Hindawi
Complexity
Volume 2018, Article ID 6932985, 38 pages
https://doi.org/10.1155/2018/6932985

Research Article

A Proactive Robust Scheduling Method for Aircraft Carrier Flight Deck Operations with Stochastic Durations

Xichao Su,[1] Wei Han,[1] Yu Wu ,[2] Yong Zhang,[1] and Jie Liu[1]

[1] *Naval Aviation University, Yantai 264001, China*
[2] *College of Aerospace Engineering, Chongqing University, Chongqing 400044, China*

Correspondence should be addressed to Yu Wu; cquwuyu@cqu.edu.cn

Received 4 April 2018; Revised 5 August 2018; Accepted 14 August 2018; Published 1 November 2018

Academic Editor: Hassan Zargarzadeh

The operations on the aircraft carrier flight deck are carried out in a time-critical and resource-constrained environment with uncertainty, and it is of great significance to optimize the makespan and obtain a robust schedule and resource allocation plan for a greater sortie generation capacity and better operational management of an aircraft carrier. In this paper, a proactive robust optimization method for flight deck scheduling with stochastic operation durations is proposed. Firstly, an operation on node-flow (OONF) network is adopted to model the precedence relationships of multi-aircraft operations, and resource constraints categorized into personnel, support equipment, workstation space, and supply resource are taken into consideration. On this basis, a mathematical model of the robust scheduling problem for flight deck operation (RSPFDO) is established, and the goal is to maximize the probability of completing within the limitative makespan (PCLM) and minimize the weighted sum of expected makespan and variance of makespan (IRM). Then, in terms of proactive planning, both serial and parallel schedule generation schemes for baseline schedule and robust personnel allocation scheme and equipment allocation adjustment scheme for resource allocation are designed. In terms of executing schedules, an RSPFDO-oriented preconstraint scheduling policy (C^{PC}) is proposed. To optimize the baseline schedule and resource allocation, a hybrid teaching-learning-based optimization (HTLBO) algorithm is designed which integrates differential evolution operators, peak crossover operator, and learning-automata-based adaptive variable neighborhood search strategy. Simulation results shows that the HTLBO algorithm outperforms both some other state-of-the-art algorithms for deterministic cases and some existing algorithms for stochastic project scheduling, and the robustness of the flight deck operations can be improved with the proposed resource allocation schemes and C^{PC} policy.

1. Introduction

Aircraft carrier is the largest vehicle sailing on the sea, and the aircraft is the main weapon equipped on it. The aircraft can turn into fight only when a certain number of them with different types are introduced on the carrier, and the aircraft must be well maintained before they are launched by the catapult. In a sortie mission, a team of aircraft is required to be launched within a short period of time. As there are complex constraints in the flight deck operations, it is difficult to make a satisfactory schedule under limited space, resource, and personnel [1]. To be specific, on the flight deck, a series of operations, such as landing, towing, inspection, fueling, arming, oxygen filling, nitrogen charging, taxiing, and launching, are required to be performed within one deck cycle to keep high-intensity sortie rates [2]. Before the aircraft finishes launching on the flight deck, the pre-flight preparation takes up the most time of the deck cycle and is the most crucial stage as the failure and delay usually occur in this stage [3].

Compared to the ground operations of aircraft on the airport, more strict constraints are imposed on the operations on the flight deck of a carrier. Firstly, the narrow space of the flight deck results in more risks during operation, especially for the taxiing aircraft. Secondly, as the launching and landing of aircraft are proceeded on the flight deck in a cycling way, and the duration of a cycle is limited to $1+00$ (1 hour and 0 minutes), $1+15$, $1+30$, or $1+45$ [4], any delay may disturb the operating tempos under such a time-critical scheduling. Moreover, precedence constraints and resource constraints are more complicated in the flight deck operations of a carrier, and various kinds of aircraft make the process of flight deck operations diverse. Besides, there are many uncertain factors during the flight deck operations, such as the stochastic service time of each operation, the failure of aircraft and support equipment, and even the change of mission. Therefore, it is imperative to make a time-saving and robust schedule and resource allocation plan for flight deck operations with uncertainty which satisfy the constraints of space, resource, timing, and personnel as the efficiency and robustness of flight deck operations are always the bottleneck of improving the sortie generation capacity and operational management level of an aircraft carrier.

To address the challenge of robust scheduling problem for flight deck operation (RSPFDO), significant research has been devoted to the methods of reactive scheduling and stochastic scheduling, which ignore the plan of baseline schedule and resource allocation. In practice, however, a proactive plan of deck operations is needed in advance of execution to take as many types of uncertainty into consideration as possible. Since the scheduling problem for flight deck operations can be regarded as a typical resource-constrained (multi)project scheduling problem (RC(M)PSP) [5], methods from the related domain of robust project scheduling (RPS) [6] are referred to. Furthermore, some adaptive adjustment can be made based on those methods according to the practical constraints in the RSPFDO.

This paper focuses on the scheduling of flight deck operations for the pre-flight preparation stage, and the durations of stochastic activities are regarded as the main uncertainty which is represented by random variables with known distribution functions. A more comprehensive model for RSPFDO is established with the robust objectives of maximizing the probability of timely completion and minimizing the weighted sum of expectation and variance of makespan. Besides, a two-level proactive robust optimization framework for the RSPFDO is proposed under the consideration of the robustness of both temporal schedule and resource allocation. In the upper level, baseline schedules are optimized based on a hybrid teaching-learning-based optimization (HTLBO) algorithm. While in the lower level, a robust personnel allocation scheme and a robust equipment allocation adjustment scheme for robust resource allocation are designed corresponding to the baseline schedule of the upper level. Besides, a novel scheduling policy is proposed to cope with the execution of schedule with complicated resource constraints and stochastic durations of flight deck operations.

The main contributions of this research are as follows:

(1) The procedures, required resource, and uncertainty in durations of pre-flight preparation for a team of aircraft on flight deck are described in detail, and the RSPFDO is presented and is formulated as a mathematical programming model with the objectives of maximizing the probability of timely completion and minimizing the weighted sum of expectation and variance of makespan. The constraints of various kinds of resource and operations precedence relationships are taken into consideration in this model.

(2) A set of robust scheduling strategies for flight deck operations is proposed. Firstly, both serial and parallel schedule generation schemes for baseline schedule are presented. Then, a robust personnel allocation scheme and a robust equipment allocation adjustment scheme for a fixed baseline schedule are designed. Furthermore, a novel scheduling policy called preconstraint policy is developed to turn baseline schedules into executed schedules under complex constraints and stochastic durations.

(3) A proactive robust scheduling optimization algorithm based on HTLBO is proposed, which is integrated with the robust resource allocation schemes and the preconstraint policy. The HTLBO algorithm has better performance than other alternatives in terms of makespan under deterministic durations and robust objectives under stochastic durations.

The remainder of this paper is organized as follows. Section 2 presents a literature review of related work. In Section 3, a problem statement for the RSPFDO is given. A mathematical programming model aiming at the robustness of timely completion and makespan is formulated for the RSPFDO in Section 4. The proactive robust scheme formulation and scheduling strategy are proposed by Section 5. In Section 6, a proactive robust scheduling optimization algorithm based on HTLBO is developed. Section 7 is devoted to the presentation of computational experiments. Finally, this paper is concluded by Section 8.

2. Literature Review

2.1. Scheduling of Flight Deck Operations under Uncertainty. There are several variations of the scheduling model of flight deck operations according to different degrees of abstraction. Dastidar and Frazzoli [7] took the key support equipment as the path nodes and constructed a queueing network model of the flight deck operations. Yu et al. [8] established an extended flexible job shop scheduling model for flight deck scheduling considering the serial and parallel precedence relations between operations. Shi et al. [9] regarded the support groups as multistage machines, and the hybrid flow shop scheduling model was applied. However, all of these models simplified one or both of the two major factors. Firstly, the precedence relations of the flight deck operations are actually an activity on node network instead of serial relations. Secondly, for each operation, more than one kind of resource is required for execution. Hence, more recent research work took it as a resource-constrained (multi)project scheduling problem [5, 10, 11]. Apart from the schedule in RCPSP, allocation of renewable resources, such as labor and equipment, is an equally important decision-making object [12].

Researches on uncertain scheduling for flight operations have been a focus in recent years. Considering the complex deck operations and different distributions for activity times, Wu et al. [13] presented a GERT-based global sensitive method to analyze the uncertainty of deck operations. To reduce the influence caused by the uncertainties, methods in the existing literatures are categorized into two groups: reactive scheduling and stochastic scheduling. In terms of reactive scheduling, researchers from the MIT developed the Deck operations Course of Action Planner (DCAP) system to achieve a high-level interaction between human operators and automated planners during scheduling and planning. The DCAP system can act as an aid decision-making tool in a highly dynamic manned-unmanned environment [14]. Based on the DCAP simulation environment, Ryan et al. [10] proposed a conventional integer linear program-based planning algorithm for online scheduling, and the performance of the algorithm is inferior to that of expert user heuristics in the majority of test scenarios. Due to the advantage of coordinability and real-time capability, the multiagent-based model was utilized widely for dynamic scheduling on the flight deck in both manned [15, 16] and manned-unmanned environments [17]. In terms of response time, another efficient method called hierarchical task network (HTN) planning was applied in this field [18], and the method is based on an expert system in nature. In the taxiing phase which can be abstracted to obstacle avoidance and path planning problem, Wu and Qu [19] proposed a collision detection model and a dynamic multistep optimization algorithm to provide a real-time path under a static and dynamic deck environment. As for stochastic scheduling, based on the queueing network model of flight deck operations, Dastidar and Frazzoli [7] adopted a differential evolution algorithm to optimize the policies of stochastic scheduling in each node of the network. Michini and How [20] cast the scheduling of aircraft carrier flight deck operation in the MDP framework and transformed the scheduling experience of experts into a dynamic intelligent scheduling strategy based on reverse reinforcement learning. Feng et al. [21] built an improved direct graph to model the operation process and constraints and designed an improved ant colony optimization to optimize the scheduling strategy.

The main scheduling techniques to deal with uncertainty include reactive scheduling, stochastic scheduling, and robust scheduling. To the best of our knowledge, there is no prior research work addressing the proactive robust scheduling for flight deck operations of an aircraft carrier, while in the area of project scheduling with uncertainty, robust project scheduling has been a prominent direction and offers a new method for the RSPFDO with stochastic durations.

2.2. Proactive Project Scheduling under Uncertainty. At present, the research work of project scheduling under uncertainty mainly focuses on duration uncertainty, since the uncertainty of resource availability can be translated into the uncertainty of activity duration through resource slack [22]. According to the integrity of statistical data, uncertain activity durations are usually modeled as stochastic durations [23], fuzzy durations [24], and interval durations [25]. According to the approaches to deal with these uncertain durations, the proactive project scheduling under uncertainty falls into four broad categories: robust scheduling [6], stochastic project scheduling [23], fuzzy project scheduling [24], and interval project scheduling [25]. The fuzzy project scheduling approach and interval project scheduling approach can be utilized when the historical data of flight deck operations is insufficient. Therefore, only robust project scheduling and stochastic project scheduling are applicable for the flight deck operations with stochastic durations.

