IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 24, 2021, accepted January 7, 2022, date of publication February 10, 2022, date of current version February 18, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3149961

Capacity Analysis of LEO Mega-Constellation Networks

NINGYUAN WANG[1,2], LIANG LIU[2], ZHAOTAO QIN[2], BINGYUAN LIANG[2], AND DONG CHEN[1,2]

[1]Peng Cheng Laboratory, Shenzhen, Guangdong 518052, China
[2]Institute of Telecommunication and Navigation Satellites, China Academy of Space Technology, Beijing 100094, China

Corresponding author: Ningyuan Wang (ningyuan.wang@foxmail.com)

This work was supported in part by the National Key Research and Development Program of China under Grant 2000YFB1806102, and in part by the National Natural Science Foundation of China under Grant 61875230.

ABSTRACT This paper proposes a complete network capacity analysis framework for low-earth-orbit (LEO) mega-constellations, where a network capacity estimation problem considering the link packet loss rate is formulated with the support of a time-variant network topology model and a task distribution model. This problem is solved in two steps. First, without considering the link packet loss rate, an improved fully polynomial-time approximation (IFPTA) algorithm is proposed to provide a sub-optimal solution to the multi-commodity flow (MCF) problem, in which a simpler definition of a commodity is given and proved to be equivalent to the original problem. Second, a Jackson-network-based capacity fallback approach is proposed to control the link packet loss rate below a given threshold. Numerical results illustrate the superiority of the proposed IFPTA algorithm in terms of accuracy and time complexity compared to existing solutions. In addition, the capacity characteristics of mega-constellations are analyzed by utilizing the proposed capacity analysis framework, including the relationship between constellation size and capacity, network capacity bottlenecks, and the influence of task distributions.

INDEX TERMS LEO mega-constellation, network capacity, multi-commodity flow problem, Jackson network.

I. INTRODUCTION

The low-earth-orbit (LEO) telecommunication constellation network is foreseen as a merging interconnection method for future sixth-generation (6G) systems [1], possessing the advantages of seamless global coverage, high robustness, and low delay compared with other space-based communication systems. In recent years, the upsurge of LEO constellation construction has reappeared around the world [2]. In particular, some LEO mega-constellation projects, such as Starlink, Oneweb, and Lightspeed, have entered the deployment phase, launching hundreds or thousands of satellites to forge giant and broadband constellation networks [3]. The tremendous constellation scale has resulted in a substantial increase in the number of accessible users, providing a large network capacity by cooperating with the application of laser inter-satellite links (ISLs) and high-frequency user-to-satellite links (USLs) [4].

In constellation design, network capacity is an important metric for evaluating the service capability of LEO constellation networks. Unlike the ground network, the LEO network shows complete symmetry, that is, each satellite has an equal opportunity to shift above any user over time. Consequently, it is impossible to increase the network capacity by expanding congested links, which is a common solution for terrestrial networks. Therefore, it is indispensable for constellation designers to understand the network capacity performance of the LEO constellation in different application scenarios and ensure that the capacity meets the requirements. Therefore, the capacity analysis of the LEO constellation is particularly vital.

The network capacity dimensioning technique has been widely studied. Dai. gave theoretical results of minimum link capacity to meet the requirement when all nodes in the network send an equal amount of traffic to every other node in the same network except itself (''all-to-all'') [5], using stochastic processing model, while the same goal was also achieved by Sun *et al.* via ''cut on a graph'' method [6]. Liu *et al.* extended the above work to the LEO-MEO hybrid

The associate editor coordinating the review of this manuscript and approving it for publication was Rute C. Sofia.

constellation and provided an explicit expression of the constellation capacity in the "all-to-all" scenario of the 2D-Torus network [7]. Werner *et al.* shifted the capacity dimensioning problem into minimizing the worst-case link capacity, which was solved using linear programming (LP) [8], [9]. Aiming at the walker-star constellation, Xiao *et al.* used discrete event simulation to analyze network capacity, as well as the relationship between link bandwidth and link packet loss rate of different routing algorithms [10]. However, theoretical methods in [5]–[9] have very limited scope of application, which are only competent for 2D-Torus topology, and the task distribution can only be "all-to-all." Moreover, both LP and discrete event simulation are quite time consuming for the capacity analysis of mega-constellations.

The network capacity analysis problem of the LEO constellation can be classified as a multi-commodity flow (MCF) problem [11] because the traffic between different origin-destination pairs is not exchangeable. To make the traffic throughput approach the upper bound of the network capacity, a number of studies have been dedicated to both accuracy and low-complexity. Documents based on Lagrange relaxation and linear programming have provided an algorithm for solving arbitrary MCF problems [12], [13]. Garg and Konemann proposed a fully polynomial-time approximation (FPTA) algorithm [14], converting the LP problem into multiple iterations of shortest path routing with path length updating and traffic accumulation. Karakosta further reduced the time complexity of the FPTA algorithm [15], using a Dijkstra algorithm to calculate the path length of all commodities sharing one source, and provided an explicit algorithm for both network capacity and traffic on each edge. However, its capacity result is not sufficiently close to the optimal value; therefore, the link capacity cannot be effectively utilized. Moreover, the aforementioned solutions do not consider the influence of queuing in the network, which can suffer severe packet loss when links are fully occupied. Research on capacity analysis for LEO mega-constellations is still open.

In this paper, a complete framework of network capacity analysis for LEO mega-constellation is envisaged, which has low time complexity and considers the impact of queuing on network capacity. The main contributions of this study are summarized as follows.

- We propose a framework for network capacity analysis that includes network topology modeling, task distribution modeling, and a network capacity estimation algorithm. Subsequently, the network capacity estimation problem considering the link packet loss rate is formulated into an MCF problem. A less complex definition of the commodity is given by proving its equivalence with the original definition.
- This problem is solved in two steps. First, an improved fully polynomial-time approximation (IFPTA) algorithm is proposed to determine the maximum network capacity without considering network queuing. Then, the network capacity fallback is studied by utilizing the Jackson queuing network model, deducing the network capacity to maintain the maximum packet-loss rate under a threshold.
- Simulations are performed in two aspects. Firstly, the algorithm performance is analyzed based on the metrics of accuracy, complexity, and time sampling interval. Second, we explore the capacity characteristics of the LEO mega-constellation network, including the impacts of constellation size, ratio of USL to ISL capacity, and different task distributions.
- Based on the simulation results, we make two conjectures of the network capacity by observation and justify the correctness in the case of a 2D-torus network with a uniform distribution. Therefore, an empirical algorithm is proposed to provide a rough estimation of network capacity when the constellation size is too large.

The remainder of this paper is organized as follows. In Section II, the framework of the time-variant capacity analysis system is presented, including the network topology and task distribution modules. The MCF problem of network capacity estimation is formulated in Section III and solved using the proposed IFPTA algorithm while considering network queuing. Simulation results and discussions are provided in Section IV, along with an empirical method for capacity estimation. The paper is finally concluded with section V, where the main findings of our work are highlighted.

II. MODELING OF CAPACITY ANALYSIS SYSTEM FOR LEO MEGA-CONSTELLATION

A. ARCHITECTURE OVERVIEW

We consider a constellation-terrestrial network that consists of a single-layer LEO constellation and ground users, as shown in FIGURE 1. Satellites in the constellation are connected with neighbors by laser inter-satellite links (ISLs). Meanwhile, each terrestrial user connects with its visible satellites with microwave user-satellite-links (USLs) in the form of spot beams enabled by phased-array antennas, where all links are bidirectional. We do not distinguish earth stations and individual users in this work because their difference in capacity is only in traffic throughput, which can be flexibly modified in the task weight distribution model (Section II-F). Among the users, there are traffic tasks, which are communication requirements through the constellation network. In this study, we use the description of graph theory, where both satellites and users are nodes, and both ISLs and USLs are edges. As the objective is to estimate the network capacity, its definition must be clarified. The network capacity of a constellation-terrestrial network is defined as follows:

Definition 1: The network capacity $\lambda_\eta(t)$ of a constellation-terrestrial network at instant t is the maximum possible flow of the network under a given topological connection, link capacities, and service distribution such that the maximum packet loss rate of edges does not exceed a threshold η.

To dimension the maximum network carrying ability, we propose a complete framework for network capacity

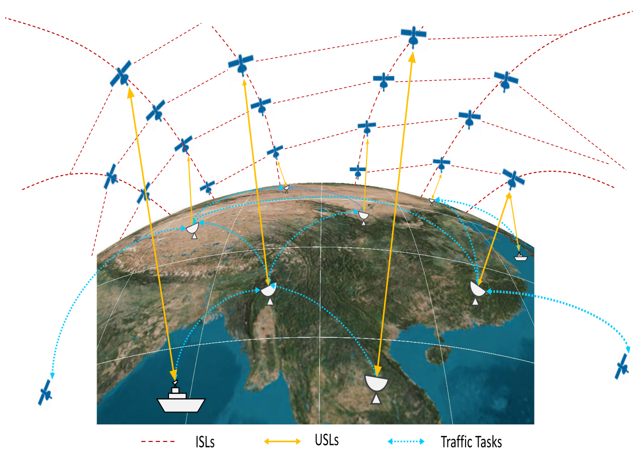


FIGURE 1. The scenario of the constellation-terrestrial network, in which terrestrial users deliver traffic tasks to each other through ISLs and USLs in the constellation network.

analysis, as illustrated in FIGURE 2. The framework can be decomposed into three modules: network topology, task weight distribution, and capacity estimation. The network topology module constructs a time-variant adjacency matrix that describes the connections among nodes and edge capacities by letting the element in the row v and the column w be the edge capacity from node v to node w. The task weight distribution module formulates a normalized matrix that describes the task weight among user nodes, which represents the global task weight distribution. We denote the task weight from user i to user j as the ratio of the amount of traffic from user i to user j to the sum of all traffic between any two user nodes. Because the actual global task weight distribution is unknown, some typical distributions are given in this section. Finally, using the results of the aforementioned modules as input, the capacity estimation module solves the network capacity estimation problem in two steps: solving the MCF problem by IFPTA algorithm and falling back the obtained maximum capacity according to the required packet loss rate. The construction method of the network topology module and task weight distribution module is detailed in this section from B to F, and that of the capacity estimation module is given in Section III.

