Research Article

Imaging Tasks Scheduling for High-Altitude Airship in Emergency Condition Based on Energy-Aware Strategy

Li Zhimeng, He Chuan, Qiu Dishan, Liu Jin, and Ma Manhao

Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha 410073, China

Correspondence should be addressed to He Chuan; chuanhe@nudt.edu.cn

Received 12 April 2013; Accepted 27 May 2013

Academic Editors: C. Bigongiari and B. Vrsnak

Copyright © 2013 Li Zhimeng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Aiming to the imaging tasks scheduling problem on high-altitude airship in emergency condition, the programming models are constructed by analyzing the main constraints, which take the maximum task benefit and the minimum energy consumption as two optimization objectives. Firstly, the hierarchy architecture is adopted to convert this scheduling problem into three subproblems, that is, the task ranking, value task detecting, and energy conservation optimization. Then, the algorithms are designed for the subproblems, and the solving results are corresponding to feasible solution, efficient solution, and optimization solution of original problem, respectively. This paper makes detailed introduction to the energy-aware optimization strategy, which can rationally adjust airship’s cruising speed based on the distribution of task’s deadline, so as to decrease the total energy consumption caused by cruising activities. Finally, the application results and comparison analysis show that the proposed strategy and algorithm are effective and feasible.

1. Introduction

One of the most significant features of emergency scheduling problem (ESP) is timeliness; that is, the execution of task must be completed in its deadline. Otherwise, the task will lose its executive value or become invalid [1–3]. Under the emergency condition, the imaging task has its observation slot to reflect the requirements on the execution timing interval. For example, the emergency tasks, such as the observations on the targets about moving missile system, massing troops, and cruising battleship, generally need the responding agencies to scout timely in order to rapidly analyze the situation and to plan the operational activity.

Over the last decade, many military groups such as the US army have been devoted to development of the emergency imaging technology and improvement of the quick response ability of the reconnaissance system by incorporating multiple platforms. High-altitude airship is a promising solution for the emergency observation platform in the near-space [4, 5]. Unlike conventional heavier-than-air (HTA) aircraft, high-altitude airship is a lighter-than-air (LTA) aircraft equipped with steering and propulsion systems, and it generates lift force through the buoyancy instead of aerodynamics [6]. At present, high-altitude airship located in the near-space has attracted wide attention in many countries, and it is well known that some projects have been studied, for example, the HARV and HAA projects [7] in USA, Sky cat and CL-160 projects [8] in the European Union, ETRI [9] in South Korea, and Sky Net in Japan [10]. The scientists and engineers in China have conducted corresponding researches since the last century, and the verification airship has completed its low-altitude flight experiment in 2003.

As a new application platform, the high-altitude airship has many advantages in reconnaissance activities. For instance, it has a long duration, and a great deal of load carrying, and can achieve the fixed-point successive observation, and so forth [11]. In comparison with the traditional unmanned aircraft vehicle (UAV) [12, 13], the high-altitude airship can be operated continuously for several months, even for more than one year in the assigned airspace. It is also easy to acquire data and information uninterruptedly in a long period. Due to the fuel restriction, UAV has to implement the aerial refueling or return to the base frequently, so it is impossible to achieve the long-term and continuous monitoring at a lower cost. The capsule of the high-altitude airship is usually made from
the nonmetallic materials with less electromagnetism and heat reflecting, which makes it hard to be captured by radar. In addition, the high-altitude airship is vulnerable to be attacked and intercepted by many air-defense missiles, due to the operational height which is out of their fire range. Compared with the imaging reconnaissance satellite [14–16], the high-altitude airship has stronger ability of rapid response. Generally speaking, the ground support equipments for launching a high-altitude airship are fewer in number and have shorter period of launching preparation. Therefore, the theater reconnaissance, surveillance, and warning system can be established by high-altitude airship in a few hours, and the mass deployment can be rapidly implemented with its strong maneuverability. In terms of the efficiency-cost ratio, the in-orbit time of a high-altitude airship is nearly equal with that of an imaging reconnaissance satellite, but the usage cost is far less than the latter. In addition, the satellite is restricted by the fixed orbit in use and only can observe the targets in a certain time slot. On the contrary, the high-altitude airship can achieve the long-term and continuous observation on the target in the hover-and-stare way. Due to the previous advantages, the high-altitude airship has huge application potential in the emergency activities, such as the antiterrorism, disaster relief, and regional battles. The aforementioned advantages have turned high-altitude airship into an ideal imaging observation platform.

Existing studies on high-altitude airship are scattered over a range of journals, conferences, books, and reports. Rao et al. [17] presented a mission path following controller for the airship by employing artificial neural network (ANN). Tan et al. [18] introduce some methods and techniques to realize lightweight structure and present a review of current research on high-altitude airship with lightweight structures. Bessert and Frederich [19] investigated the aerodynamics behavior of high-altitude airship and presented a novel technology to test the aerodynamics on the structural behavior of airship. Ren et al. [20] analyzed the aerodynamics problems of high-altitude airship while launching, recovering, hovering, and introducing the achievement of airship dynamics research. Especially, there are numerous studies on energy system of high-altitude airship. Wang et al. [21] presented a novel computation method for solar radiation on solar cells of the airship, given the effect of the airship’s attitude on the performance of its energy system. Ma and Sun [22] developed a power management framework of high-altitude airship, which can rationally distribute power to subsystems so as to lighten the energy consumption in certain situation. Wang et al. [23] proposed an energy balance method to analyze the regenerative energy system, which can streamline the configuration design of high-altitude airship. In addition, there are also great deal of works focusing on the propulsion system. Chen et al. [24] constructed a simulation model and made some analyses about propeller of high-altitude airship. Jordi et al. [25] discussed the biomimetic principles for the structural design of airship. Various development tests are completed in their research, including wind tunnel testing and flight trials. Unfortunately, there are only a handful of works reported to date in the literature that propose the task planning of high-altitude airship, which greatly degrades the system performance such as task guarantee ratio and energy consumption.