The main techniques for proactive robust project scheduling with uncertain activity durations contain the centralized buffer management based on critical chain [26], scattered buffer management [27], and robust resource allocation [28]. Pang et al. [29] proposed a unit activity slack algorithm which combined the scattered buffer heuristics algorithm and resource allocation algorithm to improve the solution robustness of project scheduling. Lamas and Demeulemeester [30] designed a new robustness measure, i.e., confidence level, which is completely independent of the reactive policy. They also built a new chance-constrained formulation and developed a branch-and-cut algorithm to solve a sample average approximation of the RCPSP with stochastic durations. Bruni et al. [31] proposed an adaptive robust optimization model to derive the resource allocation decisions that minimize the worst-case makespan under general polyhedral uncertainty sets. Ghoddousi et al. [32] applied a two-stage multiobjective buffer allocation approach to optimize the baseline. In the first stage, start time and buffer allocation of each activity are optimized with NSGA-II. In the second stage, the best schedule is selected from the Pareto optimal solutions based on Monte-Carlo simulation. Depending on different uncertainty characteristics, Chakrabortty et al. [33] developed six different heuristics to incorporate the uncertain duration as a deterministic constraint in a robust optimization model, and a Coin-Branch & Cut algorithm was adopted as the solver.

These methods offer good guidance for RPS of deck operations, but they still have deficiencies. The insertion of time buffers can help prevent the propagation of disruptions as much as possible throughout the schedule, but the time buffers are difficult to be determined, which the redundant buffers may result in a waste of time for operation execution. In addition, both the centralized and scattered buffer management neglect the robustness of resource allocation. Robust resource allocation, on the other side, pays more attention to the improvement of the algorithm. For example, MABO (myopic activity-based optimization) [28] combined with robust temporal schedule optimization is neglected.

Stochastic project scheduling is another proactive technique dealing with uncertain activity durations, but it suffers from the major drawback that no fixed baseline schedule is generated in advance of project execution. When the scheduling decisions are made dynamically, scheduling policies are usually used and act as the core mechanism, including the earliest-start policies (C^{ES}), the preselective policies (C^{PS}), the linear preselective policies (C^{LPS}), the resource-based policies (C^{RB}), the activity-based policies (C^{AB}), the preprocessor policies (C^{PP}), and the generalized preprocessor policies (C^{GP}) [34]. However, most of them are based on the minimal forbidden sets, except for C^{AB} and C^{RB}. As the number of minimal forbidden sets grows exponentially with

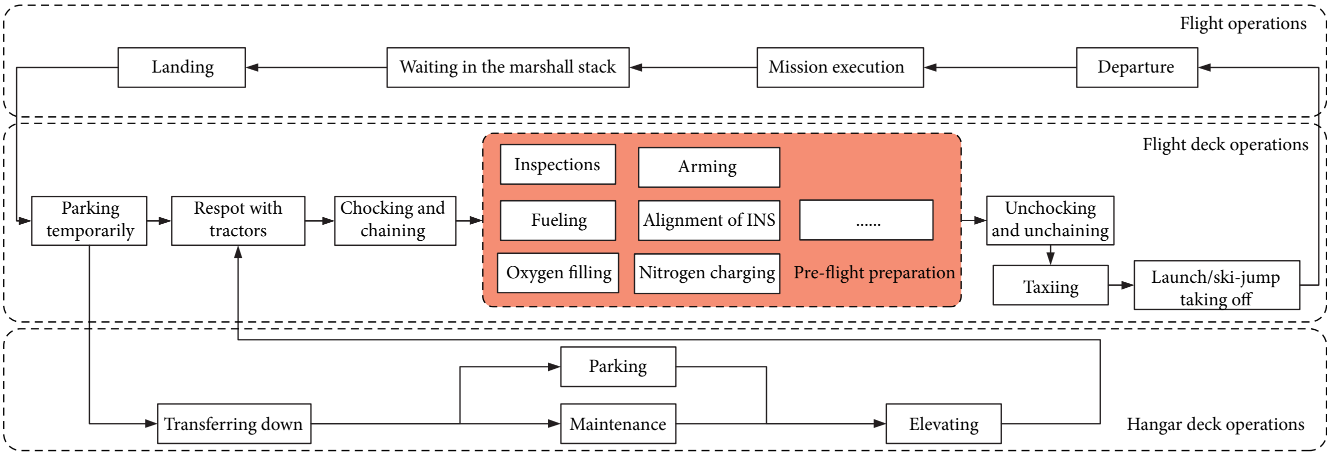


Figure 1: The procedure of aircraft carrier operations.

the increasing activities, the computational time will become unacceptable when dealing with practical large-scale project scheduling problems. Therefore, C^{AB} and C^{RB} become good choices for our problem. Apart from the scheduling policy, Tao et al. [35] studied the stochastic project scheduling problem with hierarchical alternatives and proposed the stochastic chance constraint model. To solve this problem, an artificial algae algorithm combined with sampling average approximation was designed. Li and Womer [36] designed closed-loop approximate dynamic programming algorithms based on the rollout (look-ahead) policy and lookup table (look-back) approach.

2.3. Solving Methods for RCPSPs. Due to the NP-hardness nature of RCPSP and its variations (RCPSPs for short), many techniques including exact methods, heuristics, metaheuristics, and hyperheuristics have been proposed. From the perspective of practicability and performance, the exact methods, represented by the branch-and-bound algorithm [37], are computationally expensive for large-scale scheduling problems though producing promising results. Priority rule-based heuristic [38], on the other side, is easy to be implemented and is computationally cheaper than exact methods. However, it is difficult to find an efficient rule that is well applicable for every RCPSP. As a consequence, metaheuristics provide the best trade-off between practicability and performance, and many metaheuristics have attracted the attention of researchers and shown promise in recent years including genetic algorithm [39, 40], differential evolution algorithm [41], particle swarm optimization, estimation of distribution algorithm [42, 43], and hybrid algorithms like consolidated optimization algorithm [44]. It is noteworthy that hybrid algorithms and hyperheuristics [45] which combine and make the most of different heuristics have attracted more attention recently.

Among these algorithms, the teaching-learning-based optimization (TLBO) algorithm [46] is a relatively new metaheuristic algorithm, which provides a general optimization framework that simulates the teaching-learning process including a teacher phase and a student phase. Thus far, the TLBO has been applied to various scheduling optimization problems, such as flow shop scheduling [47], job-shop scheduling [48], hydrothermal scheduling [49], steelmaking and continuous casting scheduling [50], and resource-constrained project scheduling [51]. The TLBO has shown to be more competitive than other metaheuristic algorithms. For example, the successful applications of the TLBO in stochastic RCPSP [52], interval RCPSP [51], and MS-RCPSP [53] have indicated that the TLBO outperforms many other algorithms for RCPSPs. However, to the best of our knowledge, existing research about the TLBO-based algorithm for the RCPSP is still rare, and no study is reported for robust project scheduling.

3. Problem Description of the RSPFDO

3.1. Description of Flight Deck Operations. The conduct of flight operations involves the intricate scheduling of aircraft, support equipment, and personnel. The types of aircraft in a wave of flight program include fighter aircraft, electronic warfare aircraft, early warning aircraft, antisubmarine warfare aircraft, and refueling aircraft, in the form of rotorcraft or fixed wing aircraft. The operation process complexity is closely related to the mission requirement and varies from different aircraft types. The typical procedure of aircraft carrier operations is shown in Figure 1.

During the daily flight deck operations, a team of aircraft starts with respot, which comes from either the air or the hangar deck. To respot an aircraft, a tractor will tow it from the temporal parking spot to the prespecified spot for pre-flight preparation. Once chocked and chained, the aircraft will turn to the pre-flight preparation stage, during which inspections of each part, fueling, arming, oxygen filling, nitrogen charging, alignment of INS (inertial navigation system), and so forth, are completed according to the technological process, namely, the specific precedence relationships for each aircraft. In the final stage, the aircraft will taxi to the spots for launch or ski-jump take-off. After the aircraft leaves the carrier, they execute specific missions in the air and will return to the carrier for the next cycle once the missions have been finished. It is obvious that the first and last stages of flight deck operations concern more about the

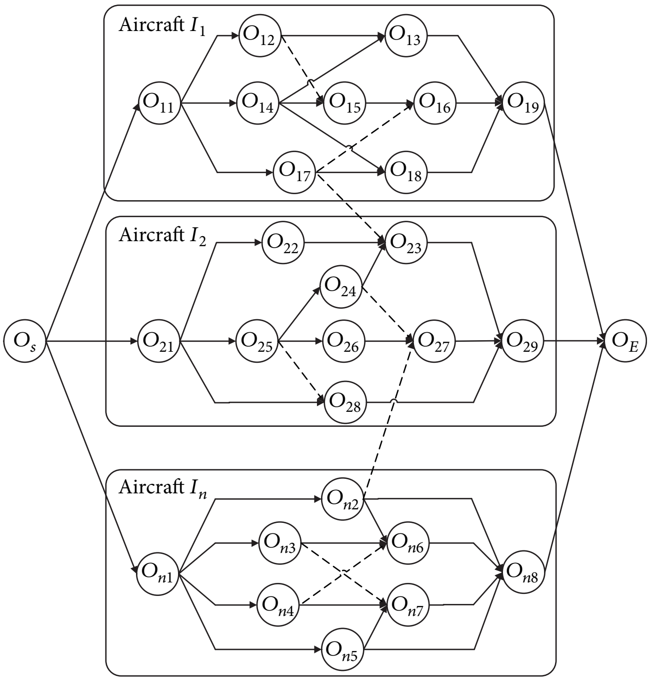


FIGURE 2: An example of the OONF network for multi-aircraft.

route planning problem, which has been intensively studied in [19, 54]. This paper concentrates on the scheduling problem of the pre-flight preparation stage, which is the key stage of flight deck operations and costs the most time and manpower.

Referring to RCMPSP, a single aircraft can be regarded as a project, and a team of aircraft (denoted as a set $I = \{1, 2, \ldots, N\}$) is considered in our study. For each aircraft $i \in I$, a set of real operations (denoted as J_i), a dummy start operation, and a dummy end operation are contained, and the aircraft must finish those operations before it can taxi to the launch position. Besides, a release time Er_i is defined for each aircraft, which represents the earliest start time for the set of operations J_i.

The precedence relationships of RCMPSP are usually modeled as activity on node network (AON). In an AON, each activity is represented by a node in a graph, and the precedence relationships between activities are denoted by the edges. As the resource allocation is considered in RSPFDO, an operation on node-flow (OONF) network is adopted for a team of aircraft, as shown in Figure 2.

In Figure 2, O_{ij} denotes the jth operation of ith aircraft, and O_S and O_E are the dummy start operation and end operation of the aircraft, respectively. In the OONF network, precedence relations of aircraft i are denoted as $D_i = (V_i, \mathrm{AN}_i)$, where V_i is the set of nodes presenting all the operations belonging to J_i, and AN_i is the set of solid arcs representing original precedence relations between operations. With the connection of O_S, O_E, and extra resource flows, D_i is integrated into $D = (V, \mathrm{AN} \cup \mathrm{AP} \cup \mathrm{AE} \cup \mathrm{AS})$, where V and AN indicate the sum of V_i and AN_i, respectively. AP, AE, and AS denote the extra precedence relations imposed by personnel, equipment, and workstation space, respectively, and the extra precedence relations are represented by dashed arcs in Figure 2. A resource flow from O_{ij} to O_{eg} is generated when there are some personnel or equipment allocated to the two operations, or the two operations are executed in the same workstation space, and the execution sequence is identified with the flow direction. It should be noted that the resource flow network can be changed according to the resource allocation plan.

As for each operation O_{ij}, a set of different kinds of resources are required, which will be discussed in Section 3.2. In addition, each operation O_{ij} has a duration $\tilde{d}_{ij}$, which is a stochastic value that follows a known probability distribution, and its expectation is relevant to the mission requirement while its variance is closely related to the uncertain environment (see Section 3.3).

