Journal of Computers Vol. 34 No. 1, February 2023, pp. 225-238
doi: 10.53106/199115992023023401017

Multi-satellite Mission Planning Algorithm Based on Preemptive Priority Model

Hai-Ping Gan[1,3], Lin Cao[1,2,3], Pei-Ran Song[1,3*], Xiao-Peng Cao[3],
Bing-Nan Du[3], Kang-Ning Du[2,3], Ya-Nan Guo[3]

[1] School of Information and Communication Engineering, Beijing Information Science and Technology University, Beijing 100101, China
{destinygan, charlin26}@163.com, peiransong@bistu.edu.cn
[2] Key Laboratory of the Ministry of Education for Optoelectronic Measurement Technology and Instrument, Beijing Information Science and Technology University, Beijing, 100101, China
kangningdu@bistu.edu.cn
[3] Key Laboratory of Information and Communication Systems, Ministry of Information Industry, Beijing Information Science and Technology University, Beijing, 100101, China
1752761373@qq.com, 17302287026@163.com, yananguo@bistu.edu.cn

Received 1 January 2023; Revised 18 January 2023; Accepted 19 January 2023

Abstract. Aiming at the problems of high time-consuming and insufficient guarantee of high-priority task completion in the large-scale multi-satellite multi-observation mission planning application, a multi-satellite multi-mission planning model based on preemptive priority model is proposed. Combined with the whole-neighborhood greedy search strategy, an improved tabu search algorithm is designed, taking the combination of satellite observation windows as the decision variables. The designed algorithm realizes an efficient solver to the proposed mission planning model. The simulation results show that the average solving time of the algorithm is 144s under the problem scale of 100 satellites and 3000 tasks. Compared with the existing algorithm, the calculation time is shortened by at least 75% under the premise that the total number of planning tasks is comparable, which is more in line with the high time efficiency requirements for satellite mission planning in engineering applications.

Keywords: preemptive priority, whole-neighborhood tabu search, mission planning model, remote sensing satellites

1 Introduction

Remote sensing satellites are the spacecraft for observing the Earth and its atmosphere, which have great significance in the fields of comprehensive planning of national land resources, national defense security and modern warfare [1-3]. With the development and evolution of aerospace technology, the amount of remote sensing satellites in orbit is increasing as time goes by, the application fields are more diversified, and the demand for earth observation is increasing dramatically. How to efficiently use the limited Earth observation resources in a large-scale multi-satellite mission planning situation has become an urgent problem in satellite applications [4-6].

Remote sensing satellite mission planning is the process of allocating limited satellite resources to different missions and generating observation planning schemes. The generated observation schemes are required to satisfy the satellite constraints and maximize the use of satellite resources to achieve the user requirements. In practical engineering applications, observation tasks usually contain priority attributes, which is the degree of urgency to schedule the execution of that task. Therefore, the priority factor should be considered when modeling the satellite mission planning problem. Researchers have carried out a lot of work on this, including two main types of modeling approaches. The first class of methods uses hierarchical planning models [7-9], which divides tasks according to priority, decomposes the original problem into multiple sub-problems, and plans them in order based on priority from highest to lowest, but this leads to a fixed occupation of satellite resources by high-priority tasks, and the allocation cannot be flexibly adjusted based on the global situation, thus resulting in a waste of satellite resources. Another type of approach uses a quantitative priority model, which quantifies task priorities into specific values to represent the benefits of task completion, and transforms the problem into a priority-free task planning problem by means of linear weighting [10-12]. Since the linear weighting method allows a small set of high-priority tasks

* Corresponding Author

to be dropped to complete more low-priority tasks without reducing overall revenue, it is difficult to ensure that high-priority tasks are prioritized [13].

In addition, most of the existing modeling approaches are based on the assumption that a satellite can only perform one mission in one visible time window, however, in practical engineering applications, one visible time window of a satellite may cover multiple observation missions; one observation mission may require multiple windows of multiple satellites to jointly complete the observation, as shown in Fig. 1. Most current studies do not consider such scenarios simultaneously.

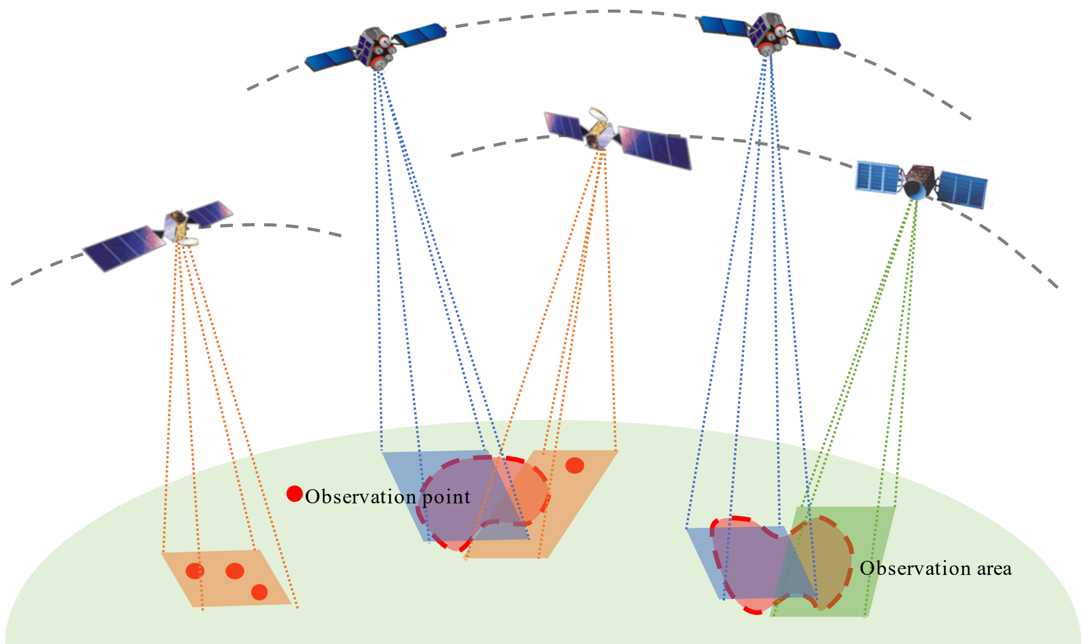

Fig. 1. Diagram of the coverage relationship between the visible window of the satellite and the observation mission