In this paper, we focus on the imaging tasks scheduling problem on the high-altitude airship under the emergency condition. The power-speed model is constructed, which is employed to evaluate the energy consumption during the airship’s reconnaissance actives. We convert this scheduling problem to constrain satisfaction problem (CSP), then a heuristic algorithm based on the optimization sequence rule (OSR) is presented to obtain the task ranking scheme, and a value task detecting (VTD) method is provided to detect the key nodes that each airship needs to fly through in sequence. An energy-aware strategy (EAS) is also provided to optimize the task planning by rationally adjusting the cruising speed of airship. The simulation results show the effectiveness of this strategy.

The reminder of this paper is structured as follows. Section 2 makes detailed description on the reconnaissance process of high-altitude airship and establishes the corresponding models and proposed optimization objects. Section 3 converts the original problem into three sub-problems by adopting hierarchy architecture and design the solution algorithms, respectively. The simulation experiments and performance analysis are given in Section 4. The final section will conclude this paper and discuss the future research direction.

2. Problem Description and Modeling

The application of high-altitude airship in imaging reconnaissance activity is an asset for other reconnaissance equipments, and it is of great significance to build and improve the reconnaissance network. To facilitate analysis and modeling of this problem, we summarize the main notations used throughout this paper as follows:

\[ T_p = [t_{start}, t_{end}]: \] the active period of airship, where \( t_{start} \) refers to the starting time and \( t_{end} \) refers to the completion time of observation activity;

\[ Task = \{task_1, task_2, \ldots, task_n\} \] : the imaging task set, where the element \( task_i \) refers to the \( i \)th task, and \( n \) refers to the task number;

\( sit_i = (x_i, y_i) \) : the observation projection position of \( task_i \), where \( x_i, y_i \) denote the horizontal coordinate and vertical coordinate, respectively;

\( sit_0 = (x_0, y_0) \) : the projection coordinate of airship at the beginning;

\( S_{max} \) : the maximum cruising speed of airship;

\( td_i \) : the deadline of \( task_i \);

\( tb_i \) : the beginning timing instant of \( task_i \);

\( te_i \) : the completion timing instant of \( task_i \);

\( t_j \) : the duration time of \( task_j \), which includes the system stability time, load switch time, and data storage time;

\( S_{ij} \) : the average cruising speed of airship between \( sit_i \) and \( sit_j \).

The reminder of this paper is structured as follows. Section 2 makes detailed description on the reconnaissance process of high-altitude airship and establishes the corresponding models and proposed optimization objects. Section 3 converts the original problem into three sub-problems by adopting hierarchy architecture and design the solution algorithms, respectively. The simulation experiments and performance analysis are given in Section 4. The final section will conclude this paper and discuss the future research direction.
2.1. The Process of Task Execution. The imaging payload is usually installed in the cabin of high-altitude airship, which can be tilted or rotated within a certain angle to observe targets on the ground. During the task execution, the high-altitude airship flies according to the predetermined route and hover at a certain observation position, and in this way, the target can be observed by imaging payload.

As shown in Figure 1, task $i$ and task $j$ are located in different positions. After completing task $i$, the airship moves to another observing position to execute task $j$. The cruising of airship will take a long time due to the limited speed, which makes it almost impossible to execute task $j$ timely.

**Theorem 1.** If task $i$ can be observed before its deadline, it is called a value task; otherwise, it is called an invalid task.

Assume that the current time is $T_0$, and $L$ is the distance between the airship and task $i$. If task $i$ is a value task, the following conditions must be met:

$$T_0 + \frac{L}{S_{\text{max}}} + t_i \leq t_{d_i}. \tag{1}$$

Obviously, the number of value tasks decreases with the time advancement, and this trend is irreversible. Considering that the imaging targets are widely distributed in the battle area, it is nearly impossible to ensure all tasks to be observed timely. Therefore, it is necessary to choose a reasonable task set and allocate the observation time for airship in accordance with various constraint conditions, so as to realize maximum efficiency of observing activity.

2.2. The Power-Velocity Model. The energy system of high-altitude airship converts the solar radiation into electrical energy, thereby providing energy to the entire platform. Assume that the power of the propulsion system is $P_t$, and the efficiency is $\eta$; then, the actual propulsion power is

$$P_r = P_t \times \eta. \tag{2}$$

The cruising of high-altitude airship is subject to the impact of wind. The magnitude and direction of the propeller thrust are adjusted to balance the wind resistance, so as to realize continuous monitoring at fixed-point position. In this paper, it is assumed that the airship mainly relies on the electric propeller device to provide the thrust, which can be quickly adjusted in accordance with the wind direction and task position [26]. Since the working height of the airship is maintained, we only need to consider its horizontal movement as shown in Figure 2.

Normally, the wind field in near-space is stable; hence, the wind speed can be decomposed along the axial direction and normal direction of the airship. Consider

$$W_a = W \cos(\theta_2 - \theta_1),$$
$$W_n = W \sin(\theta_2 - \theta_1), \tag{3}$$

where $W$ is the velocity of wind, $\theta_1$ is the cruising direction of airship, and $\theta_2$ is the direction of wind field.

The typical power-velocity model of airship is [24, 27, 28]:

$$P_a = \frac{1}{2} \rho V^{2/3} (S - W_a)^2 C_a$$
$$P_n = \frac{1}{2} \rho V^{2/3} W_n^2 C_n, \tag{4}$$

where $\rho$ is the air density, $V^{2/3}$ is the characteristic area of airship based on its volume $V$, $S$ is the airship’s cruising speed relative to the ground, and $C_a$ and $C_n$ refer to the aerodynamic coefficients.

