IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 7 October 2022, accepted 13 October 2022, date of publication 26 October 2022, date of current version 8 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3217491

RESEARCH ARTICLE

Dynamic Beam Scheduling of Multibeam Low Earth Orbit Satellites Based on an Enhanced Artificial Bee Colony Algorithm

DEBIN WEI[1,2], DONGDONG ZHENG[2], CHENGSHENG PAN[1,3], AND LI YANG[1]

[1]School of Automation, Nanjing University of Science and Technology, Nanjing 210094, China
[2]Communication and Network Laboratory, Dalian University, Dalian 116622, China
[3]School of Electronic and Information Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China

Corresponding authors: Debin Wei (weidebin@163.com) and Chengsheng Pan (pancs@sohu.com)

This work was supported by the National Natural Science Foundation of China under Grant 61931004 and Grant U21B2003.

ABSTRACT The application of beam-hopping technology to low earth orbit satellites can effectively achieve flexible allocation and efficient utilization of on-board resources. Considering that the power resources on low earth orbit satellites are limited, the electromagnetic environment is complex and changeable, and the terminal distribution and service requirements are highly dynamic. We established the service model, service priority model and multibeam resource scheduling model under the constraints of beam bandwidth, on-board power, service priorities, etc. To solve the catastrophic problem of a large solution space in the resource scheduling model and to improve the convergence of the algorithm, we propose an enhanced artificial bee colony algorithm. The optimization strategy improves the process of population initialization, solution updates, and search for the global optimal solution. The simulation results show that under the constraints of cochannel interference and on-board resource utilization, the algorithm always converges to the objective function at the fastest speed, which proves that the algorithm has high applicability to the high dynamic characteristics of LEO satellites. In addition, the algorithm can obtain the global optimal solution, and thus, it can ensure the fairness of resource allocation and the effectiveness of service completion.

INDEX TERMS Satellite communication, beam hopping, resource scheduling, enhanced artificial bee colony algorithm.

I. INTRODUCTION

Spectrum resources of terrestrial communication systems are relatively scarce [1]. As a strong complement to terrestrial communication, satellite communication plays an important role in emergency communication, maritime search and rescue, disaster warning and in the military. The service types and needs of satellite terminal users are diverse, and the time and space distributions are very uneven, which requires the satellite to have efficient transmission capacity and resource utilization [2].

Traditional communication satellites mostly use single-beam communication. This static beam enables the satellite system to call allocable resources under the limitation of a single beam and is unable to adjust or rarely adjust the beam capacity to adapt to changing service needs. While using fixed beams in multibeam communication, resources are allocated with fixed capacities over a wide coverage area. In addition, the frequency and resources allocated to each beam are fixed, which causes a lack of flexibility and can easily lead to limited satellite functions and a waste of resources [3]. Therefore, studying the on-board resource allocation scheme that adapts to the needs of users and improves the flexibility of satellite system resource scheduling are particularly important.

Flexible payload (FP) multibeam and beam hopping (BH) technology that was developed on the basis of traditional multibeam has become one of the important technologies in

The associate editor coordinating the review of this manuscript and approving it for publication was Mohammad Tariqul Islam.

satellite communication systems [4], which is the key to the flexible allocation and satellite resource distribution. Many studies have shown the superiority of beam-hopping technology, which provides a basis for the flexible allocation and effective utilization of satellite resources [5], [6]. Multibeam satellites send beam-hopping control commands through the beam-hopping controller to achieve synchronous beam hopping [7]. The main idea of beam hopping is to use time slicing, meaning that at the same time, only part of the beams are in the working state, and in satellite resource scheduling, continuous bursts are changed to time-based bursts. According to service requirements, the dynamic allocation of communication resources is achieved by controlling the spatial orientation, bandwidth, frequency and transmission power of the spaceborne multibeam antenna.

A survey found that research on satellite beam-hopping technology mainly focuses on Geostationary Earth Orbit (GEO) satellites, and there is a lack of research on LEO satellites [8]. GEO satellites have a fixed relative position to the ground area, so the beam-hopping technology of GEO satellites is relatively slow [9]. In contrast, the moving speed of LEO satellites is much higher than that of GEO satellites, and the ground area covered by these satellites and the electromagnetic environment they face rapidly change. At the same time, the small size of LEO satellites leads to very limited on-board resources and improves resource utilization, which need to be more urgent. Moreover, in the face of uneven terminal distribution in different countries and regions and great differences in service requirements of the various terminals, LEO faces more highly dynamic changes in terminal distribution and service requirements than GEO [10], which aggravates the difficulty of LEO resource adjustment. Therefore, designing a reasonable resource scheduling policy to achieve efficient matching between on-board resources and service requirements has become a difficult problem to be solved urgently.

The above resource scheduling problems are usually multiobjective optimization problems. Additionally, heuristic algorithms are widely used in solving multiobjective optimization problems. Examples include Mechanism Multi-Objective Particle Swarm Optimization (MCMOPSO) [11], [12], the multiobjective ant colony optimizer (MOACO) [13], [14], the multiobjective binary artificial bee colony algorithm (MOBABC) [15], etc.

In these algorithms, the artificial bee colony (ABC) algorithm has the advantages of few control parameters, strong robustness and excellent local convergence and optimizability. It has certain advantages in solving the problem of multibeam LEO satellite resource scheduling, but with the increase in the number of ground partitions, the solution space increases dramatically, the search speed becomes slower, and the randomness in the solution update process restricts the real-time performance of the algorithm. Therefore, the application of the ABC algorithm to the multibeam LEO satellite resource scheduling problem still needs to be further studied.

The rest of this paper is organized as follows. We analyze the related works on resources and dynamic beam scheduling of LEO Satellites in Section II. In Section III, we describe the communication model of the LEO satellite coverage area system, service model, and service dynamic priority model in detail and construct two objective functions of satellite resource utilization and resource allocation fairness. Section IV extends the preprocessing method of ground partitioning based on the double-loop learning idea, the process of self-adaptive solution updating and the implementation of a dynamic neighborhood optimization strategy. In Section V, the multibeam satellite resource scheduling algorithm based on the improved ant colony (MB-SRSA-IACO) algorithm [16], dual-population artificial bee colony algorithm (DPABC) [17] and ABC algorithm are used as comparative experiments for simulation, which proves the superiority of the algorithm proposed in this paper. Finally, Section VI concludes the paper.

II. RELATED WORK

To effectively utilize on-board resources, many scholars have conducted research on resource allocation [18].

References [19], [20] studied the forward link of satellites in detail and provided the quantification results of system parameters, such as the satellite antenna gain, interbeam interference, number of beams, service requirements and coverage area on downlink channel capacity. Wang et al. [21] eliminated the interference between beams by controlling the space interval between beams and proposed a clustering beam allocation method based on full frequency reuse, which effectively improves the system's actual throughput. Lei et al. [22] employed deep learning (DL), which is widely used in terrestrial wireless networks, to design the beam pattern of beam hopping, but the complexity of the algorithm also increases. A novel multiobjective reinforcement learning framework has been introduced [23], but this method can only achieve low-complexity dynamic multiobjective resource management. [24] combined wide beam and beam hopping and obtained signaling information from wide beams and assisted spot beams in performing beam hopping on-demand services; however, this process only has better delay performance when the traffic is light. Tian et al. [25] transformed the dynamically moving LEO satellite communication beam deployment problem into a static beam deployment problem in the beam deployment period and proposed a beam scheduling algorithm based on a greedy algorithm. However, the above assumptions ignore the high-speed mobility of LEO satellites. Moreover, researchers [26] regard different sensitivities of terminals in terms of delay, bandwidth and connection time as resource allocation factors and improve the algorithm of the breadth-first-search-based spanning tree (BFST) to realize the dynamic scheduling of information and resources in emergency situations. Shi et al. [27] used a subgradient algorithm to update the dual variables and reallocate the power and bandwidth after power and bandwidth allocation is completed for the first time; in this way, the optimal allocation result is obtained through iterative calculation, and the algorithm improves the system capacity.

