Application of Population Based Incremental Learning Algorithm in Satellite Mission Planning

Yuqing Li[1], Xiaoen Feng[1(✉)], Gang Wang[2], Pengpeng Liu[3], and Chao Zhang[2]

[1] Harbin Institute of Technology, Harbin 150001, China
fengxiaoen0923@163.com
[2] CETC Key Laboratory of Aerospace Information Applications, Shijiazhuang 050081, China
[3] Naval Research Academy, Beijing 10061, China

Abstract. Considering the increasing demand for earth observation missions, aiming at the centralized cooperative mission planning problem of remote sensing satellites, analyzing the constraints in the operation of satellites while considering the load and platform operation, and establishing a reasonable mathematical calculation of satellite missions model. The population incremental learning (PBIL) algorithm is used to solve the satellite mission planning problem. The binary coding method of the traditional PBIL algorithm is improved to the real coding method, and the value matrix correction method is improved. The computational efficiency of PBIL algorithm based on real number coding is verified by numerical examples. The performances of genetic algorithm and PBIL algorithm in solving satellite mission planning problems are compared and analyzed.

Keywords: Satellite mission planning · PBIL algorithm · Earth observation · Remote sensing satellite

1 Introduction

With the development of space technology and the popularity of satellite applications, the number and types of Earth observation satellites are increasing, which are playing a quite important role in the fields of economy, military and people's livelihood. Facing the growing demand for large-scale and diversified tasks of users from all walks of life, satellite mission planning for constellation coordination has become a hot research issue at home and abroad.

Satellite mission planning refers to the process that the satellite control department allocates satellite payloads and ground control resources according to satellite missions

Supported by the Open Fund Project of CETC key laboratory of aerospace information applications (SXX18629T022), and the Key Laboratory Opening Funding of Harbin Institute of Technology of Deep Space Exploration Landing and Return Control Technology (HIT.KLOF. 2018.076, HIT.KLOF. 2018.074), and the pre-research project of equipment development department of China Central Military Commission (JZX7Y20190243001201).

Q. Wu et al. (Eds.): WiSATS 2020, LNICST 358, pp. 220–232, 2021.
https://doi.org/10.1007/978-3-030-69072-4_19

and routine maintenance needs, and satisfies various constraints, with maximizing the benefits of satellite missions during the satellite in-orbit operation [1, 2].

Obviously, satellite mission planning involves not only satellite payload resources, but also ground management resources to ensure their normal operation. It can be seen that there are many constraints in satellite mission planning. Therefore, reasonable and effective modeling methods and efficient mission planning methods are of great significance for describing and solving satellite mission planning problems.

As for the algorithm of the satellite mission planning problem solving, due to the NP characteristics of mission planning problems, most of the current researches use heuristic methods to solve the problem, mainly including tabu search [3], ant colony algorithm [4], genetic algorithm [5, 6], particle swarm algorithm [7, 8] and so on.

These heuristic optimization method is easy to implement and widely used. Arezoo Sarkheyli [3] applied the new tabu search algorithm to solve the problem of low-orbit satellite mission planning by considering the priority of the task and satisfying the time and resource constraints. However, the limiting factors are considered only include coverage rate, data storage capacity and battery capacity in that paper.

Zixuan Zheng et al. [6] used the improved genetic algorithm (GA) to solve the satellite mission planning problem, and proposed a Hybrid Dynamic Mutation (HDM) strategy, which overcomes the early convergence and long calculation time to some extent. But the simulation model used by it did not adequately consider the constraints, and only the constraints of satellite data transmission are considered.

However, in the face of complex large-scale satellite mission planning problems, in addition to the adequate consideration of the constraints of the model, the computational complexity and computational speed of the algorithm are also important aspects to achieving efficient and timely satellite mission planning.

Baluja, the professor of Carnegie Mellon University in the United States, proposed an evolutionary algorithm based on Population Based Incremental Learning (PBIL) [9]. The basic idea of the algorithm is to regard the evolution process as a learning process. The knowledge, also called the probability of learning, which is obtained by learning, to guide the generation of offspring. This probability is the accumulation of information throughout the evolutionary process, and it will be better to guide the resulting offspring (compared to GA's parental genetic recombination and single parent Gaussian variation of EP and ES), which results in faster convergence and better results.

