IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 25, 2019, accepted August 7, 2019, date of publication August 12, 2019, date of current version August 26, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2934932

Load-Balancing Routing Algorithm Based on Segment Routing for Traffic Return in LEO Satellite Networks

WEI LIU, YING TAO, AND LIANG LIU
Institute of Telecommunication Satellite, China Academy of Space Technology, Beijing 100094, China
Corresponding author: Ying Tao (tao.ying@126.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61675232 and Grant 61775237.

ABSTRACT To tackle the network congestion problem caused by ground gateway stations arranged within a limited area in low earth orbit (LEO) satellite networks, a routing algorithm based on segment routing for traffic return is proposed. Light and heavy load zones are dynamically divided according to the relative position relationship between gateways and the reverse slot. The pre-balancing shortest path algorithm is used in the light load zone, and the minimum weight path defined by congestion index is the routing rule in the heavy load zone. Then, the consistent forwarding is performed referring to segment routing in all zones. Simulation conditions are different sizes of heavy load zone, different traffic density distributions, and different traffic demands. Simulation results confirm that the load-balancing performance is improved significantly with the extension of the heavy load zone size in terms of the average rejection ratio, the average relative throughput, the maximum link utilization, and the average delay. The proposed algorithm is an alternative solution and guidance for routing strategy in LEO satellite networks.

INDEX TERMS Load balancing, LEO satellite networks, routing algorithm, segment routing.

I. INTRODUCTION

Low earth orbit (LEO) satellite networks, represented by Iridium NEXT [1] and Starlink [2], are designed to supply global coverage and real-time services, and contribute to the development of space-ground integrated communication systems [3]. Routing strategy is the core of communication networks. Due to the difference between LEO satellite networks (LSNs) and terrestrial networks, like topology dynamic, LSNs are difficult to adopt mature routing technologies in terrestrial networks. Meanwhile, ground gateway stations, arranged within a limited area, are creating enormous challenges for satellite communications, such as severe link congestion.

The centralized distribution of gateways and larger traffic return of various LSN services, e.g., Internet of Things service and the return of data, can converge traffic to the limited gateways area seriously, while severe network congestion and excessive link load would exist [4], [5]. Effective routing techniques must be considered to overcome network congestion. In order to borrow from terrestrial networks, software-defined network (SDN) and network virtualization (NV) are introduced to facilitate the integration of satellite networks and terrestrial networks [6], and virtual topology (VT) provides a choice for LSN dynamic. The LSN period is divided into several time slices according to periodic changes, and the topology is regarded as static in each time slice [7]. The LSN load-balancing routing algorithm is necessary to study to meet the quality of service (QoS) and improve the network throughput. At the same time, the restriction that gateways are arranged within a limited area must be broken.

Therefore, the load-balancing routing algorithm based on segment routing (SR) is proposed for the LSN traffic return under the centralized distribution of gateways. Constructing LSN system model is the basis of the proposed algorithm. Light and heavy load zones are dynamically divided according to the relative position relationship between gateways and the reverse slot, and different routing rules are adopted in different zones to improve network throughput and avoid congestion. The pre-balancing shortest path algorithm is used in the light load zone, and the minimum weight

The associate editor coordinating the review of this article and approving it for publication was Qiang Yang.

routing is based on congestion index when the traffic is converged to the heavy load zone.

The rest is organized as follows. Related works are summarized in section II, where concepts and applications of segment routing are also introduced. Section III constructs the system model from LSN communication scenario and traffic transmission model, and defines the problem clearly. Section IV proposes the algorithm in detail, including the division of the light and heavy load zone, routing in the light and heavy load zone, and analyzing the time complexity. Simulation results confirm that the load balancing performance is improved in section V, in terms of the average rejection ratio, the average relative throughput, the maximum link utilization, the average delay and the average jitter. Conclusion is shown in section VI.

II. RELATED WORKS

A. LOAD BALANCING ROUTINGS

Load balancing is significant for LSNs to improve communication quality. Minimum interference routing algorithm (MIRA) [8] defines network interference and selects minimum interference paths maximizing the minimum maxflow. Lots of network knowledge and potential demands are used combining with link state and auxiliary capacity information. MIRA is a terrestrial routing algorithm but inspires LSNs traffic engineering researches, such as concepts of critical links and feasible network.

DT-TTAR algorithm [9] uses discrete model for LSN communication, and adopts link cost metric to improve network throughput. Link state and processing delay get attention, and arrival speed, time and locations determine link cost. Distributed traffic balancing routing (DTBR) [10] utilizes VT model for traffic balancing. Traffic prediction is used to update link weight, which is the product of location factor coefficient and delay. Failed satellites are also considered to guarantee communication survivability. Leftover load rate is the core of hybrid routing algorithm (HRA) [11], which identify the node leftover load ability. Link state is determined by node ability at both ends. At the same time, HRA uses the ant colony algorithm to find the best routing, while initial pheromones are from genetic algorithm. Joint Depth-First-Search (DFS) and Dijkstra algorithm (JDDA) [12] combines SDN and VT. DFS is used to find necessary nodes, while high traffic load cases utilize Dijkstra algorithm. In general, the Dijkstra algorithm is used between the source node and the first necessary node, or the last necessary node and the end node, while DFS is used between necessary nodes.

However, the restriction that ground gateway stations are arranged within a limited area does not receive enough attention. The proposed algorithm focuses on load balancing under the centralized distribution of gateways, while segment routing could have advantages in tackling the congestion.

B. SEGMENT ROUTING

Segment routing (SR) is a tunneling technology based on source routing, which allows hosts and edge routers to conduct traffic through network by some segments and intermediate routers do not have to maintain all information of paths [13]. The Internet Engineering Task Force (IETF) is promoting the standardization of segment routing (SR) that is associated with SDN, which can be divided into control plane and data plane. SR supports Multi-Protocol Label Switching (MPLS) and Internet Protocol (IP). The SR control plane is based on the extension of the Interior Gateway Protocol (IGP), while the SR data plane simplifies and reuses MPLS [14]. Thereby, SR is flexible and scalable.

A segment is an identifier for conducting traffic to the corresponding node, link, and service. The active segment is the current executing segment which is the header of segment list (SL). Each segment is identified by ID, i.e., Segment ID (SID). Three types are defined for segments. Node SID is unique in the network and each node has the Node SID, which means forwarding traffic towards the node associated with that ID by IGP. Adjacency SID only has the local meaning in one node and is assigned to adjacency links of the node, which means forwarding traffic over the corresponding adjacency. Service SID also has the local meaning in one node and is assigned to one service provided by the node that would process traffic. Besides, three operations are defined including CONTINUE, PUSH, and NEXT. CONTINUE is the forwarding action based on the active segment. PUSH is adding a segment ahead of the SL header and setting that segment as the active segment. NEXT is marking the next segment as the active segment.