2.3. The Optimization Objectives. Let $X = (x_1, x_2, \ldots, x_n)$, $Y = [y_{i,j}]_{n \times n}$, and $S = [S_{i,j}]_{(n+1) \times (n+1)}$ be the decision variables. If task $i$ is effectively performed, then it is a value task (assume that $x_i = 1$); otherwise, let $x_i = 0$. If $x_i x_j = 1$ and task $j$ is the preceding task of task $i$, then let $y_{i,j} = 1$; otherwise, let $y_{i,j} = 0$.
The primary optimization objective of the tasks scheduling problem on the high-altitude airship is to maximize the guarantee ration $ER(X, Y, S)$:

$$ER(X, Y, S) = \frac{1}{n} \sum_{i=1}^{n} x_i.$$  

(5)

The total energy consumption $En_{total}(X, Y, S)$ should be minimized on the basis of the maximization of $ER(X, Y, S)$. $En_{total}(X, Y, S)$ includes two parts; $En_{active}(X, Y, S)$ and $En_{static}(X, Y, S)$ are the energy consumptions caused by cruising and balance resistance in the suspension position, respectively.

According to (2)–(4), the energy consumption of airship's cruising from $s_{it}$ to $s_{jt}$ is

$$en_{active}^{i,j} = \frac{\rho v^{2/3} C_a d_{i,j}}{2\eta S_{i,j}}[W \sin(\theta_2 - \theta_1)]^{3/2},$$

(6)

$$+ \frac{\rho v^{2/3} C_a d_{i,j}}{2\eta S_{i,j}}[S_{i,j} - W \cos(\theta_2 - \theta_1)]^{3/2}.$$  

(7)

If the high-altitude airship locate in the observation position of task $t_k$, the energy consumption caused by resisting the effect of wind is

$$en_{static}^{i,j} = \frac{\rho v^{2/3} C_a t_k}{2\eta}W \cos(\theta_2 - \theta_1)^{3/2},$$

(8)

$$+ \frac{\rho v^{2/3} C_a t_k}{2\eta}W \sin(\theta_2 - \theta_1)^{3/2}.$$  

According to (6) and (7), the total energy consumption of airship during execution of the tasks is

$$En_{total}(X, Y, S) = En_{active}(X, Y, S) + En_{static}(X, Y, S) = \sum_{i=1}^{n} \sum_{j=1}^{n} y_{i,j} en_{active}^{i,j} + \sum_{i=1}^{n} x_i en_{static}^{i,j}.$$  

(9)

2.4. The Programming Model. In the typical route planning of UAV, it is necessary to simultaneously consider the maximum turning angle, maximum climbing angle, minimum flight altitude, minimum path length, and other constraints. The purpose is to ensure that the cruising path can meet the aircraft's maneuvering characteristics and reduce the probability of damaging the aircraft in the no-fly zone and threatened area. As for the high-altitude airship in this paper, its working space has no spatial constraint, so the no-fly zone is an unnecessary consideration. At the same time, the low speed and slow dynamics provide the airship with a large-angle cornering ability, and it is also unnecessary to consider the minimum path length due to its suspension ability. Moreover, the threatened area of the high-altitude airship can be ignored due to the difficulty of being captured by the radar. However, the main constraints of tasks scheduling on the high-altitude airship are listed as follows.

**Constraint 1.** The high-altitude airship only executes the observation task within its active period.

**Constraint 2.** Each task can be executed only once, and it must be completed before its deadline.

**Constraint 3.** If a task can be executed, the execution time should be no less than the required continuous working time.

**Constraint 4.** Only one preceding task or one following task of each task is allowable at most.

**Constraint 5.** The preemptive service in the task execution is prohibited. Once the execution starts, the process cannot be terminated until completion.

**Constraint 6.** Before performing a new task, the airship needs sufficient time to change the observation position.

**Constraint 7.** For any two tasks to be executed, the certain priority order exists.

**Constraint 8.** The moving processes of airship only exist in different observation positions.

**Constraint 9.** The cruising speed of high-altitude airship in each path segment cannot be higher than the maximum cruising speed.

Let $R_1$, $R_2$, and $R_3$ be the feasible solution space of the decision variables $X$, $Y$, and $S$. In the separate optimization of $ER(X, Y, S)$, their optimal solution spaces are $Q_1$, $Q_2$, and $Q_3$, respectively. The programming model of this scheduling problem is given as follows:

$$Z_1(X, Y, S) = \max_{X \in Q_1, Y \in Q_2, S \in Q_3} \left\{ \frac{1}{n} \sum_{i=1}^{n} x_i \right\},$$

$$Z_2(X, Y, S) = \min_{X \in Q_1, Y \in Q_2, S \in Q_3} \left\{ \sum_{i=1}^{n} \sum_{j=1}^{n} y_{i,j} en_{active}^{i,j} + \sum_{i=1}^{n} x_i en_{static}^{i,j} \right\}$$

subject to

$$[t_{b_i}, t_{e_i}] \subseteq [t_{start}, t_{end}],$$

$$\sum_{i=1}^{n} x_i \leq 1, \quad t_{e_i} \leq t_{d_j},$$

$$t_{e_i} - t_{b_j} \geq t_j, \quad \text{if } x_i = 1,$$

$$\sum_{j=1}^{n} y_{i,j} \leq 1, \quad \sum_{j=1}^{n} y_{j,i} \leq 1,$$

$$[t_{b_i}, t_{e_i}] \cap [t_{b_j}, t_{e_j}] = \emptyset, \quad \text{if } x_i \cdot x_j = 1,$$

$$t_{e_i} + \frac{d_{i,j}}{S_{i,j}} - K(1 - y_{i,j}) \leq t_{b_j},$$

$$y_{i,j} \cdot y_{j,i} = 0,$$

$$\sum_{j=1}^{n} y_{i,j} = 0,$$

$$S_{i,j} \leq S_{max},$$

$$x_i, y_{i,j} \in \{0, 1\},$$

$$\forall i, j \in \{1, 2, \ldots, m\}, \quad k \in \{1, 2, \ldots, n\}.$$
where the former nine inequality formulas correspond to the aforementioned constraints, respectively, and the tenth inequality formula restricts the range of the decision variables.