In this paper, the following research work is completed for the satellite collaborative mission planning problem: (1) Under the condition of considering load and platform operation, the constraints of satellite operation are analyzed, a reasonable mathematical model of satellite mission planning is established, the optimization goal of satellite mission planning is proposed. (2) The Population Based Incremental Learning (PBIL) algorithm is used to solve the satellite mission planning problem. Combining with the characteristics of the mission, the binary coding method of the traditional PBIL algorithm is improved to the real coding method, and the correction method of value probability matrix is also improved. The satellite mission planning model based on PBIL algorithm is established and it is verified by numerical examples. (3) The computational efficiency of PBIL algorithm based on real coding is verified. And the performances of genetic algorithm and PBIL algorithm in solving satellite mission planning problems are compared and analyzed by numerical examples.

2 Terminology and Mathematical Statement

2.1 Problem Description and Basic Assumptions

The satellite cooperative mission planning problem can be described as that M satellites cooperatively observe R targets in a planning cycle, so that the objective function is optimal. The final output of the mission planning is mainly the allocation scheme of the observation mission. For a satellite, the distribution result can be expressed as a six-element array as follows:

$$[m, r, ST_{rm}, ET_{rm}, D_{rm}, V_{rm}]$$

Where $S = \{S_1, S_2, \ldots, S_m, \ldots S_M\}$ is the satellite collection, and m is the number of m-th satellite; $T = \{t_1, t_2, \ldots, t_r, \ldots t_R\}$ is the target collection, and r is the number of the r-th target; ST_{rm}, ET_{rm} are the start time and end time of the satellite S_m observing the target t_r;.D_{rm} is the duration of the satellite S_m observing the target t_r; V_{rm} is the benefit of the target t_r observed by the satellite S_m.

Considering the actual satellite system, some reasonable simplifications and basic assumptions for the satellite collaborative mission planning problem are made as follows:

(1) The target is a regional target, and the observation of the target by the satellite requires a certain image scanning time, that is, the observation activity has a certain duration.
(2) The satellite resources involved in mission planning are the satellites with side-swing capability carrying only one spaceborne remote sensor.
(3) The satellite needs to maintain a stable attitude during the execution of the observation mission. After completing the task, it needs to adjust the posture so that the observation task for the next target can be performed smoothly. From the start of the attitude adjustment to the stable attitude of the satellite, the time taken for this process is the satellite attitude adjustment time (also called the attitude maneuver stabilization time).

2.2 Constraints

Task Time Constraint. A satellite can only observe one target at a time, that is, each satellite-borne remote sensor can only perform one observation task at any time. The task start time and end time for each target t_r shall be within its corresponding visible window time range.

Data Storage Constraints. Due to the limited storage space on the satellite, the data size between the two missions of the satellite cannot exceed the capacity of the storage device.

Energy Constraints. Mainly consider two energy conflicts: the discharge depth of the battery for each discharge activity cannot exceed 20%; the satellite must achieve the energy balance of the circle during each illumination ground period, that is, the discharge energy of the battery pack during the grounding can be fully replenished during the subsequent illumination period.

2.3 Objective Function

In this paper, the optimization goal is to maximize the benefits, the objective function Q_1 is as shown in Eq. (1).

$$Q_1 = \max\left(\sum V_i x_i\right), i \in I \tag{1}$$

V_i—The benefit of observing the target t_i.
x_i—Whether the target t_i is selected for observation. If it is, $x_i = 1$, otherwise $x_i = 0$.

3 PBIL Algorithm Based on Real Number Coding

3.1 Encoding

When using the traditional PBIL algorithm for satellite mission planning, binary coding is often used, as shown in Fig. 1.

Each bit of the chromosome represents a time window corresponding to a target, and its value is 0 or 1, which indicates whether the time window is selected to arrange the observation task. The length of the chromosome is the number of visible time windows for all satellites toward all targets.