Heuristic algorithms play an important role in the study of resource allocation and scheduling schemes. [28] proposed a two-stage annealing algorithm (TPGA) by combining a genetic algorithm and a simulated annealing algorithm to solve the satellite resource scheduling problem. Tian et al. [29] proposed a hierarchical scheduling method for real-time scheduling problems that combines three scheduling steps: prescheduling, coarse scheduling and fine scheduling. Thus, a hierarchical scheduling algorithm based on ant colony optimization is proposed. Song et al. [30] first analyzed the influence of satellite distance on resource scheduling and proposed an algorithm that combines an improved genetic algorithm and a local search method to rapidly improve the quality of the scheduling scheme. Reference [31] first used the heuristic-based optimization method to coarsely search the feasible region space where the optimal solution is most likely to appear. However, due to the diversity of services and the high dynamics of satellites, the time slot preallocation will not be able to meet the real-time and changeable service requirements well. A traffic-aware dynamic resource allocation (TADRA) algorithm based on NSGA-II is applied to solve the resource allocation problem of mobile satellite communication systems [32], but the complexity of the algorithm is high. Reference [33] proposed a multisatellite control resource scheduling problem based on ant colony optimization (MSCRSP–ACO), with the optimization goal of minimizing the workload, and established a complex independent set model to solve the problem of satellite resource scheduling. As a result of the joint observation satellite mission planning problem, Jiang et al. [17] proposed a dual-population artificial bee colony algorithm (DPABC) and a heuristic task scheduling algorithm. Zhuang et al. [34] focused on dynamic relay satellite scheduling and proposed a scheduling algorithm, ABC-TOPSIS, which combines an artificial bee colony (ABC) and a technique for order preference by similarity to an ideal solution (TOPSIS). Liu et al. [35] proposed three swarm optimization algorithms: discrete artificial bee colony (DABC), discrete artificial fish swarm (DAFS), and discrete shuffled frog leaping (DSFL) for fair resource allocation in heterogeneous cloud computing systems. Sun et al. [36] proposed a novel artificial bee colony algorithm with updated quantities (ABC-UQ) of nectar sources for OFDMA resource allocation.

Although the above heuristic algorithms can solve the problem of resource allocation and scheduling in their respective networks, the power resources on low-orbit satellites are limited, the electromagnetic environment is complex and changeable, and the terminal distribution and the service requirements are highly dynamic, so it is necessary to make adaptive improvements when using heuristic algorithms.

To solve the problems of low resource utilization, serious beam interference, high computational complexity, and discrete time slot allocation in existing research, a multibeam LEO satellite dynamic beam scheduling model based on dynamic service requests and on-board resources is established in this paper. The beam spatial isolation and double-loop learning idea preprocesses the ground partition to reduce the population. Moreover, an adaptive solution update strategy to speed up the algorithm convergence is proposed, and a dynamic optimization strategy to optimize the scout bee stage to find the global optimal solution is designed.

III. SYSTEM MODELING

A. COMMUNICATION MODEL OF THE SATELLITE COVERAGE AREA SYSTEM

In multibeam LEO satellite communication systems, the beam hop period (BHP) is the effective coverage time of the LEO satellite for the current terrestrial area, starting from ST and ending at ET. It is only possible for the satellite to serve when the start time of the service is included in the effective coverage time window of the satellite. The beam hopping time slot Ts is also called the beam dwell time, and 128 or 256 components are recommended in the DVB-S2X standard. The j-th time slot $Ts_j \in \{1, 2, 3, \cdots, W\}$ is also the beam hopping slot sequence number (the hop number for short). Fig. 1 shows the working mechanism of the beam hopping period and time slot.

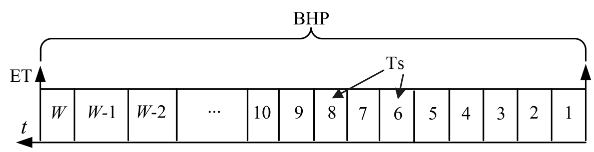


FIGURE 1. Slot model for beam hopping.

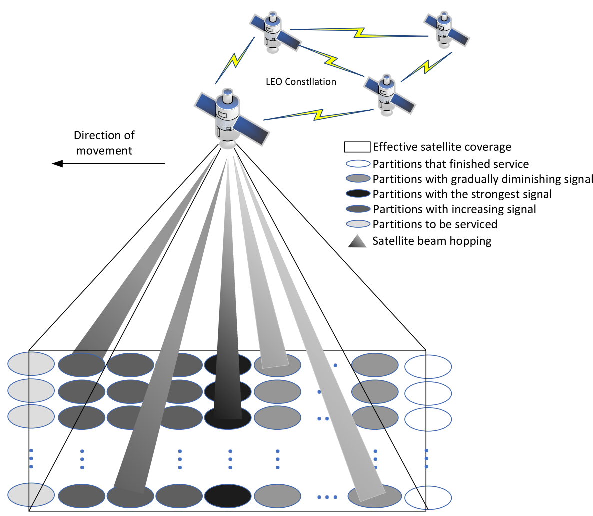


FIGURE 2. LEO satellite beam hopping scheduling model.

For the sake of modeling, the satellite coverage area is simplified to a rectangle, and the rectangular area is divided into $x \times y = M$ ground partitions, as shown in Fig. 2. The satellite beam scheduling model is based on these ground partitions. Within the effective coverage of the satellite, the beamforming antenna generates $N(N \leq M)$ spot beams in the coverage area. The system can generate beam hopping according to the service requested capacity and spatial distribution characteristics of each ground partition, as well as adjust the beam service location to achieve flexible coverage of terrestrial user services.

B. SERVICE MODEL

The total bandwidth of the satellite is B_{total}, and the total power is P_{total}. There are I_m services in the m-th ground partition cell$_m$. To consider the service status of the satellite beam, the service U_m^i within the beam coverage is expressed as follows:

$$\{U_m^i = (\hat{B}_m, \hat{P}_m, C_m^i, V_m^i, Ts_m^i, S_m^i) | \sum_{m=1}^{M} \hat{B}_m \le B_{\text{total}}, \sum_{m=1}^{M} \hat{P}_m \le P_{\text{total}}, V_m^i \in \{1, 2, 3, 4\}, ST \le Ts_m^i \le ET, S_m^i \le W, 1 \le m \le M, 1 \le i \le I_m\}$$

In the ground partition cell$_m$, $\hat{B}_m$ and $\hat{P}_m$ represent the bandwidth and power allocated to cell$_m$, respectively; C_m^i is the capacity that the i-th service needed, and V_m^i is the initial priority of the i-th service. We divide services into five categories: background services, interactive services, streaming media services, voice services, and handover services. These services are ranked in order of priorities from bottom to top. Ts_m^i represents the hop number when the i-th service arrives; S_m^i is the number of time slots theoretically required to complete the i-th service under the given conditions of $\hat{B}_m$ and $\hat{P}_m$.

