Optimal Gateway Placement for Minimizing Intersatellite Link Usage in LEO Megaconstellation Networks

Quan Chen, Lei Yang, Jianming Guo, Xianfeng Liu, and Xiaoqian Chen

***Abstract*—Megaconstellation networks, represented by Starlink and OneWeb, have become a promising solution for the wide-area Internet of Things (IoT). IoT messages collected by the satellite can be routed to the ground gateway via multiple intersatellite link (ISL) relays and then access the ground network. Compared to traditional constellations, megaconstellations with massive satellites require more ISL relays that are greatly affected by the number and location of ground gateways. In this article, we focus on the ISL usage for connecting satellites and gateways and propose a novel method with low computation cost to evaluate the ISL usage metric. Then, we formulate a mixed-integer optimization model for the gateway site optimization (GSO) problem with minimizing the overall ISL usage, which is then simplified through model transformation. An IBD-PSO algorithm is proposed to solve the transformed GSO problem. Based on the Starlink constellation, simulation results have verified the proposed method by comparisons with previous studies. We further investigate how the gateway placement is affected by the gateway number and traffic demand pattern. The relations between gateway placement and ISL hop count of different satellites are also studied.**

***Index Terms*—Gateway placement, hop count, intersatellite relay, megaconstellation networks (MCNs), Starlink.**

I. Introduction

RECENTLY, the megaconstellation networks (MCNs), consisting of thousands of low-Earth orbit (LEO) satellites, have become an important solution to supplement the ground networks [1] and also a promising solution for wide-area Internet of Things (IoT) applications both in space [2] and on the ground [3]. The LEO satellite constellations can provide low-latency and global-coverage network services for IoT applications [4]. But since the coverage of a single satellite is vast, the channel resources are shared by massive devices, which requires a lot of onboard capacity [5]. With hundreds and thousands of broadband satellites, MCNs can provide larger bandwidth and serve more IoT devices than traditional LEO constellations [6]–[8].

In the MCN-assisted integrated space–ground network, the space segment can act as the access network to serve the ground devices and forward the IoT messages to the ground network [5]. The ground gateway plays the role of a bridge that connects the space and ground segments [9], [10]. Here, the ground gateway refers to the ground station that communicates with satellites and collects the IoT message gathered by the satellite rather than the IoT gateway [11].

Some MCN projects, such as Starlink [12], plan to equip the satellites with onboard processing (OBP) capability and intersatellite links (ISLs) to enhance the network connectivity [13]. When satellites are out of the gateway coverage, they can still access the ground core network through ISL relays. In this article, each ISL relay is also called one hop. The implementation of ISLs effectively reduces the dependence on ground gateways, while adding more gateways can reduce the required ISL hops to reach gateways [14]. There is a tradeoff between the gateway number and ISL usage [13].

In addition to the gateway number, the gateway locations also affect the ISL usage. Satellites may require different ISL hop counts and bandwidths to reach gateways at different locations. Given a limited number of gateways, the gateway locations can be globally optimized to minimize the ISL usage.

In MCNs, since the satellites are densely distributed, the required ISL hop count is significantly higher than traditional constellations. The high user demands also require more ISL resources. Moreover, although ISLs can adopt laser or radio links [15], the ISL bandwidth is relatively smaller than the satellite–ground links. The limited onboard energy and high requirements of the antenna pointing system constrain the bandwidth of ISLs. Thus, the ISL usage is expected to be minimal. In this article, we aim to minimize the overall ISL usage in MCN by optimizing the gateway locations when the gateway number is fixed.

To minimize the overall ISL usage, this article formulates a gateway site optimization (GSO) problem to find the optimal sites from the given candidate set. To lower the computation cost, the ISL usage metric is evaluated by a novel analytical model that avoids complex routing simulations. The GSO problem is decoupled and then solved by an improved binary

Manuscript received 22 June 2021; revised 27 February 2022 and 8 May 2022; accepted 6 June 2022. Date of publication 13 June 2022; date of current version 7 November 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 11725211 and Grant 12002383, and in part by the Research Project of National University of Defense Technology under Grant ZK22-02. *(Corresponding author: Lei Yang.)*

Quan Chen and Lei Yang are with the College of Aerospace Science and Engineering, National University of Defense Technology, Changsha 410073, China (e-mail: chenquan11@foxmail.com; craftyang@163.com).

Jianming Guo is with the Science and Technology on Complex Aircraft System Simulation Laboratory, Beijing 100094, China.

Xianfeng Liu is with the Beijing Institute of Tracking and Telecommunications Technology, Beijing 100095, China.

Xiaoqian Chen is with the National Innovation Institute of Defense Technology, Chinese Academy of Military Science, Beijing 100071, China (e-mail: chenxiaoqian@nudt.edu.cn).

Digital Object Identifier 10.1109/JIOT.2022.3182412

discrete particle swarm optimization (IBD-PSO) algorithm. Then, we thoroughly study the relations between gateway placement and ISL usage. The considered factors include gateway number, the difference between ascending and descending satellites (SatD), ISL configuration, and ground traffic pattern. The main contributions are summarized as follows.

1) A novel ISL usage evaluation method is proposed: a) the ISL usage metric is formulated as a traffic-weighed hop count and b) an analytical hop-count model is proposed to calculate the required ISL hop count, which has lower computational complexity than traditional routing calculation methods.
2) Aiming at minimizing the overall ISL usage, we establish a mixed-integer optimization model to solve the GSO problem. After gateway assignment decoupling, the GSO problem is simplified and converted to an integer nonlinear programming problem. An IBD-PSO algorithm with three dedicated schemes is proposed to solve the GSO problem.
3) Simulations verify the proposed ISL usage evaluation method and show it saves over 80% computation time than the traditional method. The results also verify the proposed IBD-PSO algorithm.
4) Based on Starlink, we further investigate the effects of gateway number and traffic demand pattern. The hop-count difference between ascending and SatDs is found and analyzed.

In the remaining sections, we first review the related works in Section II and then introduce the integrated space–ground network scenario in Section III. The ISL usage evaluation method is proposed in Section IV. In Section V, the basic GSO problem is formulated and further simplified through problem transformation. Then, Section VI proposes an IBD-PSO algorithm to solve the GSO problem. In Section VII, simulations verify the proposed model and algorithm. Finally, Section VIII concludes this whole article.

II. Related Work

In the field of ground wireless networks, there have been many studies on the gateway placement [16]–[19]. Previous studies have investigated the gateway placement or deployment in different scenarios, such as the wireless mesh network (WMN) [16], ground IoT [17], [18], and 5G ultradense networks [19]. The general ideas are similar, i.e., to formulate a combination optimization or integer programming model to select the optimal gateway placement from the candidate gateway set. However, the above studies cannot directly apply to the satellite constellation network because of its unique features, e.g., satellite dynamics, global-scale scenario, and dynamic but predictable hopping distance between nodes and gateways [20].

With the growing concerns on MCNs, some researchers have started to study the gateway placement in LEO constellation networks in recent years. Guo *et al.* [21] formulated a combination optimization model to optimize the GSO problem in a traditional Iridium-like constellation scenario and propose a heuristic algorithm to solve it. The gravity model is adopted to estimate the traffic aggregated at the gateways, which avoids the traffic routing calculation. But the intermediate ISL transmission is not considered, and the gravity model cannot estimate the ISL usage. Zhu *et al.* [22] studied the optimization of ground gateway deployment to minimize ISL usage and maximize the satellite–ground link capacity for gateways in China. But the gateway placement is limited to China and both the satellite number and gateway number are relatively smaller. Also, the objective evaluation adopts the traditional network simulation method, which costs too much computation to calculate the packet route.

Compared to traditional LEO constellations, gateway placement in MCNs has different features, Vasavada *et al.* [23] gave an empirical gateway placement for minimizing ISL usage in MCNs. The results show that the ISL hop count is rather high in MCN and is closely affected by the gateway placement. But the gateway placement is not optimized or investigated in this article. Kopacz *et al.* [24] gave a simple design of the ground station placement for connecting satellites in an MCN, but the optimization model is not established, and ISLs are not considered. Chen *et al.* [10] obtained a coarse gateway placement result based on geographical grid division using the genetic algorithm (GA). But the optimization relies on complex network simulation and requires huge computations when calculating routes for packets.

Based on the above literature review, this article aims to propose an ISL usage metric evaluation method with lower computational cost, and solve the gateway placement problem by minimizing the ISL usage in an MCN scenario. Based on the optimization results, relations between gateway placement and intersatellite transmission in the MCN will be further investigated.

III. System Model

A. Satellite Constellation and ISLs

Since Walker-δ type constellations are widely adopted in the emerging MCN projects, e.g., Starlink [22], [25], we consider a Walker-δ constellation, which can be represented by a formal notation α: $N_S/N_P/F$, where α is the orbit inclination, N_S is the total satellite number, N_P is the number of orbit plane, and F is the phasing factor, $F \in \{1, \ldots, N_P - 1\}$. N_P orbit planes are evenly distributed along the equator, and the right ascension of ascending node (RAAN) difference between adjacent orbits is $\Delta\Omega = 2\pi/N_P$. In each orbit plane, M_P satellites are also evenly spaced, where $M_P = N_S/N_P$. The satellite location within the plane is specified by the satellite phase angle u ($u \in [-\pi, \pi]$). The phase angle differences of adjacent satellites within the plane and between adjacent planes are $\Delta\Phi = 2\pi/M_P$ and $\Delta f = 2\pi F/N_S$, respectively.

Based on the satellite flying direction and phase angle, satellites in the constellation are classified into two types: 1) ascending satellite (SatA) ($u \in [-\pi/2, \pi/2]$) and 2) SatD ($u \in [-\pi, -\pi/2) \cup (\pi/2, \pi]$). Assume $\alpha \in (0, \pi/2)$, then ascending and SatDs fly toward the northeast and southeast, respectively (see Fig. 1).