3. Scheduling Algorithms

There exist numerous constraints in the tasks scheduling problem, so it is difficult to solve this problem directly. Therefore, the hierarchical optimization can be used to convert the original problem to the following sub-problems.

1. Determine the priority execution order of the tasks, that is, the task sorting problem, which can be solved by the OSR algorithm.

2. Select the observation tasks for the airship, that is, the value tasks detection problem, which can be solved by the OSR algorithm.

3. Adjust the planning scheme to reduce energy consumption of the airship, that is, the energy conservation problem, which can be solved by the EAS algorithm.

Firstly, the task guarantee ratio is the primary optimization objective of this scheduling problem. Thus, this paper considers that maximum cruising speed should be used by the airship to execute tasks as frequently as possible. Then, the distributions of value tasks are tested, and the initial scheduling scheme of the original problem is obtained. On this basis, energy conservation is regarded as another optimization goal. The cruising speed of airship at each leg is adjusted, and the execution time of all value tasks is updated.

Theorem 2. The feasible solution of the original problem is expressed with two-tuples $SP = (vq, sq)$, where $vq = \{task_{v1}, task_{v2}, ..., task_{vn}\}$ is the task set arranged by priority execution order and $sq = \{s_{q1}, s_{q2}, ..., s_{q(n-1),m}\}$ is the speed set of airship at each leg.

Theorem 3. For any feasible solutions $SP_1$ and $SP_2$, if $SP_1$ is superior to $SP_2$, then $SP_1 \succ SP_2$ or $SP_1 \bowtie SP_2$; if $SP_1$ is inferior to $SP_2$, then $SP_1 \prec SP_2$ or $SP_2 \succ SP_1$; otherwise, $SP_1 = SP_2$.

The comparison method of any two feasible solutions $SP_1$ and $SP_2$ is presented as follows:

- $SP_1 \succ SP_2$, if $ER(SP_1) > ER(SP_2)$,
- $SP_1 \succ SP_2$, if $ER(SP_1) = ER(SP_2)$, $E_{n_{total}}(SP_1) < E_{n_{total}}(SP_2)$,
- $SP_1 = SP_2$, if $ER(SP_1) = ER(SP_2)$, $E_{n_{total}}(SP_1) = E_{n_{total}}(SP_2)$,
- $SP_1 \prec SP_2$, otherwise.

Theorem 4. For any effective solution $SP_1 = (vq, sq)$ and feasible solution $SP_2 = (vq', sq')$, if $SP_1 \prec SP_2$, then $SP_2$ is an optimal solution of $SP_1$.

The conventional methods to obtain effective solutions including (a) improving the priority execution order of tasks in order to obtain more value tasks; (b) reducing energy consumption by adjusting the cruising speed of airship on the basis of ensuring the quantity of value tasks. The previous methods are realized by the OSR algorithm and EAS algorithm, respectively.

3.1. The OSR Algorithm. The OSR algorithm sorts all elements in Task to get the priority execution order of tasks, and the guarantee ratio is the primary optimization objective. Although all tasks are ranked, only part of the tasks can be observed timely; that is, the ranking result is only available for value tasks tested by the VTD algorithm, and it shows the observation sequence of executable tasks. The analysis of sorting rules in two tasks is as follows.

Let $Task = \{task_1, task_2\}$ denote the task set, $T_0$ the current timing instant and $s_{t0}$ the initial position of airship. Given two feasible solutions $SP_1 = (vq_1, sq)$ and $SP_2 = (vq_2, sq)$, where $vq_1 = \{task_{k1}, task_{k2}\}$ and $vq_2 = \{task_{k2}, task_{k1}\}$ and $sq = \{s_{q_{max}}, s_{max}\}$, there is no harm in assuming that $td_1 - d_{0.2}s_{max} \geq td_2 - d_{0.1}s_{max}$, and then, the relationship between $ER(SP_1)$ and $ER(SP_2)$ is discussed as follows.

(a) If $task_1$ and $task_2$ are both value tasks in $SP_1$, then

$$T_0 + d_{0.1}s_{1}^{\text{max}} + t_1 \leq td_1,$$

$$T_0 + (d_{0.1} + d_{1.2})s_{1}^{\text{max}} + t_1 + t_2 \leq td_2.$$  \hspace{1cm} (10)

In accordance with the assumption, we can get

$$T_0 + d_{0.2}s_{1}^{\text{max}} + t_2 \leq td_2,$$

$$T_0 + (d_{0.2} + d_{1.2})s_{1}^{\text{max}} + t_1 + t_2 \leq td_1.$$  \hspace{1cm} (11)

Then, $task_1$ and $task_2$ are both value tasks in $SP_2$; thereby, $ER(SP_1) = ER(SP_2)$.

(b) As for $SP_1$, if $task_1$ is a value task while $task_2$ is an invalid task, then:

$$T_0 + d_{0.1}s_{1}^{\text{max}} + t_1 \leq td_1,$$

$$T_0 + (d_{0.1} + d_{1.2})s_{1}^{\text{max}} + t_1 + t_2 \leq td_2.$$  \hspace{1cm} (12)

As for $SP_2$, if $task_2$ is a value task, then $ER(SP_1) \leq ER(SP_2)$; if $task_2$ is an invalid task, we can still ensure that $ER(SP_1) = ER(SP_2)$ due to $T_0 + d_{0.1}s_{1}^{\text{max}} + t_1 \leq td_1$.