C. SERVICE DYNAMIC PRIORITY MODEL

We comprehensively consider the importance and urgency of different types of services and propose a definition of dynamic priority. In this paper, the dynamic priority of services (DyPr) is determined by the residual value (RV) of the service and the execution urgency of service (EU) as follows:

$$\begin{aligned} \text{DyPr}(U_m^i) &= \text{RV}(U_m^i) + \text{EU}(U_m^i) \\ &= V_m^i \times \left(\frac{S_m^i - s_m^i}{S_m^i} + \frac{S_m^i - s_m^i}{W - Ts_i} \right) \end{aligned} \tag{1}$$

where s_m^i is the number of time slots that the service has been served and Ts_i represents the current hop number.

In (1), RV can be used to represent the degree of service completion, and the EU indicates how urgently the service needs to be done. It is clear that the greater the amount of residual service, the greater the residual value of the service. The greater the ratio of the number of time slots required by the remaining services to the remaining time slots, the more urgent the service is.

We define $\sum_{i=1}^{I_m} \text{DyPr}(U_m^i)$ as the dynamic priority of the ground partition cell$_m$.

D. MULTIBEAM RESOURCE SCHEDULING MODEL

According to Shannon's theorem, the capacity allocated to the ground partition cell$_m$ is as follows:

$$\hat{C}_m = \hat{B}_m \times \log_2(1 + \text{SINR}_m) \tag{2}$$

where SINR_m is the signal to interference plus the noise ratio of beam m, and the specific relationship is shown in (3).

$$\text{SINR}_m = \frac{A_m \hat{P}_m}{N_0 \hat{B}_m + \sum_{n \in M, n \ne m, x_n^{Ts_j}=1} A_n \hat{P}_n} \tag{3}$$

where A_m is the channel attenuation coefficient in ground partition cell$_m$; N_0 is the power spectral density of additive white Gaussian noise; and $\sum_{n \in M, n \ne m, x_n^{Ts_j}=1} A_n \hat{P}_n$ is the sum of the cochannel interference caused by other beams to the m-th beam. When the beam space interval is greater than or equal to $4 \times R$, the cochannel interference can be ignored. R is the radius of the spot beam.

Based on the above analysis of the system model, we establish the following satellite beam resource scheduling model under the condition that the beam spatial isolation criterion is satisfied.

$$\begin{aligned} &\min \begin{cases} \mathcal{O}_1 : \sum_{m=1}^{M} x_m^{Ts_j} \dfrac{1}{\hat{C}_m} \\ \mathcal{O}_2 : \sum_{m=1}^{M} x_m^{Ts_j} \dfrac{C_m}{\hat{C}_m} \\ \mathcal{O}_3 : \sum_{m=1}^{M} \sum_{i=1}^{I_m} x_m^{Ts_j} \dfrac{1}{\text{DyPr}(U_m^i)} \end{cases} \\ &s.t.\ C_1 : \hat{C}_m \le C_m \\ &\quad C_m = \sum_{i=1}^{I_m} C_m^i \\ &\quad \hat{C}_m = \hat{B}_m \log_2 \left(1 + \frac{A_m \hat{P}_m}{N_0 \hat{B}_m} \right) \\ &\quad C_2 : \sum_{m=1}^{M} x_m^{Ts_j} = N, \quad x_m^{Ts_j} \in \{0, 1\} \\ &\quad C_3 : \text{DyPr}(U_m^i) = V_m^i \times \left(\frac{S_m^i - s_m^i}{S_m^i} + \frac{S_m^i - s_m^i}{W - Ts_i} \right) \\ &\quad 1 \le Ts_i \le W, \quad s_m^i \le S_m^i, S_m^i \ge 1 \\ &\quad C_4 : \sum_{m=1}^{M} \hat{B}_m \le B_{total} \\ &\quad C_5 : \sum_{m=1}^{M} \hat{P}_m \le P_{total} \end{aligned} \tag{4}$$

$\hat{C}_m$ and C_m represent the actual allocated capacity and the requested capacity of the ground partition cell$_m$, respectively. $x_m^{Ts_j} = 1$ represents that the m-th ground partition cell$_m$ is selected by the spot beam in time slot Ts_j; otherwise, it is not selected. $\mathcal{O}_1$ is the satellite resource utilization function, which maximizes the utilization of satellite resources to ensure the full utilization of satellite resources. $\mathcal{O}_2$ is a function that reflects the fairness of resource allocation, which can meet the capacity requirements of the service as much as possible and ensure service satisfaction. $\mathcal{O}_3$ is a function that reflects the dynamic priority of the service, which means that the LEO satellite communication system completes the service of high-priority tasks as much as possible.

Constraint C_1 indicates that the capacity of the beam provided by the satellite is usually not higher than the requested capacity, ensuring maximum utilization of on-board resources and avoiding idle resources. Constraint C_2 indicates that in the same time slot, a beam can only serve one ground partition. Constraint C_3 indicates that the Ts_j that a service is served for the first time should be after the Ts_j that the service applies to access. Additionally, before the effective satellite coverage time window ends, simultaneously, the number of time slots that the current service has been served should be less than or equal to the number of time slots applied for by the service (the number of time slots requested by the service is at least 1). C_4 and C_5 are bandwidth resource constraints and power resource constraints, respectively, which indicate that the bandwidth and power actually allocated to the service should not exceed the satellite on-board bandwidth and power resources.

IV. MULTIBEAM RESOURCE SCHEDULING ALGORITHM FOR SATELLITE COMMUNICATION SYSTEMS

To solve the above resource optimization problem, we commonly use intelligent algorithms, including genetic algorithms, ant colony algorithms, particle swarm algorithms, ABC algorithms and their combinations. The beam hopping pattern based on the traditional ABC algorithm is optimized in this paper. In the ABC algorithm, the employed bee updates the solution according to (5) as follows:

$$x_i' = x_i + \phi_i(x_i - x_k), \quad i = 1, 2, \ldots, \mathrm{SN} \tag{5}$$

where SN is the number of solutions; $x_i = (x_{i_1}, x_{i_2}, \ldots, x_{i_D})$ is an original solution, which is a D-dimensional vector; $k \in \{1, 2, \ldots, \mathrm{SN}\}$ and $k \neq i$; x_i' is the updated solution; and ϕ_i is a random number in the interval $[-1, 1]$.

The onlooker bee stage updates the solution according to (6) as follows:

$$x_{id}' = x_{id}^{\min} + r(x_{id}^{\max} - x_{id}^{\min}) \tag{6}$$

where x_{id}' is the d-th component of the updated solution x_i', $x_{id}^{\min}$ and $x_{id}^{\max}$ are the lower and upper bounds of the d-th component of all solutions, respectively, and r is a random number in the interval [0, 1].

The important factors affecting the convergence speed of the ABC algorithm mainly come from the following aspects:

(1) The selection of the reference solution x_k is random when the bee is in the process of updating the solution, which leads to the uncertainty of the reference standard when the update range is generated.

(2) Randomness is generated by the acceleration factor ϕ_i in the bee's process of updating the solution, which leads to the uncertainty of the update range.

(3) The selection of parameter r is also random when the onlooker bee searches for a new solution in the neighborhood.

So, the ABC algorithm will be improved from three aspects: optimizing the population initialization process through the double loop learning idea, optimizing the original solution update process through the adaptive solution update method, and optimizing the scout bees' jumping out of the local optimal solution process through the dynamic neighborhood optimization strategy. A dynamic beam scheduling method based on the Enhanced Artificial Bee Colony algorithm (DBSM-EABC) is proposed. The specific process is as follows.