Fig. 1 also illustrates the classical ISL connecting mode [6], [26]. Each satellite establishes four ISLs: two intraplane ISLs

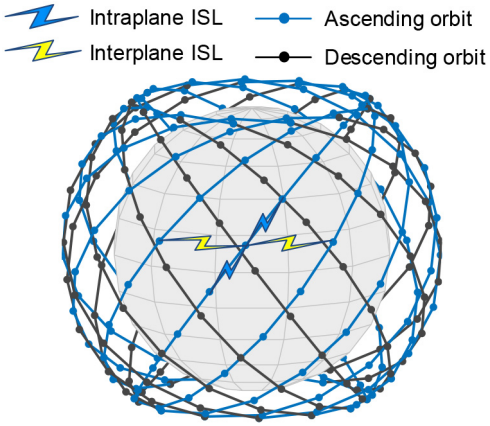

Fig. 1. ISL connecting mode and satellite flying directions in a satellite constellation. For brevity, only the ISLs of one representative satellite are shown.

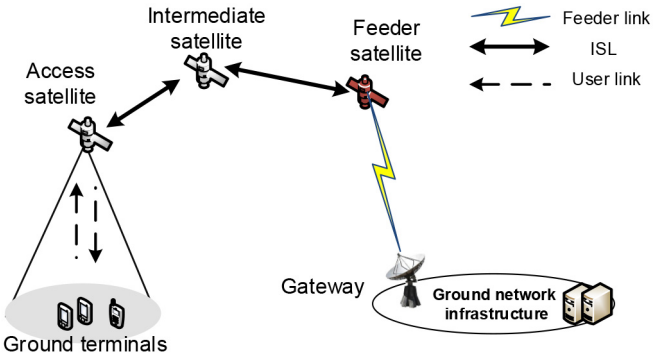

Fig. 2. Multihop path to reach a gateway using ISLs. The satellite acts as the access point, and the users' traffic is routed to the ground network via the gateway.

with adjacent satellites in the same plane and two interplane ISLs with satellites in adjacent orbits. For an inclined constellation, these ISLs can be assumed constantly maintained during the satellite movement [26]. This ISL connecting pattern forms a multihop Mesh-like network topology.

B. Integrated Space–Ground Network Architecture

In an integrated space–ground network [6], the space segment acts as the extended access network for the ground core network, as shown in Fig. 2. Gateway functions as the bridge connecting the space and terrestrial segment. Satellites that directly communicate with the gateway via Sat-GW links are *Feeder Satellites*, and those satellites directly communicating with ground users via Sat-User links are named as *Access Satellites*. When the gateway and ground terminals are covered by the same satellite, intermediate satellites are not required, and access satellite can also be the feeder satellite.

Data packets from ground users are collected by the access satellite and then reach the ground network via the feeder satellite and gateway [6]. If the user and the gateway are not covered by the same satellite (i.e., access and feeder satellites are different), the access to the gateway requires one or more ISL relays. Each ISL relay is defined as one hop. When the access satellite is far from the gateway, it requires many hops to access a gateway. The ISL hops for reaching the gateway are the same as reaching the corresponding feeder satellite.

After they reach the gateway, the transmission of packets follows the same way as in the traditional ground network. Therefore, this article mainly focuses on the transmission between the satellite and gateway rather than an end-to-end connection. The packet transmission is bidirectional, and the returned data follow the reverse path. To simplify the problem, in this article, only the transmission from the user to gateway is studied.

C. Multihop ISL Path

In an MCN with dense satellites, the required hop number is larger than traditional constellation networks. The capacity at each satellite node is occupied by traffic from both its coverage regions and neighboring satellites. Since all the traffic is destined to gateways, the satellites closer to the gateway need to carry more traffic load. In a multihop transmission, each packet is relayed through multiple ISLs, one for each hop, occupying a large number of redundant spectrum resources of ISL [27]. Ideally, the system capacity is proportional to the number of ISLs, while inversely proportional to the average hop count [28]. Therefore, in the case of limited ISL resources, minimizing the hop count can effectively save the ISL usage and increase the capacity utilization efficiency.

To minimize the usage of ISL, packets should select the minimum hop path and select the nearest gateway in terms of hop count as their destination. Although the paths with the minimum hop count might not be unique, we mainly focus on the required minimum hop count but not the path selection. All the hop counts in the following sections refer to the minimum value. The required ISL hop count is determined by the locations of gateway and satellites. The ISL usage depends on the traffic load and the required ISL hops to reach a gateway. In this article, the gateway locations are globally optimized to minimize the overall ISL usage.

IV. ISL Usage Evaluation Model

The proposed ISL usage evaluation method contains two aspects: 1) the ISL usage metric is formulated as a traffic-weighed hop count and 2) an analytical hop-count model is proposed to calculate the required ISL hop count. With the above two schemes, the proposed method can achieve a lower computational complexity than traditional routing calculation methods.

A. ISL Usage Metric

The constellation network with ISLs can be represented by an undirected graph $\mathcal{G}_{\text{sat}} = (\mathcal{S}, \mathcal{E})$, where $\mathcal{S}$ is the satellite node set, $|\mathcal{S}| = N_S$ and $\mathcal{E}$ is the ISL set, $|\mathcal{E}| = N_L$. As the mesh network topology is a 2-D torus, $N_L = 2N_S$ [28]. The feeder satellite set is denoted by $\mathcal{F}$, $\mathcal{F} \subseteq \mathcal{S}$ and $|\mathcal{F}| = N_F$.

This article takes the overall traffic load on ISLs as the ISL usage. The traffic flow in the network is illustrated in Fig. 3. Let l_j be the traffic load of the jth ISL. To minimize the ISL usage, the traffic sum of all ISLs $\sum_{j=1}^{N_L} l_j$ should be minimized. The traffic on the ISLs is generated along with the forwarding

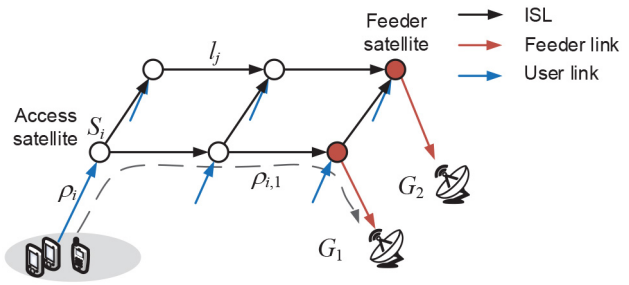

Fig. 3. Traffic flow over the constellation network. Traffic loads concentrate on the feeder satellites through ISL relay and are routed to the gateways.

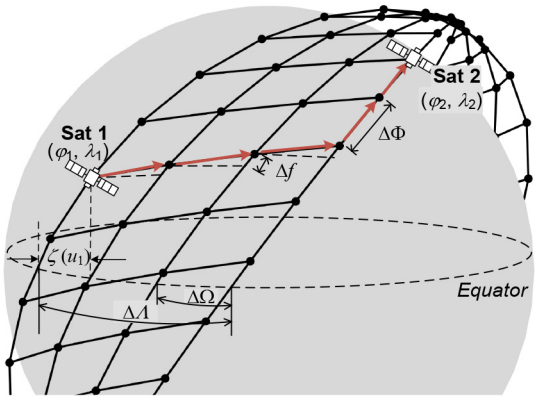

Fig. 4. ISL hop-count calculation between two satellites.

of packets that originate from the ground users' input traffic. Each hop adds the original input traffic to an ISL, and the accumulation of input traffic along the path equals the sum of the traffic of all ISLs (e.g., the black links shown in Fig. 3). Let ρ_i be the user traffic input of satellite S_i from its coverage region. Assume that all packets of ρ_i are routed to the same feeder satellite and let H_i be the required hop count of satellite S_i to reach the feeder satellite, then we have the total ISL usage [29]

$$\sum_{j=1}^{N_L} l_j = \sum_{i=1}^{N_S} \rho_i H_i. \tag{1}$$

Equation (1) indicates that the ISL usage can be evaluated by ρ_i and H_i. The ISL usage metric is converted to a traffic-weighed hop count, which is the linear summation of the satellite's traffic input from its coverage ρ_i multiplying its required hop count H_i. ρ_i is determined by the satellites location and H_i depends on both the satellite locations and gateway placement. Since the packets of ρ_i may have multiple destinations, we further develop (1) and consider that the packets of ρ_i are routed to different feeder satellites and gateways [27], as shown in Fig. 3. Given satellite S_i and gateway candidate set $\mathcal{G} = \{G_1, \ldots, G_M\}$, let $\rho_{i,m}$ be the user traffic originating from S_i's coverage region and routed to G_m, $y_{i,m}$ denotes the ratio of the traffic routed to G_m in ρ_i, i.e., $y_{i,m} = (\rho_{i,m}/\rho_i)$. We have $\sum_{m=1}^{M} \rho_i y_{i,m} = \rho_i$, where M is the number of available gateways. Then, the ISL usage can be calculated by

$$\sum_{j=1}^{N_L} l_j = \sum_{m=1}^{M} \sum_{i=1}^{N_S} \rho_i H_{i,m} y_{i,m} \tag{2}$$

where $H_{i,m}$ denotes the required hop count between satellite S_i and gateway G_m. Equation (2) indicates that the ISL usage metric can be evaluated by ρ_i, $H_{i,m}$, and $y_{i,m}$. Although values in (2) are time dependent, the equality holds at all times. The average ISL usage will be adopted as the objective function.

B. Hop-Count Calculation Model

According to (2), to evaluate the ISL usage, it is significant to rapidly obtain the hop count $H_{i,m}$. The traditional approach to calculate $H_{i,m}$ is to first construct a graph based on the connections and then solve the shortest path. Generally, this approach consumes much time and computational resources. This section adopts a new approach to calculate the required hop count by considering the MCN's characteristics, including the regular ISL connection pattern and the uniform satellite distribution [20].

We first adopt a theoretical model to calculate the hop count between two satellites when their latitudes and longitudes are given. Then, the model is modified to estimate the required hop count between the satellite and a gateway. The total hops are divided into interplane and intraplane types. According to [30], the minimum total hop count is the sum of the respective minimum of the two types.