A SR example is shown as Fig. 1. The data of router 101 is sent to router 108. The controller computes the path by routing algorithm that is converted into segment list {104, 1001, 108}. Then, the ingress router 101 is configured with that segment list. Routers forward data to router 104 with IGP when the active segment is 104. Before entering the router 104, the active segment is changed to 1001 with operation NEXT. The corresponding link is selected which sends data to router 105 according to the Adjacency SID 1001. Finally, the data is sent to the egress router 108 with IGP when the active segment is 108.

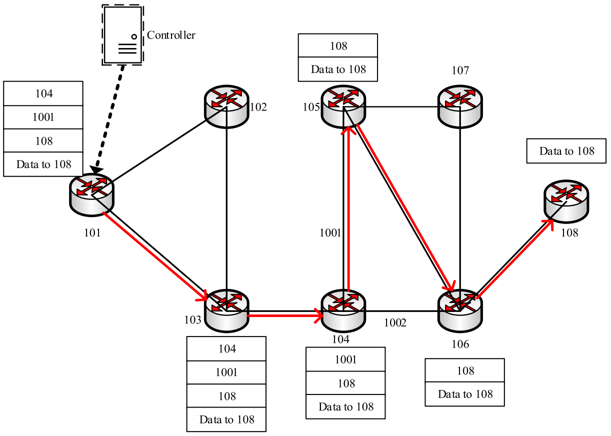


FIGURE 1. Segment routing example.

Researches have proven that SR is suitable for load balancing routing in terrestrial networks. A multicast algorithm, based on segment routing, considers betweenness centrality and congestion index and reduces the number of forked nodes to implement the congestion avoidance in SDN [15]. A SR end-to-end routing algorithm in SDN also uses betweenness centrality and congestion index to define the link weight, and obtains the forwarding path which improve network throughput [16]. Another SR routing algorithm in SDN also combines the multiple objective particle swarm optimization algorithm and builds a three-layer evaluation model of key performance evaluation index, business scheme, and evaluation results [17]. So, SR can achieve excellent traffic engineering performance with small segment list cost and simplify MPLS [18]. However, joint SR and LSN gets little attention. The proposed algorithm in this paper takes advantage of SR to implement LSN load balancing routing for improving network QoS.

III. LEO SATELLITE NETWORK SYSTEM MODEL

A. LEO SATELLITE NETWORK COMMUNICATION SCENARIO

The LEO satellite network considers Walker constellation with satellite links. Each satellite communications with several adjacent satellites in the same orbit or in different orbits through satellite links. Some satellites communicate with ground gateway stations through satellite-ground feedback links to ensure information transmission. Gateways access the ground central station or ground network through ground links. All gateways are arranged within a limited area considering the actual situation, and each gateway communicates with only one satellite anytime. In addition, due to the high speed relative motion between satellites in the first and the last orbit, Walker constellation exists the reverse slot where satellites in different orbits do not set inter-satellite links. The LSN scenario is shown as Fig. 2. LEO satellite constellation has $N = n \times m$ satellites, where n is the number of orbits and m is the number of satellites per orbit. Each satellite establishes intra-satellite links with two adjacent satellites in the same orbit and inter-satellite links with two adjacent satellites in the adjacent orbits, excluding the reverse slot. The polar orbit is adopted and satellites in each orbit distribute evenly. X gateways are arranged within a limited area with one central station.

In the scenario, the topology could be presented by $G = (\boldsymbol{V}, \boldsymbol{E})$. Node set $\boldsymbol{V}$ consists of the satellite node set $\boldsymbol{V}_{\mathrm{S}}$ which includes N satellite nodes, the gateway node set $\boldsymbol{V}_{\mathrm{GW}}$ which includes X gateway nodes, and a central station node V_{C}. Link set $\boldsymbol{E}$ consists of the satellite link set $\boldsymbol{E}_{\mathrm{ISL}}$, the feedback link set $\boldsymbol{E}_{\mathrm{F}}$, and the ground link set $\boldsymbol{E}_{\mathrm{G}}$. Each satellite node, not close to the reverse slot, communications with four adjacent satellites through directed satellite links, as shown in Fig. 3. Besides, some satellite nodes communicate with gateway nodes via directed feedback links, and the gateway nodes also access the central station node. The bandwidth of $\boldsymbol{E}_{\mathrm{ISL}}$ is B_{ISL}, and the bandwidth of $\boldsymbol{E}_{\mathrm{F}}$ is B_{F}. $\boldsymbol{E}_{\mathrm{G}}$ is considered with infinite bandwidth.

In order to solve the topology dynamic of LSN, VT model is adopted by means of the predictability and periodicity of LEO satellite constellation. Geographical cell discretization is used to process ground traffic sources. Furthermore, communication channels are considered ideal channels, regardless of attenuation, multipath, and so on, and mobility management is simplified as an ideal way without abnormal addressing.

B. TRAFFIC TRANSMISSION MODEL

The ground traffic is processed with geographical cell discretization. The ground surface is divided into rectangular traffic cells according to location and constellation. Each traffic cell should be bind to one satellite anytime which is responsible for communication and traffic in that cell. The handover condition is that the satellite communication range covers another cell [19]. At the same time, the traffic density of each traffic cell is predicted and marked referring to the geographical location, population, *et al*. The traffic density is the ratio of the traffic demand of the cell to the maximum flow demand of the system, and the actual traffic demand value is product of traffic density and unit service value. The unit service value is recorded as u. Units of traffic value and u are the same as units of B_{ISL} and B_{F}, so the traffic in this paper can be considered as the relative traffic based on 1 unit bandwidth. The traffic density of cell k is denoted as $f^{(k)}$, and the relative traffic is $u \times f^{(k)}$.

$\boldsymbol{P}^{(k)}$ stands for the path where relative traffic $u \times f^{(k)}$ of cell k returns to the central station node through satellite nodes, and $\boldsymbol{P}^{(k)} = \{\boldsymbol{P}^{(k)}_{\mathrm{ISL}}, \boldsymbol{P}^{(k)}_{\mathrm{F}}\}$ where $\boldsymbol{P}^{(k)}_{\mathrm{ISL}}$ is the path between satellites and $\boldsymbol{P}^{(k)}_{\mathrm{F}}$ is the path consisting of feedback links and ground links. The traffic of any link in $\boldsymbol{P}^{(k)}$ increases $u \times f^{(k)}$ while the residual bandwidth decreases $u \times f^{(k)}$. The coefficient function $\lambda(\bullet)$ is defined as (1) where e stands for links and $\boldsymbol{P}$ stands for paths. The path $\boldsymbol{P}$ passes the link e when $\lambda(e, \boldsymbol{P}) = 1$ while the path $\boldsymbol{P}$ does not pass the link e when $\lambda(e, \boldsymbol{P}) = 0$.

$$\lambda(e, \boldsymbol{P}) = \begin{cases} 1, & e \in \boldsymbol{P} \\ 0, & e \notin \boldsymbol{P} \end{cases} \tag{1}$$

$\boldsymbol{E}_{\mathrm{ISL}} = \{e^{(\mathrm{ISL})}_{ij}\}, i \neq j, i, j \in [0, N-1]$ where $e^{(\mathrm{ISL})}_{ij}$ stands for satellite links actually existing in the network. Total traffic

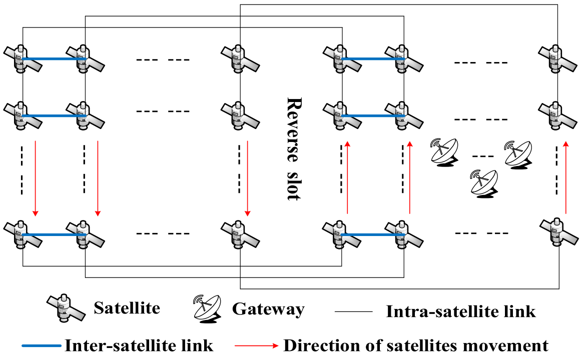


FIGURE 2. LEO satellite network scenario.