Satellite mission planning problems have been proven to be NP-hard problems [14], requiring a huge amount of computation to find the optimal solution to this kind of problems. Moreover, the increase in the dimensionality of the search space as the problem size increases leads to a sharp increase in the model solving time [15]. Heuristic algorithms are widely used for solving satellite mission planning problems owing to their excellent complex state-space seeking capability [16-21]. Among them, the tabu search algorithm has the characteristic of fast optimization and is one of the common heuristics used in recent years in the field of mission planning [22-24]. However, the tabu search algorithm usually searches only in a small neighborhood during iterative search, so the optimization speed of the tabu search algorithm decreases significantly with the proliferation of mission size in practical applications. He et al. [25] proposed a large-neighborhood local search algorithm to expand the local search by constructing the search neighborhood through constant destruction and repair operations, and experimentally proved that their algorithm can reduce the optimization time when dealing with larger task-scale problems, but still cannot fulfill the demand of large-scale and high timeliness in practical engineering applications.

To address the above problems, we investigate the modeling and optimization algorithms of large-scale multi-satellite mission planning problems, establish a preemptive priority-based mission planning model, and propose an improved tabu search algorithm based on a whole-neighborhood greedy search strategy, which effectively improves the efficiency of solving large-scale multi-satellite multi-task planning problems. The specific work is as follows:

1) Using the concept of window combination, form a relationship mapping between window combination and observation tasks and time windows to achieve a unified description of different types of task observations.

2) Construct a vectorized task priority optimization objective and establish a preemptive priority task planning model.

3) Propose a tabu search algorithm based on the whole-neighborhood greedy search strategy, which effectively improves the efficiency of solving large-scale multi-satellite mission planning problems by rapidly maintaining the full-order gain list of window combinations.

2 Problem Model Construction

The following assumptions are made for the multi-satellite mission planning problem:

1) Communication resources are sufficient and the data transmission process is not considered.

2) The visible time windows of all satellites to the mission have been discretized, and the discretization interval is the shortest duration of mission observation. In the latter part, all discretized satellite visible time windows are collectively referred to as observation windows

3) The observation windows to be planned all satisfy the imaging resolution requirements needed for the observation missions they cover.

The observation mission can contain point targets, area targets and hybrid targets. Assume that the index set of the mission to be observed is $\mathcal{T} = \{1, \cdots, N_T\}$; the index set of the available satellites is $\mathcal{S} = \{1, \cdots, N_S\}$; the index set of the observation windows of the satellites to the mission is $\mathcal{W} = \{1, \cdots, N_W\}$. Establish the relationship matrix between the visible windows and the satellites as $\mathbf{A} = \left(a_{js}\right)_{j\in\mathcal{W},\, s\in\mathcal{S}}$, where

$$a_{js} = \begin{cases} 1 & \text{Observation window } j \text{ belongs to satellite } s \\ 0 & \text{Otherwise} \end{cases}. \tag{1}$$

Since the observation task may require the joint coverage of multiple observation windows, the concept of window combination is used, where each window combination contains one or more observation windows and one or more tasks can be observed simultaneously. Let the set of window combinations be $\mathcal{X} = \{1, \cdots, N_X\}$, and establish the inclusion relation matrix of window combinations and observation windows as $\mathbf{B} = \left(b_{gj}\right)_{g\in\mathcal{X}, j\in\mathcal{W}}$, where

$$b_{gj} = \begin{cases} 1 & \text{Window combination } g \text{ contains window } j \\ 0 & \text{Otherwise} \end{cases} \tag{2}$$

Build a matrix of coverage relations of window combinations for observation tasks as $\mathbf{C} = \left(c_{gi}\right)_{g\in\mathcal{X}, i\in\mathcal{T}}$, where

$$c_{gi} = \begin{cases} 1 & \text{Window combination } g \text{ covers task } i \\ 0 & \text{Otherwise} \end{cases} \tag{3}$$

Define the decision variables: $x_g \in \{0,1\}$, $g = 1, \cdots, N_X$, when $x_g = 1$ indicates that window combination g is used and conversely not used; $w_j \in \{0,1\}$, $j = 1, \cdots, N_W$, when $w_j = 1$ indicates that window j is used and conversely not used; $t_i \in \{0,1\}$, $i = 1, \cdots, N_T$, when $t_i = 1$ indicates that task i has scheduled observations and conversely not completed observations.

2.1 Objective Function

Each observation task contains a priority attribute. Suppose the set of priorities of the observed tasks is $\mathcal{P} = \{1, \cdots, K\}$, where the smaller the value, the higher the corresponding task priority. Let the priority of the i-th task be p_i . For task i , let its priority be p_i and define the vector $\mathbf{o}_i \in \mathbb{R}^K$ as one-hot encoding of task priority.

$$o_i^k = \begin{cases} 1, & \text{if } k = p_i \\ 0, & \text{if } k \neq p_i \end{cases}, \tag{4}$$

where o_i^k denotes the kth component of $\mathbf{o}_i$. Then we have the optimization objective function:

$$\mathbf{f} = \sum_{i=1}^{N_T} t_i \mathbf{o}_i , \tag{5}$$

where the vector $\mathbf{f} \in \mathbb{R}^K$ represents the number of planned tasks of each priority level and the kth component of $\mathbf{f}$ represents the number of tasks that have been planned with priority k.

We consider the preemptive priority model, which means that the number of completed high priority tasks cannot be affected by low priority tasks, and even few high priority tasks cannot be sacrificed to complete more low priority tasks. Define the dictionary order on $\mathbb{R}^K$: $\mathbf{f}_1 \leq \mathbf{f}_2$ when and only when the previous n components of $\mathbf{f}_1$ and $\mathbf{f}_2$ are the same and $\mathbf{f}_1$ is smaller in the next components or $n = K$. Maximizing $\mathbf{f}$ according to the above dictionary order maximizes the preemptive priority.