1) Hop-Count Between Two Satellites: As shown in Fig. 4, the satellite location is represented by its ground projection: latitude φ and longitude λ. Within the orbit plane, its location can also be characterized by satellite phase angle u. According to the satellite motion, at moment t the transformation is

$$\varphi = \arcsin(\sin\alpha \sin u) \tag{3}$$

$$\lambda = \zeta(u) + \Lambda - \omega_e t \tag{4}$$

where Λ is the initial orbit ascending longitude and ω_e is the Earth rotation angular velocity. $\zeta(u)$ is the longitude difference from the ascending node to current satellite and $\zeta(u)$ can be given by

$$\zeta(u) = \begin{cases} \arctan(\cos\alpha \tan u), & \text{ascending satellite} \\ \arctan(\cos\alpha \tan u) + \pi, & \text{descending satellite.} \end{cases} \tag{5}$$

Given the locations of satellite S_1 and S_2: (φ_1, λ_1) and (φ_2, λ_2). Based on (4), we have $\lambda_2 - \lambda_1 = \zeta(u_2) - \zeta(u_1) + \Lambda_2 - \Lambda_1$. Besides, considering the RAAN difference between the two satellites, we also have $\Lambda_2 - \Lambda_1 = H^h \Delta\Omega$, where H^h is the difference of plane number and also the required interplane hop count between S_1 and S_2. Combining the two expressions above, we have

$$H^h = \frac{\lambda_2 - \lambda_1 + \zeta(u_1) - \zeta(u_2)}{\Delta\Omega} = \frac{\lambda_2 - \lambda_1 + \zeta(u_1) - \zeta(u_2)}{2\pi/N_P}. \tag{6}$$

When packets are forwarded between satellites with different u, we use the satellite phase u to characterize the current packet position. Each intraplane hop causes $\Delta\Omega$ change of the phase, and each interplane hop causes Δf phase change. When the required intraplane hop count is H^v, we have

$\Delta u = u_2 - u_1 = H^v \Delta\Phi + H^h \Delta f$. Therefore

$$H^v = \frac{u_2 - u_1 - H^h \Delta f}{\Delta\Phi} = \frac{u_2 - u_1 - 2\pi H^h F/N_S}{2\pi/M_P}. \quad (7)$$

The total required hop count between S_1 and S_2 is

$$H(S_1, S_2) = \left|H^h\right| + \left|H^v\right|. \quad (8)$$

In the above model, subscripts 1 and 2 represent S_1 and S_2, respectively. Note that the satellites connected by the same type of ISLs form a biconnected "ring," and packets can reach their destination along opposite directions. Both H^h and H^v can be positive or negative. Also, the numerator of (6) and (7) should be normalized into $[-\pi, \pi]$.

Besides, $\alpha \in (-\pi/2, \pi/2)$, whereas $u \in [-\pi, \pi]$. When calculated from (3), the value of u depends on the satellite flying direction, i.e., whether the satellite is ascending or descending type. $\zeta(u)$ depends on the satellite flying direction. Therefore, the hop count $H(S_1, S_2)$ also depends on the flying directions of the two satellites.

2) Hop-Count Between Satellite and Gateway: We further use the above model to solve the required hop count between satellite S_i and gateway G_m. The locations of S_i and G_m are (φ_i, λ_i) and (φ_m, λ_m), respectively. The key problem is to determine the location of the feeder satellite connecting G_m.

Because satellites are densely distributed in the mega-constellation, and generally, the gateway establishes feeder link with the satellite near it, we assume the feeder satellite S_2 is at the same location as G_m. Therefore, we can replace (φ_2,λ_2) by (φ_m,λ_m), and the required hop count between S_i and G_m becomes hop count between S_i and S_2, i.e., $(\varphi_2, \lambda_2) = (\varphi_m, \lambda_m)$ and $H_{i,m} = H(S_i, S_2)$. Note that in this case, the position of S_2 is an approximate and the calculated values of H^h and H^v in (6) and (7) should adopt the nearest integers. Equations (6) and (7) become

$$H^h_{i,m} = \mathcal{R}\left(\frac{\lambda_m - \lambda_i + \zeta(u_i) - \zeta(u_m)}{2\pi/N_P}\right) \quad (9)$$

$$H^v_{i,m} = \mathcal{R}\left(\frac{u_m - u_i - 2\pi H^h_{i,m} F/N_S}{2\pi/M_P}\right) \quad (10)$$

where u_i and u_m can be calculated from (3) with given φ_i and φ_m, and $\mathcal{R}(x)$ is the rounding function that returns the nearest integer of x, $\mathcal{R}(x) = \lfloor x + 0.5 \rfloor$. Therefore, the required hop-count between S_i and G_m is

$$H_{i,m} = \left|H^h_{i,m}\right| + \left|H^v_{i,m}\right|. \quad (11)$$

Note that since the satellites are dense, we further assume the feeder satellite can be either ascending or descending type when calculating u_m, and we adopt the one with the minimal hop-count value to connect S_i. Therefore, given the constellation parameters, $H_{i,m}$ can be directly calculated with the locations of satellite and gateway. Since no recursive or iterative calculations are involved, the above model can achieve a lower computational complexity than traditional graph-based methods. The computational complexity will be further studied in Section VI-D.

V. GSO Problem Formulation

To minimize ISL usage, this section proposes a GSO problem and formulates it as a mixed-integer optimization problem. First, the basic model of the GSO problem is given, which is then discretized in time domain. Then, by introducing a gateway assignment strategy, the GSO problem is decoupled and simplified into a binary integer optimization problem. Finally, an analytical evaluation approach for the problem objective metric is proposed.

A. Basic Model of GSO Problem

The required ISL hops and capacity are affected by the gateway locations. This work aims at minimizing the average ISL usage by optimizing the locations of N_G gateways from M candidate sites. We use an M-dimension 0-1 variable to determine the selected gateway sites: $\mathbf{x} = [x_1, x_2, \ldots, x_M]$. x_m is defined by

$$x_m = \begin{cases} 1, & \text{if a gateway is placed at candidate site } G_m \\ 0, & \text{else.} \end{cases} \quad (12)$$

The objective is to find the best combination of N_G sites so that the average ISL usage of the whole system is minimized. Based on (2), the average ISL usage objective is

$$J \triangleq \frac{1}{T}\int_0^T \frac{1}{N_S}\sum_{m=1}^{M}\sum_{i=1}^{N_S} \rho_i H_{i,m} y_{i,m} \mathrm{d}t \quad (13)$$

where T is the system operation time. At moment t, ρ_i is the traffic input from users to satellite S_i, $H_{i,m}$ denotes the required hop count between S_i and G_m, and $y_{i,m}$ is the ratio of the traffic routed to G_m in ρ_i, $y_{i,m} \in [0, 1]$. The set of $y_{i,m}$ is also the decision variable of the GSO problem. During the system operation, the location and coverage regions change with the satellite movement. Therefore, ρ_i,$H_{i,m}$ and $y_{i,m}$ are all time dependent, and in the objective we consider the time-averaged ISL usage over the operation time. Finally, the GSO problem is formulated as

$$\mathbf{P0:} \min_{\mathbf{x},\mathbf{y}} \; J = \frac{1}{T}\int_0^T \frac{1}{N_S}\sum_{m=1}^{M}\sum_{i=1}^{N_S} \rho_i H_{i,m} y_{i,m} \mathrm{d}t \quad (14a)$$

$$\text{s.t.} \quad \mathbf{x} = [x_1, x_2, \ldots, x_M], x_m \in \{0, 1\} \quad (14b)$$

$$\mathbf{y} = \left[\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_{N_S}\right]^{\mathrm{T}} \quad (14c)$$

$$\mathbf{y}_i = \left[y_{i,1}, y_{i,2}, \ldots, y_{i,M}\right], \quad y_{i,m} \in [0, 1] \quad (14d)$$

$$\sum_{m=1}^{M} x_m = N_G \quad (14e)$$

$$\sum_{m=1}^{M} y_{i,m} = 1 \quad \forall i \in \{1, \ldots, N_S\} \quad (14f)$$

$$y_{i,m} - x_m \le 0 \quad \forall i \in \{1, \ldots, N_S\}, m \in \{1, \ldots, M.\} \quad (14g)$$

Constraints (14b)–(14d) specify the range of the design variables. Constraint (14e) ensures the total number of selected gateway sites equals N_G, and (14f) ensures all the traffic of S_i is routed to gateways. Constraint (14g) ensures that only those selected gateway sites can be the destination of traffic.

TABLE I
NOTATIONS USED IN THE GSO PROBLEM

Notation	Description
S_i	i-th satellite
ρ_i	Traffic input from ground users to satellite S_i
l_j	Traffic on the j-th ISL
t_n	n-th time slot
G_m	m-th candidate gateway site
$H^n_{i,m}$	Hop-count between S_i and G_m at time t_n
H, $\overline{H}$	Hop-count and average hop-count over the whole network
x_m	Design variable indicating if G_m is selected
$y^n_{i,m}$	Ratio of traffic routed to G_m in ρ_i at time t_n
N_S, N_L	Number of total satellites and ISLs, respectively
N_F	Number of feeder satellites
T, N_T	System operation time and number of discretized time slot
M	Number of gateway candidates
N_G	Number of selected gateways
$\mathcal{G}$, $\mathcal{G}'$	Set of gateway candidates and selected gateways, respectively
N_P	Number of orbit planes
M_P	Number of satellites per plane
F	Phasing factor
α	Orbit inclination angle
u	Satellite phase within the orbit
(φ, λ)	Latitude and longitude of satellite
ζ	Longitude difference from the orbit ascending node

The variables used in the GSO problem are listed in Table I. **x** and **y** are integer and continuous design variables, respectively, thus, **P0** is a mixed-integer optimization problem.

B. Time-Domain Discretization

During the system operation, the coverage region and connections change with the satellite movement. In the above model, **x** is time independent, while ρ_i,$H_{i,m}$ and $y_{i,m}$ are time dependent. It is challenging to directly solve the integral of **P0** in the continuous time domain. A widely adopted solution is to discretize the time domain and covert the time domain integral into a summation of several time slots.