3.2. Analysis of Support Resources. Resource constraints are the key factors that make it more difficult to conduct and schedule operations on the flight deck, compared to those on the airport ground. According to the different effects of these resources on the flight deck operations, the resource constraints are categorized into four groups as follows.

(1) Personnel Constraints. Personnel play a leading role in flight deck operation. Each operation requires personnel with specific trades which contain special equipment, machinery, ordnance, avionics, and so forth. For example, inspections of engines and airframe are done by machinists (i.e., personnel of machinery trade), while ammunition must be armed by ordnance personnel. In real situations, personnel are usually divided into group(s) according to specific personnel organization mode, and each group is responsible for a certain set of aircraft.

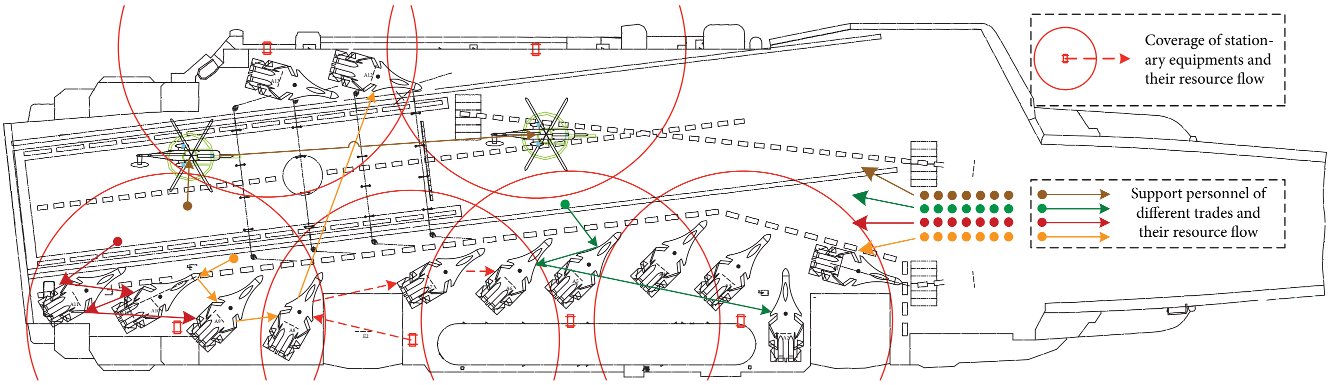

Coverage of stationary equipments and their resource flow
Support personnel of different trades and their resource flow

Figure 3: Resource flow on flight deck of the Admiral Kuznetsov aircraft carrier.

(2) Support Equipment Constraints. There are two types of support equipments on the flight deck, i.e., stationary equipments and mobile equipments. They are used to provide the aircraft with resources such as jet fuel, electric power, oxygen, nitrogen, and weapons. As the stationary equipments are fixed on the flight deck, they can only support the aircraft which are within the range of their pipelines. The mobile equipments can be regarded as stationary equipments whose support coverage is the whole area of the flight deck. Note that the number of both the stationary and mobile equipments on the flight deck is limited. Figure 3 shows a sketch map of resource (personnel and equipment) flow on flight deck of the Admiral Kuznetsov aircraft carrier.

(3) Workstation Space Constraints. This constraint is considered only for the workstations with limited space for operations. A typical instance is presented in Section 3.1 that as the space of cockpit can only accommodate at most one person, inspections of different trades must be done in sequence.

(4) Supply Resource Constraints. The resources are from the support equipments and are supplied for the aircraft. Assume that the resources are always sufficient during each deck cycle, and the constraint of each resource is the maximum number of aircraft that the resource can support concurrently. Take the jet fuel as an example, the maximum number of aircraft which the oil stations can support concurrently (the number is 4 on some aircraft carrier) is limited by the permitted total fueling pressure on the flight deck.

3.3. Analysis of Uncertain Durations. There are several key factors causing uncertainties in flight deck operations. First of all, as the personnel act as the main operators or participants in all the flight deck operations, the subjective factors (proficiency, degree of fatigue, motivation, etc.) and the objective factors make the durations of each operation uncertain. Secondly, the aircraft can become partly unavailable for a while due to mechanical or electronic problems. This makes the corresponding operations stagnant before the faults are repaired, and the repairing time is stochastic which is considered in the operation duration. It is noted that major failures that cannot be repaired timely are not taken into account because such cases will cause the interruption of operations and rescheduling. It is the same with the support equipments. As for the supply rates of resources (jet fuel, oxygen, nitrogen, etc.), they are also unsteady, and the supply duration for a certain quantity is stochastic. Finally, the aircraft executes different missions in the air, which make the resource consumption unknown. It is unnecessary to supply resources for the aircraft in every deck cycle, and the probability can be calculated by statistics. This case can be considered as n-fold Bernoulli trials where $n = 1$, and thus the operation duration is assumed to be a random variable which follows the Bernoulli distribution. In other cases of uncertainty, however, the duration uncertainty may be represented by random variables with different continuous general distributions (i.e., truncated normal distribution, uniform and exponential). The effect that uncertain durations exert on the makespan of a complex operation network can only be evaluated by simulations [55].

In general, the ultimate goal of RSPFDO is to maximize the probability of meeting the demand of the deck cycle while minimizing the expectation and variance of makespan, and the goal is reached by making a proactive robust schedule of deck operations for a team of aircraft under the complicated precedence constraints, resource constraints, and duration uncertainty. The framework for solving the RSPFDO can be illustrated in Figure 4.

4. Mathematical Formulation for RSPFDO

4.1. Problem Assumptions. The assumptions and conditions of RSPFDO are summarized as follows:

(1) Each operation cannot be interrupted once started, and the availability of personnel and support equipments remains unchanged during the whole operation.

(2) Each personnel or support equipment can work on at most one operation in each time period.

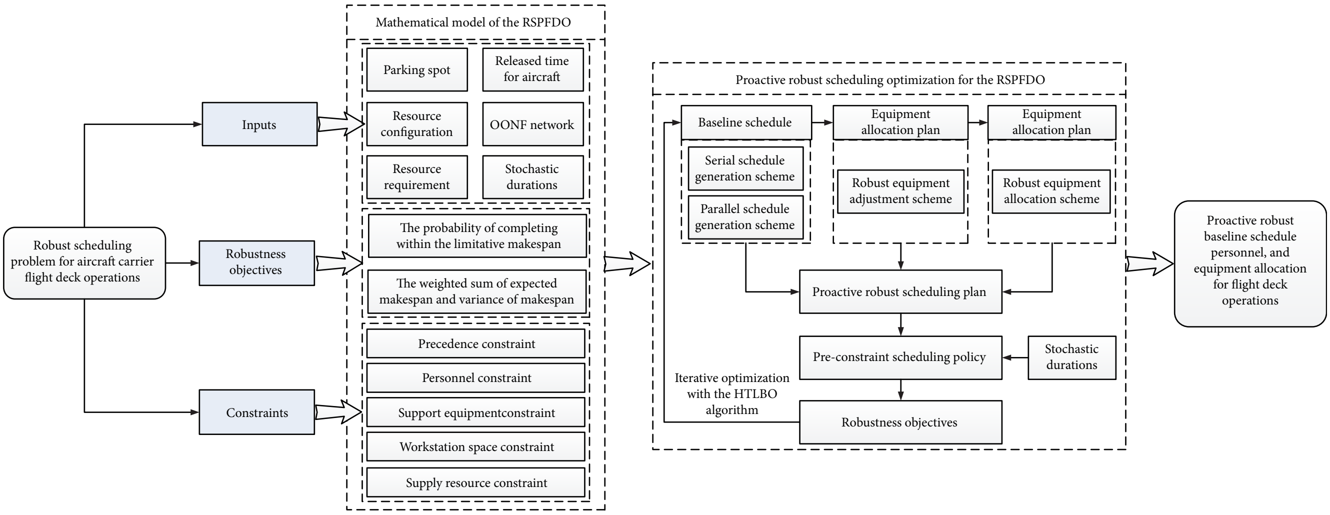

Mathematical model of the RSPFDO
Parking spot
Released time for aircraft
Resource configuration
OONF network
Resource requirement
Stochastic durations
The probability of completing within the limitative makespan
The weighted sum of expected makespan and variance of makespan
Precedence constraint
Personnel constraint
Support equipmentconstraint
Workstation space constraint
Supply resource constraint
Robust scheduling problem for aircraft carrier flight deck operations
Inputs
Robustness objectives
Constraints
Proactive robust scheduling optimization for the RSPFDO
Baseline schedule
Serial schedule generation scheme
Parallel schedule generation scheme
Equipment allocation plan
Robust equipment adjustment scheme
Equipment allocation plan
Robust equipment allocation scheme
Proactive robust scheduling plan
Pre-constraint scheduling policy
Stochastic durations
Iterative optimization with the HTLBO algorithm
Robustness objectives
Proactive robust baseline schedule personnel, and equipment allocation for flight deck operations

Figure 4: The framework for solving the RSPFDO.

Table 1: Parameter definition in the mathematical model.

Parameter	Definition
I	The set of aircraft, $I = \{1, 2, \ldots, Ni\}$
$p(i)$	Parking spot of aircraft i
J_i	The set of operations that are contained in the ith aircraft, $J_i = \{1, 2, \ldots, \lvert J_i \rvert\}$
J	The set of all the operations, $J = \{(i, j) \mid i \in Ij \in J_i\}$
A_t	The set of operations which are active in period t
A_{it}	The set of operations of aircraft i which are active in period t
O_{ij}	jth operation of ith aircraft, $i \in I, j \in J_i$
P_{ij}	The set of direct predecessors of operation O_{ij} in the OONF network $D' = (V, \mathrm{AN})$
Kp	The set of personnel trades types, $Kp = \{1, 2, \ldots, \lvert Kp \rvert\}$
Er_i	Release time of aircraft i
d_{ij}	Duration of operation O_{ij} used in the temporal schedule
Lp_k	The set of personnel with trade $k(k \in Kp)$, $Lp_k = \{1, 2, \ldots, \lvert Lp_k \rvert\}$
Ke	The set of support equipments types, $Ke = \{1, 2, \ldots, \lvert Ke \rvert\}$
Le_k	The set of support equipments of the type $k(k \in Ke)$, $Le_k = \{1, 2, \ldots, \lvert Le_k \rvert\}$
Ks	The set of workstation space types, $Ks = \{1, 2, \ldots, \lvert Ks \rvert\}$
Kw	The set of supply resource types, $Kw = \{1, 2, \ldots, \lvert Kw \rvert\}$
Lw_k^t	The maximum number of aircraft that supply resource of the type $k(k \in Kw)$ can support in period t
Rp_{kl}	The set of aircraft that the lth personnel with trade $k(k \in Kp)$ can be allocated to
me_{klp}	=1 if the lth support equipment of the type $k(k \in Ke)$ can reach the pth parking spot; =0 otherwise
Re_{kl}	The set of aircraft within the range that the lth support equipment of the type $k(k \in Ke)$ can reach, $\mathrm{Re}_{kl} = \{i \mid me_{klp(i)} = 1 i \in I\}$
rp_{ijk}	The number of personnel with trade $k(k \in Kp)$ required for performing operation O_{ij}
re_{ijk}	=1 if support equipment of the type $k(k \in Ke)$ is required for performing operation O_{ij}; =0 otherwise
rs_{ijk}	=1 if operation O_{ij} is performed in workstation space of the type $k(k \in Ks)$
rw_{jk}	=1 if supply resource of the type $k(k \in Kw)$ is required for operation O_{ij}; =0 otherwise
$\tilde{C}_{\max}$	Stochastic makespan of flight deck operations for a team of aircraft
$C^*_{\max}$	Limitative makespan of flight deck operations for a team of aircraft
$EC_{\max}$	Expectation of $\tilde{C}_{\max}$
$VC_{\max}$	Variance of $\tilde{C}_{\max}$

(3) The setup times and transfer times of personnel or support equipments between operations are short enough and can be negligible, compared to those of operations.