A. INITIALIZATION RULES BASED ON DOUBLE-LOOP LEARNING

To use beam hopping technology in beam scheduling of multibeam LEO satellites, it is necessary to select N out of M ground partitions at the beginning of each time slot to provide services. When $M = 20$ and $N = 10$, the size of the solution space is 184756. When N remains unchanged and $M = 40$, the size of the solution space will be hundreds of millions, and the calculation delay will be greatly increased, so the complete traversal of the solution space is unrealistic.

To solve the catastrophic problem of a large solution space, this study proposes a ground partition preprocessing method based on the idea of double-loop learning. According to the service needs of terrestrial terminal users, we calculate the priority index $index_m$ of the ground partition cell_m, and the ground partitions with poor quality are eliminated. The search space of the algorithm is reduced so that the algorithm can converge in a short time. The calculation of $index_m$ is as follows:

$$\mathrm{index}_m = \phi \times \frac{C_m}{C_{\mathrm{toltal}}} + \varphi \times \frac{S_m}{W \times N} + \psi \times \mathrm{DyPr}_m \tag{7}$$

where

$$C_{\mathrm{total}} = \sum\nolimits_m^M C_m,\ S_m = \sum\nolimits_{i=1}^{I_m} S_m^i,$$
$$\mathrm{DyPr}_m = \sum\nolimits_{i=1}^{I_m} \mathrm{DyPr}(U_m^i)$$

A schematic diagram of the preprocessing method for ground partition selection based on double-loop learning is shown in Fig. 3.

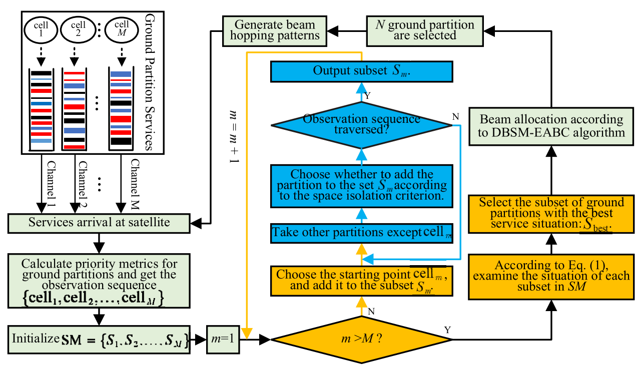


FIGURE 3. Schematic diagram of double-loop learning.

The main idea is to select $L(L > 1.5N)$ waiting service partitions from M ground partitions and construct the solution space of the above optimization problem in these L partitions. In loop 1, one of the M ground partitions $\text{cell}_m, m = 1, 2, \ldots, M$ is selected in turn as the initial selection partition. In loop 2, cell_m is used as the initial selection partition, and the ground partitions that satisfy the spatial isolation conditions are searched among the M ground partitions to be added to the ground partition subset S_m. When Loop 1 ends, M ground partition subsets $S_m, m = 1, 2, \ldots, M$ are obtained. According to (7), the subset of ground partitions with the highest sum of partition priorities is selected as the set of partitions to be served, and the DBSM-EABC algorithm uses this to construct a feasible solution set of the optimization problem.

Table 1 gives a detailed description of the method.

TABLE 1. Ground partition preprocessing algorithm based on double-loop learning.

Input: Collection of ground partitions: $\{\text{cell}_1, \text{cell}_2, \ldots, \text{cell}_M\}$
Output: Preprocessed collection of ground partitions: $\{\text{cell}_1, \ldots, \text{cell}_L\}$
01: Loop 1:
02: To construct M ground partition subsets $S_1, S_2, \ldots, S_M$, initialize each $S_m (1 \le m \le M)$ as an empty set.
03: Take $\text{cell}_m (1 \le m \le M)$ from $\{\text{cell}_1, \text{cell}_2, \ldots, \text{cell}_M\}$ and join subset S_m.
04: Loop 2:
05: Traverse the other elements $\text{cell}_i (1 \le i \le M, i \ne m)$ except cell_m in the queue $\{\text{cell}_1, \text{cell}_2, \ldots, \text{cell}_M\}$ in turn
06: if cell_i and each element in the set S_m meet the spatial isolation criterion.
07: Add cell_i to subset S_m.
08: else if: There are still elements in $\{\text{cell}_1, \text{cell}_2, \ldots, \text{cell}_M\}$ that have not been traversed?
09: return Loop 2.
10: else: end Loop 2.
11: if there is an empty set S_m in the set $S_1, S_2, \ldots, S_M$.
12: return Loop 1.
13: else: end Loop 1.
14: According to formula (7), examine each ground partition subset in $S_1, S_2, \ldots, S_M$, and select the ground partition subset with the highest total partition priority as the set $S_{\text{best}} = \{\text{cell}_1, \text{cell}_2, \ldots, \text{cell}_L\}$ of the partition to be served.

B. ADAPTIVE SOLUTION UPDATE STRATEGY

Considering the local coverage of the fireworks algorithm, the number and the range of its search are used to improve the update strategy of feasible solutions of employed bees and onlooker bees. Dynamically adjusting the number of fireworks explosions can effectively avoid repeated searches, jump out of the local optimal solution, and reflect a good global searchability. Therefore, for the solution update of the employed bee and the onlooker bee, based on the fireworks algorithm, this study carries out research from three aspects: the range of the update solution, the number of the update solution, and the retention of the update solution. Then, the adaptive update solution process is designed to speed up the convergence of the algorithm.

1) UPDATE RANGE OF THE SOLUTION SEARCH

The updated solution search range A_i is related to the fitness of solutions and computed by (8) as follows:

$$A_i = A_{\max} \times \frac{f(x_i)}{\sum_{j=1}^{N}(f_{\max} - f(x_i)) + \varepsilon}, \quad f_{\max} = f_{best} \tag{8}$$

$A_{\max}$ is a constant used to adjust the range of the solution search, $f_{\max}$ is the maximum fitness value in the current solution set, $f(x_i)$ is the fitness of the current solution x_i, and ε is a small constant, which ensures that the denominator is not 0 and is taken as 0.0001. (8) shows that the solution with better fitness has a smaller search range, and its update result will revolve around itself. In contrast, the solution with poor fitness has a larger search range that ensures that a better fitness can be found.

2) UPDATE NUMBER OF THE SOLUTION SEARCH

The number N_i of search results for each solution is determined by its fitness value as follows:

$$N_i = N_{\text{all}} \times \frac{f_{\max} - f(x_i) + \varepsilon}{\sum_{j=1}^{N}(f_{\max} - f(x_i)) + \varepsilon} \tag{9}$$

where N_{all} is the total number of normal results produced by the control N_i solutions. Since the value calculated by (9) may be a decimal, (10) is used to convert the value to an integer.

$$\hat{N}_i = \begin{cases} \text{round}(a \times N_{\text{all}}), & N_i < a \times N_{\text{all}} \\ \text{round}(b \times N_{\text{all}}), & N_i > b \times N_{\text{all}}, a < b < 1 \\ \text{round}(N_i), & a \times N_{\text{all}} \le N_i \le b \times N_{\text{all}}, a < b < 1 \end{cases} \tag{10}$$

where $a, b \in (0, 1)$ and are set as 0.04 and 0.8, respectively, in the simulation.

3) RETENTION POLICY FOR SEARCH RESULTS

It can be seen from the above process that in the process of each solution update, N_i solutions will be generated within the range of the search range A_i. In this paper, Arena's principle is used to select one of the N_i solutions for reservation.

a: THE DOMINATION RELATIONSHIP BETWEEN INDIVIDUALS IN THE DECISION SPACE

X and Y are set as two different individuals in the evolutionary population. If X dominates Y, the following two conditions must be satisfied:

(i) For all subgoals, X is not worse than Y;

(ii) At least one subgoal exists where X makes Y better.

b: CONSTRUCTING PARETO OPTIMAL SOLUTIONS USING ARENA'S PRINCIPLE

In the process of using Arena's principle to retain the solutions, one solution is selected from the N solutions as the ring master, and the ring master is compared with other individuals in the structure set. The loser is eliminated from the game, and the winner becomes the new ring master and continues the round of comparison. After the round is over, the final ring master is the nondominant individual, that is, the updated solution to be retained.