We divide the continuous time domain into N_T time slots $\mathcal{T} = \{t_1, t_2, \ldots, t_{N_T}\}$. t_n is the nth slot and $\sum_{n=1}^{N_T} t_n = T$. Within each time slot, the network states can be assumed as static. Then, **P0** can be converted to

$$\textbf{P1:} \min_{\mathbf{x},\mathbf{y}} \; J = \frac{1}{N_T}\frac{1}{N_S}\sum_{n=1}^{N_T}\sum_{m=1}^{M}\sum_{i=1}^{N_S}\rho_i^n H_{i,m}^n y_{i,m}^n \quad \text{s.t.} \;\; (14b) - (14g) \tag{15}$$

where the superscript n indicates the corresponding value in the nth time slot. **y** varies between slots. Next, **P1** is further simplified by solving the optimal **y**.

C. Gateway Assignment Decoupling and Model Simplification

Among all the selected gateway sites, the satellite traffic can reach multiple gateways and the corresponding hop-count numbers are different. The destination of the traffic is specified by design variable **y**, which can be affected by the gateway assignment strategy. To minimize ISL usage, the satellite should forward its traffic to the gateway with the minimum hops. If there exist multiple gateways with the minimum hop count, the traffic should be routed to the gateway with less load for load balance. But in terms of the objective of **P1**, the minimum J can be achieved as long as the traffic is routed to a gateway through the minimum hops. The solution of **y** can be decoupled and **P1** can be simplified.

Given the set of gateway candidate sites $\mathcal{G} = \{G_1, \ldots, G_M\}$. Suppose the selected gateway set is $\mathcal{G}'$, then the remaining unselected gateway site set $\overline{\mathcal{G}'} = \mathcal{G} - \mathcal{G}'$.

Proposition 1: $\forall S_i \in \mathcal{S}$, $\mathbf{y}_i = [y_{i,1}, y_{i,2}, \ldots, y_{i,M}]$

$$\min_{\mathbf{y}_i} \sum_{m=1}^{M} H_{i,m}^n y_{i,m}^n = H_{i,m^*}^n \tag{16}$$

where m^* satisfies $m^* = \arg\min_m H_{i,m}^n$, $m \in \{1, \ldots, M\}$, and $x_{m^*} = 1$.

Proof: According to constraint (14g), if $x_m = 0$, then $y_{i,m}^n = 0 \; \forall S_i \in \mathcal{S}$. $x_m = 0$ indicates $G_m \in \overline{\mathcal{G}}$, thus, $\sum_{G_m \in \overline{\mathcal{G}'}} y_{i,m}^n = 0$, constraint (14f) becomes $\sum_{G_m \in \mathcal{G}'} y_{i,m}^n = 1$, and we have $\sum_{m=1}^{M} H_{i,m}^n y_{i,m}^n = \sum_{G_m \in \mathcal{G}'} H_{i,m}^n y_{i,m}^n$.

At time slot t_n, each satellite $S_i \in \mathcal{S}$ has a minimum hop-count value H_i^* to reach those selected gateways, $H_i^* = \min_{G_m \in \mathcal{G}'} H_{i,m}^n$. Thus $\forall G_m \in \mathcal{G}'$, $H_{i,m}^n \le H_i^*$. $\sum_{G_m \in \mathcal{G}'} H_{i,m}^n y_{i,m}^n \le \sum_{G_m \in \mathcal{G}'} H_i^* y_{i,m}^n = H_i^* \sum_{G_m \in \mathcal{G}'} y_{i,m}^n$. Since $\sum_{G_m \in \mathcal{G}'} y_{i,m}^n = 1$, we have $\sum_{G_m \in \mathcal{G}'} H_{i,m}^n y_{i,m}^n \le H_i^*$, where the equality holds if

$$y_{i,m}^n = \begin{cases} 1, & \text{if } H_{i,m}^n = H_i^*, G_m \in \mathcal{G}' \\ 0, & \text{else.} \end{cases} \tag{17}$$

Finally, $\forall S_i \in \mathcal{S}$, $\min_{\mathbf{y}_i} \sum_{m=1}^{M} H_{i,m}^n y_{i,m}^n = H_i^*$, and (17) gives the corresponding solution of $\mathbf{y_i}$. The proposition is proved. ■

Proposition 1 indicates that for any given **x**, **y** can be solved directly from (17). Since the traffic input ρ_i^n is independent of the design variables, based on Proposition 1, the objective of **P1** can be transformed to

$$J = \frac{1}{N_T}\frac{1}{N_S}\sum_{n=1}^{N_T}\sum_{i=1}^{N_S}\rho_i^n H_{i,m^*}^n \tag{18}$$

where m^* is specified by (16), and H_{i,m^*}^n is related to **x**. Therefore, problem **P1** is decoupled and the solution of **y** can be obtained from (17). Thus, **P1** can be equivalently converted to

$$\textbf{P2:} \min_{\mathbf{x}} \; J = \frac{1}{N_T}\frac{1}{N_S}\sum_{n=1}^{N_T}\sum_{i=1}^{N_S}\rho_i^n H_i^n \tag{19a}$$

$$\text{s.t.} \;\; (14b), \; (14e)$$

$$H_i^n = \min_{G_m \in \mathcal{G}'} H_{i,m}^n. \tag{19b}$$

In the above model, $\mathcal{G}'$ is determined by the value **x**. H_i^n is related to **x** and specified by (19b), which is nonlinear. Thus, **P2** is an integer nonlinear programming problem.

Algorithm 1 Calculation of GSO Objective Metric

Input: $\mathbf{x}$, $\mathcal{G}$
Output: J
1: **for** $G_m \in \mathcal{G}$ **do**
2: **if** $x_m = 1$ **then**
3: $\mathcal{G}' \leftarrow \mathcal{G}' \cup \{x_m\}$
4: **end if**
5: **end for**
6: **for** $t_n \in \mathcal{T}$ **do**
7: Update $(\varphi_i^n, \lambda_i^n)$ and ρ_i^n based on satellite motion
8: **for** $S_i \in \mathcal{S}$ **do**
9: **for** $G_m \in \mathcal{G}'$ **do**
10: Calculate $H_{i,m}^n$ according to (3)-(11)
11: **end for**
12: $H_i^n \leftarrow \min_{G_m \in \mathcal{G}'} H_{i,m}^n$
13: **end for**
14: **end for**
15: Calculate J according to (19a)
16: **return** J

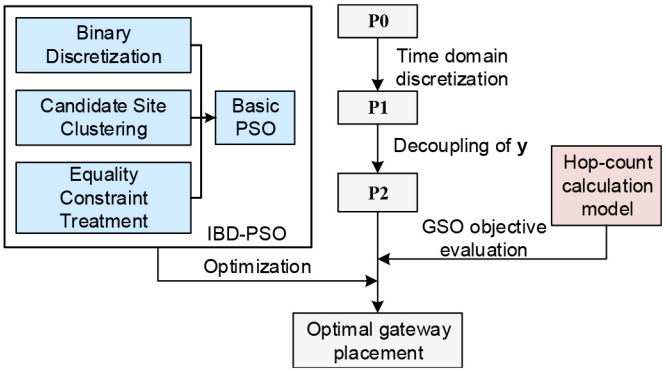

Fig. 5. Overall framework of the GSO solution.

D. Evaluation Method of GSO Objective Metric

From **P1** to **P2**, $\mathbf{y}$ has been decoupled. When evaluating the objective in **P2**, it is necessary to obtain ρ_i^n and $H_{i,m}^n$ that vary with the satellite movement. ρ_i^n can be obtained by adding up the ground demand of the current coverage region, and $H_{i,m}^n$ can be calculated by the analytical model in Section IV-B. Algorithm 1 summarizes the procedures to solve the objective metric with given design variables. Note that no iterative calculation is involved in step 10.

From **P0** to **P1**, the problem becomes solvable by the widely adopted time-domain discretization. Then, by equivalent transformation from **P1** to **P2**, $\mathbf{y}$ is decoupled and the solution of **P1** is simplified. In the next section, we adopt a heuristic PSO algorithm and modify it to solve the GSO problem **P2**. The overall framework of the GSO solution is illustrated in Fig. 5.

VI. IBD-PSO Algorithm for GSO Problem

The converted GSO problem **P2** is still nonlinear and requires massive computations when the gateway candidate number is large. This section proposes an IBD-PSO algorithm to solve the optimization problem. Considering the features of the formulated GSO problem, we improve the basic PSO algorithm in three aspects: 1) binary discretization; 2) candidate site clustering; and 3) equality constraint treatment.

The basic version of the PSO algorithm [31] is described as follows. In an M-dimensional PSO problem with a population size of N_{pop}, particle i $(1 \le i \le N_{pop})$ has a position vector $\mathbf{x_i} = [x_{i1}, x_{i2}, \ldots, x_{iM}]$ and a velocity vector $\mathbf{v_i} = [v_{i1}, v_{i2}, \ldots, v_{iM}]$. Each $\mathbf{x_i}$ is regarded as a solution to the optimization problem, and the corresponding objective function value is the particle's *fitness*. Thus, those particles' positions can be evaluated by comparing their fitness value. During the iteration, the elements of $\mathbf{v_i}$ and $\mathbf{x_i}$ are updated by

$$\begin{aligned} v_{ij} &\leftarrow \omega v_{ij} + c_1 r_{1j}(p_{ij} - x_{ij}) + c_2 r_{2j}(g_j - x_{ij}) \\ x_{ij} &\leftarrow x_{ij} + v_{ij} \end{aligned} \tag{20}$$

where p_{ij} and g_j are elements in the individual historical best position $\mathbf{p_i^{best}} = [p_{i1}, p_{i2}, \ldots, p_{iM}]$ and the global best position among the current population $\mathbf{g^{best}} = [g_1, g_2, \ldots, g_M]$. ω, c_1, and c_2 are weight coefficients when the velocity is updated. r_{1j} and r_{2j} are random numbers uniformly distributed in $[0, 1]$. The final PSO output $\mathbf{g^{best}}$ can be taken as the output solution $\mathbf{x}$ of the GSO problem. Next, considering the features of **P2**, three schemes are implemented on the above basic PSO algorithm.