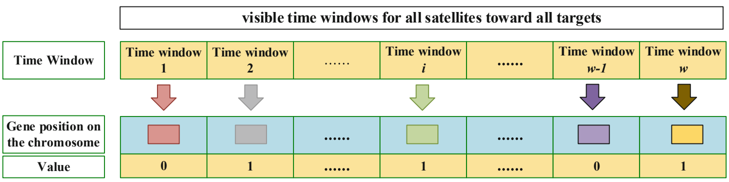


Fig. 1. Binary coding method of PBIL algorithm

However, in the actual satellite mission planning process, the number of satellites in a constellation and observation targets will be large, the planning interval will be long, and the number of visible time windows for the target will be correspondingly larger. If the binary coding method is used, the chromosome will be so long that it will take a long time for each bit of the chromosome to do the constraint collision check, which will result in very low algorithm efficiency.

To enhance the computational efficiency of the algorithm, a real number encoding method is used in this paper, as shown in Fig. 2.

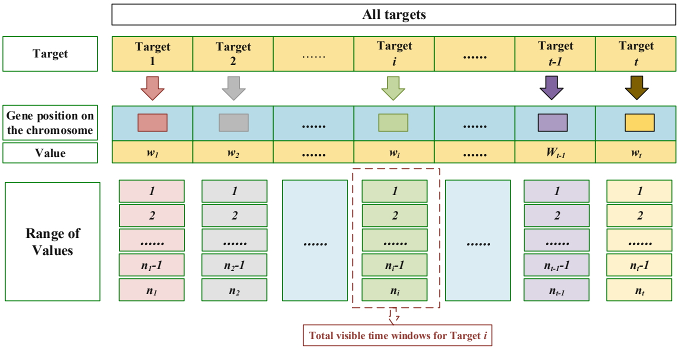


Fig. 2. Real coding method of PBIL algorithm

In the real number coding method, each bit of chromosome represents a target. For one target i in the overall target set I, it corresponds to the i-th bit of the chromosome.

Assuming that the total number of visible time windows for the target i is n_i, and each time window is numbered, corresponding to 1 to n_i.

If the value of the i-th gene position of the chromosome is w_i, one of the natural numbers from 1 to n_i, it means that the time window w_i of target i is selected to complete the observation task of this target. Thus the mapping relationship between chromosomes and problem search space points is established.

3.2 Fitness Function

In the PBIL algorithm, the fitness function represents the direction of evolution. It determines which individual will be chosen to learn and generate the probability of value to guide the generation of offspring. Generally, different fitness functions can be established according to different optimization goals.

In this paper, the objective function in the mathematical model is directly taken as the fitness function.

3.3 Value Probability

In the real coding PBIL algorithm, the probability is in the form of a matrix. The initial probability matrix is shown in Fig. 3.

A column of the probability matrix corresponds to a gene position of a chromosome, in other word, corresponds to an observation target. For the target i in the overall target set I, it has n_i observable time windows in total. This means that the value of the i-th gene position of the chromosome has n_i selections.

The matrix P is the probability of value selections in the algorithm. The j-th row of the i-th column of the probability matrix P represents the probability of selecting the j-th value of the i-th gene position. The probability matrix P is initialized as shown in Eq. (2), which to ensure that each value of each gene position has the same probability at the beginning of evolution.

$$P_{ij} = 1/n_i \tag{2}$$

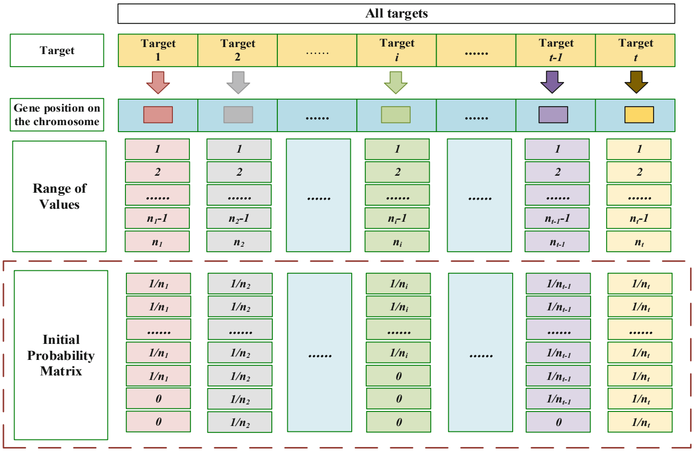


Fig. 3. The initial probability matrix

Among the newly generated populations of each generation, the individual A with the highest fitness value is selected for learning, and the probability matrix is updated, which will guide the population to update.

