing Method for LEO Satellite Edge-Computing Networks Based on the Maximum Flow of Virtual Links

YUQI WANG[1,2], JIBIN CHE[3], NINGYUAN WANG[1,2], LIANG LIU[1,2], NAN WU[4], XIAOQING ZHONG[1,2], AND XIAODONG HAN[1]

[1]Institute of Telecommunication and Navigation Satellites, China Academy of Space Technology, Beijing 100094, China
[2]Peng Cheng Laboratory, Shenzhen, Guangdong 518052, China
[3]Key Laboratory of Electronic Information Countermeasure and Simulation Technology, Ministry of Education, Xidian University, Xi'an, Shanxi 710071, China
[4]Beijing Institute of Control Engineering, Beijing 100190, China

Corresponding author: Xiaodong Han (willingdong@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61972398, and in part by the 173 Plan Key Projects under Grant 2019-JCJQ-ZD-342-00.

ABSTRACT With the increasing number of satellites in orbit, traditional scheduling methods can no longer satisfy the increasing data demands of users. The timeliness of remote sensing images with large data volumes is poor in the backhaul process through low-earth-orbit (LEO) satellite networks. To address the above problems, we propose an edge-computing load-balancing method for LEO satellite networks based on the maximum flow of virtual links. First, the minimum rectangle composed of computing nodes is determined by the source and destination nodes of the transmission task under the configuration of the 2D-Torus topology of LEO satellite networks. Second, edge computing virtual links are established between computing nodes and users. Third, the Ford-Fulkerson algorithm is used to obtain the maximum flow of the topology with virtual links. Finally, a strategy is generated for computing and transmission resource allocation. The simulation results show that the proposed method can optimize the total capacity of the multi-node information backhaul in the remote sensing scenario of LEO satellite networks. The effectiveness of the proposed algorithm is verified in several special scenarios.

INDEX TERMS 2D-torus topology, information backhaul, LEO satellite networks, load balancing, virtual links.

I. INTRODUCTION

Satellite communication networks can provide seamless wireless coverage and global access as supplements to existing terrestrial communication networks. Low-earth-orbit (LEO) satellite networks are considered a promising solution for future wireless network architecture [1], [2], [3], [4], [5], [6]. In the past two decades, the development of satellite Internet has entered an unprecedented boom. Large-scale LEO satellite Internet constellations such as Starlink and Lightspeed have developed rapidly. They have received considerable attention from industry capitals, operators, and users [7], [8]. In the latest 6G non-terrestrial network (NTN) proposal, aviation and maritime cases in unserved and underserved areas were expanded to collect large amounts of remote sensing data, including large amounts of backhaul earth observation data [9]. It is important for latency-sensitive earth observation applications, including emergency communications and real-time surveillance.

LEO satellite networks are usually deployed in space at an orbital altitude of 500-2000 km. Compared to other communication systems, LEO satellites are inexpensive to manufacture and launch. Their constellation orbit design was streamlined and modularized. The satellite node deployment is flexible. Because the network is closer to the ground, the link resistance function is better when it is not constrained

The associate editor coordinating the review of this manuscript and approving it for publication was Vittorio Camarchia.

by ground terrain. It also has the advantages of a relatively small round-trip time of approximately 10∼25ms and small channel fading [10], [11]. At present, research on LEO satellite networks mainly focuses on the optimization of spectrum resource sensing strategies, enhancement of beam deployment coverage, and improvement of backhaul links [12], [13], [14]. However, in next-generation communication networks, target recognition, efficient video transcoding and distribution, situational awareness, and other tasks suitable for onboard tasks have high requirements for onboard processing capabilities and resource allocation. With the continuous enhancement of onboard processing capability, edge-computing technology that settles computing and storage resources onboard can enable fast task response for requests [15]. At the same time, LEO satellite communication has flexible coverage, which can collect available data for the aforementioned tasks. It can provide a training model data set support for intelligent network computing. In addition, modern satellites can process data onboard, which can improve remote sensing tasks. It can offload several types of processing data onboard, such as Earth and weather observations. This can reduce the pressure of the downlink associated with the data backhaul. In this regard, studies on edge computing have developed the processing capabilities of satellite segments. This renders the satellite more than a simple relay system.

Currently, the existing satellite edge computing research has mainly focused on three aspects. The first is architecture-related research. Space-air-ground networks for edge computing applications have recently been explored to alleviate the heavy computational tasks of resource-limited, densely distributed terrestrial terminal devices [16], [17], [18]. Although satellite-assisted edge computing may have a higher latency than ground-to-air edge computing, it can still provide significant latency performance improvements compared to long-range cloud computing. Various aspects of space-air-ground edge computing have been studied in the literature. The second is the research on computational offloading and resource allocation. Wang *et al.* [19] introduced computational offloading with bilateral computations for space-air-ground networks. In particular, according to a certain threshold mechanism, computing tasks are offloaded to LEO satellites or terrestrial edge computing where edge computing servers are deployed. Wang *et al.* [20] proposed a game-theory-based approach to optimize computational offloading in satellite edge computing networks. Wang *et al.* [21] proposed a joint offloading and resource allocation method for LEO satellite edge-computing networks. Cui *et al.* [22] studied latency and energy cost optimization for edge-computing satellite Internet-of-Things (IoT) networks. Abderrahim [23] considered an integrated terrestrial space network, in which a traffic offloading scheme was proposed to offload ultra-reliable low-latency communication (uRLLC) traffic to the ground segment and enhanced mobile broadband (eMBB) traffic to the satellite segment. The third is research on performance evaluation. Kim and Choi [24] studied the propagation and queuing delay performance of satellite edge-computing networks under the uplink/downlink packet error rate. Existing methods are mainly based on mixed-integer programming, which has high time-complexity. Satellite networks with high mobility are different from terrestrial networks. Satellites are in the process of periodic high-speed motion and must be solved quickly.