A. Binary Discretization

Since the variables in the GSO problem are all binary integers, i.e., either 1 or 0, we adopt a bivelocity to discretize the PSO algorithm. The position vector $\mathbf{x_i}$ maintains unchanged

$$\mathbf{x_i} = [x_{i1}, x_{i2}, \ldots, x_{iM}], x_{ij} \in \{0, 1\} \tag{21}$$

while the velocity vector $\mathbf{v}_i$ adopts a bivelocity form [21], [32]

$$\mathbf{v_i} = \begin{bmatrix} \mathbf{v_i^0} \\ \mathbf{v_i^1} \end{bmatrix} = \begin{bmatrix} v_{i1}^0, v_{i2}^0, \ldots, v_{iM}^0 \\ v_{i1}^1, v_{i2}^1, \ldots, v_{iM}^1 \end{bmatrix}, \quad v_{ij}^0, v_{ij}^1 \in [0, 1]. \tag{22}$$

Here, v_{ij}^0 and v_{ij}^1 indicate the possibilities of x_{ij} equaling 0 and 1, respectively. The bivelocity updating rule is as follows. Assume that $\mathbf{x_2}$ is a better position than $\mathbf{x_1}$ in terms of the fitness function, and $\mathbf{x_1}$ will learn from $\mathbf{x_2}$. We compare $\mathbf{x_2}$ and $\mathbf{x_1}$ and give the updating bivelocity increment $\Delta\mathbf{v} = [\Delta v_j^0 \quad \Delta v_j^1]^{\mathrm{T}}$. If $x_{2j} = x_{1j}$, then $\Delta v_j^0 = 0$ and $\Delta v_j^1 = 0$. When $x_{2j} \neq x_{1j}$, there are two situations: if $x_{2j} = 1$ and $x_{1j} = 0$, then $\Delta v_j^0 = 0$ and $\Delta v_j^1 = 1$; if $x_{2j} = 0$ and $x_{1j} = 1$, then $\Delta v_j^0 = 1$ and $\Delta v_j^1 = 0$.

With the obtained $\Delta\mathbf{v}$ and a given critical coefficient β, $\mathbf{v_i}$ and $\mathbf{x_i}$ can be updated as follows:

$$v_{ij} = \max\left\{\omega v_{ij}, c_1 r_{1j} \Delta v_{ij}^p, c_2 r_{2j} \Delta v_{ij}^g\right\} \tag{23}$$

$$x_{ij} = \begin{cases} \text{rand}(0, 1), & \text{if } v_{ij}^0 > \beta \text{ and } v_{ij}^1 > \beta \\ 0, & \text{if } v_{ij}^0 > \beta \text{ and } v_{ij}^1 \le \beta \\ 1, & \text{if } v_{ij}^0 < \beta \text{ and } v_{ij}^1 > \beta \\ x_{ij}, & \text{if } v_{ij}^0 < \beta \text{ and } v_{ij}^1 \le \beta. \end{cases} \tag{24}$$

During the calculation of $\mathbf{v_i}$ and $\mathbf{x_i}$, to meet the boundary conditions, we regulate that any element of $\mathbf{v}$ that exceeds one is set to one.

B. Candidate Site Clustering

Candidate site clustering is only used for generating the initial population of the PSO algorithm, but not affects the problem model. Because the candidate sites are not uniformly distributed, some sites can be close to each other, which is not desired. To speed up the optimization, we divide the candidate sites into several clusters and avoid site gathering in clusters in the PSO population initialization stage. We use K-means clustering [33] to partition M candidate sites into N_K^* clusters $\mathcal{G} = \{\mathcal{G}_1, \ldots, \mathcal{G}_{N_K^*}\}$ based on their geographical locations, where N_K^* is the predetermined number of clusters. The objective of clustering is to find $\arg\min_{\mathcal{G}_i} \sum_{i=1}^{N_K^*} \sum_{G_m \in \mathcal{G}_i} \|\mathbf{q}_m - \boldsymbol{\mu}_i\|^2$, where $\mathbf{q}_m$ is the position of candidate site G_m and $\boldsymbol{\mu}_i$ is the mean position of cluster $\mathcal{G}_i$. $\boldsymbol{\mu}_i = (1/|\mathcal{G}_i|) \sum_{G_m \in \mathcal{G}_i} \mathbf{q}_m$, and $|\cdot|$ denotes the cardinality of the set.

To avoid the site concentration, those obtained clusters will be further revised. We define the diameter of each cluster as $d_i = (1/|\mathcal{G}_i|) \sum_{G_m \in \mathcal{G}_i} \|\mathbf{q}_m - \boldsymbol{\mu}_i\|$, and set a diameter threshold d_{th}. If $d_i < d_{th}$, $\mathcal{G}_i$ will be merged with the cluster with the current maximum diameter. After the cluster combination, the final cluster number becomes N_K. When generating the initial population, we proportionally select sites in a random way for each cluster so that the total number of selected sites equals N_G. Note that the site clustering is only applied in the population initialization phase but does not constrain the site number of each cluster in the final optimal gateway layout.

C. Equality Constraint Treatment

Through the problem transformation from **P0** to **P2**, the constraints are simplified. Those $y_{i,m}^n$-related gateway assignment constraints have been solved by (16) and (17), and constraint (19b) can be solved when calculating the objective metric according to Algorithm 1. The only remaining constraint is the equality constraint of N_G in (14e).

To solve (14e), we modify the PSO algorithm by simultaneously adding and removing the selected sites to guarantee a constant N_G. In each iteration, after $\mathbf{x_i}$ is updated according to (20)–(24), the current total number of selected sites is updated and denoted as N'_G. If $N'_G - N_G \neq 0$, then $\mathbf{x_i}$ is further modified by

$$\begin{cases} \arg\min_{x_{ij}} v_{ij}^1 \leftarrow 0, & \text{if } N'_G - N_G > 0 \\ \arg\min_{x_{ij}} v_{ij}^0 \leftarrow 1, & \text{if } N'_G - N_G < 0. \end{cases} \tag{25}$$

Equation (25) is repeated until $N'_G - N_G = 0$. Thus, at the end of each iteration, the selected gateway number maintains N_G, and the equality constraint is solved.

The three schemes of IBD-PSO, i.e., binary discretization, candidate site clustering, and equality constraint treatment, have jointly improved the classical PSO algorithm for solving the GSO problem. The overall calculation of the proposed IBD-PSO algorithm is summarized in Algorithm 2.

Algorithm 2 IBD-PSO Algorithm

Input: M,N_G, $\mathcal{G}$
Output: $\mathbf{x}^{\mathbf{best}}$

Initialization:
Cluster candidate sites $\mathcal{G} = \{\mathcal{G}_1, ..., \mathcal{G}_{N_K^*}\}$ based on K-means
Merge close clusters
Generate the initial population based on the clustering results
Iteration:
while $k \leq N_{iter}$ **do**
  Update ω
  for i = 1 to N_{pop} **do**
    Update $\mathbf{v_i}$, $\mathbf{x_i}$ according to (20)-(24)
    while $N'_G \neq N_G$ **do**
      Modify $\mathbf{x_i}$ by (25)
      Update N'_G
    end while
    Evaluate the fitness of $\mathbf{x_i}$ and update $\mathbf{p_i^{best}}$,$\mathbf{g^{best}}$
  end for
  $k \leftarrow k + 1$
end while
return $\mathbf{x}^{\mathbf{best}} \leftarrow \mathbf{g}^{\mathbf{best}}$

D. Computational Complexity Analysis

The computation complexity can be divided into two parts: 1) computation for PSO optimization and 2) computation for evaluating the objective function value. For the first part, the computational complexity is $O(MN_{KI} + N_{pop}N_{iter})$ [19]. The K-means takes $O(MN_{KI})$ to obtain the best clustering result, where M is the number of candidate sites and N_{KI} is the iteration number of the K-means algorithm. As for the remaining PSO algorithm operation, the computational complexity is $O(N_{pop}N_{iter})$ in the worst case [34].

In terms of the second part, the computations for evaluating the objective metric mainly come from the satellite trajectory propagation and hop-count calculation. The satellite trajectory propagation is performed by a dedicated simulation platform and is not discussed here. Once the satellite position is obtained, the hop count can be directly calculated using the analytical model in Section IV-B without graph construction or iterative calculation. This method is independent of the network size, and the computation complexity for evaluating the hop count between a satellite and a gateway is $O(1)$. Simulation results have shown that evaluating the hop count of a satellite takes only 7 μs on average with an Intel Core i7-7700 3.6-GHz processor.

However, the hop-count evaluation will be more costly if classical shortest path algorithms are adopted. The Bellman-Ford algorithm and Dijkstra's algorithm have $O(N_S N_L)$ and $O(N_S^2)$ time complexities [35], respectively, for the single-source shortest-path problem, where N_S and N_L are the total satellite number and total edge number. Since in our network topology, $N_L = 2N_S$, the complexity of both algorithms can be regarded as $O(N_S^2)$, which is much higher than the proposed analytical approach. Furthermore, the computation complexity for calculating the hop count between all satellites and all gateways through T should be $O(N_S N_G N_T)$, where N_T is the number of total time slots. In the following section, the time consumption will be compared in the simulations.

TABLE II
BASIC SETTINGS OF SIMULATION SCENARIO

Variable	Description	Value
N_P	Number of orbit planes	24
M_P	Number of satellites per plane	66
F	Phasing factor	0
α	Orbit inclination angle	53°
M	Number of gateway candidates	74
N_G	Number of selected gateways	30
T	System operation time	30 min
N_{pop}	Number of particles in PSO	50
N_{iter}	Number of the maximum iteration	100
c_1	Weight coefficient	2
c_2	Weight coefficient	2
β	Critical coefficient	0.5

VII. SIMULATION AND RESULTS

In this section, we first give the optimized results of the GSO problem with uniformly distributed user demand and compare the proposed evaluation method and solving algorithm with existing studies. Then, we investigate the relations between ISL hop-count and multiple factors, including gateway number, satellite flying direction, and traffic demand pattern. Then, we study the gateway placement with the nonuniform demand pattern.