After the optimal individual A is generated, the update formula of the probability P is as shown in Eq. (3).

$$P_{ij} = P_{ij} + X\,(X \text{ is a constant, and } j = A_i) \tag{3}$$

Obviously, the probability of the j-th value of the i-th gene position is increased. In order to keep the sum of the probabilities of all the values of the i-th gene position to be 1, it is necessary to normalize the probability of them. The normalization formula is as shown in Eq. (4).

$$P_{ij} = P_{ij}/(X + 1)(1 \leq j \leq n_i) \tag{4}$$

Take the probability matrix update of the second-generation as an example. The update of probability matrix is shown in Fig. 4.

As the evolution process progressing, the probability of each value will deviate differently from the initial probability. And the offspring generated according to the probability update will be more likely to be highly adaptable.

3.4 Population Update

The update of the population is under the guidance of the probability matrix taking. With the real number coding, the way of population update is slightly different from that with the binary coding.

Similar to the roulette selection strategy, the probability of n different values of the same gene position is sequentially accumulated to obtain n cumulative probabilities. Then a random number r between 0 and 1 is generated.

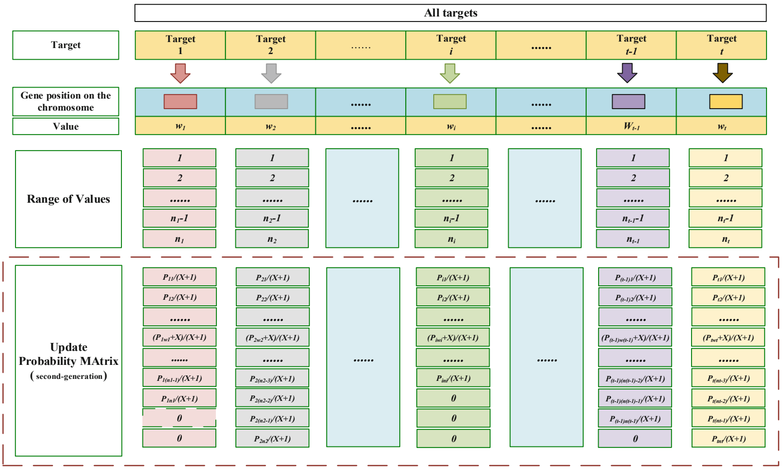


Fig. 4. The probability matrix update of the second-generation

The value corresponding to the smallest cumulative probability of the cumulative probability greater than r is the value of the genetic position. In this way, new individuals generation and population update can be completed.

3.5 End Condition

In this paper, the end condition is the number of evolutions determined by some numerical experiments, in which the population fitness value is not significantly improved in the late stage of evolution and the algorithm stops when the population completes these iterations. Usually the number of iterations will be related to the size of the population. The larger size of population is, the larger number of iterations is, and vice versa.

3.6 Algorithm Steps

The algorithm flow chart shown in Fig. 5.

Step 1. Encode each task according to the real number encoding method, and initialize the probability matrix.

Step 2. According to the above update method, use the probability matrix to guide the generation of the new population.

Step 3. According to the constraints in the mathematical model, each of the genetic positions of each chromosome in the population, that is, each task, is checked for conflict. A task that does not pass the conflict check will be abandoned, that is, the value of this gene position of the chromosome will be set as zero.

Step 4. The fitness value of each individual is calculated to obtain the best individual with the highest fitness value.