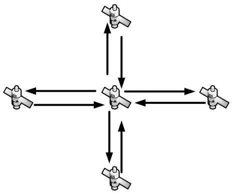


FIGURE 3. Satellite links.

carried by $e_{ij}^{(\mathrm{ISL})}$ could be presented by

$$F[e_{ij}^{(\mathrm{ISL})}] = u \times [\sum_{k=0}^{N-1} f^{(k)} \times \lambda(e_{ij}^{(\mathrm{ISL})}, \boldsymbol{P}_{\mathrm{ISL}}^{(k)})] \tag{2}$$

$\boldsymbol{E}_{\mathrm{F}} = \{e_{ij}^{(F)}\}, i \in [0, N-1], j \in [0, X-1]$ where $e_{ij}^{(F)}$ stands for feedback links existing in the network. Total traffic carried by $e_{ij}^{(F)}$ could be presented by

$$F[e_{ij}^{(\mathrm{F})}] = u \times [\sum_{k=0}^{N-1} f^{(k)} \times \lambda(e_{ij}^{(\mathrm{F})}, \boldsymbol{P}_{\mathrm{F}}^{(k)})] \tag{3}$$

C. PROBLEM DEFINITION

To solve network congestion problem caused by the centralized distribution of gateways and improve the transmission and communication QoS, the problem combined with the above models is defined by

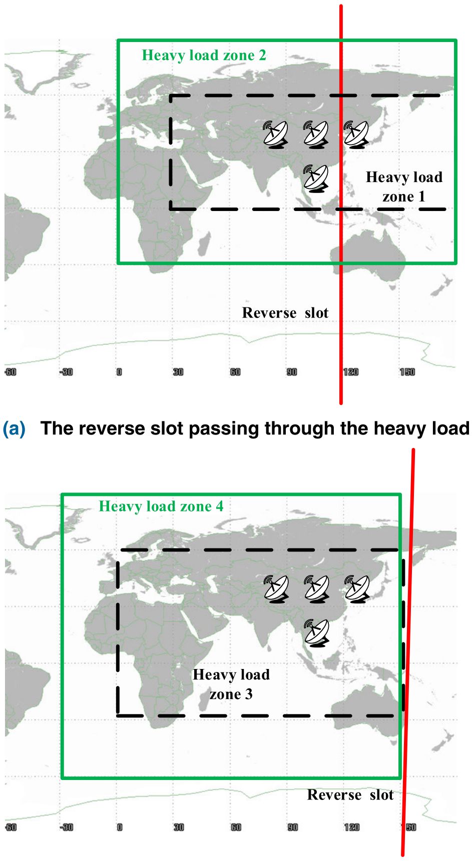
- Objective:
  appr(maximum relative throughput) by $\boldsymbol{P}$
- Conditions:
  $G = (\boldsymbol{V}, \boldsymbol{E})$
  $f^{(k)}, k \in [0, N-1]$
- Subject to:
  $F[e_{ij}^{(\mathrm{ISL})}] \le B_{\mathrm{ISL}}, i \neq j, i, j \in [0, N-1]$
  $F[e_{ij}^{(\mathrm{F})}] \le B_{\mathrm{F}}, , i \in [0, N-1], j \in [0, X-1]$
  Acceptable cost

IV. LOAD-BALANCING ROUTING ALGORITHM BASED ON SEGMENT ROUTING

A. ROUTING ALGORITHM

The traffic will be excessively converged to the limited zone because of gateways arranged within the limited area, causing serious network congestion. The network load exceeds the threshold seriously surrounding the gateways area, while the network load is at a low level in areas far from gateways area, especially the sparsely populated areas.

The light load zone and the heavy load zone are divided dynamically according to the relative position relationship between gateways and the reverse slot. The limited gateways area is extended into a regular rectangular zone through ground traffic cells. The heavy load zone is inside the rectangular zone, while the light load zone is outside the rectangular zone. The dynamic division of load zones is determined by the routing in the heavy load zone because the phenomenon that the reverse slot blocks communication would be avoided. Different routing strategies are adopted in different zones. Therefore, paths in the light or the heavy load zone would generate a series of segments to ensure forwarding coherence based on SR.



(a) The reverse slot passing through the heavy load zone.

(b) The reverse slot outside the heavy load zone.

FIGURE 4. Examples for division of the heavy load zone and the light load zone.

The size (y_{n}, y_{m}) of heavy load zone is defined as y_{n} orbits and y_{m} satellite nodes on each orbit within heavy load zone, while the light load zone has ($N - y_{\mathrm{n}} \times y_{\mathrm{m}}$) satellite nodes. The size of heavy load zone is the key to the proposed algorithm, and two examples, including two types of position relationships between gateways and the reverse slot, are shown in Fig. 4. The zone with gateways as the center is extended into the heavy load zone when the reverse slot traverses the gateways area, as Fig. 4(a) presents, where size of heavy load zone 1 is (5,2) and size of heavy load zone 2 is (6,4). For another, when reverse slot is outside the gateways area, the zone containing gateways but not the reverse slot is extended into the heavy load zone to avoid the reverse slot passing through the heavy load zone, as Fig. 4(b) presents, where size of heavy load zone 3 is (5,3) and size of heavy load zone 4 is (6,5).

In the light load zone, routing paths of satellite nodes will pass through the heavy load zone, and must include one of outermost circle satellite nodes inside the heavy load zone, recorded as outermost nodes. Then, outermost nodes are responsible for subsequent routing. The shortest path algorithm is used to generate minimum spanning tree (MST) because the traffic of light load zone is at a low level, which can improve the delay performance and reduce SR overhead. A pre-equalization is adopted in the light load zone to avoid excessive traffic selecting the same outermost node that causes unexpected congestion. The times of each outermost node could be counted according to original paths from the initial MST. For outermost nodes with high frequency, the weight of their links is increased to reduce frequency selected. The number of outermost node i occupied is assumed to x_i, and the weight of links is adjusted to $(0.5 + 0.1 \times x_i)$ where 0.5 is the initial link weight in the network, and 0.1 is used to reduce the order of magnitude for x_i avoiding over-adjustment. In the pre-balancing network, the MST is calculated again where the source node is the central station, and satellite nodes in the light load zone are destination nodes. The path per satellite node in the light load zone is the retrorse path in the MST, but only portions in the light load zone will be retained. Finally, each path starts from a satellite node in the light load zone and ends at an outermost node.