A. Simulation Settings

The basic settings of the simulation scenario are listed in Table II. We adopt the first layer of Starlink constellation [36]: $N_P = 24$, $M_P = 66$, $\alpha = 53°$ with $F = 0$. Then, the satellite ephemerides are generated by orbital simulator. We adopt the same candidate gateway site set as in [37], where the candidate sites are selected based on the following considerations: 1) global coverage; 2) no strong spatial correlations; and 3) realistic potential sites. Since the gateways should have good connections with the ground core network, we further remove the candidate sites in the Antarctic. Finally, 74 sites are selected as potential candidate sites (see Fig. 9).

The parameters for the IBD-PSO algorithm refer to [21] and are set as $N_{pop} = 50$, $c_1 = c_2 = 2$, and $\beta = 0.5$. ω ranges from 0.9 to 0.4 and varies linearly with the iteration. The maximum iteration number N_{iter} is 100. Note that the iteration will stop before N_{iter} if the best-obtained solution maintains unchanged for 30 iterations. To evaluate the optimization performance, we also adopt three approaches for comparison: 1) greedy strategy (GS) that successively traverses and finds the local optimal sites in each cluster without recursion [16]; 2) modified discrete PSO (MD-PSO) as in [21]; and 3) GA as in [10]. We adopt the same population size and stop criterion as the IBD-PSO algorithm. All the simulations are run in an STK/MATLAB integrated simulation platform with Intel Core i7-7700 3.6-GHz CPU and 16-GB memory.

To investigate the sole effect of the gateway placement on the Sat-to-Gateway hops, we first consider a uniform user demand scenario by setting $\rho_i^n = 1\ \forall S_i \in \mathcal{S}, t_n \in \mathcal{T}$ and investigate the optimal gateway placement affected by the ISL hop count. The objective becomes

$$J' = \frac{1}{N_T}\frac{1}{N_S}\sum_{n=1}^{N_T}\sum_{i=1}^{N_S} H_i^n. \tag{26}$$

The above objective can be interpreted as the average hop-count $\overline{H}$ over the whole constellation, i.e., $J' = \overline{H}$.

Next, in Section VII-D, we adopt a nonuniform user demand distribution and analyze the gateway placement of **P2**. Compared to (26), the objective of **P2** can be regarded as a traffic-weighted average hop count. The total traffic demand is set as 20 Gb/s, and generally, the demand density is proportional to the user density [38]. Considering the IoT applications in the marine regions, 5% of the total traffic is assumed to be evenly distributed in all the ocean regions.

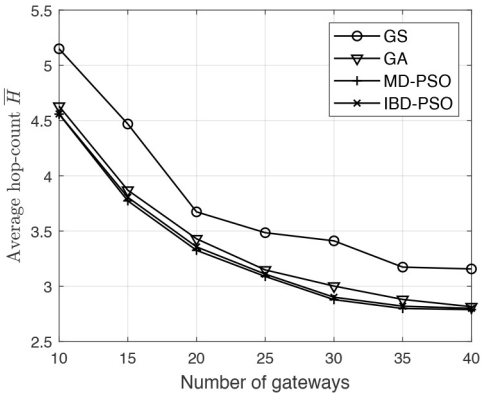

Fig. 6. Average hop counts in the best-obtained gateway placement with different optimization algorithms.

B. Results of Best-Obtained Gateway Placement

Fig. 6 shows the average hop count $\overline{H}$ in the best-obtained gateway placement with different gateway numbers. With more gateways, the satellites have more gateway choices and require fewer hops to reach a gateway. $\overline{H}$ drops fast when $N_G \leq 30$ and the decline trend gets slower when N_G increases. With 30 gateways, the required average hops for the satellites can be reduced to less than three hops when the gateway placement is optimized.

1) Performance Comparison: Fig. 6 also compares the performances of the three algorithms. For each point in Fig. 6, the optimization is run ten times, and the averaged $\overline{H}$ is plotted. The results of GS strategy can be regarded as the nonoptimized results. Compared to the GS strategy, Fig. 6 shows that the optimized gateway placement can achieve a 13.3% lower usage of ISL on average. While compared to the GA algorithm [10], the IBD-PSO algorithm can achieve a better performance in all the cases. On average, the objective function value $\overline{H}$ is 2.43% lower than that obtained by GA. In terms of the algorithm efficiency, with the same population size, the average computation times are 1.13×10^4 s and 1.18×10^4 s for IBD-PSO and GA, respectively. Therefore, with 4.42% less time consuming, the IBD-PSO algorithm can still achieve a better result than GA. Compared to the MD-PSO algorithm [21], the two algorithms have very close performance on objective value, but the IBD-PSO consumes 1.96% less time than MD-PSO. The reason

TABLE III
COMPARISON AND VERIFICATION OF THE ISL USAGE EVALUATION METHODS

Gateway number	Average traffic on ISLs (Mbps)		Relative error
	Proposed approach	Simulation approach in [10]	
10	28.8	29.5	2.44%
20	21.2	21.5	1.42%
30	18.3	19.0	3.83%
40	17.7	17.8	0.56%

may be that the population initialization scheme speeds up the optimization.

Next, we also verify the proposed ISL usage evaluation method by comparing it to the traditional method with network simulator (e.g., in [10] and [22]). We establish the same network simulation platform as in [10] and run the network simulation with the same scenario settings (e.g., constellation parameters, traffic demand, and optimized gateway placement). The total traffic demand is set to 20 Gb/s, and ISL bandwidth is set to unlimited [22]. Then, we calculate the traffic on each ISL and obtain the average ISL usage (see Table III). The comparison results show that the obtained ISL usages of the two approaches are close, and the relative error is less than 4%, which verifies the proposed evaluation method of ISL usage. With the same candidate gateways, we also compare the gateway optimization results of the two methods and find that the gateway distribution patterns are similar. The optimized ISL usage has an average deviation of less than 5%. Considering the randomness of the PSO algorithm, the comparison results can verify the proposed evaluation method.

Furthermore, the computation time consumption of the proposed method is also compared with the overall method as in [10]. Note that all the satellite ephemerides are precalculated and not counted in the computation time. The computation time with the simulation method in [10] is 7.23×10^4 s, while the corresponding time with the proposed method is 1.13×10^4 s, saved by 84.4%. The reason is that when evaluating the objective, the traditional network simulation method consumes much time for the packets to calculate the optimal route. In the proposed solving framework, this step is avoided and replaced by the faster hop-count evaluation method. Therefore, the proposed method greatly reduces the overall computation time compared to the traditional method.

2) Sensitivity Analysis of the IBD-PSO Algorithm: Since the algorithm parameters listed in Table II will affect the optimization performance, sensitivity analysis is studied on those parameters, including population size N_{pop}, critical coefficient β, and weight coefficient c_1 and c_2. When a parameter is analyzed, the other parameters keep unchanged. We adopt two metrics to evaluate the algorithm performance, i.e., the best fitness value J' and the count of objective function evaluation N_{obj}. Every simulation with the same settings is repeated for ten times and the average performance is adopted.

The sensitivity analysis results are shown in Fig. 7. With the increase of β, the best fitness also increases with fluctuations, while the required calculation of objective function decreases significantly, especially when $\beta \leq 0.5$. To balance

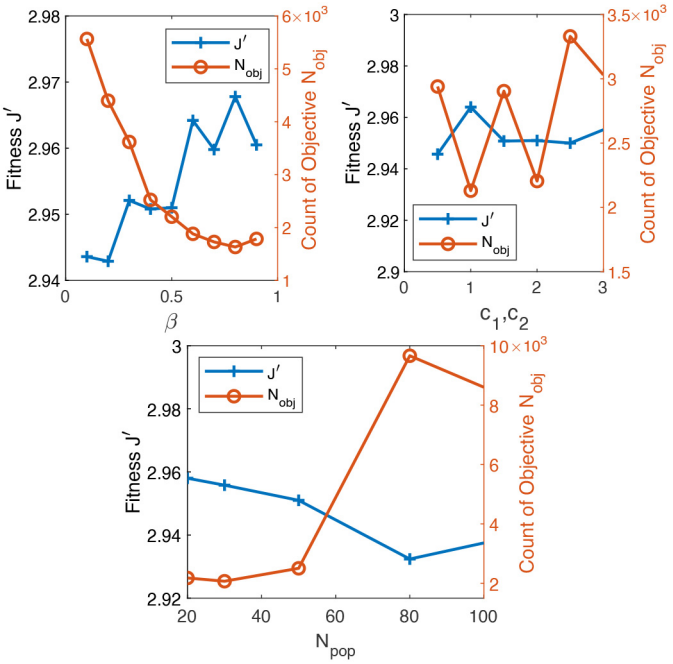

Fig. 7. Sensitivity analysis of the IBD-PSO algorithm.

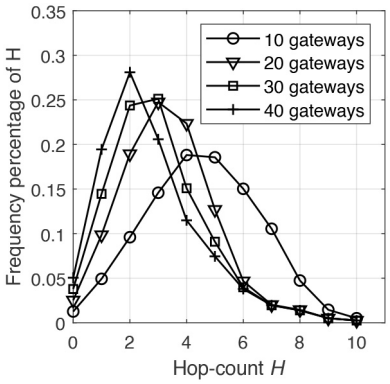

Fig. 8. Normalized histogram of required hop count of all satellites. The plotted proportion is the ratio of required hop-count value among all the satellites under the best-obtained gateway placement.

the optimization performance and computation overhead, β within [0.4, 0.5] would be suitable. In terms of c_1 and c_2, the variation of the performance metrics is more random, and 2 can be acceptable. As for N_{pop}, a larger N_{pop} increases the possibility to find a better solution, but it also causes a much higher computation overhead. $N_{pop} \in [50, 60]$ can achieve a good tradeoff.

3) Hop-Count Distribution: The hop-count histogram of all the satellites is illustrated in Fig. 8. In all the cases, the hop-count ranges from 0 to 10. But when the gateway number increases, the peak of the curve moves to the left, which means more satellites can reach a gateway with fewer hops. When $N_G = 30$, over 90% of the satellites can reach a gateway within 5 hops.