(c) As for $SP_1$, if $task_1$ is an invalid task while $task_2$ is a value task, then

$$T_0 + d_{0.1}s_{1}^{\text{max}} + t_1 \leq td_1,$$

$$T_0 + (d_{0.1} + d_{1.2})s_{1}^{\text{max}} + t_1 + t_2 \leq td_2.$$  \hspace{1cm} (13)

As for $SP_2$, $task_2$ is a value task; thereby, $ER(SP_1) \leq ER(SP_2)$.

(d) If $task_1$ and $task_2$ are both invalid tasks in $SP_1$, then

$$T_0 + d_{0.1}s_{1}^{\text{max}} + t_1 \leq td_1,$$

$$T_0 + d_{0.2}s_{1}^{\text{max}} + t_2 \leq td_2.$$  \hspace{1cm} (14)

There is an equation $ER(SP_1) = ER(SP_2)$.
If \( t_{d1} - d_{0.2S_{\text{max}}} \leq t_{d2} - d_{0.1S_{\text{max}}} \), the similar method can be used to analyze the sorting rules. However, the following conclusions can be acquired:

(i) If \( t_{d1} - d_{0.2S_{\text{max}}} \geq t_{d2} - d_{0.1S_{\text{max}}} \) then \( \text{ER}(SP_1) \leq \text{ER}(SP_2) \).

(ii) If \( t_{d1} - d_{0.2S_{\text{max}}} \leq t_{d2} - d_{0.1S_{\text{max}}} \) then \( \text{ER}(SP_1) \geq \text{ER}(SP_2) \).

According to the previous conclusions, the OSR algorithm is proposed to solve the task sorting problem. The main steps of OSR are presented as in Algorithm 1.

The task sorting problem is a combinatorial optimization problem. At present, there is no algorithm which can be used to obtain the optimal solution within the polynomial time complexity. Similar to the EDF algorithm, the OSR algorithm is a heuristic algorithm, which only generates the optimal scheme instead of a common optimized scheme. The effectiveness of the OSR algorithm will be verified in the subsequent experiments.

3.2. The VTD Algorithm. According to the task sorting result obtained by OSR, the deadline constraints for each element in set \( \text{Task} \) are checked in order. The value tasks in set \( \text{Task} \) are considered as the key nodes, and the cruising path optimization method between the successive key nodes can be learned from [17, 29, 30].

**Theorem 5.** If both \( \text{task}_i \) and \( \text{task}_j \) are value tasks, and \( \text{task}_j \) is arranged to be executed just next to \( \text{task}_i \), then \( \text{task}_i \) is called the preceding task of \( \text{task}_j \), and \( \text{task}_j \) is the following task of \( \text{task}_i \).

Consider \( \text{task}_j \) to be the following task of \( \text{task}_i \), so the execution time interval of \( \text{task}_j \) is presented as follows:

\[
\begin{align*}
\text{tb}_j &= \text{te}_i + d_{i,j} S_{i,j}^{-1}, \\
\text{te}_j &= \text{tb}_j + t_j.
\end{align*}
\]

(15)

Assume that the airship has maximal cruising speed while solving the value tasks detection problem based on the hierarchy architecture. The pseudocode of VTD is shown as Algorithm 2.

3.3. The EAS Algorithm. If \( \text{task}_i \) is executed, the power consumption of the high-altitude airship caused by cruising in speed \( S^* \) to the position \( \text{sit}_j \) and to observe \( \text{task}_j \) can be calculated as

\[
\begin{align*}
\text{energy}_{i,j}(S^*) &= \text{energy}_{i,j}^{\text{active}} + \text{energy}_{i,j}^{\text{static}} \\
&= \alpha C_a \left[ W \cos(\theta_2 - \theta_1) \right] t_j \\
&\quad + \frac{d_{i,j}}{S^*} \left( W \cos(\theta_2 - \theta_1) \right)^3 \\
&\quad + \alpha C_n \left( \frac{d_{i,j}}{S^*} + t_j \right) \left| W \sin(\theta_2 - \theta_1) \right|^3.
\end{align*}
\]

(16)

where \( \alpha = \rho V^{2/3}(2\eta)^{-1} \) is a constant.

If the cruising speed of airship is limited at \( S_{i,j} \in [S_{\text{min}}, S_{\text{max}}] \), the optimal cruising speed of airship between \( \text{sit}_i \) and \( \text{sit}_j \) will be \( S^* = \{ S^* | \text{energy}_{i,j}(S^*) \leq \text{energy}_{i,j}(S) \text{ for all } S \in [S_{\text{min}}, S_{\text{max}}] \} \). For the convenience to describe this problem, we define the function of the optimal cruising speed (value range: \([S_{\text{min}}, S_{\text{max}}]\)) between \( \text{sit}_i \) and \( \text{sit}_j \) as

\[
S^* = \text{OPTE}_{i,j}(S_{\text{min}}, S_{\text{max}}).
\]

(17)

**Theorem 6.** If \( \{ S_1, S_2 \} \supset \{ S_3, S_4 \} \), \( S_a = \text{OPTE}_{i,j}(S_1, S_2) \), and \( S_b = \text{OPTE}_{i,j}(S_3, S_4) \), then one can get \( \text{energy}_{i,j}(S_a) \leq \text{energy}_{i,j}(S_b) \).

**Proof.** Since \( S_a = \text{OPTE}_{i,j}(S_1, S_2) \), we obtain that \( \text{energy}_{i,j}(S_a) \leq \text{energy}_{i,j}(S) \), for all \( S \in \{ S_1, S_2 \} \). For \( S_b \in \{ S_1, S_2 \}, \text{energy}_{i,j}(S_b) \leq \text{energy}_{i,j}(S_b) \) exists.