2.2 Constraints

Satellite Storage Constraint and Energy Constraint. Eq. (6) is the satellite storage constraint, where m_{js} is the storage capacity occupied by satellite s to acquire observations on window j, and M_s is the maximum storage capacity of satellite s. Eq. (7) is the satellite energy constraint, where e_{js} is the energy consumed by the satellite s to make observations using the window j, and E_s is the maximum energy supported by the satellite s.

$$\sum_{j=1}^{N_W} m_{js} w_j a_{js} \leq M_s, \forall s \in \mathcal{S} . \tag{6}$$

$$\sum_{j=1}^{N_W} e_{js} w_j a_{js} \leq E_s, \forall s \in \mathcal{S} . \tag{7}$$

Window Conflict Constraint. When the same satellite completes two consecutive observations, the start time of the next observation must be longer than or equal to the sum of the completion time of the current observation and the transition time. Assume that the start and end moments of the observation window j of satellite s are tws_j and twe_j, respectively; d_{ju} is the entire transition time required for satellite s to adjust from the observation state of window j to the observation state of window u.

$$h_{ju} = tws_j - twe_u - d_{ju} . \tag{8}$$

It represents the time difference between two consecutive mission observation windows of satellite s and the transition time of the satellite observation state, which is the difference between the start time of window j and the end time of window u minus the transition time for satellite ss to adjust from window j to window u. If $h_{ju} < 0$, it is unable to continue observing with window j after observing by using window u. Define the continuous window time conflict indicator:

$$z_{ju} = \sum_s a_{js} a_{us} \mathbf{1}\left(h_{ju} < 0\right) \mathbf{1}\left(h_{uj} < 0\right) . \tag{9}$$

$$\mathbf{1}(cond) = \begin{cases} 1, & \text{Meet the condition } cond \\ 0, & \text{Fail to meet the condition } cond \end{cases} . \quad \textbf{(10)}$$

Where Eq. (10) is the indicator function. As $z_{ju} = 1$, window j and window u will conflict because of too close or overlapping. As a result, the window conflict constraint is shown below. It indicates that the total number of conflicting window pairs allowed in the satellite mission planning scheme is 0.

$$\sum_{j=1}^{N_W-1} \sum_{u=j+1}^{N_W} w_j w_u z_{ju} = 0 . \quad \textbf{(11)}$$

Decision Variable.

$$w_j \geq x_g b_{gj}, \quad \forall g \in \mathcal{X}, \; j \in \mathcal{W} . \quad \textbf{(12)}$$

$$w_j \leq \sum_{g\in} x_g b_{gj}, \quad \forall j \in \quad . \quad \textbf{(13)}$$

$$t_i \geq x_g c_{gi}, \quad \forall g \in \mathcal{X}, \; i \in \mathcal{T} . \quad \textbf{(14)}$$

$$t_i \leq \sum_{g\in\mathcal{X}} x_g c_{gi} \quad \forall i \in \mathcal{T} . \quad \textbf{(15)}$$

Where Eq. (12) and (13) show that a window is selected when and only when it is covered by one or more selected window combinations, and Eq. (14) and (15) indicate that a mission can be planned for observation when and only when it is covered by one or more selected window combinations.

In summary, the multi-satellite mission planning model is as follows:

$$\begin{aligned}
\underset{\mathbf{x},\mathbf{w},\mathbf{t}}{\text{maximize}} \quad & \sum_{i=1}^{N_T} t_i \mathbf{o}_i \\
\text{subject to} \quad & \sum_{j=1}^{N_W} m_{js} w_j a_{js} \leq M_s, \quad \forall s \in \mathcal{S} \\
& \sum_{j=1}^{N_W} e_{js} w_j a_{js} \leq E_s, \quad \forall s \in \mathcal{S} \\
& \sum_{j=1}^{N_W-1} \sum_{u=j+1}^{N_W} w_j w_u z_{ju} = 0 \\
& w_j \geq x_g b_{gj}, \quad \forall g \in \mathcal{X}, \; \forall j \in \mathcal{W} \\
& w_j \leq \sum_{g\in\mathcal{X}} x_g b_{gj}, \quad \forall j \in \mathcal{W} \\
& t_i \geq x_g c_{gi}, \quad \forall g \in \mathcal{X}, \; \forall i \in \mathcal{T} \\
& t_i \leq \sum_{g\in\mathcal{X}} x_g c_{gi}, \quad \forall i \in \mathcal{T} \\
& w_j \in \{0,1\}, \quad \forall j \in \mathcal{W} \\
& t_i \in \{0,1\}, \quad \forall i \in \mathcal{T} \\
& x_g \in \{0,1\}, \quad \forall g \in \mathcal{X}
\end{aligned}$$

Where $\mathbf{x}=\left[x_1,\cdots,x_{N_X}\right]^T$ denotes the window combination decision variable; $\mathbf{w}=\left[w_1,\cdots,w_{N_W}\right]^T$ denotes the window decision variable; and $\mathbf{t}=\left[t_1,\cdots,t_{N_T}\right]^T$ denotes the task planning decision variable.

3 Whole-Neighborhood Tabu Search Algorithm

3.1 Status Encoding Objective Gain

The usage status of each window combination is constituted as a N_X dimensional 0-1 vector $\mathbf{x}=\left[x_1,\cdots,x_{N_X}\right]^T$, where each position corresponds to a decision variable of a window combination. The neighborhood is constructed by inserting new window combinations or deleting selected window combinations in the list of selected window combinations. Since each step changes the state of at most one window combination, the neighborhood contains at most N_X neighboring states.

In the mission planning model, the decision variables $\mathbf{w}$ and $\mathbf{t}$ can be determined by $\mathbf{x}$; from Eq. (12)-(15), the equations can be obtained as follows:

$$w_j=\max_{g\in\mathcal{X}}\{x_g b_{gj}\},\quad \forall j\in\mathcal{W}. \tag{16}$$

$$t_i=\max_{g\in\mathcal{X}}\{x_g c_{gi}\},\quad \forall i\in\mathcal{T}. \tag{17}$$

Therefore, the state only needs to be encoded using $\mathbf{x}$. It is sufficient to update $\mathbf{w}$ and $\mathbf{t}$ according to Eq. (16) and (17).