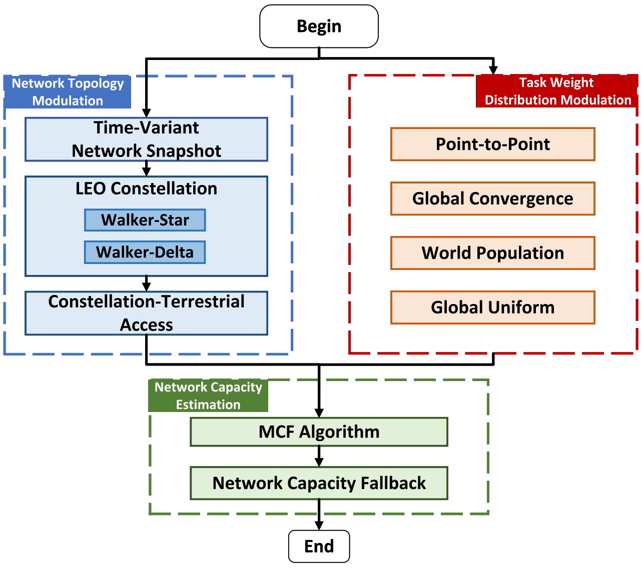

Begin
Network Topology Modulation
Time-Variant Network Snapshot
LEO Constellation
Walker-Star
Walker-Delta
Constellation-Terrestrial Access
Task Weight Distribution Modulation
Point-to-Point
Global Convergence
World Population
Global Uniform
Network Capacity Estimation
MCF Algorithm
Network Capacity Fallback
End

FIGURE 2. The complete network capacity analysis framework for LEO mega-constellations.

B. TIME-VARIANT NETWORK TOPOLOGY SNAPSHOT

In contrast to the ground network, the LEO satellites that form the constellation network exhibit high-speed movement. Not only the connection among satellites, but also the satellite-user access relationships are highly dynamic. Therefore, the network capacity is also time variant; therefore, it is necessary to consider the impact of topology changes on the constellation network capacity. The time-varying satellite-ground network topology matrix can be expressed as

$$\mathrm{A}(t) \in N \times N, \quad t \in [0, T] \tag{1}$$

where N is the total number of nodes of the constellation-terrestrial network, including satellite and user nodes, and T is the length of the simulation. However, the topology among nodes changes frequently and is likely to be non-regressive. Therefore, it is time-consuming to analyze the capacity by traversing each topology connection state. Here, we use discrete sampling with equal spacing for the topology variance. In other words, the network topology is sampled as a series of "snapshots" at a certain time interval. By taking $N_T + 1$ samples in $t \in [0, T]$ with a sample interval $\Delta t = T/N_T$, the time-varying network topology matrix can be represented by a three-dimensional tensor:

$$\boldsymbol{A} \in N \times N \times N_T \tag{2}$$

where $\boldsymbol{A}[t] \in N \times N$ is a slice of the cube, called a "snapshot" of the network, representing the topological connection of the network at the sampling time t. In this work, it is assumed that the flow in each snapshot has reached a steady state, that is, the incoming traffic in the network is equal to the outgoing traffic at each moment; therefore, different snapshots are independent of each other. Therefore, all network modeling and capacity estimation processes below are dedicated to ONE SNAPSHOT, and we omit the argument t in the following expression for simplicity.

C. ADJACENCY MATRIX DESCRIPTION OF CONSTELLATION-TERRESTRIAL NETWORK

Consider a constellation-terrestrial network topology snapshot at instant t, including N_{Sat} satellite nodes and N_U user nodes on the ground. In addition, the user node performs two-way communication with other users, behaving as both source and destination. However, user nodes do not forward traffic as satellite nodes do. To avoid this, the sending and receiving functions of a user node are equivalent to two nodes located in the same place without connection. Thus, the set of constellation satellites can be expressed as $V_{Sat} = \{v_1, \cdots, v_{N_{Sat}}\}$, the set of ground users is $V_U = \{v_1, \cdots, v_{N_U}\}$, the set of ground sending nodes is $V_{US} = \{v_1, \cdots, v_{N_{US}}\}$, and the set of receiving ground stations is $V_{UD} = \{v_1, \cdots, v_{N_{UD}}\}$, where $0 < N_{US}, N_{UD} \leq N_U$, $N = N_{Sat} + N_{US} + N_{UD}$. We denote by $V = \{v_1, \cdots, v_N\}$ the set includes all nodes in the network. Hereby, the network topology can be represented by the block adjacency matrix $\boldsymbol{A} \in N \times N$ as:

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_{Sat} & \boldsymbol{0} & \boldsymbol{A}_{UD} \\ \boldsymbol{A}_{US} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}. \tag{3}$$

The square matrix $\boldsymbol{A}_{Sat} \in N_{Sat} \times N_{Sat}$ is the adjacency matrix of the low-orbit constellation that describes the topological connection relationship of the low-orbit constellation network. $\boldsymbol{A}_{\text{US}} \in N_{US} \times N_{Sat}$ and $\boldsymbol{A}_{UD} \in N_{Sat} \times N_{UD}$ are the connection relationship matrices from the sending node to the satellite and from the satellite to the receiving node, respectively. Because there is no connection between the sending and receiving nodes or edges among ground users, the corresponding parts are all-zero matrices.

D. CONSTELLATION MODEL

In this section, we focus on the topology of the constellation merely, which is represented by the matrix $\boldsymbol{A}_{Sat}$. The LEO satellites in a constellation are distributed on N_O orbital planes, and each orbital plane has N_{SpO} satellites, $N_{Sat} = N_O \times N_{SpO}$. According to its orbital plane distribution, the constellation can be classified as Walker-Star and Walker-Delta configurations.

The Walker-Star configuration is mostly used in polar orbit constellations, where the ascending nodes of the N_O orbital planes in the constellation are equally spaced within a range of 180° longitude, as shown in FIGURE 3(a). In the same orbital plane, satellites moving from the south pole to the north pole are called ascending satellites; otherwise, they are called descending satellites. The ascending and descending orbit satellites in the Walker-Star configuration are in two hemispheres. The adjacent ascending and descending orbiting satellites cannot establish ISLs owing to their high relative speed, thus forming a reverse seam. Except for the satellites on both sides of the reverse seam, each satellite in the remaining orbital planes contains four ISLs, two of which are connected to two adjacent co-orbiting satellites, and the other two are in different orbits. The ISLs are connected to the satellites in two adjacent orbital planes. The satellites on both sides of the reverse seam contains three ISLs.

The walker-delta constellation configuration is primarily used in low-inclination constellations. The ascending nodes of the N_O orbital planes in the constellation are equally spaced at 360° longitude, as shown in FIGURE 3(b). The orbits of the Walker-Delta constellation are evenly distributed around the world; therefore, there is no reverse seam. Each satellite in the constellation contains two ISLs connected with co-orbiting neighboring satellites, and the other two with satellites in neighboring orbits. Interestingly, an ascending satellite might be geographically close to a descending satellite because their orbits are crossed. However, the ISLs between them are difficult to establish because of their high relative speed, which is the same reason as the formation of the reverse seam in the Walker-Star configuration. Therefore, we can also assume that there are reverse seams everywhere in the walker-delta constellation configuration.

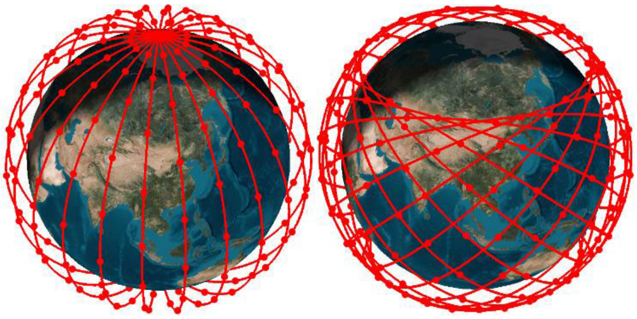

(a) Walker-Star (b) Walker-Delta

FIGURE 3. LEO constellations of 216 satellites with different configurations.

Thus far, two types of constellation configurations have been represented by the adjacency matrix $\boldsymbol{A}_{Sat}$.The variable $a_{i,j}^{Sat} \in \boldsymbol{A}_{Sat}$ is assigned as the capacity of the ISL from node i to node j at the instant t, which can be represented as

$$a_{i,j}^{Sat} = \begin{cases} C_{ISL}, & \exists \left(v_i, v_j\right), \ \forall v_i, v_j \in V_{Sat}, \ i \neq j \\ 0, & Otherwise, \end{cases} \tag{4}$$

$\left(v_i, v_j\right)$ represents the connection where the first item points to the second item and it is assumed that the capacity of the ISLs is a constant C_{ISL}.

E. USER ACCESS MODEL

To get closer to reality, the access relationships between the constellation and terrestrial users are determined by their relative positions. However, the problem arises as to how to model massive numbers of users worldwide. The idea is to represent all users in a certain area by a single node in the center. A common practice is to divide the Earth's surface into small grids at equal intervals of latitude and longitude [16]. Consequently, there are many more user nodes in polar regions than in equatorial regions, which is contrary to the actual situation.

For this reason, we proposed an approximate equal-area subdivision method for global user representation, which generates earth surface subdivisions whose area almost equals the target one. The ideal grid area is given as $L_X \times L_Y$, where L_X and L_Y are the lengths along the lines of latitude and longitude, respectively. **Algorithm 1** summarizes the occurrence of the equal-area subdivision.