C. DYNAMIC NEIGHBORHOOD OPTIMIZATION STRATEGY

If a solution is not updated in successive Lim generations, it means that the algorithm is in a local optimum dilemma. The DBSM-EABC algorithm jumps out of the local optimum dilemma through the dynamic neighborhood optimization strategy of scout bees.

This strategy is based on the set $\{\text{cell}_1, \text{cell}_2, \ldots, \text{cell}_L\}$ of ground partitions obtained after double-loop learning, comprehensively considers the services requirements of cell_m and the historical experience of cell_m in the limited iteration process, calculates the probability pro_m that the ground partition cell_m is retained, and determines the probability that the partition is reserved of cell_m by roulette. pro_m is given as follows:

$$\text{pro}_m = \eta \times \frac{\text{index}_m}{\sum_{i=1}^{M} \text{index}_m} + (1-\eta) \times \frac{\text{St}}{\text{Stb}} \tag{11}$$

where St is the number of ground partitions cell_m selected during the most recent Lim generation iteration process; Stb represents the number of higher quality solutions generated by using the partition; η is the inertia weight factor, indicating the proportion of cell_m in the calculation of selection probability; and $1-\eta$ represents the weight of the latest historical search experience on the update probability.

Finally, according to the principle of priority with higher retention probability, N ground partitions are selected from the observation scheduling queue as scout bees to update the solution: $\{\text{cell}_1, \text{cell}_2, \ldots, \text{cell}_L\}$.

Table 2 provides a detailed description of the algorithm.

D. DETERMINE THE FITNESS FUNCTION

We use the method of linear weighting to transform the objective functions $\mathcal{O}_1$, $\mathcal{O}_2$, and $\mathcal{O}_3$ into the fitness function of the DBSM-EABC algorithm, which is shown in (12) as follows:

$$f = \lambda_1 f_1 + \lambda_2 f_2 + \lambda_3 f_3 \tag{12}$$

where $\lambda_1 + \lambda_2 + \lambda_3 = 1$. Before the calculation, the three objective functions are normalized, and then the weight of the response is determined according to their importance. In this paper, we first ensure the resource utilization of LEO satellites, and then ensure that resources are allocated as fairly as possible and serve as many high-priority services as possible. Thus, the corresponding weight relationship is $\lambda_1 \succ \lambda_2 \succ \lambda_3$, and $\lambda_1 = 0.5$, $\lambda_2 = 0.3$, $\lambda_3 = 0.2$ is set in our simulation.

In summary, the flow chart of the DBSM-EABC algorithm is shown in Fig. 4.

TABLE 2. Dynamic neighborhood optimization strategy.

Input: Subset of ground partitions selected after double-loop learning: $\{\text{cell}_1, \text{cell}_2, \ldots, \text{cell}_L\}$
Output: updated solution for scout bees: $\{\text{cell}_1, \text{cell}_2, \ldots, \text{cell}_N\}$
01: iterate over the collection $\{\text{cell}_1, \text{cell}_2, \ldots, \text{cell}_L\}$.
02: Loop 1.
03: Take elements $\text{cell}_m, m = 1, 2, \ldots, L$ from $\{\text{cell}_1, \text{cell}_2, \ldots, \text{cell}_L\}$ and insert it into the observation scheduling sequence.
04: Calculate the priority index index_m of cell_m and the probability pro_m of being reserved.
05: According to the roulette rule, it is determined whether cell_m is reserved, and if it is reserved, it is put into the set SN.
06: If There are still elements in the collection $\{\text{cell}_1, \text{cell}_2, \ldots, \text{cell}_L\}$ that have not been traversed.
07: return Loop 1.
08: else: end Loop 1.
09: According to the principle of priority with higher retention probability, N ground partitions are selected from the observation scheduling queue as scout bees to update the solution: $\{\text{cell}_1, \text{cell}_2, \ldots, \text{cell}_N\}$.
10: End

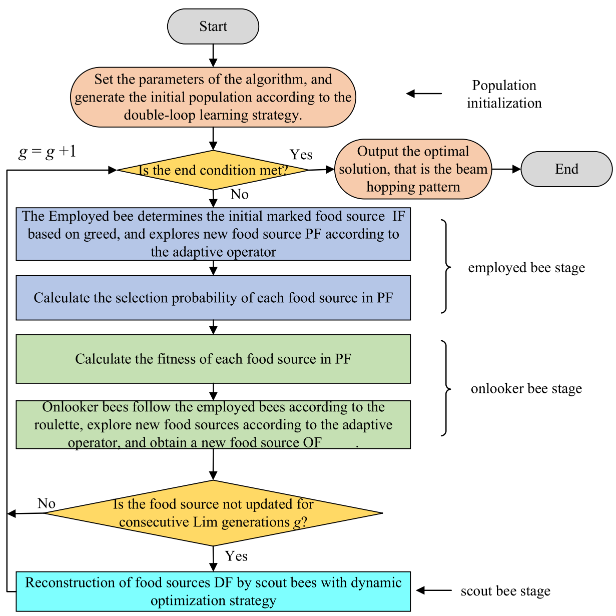


FIGURE 4. Flow chart of the LEO satellite resource scheduling algorithm based on beam hopping.

This research solves the problem of generating beam hopping patterns in LEO satellites based on the EABC. In the population initialization stage, the ground partition that conforms to the space interval is selected through the double-loop learning idea, and the size of the solution space is compressed while the cochannel interference is eliminated. In the employed bee and onlooker bee stage, the search range and quantity of the current solution during updating are determined according to the fitness of the solution, avoiding invalid and repeated searches. In the scout bee stage, we determine the update strategy of the scout bee on the basis of analyzing the fitness of the solution and considering the historical experience of solution updating. Through the above three aspects, the convergence speed of the algorithm is accelerated, and it is further adapted to the high dynamic characteristics of LEO satellites.

V. SIMULATION RESULTS AND ANALYSIS

A. REALIZATION OF THE SIMULATION PROCESS

Based on the Iridium Satellite and Satellite Tool Kit (STK), the coverage time of the satellite is obtained for each ground partition. Then, we establish a simulation system according to the LEO satellite coverage model in Section 3 and implement the beam-hopping scheduling algorithm through MATLAB. In the simulation scenario, the distribution of users in the coverage area follows a normal distribution [8]. The other simulation parameters are shown in Tables 3 and 4.

TABLE 3. System simulation parameters.

Symbol	Category	Value
P_{total}	Total satellite power/W	200
B_{total}	Total satellite bandwidth/MHz	500
H	Satellite orbit altitude/km	780
M	Number of ground partitions/piece	100
N	Number of satellite beams/piece	10
W	Number of time slots for beam scheduling/piece	128
Ts	Time length of a time slot/ms	50
V_m^i	Priority of the services	1,2,3,4,5
C_m^i	Throughput required by the services/bps	$[2,4,8,16,12,24,32]\times 300$

TABLE 4. Parameters of the DBSM-EABC algorithm.