**Inference 1.** If \( S_3 \in \{ S_1, S_2 \}, S_4 = \text{OPTE}_{i,j}(S_1, S_2) \), then \( \text{energy}_{i,j}(S_3) \geq \text{energy}_{i,j}(S_4) \) is proven.

**Proof.** It can be considered that \( S_4 = \text{OPTE}_{i,j}(S_1, S_2) \). According to Theorem 6, since \( \{ S_3, S_4 \} \in \{ S_1, S_2 \} \), so Theorem 6 exists.

Assume that \( SP = \{ \text{wq, sq} \} \) is an efficient solution, where \( \text{wq} = \{ \text{task}_{i1}, \text{task}_{i2}, \ldots, \text{task}_{ig} \} \), and \( \text{sq} = \{ S_{i1}, S_{i2}, \ldots, S_{i(g-1)+1} \} \). Let \( SP^* = \{ \text{wq, sq}^* \} \) denote the improved solution of \( SP \), and it is obtained by the EAS listed as follows

**Step 1.** Let \( \text{task}_0 \) denote a virtual task, and set \( t_0 \leftarrow 0, t_b \leftarrow 0, t_b \leftarrow 0, i \leftarrow 0 \).

**Step 2.** On the premise of ensuring the value tasks, the speed range \([S_{\text{min}}, S_{\text{max}}]) \) of airship cruised from \( \text{sit}_a \) to \( \text{sit}_b \) is calculated.

**Step 3.** The optimal speed \( S_{\text{min}}^{\nu,t} = \text{OPTE}(S_{\nu,t}^{\min}, S_{\nu,t}^{\max}) \) of the airship cruised from \( \text{sit}_a \) to \( \text{sit}_b \) is calculated by (17). Let \( S_{\text{min}}^{\nu,t} \leftarrow S_{\text{min}}^{\nu,t} \) and \( i \leftarrow i + 1 \).

**Step 4.** The execution period of \( \text{task}_{i} \) is updated by (15).

**Step 5.** If \( i \leq g \), go to Step 2; otherwise, the iteration ends.

In the previous algorithm, \( S_{\text{max}} \) is the maximum cruising speed of airship along each leg. The minimum cruising speed of airship from \( \text{sit}_a \) to \( \text{sit}_b \) is

\[
S_{\text{min}}^{\nu,t} = \max_{j=0}^{j=g} \left\{ \frac{\sum_{k=1}^{j-1} d_{k,v}(k+1)}{\sum_{k=1}^{j} t_{v,k}(k+1) - t_{v,a}} \right\}.
\]

(18)

**Theorem 7.** As for \( \text{task}_i \) and \( \text{task}_j \), if \( \text{task}_i \) is priority to \( \text{task}_j \), let \( \text{task}_i \succ \text{task}_j \) or \( \text{task}_j \prec \text{task}_i \); otherwise, let \( \text{task}_i \prec \text{task}_j \) or \( \text{task}_j \succ \text{task}_i \).

**Theorem 8.** Assume that \( SP = \{ \text{wq, sq} \} \) is the optimized solution obtained by the EAS algorithm, and then for all \( \text{task}_{i} \in \text{wq} = \{ \text{task}_{i1}, \text{task}_{i2}, \ldots, \text{task}_{ig} \} \), \( S_{\text{min}}^{\nu,t} \leq S_{\text{max}} \) exists.
(1) Queue ← Task; NQueue ← NULL; point₀ ← sit₀;
(2) while Queue ≠ NULL do
(3) Let set G ← NULL
(4) for each task taskᵢ in Queue do
(5) Calculate the distance L₀,₁ between point₀ and taskᵢ;
(6) Add gᵢ = tdᵢ + L₀,₁S⁻¹ max to set G;
(7) end for
(8) g_k = min(g_i); /*Select the minimal member in G*/
(9) Search the last element taskᵢ in NQueue; /*The set
NQueue is used to save the execution sequence*/
(10) Let task_k > taskᵢ/*Assign the order for task_k*/
(11) Remove task_k from Queue to NQueue, and task_k is the last element in NQueue;
(12) point₀ ← sit_k;
(13) endwhile

Algorithm 1: Pseudocode of OSR algorithm.

(1) idle ← 0; VQueue ← NULL; point₀ ← sit₀/*Initialization*/
(2) for each task taskᵢ ∈ NQueue in sequence do
(3) pointᵢ ← sitᵢ; /*Search the value task in NQueue*/
(4) Calculate the distance L₀,₁ between point₀ and pointᵢ;
(5) if idle + L₀,₁S⁻¹ max + tᵢ ≤ tᵢ then/*taskᵢ is a value task*/
(6) Calculate thᵢ and teᵢ by (15);
(7) idle ← teᵢ; point₀ ← pointᵢ;
(8) Add taskᵢ to the value task set VQueue;
(9) else
(10) Reject taskᵢ/*taskᵢ is an invalid task, which can not been executed in its deadline*/
(11) end if
(12) end for
(13) for each task taskᵢ ∈ G in sequence do
(14) Delete taskᵢ from G, and set xᵢ₀ ← 1; /*Assign value to the decision vector X*/
(15) if G ≠ NULL then
(16) Select the first task taskᵢ in G; /*taskᵢ is the next value task behind taskᵢ*/
(17) Let yᵢ₁₀ ← 1;
(18) end if
(19) end for

Algorithm 2: Pseudocode of VTD algorithm.