In this paper, we study an edge computing load-balancing method for LEO-satellite-network backhaul tasks, which has low time complexity and engineering achievability. The main contributions of this study are summarized as follows:

- We designed an LEO satellite networks edge-computing architecture that combines the optimization of transmission and computing. The architecture models the relationship between transmission and computing resources.
- We proposed a 2D-Torus network minimum rectangle computing node selection method. The method selects the calculation offload of sensing information back to the ground station.
- We proposed a computational load-balancing algorithm based on the maximum flow of virtual links. The algorithm determines the size of data processed by each routing node.

The reminder of this paper is organized as follows. In Section II, the application scenario, network model, transmission model, and calculation model are presented. In Section III, a problem model that needed to be optimized was formulated. An edge-computing load-balancing method based on the maximum flow of virtual links is proposed. Simulation results and discussions are provided in Section IV. Finally, Section V concludes the paper.

II. SYSTEM MODEL

For the aforementioned scenario description, a real-time information acquisition and transmission LEO constellation with Earth observation, onboard processing and routing is modeled as follows.

A. CONSTELLATION SCENARIO

We consider an application scenario in which the Earth observation satellite obtains image information and transmits the data back to the ground station through LEO satellite networks. This scenario is illustrated in FIGURE 1.

The space segment consists of a single-layer Walker constellation. The constellation configuration is Walker-Delta, where the number of orbital planes is M_p, and the number of satellites per orbit is M_s. It has a relatively stable topology. The main function of the system is to monitor global disaster. After the detection information is generated by the Earth observation satellite, transmission and computing resources are called within a predetermined time window so that the detection information is processed and transmitted to a limited area in real time.

The Walker-Delta constellation configuration is represented by adjacency matrices $\mathbf{A}_{Sat}$, where the element

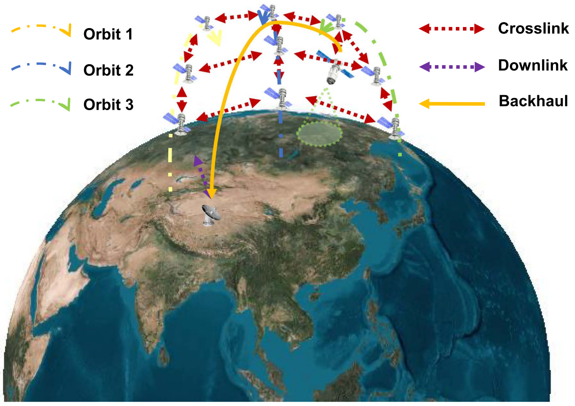

FIGURE 1. The scenario of data backhaul in LEO satellite networks.

$a_{i,j}^{Sat} \in \mathbf{A}_{Sat}$ is the capacity of the crosslink from node i to node j at the instant t, which can be expressed as

$$a_{i,j}^{Sat} = \begin{cases} C_l, & \exists \left(v_i, v_j\right), \ \forall v_i, \ v_j \in V_{Sat}, \ i \neq j \\ 0, & Otherwise \end{cases} \tag{1}$$

where $\left(v_i, v_j\right)$ represents the crosslink where the first item points to the second item. The capacity of the ISL is given by the Gaussian channel capacity formula.

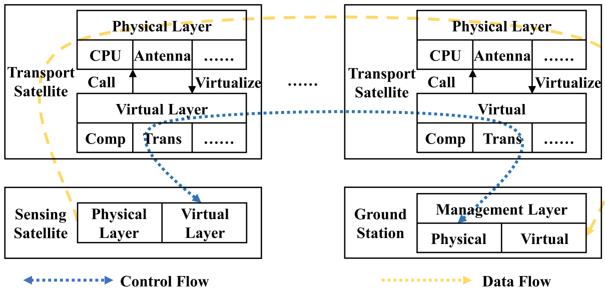

FIGURE 2. Schematic diagram of information flow in LEO satellite networks.

The transmission of the information flow is shown in FIGURE 2. The ground segment is the resource management center responsible for managing the resource configuration of all satellite nodes in constellation networks and receiving observation data. The centralized controller of the resource management center obtains the node resource status through double-layer SDN flooding signaling [25]. It configures node transmission and computing resources through control signaling.

B. OBSERVATION TASK

We consider the distribution of various geological disasters to be universal and random. Relevant information may be collected from all regions of Earth. Therefore, the task distribution weight model is based on the randomness of the observation task. Any node has a certain probability of being connected to a backhaul. To facilitate the description of the traffic, the data traffic generated in the current snapshot is called a task. The ratio between different tasks is called the weight. It is assumed that all tasks originate from the set of sending nodes V_T, and the task eventually flows to the set of receiving nodes in a limited area. This forms the task weight matrix $\mathbf{B} \in N_T \times N_R$

$$\mathbf{B} = \begin{bmatrix} \beta_{1,1} & \cdots & \beta_{1,N_R} \\ \vdots & \ddots & \vdots \\ \beta_{N_T,1} & \cdots & \beta_{N_T,N_R} \end{bmatrix} \tag{2}$$

where $\beta_{i,j} \in \mathbf{B}$ represents the proportion of traffic sent by the observation node $v_i \in V_{US}$ to the ground node $v_j \in V_{UD}$. The i-th row represents all tasks sent by v_i. The j-th column represents all tasks received by v_j, satisfying:

$$\beta_{i,j} = \begin{cases} \dfrac{1}{N_{US}}, & k = 1, \ \forall i, \ j = j_{recv}, \ i \neq j \\ 0, & otherwise \end{cases} \tag{3}$$

This indicates the N_{US} number of the transmission satellites that simultaneously access the observation task at the same time. Binary $k = 1$ indicates that the node has been accessed by the observation task.