(4) The factors such as major faults of aircraft or equipments and changes of flight missions that may lead to the failure of origin schedule are not taken into consideration.

(5) The probability distribution of each operation is known by means of the historical data.

(6) The set of aircraft which each personnel is responsible for is known beforehand.

(7) All kinds of resources are available for the aircraft on each parking spot, so the movement of the aircraft is not considered.

4.2. Notations. The table parameters are used in our mathematical model (Table 1). In addition, a comprehensive description of notations and acronyms used in the paper is shown in Tables 2–5.

The table decision variables are used (Table 6).

4.3. Mathematical Expression of RSPFDO. Flight deck operations are time-critical because the aircraft are launching and landing in a cycling way, according to which a limitative makespan $C_{\max}^{*}$ is decided. It is significant to maintain the operating tempos for improving the combat effectiveness of the aircraft. Hence, the primary objective requires that the flight deck operations should be finished within the due makespan, otherwise the operating tempos of multiple waves of aircraft fleet will be broken, and the operation safety cannot be guaranteed. Besides, although temporal schedule is essential for flight deck operation, it is more like a reference, and we lay emphasis on the "quality robustness" instead of "solution robustness" [6], To this point, $EC_{\max}$ and $VC_{\max}$ are another two major criteria that the commanders concern. The combinatorial robust objective is expressed as

$$\min \left\{ EC_{\max} + \omega VC_{\max} \mid \max \left(\Pr \left(\tilde{C}_{\max} \le {C_{\max}}^{*} \right) \right) \right\}, \quad (1)$$

where ω represents the weight between $EC_{\max}$ and $VC_{\max}$, and the balance between aggressiveness and conservativeness of schedule can be adjusted by this factor. $EC_{\max} + \omega VC_{\max}$ represents the integrated robustness of makespan (IRM), and $\Pr\,(\tilde{C}_{\max} \le {C_{\max}}^{*})$ denotes the probability of completion within the limitative makespan (PCLM). The two objectives are considered in lexicographical order, where PCLM is the primary objective, while IRM is used as a tie breaker.

Since the durations of operations are stochastic, the criteria in the objectives are implicit functions of the five decision variables, i.e., a probability mapping relation between the decision variables and the objectives. To be specific, given the values of decision variables, the baseline schedule and resource allocation plan is constructed under specific constraints. Then, through Monte-Carlo simulation with multiple independent sampling of the uncertain durations $\tilde{d}_{ij}$, the statistic values of $EC_{\max}$, $VC_{\max}$, and $\Pr\,(\tilde{C}_{\max} \le {C_{\max}}^{*})$ of the plan are obtained, and the detailed process is introduced in Section 5.2.

The constraints of RSPFDO can be expressed by the following formulas.

$$S_{i1} \ge Er_i, \quad \forall i \in I, \quad (2)$$

$$S_{ij} \ge S_{ih} + d_{ih}, \quad \forall (i,h) \in P_{ij}, \forall i \in I, \forall j \in J_i, \quad (3)$$

$$S_{ij} + d_{ij} \le S_{eg} + BM \cdot \left(1 - Y_{ijeg}\right), \quad \forall i, e \in I, \forall j \in J_i, \forall g \in J_e, \quad (4)$$

$$\sum_{i\in I} \sum_{j\in A_t} rp_{ijk} \le |Lp_k|, \quad \forall k \in Kp, \forall t > 0, \quad (5)$$

$$\sum_{i\in I} \sum_{j\in A_t} re_{ijk} \le |Le_k|, \quad \forall k \in Ke, \forall t > 0, \quad (6)$$

$$\sum_{j\in A_{it}} rs_{ijk} \le 1, \quad \forall i \in I, \forall k \in Ks, \forall t > 0, \quad (7)$$

$$\sum_{i\in I} \sum_{j\in A_t} rw_{ijk} \le Lw_k^t, \quad \forall k \in Kw, \forall t > 0, \quad (8)$$

$$\sum_{i\in I-\mathrm{Re}_{kl}} \sum_{j\in J_i} Xe_{ijkl} + \sum_{i\in I-Rp_{k'l'}} \sum_{j\in J_i} Xp_{ijk'l'} = 0, \quad (9)$$
$$\forall k \in Ke, \forall l \in Le_k, \forall k' \in Kp, \forall l' \in Lp_k,$$

$$\sum_{l\in L\mathrm{p}_k} Xp_{ijkl} = rp_{ijk}, \quad \forall i \in I, \forall j \in J_i, \forall k \in Kp, \quad (10)$$

$$\sum_{l\in Le_k} Xe_{ijkl} = re_{ijk}, \quad \forall i \in I, \forall j \in J_i, \forall k \in Ke. \quad (11)$$

Formulas (2), (3), and (4) correspond to temporal scheduling constraints, while Formulas (5), (6), (7), (8), (9), (10), and (11) represent resource scheduling constraints. To be specific, Constraint (2) indicates that aircraft i cannot start its operations until it is chocked and chained at the released time Er_i; S_{i1} represents the start time of dummy start operation of aircraft i and actually the start time of aircraft i. Constraint (3) denotes the precedence relationships of operations, and operation O_{ij} cannot be executed earlier than its direct predecessors. Hence, the start time of O_{ij} is enforced to be larger than those of operations in set P_{ij}. Constraint (4) ensures that if two operations O_{ij} and O_{eg} need the same resources (personnel, equipment, and workstation space) that $Y_{ijeg} = 1$, the operation with higher priority O_{eg} is processed firstly and the start time of O_{eg} is enforced to be larger than that of O_{ij}, where BM is a large number to ensure that the inequality always holds even when $Y_{ijeg} \ne 1$. Here, considering the duration of flight deck operations, $BM = 10^3$ is large enough. Constraints (5) and (6) set the limitation on the personnel with trade $k(k \in Kp)$ and the equipments of type $k(k \in Ke)$ demanded by operation O_{ij} at time t to the available capacity, respectively. Constraint (7) denotes that for each aircraft i, the number of active

Table 2: Notation for model expression.

Parameter/ variable	Definition
I	The set of aircraft, $I = \{1, 2, \dots, Ni\}$
$p(i)$	Parking spot of aircraft i
J_i	The set of operations that are contained in the ith aircraft, $J_i = \{1, 2, \dots, \lvert J_i \rvert\}$
J	The set of all the operations, $J = \{(i, j) \mid i \in Ij \in J_i\}$
D_i	The precedence relations of operations of aircraft i
D	The precedence relations of operations of all the aircraft
V	The set of nodes in OONF network D presenting all the operations
AN	The set of solid arcs in OONF network D representing precedence relations between operations
AP	The extra precedence relations imposed by personnel
AE	The extra precedence relations imposed by equipment
AS	The extra precedence relations imposed by workstation space
A_t	The set of operations which are active in period t
A_{it}	The set of operations of aircraft i which are active in period t
O_{ij}	jth operation of ith aircraft, $i \in I, j \in J_i$
P_{ij}	The set of direct predecessors of operation O_{ij} in the OONF network $D' = (V, \mathrm{AN})$
Kp	The set of personnel trades types, $Kp = \{1, 2, \dots, \lvert Kp \rvert\}$
Er_i	The released time of aircraft i
$\tilde{d}_{ij}$	The stochastic duration of operation O_{ij}
d_{ij}	Duration of operation O_{ij} used in the temporal schedule
Lp_k	The set of personnel with trade $k(k \in Kp)$, $Lp_k = \{1, 2, \dots, \lvert Lp_k \rvert\}$
Ke	The set of support equipments types, $Ke = \{1, 2, \dots, \lvert Ke \rvert\}$
Le_k	The set of support equipments of the type $k(k \in Ke)$, $Le_k = \{1, 2, \dots, \lvert Le_k \rvert\}$
Ks	The set of workstation space types, $Ks = \{1, 2, \dots, \lvert Ks \rvert\}$
Kw	Set of supply resource types, $Kw = \{1, 2, \dots, \lvert Kw \rvert\}$
Lw_k^t	The maximum number of aircraft that supply resource of the type $k(k \in Kw)$ can support in period t
Rp_{kl}	The set of aircraft that the lth personnel with trade $k(k \in Kp)$ can be allocated to

Table 2: Continued.

Parameter/ variable	Definition
me_{klp}	=1 if the lth support equipment of the type $k(k \in Ke)$ can reach the pth parking spot; =0 otherwise
Re_{kl}	The set of aircraft within the range that the lth support equipment of the type $k(k \in Ke)$ can reach, $\mathrm{Re}_{kl} = \{i \mid me_{klp(i)} = 1 i \in I\}$
rp_{ijk}	The number of personnel with trade $k(k \in Kp)$ required for performing operation O_{ij}
re_{ijk}	=1 if support equipment of the type $k(k \in Ke)$ is required for performing operation O_{ij}; =0 otherwise
rs_{ijk}	=1 if operation O_{ij} is performed in workstation space of the type $k(k \in Ks)$
rw_{jk}	=1 if supply resource of the type $k(k \in Kw)$ is required for operation O_{ij}; =0 otherwise
$\tilde{C}_{\max}$	Stochastic makespan of flight deck perations for a team of aircraft
$C^*_{\max}$	Limitative makespan of flight deck operations for a team of aircraft
$EC_{\max}$	Expectation of $\tilde{C}_{\max}$
$VC_{\max}$	Variance of $\tilde{C}_{\max}$
$\Pr\ (\tilde{C}_{\max} \le C_{\max}{}^*)$	The probability of completion within the limitative makespan
ω	The weight between $EC_{\max}$ and $VC_{\max}$
BM	A large number
S_{ij}	Starting time of operation O_{ij} in the temporal schedule
E_{ij}	Completion time of operation O_{ij} in the temporal schedule
Xp_{ijkl}	=1 if operation O_{ij} is allocated to personnel $l(l \in Lp_k)$ with trade $k(k \in Kp)$; =0 otherwise
Xe_{ijkl}	=1 if operation O_{ij} is allocated to equipment $l(l \in Le_k)$ of the type $k(k \in Ke)$; =0 otherwise
Y_{ijeg}	=1 if operation O_{ij} precedes operation O_{eg} when allocated to the same resources; =0 otherwise

operations at time t that are executed in the workstation space of the kth type is less than 1, thus ensuring that operations that require the same kind of workstation space cannot be in progress at the same time instant t. Constraint (8) states that the total number of aircraft on flight deck that are supported by supply resource of type $k(k \in Kw)$ concurrently at time t is less than the maximum number limitation Lw_k^t. Constraint (9) ensures that the first and second items on the left side of the equation are equal to zero, i.e., the decision variables of personnel allocation and equipment allocation are always equal to zero when the aircraft are outside the range of the personnel or equipment, thus guaranteeing that personnel and support equipments can only be allocated to those aircraft within their range. Constraints (10) and (11) together make sure that the number of personnel or equipment required for performing operation O_{ij} is equal to the total number of personnel or equipment allocated to it, respectively.