The algorithm firstly divides the earth area by N_Y rows, which are in the shape of spherical belts, with the width as

Algorithm 1 Approximate Equal-Area Global Subdivi-Sion

```
Input: target length along line of latitude $L_X$, target length along line of longitude $L_Y$, Earth radius $R$
Output: total grid number $N_U$, latitude boundaries $\varphi_i^-$ and $\varphi_i^+$, longitude boundaries $\theta_{i,j}^-$ and $\theta_{i,j}^+$, $\forall i \in [1, \cdots, N_Y], \forall j \in [1, \cdots, N_X^i]$
Begin
$N_Y \Leftarrow \lfloor \pi \cdot R / L_Y \rfloor$
$\Delta\varphi \Leftarrow \pi / N_Y$
for each $i \in [1, \cdots, N_Y]$ do
    $\varphi_i^- \Leftarrow -\pi/2 + (i-1) \cdot \Delta\varphi$, $\varphi_i^+ \Leftarrow -\pi/2 + i \cdot \Delta\varphi$
    $S_i \Leftarrow 2\pi R^2 \left(\cos\varphi_i^- - \cos\varphi_i^+\right)$
    $N_X^i \Leftarrow \lfloor S_i / (L_X L_Y) \rfloor$
    $\Delta\theta_i \Leftarrow 2\pi / N_X^i$
    for each $j \in \left[1, \cdots, N_X^i\right]$ do
        $\theta_{i,j}- \Leftarrow (j-1) \cdot \Delta\theta_i$, $\theta_{i,j}^+ \Leftarrow j \cdot \Delta\theta_i$
    end for
end for
$N_U \Leftarrow \sum_{i=1}^{N_Y} N_X^i$
End
```

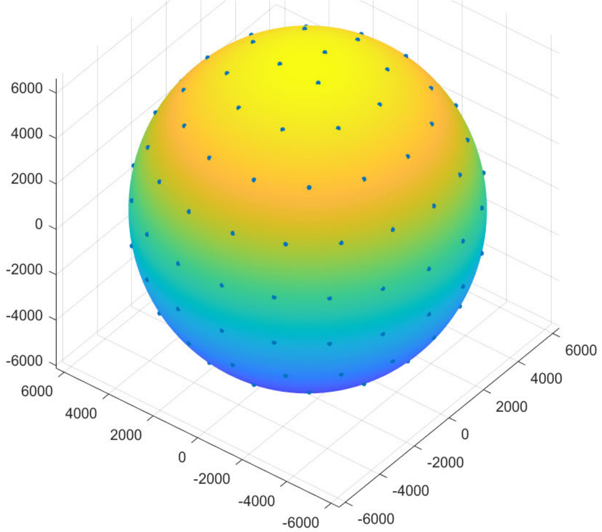


FIGURE 4. User distribution generated by Algorithm 1 with $L_X = L_Y = 2000$ *km*.

close to L_Y as possible. And all latitude boundaries φ_i^- and φ_i^+ can be obtained. Then, for each belt, the algorithm decides the number for grids by taking the ratio of the belt area S_i to the target area $L_X \times L_Y$ and rounding down. Thus, longitude boundaries $\theta_{i,j}^-$ and $\theta_{i,j}^+$ can also be calculated. So far, we completed the partition of grids, which is a total of $N_U = \sum_{i=1}^{N_Y} N_X^i$. Thus, the coordinates of the user node in the grid g_{ij} can be derived as $\left(\left(\varphi_i^- + \varphi_i^+\right)/2, \left(\theta_{i,j}^- + \theta_{i,j}^+\right)/2\right)$, since the user site in the center. It can be observed from FIGURE 4 that users are almost uniformly distributed globally.

For each user, there is an effective elevation range, $\theta_{beam} \in [\theta_0, \pi/2]$, forming a visibility cone with the user as the apex. Therefore, by grasping all the coordinates of both satellites and users at instant t, we can obtain the visible satellite set for user $v_U \in V_U$ as $V_{Visible}(v_U) = \{v_1, \cdots, v_{N_V}\}$, $V_{Visable}(v_U) \subset V_{Sat}$. Because each user is on behalf of all satellite terminals in the grid, the user's beam number is unlimited, and each beam can be connected to a satellite. Assuming that the line-of-sight (LoS) channel is dominated between the satellite and the ground, the channel model h_{v_S,v_U} from the user node v_U to the visible satellite v_S can be expressed as

$$h_{v_S,v_U} = \sqrt{\rho_{v_S,v_U} \cdot d_{v_S,v_U}^{-2}}, v_S \in V_{Visable}, \; v_U \in V_U, \quad (5)$$

where d_{v_S,v_U} is the distance between v_S and v_U, and ρ_{v_S,v_U} denotes the channel gain at $d_{v_S,v_U} = 1$ m. Because the same antenna is used for transmission and reception, $\rho_{v_S,v_U} = \rho_{v_U,v_S}$ and $h_{v_S,v_U} = h_{v_U,v_S}$. Thus far, the bidirectional channel capacities between the satellite and user nodes are

$$C_{v_S,v_U} = B \cdot \log_2 \left(1 + \left|h_{v_S,v_U}\right|^2 P_S / \sigma^2\right), \quad (6)$$

$$C_{v_U,v_S} = B \cdot \log_2 \left(1 + \left|h_{v_U,v_S}\right|^2 P_U / \sigma^2\right), \quad (7)$$

where B denotes the channel bandwidth, P_S and P_U are the transmit power for the satellite and user antenna, respectively, σ^2 is the noise power. Then, we fill in the capacities into the connection relationship matrices $\boldsymbol{A}_{US}$ and $\boldsymbol{A}_{UD}$. For $\forall v_S \in V_{Visible}(v_U)$ and $\forall v_U \in V_{US}$, $a_{v_S,v_U}^{US} \in \mathbf{A}_{US}$ and $a_{v_U,v_S}^{UD} \in \mathbf{A}_{UD}$ can be expressed as

$$a_{v_S,v_U}^{US} = \begin{cases} C_{v_S,v_U}, & \exists (v_S, v_U) \\ 0, & otherwise, \end{cases} \quad (8)$$

$$a_{v_U,v_S}^{UD} = \begin{cases} C_{v_U,v_S}, & \exists (v_U, v_S) \\ 0, & otherwise. \end{cases} \quad (9)$$

F. TASK WEIGHT DISTRIBUTION MODEL

The task weight model cooperates with the user-access model. As mentioned earlier, the task weight from user i to user j is the ratio of the amount of traffic from user i to user j to the sum of all the traffic between any two user nodes. In the capacity analysis, we assumed that the network tasks always increase in proportion to the fixed task weight distribution; that is, the sending node of each ground user generates tasks to the receiving nodes of other ground users according to a certain proportion. Because the satellite itself does not generate tasks, all tasks originate from the sending node set V_{US}, and eventually flow to the receiving node set V_{UD}. As a result, the proportions of generated tasks are summarized to form a task weight matrix, $\boldsymbol{B} \in N_{US} \times N_{UD}$:

$$\boldsymbol{B} = \begin{bmatrix} \beta_{1,1} & \cdots & \beta_{1,N_{UD}} \\ \vdots & \ddots & \vdots \\ \beta_{N_{US},1} & \cdots & \beta_{N_{US},N_{UD}} \end{bmatrix}, \quad (10)$$

where $\beta_{i,j} \in \boldsymbol{B}$ represents the proportion of traffic sent by the ground node $v_i \in V_{US}$ to the ground node $v_j \in V_{UD}$,

$\forall i \in [1, \cdots, N_{US}]$, and $\forall j \in [1, \cdots, N_{UD}]$. The i-th row represents all tasks sent by v_i, and the j-th column represents all tasks received by v_j, with

$$\sum_i \sum_j \beta_{i,j} = 1. \tag{11}$$

Because the capacity of the low-orbit constellation network is closely related to the distribution of ground tasks, it is necessary to make assumptions regarding the distribution of ground tasks according to different application scenarios. However, the actual task distribution model is not yet clear; therefore, we create four types of hypothetical distribution models to explore the impact of different distributions on network capacity: two-point distribution, global convergence distribution, global population density distribution, and global uniform distribution.

a) Point-to-point distribution: Traffic flow from one user to another. The element $\beta_{i,j} \in \boldsymbol{B}$ of the task weight matrix can be expressed as:

$$\beta_{i,j} = \begin{cases} 1, & i = i_{send},\ j = j_{recv},\ i \neq j \\ 0, & otherwise. \end{cases} \tag{12}$$

b) Global convergence distribution: All user nodes deliver the same amount of traffic to one node in backhauling scenarios. The element $\beta_{i,j} \in \boldsymbol{B}$ of the task weight matrix can be expressed as:

$$\beta_{i,j} = \begin{cases} \dfrac{1}{N_{US} - 1}, & \forall i,\ j = j_{recv},\ i \neq j \\ 0, & otherwise. \end{cases} \tag{13}$$

c) World population distribution: This task distribution is associated with global population density data [17]. By dividing the global population data into grids in the same manner as in Section 2.1D, and by adding up the population data in the grids, the total population Q_i in the grid i can be obtained, where $i \in [1, N_U]$. The element $\beta_{i,j} \in \boldsymbol{B}$ of the task weight matrix can be expressed as:

$$\beta_{i,j} = \begin{cases} \dfrac{Q_i \cdot Q_j}{\sum\limits_i \sum\limits_j Q_i \cdot Q_j}, & i \neq j \\ 0, & i = j. \end{cases} \tag{14}$$

d) Global uniform distribution: The aforementioned ‘‘all-to-all’’ model, in which every node in the world generates the same amount of traffic to every other node, where the element $\beta_{i,j} \in \boldsymbol{B}$ of the task weight matrix is represented by:

$$\beta_{i,j} = \begin{cases} \dfrac{1}{N_{US} N_{UD} - \min\left(N_{US}, N_{UD}\right)}, & i \neq j \\ 0, & i = j. \end{cases} \tag{15}$$

III. NETWORK CAPACITY ESTIMATION

This section discusses the calculation method for low-complexity constellation capacity considering queuing. The algorithm flow is divided into two parts: MCF capacity measurement and Jackson-network-based capacity rollback. The former replaces the high-complexity LP algorithm and proposes an improved fully polynomial-time approximation (IFPTA) algorithm to estimate network capacity from the perspective of the flow model. Based on the capacity results obtained in the first part, the latter considers network queuing and packet loss, uses the nature of the Jackson network to analyze the relationship between network capacity and packet loss rate, and provides the packet loss on the target edge. Finally, a calculation method for the capacity roll-back index under the target packet loss rate threshold was presented.