C. ONBOARD PROCESSING

We consider that the processing of observation information mainly involves preprocessing a large number of Earth observation images. Data processing can reduce the size of backhaul data by extracting feature information from the data. For the information received by a single satellite node S_i, D_i represents the size of the original data and F_i represents the size of the processed data. If the satellite S_i performs edge computing, we define the calculation transfer ratio $\rho_i = \left(D_i - F_i\right) / F_i$. At the same time, the decision variable is defined as the selected calculation mode. $l_i = 1$ indicates that edge computing processing is performed on the data, whereas $l_i = 0$ represents no calculation processing. The data size of the information generated after the original information passes through the satellite S_i is

$$L_i = l_i F_i + (1 - l_i) D_i = D_i \left(l_i \frac{1}{\rho_i} + (1 - l_i) \right) \tag{4}$$

The processing time at the satellite S_i is

$$T_i^{proc} = \frac{D_i z}{f_{\mathrm{CPU}}} \tag{5}$$

D. INFORMATION BACKHAUL

In this study, it is assumed that the set of all low-orbit satellites is S and the set of ground stations is G. The set of all nodes in the network is called $\mathbf{A} \in N \times N$.

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_S & \mathbf{A}_R \\ \mathbf{A}_T & \mathbf{A}_G \end{bmatrix} \tag{6}$$

where $N = N_S + N_G$. The matrix $\mathbf{A}_S \in N_S \times N_S$ represents the crosslink connectivity matrix of LEO satellite networks. $\mathbf{A}_S(i,j) = 1$ indicates that there is a connected crosslink between satellite i and satellite j, otherwise $\mathbf{A}_S(i,j) = \infty$.

Similarly, $\mathbf{A}_R \in N_S \times N_G$ and $\mathbf{A}_T \in N_G \times N_S$ represent the connected downlink between the satellite and ground station, and $\mathbf{A}_G \in N_G \times N_G$ represents the connection relationship between the ground stations.

The channel capacity is the maximum data rate for reliable transmission. The power and bandwidth-limited Gaussian channel capacity is given by

$$\begin{aligned} C_l &= W \log_2 \left(1 + \frac{P_r}{kTW}\right) \\ &\approx \frac{1}{\ln(2)} \frac{P_r}{kT} \quad (\mathrm{b/s}) \end{aligned} \tag{7}$$

C_l limits the maximum data rate R_i of information transmitted over the channel. Then, the communication delay can be defined as

$$T_n^{DL} = \sum_{i=1}^{n} \left(T_i^{comm} + T_i^{prop}\right) = \sum_{i=1}^{n} \left(\frac{L_n}{R_i} + T_i^{prop}\right) \tag{8}$$

where T_i^{comm} is the propagation time between the i-th node and the (i + 1)-th node.

III. PROBLEM FORMULATION AND PROPOSED SOLUTION

This section discusses the problem formulation and proposed solution method. For the task of large data volumes in LEO satellite networks, it is necessary to study how to maximize the backhaul throughput by selecting routes and allocating computing resources while ensuring the balanced utilization of network computing resources. We propose a computing load balancing method for low-orbit satellite networks based on the maximum flow of virtual links.

A. PROBLEM FORMULATION

For the feasibility of the numerical calculation, a discrete state-space model is adopted. By selecting the sampling interval $\Delta\tau = T/N$, the information flow can be divided into $N + 1$ time slices. FIGURE 3 shows the spatiotemporal logic diagrams of some nodes in different discrete-time slices. Satellite nodes perform different behaviors of information transmission or information processing in different time slices. The following constraints must be considered in the data transmission process:

1) The size of data transmitted or received in the event is less than or equal to the capacity of links.

2) The size of the data transmitted during the task is less than or equal to the size of the data available on the satellite at the instant of the task.

3) All observation data sent through the backhaul are transmitted through the crosslinks and arrive at the downlink according to the planned timing.

According to the above objectives and constraints, the optimization problem of transmission and computing resources in the backhaul can be described by the following optimization problems:

$$\max_{t \in T} \sum_{d} C_d \tag{9}$$

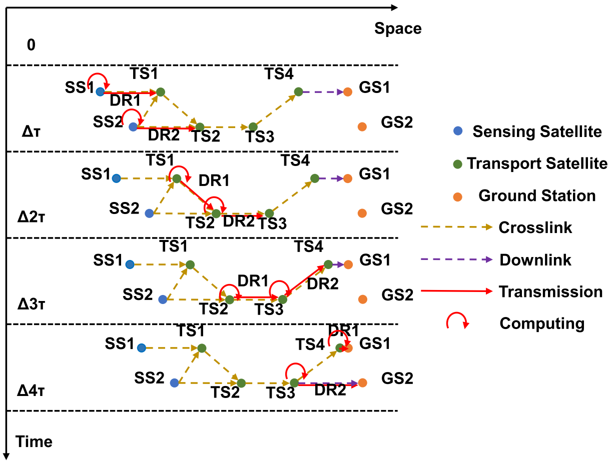

FIGURE 3. Spatiotemporal logic diagrams of transmission and computing resources allocation.

$$\text{s.t.} \quad C_{s,o,t} = \sum_{\tau \in T | \tau < t} \left(\sum_{l \in L_{o,\tau}} \left(\mu_l^s \cdot C_{l,o}\right)\right) + C_p^s,$$

$$\forall p \in P, \quad \forall s \in S \tag{10}$$

$$C_a \leq D_a, \quad \forall a \in A \tag{11}$$

$$I_a \geq \frac{C_a}{D_a}, \quad \forall a \in A, \quad I_a \in \{0, 1\} \tag{12}$$

$$C_l \geq \sum_{o \in O_l} \left(C_{l,o}\right), \quad \forall l \in L \tag{13}$$

$$C_{l,o} \leq C_o, \quad \forall l \in L, \quad \forall o \in O \tag{14}$$

$$\left|C_p^i - C_p^j\right| \leq C_0, \quad \forall p \in P, \quad \forall i,j \in S \tag{15}$$

The problem is NP-hard. Optimization 9 indicates that the optimization goal is to maximize the capacity of the information backhaul per unit time. Constraint 10 indicates that the data entering the node are conserved with the data processed by the node and the data flowing out of it. Constraint 11 indicates that the data transmitted or received in a single task are less than or equal to the data generated by the task. Constraint 12 indicates that when the task data are backhauled, the transmission decision variable I_a is set to one. Constraint 13 limits the maximum data capacity that can be transmitted per unit time in a single link. Constraint 14 limits the data capacity transmitted on a single link for a single observation task. Constraint 15 indicates that the difference between the computing resource occupancy of any two nodes participating in the calculation cannot exceed the constraint C_0.