5. Proactive Robust Scheme Formulation and Scheduling Policy

Scheduling and sequencing are concerned with the optimal allocation of scarce resources to activities over time. As far as the stochastic operation duration is concerned, some proactive procedures and proper scheduling policies should be taken into account. The proactive procedures can protect the baseline schedule against time uncertainty, and the proper scheduling policies are applied for instant decision during operation execution. There are three key elements that affect the performance of RSPFDO: baseline schedule, resource allocation, and scheduling policy. In this section, a three-stage process is proposed to cope with duration uncertainty:

(1) Generation of a precedence-feasible and resource-feasible baseline schedule. The baseline schedule is very crucial for the operations of each aircraft, in which the planned starting and finishing times are given. Note that the baseline schedule is generated along with an initial feasible equipment allocation plan because the stationary equipments can work only within a certain range and cannot be shared. This feature makes the stationary equipments different from the personnel which are replaceable with the same trade in group.

(2) Allocation of personnel and support equipment. After the baseline schedule and the equipment allocation plan are determined, the personnel allocation plan is generated accordingly to improve the robustness. A resource flow network is also formulated through the processes of resource allocation. The allocation decision is based on the simple principle which was verified in [28]. That is, a more stable scheme can be obtained when the number of extra precedence relations imposed by resource flows is reduced and the time difference between operations is increased.

(3) Stochastic evaluation with scheduling policy. The flight deck operations of aircraft are treated as a multistage decision-making process, and the scheduling policy acts as a rule at each decision point to determine which operations are to be started next. Note that the decision-making process is dynamical and unpredictable due to the stochastic operation durations, and a novel preconstraint policy is proposed to deal with the complicated constraints of flight deck operations.

Next, the above three-stage process will be elaborated in the subsequent three sections respectively.

5.1. Generation of Baseline Schedule. The baseline schedule with deterministic durations is usually generated by a serial schedule generation scheme (SSGS) or parallel schedule generation scheme (PSGS), in the way of either forward scheduling or backward scheduling. The performance of SSGS and PSGS must be evaluated in a specific problem, but the SSGS outperforms the latter in general. Besides, forward scheduling is usually adopted as it conforms to the convention that activities should be begun as soon as possible. In this paper, forward scheduling and backward scheduling are combined to reduce the makespan [56].

5.1.1. Serial Schedule Generation Scheme. The SSGS is operation-oriented and builds the schedule in n stages with n operations. At each stage g, an operation is selected and scheduled as early as possible such that precedence and resource feasibility are given. The sequence in which operations are considered is determined by a priority value. Let x_{ij} be the priority value of operation O_{ij}; a smaller value of x_{ij} means greater priority. To generate a baseline schedule (denoted as $\{[S_{ij}, E_{ij}]\}(i \in I, j \in J_i)$), the expectation of duration $\tilde{d}_{ij}$ of each operation O_{ij} is selected as the deterministic duration d_{ij}. As to the initial equipment allocation, the rule called "the minimum total processing time remaining in covering area" (MTRCA) [11] is used at every decision point that the equipments are in demand. Specifically, let S_g be the set of scheduled operations at stage g; the set of eligible operations within the range of the lth support equipment of type $k(k \in Ke)$ is defined as $Je_{kl} = \{(i, j) \mid (i, j) \in \bar{S}_g, re_{ijk} > 0, i \in \mathrm{Re}_{kl}\}$, and the total operation time in the set is represented by $TR_{kl} = \sum_{(i,j) \in Je_{kl}} d_{ij}$. At each decision point, the equipment with the minimum TR_{kl} is selected by the MTRCA rule.

To design the SSGS for flight deck operations of aircraft, some other parameters are also defined as follows.

ES_{ij} The earliest precedence-feasible starting time of operation O_{ij}

SP_{ij} The earliest personnel-feasible starting time of operation O_{ij}

SE_{ij} The earliest equipment-feasible starting time of operation O_{ij}

SS_{ij} The earliest space-feasible starting time of operation O_{ij}

SW_{ij} The earliest supply resource-feasible starting time of operation O_{ij}

ERS_{ij} The earliest precedence-resource-feasible starting time of operation O_{ij}

Up_{ik}^{t} The number of available personnel with trade $k(k \in Kp)$ that can support aircraft i in period t

Ue_{ik}^{t} The set of available equipments of type $k(k \in Ke)$ that can reach and support aircraft i in period t

Table 3: Notation for scheduling and allocation scheme.

Parameter/ variable	Definition
x_{ij}	The priority value of operation O_{ij}
S_g	The set of scheduled operations at stage g
Je_{kl}	The set of eligible operations within the range of the lth support equipment of type k
TR_{kl}	The total time of operations in the set Je_{kl}
ES_{ij}	The earliest precedence-feasible starting time of operation O_{ij}
SP_{ij}	The earliest personnel-feasible starting time of operation O_{ij}
SE_{ij}	The earliest equipment-feasible starting time of operation O_{ij}
SS_{ij}	The earliest space-feasible starting time of operation O_{ij}
SW_{ij}	The earliest supply resource-feasible starting time of operation O_{ij}
ERS_{ij}	The earliest precedence-resource-feasible starting time of operation O_{ij}
Up^t_{ik}	The number of available personnel with trade $k($ $k \in Kp)$ that can support aircraft i in period t
Ue^t_{ik}	The set of available equipments of the type $k(k \in Ke)$ that can reach and support aircraft i in period t
Us^t_{ik}	=1 if workstation space of the type $k(k \in Ks)$ in aircraft i in period t is occupied; =0 otherwise
Uw^t_k	The remaining number of aircraft that supply resource of the type $k(k \in Kw)$ can support in period t
D_g	The set of eligible operations at stage g
(i^*, j^*)	The index of operation with the highest priority during each iteration
A_g	The set of operations active in period t_g of stage g
C_g	The set of completed operations at scheduling time t_g of stage g
$D'_g(t_g)$	The set of operations which are eligible both for precedence and resource at scheduling time t_g of stage g
S^{R}_{ij}	The starting time of operation O_{ij} in the right-justified schedule
E^{R}_{ij}	The completion time of operation O_{ij} in the right-justified schedule
ΔT^{L}	The span between left-justified schedule and right-justified schedule
PF_{ijeg}	The pairwise float between the start of operation O_{eg} and the end of O_{ij}
P^{s}_{eg}	The set of direct predecessors of operation O_{eg} in the OONF network $D'' = (V, \text{AN} \bigcup \text{AS})$
Q_k	The set of operation pairs which possess the same equipment of the type $k(k \in Ke)$
Le^l_k	The lth support equipment of the type $k(k \in Ke)$
$\text{LRO}^{\text{n}}_{ij}$	The local robustness index of operation O_{ij} before adjustment
PF^f_{ij}	The pairwise float between operation O_{ij} and its predecessor in the same equipment
PF^b_{ij}	The pairwise float between operation O_{ij} and its successors in the same equipment
$\text{LR}E^{\text{n}}_{kl}$	The local robustness index before adjustment of the Le^l_k which is reachable for operation O_{ij}
$\text{LRO}^{\text{c}}_{ij}$	The local robustness index of operation O_{ij} after the adjustment
$\text{LR}E^{\text{c}}_{kl'}$	The local robustness index after adjustment of Le^l_k that operation O_{ij} originally adheres to
ΔRE_{kl}	The difference of the local robustness index of Le^l_k between after and before adjustment
N^k_{justment}	The number of adjustment in an iteration for equipment of the type $k(k \in Ke)$
RCF_{ij}	The free slack of operation O_{ij}
IF_{ijkl}	The float time between operation O_{ij} and operations adhere to the lth personnel with trade $k(k \in Kp)$
LFT_{kl}	The completion time of O_{eg} corresponds to the latest completion time of Lp^l_k
P^{se}_{ij}	The set of direct predecessors of operation O_{ij} in the OONF network $D''' = (V, \text{AN} \bigcup \text{AS} \bigcup \text{AE})$
RP_{ijkl}	Robustness index of the lth personnel with trade k for operation O_{ij}
$Up_k(t, i)$	The set of available personnel that can support aircraft i at decision point t
$\tilde{S}_{ij}$	The stochastic starting time of operation O_{ij} in the temporal schedule

Us^t_{ik} =1 if workstation space of type $k(k \in Ks)$ in aircraft i in period t is occupied; =0 otherwise

Uw^t_k The remaining number of aircraft that supply resource of type $k(k \in Kw)$ can support in period t

D_g The set of eligible operations at stage g

(i^*, j^*) Index of operation with the highest priority during each iteration.

The overall process of the proposed SSGS for aircraft deck operations is derived from the conventional one for RCPSP in [57], and the difference lies in the complexity of finding the earliest precedence-resource-feasible starting time and selecting the initial equipment allocation for each

Table 4: Notation for optimization algorithm.

Parameter/variable	Definition
$\mathbf{x}$	The representation of individual in population
$\mathbf{x}^{\mathrm{L}}$	The vector of $\mathbf{x}$ that is used for generating a left-justified schedule
$\mathbf{x}^{\mathrm{R}}$	The vector of $\mathbf{x}$ that is used for generating a right-justified schedule
$\mathbf{D}_k$	The kth scenarios of durations by random sampling
Ns	The number of replications for evaluation an individual
Rf	The integrated fitness value
Np	The size of the population
Nt	The number of the elite individuals in the teacher group
G	The generation of evolution
E	The eligible operation set
η_{ij}	The probability of being chosen for eligible operation O_{ij}
μ_{eg}	The priority value of operation O_{eg} based on the LFT rule
LFT_{ij}	The latest finish time of operation O_{ij}
$\mathbf{x}_{i,G}$	The ith target vector at the Gth generation
$\mathbf{v}_{i,G}$	The ith donor vector at the Gth generation
$\mathbf{x}^{\mathrm{r}}_{t\mathrm{best},G}$	A individual randomly selected from the teacher group
F_i	The scale factor for the ith individual
Cr_i	The crossover probability for the ith individual
M_{F}	The mean values of F_i
M_{Cr}	The mean values of Cr_i
S_{F}	The set of the successful scale factors
S_{Cr}	The set of the successful crossover probabilities
ω_{F}	The weight factor of scale factor
ω_{Cr}	The weight factor of crossover probabilities
$\mathrm{mean}_{WA}(S_{\mathrm{F}})$	The mean of all successful scale factors
$\mathrm{mean}_{WA}(S_{\mathrm{Cr}})$	The mean of all successful crossover probabilities
ΔRf_m	The improvement of fitness
IRUR	The integrated resource utilization ratio
λ_k^{p}	The weight of type k of personnel in IRUR
λ_k^{e}	The weight of type k of support equipment in IRUR
λ_k^{s}	The weight of type k of workstation space in IRUR
λ_k^{w}	The weight of type k of supply resource in IRUR
TIRU	The total integrated resource utilization
$tp_1(S)$	The start point of the peak for schedule S
$tp_2(S)$	The end point of the peak for schedule S
es_{ij}	The earliest start time of operation O_{ij} in the schedule
ls_{ij}	The latest start time of operation O_{ij} in the schedule
Na	A random number of local operations
p_{ij}^{r}	The selection probability for operation O_{ij} in the reshuffle operations
$\mathrm{center}(i)$	The center of start time of all the aircraft
$\Delta\mathrm{centre}$	The difference of center of two aircraft
$pl_i(g)$	The probability of selecting the ith neighborhood at the gth iteration
α_{reward}	The reward parameter
$\alpha_{\mathrm{penalty}}$	The penalty parameter
Nl	The number of neighborhoods for selection
$\beta(g)$	The reward received by the reinforcement feedback for an neighborhood selected at iteration g
$N_{\mathrm{iter}}(n_{\mathrm{gs}})$	The iteration number for performing the reinforcement phase
$N_{\mathrm{iter}}^{\mathrm{lb}}$	The lower bound of the iteration number N_{iter}
$N_{\mathrm{iter}}^{\mathrm{ub}}$	The upper bound of the iteration number N_{iter}
$R\_C_{\max}$	The response of algorithm under different mission cases and deterministic durations
p_j^{mc}	The probability of occurrence of jth mission case
$C_{\max}^{j}$	The makespan of the jth mission case
N_{mc}	The number of mission case
Lp_k^l	The lth personnel of the trade $k(k \in Ke)$
PCLM_i	The responses of PCLM with the ith distribution
IRM_i	The responses of IRM with the ith distribution

operation. The serial schedule generation scheme for aircraft deck operations is presented in Algorithm 1.