A. PROBLEM FORMULATION

For LEO mega-constellations, the capacity estimation problem with link packet-loss rate constraints can be described by the MCF problem. Because the task weight distribution is fixed, it can be further summarized as the maximum concurrent flow problem. The linear programming form of this problem is

$$\underset{f^k_{(v,w)}}{\text{maximize}}\ \lambda \tag{16}$$

$$\begin{aligned} s.t.\ & \sum\nolimits_{w \in V} f^k_{(v,w)} - \sum\nolimits_{w \in V} f^k_{(w,v)} \\ & = \begin{cases} 0, & v \notin V_{US},\ v \notin V_{UD} \\ 0, & v \in V_{UD},\ v \neq v_k \in V_{UD} \\ \lambda \beta_{v,v_k}, & v \in V_{US},\ v_k \in V_{UD} \\ -\lambda \beta_k, & v = v_k \in V_{UD}, \end{cases} \\ & \quad \forall v \in V, \quad \forall k \in \{1, \cdots, N_{US}\} \end{aligned} \tag{17}$$

$$\beta_k = \sum\nolimits_{i=1}^{N_{US}} \beta_{i,k}, \quad \forall k \in \{1, \cdots, N_{UD}\} \tag{18}$$

$$\sum\nolimits_{k=1}^{N_{UD}} f^k_{(v,w)} \leq a_{v,w}, \quad \forall v,\ w \in V,\ a_{v,w} \in \boldsymbol{A} \tag{19}$$

$$f^k_{(v,w)} \geq 0, \quad \forall k \in \{1, \cdots, N_{UD}\} \tag{20}$$

$$L_{(v,w)} < \eta \tag{21}$$

where $k \in [1, \cdots, N_{US} N_{UD}]$ is the commodity of traffic, representing the traffic between a source-destination pair (S-D pair), $f^k_{(v,w)}$ is the traffic of commodity k routed in edge (v, w), $L_{(v,w)}$ is the packet loss rate of edge (v, w), and η is the link packet loss rate threshold. The idea is simple: for each commodity, the traffic is conservative for each node, that is, its traffic income should equal its outcome. λ denotes the factor of the task weight matrix, $\boldsymbol{B}$. It is worth noting that because the sum of the elements in $\boldsymbol{B}$ equals 1, the physical meaning of λ is the total throughput of the entire network. Therefore, the objective of this MCF problem is to solve for the maximum value of λ.

However, constraint (21) cannot be simply determined because the link packet loss rate is statistical. Therefore, we propose a two-step approach. First, without considering the link packet loss rate, the remaining problem is solvable

MCF. Second, we utilize a Jackson-network-based capacity fallback approach to control the link packet loss rate below a given threshold, without loss of optimality. In the first step, although the linear programming algorithm can solve the MCF problem, it faces the operational time problem. The complexity of the linear programming algorithm is $O(N_{var}^{3.5})$, where N_{var} denotes the number of variables. Consequently, an enormous amount of time is required for LP calculation when faced with a large number of nodes in a mega-constellation network. In practice, the running time of a single snapshot is often calculated in units of hours or days. Consequently, LP is incapable of capacity analysis for mega-constellation configurations, particularly when multi-snapshots and multi-user distributions are mentioned. In the following, we give a less complex definition of the notion ''commodity'' based on which the low complexity IFPTA algorithm is proposed to substitute LP.

B. LESS COMPLEX DEFINITION OF COMMODITY

As mentioned earlier, a commodity is all traffic between an S-D pair; therefore, the number of commodities in our scenario is $N_{US}N_{UD}$, which is extremely large when the number of ground users is large. Therefore, we seek a low-complexity method for commodity definitions. To achieve this goal, we must prove the following lemma:

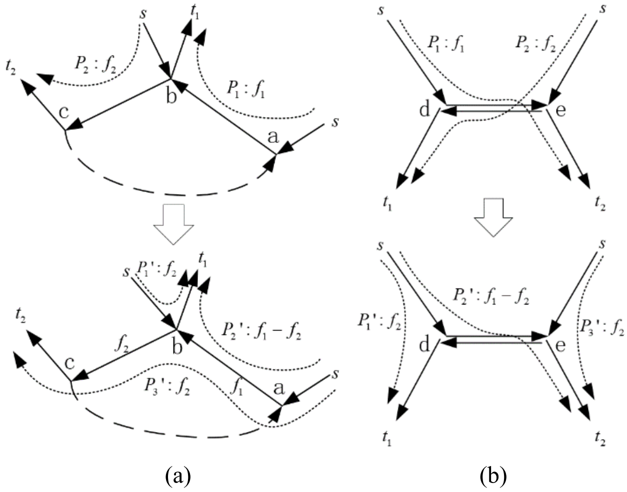
Lemma 1: In a directed graph, all the traffic originating from the same node constitutes a directed acyclic graph when the network reaches its maximum capacity.

Proof: We prove this by a contradiction. A network reaching its maximum capacity means that no traffic distribution can be found to increase the total traffic. In other words, there is no traffic distribution that can reduce traffic on any edge while maintaining the same capacity. Assume that a loop is formed by traffic flows from the same source when the network reaches its maximum capacity, and the loop is composed of multiple paths. We perform following steps.

Step 1: Arbitrarily take two adjacent paths $P_1(s \to a \to b \to t_1)$ and $P_2\,(s \to b \to c \to t_2)$ of different commodities in the loop with traffic f_1 and f_2, respectively, as shown in the upper part of FIGURE 5 (a). Then, we redistribute the traffic such that the two edges in the loop become one edge, which can be discussed separately from the following two situations. When $f_1 \geq f_2$, f_1 in commodity P_1 can be split into $P'_1\,(s \to b \to t_1) : f_2$ and $P'_2\,(s \to a \to b \to t_1) : f_1 - f_2$. Meanwhile, f_2 in commodity P_2 can be reassigned to $P'_3\,(s \to a \to b \to c \to t_2) : f_2$, as shown in the lower part of FIGURE 5(a). Similarly, when $f_1 \leq f_2$, reassign f_1 in commodity P_2 as $P'_1\,(s \to b \to t_1) : f_1$ and split P_2 into $P'_2\,(s \to a \to b \to c \to t_2) : f_1$ and $P'_3\,(s \to b \to c \to t_2) : f_2 - f_1$. In either case, the total traffic of each edge in the graph remains unchanged, and the two edges in the loop become one.

Step 2: Repeat **Step 1** until only two edges remain in the loop, as shown in the upper part of Fig. 5(b).

Step 3: Redistribute the traffic flows in the upper part of FIGURE 5(b). Assume that $f_1 \geq f_2$, split f_1 in P_1 into $P'_1\,(s \to e \to t_1) : f_2$ and $P'_2\,(s \to d \to e \to t_1) : f_1 - f_2$. Reassign f_2 in P_2 to $P'_3\,(s \to d \to t_2) : f_2$, as shown in the lower part of FIGURE 5(b).



(a) (b)

FIGURE 5. Traffic flows merging diagram for Lemma 1. (a) Step 1. (b) Step 2 & Step 3.

Thus far, after redistribution, the traffic in edge $(d \to e)$ is reduced to $f_1 - f_2$, and that in edge $(e \to d)$ becomes 0, which contradicts the assumption. Therefore, there is no loop in the traffic from the same source, that is, all flows from the same node form a directed acyclic graph. □

With the support of **Lemma 1**, we can redefine commodity to reduce its number, as in Definition 2.

Definition 2: A commodity of traffic flows in a constellation network refers to all the traffic originating from the same source node. Traffic flows between commodities are not exchangeable.

The consistency of the new definition can be proved by the following theorem.

Theorem 1: In the MCF problem, combining all commodities from the same source into one does not change the capacity result of the MCF problem.

Proof: Assuming that the original problem has reached the optimum, we merge all commodity traffic flows with a shared source. It can be seen from Lemma 1 that when the system reaches its maximum capacity, all flows from the same source constitute a directed acyclic graph; therefore, the traffic on each edge will not decrease after merging. Thus, the solution to the new problem is not less than that of the original problem.

Assuming that the optimal value of the merged commodity is greater than that of the original problem, the merged commodity traffic can be split into subtraffic flows of S-D pairs. Then, there must be a subtraffic flow of the new problem that is greater than the corresponding flow of the original problem, which means that the result of the original problem is not optimal, contradicting the assumptions. Therefore, the solution to the new problem is not greater than that of the original problem.

In summary, the solution to the new problem is equivalent to the original problem. □

Therefore, it is proven that the traffic originating from the same source can be defined as the same commodity. This definition sharply reduced the number of commodities from $N_{US}N_{UD}$ to N_{US}.