Take $N_G = 30$ as an example, Fig. 9 gives the selected sites and candidate sites. Although the candidate sites are not uniformly distributed in space, the selected sites show a balance in overall distribution: the western/eastern hemisphere

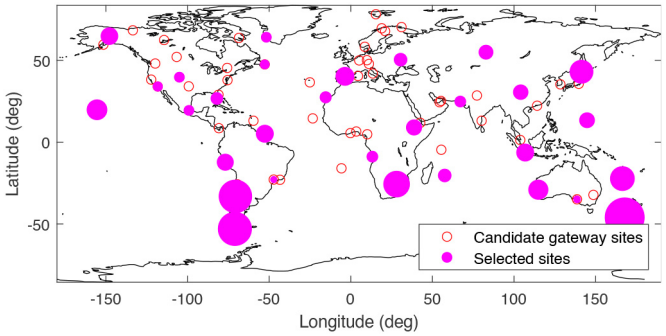

Fig. 9. Best-obtained 30 gateway sites. The pink circle size indicates the number of satellites that access the gateway.

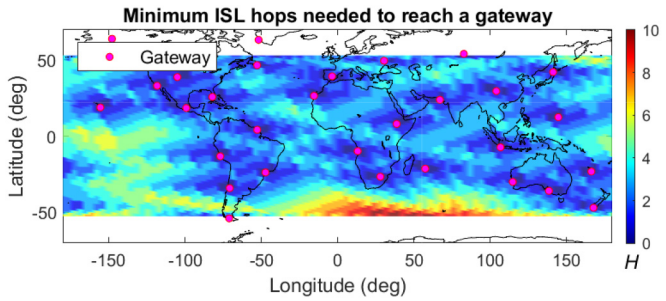

Fig. 10. Spatial distribution of the required hops to reach a gateway. The color indicates the required hop-count number.

and eastern/western hemisphere each have approximately half of the gateways, respectively. Fig. 9 also shows how many satellites access the same gateway, which can be interpreted as the service scope of each gateway. Gateways in the south end of Chile, South Africa, and New Zealand have a significantly larger service scope. The reasons are as follows: one is that satellites are more densely distributed in the high-latitude regions [1]; the other is that these sites are surrounded by oceans without available sites. Furthermore, comparing the results obtained from scenarios with different N_G, we find that sites in Hawaii, New Zealand, Guam, South Africa, and the south end of Chile are more popular in the selected sites.

Fig. 10 illustrates the hop-count spatial distribution for all the satellites. Because the orbit inclination is 53°, the latitudes of satellites are also within [53 °S, 53 °N] range. The required ISL hop count increases when satellites are far from a gateway. The highest hop-count regions appear over the South Atlantic and South India Oceans. Satellites in these regions may need up to ten hops to reach a gateway. The multihop path examples in these regions will be later discussed in the next section. Combined with Fig. 8, we find that although N_G is increased, the max H remains up to 10 and always occurs in these regions. If it is possible to construct a gateway in the ocean of these regions, the max H can be further reduced.

C. Difference Between Ascending and Descending Satellites

When investigating the hop-count distribution, we found that some satellites require more hops than their neighboring satellites even though they are closer to the gateway. The required hop count is not only determined by the distance to the gateway, but also affected by the satellite flying directions. Because the required hops for connecting a gateway depend on whether the satellite is ascending or descending, even when two satellites of different types are at the same location, they may select different gateways with different H.

Fig. 11 explains the difference. We select a moment when an SatA and a SatD are both at (48 °S, 5 °W). According to Fig. 11(a), SatA only needs five hops to reach the gateway GW1 in South Africa. SatA-GW1 line is along the southwest–northeast direction, which agrees with the intraplane ISL direction of SatA so that the usage of interplane ISL is minimized. However, if a SatD is at the same location, the least required hop count becomes 10 [see Fig. 11 (b)]. Because its intraplane ISLs are along the northwest–southeast direction, Sat D needs extra interplane hops to reach GW1. For SatD, the optimal destination is GW2 in Brazil with $H = 7$. Comparing SatA and SatD, we find that although they are at the same location, they select different gateways as their destination and have different H. From the aspect of geographical distance, both SatA and SatD are closer to GW1. But in this case, SatD should access the gateway GW2 with a longer distance but fewer hops due to the difference of intraplane ISL directions.

Note that this phenomenon is rare in a polar orbit MCN as in [23]. Because polar constellations are usually π-type, ascending and descending orbits are separated, and in most regions, there is only one type of satellite overhead. But satellites at different sides of the *Seam* [20] will have quite different hop counts for reaching the same gateway.

D. Nonuniform Ground Traffic Pattern

The above analyses are based on the uniform demand distribution. In this section, we adopt the nonuniform demand pattern as described in Section VII-A, and the total traffic load is 20 Gb/s. Other scenario settings remain the same as in Section VII-B.

The best-obtained gateway placement is illustrated in Fig. 12. Comparing Fig. 12 with Fig. 9, the general gateway distribution patterns are similar. But those gateway sites in the outer ocean areas are not preferred, and gateways are more placed in North America, Europe, and East Asia. Since satellites with higher traffic load consume more ISL resources, they desire as few ISL hops as possible to reach the gateway. Thus, gateways will be placed close to those regions with high traffic demand. Although there is traffic demand in vast ocean areas, the volume is still smaller compared to the land area. Thus, fewer gateways are placed in ocean areas. But this will also lead to a higher hop count for these remote regions to connect a gateway. Fig. 12 also illustrates the traffic load that converges at each gateway. Due to the traffic convergence effect, gateways at the demand-dense regions have higher traffic loads, e.g., those in North America and Europe. In these regions, the system capacity can be enhanced by adding more antenna dishes or improving the feeder link bandwidth.

Furthermore, we repeat the simulations ten times and calculate the occurrence frequency of each gateway site in the best-obtained placement. The results show that gateway sites

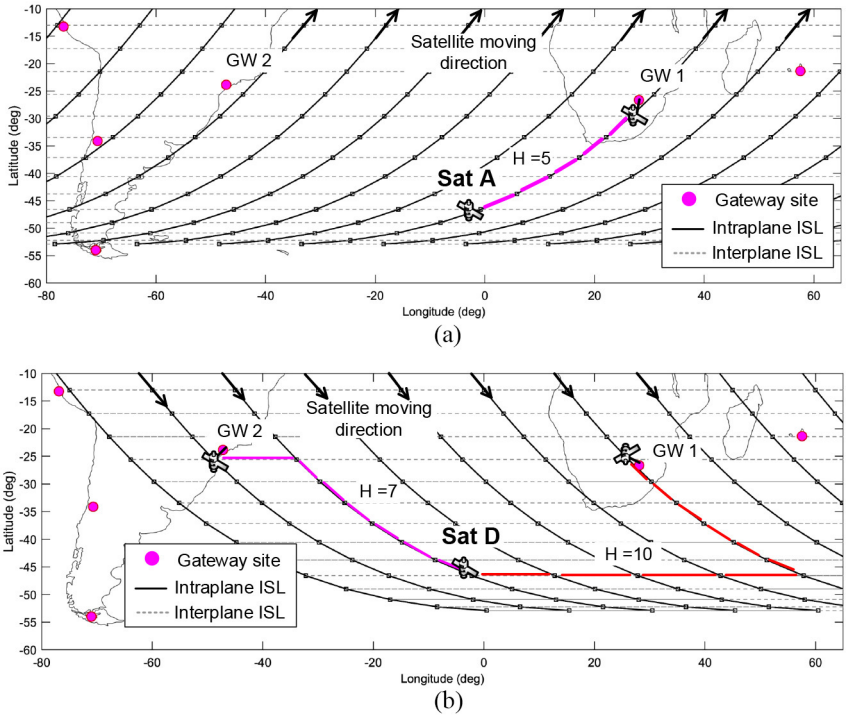

Fig. 11. Required hop count of satellites with different flying directions in the South Atlantic. Although SatA and SatD are at the same location (48 °S, 5 °W), when they are, respectively, ascending and SatDs, the required hops to reach the gateway are different. (a) SatAs. (b) SatDs.

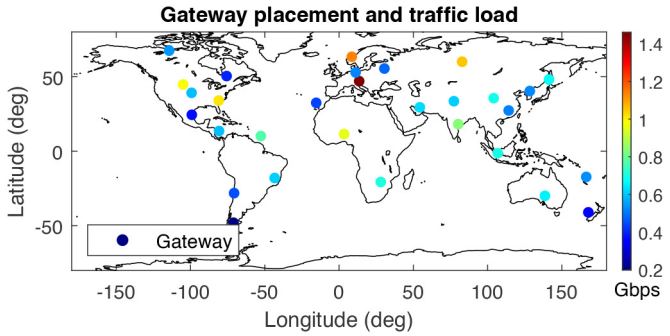

Fig. 12. Best-obtained 30 gateway sites under nonuniform user demand. The color indicates the traffic load that converges on the gateway.

in the eastern U.S., western Europe, the trait of Malacca, and the south end of Chile are of higher popularity in the optimal gateway placement, which accords with the results in [10].

VIII. Conclusion

This article proposes a GSO model to minimize the ISL usage in MCNs by optimizing the gateway locations. The GSO problem is formulated as a mixed-integer optimization model and further simplified to an integer optimization problem. A novel analytical hop-count model is adopted to calculate the required hop count and evaluate the ISL usage metric, which avoids complex graph-based routing calculations. An improved binary discrete PSO algorithm is proposed to solve the GSO problem. The results of case studies show that the proposed algorithm outperforms other similar algorithms. The proposed ISL usage evaluation method is also verified by comparisons.

We further study the effects of several system parameters on the ISL hop-count distribution and have the following findings: 1) the ISL usage can be characterized by a traffic-weighted average hop-count metric; 2) increasing the gateway number and optimizing the gateway locations can effectively reduce the ISL usage; and 3) when the traffic demand is nonuniformly distributed, gateway sites at demand-dense regions are preferred, and those sites in the outer sea are not preferred.

References

[1] 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 Astronaut.*, vol. 159, pp. 123–135, Jun. 2019.

[2] A. Kak and I. F. Akyildiz, "Designing large-scale constellations for the Internet of Space Things with CubeSats," *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1749–1768, Feb. 2021.