Proof. For all taskᵢ ∈ q, taskᵢ < taskᵢ₀, since taskᵢ is a value task, we obtain that

\[ \text{te}_i + \sum_{k=1}^{j-1} (t_{(k+1)} + d_{sk,v(k+1)}S^{-1}_{\text{max}}) \leq \text{td}_{v,j}. \]  \hspace{1cm} (19)

We may reach the following conclusion:

\[ \text{td}_{v,j} - \sum_{k=1}^{j-1} t_{v(k+1)} - \text{te}_{v,j} \geq \sum_{k=1}^{j-1} d_{sk,v(k+1)}S^{-1}_{\text{max}}. \]  \hspace{1cm} (20)

In other words,

\[ \frac{\sum_{k=1}^{j-1} d_{sk,v(k+1)}}{\text{td}_{v,j} - \sum_{k=1}^{j-1} t_{v(k+1)} - \text{te}_{v,j}} \leq S_{\text{max}}. \]  \hspace{1cm} (21)

For the discretion of taskᵢ, we may find that

\[ S_{\text{min}} = \max_{j'=1,\ldots,g} \left\{ \frac{\sum_{k=1}^{j-1} d_{sk,v(k+1)}}{\text{td}_{v,j} - \sum_{k=1}^{j-1} t_{v(k+1)} - \text{te}_{v,j}} \right\} \leq S_{\text{max}}. \]  \hspace{1cm} (22)

Hereby, Theorem 8 holds.

Theorem 9. The optimized solution SP = (qv, sq) obtained by EAS algorithm is still an efficient solution.

Proof. According to Theorem 3, if SP = (qv, sq) is still an efficient solution, the following conditions are satisfied: (a) for all taskᵢ ∈ q is a value task; (b) for all S_p,v(j+1) ∈ sq is no larger than S_max.
As for all task \( v_i \in \omega_i \), according to (18), we may find that

\[
S_{\min}^{\max}(V_i, V(i+1)) = \max_{j \in [i+1], \ldots, g} \left\{ \frac{\sum_{k=1}^{j-1} d_{v_k, v(k+1)} + \sum_{k=1}^{j-1} t v_k(k+1) - t v_i}{t d_{v_j(i+1)}} \right\}
\]

(23)

In other words,

\[
\frac{d_{v_i, v(i+1)}}{S_{\min}^{\max}(V_i, V(i+1))} \leq t d_{v(i+1)} - t v_i - t e_{v_i}.
\]

(24)

So

\[
te_{v_i} + \frac{d_{v_i, v(i+1)}}{S_{\min}^{\max}(V_i, V(i+1))} + t d_{v(i+1)} \leq t e_{v_i} + \frac{d_{v_i, v(i+1)}}{S_{\min}^{\max}(V_i, V(i+1))} + t v_i = t d_{v(i+1)}.
\]

(25)

Thus, it can be regarded that task \( v_i \in \omega_i \) is a value task. According to Theorem 8, for all task \( v_i \in \omega_i \), there always exists \( S_{\min}^{\max}(V_i, V(i+1)) \leq S_{\max} \). As for this, there exist \( S_{v_i, v(i+1)} \in [S_{\min}^{\max}(V_i, V(i+1)), S_{\max}] \) and \( S_{v_i, v(i+1)} \leq S_{\max} \).

In conclusion, Theorem 9 is valid.

The feasible solution can be converted into an efficient solution by VTD algorithm, so as to figure out the decision variable \( X, Y \); on this basis, the efficient solution can be transformed into an optimal solution by EAS strategy in order to figure out the decision variable \( S \). According to the decision variables, the power consumption of optimized solution \( SP \) can be described as follows:

\[
En_{\text{total}}(SP) = \sum_{\text{task} \in \text{Task}} y_{i,j} \cdot \text{energy}_{i,j}(S_{i,j}).
\]

(26)

Conclusion 1. If \( SP^* \) is the optimized solution of the efficient solution \( SP \), there exists \( ER(SP) = ER(SP^*) \).

Proof. According to Theorem 9, \( SP^* \) is still an efficient solution, so that \( SP \) and \( SP^* \) have the same value tasks that is, \( ER(SP) = ER(SP^*) \).

Theorem 10. Assume that \( SP = (\omega, s) \) is an efficient solution and \( SP^* = (\omega, s^*) \) is an optimized solution obtained with EAS algorithm; as for this, \( SP^* \triangleright SP \) exists.

Proof. For any \( S_{\min}^{\max}(V_i, V(i+1)) \in s q^* \), there is \( S_{\min}^{\max}(V_i, V(i+1)) \in [S_{\min}^{\max}(V_i, V(i+1)), S_{\max}] \). According to Theorem 6, energy \( S_{\min}^{\max}(V_i, V(i+1)) \leq S_{\max} \) is exists. Therefore, we can get

\[
En_{\text{total}}(SP^*) = \sum_{\text{task} \in \text{Task}} y_{i,j} \cdot \text{energy}_{i,j}(S_{i,j})
\]

\[
\leq \sum_{\text{task} \in \text{Task}} y_{i,j} \cdot \text{energy}_{i,j}(S_{\max})
\]

(27)

Based on Conclusion 1, \( ER(SP) = ER(SP^*) \); hereby, \( SP^* \triangleright SP \) exists.

The effectiveness of this algorithm will directly affect the effectiveness of the proposed method. For VTD to be the precise algorithm, we only analyze the effectiveness of OSR and EAS.

4. Experimental Analysis

In this section, simulation experiment is conducted to illustrate the effectiveness of the proposed method. For VTD to be the precise algorithm, we only analyze the effectiveness of OSR and EAS.

4.1. Experimental Parameters. The proposed algorithms are implemented by Matlab2007 on a PC with Pentium IV 3.06 GHz CPU, 2 GB memory. As far as we know, there are no accepted benchmarks yet in scheduling problem of high-altitude airships, so the random models are used to construct the application scenario and simulate the battlefield area with \( 200 \times 300 \text{km}^2 \).