Symbol	Category	Value
$\text{Ite}_{\max}$	The maximum number of iterations of the algorithm	900
$A_{\max}$	Maximum range for adaptive	40
N_{all}	The number of adaptive solution updates	20
Lim	The maximum number of times the solution has not been updated in a row	20
η	Weighting factor for index_m	0.5

B. ANALYSIS OF THE SIMULATION RESULTS

In this set of simulation experiments, we compare the proposed DBSM-EABC algorithm against MB-SRSA-IACO [16], DPABC [17] and the ABC algorithm with respect to convergence, on-board resource utilization, resource allocation fairness, the number of completed services, the sum of service priorities that are completed, etc.

The convergence of the algorithm is an important index to measure the performance of the intelligent algorithm. Better convergence can provide better adaptations to the high dynamic characteristics of LEO satellites. As shown in Fig. 5, the DBSM-EABC algorithm can reach the best fitness in 220 iterations, while the MB-SRSA-IACO, DPABC and ABC algorithms need approximately 470 times, 570 times and 630 times to reach the best fitness, respectively. Compared with the MB-SRSA-IACO, DPABC and ABC algorithms, the DBSM-EABC algorithm proposed in this paper reduces the space for updating solutions through double loop learning and makes the solution update more accurate through an adaptive updating solution strategy, so it speeds up the search for the optimal solution; thus, this algorithm has better convergence.

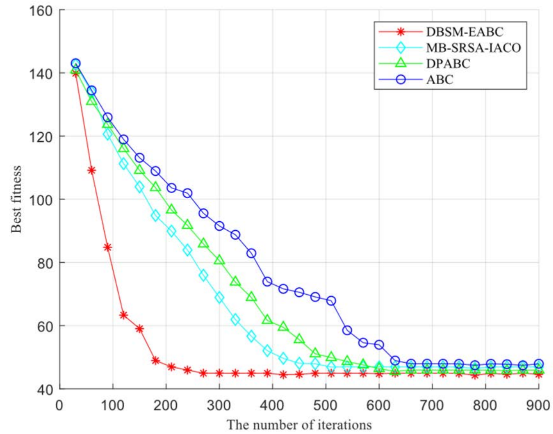


FIGURE 5. The number of iterations and best fitness in algorithms.

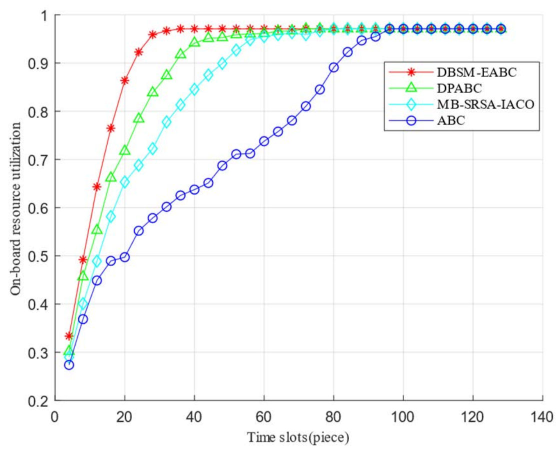


FIGURE 6. On-board resource utilization.

LEO satellite resources are limited, and avoiding idle resources and making full use of on-board resources are important goals of resource scheduling algorithms. Fig. 6 shows the on-board resource utilization of different resource scheduling calculations under the same service requirements. An analysis of Fig. 6 shows that the on-board resource utilization of the DBSM-EABC algorithm can reach 96.8% of the resource utilization in the 33rd time slot, which is much better than the other scheduling algorithms. This is because the double-loop learning idea is used to achieve spatial isolation of beams in the population initialization process, which reduces the interference between beams. At the same time, after the introduction of beam-hopping technology, the ground partitions can be divided more finely so that the resource scheduling algorithm can better match the unevenly distributed terrestrial service requirements.

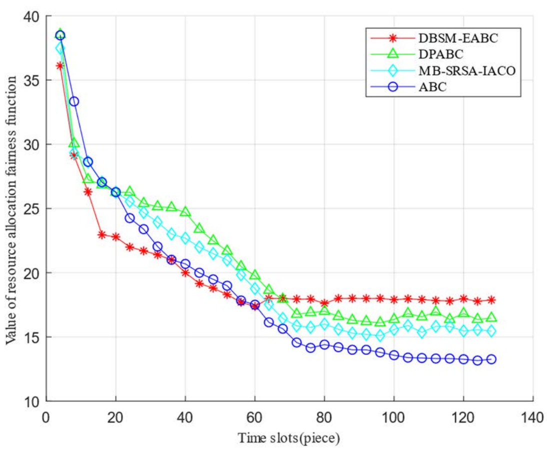


FIGURE 7. Resource allocation fairness.

The fairness of resource allocation is defined by the ratio of the allocated capacity to the requested capacity, which reflects the satisfaction of ground partition services. A fairer resource allocation can reduce unnecessary resource competition. An analysis of Fig. 7 shows that the resource allocation fairness is relatively high in the initial stage of the algorithm because there are few services connected to satellites at the beginning, and the system resources are relatively sufficient, which can better meet the service needs of the ground partition. The service is converted from a new access type to a handover type, the priority of the service is increased, and the remaining capacity of the on-board resources is reduced, which leads to more intense competition for resources. Thus, the fairness of scheduling is reduced. In the simulation, DBSM-EABC keeps the fairness function dynamic and stable at the fastest speed and has the maximum value, which proves the algorithm's superior performance.

The number of remaining services reflects the LEO satellite resource processing capacity. Fig. 8 shows that at the same time, DBSM-EABC comprehensively considers the requested capacity and allocation fairness of the ground partition, which breaks through the space limitation of resource scheduling and seeks to maximize the total benefit so that more services can be handled at the same time. This is an important basis for improving satellite throughput.

In Table 5, P_1 represents the number of timeslots required for the satellite resource utilization to reach 96%, P_2 represents the stable value finally reached by the resource scheduling fairness function, and P_3 represents the average number of services completed in each timeslot. The DBSM-EABC algorithm uses the fewest time slots to achieve 96% utilization of satellite resources. The final stable value of the resource scheduling fairness function of this algorithm is the largest, which is 17.87. Additionally, the average number of services completed in a time slot of this algorithm is also the largest.

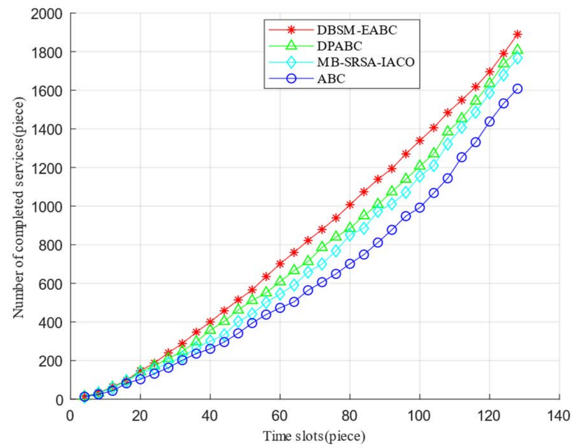


FIGURE 8. The number of completed services varies with time slot.

TABLE 5. Evaluation of simulation experiment results.