Algorithm 1 starts out with an initialization of the problem. All the state parameters of different kinds of resource are set as initial capacity or idle state. Note that S_g is initialized with the dummy start operations of each aircraft, and the start time of S_g is set as the released time of the corresponding aircraft. Afterwards, the schedule is generated by iteratively choosing the eligible operation with the highest priority, and the operation is arranged as early as possible and is allocated support equipment.

To be specific, at each stage g, the set of eligible operations is calculated, from which an operation (i^*, j^*) is

Table 5: Acronym definition.

Acronyms	Definition
OONF	Operation on node-flow
RSPFDO	The robust scheduling problem for flight deck operations
PCLM	The probability of completing within the limitative makespan
IRM	The weighted sum of expected makespan and variance of makespan
HTLBO	Hybrid teaching-learning-based optimization algorithm
RC(M)PSP	Resource-constrained (multi)project scheduling problem
RPS	Robust project scheduling
DCAP	The Deck operations Course of Action Planner
HTN	Hierarchical task network
MABO	Myopic activity-based optimization
C^{ES}	The earliest-start policies
C^{PS}	The preselective policies
C^{LPS}	The linear preselective policies
C^{RB}	The resource-based policies
C^{AB}	The activity-based policies
C^{PP}	The preprocessor policies
C^{GP}	The generalized preprocessor policies
RCPSPs	RCPSP and its variations
TLBO	The teaching-learning-based optimization
INS	Inertial navigation system
AON	Activity on node network
SSGS	Serial schedule generation scheme
PSGS	Parallel schedule generation scheme
MTRCA	The minimum total processing time remaining in covering area
DJ	Double justification
CC/BM	Critical chain/buffer management
C^{PC}	Preconstraint policy
AL	Activity list
RK	Random key
LJ	Left-justified
RJ	Right-justified
LFT	The latest finish time rule
DE	The differential evolution
RUR	The resource utilization ratio
IRUR	The integrated resource utilization ratio
AVNS	Adaptive variable neighborhood search
LA	Learning automata
T-normal	The truncated normal distribution
d-LB	The lower bound of duration
d-UB	The upper bound of duration
AP	Addition parameter
MMGA	The multimodal genetic algorithm
MDE	The modified differential evolution algorithm
IPSO	The improved particle swarm optimization algorithm
HEDA	The hybrid estimation of distribution algorithm
ABGA	The genetic algorithm with C^{AB} policy
ABGR	The greedy randomized adaptive search procedure with C^{AB} policy
RBEDA	The estimation of distribution algorithm with C^{RB} policy
UAV	The upper adjacent value
LAV	The lower adjacent value
IQR	The minimum interquartile range

Table 5: Continued.

Table 6

Decision variable	Definition
S_{ij}	The starting time of operation O_{ij} in the temporal schedule
E_{ij}	The completion time of operation O_{ij} in the temporal schedule
Xp_{ijkl}	=1 if operation O_{ij} is allocated to personnel $l(l \in Lp_k)$ with trade $k(k \in Kp)$; =0 otherwise
Xe_{ijkl}	=1 if operation O_{ij} is allocated to equipment $l(l \in Le_k)$ of the type $k(k \in Ke)$; =0 otherwise
Y_{ijeg}	=1 if operation O_{ij} precedes operation O_{eg} when allocated to the same resources; =0 otherwise

chosen according to the priority value (see Section 6 for optimization of the value). Then, the earliest precedence-resource-feasible starting time of operation $O_{i^* j^*}$ is initialized by the earliest precedence-feasible starting time of it. To arrange the operation which satisfies the constraints of personnel, equipment, workstation space, and supply resource in the duration, the time $\mathrm{ERS}_{i^* j^*}$ is postponed until the earliest resource-feasible starting time of the four kinds of resource are equal $\mathrm{SP}_{i^* j^*} = \mathrm{SE}_{i^* j^*} = \mathrm{SS}_{i^* j^*} = \mathrm{SW}_{i^* j^*}$. Afterwards, the start/finish time of the operation is set, and the required equipments are allocated according to the MTRCA rule. At the end of this loop, all the state parameters and set S_g are updated for the next scheduling.

5.1.2. Parallel Schedule Generation Scheme. Different from the SSGS, a feasible schedule is created by considering increasing decision time t in the time-oriented PSGS. At each decision time t, as many operations as possible are started such that precedence and resource feasibility are given. As the same as that in SSGS, the sequence in which operations are considered is determined by a priority value.

Except for the parameter definitions listed in Section 5.1.1, some additional parameters are used as follows. A_g is assumed to be the set of operations active in period t_g of stage g, and C_g denotes the set of completed operations at scheduling time t_g of stage g. Different from D_g, the set of operations

```
Input: Priority of support operations $x_{ij}$, initial precedence relation $D' = (V, \mathrm{AN})$.
Output: Temporal schedule $\{[S_{ij}, E_{ij}]\}$, initial equipment allocation plan $\{Xe_{ijkl}\}$.
Initialize $Up^t_{ik}, Ue^t_{ik'}, Us^t_{ik''}, Uw^t_{k'''}, S_g := \bigcup_{i\in I}(i, 1), S_{i1} := Er_i, E_{i1} := Er_i$.
While $|S_g| < |J|$
    Calculate $D_g := \{(i, j) \mid (i, j) \notin S_g, P_{ij} \subseteq S_g\}$ according to $D'$.
    $(i^*, j^*) := \min_{O_{ij}\in D_g}\{(i, j) \mid x_{ij} = \inf_{O_{pq}\in D_g}(x_{pq})\}$.
    $ES_{i^*j^*} := \max\{E_{ij} \mid (i, j) \in P_{i^*j^*}\}$, initialize $ERS_{i^*j^*} := ES_{i^*j^*}$.
    Repeat
        $SP_{i^*j^*} := \min\{t \mid t \ge ERS_{i^*j^*}, rp_{i^*j^*k} \le Up^\tau_{i^*k}, \tau = [t, t + Ed_{i^*j^*}), (\forall k \in Kp)\}$,
        $SE_{i^*j^*} := \min\{t \mid t \ge ERS_{i^*j^*}, re_{i^*j^*k'} \le |Ue^\tau_{i^*k'}|, \tau = [t, t + Ed_{i^*j^*}), (\forall k' \in Ke)\}$,
        $SS_{i^*j^*} := \min\{t \mid t \ge ERS_{i^*j^*}, Us^\tau_{i^*k''} \cdot rs_{i^*j^*k''} = 0, \tau = [t, t + Ed_{i^*j^*}), (\forall k'' \in Ks)\}$,
        $SW_{i^*j^*} := \min\{t \mid t \ge ERS_{i^*j^*}, rw_{i^*j^*k'''} \le Uw^\tau_{i^*k'''}, \tau = [t, t + Ed_{i^*j^*}), (\forall k''' \in Kw)\}$.
        $ERS_{i^*j^*} := \max(SP_{i^*j^*}, SE_{i^*j^*}, SS_{i^*j^*}, SW_{i^*j^*})$.
    Until $SP_{i^*j^*} = SE_{i^*j^*} = SS_{i^*j^*} = SW_{i^*j^*}$.
    $S_{i^*j^*} := ERS_{i^*j^*}, E_{i^*j^*} := ERS_{i^*j^*} + d_{i^*j^*}$.
    For $\forall k \in Ke \wedge re_{i^*j^*k} > 0$
        Select $l \in Ue^t_{i^*k}(t = S_{i^*j^*})$ according to MTRCA rule, $Xe_{i^*j^*kl} := 1$.
    End For
    Update $Up^t_{ik}, Ue^t_{ik'}, Us^t_{ik''}, Uw^t_{k'''}, S_g := S_g \bigcup (i^*, j^*)$.
End While
```

Algorithm 1: Algorithm of serial schedule generation scheme for aircraft deck operations.

which are eligible both for precedence and resource at scheduling time t_g of stage g is denoted by

$$D'_g(t_g) = \left\{ (i,j) \left| \begin{array}{l} (i,j) \notin \left\{C_g \bigcup A_g\right\}, rp_{ijk} \le Up^{t_g}_{ik}(\forall k \in Kp), re_{ijk'} \le \left|Ue^{t_g}_{ik'}\right|\left(\forall k' \in Ke\right), \\ Us^{t_g}_{ik''} \cdot rs_{ijk''} = 0\left(\forall k'' \in Ks\right), rw_{ijk'''} \le Uw^{t_g}_{ik'''}\left(\forall k''' \in Kw\right), P_{ij} \subseteq C_g \end{array} \right. \right\}. \quad (12)$$

The parallel schedule generation scheme for aircraft deck operations is presented in Algorithm 2.

Similarly, the overall process of the proposed PSGS for aircraft deck operations is derived from the conventional one for RCPSP in [57]. At the beginning, when $g = 1$, set A_g and C_g are initialized with the dummy start operations of each aircraft and null set, respectively, and all the state parameters of different kinds of resource and start/finish time of operations in A_g are initialized as Algorithm 1 does.

As a time-oriented scheduling scheme, PSGS creates a feasible schedule by considering increasing decision time t. At each scheduling stage g, the decision time t_g is determined by the latest finish time of operation in A_{g-1}, and then the decision times finished at t_g are selected from A_{g-1} and are added to C_g. At this point, the set of operations which are eligible both for precedence and resource $D'_g(t_g)$ is calculated according to (12). Then repeat the scheduling that selects the operation in $D'_g(t_g)$ with the highest priority, arrange the start time as $S_{i^*j^*} := t_g$ and the corresponding finish time, allocate the required equipments according to the MTRCA rule, and update the scheduling process set A_g, $D'_g(t_g)$, and state parameters until $D'_g(t_g)$ is null.