In the heavy load zone, the routing strategy differs from that in the light load zone. Satellite nodes only route within the heavy load zone and do not enter the light load zone. Traffic carried by outermost nodes consists of two parts. One is the traffic from the ground traffic cell, and the other is the total traffic from other nodes in the light load zone, both of which are routed uniformly. In order to save resources and maximize throughput, the priority of each satellite node in the heavy load is sorted from high to low with traffic from large to small considering the difference in traffic carried by each satellite. Satellite nodes route in order of priority from high to low, so that the larger the traffic, the shorter the path, which can reduce the resource occupation.

The congestion index $c(e)$ of link e in the heavy load area is defined by (4). $b(e)$ stands for the bandwidth of link e. $F(e)$ stands for the total traffic currently carried by link e. $r(e)$ stands for the residual bandwidth of link e. $c(e)$ indicates the congestion of link e. The larger $c(e)$ is, the more congested the link is. $c(e) = \infty$ when $r(e) = 0$ and $b(e) = F(e)$, which presents that link e has no available bandwidth and is open. On the other hand, $c(e) = 0$ when $r(e) = b(e)$ and $F(e) = 0$, which presents that the full bandwidth of link e is available. Compared with link utilization $F(e)/b(e)$, $c(e)$ is more monotonously incremental to $F(e)$ and more sensitive to load change. $c(e)$ would increases sharply if the traffic is too larger which is good to balance the network load.

$$c(e) = {}^{F(e)}\!/_{r(e)},\ r(e) = b(e) - F(e) \tag{4}$$

The weight $w(e)$ of link e in the heavy load zone, including $\boldsymbol{E}_{\mathrm{ISL}}$ and $\boldsymbol{E}_{\mathrm{F}}$, is set as (5). The link weight is smaller when the link carries lighter load. Particularly, the weight is 0.01 while the link is empty. On the contrary, the larger the load on the link, the larger the link weight. $w(e) = \infty$ when the link has no residual bandwidth. The weight of path $\boldsymbol{P}$ is set as (6) and the residual bandwidth of path $\boldsymbol{P}$ is set as (7).

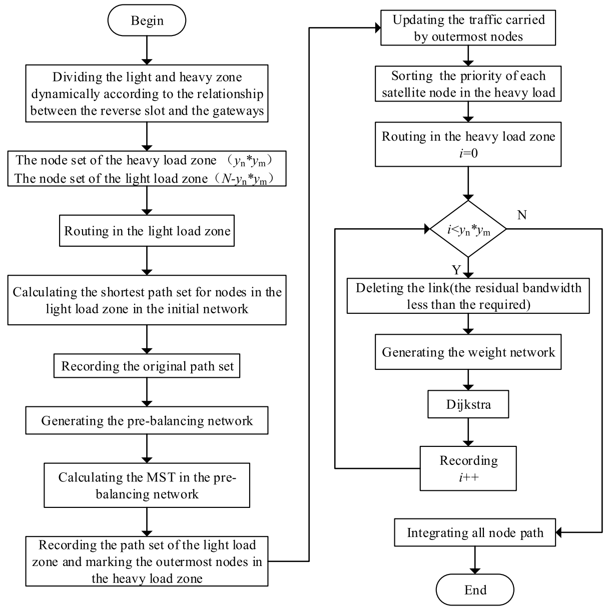

Begin
Dividing the light and heavy zone dynamically according to the relationship between the reverse slot and the gateways
The node set of the heavy load zone（y_n*y_m）
The node set of the light load zone（N-y_n*y_m）
Routing in the light load zone
Calculating the shortest path set for nodes in the light load zone in the initial network
Recording the original path set
Generating the pre-balancing network
Calculating the MST in the pre-balancing network
Recording the path set of the light load zone and marking the outermost nodes in the heavy load zone
Updating the traffic carried by outermost nodes
Sorting the priority of each satellite node in the heavy load
Routing in the heavy load zone i=0
i<y_n*y_m
N
Y
Deleting the link(the residual bandwidth less than the required)
Generating the weight network
Dijkstra
Recording i++
Integrating all node path
End

FIGURE 5. The load-balancing routing algorithm based on SR.

$$w(e) = 0.01 + c(e) = 0.01 + {}^{F(e)}\!/_{r(e)} \tag{5}$$

$$w(\boldsymbol{P}) = \sum_{e \in \boldsymbol{P}} w(e) \tag{6}$$

$$r(\boldsymbol{P}) = \min_{e \in \boldsymbol{P}} r(e) \tag{7}$$

The link, the residual bandwidth less than the required, would be deleted when satellite nodes in the heavy load zone route. Then, the link weight can be configured to generate the weight network. The greedy choice is used for link weight to find minimum weight path which can control the delay. So, Dijkstra algorithm is adopted where satellite nodes are starting points and the end is the central station node. If the satellite node has no path to the central station, the traffic carried by the node is rejected and lost. Fig. 5 shows the load-balancing routing algorithm based on SR.

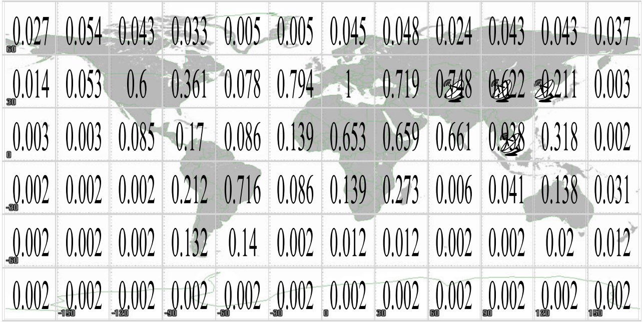
B. TIME COMPLEXITY

The time complexity analysis is based on Dijkstra algorithm, whose time complexity is $O(v^2)$, where v is the number of nodes in the network [20]. LSN has N satellite nodes, X gateway nodes, and 1 central station node. The heavy load zone has $y_{\mathrm{n}} \times y_{\mathrm{m}}$ nodes and $N \geq y_{\mathrm{n}} \times y_{\mathrm{m}}$. The complexity of routing in the light load zone is $O((N + X + 1)^2) = O(N^2)$, while the complexity of routing in the heavy load zone is $O((y_{\mathrm{n}} \times y_{\mathrm{m}}) \times (y_{\mathrm{n}} \times y_{\mathrm{m}} + X + 1)^2) = O((y_{\mathrm{n}} \times y_{\mathrm{m}})^3)$. So, the time complexity of the proposed algorithm is $O(N^2 + (y_{\mathrm{n}} \times y_{\mathrm{m}})^3)$.