B. NODE SELECTION

For a single satellite in Walker-Delta LEO satellite networks, there are four crosslinks with two adjacent satellites in the same orbital plane and two satellites in adjacent orbits. This topological connection can be regarded as a 2D-Torus network topology.

Definition 1: The rectangle formed by the source node and destination node as the diagonal in the 2D-Torus network is called the minimum rectangle of network routing. There are multiple routes with a minimum number of hops in the minimum rectangle, as shown in FIGURE 4.

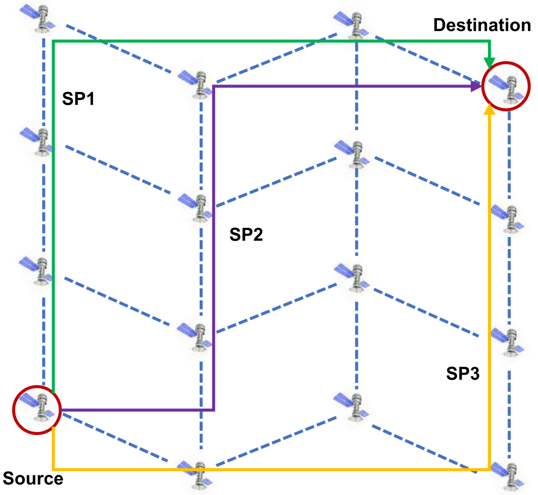

FIGURE 4. **The minimal rectangle schematic of network routing.**

Definition 2: Consider a special case in which the source and destination nodes are on a straight line. The minimum rectangle of the route is the line segment. The adjacent nodes on both sides and the original route node form an expanded minimum rectangle, as shown in FIGURE 5.

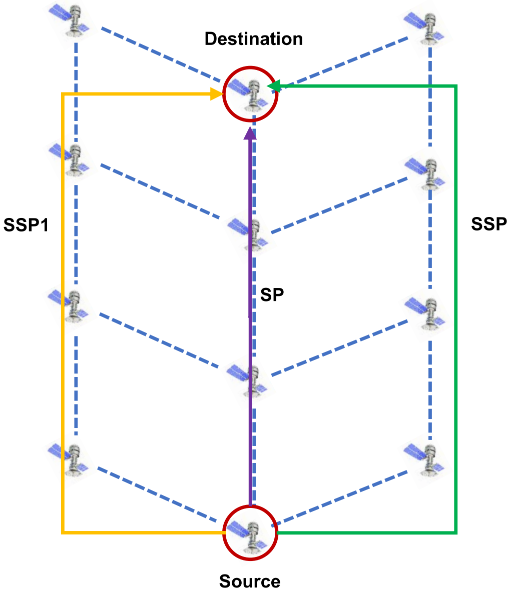

FIGURE 5. **The extended minimum rectangle schematic of line routing.**

We propose a method for selecting routing nodes. First, LEO satellite networks topology is generated according to the constellation position and adjacency relationship per unit time. Second, the source and destination nodes of the task are determined. A minimum routing rectangle is generated. If the minimum rectangle does not exist, the routing neighborhood is adopted to generate the extended minimum rectangle. Finally, all nodes in the minimum rectangle are selected as the path nodes for the information backhaul. The specific algorithm is shown in **Algorithm 1**.

Algorithm 1 Multiple Shortest-Path Nodes Selection Algorithm

```
Input: source node position $\mathbf{P}_{sn}$, destination node $\mathbf{P}_{dn}$,
network topology $\mathbf{T}_L$
Output: set of selected nodes $\mathbf{N}_r$
Begin
Calculate the network topology $\mathbf{T}_L$
Bring in the source node position $\mathbf{P}_{sn}$,
destination node $\mathbf{P}_{dn}$
Find the shortest path $\mathbf{R}_{sp}$ of $\mathbf{P}_{sn}$ and $\mathbf{P}_{dn}$ on $\mathbf{T}_L$
if $\mathbf{R}_{sp}$ is a line segment do
    Find the extended minimum rectangle $\mathbf{R}_{sd}$ of $\mathbf{P}_{sn}$ and
    $\mathbf{P}_{dn}$ according to Definition 2
else do
    Find the minimum rectangle $\mathbf{R}_{sd}$ of $\mathbf{P}_{sn}$ and $\mathbf{P}_{dn}$
    according to Definition 1
end if
Output the set of selected nodes $\mathbf{N}_r$ in $\mathbf{R}_{sd}$
End
```

C. RESOURCE ALLOCATION

After selecting the routing nodes for the information backhaul, it is necessary to allocate the computing and transmission resources of each node according to the observation tasks and resource occupancy. We propose a resource allocation method based on the maximum flow of virtual links. First, according to the computing nodes and node adjacencies selected by **Algorithm 1**, a routing topology of the source and destination node is generated. Second, according to the computing resource occupancy of each node, a virtual link between each node and the user is established. The routing topology is updated. Third, all routing nodes traverse to the full-load state in equal proportions using the available computing resources of each node as the independent variable. A maximum flow search is performed to obtain the maximum capacity of the network topology that satisfies the constraints. Finally, the flow result is output as the allocation strategy for transmission and computing resources. The specific algorithm is shown in **Algorithm 2**.