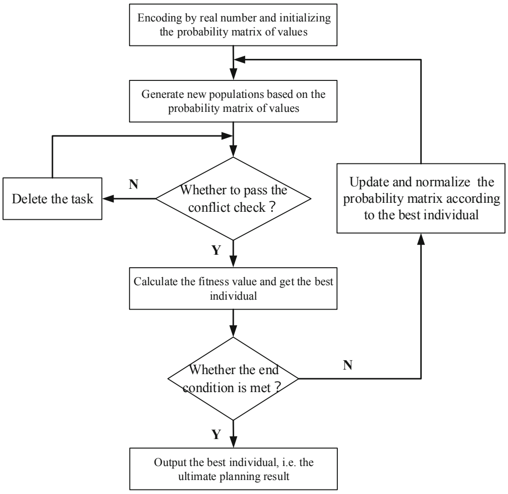


Fig. 5. The algorithm flow chart of PBIL algorithm

Step 5. If the end condition is satisfied, shift to Step 6. Otherwise, according to the best individual generated in the previous step, update and normalize the probability matrix and shift to Step 2.

Step 6. The algorithm ends. Obtain the best individual and output the corresponding mission planning scheme.

4 Numerical Examples and Results

In this paper, the model and algorithm proposed for satellite cooperative mission planning are verified by numerical examples as follows. By designing typical examples and solving them, the results are compared and analyzed, and the performance and efficiency of the PBIL algorithm for solving the satellite cooperative mission planning problem are verified.

4.1 Simulation Scenario

(1) **The satellites**

In this paper, the number of satellites is set to 10, and remote sensing satellite models S1 to S10 are established in STK and the distribution of the satellites in STK is shown in Fig. 6.

(2) **The targets**

A number of observation target points are randomly established globally using the MATLAB program and randomly assigned to each target a benefit value. The distribution of the targets in STK is shown in Fig. 7.

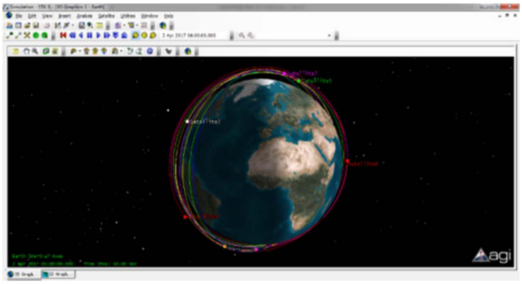


Fig. 6. The distribution of the satellites in STK

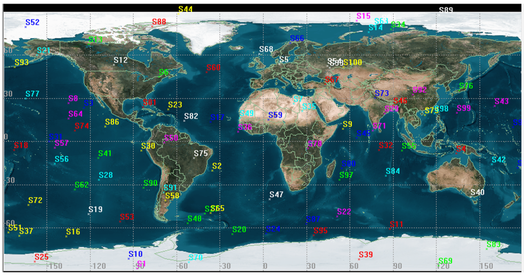


Fig. 7. The distribution of the targets in STK

(3) **The ground stations**

After the targets are observed by the satellites and the information is obtained, the data will be temporarily stored in the onboard device. When communication conditions permit, the data will be passed back to the ground station. In this paper, 12 ground stations are set up to simulate the completion of digital missions. The distribution of the ground stations in STK is shown in Fig. 8.