With the extension of heavy load zone size $(y_{\mathrm{n}},\ y_{\mathrm{m}})$, the proposed algorithm complexity would increase. On the contrary, the complexity would decrease when the heavy load zone has fewer nodes. In extreme cases, the heavy load zone size is (0,0) while the light load zone has N satellite nodes.

TABLE 1. Parameters of the constellation.

Parameter	Value
The number of orbits	6
The number of satellites per orbit	12
The inclination of orbits	90° (polar orbit)
The angle between adjacent orbits	30°
The number of gateways	4
The bandwidth of satellite links	25
The bandwidth of feedback links	100



0.027	0.054	0.043	0.033	0.005	0.005	0.045	0.048	0.024	0.043	0.043	0.037
0.014	0.053	0.6	0.361	0.078	0.794	1	0.719	0.748	0.622	0.211	0.003
0.003	0.003	0.085	0.17	0.086	0.139	0.653	0.659	0.661	0.938	0.318	0.002
0.002	0.002	0.002	0.212	0.716	0.086	0.139	0.273	0.006	0.041	0.138	0.031
0.002	0.002	0.002	0.132	0.14	0.002	0.012	0.012	0.002	0.002	0.02	0.012
0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002	0.002

FIGURE 6. Traffic cells.

TABLE 2. Sizes of the heavy load zone.

The number of orbits	Sizes				
3	(3,2)	(3,3)	(3,4)	(3,5)	(3,6)
4	(4,2)	(4,3)	(4,4)	(4,5)	(4,6)
5	(5,2)	(5,3)	(5,4)	(5,5)	(5,6)
6	(6,2)	(6,3)	(6,4)	(6,5)	(6,6)
6	(6,12), i.e., all nodes in the network				

Under this circumstance, the time complexity of the proposed algorithm is $O(N^2)$ which is the lowest.

V. SIMULATION

A. SIMULATION SCENARIO

The proposed algorithm is applied to Walker constellation with satellite links, and parameters of the constellation are shown as Table. 1.

Traffic cells are divided in Fig. 6 which gives gateways location. The value in each cell is the traffic density obtained by prediction. Two traffic density distributions are considered in simulation, uniform distribution where all traffic density is 1 and prediction distribution as Fig. 6. The unit service value u is assumed to be 1, 2, and 3 for uniform distribution. The unit service value u is considered to be 4 and 5 for prediction distribution, because some traffic densities are small and the network has sufficient capacity to meet the traffic transmission demand.

The selection of the heavy load zone is the key of the proposed algorithm. In order to analyze the influence of the heavy load zone size on the proposed algorithm and choose the excellent size to improve performance with smaller cost, lots of sizes are studied under both distributions, shown as Table. 2.

The proposed algorithm is applied to the LSN under conditions of different traffic distributions, different unit service values, and different sizes. Programming language C++ is used for simulation to simulate network scenario and implement algorithms. The simulation results show the performance in terms of the average rejection ratio, the average relative throughput, the maximum link utilization, the average delay and the average jitter. The results of size (6,12) are used as thresholds for five indicators. Then, comparison with other algorithms shows advantages.

B. SIMULATION RESULT

1) THE AVERAGE REJECTION RATIO

If the traffic has no path to the central station, the transmission request of the traffic is rejected. The rejection ratio is defined by (8) which is ratio of the sum of rejected transmission request to the total traffic of the system. R stands for the rejection ratio, and $\boldsymbol{R}_S$ stands for the set of traffic cells that are rejected. The lower the average rejection ratio, the better the load-balancing performance of the algorithm.

$$R=\frac{\sum\limits_{a\in \boldsymbol{R}_S} f^{(a)}\times u}{\sum\limits_{k=0} f^{(k)}\times u}=\frac{\sum\limits_{a\in \boldsymbol{R}_S} f^{(a)}}{\sum\limits_{k=0} f^{(k)}}\times 100\% \tag{8}$$

Fig. 7 shows the simulation result of the average rejection ratio under the uniform distribution. The average rejection ratio of size (6,12) is optimal under all unit service values, which can be compared with other sizes. The average rejection ratio increases when u increases with the same size of the heavy load zone. When the number of orbits in the heavy load zone is the same, the average rejection ratio gradually decreases with the increase of the number of satellites per orbit. When the number of satellites per orbit is the same, the rejection ratio also decreases with the increase of the number of orbits.

The simulation result of the average rejection under the prediction distribution is shown as Fig. 8. The average rejection ratio of size (6,12) is 0%. The average rejection ratio shows a decreasing trend with the extension of heavy load zone size. Therefore, the rejection ratio is improved as the

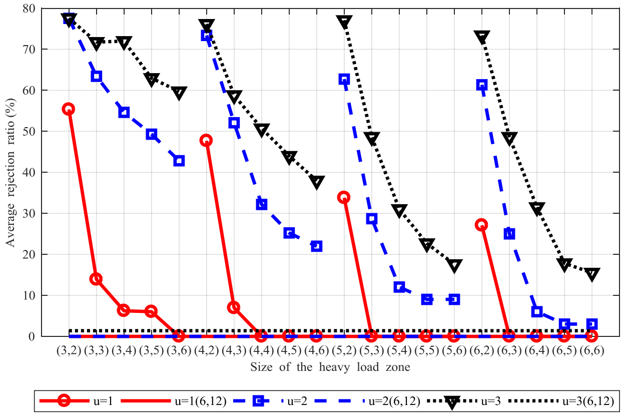


FIGURE 7. Average rejection ratio (uniform distribution).

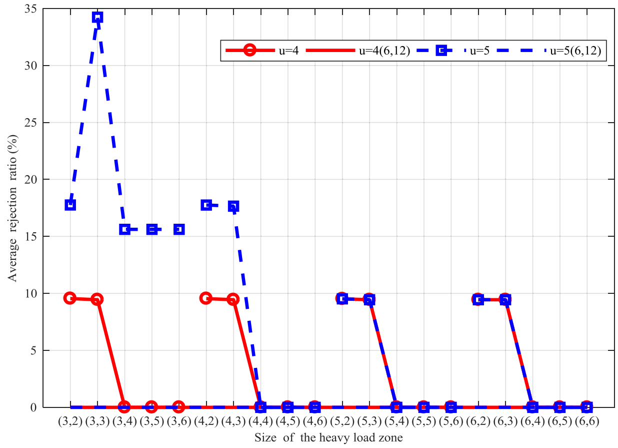


FIGURE 8. Average rejection ratio (prediction distribution).

size extends which indicates that load balancing performance becomes better under both distributions.