3.2 Objective Gain

Based on the proposed preemptive priority objective function, the priority factor of the tasks covered by the window combination is mainly considered in decision making. Define $\Delta\mathbf{f}_g$ as the value of the change in the objective function brought about by operating the window combination *g*, which is calculated as follows.

$$\Delta\mathbf{f}_g=\mathbf{f}_g^{last}-\mathbf{f}_g^{pre}, g\in\mathcal{X}. \tag{18}$$

Where $\mathbf{f}_g^{pre}$ denotes the objective function value before operating on the window combination g, and $\mathbf{f}_g^{last}$ denotes the objective function value of the neighborhood solution generated after manipulating the window combination g. In the subsequent search strategy, the window combination g to be performed is searched with the objective of maximizing $\Delta\mathbf{f}_g$ at each step of the procedure.

3.3 Search Strategy

In the traditional tabu search algorithm, the neighborhood search is achieved by setting the neighborhood size and generating the candidate neighborhood solution set by randomly changing the state encoding at each iteration step. While in large-scale mission planning, the neighborhood size needs to be increased to achieve improved optimization, but expanding the neighborhood size will lead to excessive solution time of the algorithm. Therefore, we design a whole-neighborhood greedy search strategy to break the limitation of neighborhood size by maintaining a gain list of all non-conflicting window combinations, and attain the neighborhood solution that maximizes the objective gain $\Delta\mathbf{f}$g in the maximum possible neighborhood range.

In order to efficiently find the operation with the largest objective gain, the objective gain of all non-conflicting window combinations is calculated at each iteration and sorted based on the dictionary order and recorded as a list $\mathcal{X}_k$. Most of the window combinations have the same objective gain and conflict status as in the previous iteration, so only a small part of window combinations that are affected by the manipulation need to be updated.

3.4 Tabu List

The tabu list records the recently explored status, and prohibits the successive repeated searches for the same neighborhood solutions to prevent search loops and trapping in local optima.

The selection of the tabu size is closely related to the actual problem, too small will result in loops, too long will lead to slow convergence, drawing on the results of Zheng et al. [26], the tabu size is generally in the range $4\sqrt{N_X} \sim 10\sqrt{N_X}$.

3.5 Whole-Neighborhood Tabu Search Algorithm

This algorithm is an iterative algorithm, where the kth iteration step requires the following operations.

Step 1: Start iterating from the operation with the highest objective gain, stop when finding the first operation that does not trigger the tabu, and get the neighborhood solution of this iteration.

Step 2: Let $\mathcal{X}_{k+1} = \mathcal{X}_k$.

Step 3: The operation iterates through all window combinations on the satellite and checks whether their conflict status has changed. If a window combination changes from non-conflicting to conflicting due to this operation, the window combination needs to be removed from $\mathcal{X}_{k+1}$. Conversely, if a window combination changes from conflicting to non-conflicting due to this operation, the objective gain of operating the window combination needs to be recalculated and added to $\mathcal{X}_{k+1}$.

Step 4: Iterate through all window combinations corresponding to the task involved at the time of the operation, and if the task is in $\mathcal{X}_{k+1}$, update its target gain and reinsert it in the correct position to maintain the ordering.

Storing List $\mathcal{X}_k$ can use Red-black tree as the data structure, so that insertion and deletion can be done in $O\left(\log\left(\left|\mathcal{X}_k\right|\right)\right)$ time , where $\left|\mathcal{X}_k\right|$ denotes the total number of elements in List $\mathcal{X}_k$. Also, each window combination needs to record its conflict status, so that it only takes $O(1)$ time to complete the query and update of the conflict status.

The whole-neighborhood tabu search algorithm pseudo-code is shown in Table 1.

Table 1. The whole-neighborhood tabu search algorithm pseudo-code

Algorithm 1. The whole-neighborhood tabu search algorithm	
Input:	Mission Information, Satellite Parameters, Observation Window Information, Planning Model Related Parameters, Maximum Number Of Iterations, Tabu List Size N_T, N_S, N_X , **A**, **B**, **C**, K, Q, N
Output:	Optimal solution.
1.	Initialize the tabu list, generate the initial solution
2.	for $q = 1$ to Q do
3.	Whole-neighborhood search based on steps 1-4
4.	Update the tabu list, current solution and optimal solution
5.	end
6.	Return the optimal solution

3.6 Computational Complexity Analysis

At each iteration, the main computational effort of this algorithm is to maintain a list of wholly ordered window combinations $\mathcal{X}_k$ that do not have any constraint conflicts. Depending on the change of the window combination decision variable x_g , the increment $\Delta\mathbf{f}_g$, $g \in \mathcal{X}_k$ corresponding to each window combination, and its position in the list are adjusted accordingly. Assume that n_g denotes the number of window combinations that cover at least one identical task with window combination g; m_g denotes the number of window combinations which share at least one same satellite with window combination g. Then the computational complexity required to update the window combination gain values is $O\left(n_g \log\left(N_X\right)\right)$;

The computational complexity required for window conflict checking is $O\left(m_g \log\left(N_X\right)\right)$.

Generally, $n_g = O\left(N_T\right), m_g = O\left(N_S\right), N_X = O\left(N_T N_S\right)$, At that time, the computational complexity of a single iteration is $O\left(\max\left\{N_T, N_S\right\} \log\left(N_T N_S\right)\right)$, so the actual computational complexity is much less than $O\left(N_X\right)$.

4 Simulation and Analysis

Multiple sets of simulation experiments are designed to verify the effectiveness of the proposed algorithm. All simulation experiments are implemented in Java language and run on a laptop with Intel(R) Core(TM) i7-4720 CPU @2.60Ghz and 12GB RAM. The simulation software version is IntelliJ IDEA 2020.3.2 x64.

4.1 Simulation Settings

We use the same satellite orbit parameters as in the paper by He et al. [25]. The mission planning time range is 2022-04-12 00:00:00 to 2022-04-13 00:00:00. The satellite orbit parameters are shown in Table 2.