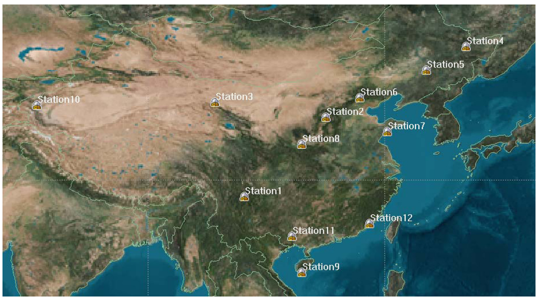


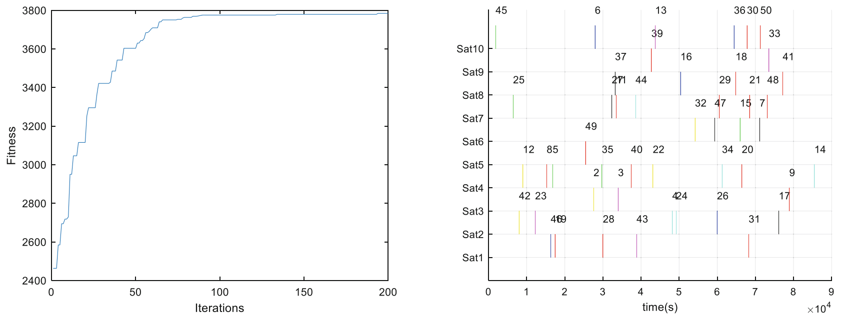
Fig. 8. The distribution of the ground stations in STK

4.2 Typical Results of Satellite Mission Planning Based on PBIL Algorithm

The average results of the 10 experiments are shown in Table 1, and one of the typical results is shown in Fig. 9.

Table 1. The statistical results of the comparison experiments

Simulation scene settings		Average optimal fitness value	
Objective function	Number of targets	Average optimal fitness value	Average running time (s)
$Q_1 = \max\left(\sum V_i x_i\right), i \in I$	50	7436	7046



a) Evolutionary curve b) Mission planning Gantt chart

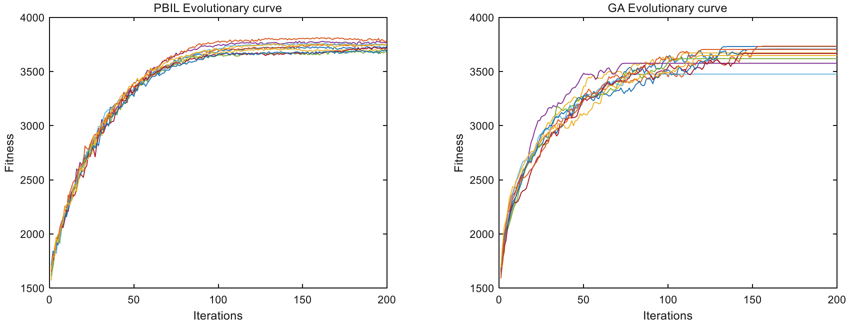
Fig. 9. The typical simulation results of PBIL algorithm

As shown in the Table and figure above, it can be seen that PBIL algorithm can effectively solve multi-satellite mission planning problem. As the evolution curve shown in Fig. 9a), it converges around generation 80 and has a high computational efficiency.

4.3 Analysis of Algorithm Performance

According to the optimization goal in Sect. 2.3, the comparison experiments of genetic algorithm and PBIL algorithm for solving satellite mission planning are respectively carried out. The Evolutionary curves of 10 consecutive simulation experiments for 50 targets are shown in Fig. 12, and the statistical results of computing power for different target quantities are shown in Table 2 and Fig. 13. All the data as follows are the average of the results of 10 consecutive simulation experiments.

As can be seen from the figure above, under the same conditions to solve the satellite mission planning problem, the PBIL algorithm converges rapidly around iteration 80, while the GA algorithm converges after iteration 100, indicating that the PBIL algorithm has faster computational performance than the GA algorithm. At the same time, comprehensive analysis of the evolution curve of 10 simulation experiments shows that the



a) PBIL b) GA

Fig. 12. The evolutionary curves of the comparison experiments

evolution curves obtained by PBIL algorithm has small fluctuation, while that obtained by GA algorithm has significant fluctuation and difference, indicating that PBIL algorithm is also better than GA algorithm in terms of computational stability.

Table 2. The statistical results of the comparison experiments

Number of Targets	Average running time (s)		Average optimal fitness value	
	PBIL real encoding	GA real encoding	PBIL real encoding	GA real encoding
25	550.2	1340.6	2015	1993
50	857.2	2113.5	3779	3635
75	1135.3	3197.2	5623	5446
100	1563.2	3718.7	7436	7046

As shown in the table and the figure above, through the quantitative comparative analysis of calculation running time and optimal fitness value, it can be seen that the running time of PBIL algorithm is obviously much shorter than that of genetic algorithm, where the computational efficiency is almost doubled, and most of the optimal fitness values are slightly higher than that of genetic algorithm.