2) THE AVERAGE RELATIVE THROUGHPUT

The relative throughput is the sum of traffic transmitted successfully, as (9). T stands for the relative throughput and $\boldsymbol{S}_{\mathrm{S}}$ is the set of traffic cells transmitted successfully. The relative throughput and the rejection ratio are complementary conceptually. The relative throughput quantitatively reflects the traffic transmitted successfully, while the rejection ratio presents the proportional relationship. The larger the average relative throughput, the better the load-balancing performance of the algorithm.

$$T = \sum_{b \in \boldsymbol{S}_{\mathrm{S}}} f^{(b)} \times u \tag{9}$$

The average relative throughput under the uniform distribution is presented in Fig. 9. The average relative throughput of size (6,12) is the largest. When the number of orbits in the heavy load zone is the same, the average relative throughput increases as the number of satellites per orbit increases. When the number of satellites per orbit is the same, the average relative throughput also increases as the number of orbits increases. As Fig. 7 and Fig. 9, the average relative throughput increases as the average rejection ratio decreases with the same u. Besides, when the size of the heavy load zone is the same, the throughput increases as u increases while the rejection ratio also increases, but the difference from the optimal threshold of throughput increases. Fig.10 is the simulation result under the prediction distribution where the characteristics are similar to Fig. 9.

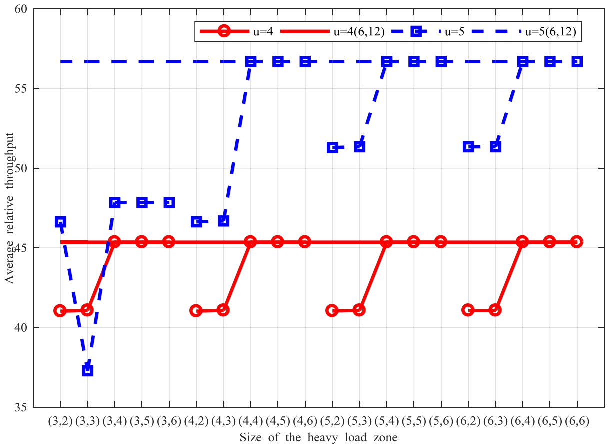


FIGURE 10. Average relative throughput (prediction distribution).

3) THE MAXIMUM LINK UTILIZATION

The maximum link utilization is defined as (10). $F(e)$ is the current traffic carried by link e and $b(e)$ is the bandwidth of link e. The maximum link utilization is significant when the rejection ratio is small or 0%. The smaller the maximum link utilization, the better the load-balancing performance of the algorithm.

$$U_{\mathrm{e}} = (\max_{e \in E} \frac{F(e)}{b(e)}) \times 100\% \tag{10}$$

Fig. 11 presents the maximum link utilization under the uniform distribution. The maximum link utilization of size (6,12) is the smallest which offers the lower threshold. When the number of orbits is the same, the maximum link utilization decreases as the number of satellites per orbit increases.

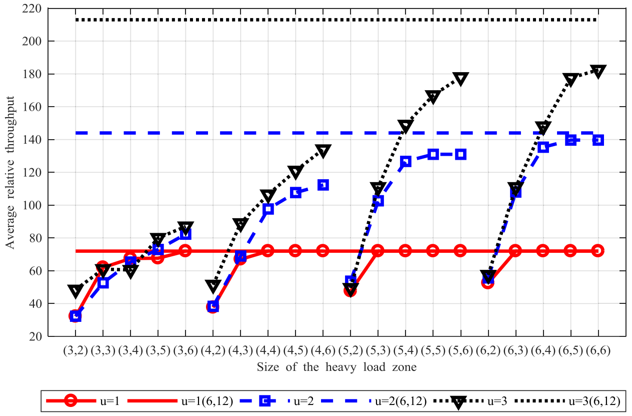


FIGURE 9. Average relative throughput (uniform distribution).

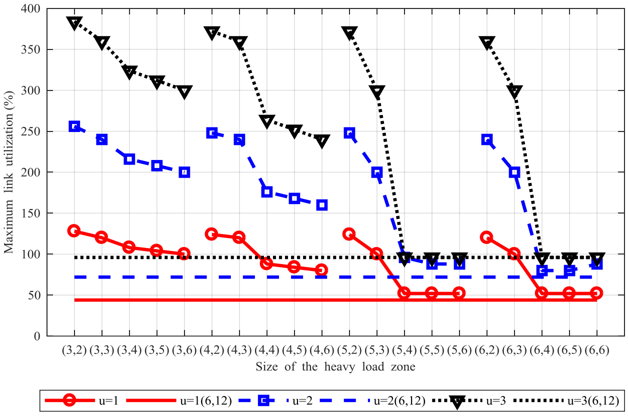


FIGURE 11. Maximum link utilization (uniform distribution).

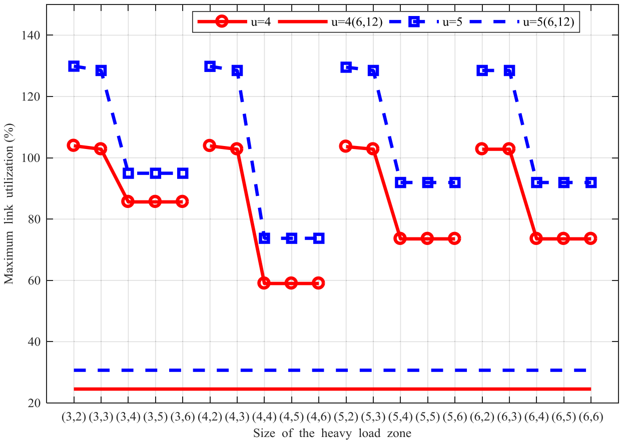


FIGURE 12. Maximum link utilization (prediction distribution).

When the number of satellites per orbit is the same, the maximum link utilization decreases as the number of orbits increases. The maximum link utilization would increase with the increase of u. This result pays more attention to sizes with small rejection ratio, because the link utilization greater than 100% will cause rejection.

Fig. 12 presents the result under the prediction distribution. The maximum link utilization decreases with the extension of size. However, due to the extreme imbalance of the prediction distribution, results have a gap from the lower threshold.

4) THE AVERAGE DELAY

The average delay is the mean of traffic delay transmitted successfully and reflects the cost of the algorithm, which is defined as (11). $\boldsymbol{S_S}$ is the set of traffic cells transmitted successfully. $|\boldsymbol{S_S}|$ is the number of cells in $\boldsymbol{S_S}$. $d^{(b)}$ is the delay of traffic $f^{(b)}$, which is the sum of routing processing delay and propagation delay. The smaller the average delay, the smaller the resources occupied.

$$D_T = \frac{\sum\limits_{b \in \boldsymbol{S_S}} d^{(b)}}{|\boldsymbol{S_S}|} \tag{11}$$

Simulation results of the average delay under the uniform distribution are shown in Fig. 13. The result of size (6,12) is given as the threshold. The delay and cost will increase with the increase of the traffic to get more throughput. However, the average delay decreases as u increases with some sizes. This phenomenon is caused by big rejection ratio, and only part of traffic can occupy resources. When the rejection ratio is small, the average delay increases as the size extends. The result of the prediction distribution is presented in Fig. 14 which confirms the conclusion again.