C. IMPROVED FULLY-POLYNOMIAL-TIME APPROXIMA-TION ALGORITHM

We propose the IFPTA algorithm to replace LP, the workflow of which is shown in **Algorithm 2**. The oriented edge from node v to w is represented by (v, w), where capacity is denoted by $a_{v,w}$. $f_{(v,w)}^{s,d} \in \boldsymbol{F}$ is the traffic component of the S-D pair s and d within edge (v, w), satisfying $\sum_s \sum_d f_{(v,w)}^{s,d} \leq a_{v,w}$. $\boldsymbol{F} \in N \times N \times N_R$ is the MCF tensor that records the traffic distribution of each commodity at each edge. Matrix $\boldsymbol{B}_{Zoom} = \sigma \boldsymbol{B}$ is the scaling of task weight matrix $\boldsymbol{B}$, $\sigma \in \mathbb{R}^+$, to control the number of iterations. $\delta \in \mathbb{R}^+$ is a small positive constant used to initialize the weight of each edge.

The main idea of the algorithm is to set the weights for each edge in the network. According to **Definition 2**, the Dijkstra algorithm can be used to calculate the smallest weight sum path for a commodity in the one-time calculation instead of the N_{UD} times. Traffic flows are routed through those paths subject to $\boldsymbol{B}$. Those traffic flows are accumulated on the path, increasing the involved edge's weight according to the ratio of traffic to edge capacity. Finally, the traffic distribution of the MCF problem is determined through adequate repetitions. The accuracy of the algorithm is controlled by the iteration step $\varepsilon \in (0, 1)$. The smaller the ε, the higher is the accuracy of the algorithm; in other words, the closer it is to the optimal value. The algorithm possesses four levels of nested loop. The objective of the first-level loops is to balance traffic through multiple iterations. The break condition is set as $\Omega \geq 1$ to determine the maximum iteration number by setting the upper bound for the non-descending variable Ω. All source nodes are traversed by the second-level loop, in which the third-level loop is responsible for ensuring that all tasks in $\boldsymbol{B}'_{Zoom}$ are allocated. The shortest path from the current source to all destinations is also calculated in this loop using the Dijkstra algorithm. In addition, in the third-level loop, the weight of edges also increases according to the amount of added traffic, influencing the path choice of the next loop. Finally, the fourth-level loop traverses all the destination nodes to route tasks according to the current shortest path.

Algorithm 2 Improved Fully-Polynomial-Time Approximation (IFPTA)

```
Input: adjacency matrix $A$, task weight matrix $B$,
iteration step $\varepsilon$, scaling factor $\sigma$ of task weight matrix,
weight initialization constant $\delta$
Output: maximum network capacity $\lambda_{\max}$,
multi-commodity traffic tensor $f_{(v,w)}^{s,d} \in F$
Initialize: edge weight $l_{(v,w)} \Leftarrow \delta / a_{v,w}, f_{(v,w)}^{k} \Leftarrow 0$,
$\forall a_{v,w} \in A, B_{Zoom} = \sigma B, \Omega = 0$
Begin
while $\Omega < 1$ do
    $B'_{Zoom} \Leftarrow B_{Zoom}$
    for each $s = 1 : N_{US}$ do
        while $\sum_i B'_{Zoom}(s, i) \neq 0$ do
            $[P(:, d), D_s] \Leftarrow$ dijkstra($A, s$), calculate the
            minimum weight path $P(s, :)$ and the weight
            $D_s$ from node $s$ to the rest nodes
            $A^* \Leftarrow A, a_{v,w}^* \in A^*$
            for each $d = 1 : N_{UD}$ do
                $C_b \Leftarrow$ bottleneck capacity of $P(s, d)$ in
                $A^*$
                if $B'_{Zoom}(s, d) \leq C_b$ do
                    $f_{(v,w)}^{s,d} \Leftarrow f_{(v,w)}^{s,d} + B'_{Zoom}(s, d)$,
                                        $\forall (v, w) \in P(s, d)$
                    $a_{v,w}^* \Leftarrow a_{v,w}^* - B'_{Zoom}(s, d)$,
                                        $\forall (v, w) \in P(s, d)$
                    $f_d^{temp} \Leftarrow B'_{Zoom}(s, d)$
                    $B'_{Zoom}(s, d) \Leftarrow 0$
                else do
                    $f_{(v,w)}^{s,d} \Leftarrow f_{(v,w)}^{s,d} + C_b, \forall (v, w) \in$
                    $P(s, d)$
                    $f_d^{temp} \Leftarrow C_b$
                    $B'_{Zoom}(s, d) \Leftarrow B'_{Zoom}(s, d) - C_b$
                End if
            end for
            $\Omega \Leftarrow \Omega + \varepsilon \sum_d \left( f_d^{temp} \cdot D_s(d) \right)$
            $l_{(v,w)} \Leftarrow l_{(v,w)} \left( 1 + \varepsilon \cdot \frac{\sum_d f_d^{temp}}{C_l} \right), \forall v, w \in V$
        end while
    end for
end while
$f_{(v,w)}^{s,d} \Leftarrow \frac{f_{(v,w)}^{s,d}}{\max\limits_{s,d} \left( f_{(v,w)}^{s,d} / a_{v,w} \right)}, \forall a_{v,w} \in A, \forall s, d, s \neq d$
$\lambda \Leftarrow \sum_s \sum_d \sum_v f_{(v,d)}^{s,d}, \forall d \in V_{UD}, \forall s \in V_{US}, \forall v \in V$
End
```

Our improvement mainly lies in the generation of the scale factor, which converts infeasible traffic into feasible traffic, so that the algorithm throughput is closer to optimal. In this approximate solution, because the traffic of each edge is likely to exceed its capacity after all iterations are completed, the traffic must be scaled down. The method adopted in [15] is to estimate an upper bound $\log_{1+\varepsilon} a_{v,w} \cdot (1 + \varepsilon) / \delta$ of the traffic that can be allocated on each edge and then reduce the allocated traffic by $\log_{1+\varepsilon} (1 + \varepsilon) / \delta$ so that the flow is less than its capacity. However, because an estimated upper bound is used as the scale factor, the capacity of each side is often not fully utilized in the actual implementation.

Herein, we present a new method for calculating the scaling factor. The traffic allocated to each edge can be measured as a numerical solution. Therefore, the maximum traffic-to-capacity ratio $\max\limits_{v,w} \left(f_{(v,w)}^{s,d} / a_{v,w} \right)$ can be calculated and used as a scaling factor for traffic reduction in $\boldsymbol{F}$. This means that the feasible traffic in $\boldsymbol{F}$ is no less than that obtained using the methods in [15], for at least one edge possesses traffic reaching the upper bound of the edge capacity, without changing the traffic distribution in $\boldsymbol{F}$. In other words, $\lambda_{\max}$ could be closer to the optimal value. According to [15], the algorithm complexity of FPTA is $O(\varepsilon^{-2}(M^2 + N_{UD}N))$, which is much lower than the linear programming algorithm complexity $O((N_{UD}(4N - q))^{3.5})$.

D. CAPACITY FALLBACK BASED ON QUEUEING NETWORK

According to the aforementioned capacity analysis algorithm, the planning-based algorithm can estimate the maximum traffic that the constellation-terrestrial network can bear in a snapshot. However, constraint (21) was not considered. In the IFPTA algorithm, the maximum traffic is reached under the assumption that all traffic flows are regarded continuous, which is unrealistic. In actual networks, information is formed in discrete packets and the arrival interval between packets is random. Consequently, queuing or even packet loss is inevitable when the edge load is high. In our case, it can be observed from the IFPTA algorithm in the previous section that the traffic of at least one edge reaches its capacity. From the perspective of queuing, the edge is severely congested, and a large number of packets are discarded, which means that the network is not usable. To avoid this, a fallback of the estimated traffic is necessary to obtain a reachable estimation of the network capacity by balancing the traffic and packet loss rate of the edge. In this section, we use Jackson network theory to analyze the fallback of network capacity. We define the fallback as

Definition 3: The fallback of the network capacity is the reduction in the maximum traffic flow that can be borne by the network to control the maximum packet loss rate under a given threshold.

Assuming that the packet length in the network follows an exponential distribution with mean L, where $L \ll \min(a_{v,w})$. The packet input of each source node obeys a Poisson stochastic process with an arrival rate of γ. Thus, the service time of each queue is an independent exponential distribution, with mean service rate of edge (v, w):

$$\mu_{(v,w)} = \frac{a_{v,w}}{L} \tag{22}$$

According to Jackson's theorem, the packet transmission on each edge of the network can be regarded as an independent M/M/1 queue if the following conditions are satisfied: 1) packets arriving from outside the network obey the Poisson distribution and 2) service rate obeying the independent exponential distribution, which exactly meets our case. Thus, the probability that the number of packets in the queue (v, w) equals n is

$$p_{(v,w)}(n, \gamma) = \rho_{(v,w)}(\gamma)^n \left(1 - \rho_{(v,w)}(\gamma)\right) \tag{23}$$

where $\rho_{(v,w)} = \gamma/\mu_{(v,w)}$ and $0 \leq \rho_{(v,w)} \leq 1$ show the occupancy rate of the edge. For an edge with a queue length of N_q packets, we define the theoretical packet loss rate of the edge as the ratio of the expectation of the number of packets exceeding the queue length to that of the total number of packets on the edge:

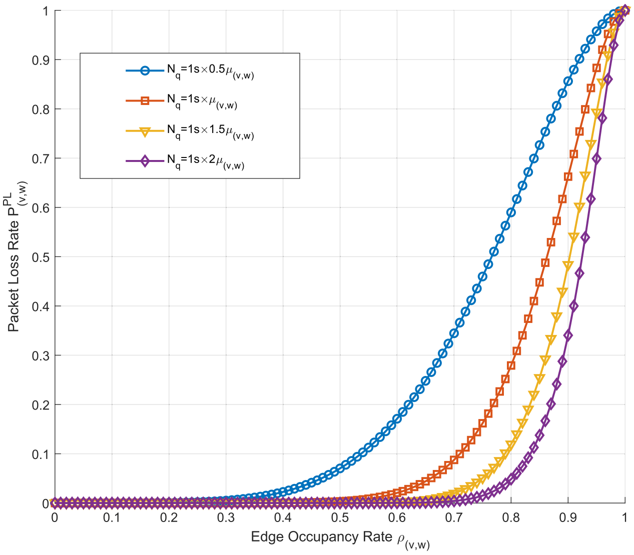


FIGURE 6. Relationship between packet loss rate and edge occupancy rate.