The solution of the maximum flow from the source node to the destination node is based on the Ford-Fulkerson algorithm. The Ford-Fulkerson algorithm aims to find an augmented path to increase the flow. It determines the path with positive tolerance that can reach the source node. In this

Algorithm 2 Resources Allocation Algorithm Based on the Maximum Flow of Virtual Links

```
Input: set of selected nodes $\mathbf{N}_r$, task weight matrix $\mathbf{B}$,
minimum rectangle $\mathbf{R}_{sd}$, the computing difference
constraint $C_0$
Output: information backhaul throughput $C_d$,
resources allocation strategy $\Re$
Begin
 for each node $C_p = 1$: floor $(f_{\mathrm{CPU}}/z)$ do
     Find occupied computing resource $R_u$ in $\mathbf{N}_r$,
     calculate processing rate $R_p = C_p - R_u$
     if $R_p < 0$ do
        $R_p = 0$
     else do
        $R_p = C_p - R_u$
     end if
     Establish the virtual link between the node and the user,
     the link capacity is $R_p$
     Add the virtual link to the minimum rectangle $\mathbf{R}_{sd}$
     Update the topology $\mathbf{R}_{sd}$
     Calculate the maximum flow of the topology $\mathbf{R}_{sd}$ under
     task weight matrix $\mathbf{B}$
     [flowval, cut, R, F] = Ford-Fulkerson ($\mathbf{B}$, $\mathbf{R}_{sd}$)
     for $i = 1 : n_c$
       for $j = 1 : n_v$
           Calculate CPU occupancy difference $\Delta C$
           between node and node
       end for
     end for
     if $\forall \Delta C < C_0$ do
         break
     else do
         $C_p = C_p + 1$
     end if
 end for
 $C_d \Leftarrow$ flowval
 $\Re \Leftarrow$ R
End
```

process, R is a set of nodes marked as visited, and S is a subset of R, consisting of nodes marked as searched in R.

It should be emphasized that the above process of selecting and allocating computing resources is only for a snapshot. Therefore, a load-balancing algorithm should be applied to all snapshots generated by all time slices to obtain the capacity changes over time. **Algorithm 3** summarizes all steps of the LEO satellite networks load balancing method based on the maximum flow of virtual links.

IV. SIMULATION RESULTS AND DISSCUSION

In this section, we introduce the parameter settings for the simulation scenario. Second, we adopted the number of routing nodes contained in different minimum rectangles as independent variables to compare the performance of always-transmission, always-computing, and the strategy of this study. The performance assessment metrics are backhauled throughput, delay of information backhaul, and average CPU occupancy rate. Finally, we verified the performance of the algorithm under different task access probabilities and resource occupancy.

Algorithm 3 LEO Satellite Networks Load Balancing Algorithm

```
Input: constellation ephemeris for $t \in [0, T]$, snapshot
interval $\Delta t$,
Output: resources allocation strategy $\Re$
Begin
Sample the constellation ephemeris with $\Delta t$, generate
$\mathbf{T}_L$ snapshots.
Determine the type of task weight distribution,
forming the task weight matrix $\mathbf{B}$.
Calculate minimum rectangle $\mathbf{R}_{sd}$ using Algorithm 1
for each snapshot do
    Determine the computing difference constraint $C_0$
    Find occupied computing resource $R_u$
    Calculate processing rate $R_p$
    Calculate $C_d$ and $\Re$ using Algorithm 2
    $C_d \Leftarrow$ flowval
    $\Re \Leftarrow$ R
end for
End
```

The parameters of the simulation are set as follows. Walker-Delta LEO satellite networks composed of 220 satellites are used in the simulation, with a total of 20 orbital planes. Each orbital plane has 11 satellites. The orbital height is $H = 1000km$. The orbital inclination angle is 60°. The sampling interval of the simulation snapshot is 5s. The simulation time is an orbital period of 105 min. The communication frequency of the crosslinks is 26 GHz. The communication frequency of the downlink is 20 GHz. The link bandwidth is 500 MHz. The minimum communication angle of the ground user is 25°.The field of view of the satellite beam is 120°. The main frequency of the satellite-computing CPU is 1Ghz. We assume that = onboard processing does not cause information carried in the image to be lost. The simulation results under different conditions are the average of 1000 Monte Carlo experiments.

A. PERFORMANCE ANALYSIS

In this section, the performance of the algorithm is characterized using three metrics. They are the backhaul throughput, delay of information backhaul, and average CPU occupancy rate. The strategy given by the algorithm in this paper is compared with the always-transmission strategy and the always-computing strategy. The always-transmission strategy involves transmitting all the data back to the user through LEO satellite networks. The always-computing strategy involves sending the processed feature information of all data to the user. There is no difference in the time complexity of the three methods. In the simulation, it is assumed that all nodes are in an idle state. After selecting a fixed source node, we select different destination nodes to verify the performance of the algorithm under different numbers of computing nodes. We select a 2D-Torus network topology ranging from 2×2 to 5×6. The network is simulated as shown

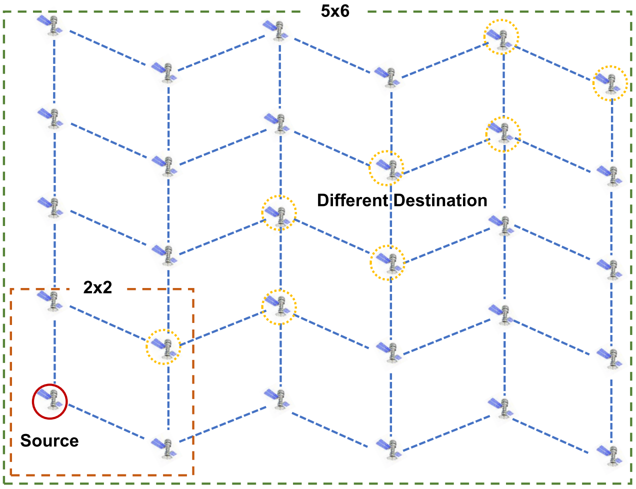

FIGURE 6. Routing nodes in different minimum rectangles.

in FIGURE 6. The total size of the data to be backhauled is 100 GB.

We use the information backhaul throughput to represent the backhaul of the image data per unit time. FIGURE 7 shows the results of the information backhaul throughput with different numbers of routing nodes. The x-axis represents the number of routing nodes occupied by the backhaul information. The y-axis represents the in backhaul throughput per unit time. It can be seen that the information backhaul throughput obtained by our strategy in this study is better than that of the other two strategies in the same scenario. The backhaul throughput of our strategy increases with an increase in the number of routing nodes. When the routing nodes reach 16 satellites, there is an inflection point in the throughput curve. The backhaul throughput no longer increases. This is owing to the limited bandwidth of the crosslinks from the source node. The maximum bandwidth of each crosslink limits information backhaul throughput. When the source node is far from the destination node in the topology, a large number of optional routing nodes can meet the computational requirements of the task. There is an intersection between the always-transmission curve and always-computing curve. In this case, the processing capability of the multi-node computing network and the downlink of the last hop for information backhaul have reached a dynamic balance.