Table 2. Satellite orbit parameters

Satellite code	a(m)	e(1)	i(°)	ω(°)	Ω(°)	f(°)
Sat1	7200000.0	0.000627	96.576	0	175.72	0.075
Sat2	7200000.0	0.000627	96.576	0	145.72	30.075
Sat3	7200000.0	0.000627	96.576	0	115.72	60.075
Sat4	7200000.0	0.000627	96.576	0	85.72	90.075
Sat5	7200000.0	0.000627	96.576	0	55.72	120.075
Sat6	7200000.0	0.000627	96.576	0	25.72	150.075

Where a denotes the orbital semi-long axis, e is the orbital eccentricity, i is the orbital inclination, ω denotes the perigee angle, Ω denotes the ascending node equinox, and f is the true perigee angle. Fig. 2 shows the two-dimensional orbital schematic of the above six satellites, from which it can be shown that the orbits of the six satellites cover the Earth's surface uniformly.

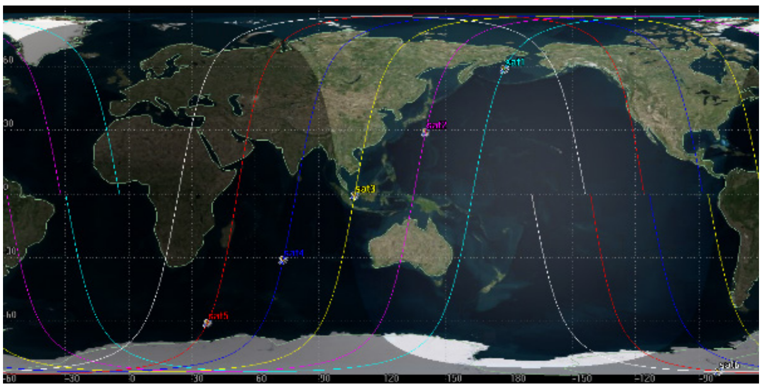

Fig. 2. Two-dimensional diagram of the satellite orbit

The observation missions are randomly selected on a global scale as shown in Fig. 3. Simulation experiments with the number of satellites 3, 4, 5 and 6 are considered for the cases of 100, 200 and 500 tasks to be observed, respectively, and the results of all experiments are averaged randomly for 10 times. The number of iterations of the algorithm in the experiments is set to 1000 and the length of the tabu list is set to 500.

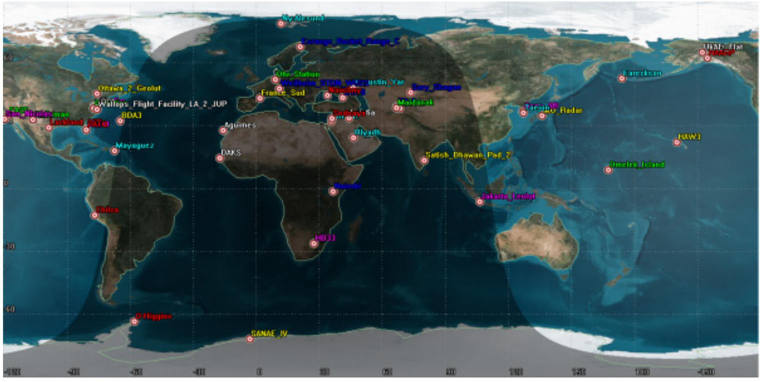

Fig. 3. Target distribution diagram

A traditional tabu search algorithm [27] and a hierarchical planning algorithm [28] were implemented for experimental comparison and the experimental results were analyzed and evaluated.

4.2 Analysis of Simulation Results

To analyze the solution effectiveness of the whole-neighborhood tabu search algorithm, we compared it with the traditional tabu search algorithm and hierarchical planning algorithm at three levels, including the total number of task planning, the number of highest priority task planning, and the number of second-highest priority task planning, respectively.

Fig. 4 shows the variation of the total number of tasks scheduled with iterations for the three algorithms in the 4-satellite 200-task scenario. As seen in Fig. 4, the whole-neighborhood tabu search algorithm has the highest number of planning tasks and the fastest convergence rate among the three types of algorithms.

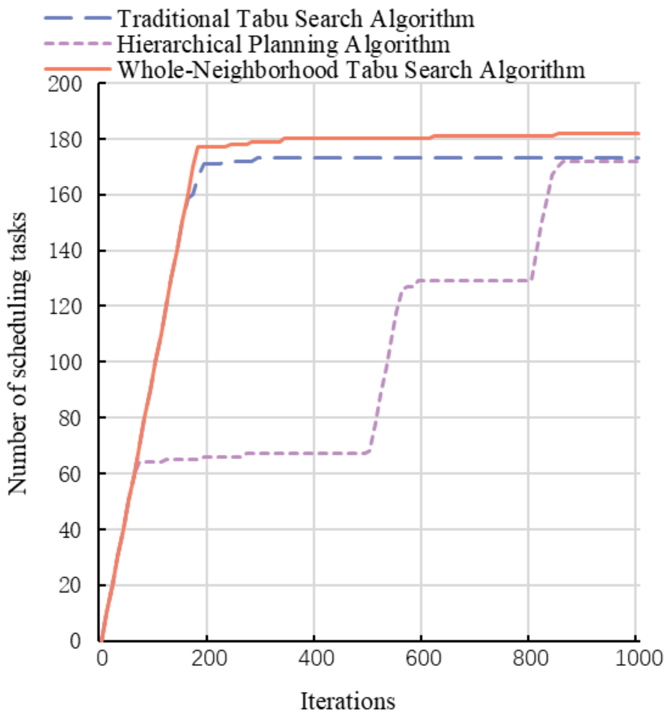

Fig. 4. Variation of the total number of task planning with the number of iterations

Fig. 5 shows the change in the number of highest priority tasks planned during the iterations in the 4-satellite 200-task scenario. From Fig. 5, it can be seen that the whole-neighborhood tabu search algorithm has planned the same number of highest-priority tasks as the hierarchical planning algorithm, and both have outperformed the traditional tabu search algorithm in terms of the number of highest-priority tasks planned.