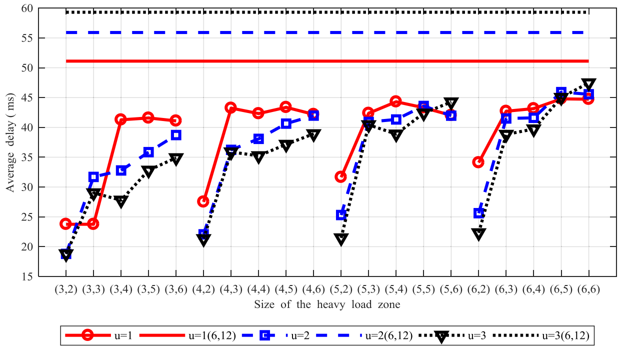


FIGURE 13. Average delay (uniform distribution).

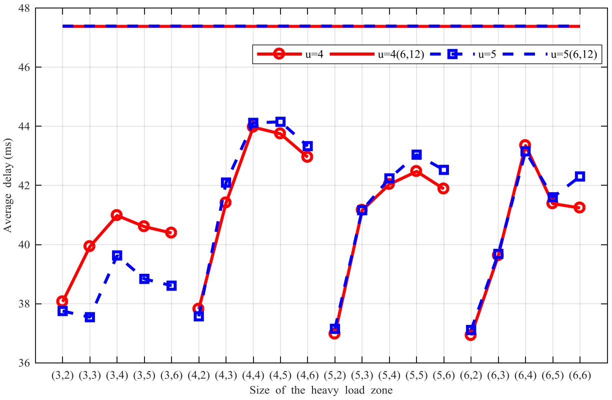


FIGURE 14. Average delay (prediction distribution).

5) THE AVERAGE JITTER

The average jitter is defined as (12) which is the mean of traffic delay jitter transmitted successfully. $\boldsymbol{S_S}$ is the set of traffic cells transmitted successfully. $|\boldsymbol{S_S}|$ is the number of cells in $\boldsymbol{S_S}$. $j_T^{(b)}$ is the jitter of traffic $f^{(b)}$, which is the standard deviation of delay $d^{(b)}$. The average jitter reflects the algorithm stability which should be smaller.

$$J_T = \frac{\sum\limits_{b \in \boldsymbol{S_S}} j_T^{(b)}}{|\boldsymbol{S_S}|} \tag{12}$$

Simulation results under uniform and prediction distribution are shown in Fig. 15 and Fig. 16, respectively. The average jitter is fluctuant as the heavy load zone size changes and is acceptable when the rejection ratio is small.

6) COMPARISON WITH DIFFERENT ALGORITHMS

The performance of the proposed algorithm is compared with Dijkstra, HRA, and JDDA in terms of the average rejection

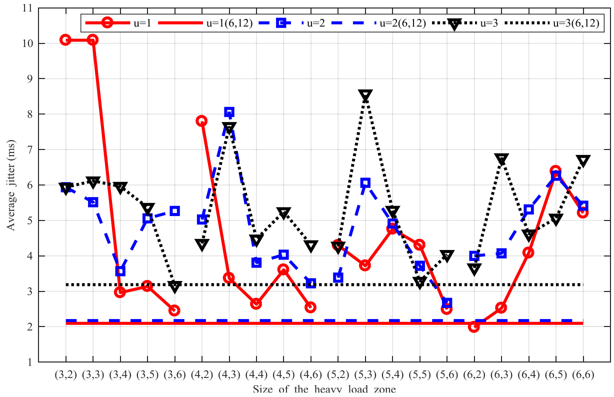


FIGURE 15. Average jitter (uniform distribution).

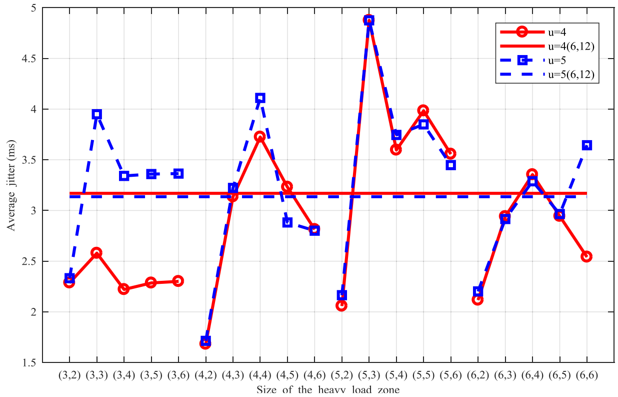


FIGURE 16. Average jitter (prediction distribution).

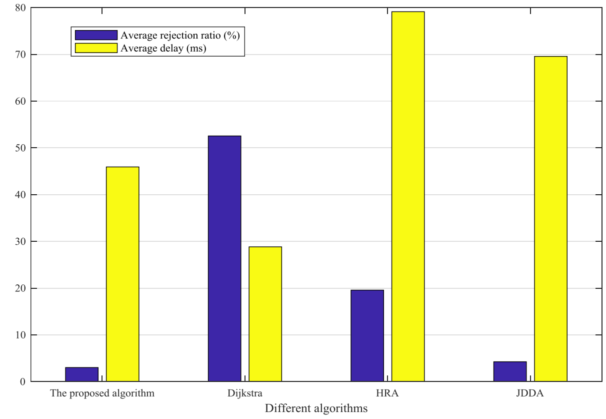


FIGURE 17. Performance comparison (rejection ratio and delay).

ratio, reflecting the benefit, and the average delay reflecting the cost, which are key indicators to system evaluation. The simulation scenario introduced in this section is used, where ground gateway stations are arranged within a limited area, which is different from initial scenarios of these routing algorithms. The uniform traffic distribution is adopted and the unit service value is assumed to be 2. The size of the proposed algorithm selects (6,5) in this scenario because of benefit, cost and time complexity.

Fig. 17 is the simulation result of different algorithms comparison. Dijkstra algorithm selects the shortest paths to route, so the average delay is the smallest but the average rejection ratio is the highest. HRA takes advantage of left load rate to improve network throughput and reduce the rejection ratio, but the average delay is the highest. JDDA combines Dijkstra algorithm and DFS which improves the network congestion, and the average rejection ratio is smaller. However, the average delay is higher with gateways arranged within the limited area. The proposed algorithm has the smallest rejection ratio and smaller delay because SR principle is used to load balancing, which confirms algorithm advantages under centralized distribution of gateways.

C. SIMULATION SUMMARY

The simulation results reflect the benefit of the proposed algorithm on load balancing, and indicate the impact of the heavy load zone size. The average rejection ratio decreases, the average relative throughput increases, and the maximum link utilization decreases as the size extends, which confirm the load balancing performance. Note that the extension of size refers to the increase of two dimensions, the number of orbits and the number of satellites per orbit. The increase of one dimension only can get little benefit, such as (3,2), (4,2), (5,2), (6,2). For another, the average delay also increases with the extension of size that contributes to the increase of cost and resources occupied. The time complexity of the proposed algorithm is $O(N^2+(y_n \times y_m)^3)$, which increases dramatically as the size extends.