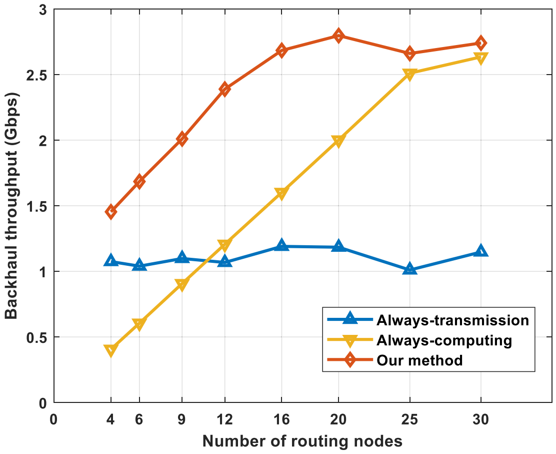

FIGURE 7. Information backhaul throughput with different number of routing nodes.

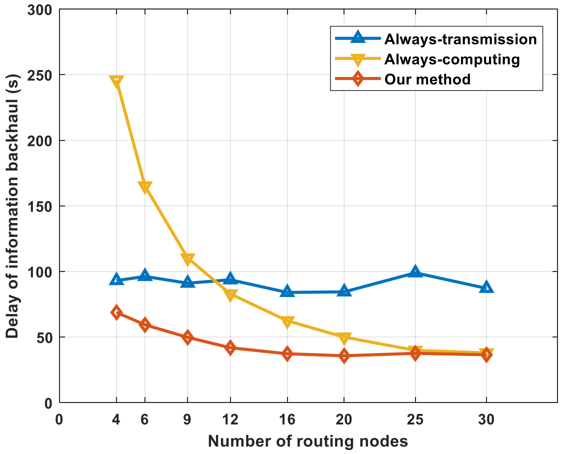

FIGURE 8. Delay of information backhaul with different numbers of routing nodes.

We use the delay of information backhaul to characterize the time consumption from information generation to the user acquiring the information. FIGURE 8 shows results of information backhaul delay for different numbers of routing nodes. The x-axis represents the number of routing nodes occupied by information backhaul. The x-axis represents the delay of information backhaul. It can be seen that the delay of information backhaul obtained by our strategy in this study is better than the other two strategies in the same scenario. Under the condition of a certain amount of remote sensing image data, the delay of information backhaul is inversely proportional to the information backhaul throughput.

We use the CPU average occupancy rate to represent the computing resource occupancy of routing nodes in a single task. FIGURE 9 shows the average CPU occupancy rate of the routing nodes with different numbers of routing nodes. The x-axis represents the number of routing nodes occupied by the backhaul information. The y-axis represents the average CPU occupancy rate. It can be seen that the always-transmission strategy only needs to perform packet routing table lookup and forwarding. It requires almost no computing resources. With an increase in the number of computing nodes, the curve of our strategy in this study and the curve of the always-computing strategy both have an inflection point that decreases from the full load state. Because our strategy balances the occupancy of the computing resources well, the drop point appears earlier.

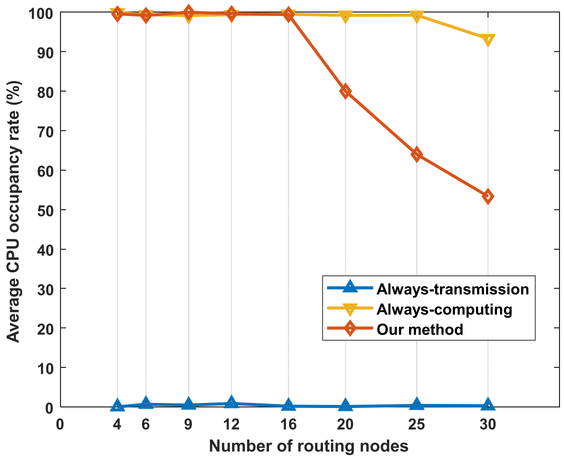

FIGURE 9. Average CPU occupancy rate with different numbers of routing nodes.

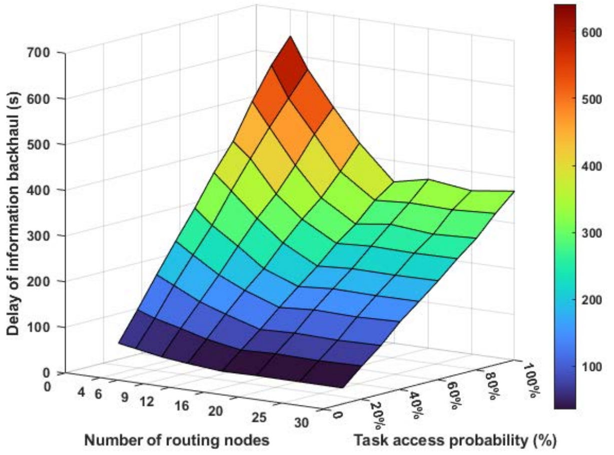

FIGURE 10. Delay of information backhaul under different task access probabilities with different numbers of routing nodes.

After comparing the results of the above three strategies, it can be seen that the strategy proposed in this study is superior to the other two strategies in terms of throughput and delay. In the always-transmission strategy, the throughput bottleneck of the information backhaul is limited by the downlink bandwidth. The backhaul throughput can only be improved by increasing the transmission capacity of the downlink. In the always-computing strategy, computing ability is limited onboard. The computing ability can satisfy the requirements of crosslinks of the source node, only when the computing network composed of routing nodes expands to a certain extent. This affects the utilization efficiency of computing resources in LEO satellite networks. Our strategy in this study balances the occupancy of the node-computing ability and the downlink bandwidth. This makes the use of the system more efficient.