$$\begin{aligned} P^{PL}_{(v,w)}\left(\rho_{(v,w)}, N_q\right) &= \frac{E\left[n > N_q\right]}{E\left[n\right]} \\ &= \frac{\sum_{n=N_q+1}^{\infty} p_{(v,w)}(n) \cdot (n+1)}{\sum_{n=1}^{\infty} p_{(v,w)}(n) \cdot (n+1)} \\ &= \rho^{N_q}_{(v,w)} \left[1+N_q\left(\rho_{(v,w)} - \rho^2_{(v,w)}\right)\right]. \end{aligned} \tag{24}$$

The packet loss rate is a function of the edge occupancy rate $\rho_{(v,w)}$ and queue length N_q, and the influence of the two on the packet loss rate is shown in FIGURE 6. It can be observed that by appropriately rolling back the edge occupancy rate, the packet loss rate of the edge can be significantly alleviated. Meanwhile, longer queues have a good buffering effect such that when the edge reaches the same packet loss rate, the magnitude of the capacity rollback is greatly reduced. However, regardless of the amount of N_q, the packet loss rate always increases to 100% when an edge is fully occupied, which emphasizes the significance of the fallback of capacity estimation.

For a fixed queue length, we set $\eta \in [0, 1]$ as the packet loss rate threshold of the edge. The maximum edge occupancy rate $\hat{\rho}_{(v,w)}$ can be solved by

$$\hat{\rho}^{N_q}_{(v,w)} \left[1 + N_q\left(\hat{\rho}_{(v,w)} - \hat{\rho}^2_{(v,w)}\right)\right] = \eta \tag{25}$$

where $\hat{\rho}_{(v,w)}$ can be solved numerically. Because the tasks are distributed according to a certain proportion, the estimated traffic distribution in $\boldsymbol{F}$ is also fixed, which means that the fallback of one edge is equivalent to that of all traffic flows in the network. However, according to **Algorithm 2**, at least one edge is fully occupied. Therefore, $\hat{\rho}(\eta)$ denotes the global capacity fallback factor, $\hat{\rho}(\eta) = \hat{\rho}_{(v,w)}(\eta), \forall v, w \in V$. The network capacity and MCF tensor after fallback can be expressed as

$$\lambda_\eta = \hat{\rho}(\eta) \cdot \lambda_{\max}, \tag{26}$$

$$\boldsymbol{F}_\eta = \hat{\rho}(\eta) \cdot \boldsymbol{F}. \tag{27}$$

It should be emphasized that the aforementioned process of network capacity estimation is only for one snapshot. This should be executed for all snapshots generated in Section II-B to obtain the capacity variation over time. The entire process of network capacity analysis for LEO mega-constellation is summarized in **Algorithm 3**.

Algorithm 3 Network Capacity Analysis Process for LEO Mega-Constellation

```
Input: constellation ephemeris for $t \in [0, T]$, snapshot interval $\Delta t$, ideal size of global area subdivision $L_X \times L_Y$, link capacity of ISLs $C_{ISL}$, threshold of link packet loss rate $\eta$, link queue length $N_q$
Output: network capacity with link packet loss rate constraints $\lambda_\eta$, MCF tensor $\boldsymbol{F}$
Begin
Sample the constellation ephemeris with $\Delta t$, generate $N_T$ snapshots.
Create approximate equal-area global subdivision following instructions in Algorithm 1, and determine the location of all ground users.
Determine the type of task weight distribution, and calculate task weight for each ground user, forming the task weight matrix $\boldsymbol{B}$.
Solve equation (25) to find the capacity fallback factor $\hat{\rho}(\eta)$.
For each snapshot do
    Generate adjacency matrix $\boldsymbol{A}_{Sat}$ of the satellite constellation at the present snapshot
    Generate $\boldsymbol{A}_{\mathrm{US}}$ and $\boldsymbol{A}_{UD}$, which are the constellation-terrestrial connection relationship matrices for both directions.
    Form network topology adjacency matrix $\boldsymbol{A}$.
    By taking $\boldsymbol{A}$ and $\boldsymbol{B}$ as parameters, calculate the maximum network capacity using the IFPTA algorithm in Algorithm 2, obtaining the maximum network capacity $\lambda_{\max}$ and the multi-commodity traffic tensor $\boldsymbol{F}$.
    Execute network capacity fallback as (26) and (27)
End for
End
```

IV. SIMULATION RESULT AND DISCUSSION

In this section, the simulation results utilizing the proposed algorithm are demonstrated from two perspectives. The first is the algorithm performance analysis, which discusses the ability of network capacity estimation from the metrics of accuracy, complexity, and time sampling interval. The second is an exploration of the capacity characteristics of LEO mega-constellation networks. The proposed algorithm is used to analyze the attributes of the constellation network, including the relationship between the capacity and constellation scale or configuration, capacity bottleneck analysis, and the relationship between service distribution and capacity.

The simulation parameters are as follows: Both Walker-Star and Walker-Delta constellation configurations were adopted in the simulation. To equalize the number of satellites with different constellation configurations, we utilized the method of exchanging the number of orbital planes and satellites per orbit of the two configurations to ensure the rationality of the configuration. For example, for a constellation with a scale of 384 satellites, when the Walker-Star configuration was adopted, there were 16 orbital planes and 24 satellites per orbit, and two parameters shifted when the Walker-Delta configuration was adopted. The orbital inclinations for the two configurations are 90° and 60°, respectively, with the same orbital height $H = 1000\ km$. The snapshot sampling step was 30s, and the simulation time T was one orbit period of the satellite, which is 105 s for $H = 1000\ km$. Link capacity of ISLs were $C_{ISL} = 1\ Gbps$. For global area subdivision, the ideal length along the line of latitude and longitude was $L_X = L_Y = 2000\ km$. The beam number was unconstrained for both satellites and ground users because the ground user was a representation of all users in the grid. For the parameters of the IFPTA algorithm, we used the iteration step $\varepsilon = 0.4$, and the weight initialization constant was $\delta = 10^{-10}$. We assume that the average length of the packets is 1000 Byte. The threshold of the edge packet loss rate was set to $\eta = 0.05$, with a link queue length of $N_q = 1s \times 2\mu_{(v,w)}$.

A. ALGORITHM PERFORMANCE ANALYSIS

In this section, we focus on the algorithm performance from three dimensions: accuracy, time complexity, and sampling interval, which are the basis of constellation network estimation.

The accuracy of the capacity estimation algorithm depends on the optimality of the planning-based algorithm, which is IFPTA in our case, compared with the original FPTA and classical LP. However, although the solution obtained by LP can be considered optimal, its runtime is considerably high for a network that possesses hundreds of nodes and multi-commodity traffic flows. Therefore, for the comparison of accuracy in an acceptable way for all candidates, we use a simplified 2-D torus network with fewer nodes for the network topology model, and the uniform distribution for the task model, where each node sends an identical amount of traffic to all nodes except itself in a 2-D torus network. The edge capacity is assumed to be 1 Gbps. The comparison results are shown in FIGURE 7. The throughputs of the proposed IFPTA and original FPTA are given with different iterative step lengths ε. It can be observed that under the same iterative step length, the improved result is better than before, and it is close to the LP result when ε is small, that is, close

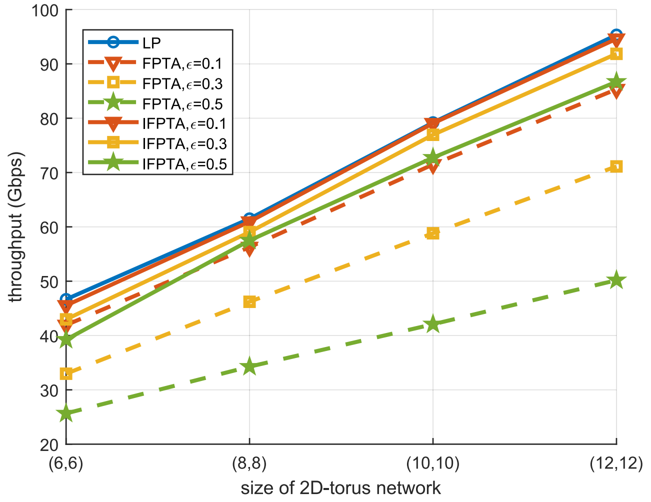


FIGURE 7. Accuracy comparison of LP, FPTA, and proposed IFPTA, under simplified circumstance of 2-D torus network and "all-to-all" uniform distribution.

to the upper bound. At the same time, the IFPTA algorithm is less sensitive to ε, which means that we can still obtain results close to optimal even using a larger ε, which is equivalent to reducing the complexity of the algorithm.

To further explain the reason for using IFPTA instead of LP, the time complexity comparison, with the same simulation circumstance as in the accuracy comparison, is given in FIGURE 8, in which the time complexities of LP and IFPTA with various ε values are compared. We use logarithmic representation of the ordinate. IFPTA has a much lower time complexity than LP, at least $1/10^4$ in the case of $\varepsilon = 0.1$. This advantage can be further expanded with larger ε and larger network scale. Therefore, by comparing both the capacity and time complexity, we can conclude that the IFPTA algorithm saves significant time complexity with a tiny price of accuracy, which is acceptable and controllable.