5.1.3. Backward Scheduling.

5.1.3. Backward Scheduling. As mentioned earlier, the forward and backward pass scheduling both can be used in the serial and parallel schedule generation. In the forward pass scheduling, all operations are shifted to the left of the timeline, which means the schedule starts from the initial operation's immediate successors, and the starting time is set as soon as possible. The schedule obtained in this way is also called the left-justified schedule. On the other hand, the operations also can be scheduled oppositely by backward pass scheduling (also called right-justified); that is, all operations are shifted to the right except for the first and last operations, and the schedule starts from the final operation's immediate predecessors. In this way, the completing time is set as late as possible, and the detailed procedure of backward

```
**Input:** Priority of support operations $x_{ij}$, initial precedence relation $D' = (V, \mathrm{AN})$.
**Output:** Temporal schedule $\{[S_{ij}, E_{ij}]\}$, initial equipment allocation plan $\{Xe_{ijkl}\}$.
Initialize $Up^t_{ik}, Ue^t_{ik'}, Us^t_{ik''}, Uw^t_{k'''}, A_1 := \bigcup_{i \in I}(i, 1), C_1 := \varnothing, S_{i1} := Er_i, E_{i1} := Er_i, g := 2$.
**While** $|C_g \bigcup A_g| < |J|$
    $t_g := \min\{S_{ij} + d_{ij} \mid (i, j) \in A_{g-1}\}$.
    $A_g := A_{g-1} - \{(i, j) \mid (i, j) \in A_{g-1} \wedge t_g = E_{ij}\}$.
    $C_g := C_{g-1} \bigcup \{(i, j) \mid (i, j) \in A_{g-1} \wedge t_g = E_{ij}\}$.
    Calculate $D'_g(t_g)$ according to Equation (12).
    **Repeat**
        $(i^*, j^*) := \min_{O_{ij} \in D'_g(t_g)}\{(i, j) \mid x_{ij} = \inf_{O_{pq} \in D'_g(t_g)}(x_{pq})\}$.
        $S_{i^* j^*} := t_g, E_{i^* j^*} := S_{i^* j^*} + d_{i^* j^*}$.
        $A_g := A_g \bigcup (i^*, j^*)$.
        **For** $\forall k \in Ke \wedge re_{i^* j^* k} > 0$.
            Select $l \in \pi Le^{t_g}_{i^* k}$ according to MTRCA rule, $Xe_{i^* j^* kl} := 1$.
        **End For**
        Update $Up^t_{ik}, Ue^t_{ik'}, Us^t_{ik''}, Uw^t_{k'''}, D'_g(t_g)$.
    **Until** $D'_g = \varnothing$
    $g := g + 1$
**End While**
```

Algorithm 2: Algorithm of parallel schedule generation scheme for aircraft deck operations.

pass scheduling of RCMPSP can be found in [58]. It has been verified that the combination of the forward and backward pass scheduling can produce significant improvements in reducing the makespan [59], and the double justification (DJ) which constantly updates a solution through iteration with forward and backward pass scheduling alternately is the most widely applied technique.

To make the backward pass scheduling suitable for aircraft deck operations, the following work is done. Firstly, the start point of the total operations must be set late enough to ensure that operations start from a positive time under a certain set of priority values. Then after the right-justified schedule $(S^{\mathrm{R}}_{ij}, E^{\mathrm{R}}_{ij})(i \in I, j \in J_i)$ is generated with different slack for each aircraft, the makespan can be reduced by subtracting a span ΔT^{L} from the whole schedule, thus eliminating the slack. Note that the above translation must satisfy the constraints in (2) since the released times for each aircraft are different; therefore, the span is set by $\Delta T^{\mathrm{L}} = \min_{i \in I}(S^{\mathrm{R}}_{i1} - Er_i)$, and the final right-justified schedule is obtained as $(S_{ij}, E_{ij}) = (S^{\mathrm{R}}_{ij}, E^{\mathrm{R}}_{ij}) - \Delta T^{\mathrm{L}}, i \in I, j \in J_i$.

5.2. Robust Resource Allocation Scheme

5.2.1. Robust Equipment Allocation Adjustment Scheme. The initial equipment allocation plan, which is generated along with the baseline schedule, can satisfy the equipment constraints, but not necessarily the most robust one. To protect the baseline schedule against the stochastic disturbance of operation durations, a heuristic algorithm for further adjustment of equipment allocation is proposed. Firstly, given a pair of operations (O_{ij}, O_{eg}) with $S_{ij} + d_{ij} \leq S_{eg}$, the time difference between the start of operation O_{eg} and the end of O_{ij} is denoted as the pairwise float:

$$\mathrm{PF}_{ijeg} = \begin{cases} S_{eg} - E_{ij}, \text{ if } (i, j) \notin P^{\mathrm{s}}_{eg}, \\ S_{eg} - E_{ij} + BM, \text{otherwise}, \end{cases} \tag{13}$$

where P^{s}_{eg} represents the set of direct predecessors of operation O_{eg} in the OONF network $D'' = (V, \mathrm{AN} \cup \mathrm{AS})$, and the resource flow AS is determined by the baseline schedule. The pairwise float serves an important function of buffer against stochastic disturbance. The larger the pairwise float is, the less impact of uncertainty will be imposed on the successor operation. However, when $(i, j) \in P_{eg}$, the successor operation will be affected by the predecessor, which denotes that the pairwise float is meaningless, so a large number, i.e., BM, is added to ignore the buffer function in such pairs.

It has been discussed that the protection effect does not increase linearly with the growth of time slack, and it is a good way to represent the protection effect using a negative exponential function [60]. The following surrogate robustness objective function for equipment allocation is suggested to guide the adjustment under the original baseline schedule and equipment constraints.

$$\min RE = \sum_{k \in Ke} \sum_{\{(i,j),(e,g)\} \in Q_k} \exp\left(-\mathrm{PF}_{ijeg}\right), \tag{14}$$

where Q_k denotes the set of operation pairs which possess the same equipment of type $k(k \in Ke)$.

Typical methods of robust resource flow construction, such as interprogramming-based heuristics (e.g., MinEA, MaxPF, and MinED) and simulation-based constructive procedure (e.g., MABO) [28], are always time-consuming for large-scale scheduling problems. In view of the above situation, a greedy heuristic algorithm is designed. In each iteration, the algorithm searches the operations one by one in each equipment of type $k(k \in Ke)$, and the adjustment is performed once the surrogate objective is optimized. The adjustment falls into two categories:

(1) Swap: two operations in different equipments are swapped with each other.

(2) Shift: an operation is shifted to another equipment.

The two kinds of adjustments are based on the premises that the equipment constraints are satisfied and there are no overlaps of time in schedule. Figure 5 shows an example of the adjustment.

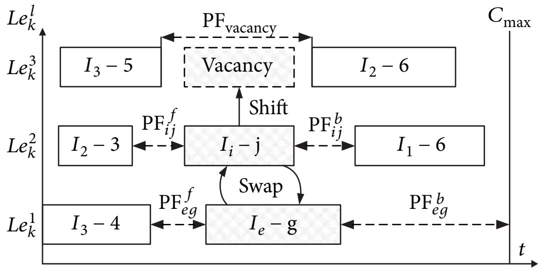


Figure 5: Robust equipment adjustment.

In Figure 5, Le_k^l denotes the lth support equipment of type $k(k \in Ke)$, and $I_i - j$ represents operation j of aircraft i. Suppose that the equipment constraints are satisfied for the operations in Figure 5, then operation O_{ij} can choose to swap with operation O_{eg} or shift to the vacancy within equipment Le_k^3. The principle of selecting the adjustment is to maximize the local robustness. Note that the local robustness of operation O_{ij} is related to not only the time difference between operation O_{ij} and other adjacent operations in the same equipment but also the time difference within equipments which operation O_{ij} can be adjusted to.

We define the local robustness index of operation O_{ij} before adjustment as $\mathrm{LRO}_{ij}^{\mathrm{n}} = \exp\left(-\mathrm{PF}_{ij}^f\right) + \exp\left(-\mathrm{PF}_{ij}^b\right)$, where PF_{ij}^f represents the pairwise float between operation O_{ij} and its predecessor in the same equipment, and PF_{ij}^b represents that between operation O_{ij} and its successors similarly. For example, $\mathrm{PF}_{ij}^f = \mathrm{PF}_{23ij}$ and $\mathrm{PF}_{ij}^b = \mathrm{PF}_{ij16}$ in Figure 5. If there are no predecessors of operation O_{ij}, let $\mathrm{PF}_{ij}^f = \mathrm{BM}$. Similarly, if there are successors of operation O_{ij}, let $\mathrm{PF}_{ij}^b = C_{\max} - E_{ij}$. The time difference of the vacancy is denoted by the pairwise float of operations on either side (as shown in Figure 5). Then, the local robustness index before adjustment of Le_k^l which is reachable for operation O_{ij} is represented as

$$\mathrm{LRE}_{kl}^{\mathrm{n}} = \begin{cases} \exp\left(-\mathrm{PF}_{eg}^f\right) + \exp\left(-\mathrm{PF}_{eg}^b\right), & \text{if } O_{ij} \text{ is swappable with } O_{eg}, Xe_{ijkl} \neq 1, Xe_{egkl} = 1, \\ \exp\left(-\mathrm{PF}_{\mathrm{vacancy}}\right), & \text{if a vacancy of } Le_k^l \text{ can be inserted by } O_{ij}, Xe_{ijkl} \neq 1. \end{cases} \quad (15)$$

After the adjustment, the operation O_{ij} is shifted in the vacancy or swapped with the corresponding operation. Now the local robustness index of operation O_{ij} is represented as $\mathrm{LRO}_{ij}^{\mathrm{c}}$, which can be calculated in the same way as LRO_{ij}^{n}. Likewise, the local robustness index after adjustment of $Le_k^{l'}$ that operation O_{ij} originally adheres to can be represented as

$$\mathrm{LRE}_{kl'}^{\mathrm{c}} = \begin{cases} \exp\left(-\mathrm{PF}_{eg}^f\right) + \exp\left(-\mathrm{PF}_{eg}^b\right), & \text{if } O_{ij} \text{ is swapped with } O_{eg}, Xe_{ijkl'} \neq 1, Xe_{egkl'} = 1, \\ \exp\left(-\mathrm{PF}_{\mathrm{vacancy}}\right), & \text{if } O_{ij} \text{ is shifted and create a vacancy in } Le_k^{l'}, Xe_{ijkl'} \neq 1. \end{cases} \quad (16)$$

Finally, the difference of the local robustness index of Le_k^l between after and before adjustment is computed as

$$\Delta\mathrm{RE}_{kl} = \left(\mathrm{LRO}_{ij}^{\mathrm{c}} + \mathrm{LRE}_{kl'}^{\mathrm{c}}\right) - \left(\mathrm{LRO}_{ij}^{\mathrm{n}} + \mathrm{LRE}_{kl}^{\mathrm{n}}\right). \quad (17)$$

At each decision-making point, robustness will be improved only when there are some equipments so that $\Delta\mathrm{RE}_{kl} < 0$, and the equipment corresponding to the minimum $\Delta\mathrm{RE}_{kl}$ is selected. Let N_{justment}^k be the number of adjustment in an iteration for equipment of the type $k(k \in Ke)$. The

```
Input: Initial equipment allocation plan $Xe_{ijkl}$, temporal schedule $\{[S_{ij}, E_{ij}]\}$ and
precedence relation $D'' = (V, \mathrm{AN} \bigcup AS)$.
Output: Adjusted equipment allocation plan $Xe_{ijkl}'$ and updated precedence relation $D''' = (V, \mathrm{AN} \bigcup AS \bigcup AE)$.
Initialize $N^{k}_{\mathrm{justment}} = 1000$.
For $\forall k \in Ke$
  While $N^{k}_{\mathrm{justment}} > 0$
  $N^{k}_{\mathrm{justment}} = 0$.
    For $\forall (i, j) \in J$
      Calculate $LRO^{\mathrm{n}}_{ij}$ and find the equipment $l$ when $Xe_{ijkl} = 1$.
      For $\forall l' \in (Le_k - l) \wedge i \in \mathrm{Re}_{kl'}$
        if vacancy for shift or operations for swap exist, then calculate $LRE^{\mathrm{n}}_{kl}$, $LRO^{\mathrm{c}}_{ij}$, $LRE^{\mathrm{c}}_{kl'}$,and obtain $\Delta RE_{kl}$.
      End For
      Select the optimal equipment $l^* = \arg\min(\Delta RE_{kl}) \wedge \Delta RE_{kl} < 0$.
      If $l^* \neq \varnothing$
        Perform the shift or swap and update $Xe_{ijkl}$.
        $N^{k}_{\mathrm{justment}} = N^{k}_{\mathrm{justment}} + 1$.
      End If
    End For
  End While
End For
Update the precedence relation $D''' = D'' \bigcup AE$ according to adjusted $Xe_{ijkl}'$.
```

Algorithm 3: Robust equipment allocation adjustment scheme.

adjustment will continue until $N^{k}_{\mathrm{justment}}(\forall k) = 0$, and then the most robust equipment allocation plan is generated. To be specific, the robust equipment allocation adjustment scheme is presented in Algorithm 3.