[3] M. De Sanctis, E. Cianca, G. Araniti, I. Bisio, and R. Prasad, "Satellite communications supporting Internet of Remote Things," *IEEE Internet Things J.*, vol. 3, no. 1, pp. 113–123, Feb. 2016.

[4] Z. Song, Y. Hao, Y. Liu, and X. Sun, "Energy efficient multi-access edge computing for terrestrial-satellite Internet of Things," *IEEE Internet Things J.*, vol. 8, no. 18, pp. 14202–14218, Sep. 2021.

[5] J. Fraire, S. Céspedes, and N. Accettura, "Direct-to-satellite IoT—A survey of the state of the art and future research perspectives: Backhauling the IoT through LEO satellites," in *Proc. ADHOC-NOW*, 2019, pp. 241–258.

[6] L. Tian, N. Huot, O. Chef, and J. Famaey, "Self-organising LEO small satellite constellation for 5G MTC and IoT applications," in *Proc. NoF*, 2020, pp. 100–104.

[7] J. A. R. D. Azúa, A. Calveras, and A. Camps, "Internet of Satellites (IoSat): An interconnected space paradigm," in *Proc. 6th FFSS Workshop*, 2017, pp. 1–8.

[8] R. Birkeland and D. Palma, "An assessment of IoT via satellite: Technologies, services and possibilities," in *Proc. IAC*, 2019, pp. 1–12.

[9] Z. Qu, G. Zhang, H. Cao, and J. Xie, "LEO satellite constellation for Internet of Things," *IEEE Access*, vol. 5, pp. 18391–18401, 2017.

[10] Q. Chen, L. Yang, X. Liu, J. Guo, S. Wu, and X. Chen, "Multiple gateway placement in large scale constellation networks with inter satellite links," *Int. J. Satell. Commun. Netw.*, vol. 39, no. 1, pp. 47–64, 2021.
[11] M. Centenaro, C. E. Costa, F. Granelli, C. Sacchi, and L. Vangelista, "A survey on technologies, standards and open challenges in satellite IoT," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1693–1720, 3rd Quart., 2021.
[12] C.-Q. Dai, M. Zhang, C. Li, J. Zhao, and Q. Chen, "QoE-aware intelligent satellite constellation design in satellite Internet of Things," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4855–4867, Mar. 2021.
[13] J. Liu, Y. Shi, Z. M. Fadlullah, and N. Kato, "Space-air-ground integrated network: A survey," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 2714–2741, 4th Quart., 2018.
[14] T. de Cola *et al.*, "Network and protocol architectures for future satellite systems," *Found. Trends Netw.*, vol. 12, nos. 1–2, pp. 1–161, 2017.
[15] S. Liu, J. Yang, X. Guo, and L. Sun, "Inter-satellite link assignment for the laser/radio hybrid network in navigation satellite systems," *GPS Solut.*, vol. 24, no. 2, pp. 1–14, 2020.
[16] B. Aoun, R. Boutaba, Y. Iraqi, and G. Kenward, "Gateway placement optimization in wireless mesh networks with QoS constraints," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 11, pp. 2127–2136, Nov. 2006.
[17] I. Gravalos, P. Makris, K. Christodoulopoulos, and E. A. Varvarigos, "Efficient network planning for Internet of Things with QoS constraints," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3823–3836, Oct. 2018.
[18] M. Rady, M. Hafeez, and S. A. R. Zaidi, "Computational methods for network-aware and network-agnostic IoT low power wide area networks (LPWANs)," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 5732–5744, Jun. 2019.
[19] M. Raithatha, A. U. Chaudhry, R. H. M. Hafez, and J. W. Chinneck, "A fast heuristic for gateway location in wireless backhaul of 5G ultra-dense networks," *IEEE Access*, vol. 9, pp. 43653–43674, 2021.
[20] Q. Chen *et al.*, "Analysis of inter-satellite link paths for LEO mega-constellation networks," *IEEE Trans. Veh. Technol.*, vol. 70, no. 3, pp. 2743–2755, Mar. 2021.
[21] J. Guo, D. Rincón, S. Sallent, L. Yang, X. Chen, and X. Chen, "Gateway placement optimization in LEO satellite networks based on traffic estimation," *IEEE Trans. Veh. Technol.*, vol. 70, no. 4, pp. 3860–3876, Apr. 2021.
[22] C. Zhu, Y. Li, M. Zhang, Q. Wang, and W. Zhou,, "An optimization method for the gateway station deployment in LEO satellite systems," in *Proc. IEEE VTC Spring*, 2020, pp. 1–7.
[23] Y. Vasavada, R. Gopal, C. Ravishankar, G. Zakaria, and N. BenAmmar, "Architectures for next generation high throughput satellite systems," *Int. J. Satell. Commun. Netw.*, vol. 34, no. 4, pp. 523–546, 2016.
[24] J. Kopacz, J. Roney, and R. Herschitz, "Optimized ground station placement for a mega constellation using a genetic algorithm," in *Proc. SSC*, 2019, pp. 1–9.
[25] F. Ma *et al.*, "Hybrid constellation design using a genetic algorithm for a LEO-based navigation augmentation system," *GPS Solut.*, vol. 24, no. 2, p. 62, 2020.
[26] M. Handley, "Delay is not an option," in *Proc. HotNet*, 2018, pp. 85–91.
[27] G. Mao, Z. Lin, X. Ge, and Y. Yang, "Towards a simple relationship to estimate the capacity of static and mobile wireless networks," *IEEE Trans. Wireless Commun.*, vol. 12, no. 8, pp. 3883–3895, Aug. 2013.
[28] T. Y. Chung, N. Sharma, and D. P. Agrawal, "Cost-performance tradeoffs in manhattan street network versus 2-D torus," *IEEE Trans. Comput.*, vol. 43, no. 2, pp. 240–243, Feb. 1994.
[29] F. Kelly and E. Yudovina, *Stochastic Networks (Institute of Mathematical Statistics Textbooks)*. Cambridge, U.K.: Cambridge Univ. Press, 2014.
[30] R. Kucukates and C. Ersoy, "Minimum flow maximum residual routing in LEO satellite networks using routing set," *Wireless Netw.*, vol. 14, no. 4, pp. 501–517, 2008.
[31] B. Jarboui, N. Damak, P. Siarry, and A. Rebai, "A combinatorial particle swarm optimization for solving multi-mode resource-constrained project scheduling problems," *Appl. Math. Comput.*, vol. 195, no. 1, pp. 299–308, 2008.
[32] M. Shen, Z.-H. Zhan, W.-N. Chen, Y.-J. Gong, J. Zhang, and Y. Li, "Bi-velocity discrete particle swarm optimization and its application to multicast routing problem in communication networks," *IEEE Trans. Ind. Electron.*, vol. 61, no. 12, pp. 7141–7151, Dec. 2014.
[33] Y. Xu, W. Xu, Z. Wang, J. Lin, and S. Cui, "Load balancing for ultra-dense networks: A deep reinforcement learning-based approach," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 9399–9412, Dec. 2019.
[34] W. Chen, J. Zhang, H. S. H. Chung, W. Zhong, W. Wu, and Y. Shi, "A novel set-based particle swarm optimization method for discrete optimization problems," *IEEE Trans. Evol. Comput.*, vol. 14, no. 2, pp. 278–300, Apr. 2010.
[35] W. Peng, X. Hu, F. Zhao, and J. Su, "A fast algorithm to find all-pairs shortest paths in complex networks," *Procedia Comput. Sci.*, vol. 9, pp. 557–566, Dec. 2012.
[36] *SpaceX Non-Geostationary Satellite System (Attachment A)*, Federal Commun. Comm., Washington, DC, USA, 2018.
[37] I. D. Portillo, B. Cameron, and E. Crawley, "Ground segment architectures for large LEO constellations with feeder links in EHF-bands," in *Proc. AeroConf*, 2018, pp. 1–14.
[38] F. Long, *Satellite Network Robust QoS-Aware Routing*. Heidelberg, Germany: Springer-Verlag, 2014.

Quan Chen received the B.E. and Ph.D. degrees from the National University of Defense Technology (NUDT), Changsha, China, in 2015 and 2021, respectively.

He is currently a Lecturer with the College of Aerospace Science and Engineering, NUDT. His research interests include megaconstellation satellite networks, UAV networks, and integrated space-terrestrial networks.

Dr. Chen has served as a reviewer for several journals, including IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON AEROSPACE AND ELECTRONIC SYSTEMS, and IEEE COMMUNICATIONS LETTERS. He has served as a TPC member of IEEE ICC workshop on megaconstellation in 2021 and 2022.

Lei Yang received the Ph.D. degree from the College of Aerospace Science and Engineering, National University of Defense Technology (NUDT), Changsha, China, in 2008.

He is currently a Professor with the College of Aerospace Science and Engineering, NUDT. His current research interests are focused on satellite communication networks, measurement and control technology for microsatellite, onboard computer, spacecraft system modeling, and simulation.

Jianming Guo received the B.S. degree from the University of Science and Technology of China, Hefei, China, in 2015, and the Ph.D. degree from the National University of Defense Technology, Changsha, China, in 2021.

He was a visiting Ph.D. student with Universitat Politècnica de Catalunya, Barcelona, Spain, from 2019 to 2020. He is with the Science and Technology on Complex Aircraft System Simulation Laboratory, Beijing, China. His current research interests include SDN, NFV, and satellite communication networks.

Xianfeng Liu received the B.E. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2014, and the Ph.D. degree from the National University of Defense Technology, Changsha, China, in 2021.

He is currently an Assistant Research Fellow with the Beijing Institute of Tracking and Telecommunications Technology, Beijing, China. His research interests include tracking and data relay technology, TT&C scheduling, and satellite network management.

Xiaoqian Chen received the M.S. and Ph.D. degrees in aerospace engineering from the National University of Defense Technology, Changsha, China, in 1997 and 2001, respectively.

He is currently a Professor and the Dean of the National Institute of Defense Technology Innovation, Beijing, China. His current research interests include spacecraft systems engineering, advanced digital design methods of space systems, and multidisciplinary design optimization.