B. VERIFICATION

In this section, we verify the performance of the proposed algorithm in special scenarios. Two special scenarios are established. The first scenario is the information backhaul for different task access probabilities. The second scenario is information backhaul, where the computing resources of some nodes are occupied.

FIGURE 10 shows the delay of information backhaul under different task access probabilities with different numbers of routing nodes. It can be seen that when the task access probability is fixed, the conclusion is the same as that in FIGURE 8. With an increase in task access probability, the task distribution model presents a mode of aggregated distribution. The data of all task access nodes are aggregated to a small number of user nodes. The delay of information backhaul increases approximately linearly with an increase in of task access probability.

Considering that the computing resources of all nodes cannot be fully available at a certain moment in practical application scenarios, we verify the performance of our algorithm when the computing resources of some nodes are occupied. FIGURE 11 shows the resource allocation strategy for a multi-node information backhaul when the computing resources of some nodes are occupied. The green nodes are the source nodes where the tasks are initiated. The orange node is the destination node for the information backhaul. The red nodes represent nodes occupied by 30% of the computing resources. The purple nodes are those occupied by 50% of the computing resources. The blue links represent crosslinks. The yellow link represents the downlink. The arrow represents the transmission direction of information flow. The value of R_p on the node represents the data processing rate of the node per unit time. The value of R_t/C_l on the link represents the current transmission rate R_t of the link and the maximum available transmission capacity C_l of the link. It can be observed that the algorithm in this

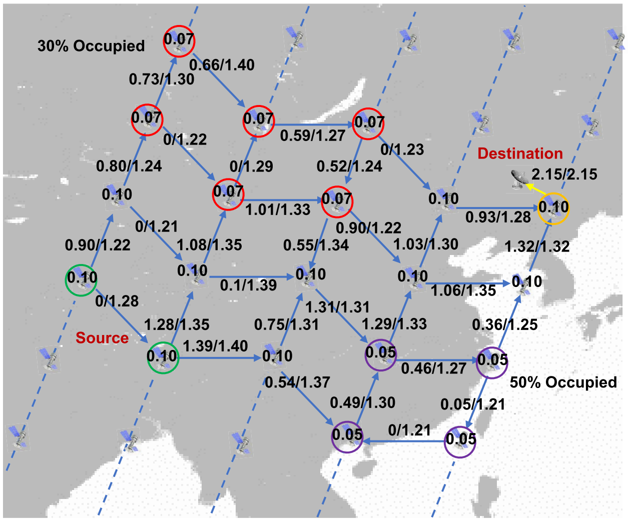

FIGURE 11. Allocation result of our strategy when computing resources of some nodes are occupied.

study can quickly provide an optimal strategy under complex constraints.

V. CONCLUSION

We study the load-balancing problem of transmission and computing resources in large-scale remote sensing data backhaul through LEO satellite networks. Aiming at the problem of low efficiency of information backhaul in existing methods, we propose a computing load balancing method for LEO satellite networks based on the maximum flow of virtual links. First, based on the particularity of the 2D-Torus topology of the low-orbit satellite network, a minimum rectangular computing node selection method is designed. Second, according to the onboard edge computing model, the virtual links of computing nodes in the network are established. We use the Ford-Fulkerson algorithm to obtain the strategy for transmission and computing resource allocation. Finally, the simulation results show that the proposed algorithm effectively balances the transmission bottleneck of the downlink and limited computing ability onboard. This is a new concept for improving the application efficiency of LEO satellite networks.

REFERENCES

[1] R. Gopal and N. BenAmmar, "Framework for unifying 5G and next generation satellite communications," *IEEE Netw.*, vol. 32, no. 5, pp. 16–24, Sep. 2018.

[2] L. Boero, R. Bruschi, F. Davoli, M. Marchese, and F. Patrone, "Satellite networking integration in the 5G ecosystem: Research trends and open challenges," *IEEE Netw.*, vol. 32, no. 5, pp. 9–15, Sep. 2018.

[3] G. Giambene, S. Kota, and P. Pillai, "Satellite-5G integration: A network perspective," *IEEE Netw.*, vol. 32, no. 5, pp. 25–31, Sep. 2018.

[4] H. Yao, L. Wang, X. Wang, Z. Lu, and Y. Liu, "The space-terrestrial integrated network: An overview," *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 178–185, Apr. 2018.

[5] M. Vondra, M. Ozger, D. Schupke, and C. Cavdar, "Integration of satellite and aerial communications for heterogeneous flying vehicles," *IEEE Netw.*, vol. 32, no. 5, pp. 62–69, Sep. 2018.

[6] Y. Ruan, Y. Li, C. Wang, R. Zhang, and H. Zhang, "Performance evaluation for underlay cognitive satellite-terrestrial cooperative networks," *Sci. China Inf. Sci.*, vol. 61, no. 10, pp. 1–11, Oct. 2018.

[7] I. Del Portillo, B. G. Cameron, and E. F. Crawley, "A technical comparison of three low Earth orbit satellite constellation systems to provide global broadband," *Acta Astronautica*, vol. 159, pp. 123–135, Jun. 2019.

[8] J. C. McDowell, "The low earth orbit satellite population and impacts of the SpaceX Starlink constellation," *Astrophys. J. Lett.*, vol. 892, no. 2, pp. 36–45, Apr. 2020.

[9] X.-H. You, C. X. Wang, J. Huang, X. Gao, Z. Zhang, M. Wang, Y. Huang, C. Zhang, Y. Jiang, J. Wang, and M. Zhu, "Towards 6G wireless communication networks: Vision, enabling technologies, and new paradigm shifts," *Sci. China Inf. Sci.*, vol. 64, no. 1, pp. 1–74, Jan. 2021.

[10] Q. Chen, G. Giambene, L. Yang, C. Fan, and X. Chen, "Analysis of intersatellite link paths for LEO mega-constellation networks," *IEEE Trans. Veh. Technol.*, vol. 70, no. 3, pp. 2743–2755, Feb. 2021.

[11] N. Wang, L. Liu, Z. Qin, B. Liang, and D. Chen, "Capacity analysis of LEO mega-constellation networks," *IEEE Access*, vol. 10, pp. 18420–18433, 2022.