	DBSM-EABC	DPABC	MB-SRSA-IACO	ABC
P_1	33	62	76	95
P_2	17.87	16.46	15.35	13.26
P_3	14.76	14.11	13.80	12.56

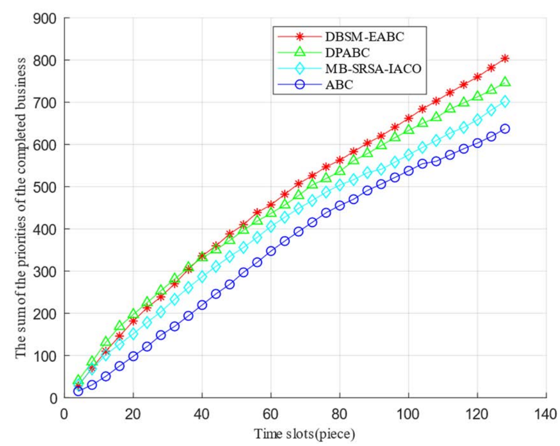


FIGURE 9. The priority of services changes with the time slot.

In Fig. 9, during the initial phase of the satellite service, a small number of services are completed. Over time, the number of completed services increases, increasing the sum of the priorities of the completed services. Since DBSM-EABC can effectively find the global optimal solution and provide services for high-value services, after a scheduling cycle is completed, the algorithm completes the largest number of services, and the corresponding services have the highest priority.

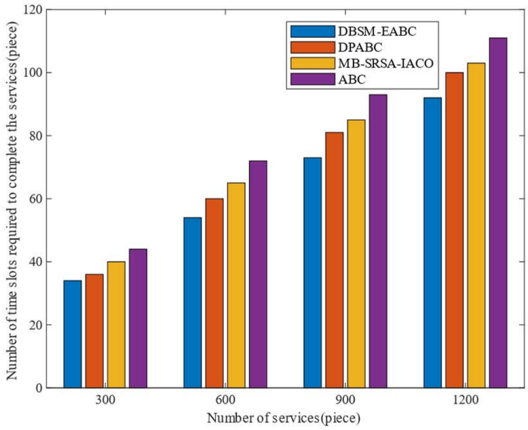


FIGURE 10. The relationship between the number of time slots required to complete the service and the number of services.

Fig. 10 reflects the relationship between the number of time slots required to complete the service and the number of services. As seen in Fig. 10, as the number of services increases, the number of time slots required by each algorithm to complete services increases. However, under different task scales, the DBSM-EABC algorithm requires the least number of time slots to complete services. In general, DBSM-EABC is better than the other algorithms, whether it is the number of completed services within the same number of time slots or the number of time slots consumed to complete the same number of services. This is mainly because the algorithm is optimized for the ABC algorithm in the solution update process and the dynamic neighborhood update process so that the algorithm can select the global optimal solution faster, which proves that DBSM-EABC has good fitness.

VI. CONCLUSION

The introduction of beam-hopping technology into the resource scheduling of LEO satellites provides a new direction for the effective use of on-board resources and meets the needs of ground services as much as possible. In addition, the introduction of beam-hopping technology into the resource scheduling of LEO satellites is difficult for adapting to its highly dynamic characteristics. In this paper, the interbeam interference, limited on-board resources and variable terminal service requirements are comprehensively considered, and a multibeam LEO satellite dynamic beam scheduling model is established. To solve the above dynamic beam scheduling optimization problem, we employ the improved ABC algorithm—DBSM-EABC. In order to speed up the convergence speed of ABC algorithm, an initialization rule based on double loop learning is designed to reduce the size of the solution space, and an adaptive solution update strategy is adopted to make the solution update of the employed bee and the onlooker bee more oriented. A dynamic neighborhood optimization strategy is designed for the local optimal solution of ABC algorithm, so that the scout bee can search the global optimal solution faster. The DBSM-EABC algorithm proposed in this paper is used for beam hopping to generate the beam-hopping pattern optimization method, which provides a feasible solution for the resource scheduling of LEO satellites based on beam hopping and meets the requirements of satellite networks for service needs, efficient use of resources, and key coverage in hotspot areas.

REFERENCES

[1] J. Zhang, X. Zhang, M. A. Imran, B. Evans, Y. Zhang, and W. Wang, "Energy efficient hybrid satellite terrestrial 5G networks with software defined features," *J. Commun. Netw.*, vol. 19, no. 2, pp. 147–161, Apr. 2017.

[2] R. Wang, X. D. Han, C. Wang, X. Zhou, and J. Long, "Resources scheduling and cooperative management of space-based information networks," *J. Commun.*, vol. 38, no. 1, pp. 104–109, Oct. 2017.

[3] A.-L. Jin, W. Song, and W. Zhuang, "Auction-based resource allocation for sharing cloudlets in mobile cloud computing," *IEEE Trans. Emerg. Topics Comput.*, vol. 6, no. 1, pp. 45–57, Mar. 2018.

[4] C. Zhang, G. X. Zhang, and X. Y. Wang, "Design of next generation high throughput satellite communication system based on beam-hopping," *J. Commun.*, vol. 41, no. 7, pp. 59–72, Jul. 2020.

[5] L. He, X.-L. Liu, Y.-W. Chen, L.-N. Xing, and K. Liu, "Hierarchical scheduling for real-time agile satellite task scheduling in a dynamic environment," *Adv. Space Res.*, vol. 63, no. 2, pp. 897–912, Jan. 2019.

[6] T. Rossi, M. De Sanctis, E. Cianca, C. Fragale, M. Ruggieri, and H. Fenech, "Future space-based communications infrastructures based on high throughput satellites and software defined networking," in *Proc. IEEE Int. Symp. Syst. Eng. (ISSE)*, Paris, France, Sep. 2015, pp. 332–337.

[7] A. Kyrgiazos, B. Evans, and P. Thompson, "Smart gateways designs with time switched feeders and beam hopping user links," in *Proc. 8th Adv. Satell. Multimedia Syst. Conf. 14th Signal Process. Space Commun. Workshop (ASMS/SPSC)*, Palma de Mallorca, Spain, Sep. 2016, pp. 1–6.

[8] W. Y. Liu, S. Y. Xia, Q. J. Jiang, G. T. Li, F. Tian, and S. Y. Sun, "Resource scheduling algorithm based on beam hopping in LEO satellite network," *J. Univ. Chin. Acad. Sci.*, vol. 37, no. 6, pp. 805–813, Nov. 2020.

[9] S. Shi, G. Li, Z. Li, H. Zhu, and B. Gao, "Joint power and bandwidth allocation for beam-hopping user downlinks in smart gateway multibeam satellite systems," *Int. J. Distrib. Sensor Netw.*, vol. 13, no. 5, May 2017, Art. no. 155014771770946.

[10] S. Panthi, D. Breynaert, C. McLain, and J. King, "Beam hopping—A flexible satellite communication system for mobility," in *Proc. 35th AIAA Int. Commun. Satell. Syst. Conf.*, Sint-Niklaas, Belgium, Oct. 2017, pp. 1–10.

[11] M.-C. Yuen, S.-C. Ng, and M.-F. Leung, "A competitive mechanism multi-objective particle swarm optimization algorithm and its application to signalized traffic problem," *Cybern. Syst.*, vol. 52, no. 1, pp. 73–104, Oct. 2021.

[12] M.-C. Yuen, S.-C. Ng, and M.-F. Leung, "An improved competitive mechanism based particle swarm optimization algorithm for multi-objective optimization," in *Proc. 10th Int. Conf. Inf. Sci. Technol. (ICIST)*, Bath, U.K., Sep. 2020, pp. 209–218.

[13] J. G. Falcón-Cardona and C. A. C. Coello, "A new indicator-based many-objective ant colony optimizer for continuous search spaces," *Swarm Intell.*, vol. 11, no. 1, pp. 71–100, Feb. 2017.

[14] L. Chen, W.-L. Liu, and J. Zhong, "An efficient multi-objective ant colony optimization for task allocation of heterogeneous unmanned aerial vehicles," *J. Comput. Sci.*, vol. 58, Feb. 2022, Art. no. 101545.