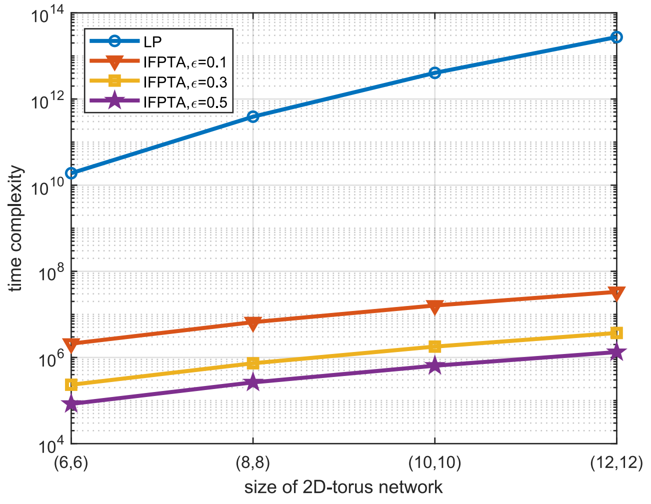


FIGURE 8. Time complexity comparison between LP and IFPTA with different iteration step length.

As for the choice of sampling interval Δt, the time step of this algorithm cannot be infinitely small because of the hardware resources limit. Therefore, an appropriate sampling interval should be selected to increase calculation efficiency. FIGURE 9 shows the results of the capacity change over time for the same constellation and same time period under different sampling intervals Δt. It can be seen that as the granularity of the snapshot becomes finer, the capacity results can more accurately reflect the capacity changes brought about by the changes in the constellation-terrestrial connection relationship.

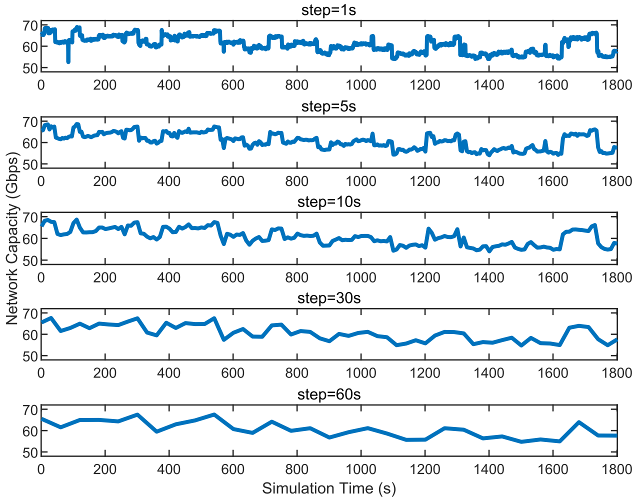


FIGURE 9. Influence on network capacity variation of time sampling.

When the sampling interval is 60s or 30s, it can only reflect the trend of the capacity on a large time scale, but the minute-level step length may lose some extreme conditions, thereby affecting the judgment of the highest or lowest possible capacity. The details are the most abundant when the step size is 1s and 5s, but this will lead to a huge amount of calculation in the estimation in hours or even days. It is worth noting that the larger the constellation scale, the more frequent are the topology changes of the satellite-ground network. Therefore, for a denser constellation scale, it is necessary to reduce the step size appropriately such that the simulation can traverse each topology.

B. RELATIONSHIP BETWEEN CONSTELLATION SIZE AND CAPACITY

The following analysis uses the capacity analysis method proposed in this study as a tool for calculating and analyzing capacity under different constellations and task distributions. First, we consider the capacity performance of the satellite network. For these estimations, we remove the restriction of the satellite-to-ground beam capacity, which is set to be $C_{v_S,v_U} = 4\ Gbps$ as the sum of all ISL capacities for a satellite. For each type of configuration, the size of the constellation is enlarged with a fixed ratio of the number of orbits to the number of satellites per orbit, which is 2:3 and 3:2 for Walker-Star and Walker-Delta, respectively.

FIGURE 10 is a box plot showing the trend of network capacity versus constellation size for the four combinations of constellation configuration, Walker-Star or Walker-Delta, and task distribution, uniform or world population. Each box represents the statistical result of the network capacity over time obtained using **Algorithm 3**. In addition, we use linear functions to fit the mean capacity of the different sizes for each combination. As shown in FIGURE 10, we propose two conjectures.

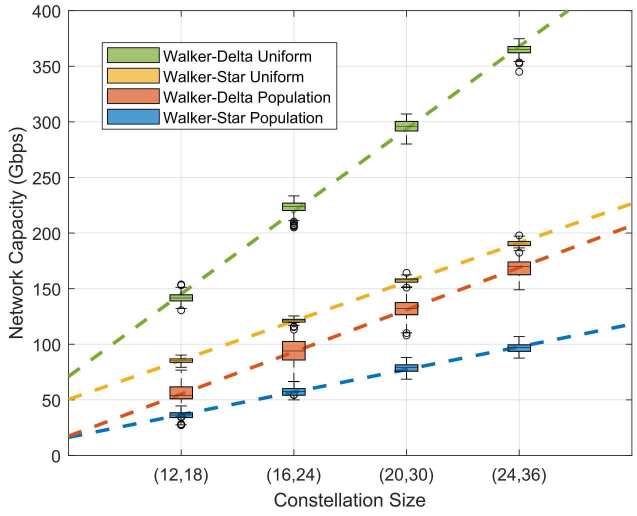


FIGURE 10. Network capacity versus constellation size for different combinations of constellation configuration and task weight distribution.

Conjecture 1: With the same number of satellites, the capacity of the Walker-Star configuration is approximately half that of the Walkers-Delta configuration.

Conjecture 2: The average constellation capacity has an affine relationship with the number of orbits or the number of satellites per orbit.

For these conjectures, we cannot deduce a mathematical proof because of the complexity and irregularity of the constellation-terrestrial network. However, we can at least theoretically explain the correctness of these conjectures in simplified scenarios. The Walker-Delta constellation network is similar to the 2D-Torus network, and becomes a 2D-Torus network when the phase factor is 0. Thus, is the Walker-Star configuration, except that there are reverse seams. When the task distribution is uniform, the network capacities for Walker-Delta and Walker-Star are as follows:

$$C_{Delta} = \frac{8\left(N_O N_{SpO} - 1\right) C_{ISL}}{N_O}, \quad N_{SpO} \leq N_O \tag{28}$$

$$C_{Star} = \frac{4\left(N_O N_{SpO} - 1\right) C_{ISL}}{N_O}, \quad N_{SpO} \geq N_O \tag{29}$$

where N_O is the orbit number in a constellation shell, and N_{SpO} is the number of satellites per orbit. The proof was derived from [5] and [7]. Both (28) and (29) have the same form, except for $C_{Delta} = 2 \times C_{Star}$ under the given circumstance, which is consistent with our simulation results. Moreover, although proportional relationships are derived in (28) and (29) with simplified network conditions, the simulation result in the complete system model yields similar conclusions, which are linear. Therefore, it can be considered that conjectures 1 and 2 are likely correct.

Thus, we can use the above conclusions to obtain a low-complexity empirical estimation method for the network capacity. When the constellation scale is large, the capacity estimation problem of the constellation network is extremely time-consuming. Therefore, we can roughly estimate the capacity of a large-scale constellation by estimating several constellation networks that have the same or similar N_O/N_{SpO}, but with a smaller scale, and then obtain the network capacity of the original large network according to **Conjecture 2**. The implementation of the empirical method is presented in **Algorithm 4.**

Algorithm 4 Empirical Algorithm for Constellation Capacity Estimation

```
Input: target constellation network size ($N_O, N_{SpO}$)
Output: average capacity of the target $C_{empirical}$
Begin
Find at least two smaller constellations possessing the
    same ratio of the number of orbits to the number of
    satellites per orbit as $N_O/N_{SpO}$.
Formulate capacity dimensioning problems for smaller
    constellations. The task distribution and the user
    sub-division remain the same as the original problem.
Solve minor problems formulated in the previous step
    using Algorithm 2, and obtain the mean capacity values.
Use least squares estimation with linear model to fit the
    mean capacity values, get the slope $\hat{\alpha}$ and intercept $\hat{\kappa}$ of
    the line.
Calculate the empirical capacity of the
    mega-constellation network by $C_{empirical} = \hat{\alpha} N_O + \hat{\kappa}$.
End
```

We can perform a simple test using this estimation method. Consider the scenario (40,60) Walker-Star constellation with a uniform task distribution as an example. According to the method above, we can calculate the estimated mean capacity equals to 332.14 Gbps, and the simulation using **Algorithm 2** obtains 331.86 Gbps. The deviation is within 1/1000 in this example.

C. BOTTLENECK ANALYSIS

In this section, we analyze the impact of the ratio of USL capacity to ISL capacity R_{UI} on network capacity to determine the bottleneck of network capacity. Because the capacity of USLs varies with distance, R_{UI} is defined by the ratio of USL capacity at a reference distance $d_{ref} = 1000km$ to ISL capacity as

$$R_{UI} = \frac{C_{v_S,v_U}\left(d_{v_S,v_U} = d_{ref}\right)}{C_{ISL}} \tag{30}$$

Here, we believe that the task is uniformly distributed because when R_{UI} is small, all USLs will be fully occupied indiscriminately, thus showing a uniform distribution. We take both the Walker-Star and Walker-Delta constellations of 384 satellites as examples. The simulation results are shown in FIGURE 11.