[12] S. Fu, J. Gao, and L. Zhao, "Integrated resource management for terrestrial-satellite systems," *IEEE Trans. Veh. Technol.*, vol. 69, no. 3, pp. 3256–3266, Jan. 2020.

[13] Y. Su, Y. Liu, Y. Zhou, J. Yuan, H. Cao, and J. Shi, "Broadband LEO satellite communications: Architectures and key technologies," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 55–61, Apr. 2019.

[14] W. Liu, Y. Tao, and L. Liu, "Load-balancing routing algorithm based on segment routing for traffic return in LEO satellite networks," *IEEE Access*, vol. 7, pp. 112044–112053, 2019.

[15] L. Yan, S. Cao, Y. Gong, H. Han, J. Wei, Y. Zhao, and S. Yang, "SatEC: A 5G satellite edge computing framework based on microservice architecture," *Sensors*, vol. 19, no. 4, pp. 831–846, Feb. 2019.

[16] Z. Zhang, W. Zhang, and F.-H. Tseng, "Satellite mobile edge computing: Improving QoS of high-speed satellite-terrestrial networks using edge computing techniques," *IEEE Netw.*, vol. 33, no. 1, pp. 70–76, Jan. 2019.

[17] R. Xie, Q. Tang, Q. Wang, X. Liu, F. R. Yu, and T. Huang, "Satellite-terrestrial integrated edge computing networks: Architecture, challenges, and open issues," *IEEE Netw.*, vol. 34, no. 3, pp. 224–231, Mar. 2020.

[18] C. Li, Y. Zhang, R. Xie, X. Hao, and T. Huang, "Integrating edge computing into low Earth orbit satellite networks: Architecture and prototype," *IEEE Access*, vol. 9, pp. 39126–39137, 2021.

[19] Y. Wang, J. Zhang, X. Zhang, P. Wang, and L. Liu, "A computation offloading strategy in satellite terrestrial networks with double edge computing," in *Proc. IEEE Int. Conf. Commun. Syst. (ICCS)*, Dec. 2018, pp. 450–455.

[20] Y. Wang, J. Yang, X. Guo, and Z. Qu, "A game-theoretic approach to computation offloading in satellite edge computing," *IEEE Access*, vol. 8, pp. 12510–12520, 2020.

[21] B. Wang, T. Feng, and D. Huang, "A joint computation offloading and resource allocation strategy for LEO satellite edge computing system," in *Proc. IEEE 20th Int. Conf. Commun. Technol. (ICCT)*, Oct. 2020, pp. 649–655.

[22] G. Cui, X. Li, L. Xu, and W. Wang, "Latency and energy optimization for MEC enhanced SAT-IoT networks," *IEEE Access*, vol. 8, pp. 55915–55926, 2020.

[23] W. Abderrahim, O. Amin, M.-S. Alouini, and B. Shihada, "Latency-aware offloading in integrated satellite terrestrial networks," *IEEE Open J. Commun. Soc.*, vol. 1, pp. 490–500, 2020.

[24] T. Kim and J. P. Choi, "Performance analysis of satellite server mobile edge computing architecture," in *Proc. IEEE 92nd Veh. Technol. Conf. (VTC-Fall)*, Nov. 2020, pp. 1–6.

[25] W. Ningyuan, D. Chen, L. Liang, M. Wang, and L. Bingyuan, "An SDN based highly reliable in-band control framework for LEO mega-constellations," in *Proc. IEEE 6th Int. Conf. Comput. Commun. Syst. (ICCCS)*, Apr. 2021, pp. 970–975.

YUQI WANG received the B.S. degree in information engineering from Tianjin University, Tianjin, China, in 2015, and the M.S. degree in information and communication engineering from the China Academy of Space Technology, Beijing, China, in 2019. He is currently pursuing the Ph.D. degree in aeronautical and astronautical science and technology with the China Academy of Space Technology. His research interests include edge computing, satellite constellation networking, and satellite mission planning.

JIBIN CHE received the B.S. and M.S. degrees in information engineering from the PLA Communication Command Institute, Wuhan, China, in 2007 and 2009, respectively. He is currently a Research Fellow of the Key Laboratory of Electronic Information Countermeasure and Simulation Technology, Ministry of Education, Xidian University, Xi'an, China. His research interests include edge computing and satellite communication.

NINGYUAN WANG received the B.S. degree in information and computation technology and the M.S. degree in aeronautical and astronautical science and technology from Beihang University, Beijing, China, in 2015 and 2018, respectively, and the Ph.D. degree in aeronautical and astronautical science and technology from the China Academy of Space Technology, Beijing, in 2022. He is currently an Engineer with the China Academy of Space Technology. His research interests include satellite constellation networking, software define networks, and routing algorithm for constellation networks.

LIANG LIU received the B.S. and Ph.D. degrees from the Beihang University of Aeronautics and Astronautics, Beijing, China, in 2009 and 2017, respectively. He is currently a Senior Engineer with the China Academy of Space Technology, Beijing. His research interests include network coding, mobile communication, and satellite communication.

NAN WU received the B.S. degree in automation from Northwest Polytechnical University, Xi'an, China, in 2016, and the M.S. degree in control science and technology from the China Academy of Space Technology, Beijing, China, in 2019. She is currently an Engineer with the Beijing Institute of Control Engineering, Beijing. Her research interests include satellite control and mission planning.

XIAOQING ZHONG received the B.S. and Ph.D. degrees in control science and technology from the Harbin Institute of Technology, Harbin, China, in 2005 and 2010, respectively. He is currently a Research Fellow of the China Academy of Space Technology, Beijing, China. His research interests include spacecraft system engineering and satellite communication.

XIAODONG HAN received the B.S. and Ph.D. degrees in control science and technology from the Nanjing University of Science and Technology, Nanjing, China, in 2005 and 2009, respectively. He is currently a Research Fellow of the China Academy of Space Technology, Beijing, China. His research interests include spacecraft integrate electronics and satellite communication.