[15] Z. Yilmaz Acar and F. Başçiftçi, "Solving multi-objective resource allocation problem using multi-objective binary artificial bee colony algorithm," *Arabian J. Sci. Eng.*, vol. 46, no. 9, pp. 8535–8547, Sep. 2021.

[16] Y. Z. He, C. Peng, J. H. Yu, and Y. Liu, "Resource scheduling algorithm of satellite communication system for future multi-beam dense networking," *J. Commun.*, vol. 42, no. 4, pp. 109–118, Apr. 2021.

[17] X. Jiang, Y. Song, and L. Xing, "Dual-population artificial bee colony algorithm for joint observation satellite mission planning problem," *IEEE Access*, vol. 10, pp. 28911–28921, 2022.

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

[19] H. Fenech, S. Amos, A. Tomatis, and V. Soumpholphakdy, "High throughput satellite systems: An analytical approach," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 51, no. 1, pp. 192–202, Jan. 2015.

[20] D. Serrano-Velarde, E. Lance, H. Fenech, and G. E. Rodriguez-guisantes, "Novel dimensioning method for high-throughput satellites: Forward link," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 3, pp. 2146–2163, Jul. 2014.

[21] Y. Wang, D. Bian, J. Hu, J. Tang, and C. Wang, "A flexible resource allocation algorithm in full bandwidth beam hopping satellite systems," in *Proc. IEEE 3rd Adv. Inf. Manage., Communicates, Electron. Autom. Control Conf. (IMCEC)*, Chongqing, China, Oct. 2019, pp. 920–927.

[22] L. Lei, E. Lagunas, Y. Yuan, M. G. Kibria, S. Chatzinotas, and B. Ottersten, "Deep learning for beam hopping in multibeam satellite systems," in *Proc. IEEE 91st Veh. Technol. Conf. (VTC-Spring)*, Antwerp, Belgium, May 2020, pp. 1–5.

[23] X. Hu, Y. Zhang, X. Liao, Z. Liu, W. Wang, and F. M. Ghannouchi, "Dynamic beam hopping method based on multi-objective deep reinforcement learning for next generation satellite broadband systems," *IEEE Trans. Broadcast.*, vol. 66, no. 3, pp. 630–646, Sep. 2020.

[24] P. Wang, X. Zhang, S. Zhang, H. Li, and T. Zhang, "Time-expanded graph-based resource allocation over the satellite networks," *IEEE Wireless Commun. Lett.*, vol. 8, no. 2, pp. 360–363, Apr. 2019.

[25] F. Tian, L. Huang, G. Liang, X. Jiang, S. Sun, and J. Ma, "An efficient resource allocation mechanism for beam-hopping based LEO satellite communication system," in *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast. (BMSB)*, Jeju, South Korea, Jun. 2019, pp. 1–5.

[26] F. Wang, D. Jiang, S. Qi, C. Qiao, and L. Shi, "A dynamic resource scheduling scheme in edge computing satellite networks," *Mobile Netw. Appl.*, vol. 26, no. 2, pp. 597–608, Apr. 2020.

[27] Y. Shi, B. N. Zhang, D. S. Guo, and L. Yang, "Joint power and bandwidth allocation algorithm with inter beam interference for multibeam satellite," *Comput. Eng.*, vol. 44, no. 2, pp. 103–106, Feb. 2018.

[28] Z. Waiming, H. Xiaoxuan, X. Wei, and J. Peng, "A two-phase genetic annealing method for integrated earth observation satellite scheduling problems," *Soft Comput.*, vol. 23, no. 1, pp. 181–196, Oct. 2017.

[29] Z. X. Tian, H. T. Tang, Z. G. Wang, D. B. Wang, and J. Cao, "Satellite utility improvement technique based on dynamic on-board command scheduling," *J. Astronaut.*, vol. 35, no. 10, pp. 1105–1113, Oct. 2014.

[30] Y.-J. Song, Z.-S. Zhang, B.-Y. Song, and Y.-W. Chen, "Improved genetic algorithm with local search for satellite range scheduling system and its application in environmental monitoring," *Sustain. Comput., Informat. Syst.*, vol. 21, pp. 19–27, Mar. 2019.

[31] L. Wang, C. Zhang, D. Qu, and G. Zhang, "Resource allocation for beam-hopping user downlinks in multi-beam satellite system," in *Proc. 15th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Jun. 2019, pp. 925–929.

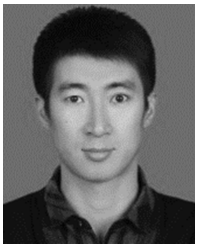
[32] Y. Z. He, Y. Z. Jia, and X. D. Zhong, "A traffic-awareness dynamic resource allocation scheme based on multi-objective optimization in multi-beam mobile satellite communication systems," *Int. J. Distrib. Sensor Netw.*, vol. 13, no. 8, pp. 1–14. Aug. 2017.

[33] Z. Zhang, N. Zhang, and Z. Feng, "Multi-satellite control resource scheduling based on ant colony optimization," *Expert Syst. Appl.*, vol. 41, no. 6, pp. 2816–2823, May 2014.

[34] S. Zhuang, Z. Yin, Z. Wu, and X. Chen, "Dynamic relay satellite scheduling based on ABC-TOPSIS algorithm," *Math. Problems Eng.*, vol. 2016, pp. 1–11, Nov. 2016.

[35] X. Liu, X. Zhang, W. Li, and X. Zhang, "Swarm optimization algorithms applied to multi-resource fair allocation in heterogeneous cloud computing systems," *Computing*, vol. 99, no. 12, pp. 1231–1255, Jun. 2017.

[36] M. Sun, Y. Huang, S. Wang, and Y. Xu, "Novel bee colony optimization with update quantities for OFDMA resource allocation," *Wireless Commun. Mobile Comput.*, vol. 2021, pp. 1–14, Jul. 2021.



DEBIN WEI was born in 1978. He received the M.E. degree from Henan Normal University, Xinxiang, Henan, China, in 2004. He is currently pursuing the Ph.D. degree with the Nanjing University of Science and Technology (NUST), Nanjing, Jiangsu, China. He is currently working as an Associate Professor at Dalian University. His research interests include space-ground integrated network transmission technology, traffic engineering, and network optimization.

DONGDONG ZHENG received the B.S. degree in electronic information science and technology from the Hubei University of Automotive Technology, in 2019. He is currently pursuing the master's degree with the Communication and Network Laboratory, Dalian University. His current research interests include space-ground integrated network transmission technology, beam hopping, and intelligent reception technology.

CHENGSHENG PAN was born in 1962. He received the B.S. and M.S. degrees from the Nanjing University of Science and Technology (NUST), in 1987, and the Ph.D. degree from Northeastern University, in 2001.

From 1988 to 1989, he was a Co-Researcher with the Institute of Mathematics, Chinese Academy of Sciences. Since 1989, he has been an Assistant Professor at Shenyang Ligong University. He is currently a Professor with the Nanjing University of Information Science and Technology and a part-time Ph.D. Tutor with the Nanjing University of Science and Technology. He is the author of three books and more than 150 articles. His research interests include intelligent network traffic theory and key technologies.

LI YANG was born in Heilongjiang, China, in 1982. She received the Ph.D. degree from the Nanjing University of Science and Technology (NUST). She is currently a Professor with the College of Automation, NUST. She was a Visiting Scholar in electrical and computer engineering with the State University of New York at Stony Brook, in 2016. Her research interests include integrated network information transmission technology and wireless communication.