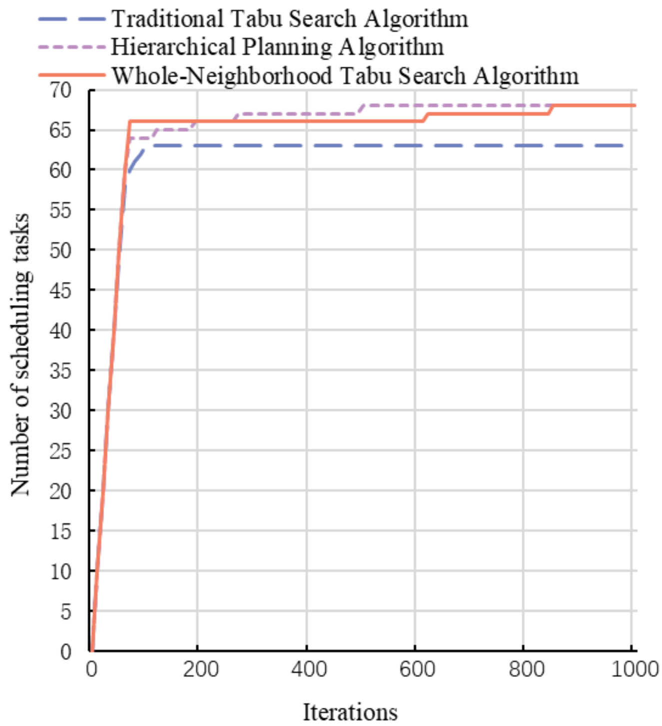

Fig. 5. Highest priority task planning situation

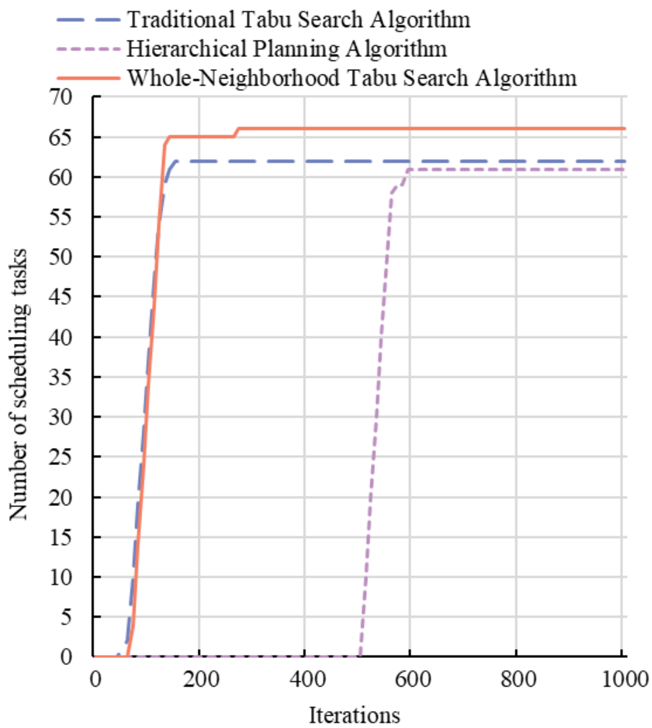

Fig. 6. Second-highest priority task planning situation

Fig. 6 shows how the number of next-highest-priority tasks planned changes during the iterations in the 4-satellite 200-task scenario. As can be seen from Fig. 6, the whole-neighborhood tabu search algorithm has planned the largest number of second-highest priority tasks and has the leading convergence rate. The hierarchical planning algorithm is unable to take into account different priority tasks and converges significantly slower than the other two algorithms due to its decomposition of the problem into a series of sub-optimization problems.

Therefore, the whole-neighborhood tabu search algorithm has the best algorithmic optimization effect in planning the high-priority tasks while being able to coordinate the low-priority tasks and maximize the use of satellite resources.

The statistics of the number of tasks planned for the three types of algorithms on the same test cases are shown in Table 3.

Table 3. Number of tasks planned by the three algorithms

Instance (satellite-task)	Hierarchical planning algorithm	Traditional tabu search algorithm	Whole-neighborhood tabu search algorithm
3-100	86.2	87.9	**93.2**
3-200	151.6	147.3	**159.6**
3-500	**176.0**	**176.0**	**176.0**
4-100	89.4	91.6	**95.4**
4-200	169.2	164.8	**177.6**
4-500	**256.0**	**256.0**	**256.0**
5-100	93.9	94.9	**97.8**
5-200	178.5	183.5	**191.4**
5-500	365.8	359.4	**371.7**
6-100	96.9	97.8	**99.3**
6-200	183.5	186.1	**194.2**
6-500	407.2	404.2	**427.4**

It is shown in Table 3 that the whole-neighborhood tabu search algorithm obtains the best optimization results in all instances, indicating that the whole-neighborhood greedy search strategy can be well combined with the tabu search algorithm and obtains better results in solving multi-satellite mission planning problems.

In order to verify the effectiveness and performance of the algorithm for large-scale mission planning problems, simulation experiments with the number of satellites of 20, 30, 50 and 100 are considered for the cases of 1000 and 3000 missions to be observed, respectively. The satellite orbital parameters of ascending node declination and true perigee angle are taken equally in the range of 0 to 360 degrees according to the number of satellites, and the orbital semi-long axis a is 7200 km, orbital eccentricity e is 0.000627, orbital inclination i is 96.576°, and perigee angle ω is 0°. The effect of the algorithm is shown in Table 4, and the performance of the algorithm is shown in Fig. 7 and Fig. 8. All experimental results are averaged randomly for 10 times.