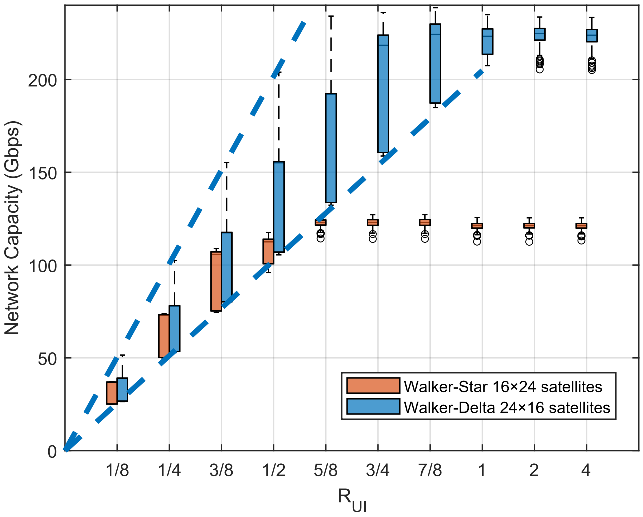


FIGURE 11. **The network capacity versus the USL to ISL ratio.**

It can be seen that for both constellation configurations, the network capacity growth has basically a proportional tendency with R_{UI} increasing when R_{UI} is small. Clearly, USLs are the bottleneck of the network capacity at this stage. However, the capacity growth here cannot be simply represented by its average, since its variation range is also expanded while R_{UI} increases. More precisely, the upper and lower bounds of the capacity grow individually with different slopes until the capacity touches the ceiling, as indicated by the dotted line.

When comparing the two constellation configurations, it is interesting that the capacities of both configurations are very close when $R_{UI} \leq 1/4$, which is in sharp contrast with the capacity comparison when R_{UI} is high. Meanwhile, the Walker-Star constellation reaches its maximum capacity at $R_{UI} = 5/8$, whereas the position for Walker-Delta is $R_{UI} = 1$. These results indicate that the Walker-Delta configuration is quite sensitive to adequate USL capacity; otherwise, its advantages will be reduced. In contrast, the Walker-Star configuration demands much less USL capacity to reach its capacity ceiling, which is easier to satisfy.

D. TASK WEIGHT DISTRIBUTION ANALYSIS

Finally, we discuss the influence of different task weight distributions on network capacity. Based on the same constellation of 384 satellites, four types of task weight distributions are illustrated in FIGURE 12: point-to-point, global convergence, world population distribution, and global uniform distribution, as introduced in Section II-F. We set $C_{v_S,v_U} = 4\ Gbps$ in this part.

The sending and receiving points in the first distribution are chosen as the two furthest ends of the Earth on the equator to avoid the influence of the single-hop relay. It is astonishing that the network capacity in this scenario is incredibly low compared with the potential of the network. This is because the number of accessible satellites for a ground station is limited by the elevation angle and the capacity bottleneck depends on the minimum number of ISLs that can leave or reach the access satellite zones. The capacity of the Walker-Delta configuration in this distribution is slightly higher because ground stations can access both ascending and descending satellites, which means more usable ISLs. Therefore, although the network has many idle ISLs, they are not helpful. This is exactly the same problem as the global convergence distribution, where the bottleneck is at the network zone above the receiver station. The solution to this problem is to use multiple ground stations for access, and the ground stations are geographically dispersed as much as possible to relieve pressure on the local network.

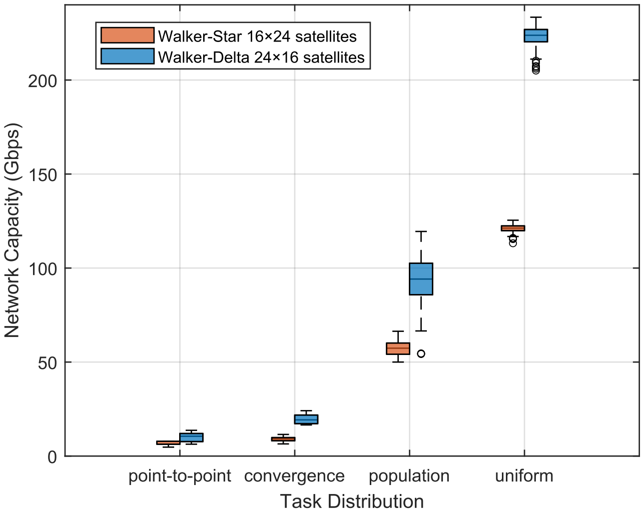


FIGURE 12. **The network capacity versus task weight distributions.**

In contrast, when tasks are widely distributed around the world, the network capacity has better performance, as shown by the global population distribution and global uniform distribution in FIGURE 12. The global uniform distribution has the highest capacity among the four distributions because the tasks exhibit good symmetry, similar to the constellation network. No local jamming occur in the network before all ISLs are evenly occupied. However, the world population distribution is far from uniform, that is, 35% of the population is concentrated in China and India, and almost 0% in ocean areas, which occupy 71% of the Earth's surface. The network capacity in this distribution is less than half that under a uniform distribution.

V. CONCLUSION

The current study analyzes the network capacity of LEO mega-constellations. A time-variant constellation-terrestrial network model was established, and various task weight distribution models were considered to present the characteristics of the network under different use cases. Based on the simpler definition of a commodity, the network capacity determination with the limit of maximum packet-loss rate is fulfilled by the combination of the proposed IFPTA algorithm and the capacity fallback approach. The most obvious finding to emerge from the simulation result is that the network capacity has an affine relationship with the number of orbits or the number of satellites per orbit, which can be used for rough capacity estimation. The results also indicate that the Walker-Delta configuration has nearly double the capacity compared to Walker-Star when the USL bandwidth is adequate, and the gap shrinks otherwise. Finally, it is shown that the network capacity is significantly influenced by task distributions.

REFERENCES

[1] B. Di, L. Song, Y. Li, and H. V. Poor, "Ultra-dense LEO: Integration of satellite access networks into 5G and beyond," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 62–69, Apr. 2019.

[2] G. Curzi, D. Modenini, and P. Tortora, "Large constellations of small satellites: A survey of near future challenges and missions," *Aerospace*, vol. 7, no. 9, p. 133, Sep. 2020.

[3] I. D. 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.

[4] T. S. Wang, P. Lin, and F. Dong, "Progress and prospect of space laser communication technology," *Strategic Study CAE*, vol. 22, no. 3, pp. 92–99, 2020.

[5] L. L. Dai, "Capacity dimensioning and routing for hybrid satellite and terrestrial networks," M.S. thesis, Massachusetts Inst. Technol., Cambridge, MA, USA, 2002.

[6] J. Sun and E. Modiano, "Capacity provisioning and failure recovery for low earth orbit satellite constellation," *Int. J. Satell. Commun. Netw.*, vol. 21, no. 3, pp. 259–284, May 2003.

[7] R. Liu, M. Sheng, K.-S. Lui, X. Wang, D. Zhou, and Y. Wang, "Capacity analysis of two-layered LEO/MEO satellite networks," in *Proc. IEEE 81st Veh. Technol. Conf. (VTC Spring)*, May 2015, pp. 1–5.

[8] M. Werner, F. Wauquiez, J. Frings, and G. Mara, "Capacity dimensioning of ISL networks in broadband LEO satellite systems," in *Proc. 6th Int. Mobile Satell. Conf. (IMSC)*, 1999, pp. 334–341.

[9] M. Werner, J. Frings, F. Wauquiez, and G. Maral, "Topological design, routing and capacity dimensioning for ISL networks in broadband LEO satellite systems," *Int. J. Satell. Commun.*, vol. 19, no. 6, pp. 499–527, Nov. 2001.

[10] Y. Xiao, T. Zhang, D. Shi, and F. Liu, "A LEO satellite network capacity model for topology and routing algorithm analysis," in *Proc. 14th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Jun. 2018, pp. 1431–1436.

[11] D. B. Shmoys, "Cut problems and their application to divide-and-conquer," in *Approximation Algorithms for NP-Hard Problems*. Boston, MA, USA: PWS Publishing Co., 1997, pp. 192–235.

[12] F. Shahrokhi and D. W. Matula, "The maximum concurrent flow problem," *J. ACM*, vol. 37, no. 2, pp. 318–334, Apr. 1990.

[13] T. Leighton, F. Makedon, S. Plotkin, C. Stein, E. Tardos, and S. Tragoudas, "Fast approximation algorithms for multicommodity flow problems," *J. Comput. Syst. Sci.*, vol. 50, no. 2, pp. 228–243, Apr. 1995.

[14] N. Garg and J. Könemann, "Faster and simpler algorithms for multicommodity flow and other fractional packing problems," *SIAM J. Comput.*, vol. 37, no. 2, pp. 630–652, Jun. 2007.

[15] G. Karakostas, "Faster approximation schemes for fractional multicommodity flow problems," *ACM Trans. Algorithms*, vol. 4, no. 1, pp. 1–17, Mar. 2008.

[16] 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.

[17] A. J. Tatem, "WorldPop, open data for spatial demography," *Sci. Data*, vol. 4, no. 1, pp. 1–4, Dec. 2017.

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. He is currently pursuing the Ph.D. degree in aeronautical and astronautical science and technology with the China Academy of Space Technology, Beijing. His research interests include the satellite constellation networking, software-defined networks, routing algorithm for constellation networks, non-orthogonal multiple access communications, and UAV-based satellite terminal technologies.

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.

ZHAOTAO QIN was born in 1989. He received the Ph.D. degree in electronic and information engineering from Beihang University (BUAA), Beijing, China, in 2019. He is currently working at China Academy of Space Technology, Beijing. His current research interests include satellite communication and LEO radio access networks-based 5G protocol.

BINGYUAN LIANG received the B.S. and M.S. degrees in electronic engineering from the Beijing Institute of Technology, Beijing, China, in 2012 and 2014, respectively, and the Ph.D. degree in electromagnetics and microwave technology from Beihang University, Beijing, in 2019. She is currently working with the China Academy of Space Technology. Her research interests include the satellite communication systems and the satellite constellation networking.

DONG CHEN received the B.S. degree in communication engineering, in 2004, and the Ph.D. degree in communication and information system from Xidian University, Xi'an, China. He is currently working with the Institute of Telecommunication and Navigation Satellites, CAST. His research interests include wireless communication, mobile communication, and satellite communication systems.