The main parameters for high-altitude airships simulation are listed in Table 1.

We divide task number into ten levels from 30 to 300 for offering the flexibility to simulate the various workloads on high-altitude airships. The positions of tasks are generated randomly in battlefield, and their deadline is distributed in a uniform distribution spanning over the active period of airship.

Table 2 gives the configuration of environment parameters employed in our experiment.

Additionally, in order to reduce the calculation complexity of experiments, we let the constant \( \alpha \) of (16) be equal to 1. Then the calculation results of total energy consumption are relation values instead of real values.

4.2. The Effectiveness of the OSR Algorithm. The heuristic algorithm OSR is employed to solve the task sorting problem. The effectiveness of this algorithm will directly affect the number of value tasks. In order to test the performance of the OSR algorithm, we have compared it with the EDF algorithm and Greedy algorithm.

<p>| Table 1: Parameters for high-altitude airship. |</p>
<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active period</td>
<td>[6, 18] h</td>
</tr>
<tr>
<td>Maximum speed</td>
<td>90 km/h</td>
</tr>
<tr>
<td>Efficiency index</td>
<td>0.5</td>
</tr>
<tr>
<td>Duration time</td>
<td>[1, 6] min</td>
</tr>
</tbody>
</table>

<p>| Table 2: Parameters for environment. |</p>
<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wind direction</td>
<td>30°</td>
</tr>
<tr>
<td>Wind speed</td>
<td>5 m/s</td>
</tr>
<tr>
<td>Aerodynamic coefficient ( C_a )</td>
<td>0.025</td>
</tr>
</tbody>
</table>
Figure 3 shows that OSR obtains a higher guarantee ration than EDF and Greedy. In various task scales, the guarantee ration obtained by OSR can be 12.84% higher than that of EDF and 8.89% higher than that of Greedy, which shows a very high scheduling performance. It can be seen in Figure 3(a) that as the quantity of tasks increases, the number of value tasks also increases gradually. However, the task guarantee ration shows a descending tendency. As the quantity of tasks increases, the number of tasks compatible with the airship observation also increases, so that more executable tasks appear. However, the observation capability of airship is limited. Thus, the increasing rate of value task will be much lower than the growth speed of task as the quantity of tasks increases to a certain degree. As for this, the task guarantee ration tends to decrease. As is shown in Figure 3(b), as the task quantity increases, the total energy consumption of airship also increases. However, when the task amount reaches the range of 30~120, the value task amount has a higher growth speed, leading to a sharp increase in energy consumption. By contrast, when task amount is around 120~300, the number of value tasks is nearly saturated, so that the growth of energy consumption slows down.

4.3. The Effectiveness of the EAS Algorithm. The EAS algorithm is used to realize energy-saving optimization of the scheduling scheme based on the deadline distribution of value tasks. In order to test the performance of this algorithm, its scheduling results will be compared with the performance before optimization (called HSA algorithm), that is, to compare the optimized solution with the efficient solution. According to Conclusion 1, the optimized solution has the same task guarantee ration as the efficient solution. As for this, the only parameter to be tested is the energy-saving performance of the EAS algorithm. The statistical indexes include the total energy consumption and energy consumption per tasks (ECPT).

It can be observed in Figure 4 that the total energy consumption of the EAS algorithm is always lower than that of the HSA algorithm in different task scales. This conclusion complies with Theorem 10. When keeping the same task guarantee ration, the total energy consumption of the EAS algorithm is 8.25% lower than that of the HSA algorithm, showing a higher energy saving performance. It can be seen in Figure 4(a) that the total energy consumption firstly increases sharply and then slows down as the task number increases. This phenomenon is in line with the analysis conclusion in Figure 3(b). The main reason for this trend is the variation tendency of the value task quantity. In Figure 4(b), we may find that ECPT reduces gradually as the task scale increases. As the area of battlefield is a constant, increase of the task scale will improve the density of task in the area. Therefore, the airship is able to execute more tasks within a short route, which in return reduces the energy consumption caused by cruising. In the meantime, we may find that the ECPT of the EAS algorithm is lower than that of the HSA algorithm in different task scales. This is because the EAS algorithm always adopts the most optimal cruising speed in each route, which reduces the energy consumption to the maximum degree.

5. Conclusions

The emergency scheduling problem on the high-altitude airship for imaging observation tasks is a multiobjective combination optimization issue. In research, this paper mainly made the following contributions

(1) The task execution process of airship is analyzed, and the method to detect the value tasks is provided. In this paper, a power-velocity model was also constructed by considering the influence of the wind field on airship’s cruising. Moreover, the programming model of this problem is presented by proposing
the optimal objectives and listing corresponding constraints.

(2) In order to simplify the solution process, a hierarchy optimization framework which divides the original problem into three subproblems is provided in the paper. As is shown in the experiment, this method is valid in reducing the solution space of the original problem, which is beneficial to efficiently obtain the scheduling scheme.

(3) The OSR algorithm is proposed to rank the priority execution order of tasks. The EDF algorithm and Greedy algorithm only use the deadline and cruising distance as the basis of task sorting. By contrast, the OSR algorithm considers both factors at the same time, so as to produce more reasonable results.

(4) The EAS algorithm is employed to optimize the scheduling scheme with minimum energy consumption as the objective. This algorithm can adjust the cruising speed of airship in each leg according to the deadline distribution of value tasks. As for this, the total energy consumption of airship can be reduced with no influence on the task guarantee ration. It is of great importance to extend the duration of airship in the observation activity.

Also for our future work, we plan to develop the cooperation scheduling problem of multiple airship in observation activity. This research will be performed based on the research in this paper, with the aim to integrate observation resources and to improve the overall observation efficiency.

References