Table 4. Statistics on the number of three algorithms planned for large scale task scenarios

Instance (satellite-task)	Hierarchical planning algorithm	Traditional tabu search algorithm	Whole-neighborhood tabu search algorithm
20-1000	894.9	893.6	**901.4**
30-1000	975.5	972.8	**981.9**
50-1000	997.6	992.4	**1000.0**
100-1000	**1000.0**	**1000.0**	**1000.0**
20-3000	1889.8	1887.3	**1897.7**
30-3000	2271.3	2274.2	**2281.1**
50-3000	2720.2	2716.8	**2725.8**
100-3000	**3000**	**3000**	**3000**

Table 4 statistically shows the mission planning of three types of methods, namely, hierarchical planning algorithm, traditional tabu search algorithm and whole-neighborhood tabu search algorithm, in a large-scale mission scenario. Fig. 7 and Fig. 8 show the variation curves of computational time consumption with the number of available satellites for the scale of 1000 and 3000 tasks to be planned for the algorithm proposed in this paper and the other two types of comparison algorithms, respectively. Combining the results shown in Table 4, Fig. 7 and Fig. 8, the computation time of the whole-neighborhood tabu search algorithm is much less than the other two algorithms under the premise that the total number of planning tasks of the three algorithms is comparable or the proposed algorithm is slightly higher than the other two comparative algorithms. The performance advantage of the whole-neighborhood tabu search algorithm becomes more and more obvious as the scale of the task planning problem increases, which is 1/4 and 1/5 of the computation time of the traditional tabu search algorithm and hierarchical planning algorithm, respectively, reflecting the superiority of the proposed algorithm.

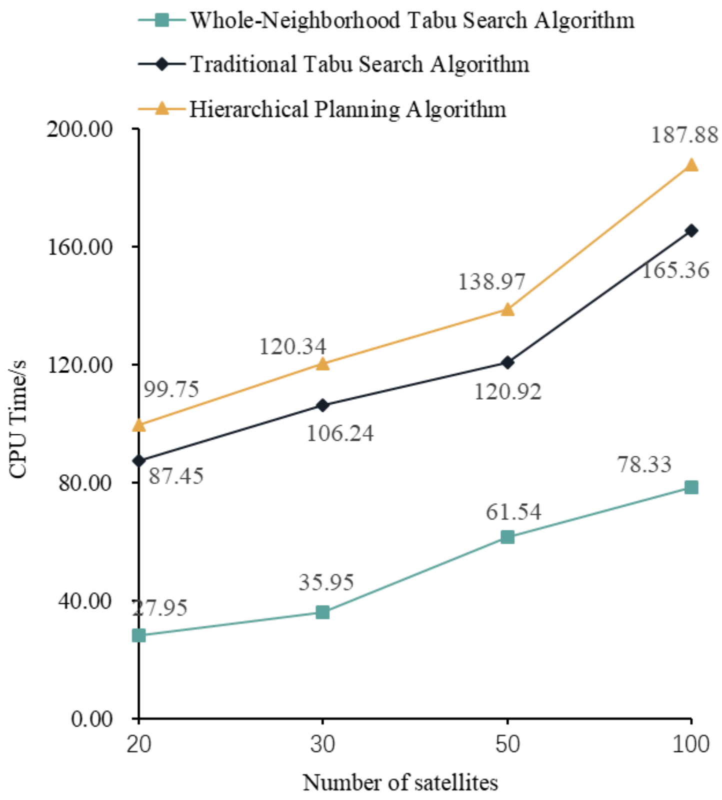

Fig. 7. 1000-task CPU Time comparison

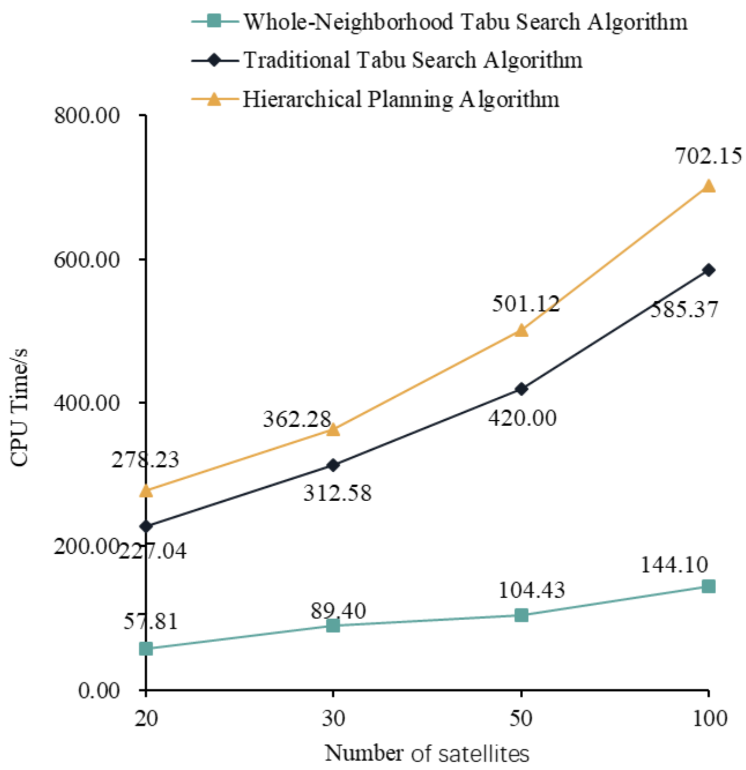

Fig. 8. 3000-task CPU Time comparison

5 Conclusion

In this paper, we study the large-scale multi-satellite mission planning problem, construct a satellite mission planning model based on the preemptive priority; and design a whole- neighborhood greedy search strategy combined with the tabu search algorithm to solve the mission planning problem. The effectiveness of the proposed search strategy is verified by numerous simulation experiments at different scales, and it achieves excellent planning results at different problem scales. Compared with other algorithms, the algorithm in this paper shows better performance in problem solving and efficiency, and has good prospects for engineering applications. In the further research work, other constraints such as data transmission will be considered to enrich the problem model and adapt the algorithm accordingly.

6 Acknowledgement

This work was supported by the National Natural Science Foundation of China (62201066, U20A20163).

References

[1] M. Lemaître, G. Verfaillie, F. Jouhaud F, J.M. Lachiver, N. Bataille, How to manage the new generation of agile earth observation satellites, in: Proc. Proceedings of the International Symposium on Artificial Intelligence, Robotics and Automation in Space, 2000.