Therefore, if the size is too large, the delay and resources consumption also increase. If the size is too small, the congestion avoidance would be very difficult. An efficient size of the heavy load zone is necessary. In this scenario, the size (6,5) gets better load balancing performance with small cost and moderate time complexity, and is used in the proposed algorithm to compare with Dijkstra, HRA, and JDDA. Comparison with other algorithms shows that the proposed algorithm can improve network congestion with smaller cost caused by gateways arranged within a limited area.

VI. CONCLUSION

The load-balancing routing algorithm based on segment routing is proposed for LEO satellite networks. The network congestion caused by gateways arranged within a limited area is improved. The proposed algorithm refers to SR and dynamically divides the light and the heavy load zone. The pre-balancing minimum spanning tree is the basis for the routing in the light load zone, while the routing in the heavy load zone refers to the minimum weight path based on congestion index. The size of the heavy load zone affects the algorithm significantly, and is analyzed to select the appropriate zone. Then, the proposed algorithm is compared with other routing algorithms to verify the load balancing performance. Simulation results confirm that the average rejection ratio, the average relative throughput, and the maximum link utilization are improved significantly with little increase of delay as the size extends, and present better load-balancing performance.

The results could be a solution for congestion problem in LEO satellite networks, and the entry point is traffic return. The extension to all types of traffic and the design of on-satellite router could get more attention in further discussion.

REFERENCES

[1] P. Noschese, S. Porfili, and S. D. Girolamo, "ADS-B via iridium NEXT satellites," in *Proc. TIWDC-ESAV*, Capri, Italy, Sep. 2011, pp. 213–218.

[2] 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. doi: 10.1109/MWC.2019.1800299.

[3] T. Taleb, Y. Hadjadj-Aoul, and T. Ahmed, "Challenges, opportunities, and solutions for converged satellite and terrestrial networks," *IEEE Wireless Commun.*, vol. 18, no. 1, pp. 46–52, Feb. 2011.

[4] J. Xu, W. Lu, and G. X. Zhang, "Traffic simulation of broadband LEO constellation satellite communication system," (in Chinese), *Appl. Electron. Techn.*, vol. 45, no. 3, pp. 67–70, 2019.

[5] H. Gao, L. Wang, W. D. Huang, and L. Y. Sun, "Routing optimization method for fast return of data on overseas satellites in beidou global navigation satellite system," (in Chinese), *Chin. Space Sci. Technol.*, vol. 38, no. 2, pp. 9–15, 2018.

[6] M. Sheng, Y. Wang, J. Li, R. Liu, D. Zhou, and L. He, "Toward a flexible and reconfigurable broadband satellite network: Resource management architecture and strategies," *IEEE Wireless Commun.*, vol. 24, no. 4, pp. 127–133, Aug. 2017.

[7] Y. Lu, F. Sun, and Y. Zhao, "Virtual Topology for LEO Satellite Networks Based on Earth-Fixed Footprint Mode," *IEEE Commun. Lett.*, vol. 17, no. 2, pp. 357–360, Feb. 2013.

[8] K. Kar, M. Kodialam, and T. V. Lakshman, "Minimum interference routing of bandwidth guaranteed tunnels with MPLS traffic engineering applications," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 12, pp. 2566–2579, Dec. 2000.

[9] W. J. Jiang and P. Zong, "A discrete-time traffic and topology adaptive routing algorithm for LEO satellite networks," *Int. J. Commun. Netw. Syst. Sci.*, vol. 4, pp. 42–52, Jan. 2011.

[10] Y. LU, F. Sun, Y. Zhao, H. Li, and H. Liu, "Distributed traffic balancing routing for LEO satellite networks," *Int. J. Comput. Netw. Inf. Secur.*, vol. 1, pp. 19–25, Nov. 2014.

[11] C. X. Liu and C. S. Miao, "A hybrid routing algorithm based on load balancing for LEO satellite networks," *Int. J. Wireless Mobile Comput.*, vol. 8, no. 4, pp. 359–365, Aug. 2015.

[12] M. Jia, S. Y. Zhu, L. F. Wang, Q. Guo, H. T. Wang, and Z. H. Liu, "Routing algorithm with virtual topology toward to huge numbers of LEO mobile satellite network based on SDN," *Mobile Netw Appl*, vol. 23, no. 2, pp. 285–300, Apr. 2018.

[13] C. Filsfils, N. K. Nainar, C. Pignataro, J. C. Cardona, and P. Francois, "The segment routing architecture," in *Proc. IEEE GLBECOM*, San Diego, CA, USA, Dec. 2015, pp. 1–6.

[14] A. Sgambelluri, F. Paolucci, A. Giorgetti, F. Cugini, and P. Castoldi, "Experimental demonstration of segment routing," *J Lightw. Technol*, vol. 34, no. 1, pp. 205–212, Jan. 2016.

[15] J. P. Sheu and Y. C. Chen, "A scalable and bandwidth-efficient multicast algorithm based on segment routing in software-defined networking," in *Proc. IEEE ICC*, Paris, France, May 2017, pp. 1–6.

[16] M.-C. Lee and J.-P. Sheu, "An efficient routing algorithm based on segment routing in software-defined networking," *Comput. Netw.*, vol. 103, pp. 44–55, Jul. 2016.

[17] X. L. Hou, M. Q. Wu, and M. Zhao, "An optimization routing algorithm based on segment routing in software-defined networks," *Sensors*, vol. 19, no. 1, pp. 49–70, Dec. 2019.

[18] E. Moreno, A. Beghelli, and F. Cugini, "Traffic engineering in segment routing networks," *Comput. Netw.*, vol. 114, pp. 23–31, Feb. 2017.

[19] Y. L. Xiao, T. Zhang, and L. Liu, "Addressing subnet division based on geographical information for satellite-ground integrated network," *IEEE Access*, vol. 6, pp. 75824–75833, 2018.

[20] S. X. Gao, *Graph Theory and Network Flow Theory*, Beijing, China: Higher Education Press, 2009, pp. 11–17.

WEI LIU was born in 1994. He received the B.S. degree in communication engineering from Xidian University, Xi'an, China, in 2017. He is currently pursuing the master's degree in information and communication engineering with the China Academy of Space Technology, Beijing, China.

His research interests include communication networks and satellite communication.

YING TAO was born in 1974. She received the B.S.E.E. degree from Northern Jiaotong University, Beijing, China, in 1997, and the Ph.D. degree from Beijing Jiaotong University, in 2004.

She is currently a Research Fellow with the China Academy of Space Technology. Her research interests include satellite communication and spatial information networks.

LIANG LIU was born in 1986. He received the B.S.E.E. and Ph.D. degrees from the Beijing University of Aeronautics and Astronautics, Beijing, China, in 2009 and 2017, respectively.

He is currently an Engineer with the China Academy of Space Technology. His research interests include network coding, mobile communication, and satellite communication.