Therefore, considering algorithm stability, computational efficiency and optimization quality, etc., PBIL algorithm is superior to genetic algorithm in solving satellite mission planning problem to some extent.

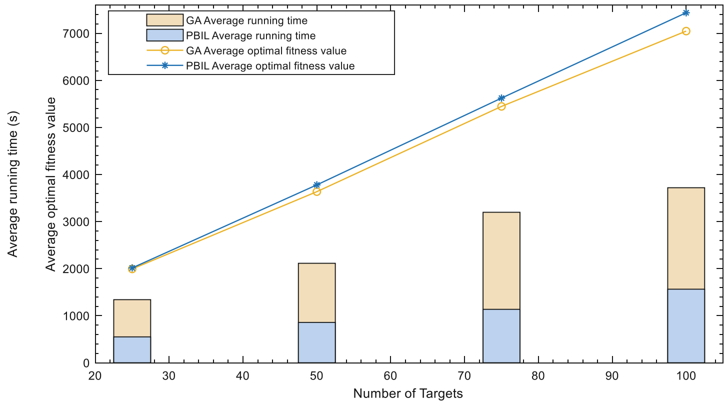


Fig. 13. The statistical results of the comparison experiments

5 Conclusions

In this paper, a quantity of theoretically analysis and numerical experiments have been done to find a better method to solve the satellite collaborative mission planning problem.

First of all, the constraints of satellite operation are analyzed and the abstract practical problems are transformed into mathematical problems. The mathematical model of satellite mission planning is established and the optimization goal of satellite mission planning proposed.

Secondly, the PBIL algorithm based on real coding is proposed for solving satellite mission planning problem. The key point of the algorithm is the real coding method and the update of the probability matrix, which is different from the traditional binary coding method. The PBIL algorithm is implemented, and the feasibility and effectiveness of the algorithm for solving satellite mission planning problems are verified by numerical experiments.

Furthermore, the performances of PBIL algorithm and genetic algorithm are compared for the same satellite mission planning scenario. According to the result data of numerical experiments, it can be seen that when solving the satellite mission planning problem, in terms of the algorithm efficiency, solution quality, and task completion rate, the PBIL algorithm based on real coding is superior to the genetic algorithm in the same situation.

References

1. Karapetyan, D., Minic, S.M., Malladi, K.T.: Satellite downlink scheduling problem: a case study. Omega **53**, 115–123 (2015)
2. Wu, K., Zhang, D.X., Chen, Z.H., et al.: Multi-type multi-objective imaging scheduling method based on improved NSGA-III for satellite formation system. Adv. Space Res. **63**(8), 2551–2565 (2019)
3. Sarkheyli, A.: Using an effective tabu search in interactive resources scheduling problem for LEO satellites missions. Aerosp. Sci. Technol. **29**, 287–295 (2013)

4. De, N.K.F., Goncalves, V.F.M.: Planning on-board satel-lites for the goal-based operations for space missions. IEEE Latin Am. Trans. **11**(4), 1110–1120 (2013)
5. Li, Y., Wang, R., Xu, M.: An improved genetic algorithm for a class of multi-resource range scheduling problem. J. Astronaut. **33**(1), 85–90 (2012)
6. Zheng, Z., Guo, J., Gill, E.: Swarm satellite mission scheduling & planning using Hybrid Dynamic Mutation Genetic Algorithm. Acta Astronaut. **137**, 243–253 (2017)
7. Coello, C.A.C., Pulido, G.T., Lechuga, M.S.: Handling multiple objectives with particle swarm optimization. IEEE Trans. Evol. Comput. **8**(3), 256–279 (2004)
8. Cheng, M., Qian, Q., Ni, Z., et al.: Co-evolutionary particle swarm optimization for multitasking. Pattern Recogn. Artif. Intell. **31**(4), 322–334 (2018)
9. Baluja, S.: Genetic algorithms and explicit search statistics. In: IEEE Advances in Neural Information Processing System, pp. 319–325. MIT Press, Cambridge (1996)