[2] Z.X. Zheng, J. Guo, G. Eberhard, Onboard mission allocation for multi-satellite system in limited communication environment, Aerospace Science and Technology 79(2018) 174-186.

[3] N. Marco, C. Camilla, T. Massimo, Coverage area determination for conical fields of view considering an oblate earth, Journal of Guidance, Control, and Dynamics 42(10)(2019) 2233-2245.

[4] X. Wang, G. Wu, L. Xing, W. Pedrycz, Agile Earth Observation Satellite Scheduling Over 20 Years: Formulations, Methods, and Future Directions, IEEE Systems Journal 15(3)(2021) 3881-3892.

[5] G. Chen, X. Zeng, X. Liu, X. Rui, Transfer matrix method for the free and forced vibration analyses of multi-step Timoshenko beams coupled with rigid bodies on springs, Applied Mathematical Modelling 87(2020) 152-170.

[6] Z.B. E, R.H. Shi, L. Gan, H.X. Baoyin, J.F. Li, Multi-satellites imaging scheduling using individual reconfiguration based integer coding genetic algorithm, Acta Astronautica 178(2021) 645-657.

[7] K. Chandiramani, R. Verma, M. Sivagami, A Modified Priority Preemptive Algorithm for CPU Scheduling, Procedia Computer Science 165(2019) 363-369.

[8] H. Baek, J. Lee, Improved Schedulability Test for Non-Preemptive Fixed-Priority Scheduling on Multiprocessors, IEEE Embedded Systems Letters 12(4)(2020) 129-132.

[9] J.S. Lin, C.Y. Huang, C.C. Fang, Analysis and assessment of software reliability modeling with preemptive priority queueing policy, The Journal of Systems and Software 187(2022).

[10] B. Zhang, H.X. Zou, The model of imaging satellite task scheduling and research progress of algorithm, Computer Engineering and Applications 50(S1)(2014) 116-120.

[11] X. Chen, G. Reinelt, G. Dai, A. Spitz, A mixed integer linear programming model for multi-satellite scheduling, European Journal of Operational Research 275(2019) 694-707.

[12] Q.Y. Qu, K.X. Liu, X.J. Li, Y.F. Zhou, J.H. Lv, Satellite Observation and Data-Transmission Scheduling using Imitation Learning based on Mixed Integer Linear Programming, IEEE Transactions on Aerospace and Electronic Systems (2022) 1-25.

[13] S.J. Chen, Z.Li, M. Hu, Research Progress on Requirements Integrated Preprocessing and Mission Planning for Earth Observation Satellites, in: Proc. 2019 International Conference on Aeronautical Materials and Aerospace Engineering, 2019.

[14] M. Lemaître M, G. Verfaillie, F. Jouhaud, J. M. Lachiver, N. Bataille, Selecting and scheduling observations of agile satellites, Aerospace Science & Technology 6(5)(2002) 367-381.

[15] Y.H. Du, L.N. Xing, F. Yao F, Y.G. Chen, Survey on Models, Algorithms and General Techniques for Spacecraft Mission Scheduling, Acta Automatica Sinica 47(12)(2021) 2715-2741

[16] M. Ranjbar, S. Ramyar, Scheduling a constellation of agile earth observation satellites with preemption, Journal of Quality Engineering and Production Optimization 2(1)(2017) 47-64.

[17] Y.Z. Geng, Y.N. Guo, C.J. Li, G.F. Ma, W.B. Li, Optimal mission planning with task clustering for intensive point targets observation of staring mode agile satellite, Control and Decision 35(3)(2020) 613-621.

[18] X. Jiang, Y. Song, L.N. Xing, Dual-Population Artificial Bee Colony Algorithm for Joint Observation Satellite Mission Planning Problem, IEEE Access 10(2022) 28911-28921.

[19] W.N. Ding, K.F. Tian, S.Y. Wang, Mission Scheduling for Agile Earth Observation Satellites Based on Genetic-Tabu Hybrid Algorithm, Aerospace Control and Application 45(6)(2019) 27-32.

[20] X.L. Liu, G. Laporte, Y.W. Chen, R.J. He, An adaptive large neighborhood search metaheuristic for agile satellite sched-

uling with time-dependent transition time, Computers & Operations Research 86(2017) 41-53.
[21] Y.H. Du, L.N. Xing, Y.W. Chen, L. Wang, T. Ren, Integrated agile observation satellite scheduling problem considering different memory environments: a case study, Journal of the Brazilian Society of Mechanical Sciences and Engineering 42(1)(2020) 1-21.
[22] J.F. Cordeau, G. Laporte, Maximizing the value of an Earth observation satellite orbit, The Journal of the Operational Research Society 8(56)(2005) 962-968.
[23] L.H. Liu, Z.H. Dong, H.X. Su, D.Z. Yu, Y. Lin, Research on a Heterogeneous Multi-satellite Mission Scheduling Model for Earth Observation Based on Adaptive Genetic-Tabu Hybrid Search Algorithm, in: Proc. 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), 2021.
[24] Z.H. Zhang, J.Y. Zhang, J.K. Zhang, S.J. Ma, Y.Z. Li, Multi-satellite Cooperative Mission Planning Method Based on GeneticTabu Algorithm, Radio Engineering 52(7)(2022) 1127-1135.
[25] L. He, X.L. Liu, G. Laporte, Y.W. Chen, Y.G. Chen, An improved adaptive large neighborhood search algorithm for multiple agile satellites scheduling, Computers & Operations Research 100(2018) 12-25.
[26] S.F. Zheng, J.D. Cao, X.M. Lian, K.Q. Li, Random Reasonable Tabu Search Algorithm for Urban Pickup and Delivery Problem, Journal of System Simulation 22(7)(2010) 1688-1692
[27] C.R. Zuo, H.Y. Wang, Research on scheduling of earth observing satellites based on taboo search algorithm, Computer Engineering and Applications 46(1)(2010) 215-217.
[28] C. Zhang, Q. Zhang, Y. Zhao, H. Liu, X.M. Liu, A Heuristic Layering Mission Planning Algorithm for Earth Observation Satellite, Aerospace Control 39(2)(2021) 45-50.