5.2.2. Robust Personnel Allocation Scheme. Since the baseline schedule is feasible for each type of resources and personnel are replaceable with the same trade in group, the proposed robust personnel allocation scheme is operation-based, i.e., the personnel allocation plan is executed according to the starting time sequence of operations.

Firstly, all the operations are placed in an increasing order according to their planned starting times. Note that if there are some operations with the same starting time, a tie-break will be conducted to distinguish the criticality of them using the notion of free slack RCF_{ij} [61] based on the critical chain scheduling method. As the free slack and the criticality are negatively correlated, and $\mathrm{RCF}_{ij} = 0$ indicates operation O_{ij} is in the critical chain.

Similar to the method in Section 5.2.1, a robustness evaluation function is adopted to guide the personnel allocation for each operation in the sequence list. The float time between operation O_{ij} and operations which adhere to the l th personnel with trade $k(k \in Kp)$ is denoted by

$$\mathrm{IF}_{ijkl} = \begin{cases} S_{ij} - \mathrm{LFT}_{kl}, \text{ if } (e, g) \notin P^{\mathrm{se}}_{ij}, \\ S_{ij} - \mathrm{LFT}_{kl} + BM, \text{ else if} (e, g) \in P^{\mathrm{se}}_{ij}, \\ S_{ij} + BM, \text{ else if} (e, g) = \varnothing, \end{cases} \tag{18}$$

where operation O_{eg} is the latest scheduled operation of the lth personnel with trade $k(k \in Kp)$ (denoted as Lp^{l}_{k}), and the completion time of O_{eg} corresponds to the latest completion time of Lp^{l}_{k}, which is denoted as LFT_{kl}. P^{se}_{ij} represents the set of direct predecessors of operation O_{ij} in the OONF network D'''. Similar to (14), a robustness index of personnel is presented as $\mathrm{RP}_{ijkl} = \exp(-\mathrm{IF}_{ijkl})$, and operation O_{ij} is allocated to the personnel with the minimum RP_{ijkl}.

To be specific, the robust equipment allocation scheme is presented in Algorithm 4, where $Up_k(t, i) = \{l \mid \mathrm{LFT}_{kl} \leq t, i \in Rp_{kl}\}$ is the set of available personnel that can support aircraft i at decision point t.

5.3. Scheduling Policy. After the baseline schedule and the resource allocation of the aircraft flight deck, operations are determined, then it comes to the execution stage to turn the baseline schedule into various executed schedules with unpredictable durations. During the multistage decision process, the scheduling policy serves a crucial function of determining dynamically which operations are to be started next in each possible scenario. In the resource-constrained project scheduling problem with stochastic activity durations, robust project scheduling and stochastic project scheduling are the major two approaches.

In robust project scheduling, a fixed baseline schedule is generated in advance of project execution, and the roadrunner scheduling policy and the railway scheduling policy are commonly used in execution. Roadrunner scheduling,

```
Input: Temporal schedule $\{[S_{ij}, E_{ij}]\}$ and precedence relation $D''' = (V, \text{AN} \bigcup AS \bigcup AE)$.
Output: Personnel allocation plan $Xp_{ijkl}$ and updated precedence relation $D = (V, \text{AN} \bigcup AP \bigcup AE \bigcup AS)$.
Sort the operations by increasing $S_{ij}$ (tie-break: increasing $RCF_{ij}$) and obtain the
sequence list of operations for allocation.
For $n = 1 : |J|$
    Select the $n$th operation $O_{ij}$ in the sequence list.
    For $\forall k \in Kp \wedge rp_{ijk} > 0$
        Calculate the idle personnel $\pi Lp_k(S_{ij}, i) = \{l \mid LFT_{kl} \le S_{ij}, i \in Rp_{kl}\}$
        For $\forall l \in Up_k(S_{ij})$
            Calculate $\text{IF}_{ijkl}$ and obtain $RP_{ijkl}$.
        End For
        Sort the personnel in $Up_k(S_{ij})$ by increasing $RP_{ijkl}$, select the
          first $m$th $(m = rp_{ijk})$ personnel to support $O_{ij}$, and update $Xp_{ijkl}$.
    End For
End For
Update the precedence relation $D = D'' \bigcup AP$ according to $Xp_{ijkl}$.
```

Algorithm 4: Robust personnel allocation scheme.

typically applied in a critical chain/buffer management (CC/BM) schedule, assumes that activities are started as soon as possible when all their predecessors have finished and enough resource units are available. The goal of this policy is to decrease the expected length of the project. Derived from the train schedule, the railway scheduling policy indicates that every activity should not be started earlier than its planned starting time in the baseline. This policy is introduced in order to increase the stability of the project. It has been investigated that railway scheduling performs better than roadrunner scheduling in terms of improving both the stability and the expected project length considering all optimal baseline schedules with the same project length [62].

As for stochastic project scheduling without a baseline schedule and resource allocation plan, the scheduling policy operates in a specific way with a vector of weights for activities (e.g., activity list.); that is, in each decision-making stage, the unscheduled activity with the maximum weight whose predecessors have already been completed is selected. The selected activity will be arranged at the point satisfying all the resource constraints. Therefore, stochastic project scheduling is also classified as online reactive scheduling.

As is illustrated in Section 2.2, only C^{AB} and C^{RB} become good choices for our problem. C^{AB} is an activity-based policy, which means that activities are scheduled as early as possible according to the order of an activity list π, with the side constraint that $s_i(d) \le s_j(d)$, if $i \prec_\pi j$ for each scenario d. $i \prec_\pi j$ denotes that activity i precedes activity j in activity list π. C^{RB} is a resource-based policy, which means it is a direct extension of deterministic parallel scheduling. In this policy, the first precedence and resource feasible activity on activity list π is selected at each decision point.

In the RSPFDO, as is shown in (1), PMLM and IRM are the main criteria to be focused on, rather than the deviation between the baseline and the executed schedule, so the railway scheduling policy is conservative in this regard. On the other hand, roadrunner scheduling ensures that operations are started as soon as possible under the constraints of resource flow, while parallel scheduling of the supply resource may change the schedules generated by SSGS even in a deterministic environment. Besides, the policies used in stochastic project scheduling suffer from the major drawback that the fixed resource allocation and the baseline information are not taken into account. To cope with the complicated resource constraints and stochastic operation durations in the RSPFDO, a preconstraint policy (C^{PC}) is proposed and elaborated as follows.

In general, the policy is executed in the way of parallel scheduling, or it can be regarded as a kind of resource-based scheduling. The improvement lies in that all the resource constraints are converted into precedence constraints before execution.

(1) The resource flows caused by personnel, equipment, and workstation space allocation are converted into the finish-start precedence relations in the OONF $D = (V, \text{AN} \cup \text{AP} \cup \text{AE} \cup \text{AS})$.

(2) As for the supply resource of each type k, the set of operations that require supply resource of the same type are found, and then a start-start precedence relation is added to each pair of operations in the set. In the start-start precedence relation, if the starting times of operation O_{ij} and operation O_{eg} satisfies the inequality $S_{ij} \le S_{eg}$, the inequality $\tilde{S}_{ij} \le \tilde{S}_{eg}$ regarding the practical starting times of the two operations makes sense. This strategy is similar to the side constraint applied in C^{AB}.

The execution of the preconstraints is a supply resource-based scheduling process in fact as the resource constraints of other types are completely converted into the precedence constraint. At each decision point t, the policy selects the eligible operation with the minimum starting time in baseline. Operations are eligible when their predecessors have already been completed, and there is enough supply capacity left to be processed at t.

6. Proactive Robust Scheduling Optimization Algorithm

The baseline schedule and resource allocation plan can be generated by the proposed proactive robust scheme formulation and scheduling policy under the certain priority value of each operation, and the practical executed schedule also can be worked out by the scheduling policy. In other words, a given duration scenario and a priority vector of operation have been formulated into mathematical forms. Besides, both the resource allocation schemes and C^{PC} policy are designed for the robustness objective. Next, the robustness optimization of the baseline schedule for RSPFDO will be dealt with. The RSPFDO is NP-hard because it can be classified as RCMPSP, which is a general form of the resource-constrained project scheduling problem (RCPSP). As is mentioned in Section 2.3, inspired by successful applications of the TLBO in RCPSPs, a hybrid TLBO (HTLBO) algorithm for solving the RSPFDO is proposed in this paper, which combines with some other efficient strategies utilized in RCPSP.

6.1. Representation of Schedules. Among different kinds of schedule representations, activity-list (AL) representation and random-key (RK) representation are most widely used. Generally, procedures that make use of the AL representation outperform the others because under the RK representation, there are many different representations for one single schedule, which make it difficult for (meta)heuristics to search for the best solution in a large space. The AL representation also suffers from the drawback that a single schedule can be represented by different activity lists, which is more obvious when a multiproject background with different released times is involved. However, this problem occurs more frequently with the form of RK in the single project [54].

To cope with this problem, Debels and Vanhoucke [56] embed the RK representation with the TO condition, which indicate that, for any two activities i and j ($S_i < S_j$) in a given left-justified schedule, activity i should have a higher priority than activity j, thus ensuring that each schedule corresponds to only one representation. Besides, inspired by the work in [56], to make use of the forward/backward scheduling technique, a population-orientated forward/backward scheduling method is proposed based on the RK representation which is used in the two cases below:

Case 1. $\mathbf{x} = \mathbf{x}^{\mathrm{L}} = [S_{11}, S_{12}, \ldots, S_{1|J_1|}, S_{21}, \ldots, S_{ij}, \ldots, S_{n|J_n|}]$ if used for generating a left-justified schedule.

Case 2. $\mathbf{x} = \mathbf{x}^{\mathrm{R}} = [E_{11}, E_{12}, \ldots, E_{1|J_1|}, E_{21}, \ldots, E_{ij}, \ldots, E_{n|J_n|}]$ if used for generating a right-justified schedule.where $\mathbf{x}$ denotes the representation of individual in the population as well as the vector of priority. When a schedule is generated, its representation is adjusted and saved by vectors of starting times or completion times according to how it will be used. In this form, the TO condition is satisfied. If a vector of priority is used for generating a left-justified schedule, then $\mathbf{x}^{\mathrm{L}}$ is selected, and when $x_i < x_j$, the ith operation will have a higher priority than the jth operation, while if a vector of priority is used for generating a right-justified schedule, then $\mathbf{x}^{\mathrm{R}}$ is selected, and when $x_i < x_j$, the jth operation will have a higher priority than the ith operation.

6.2. Robustness Evaluation. The robustness evaluation for aircraft flight deck operations includes three steps:

Step 1. Baseline schedule generation. A corresponding baseline schedule is generated by SSGS or PSGS under a given vector of priority $\mathbf{x}$, in the way of either forward scheduling or backward scheduling, as shown in Section 5.1.

Step 2. Robust resource allocation. According to the given baseline schedule, personnel and equipments are allocated to each operation based on the robust personnel allocation plan and equipment adjustment scheme, as illustrated in Section 5.2.

Step 3. Simulation with certain replications. Firstly, Ns scenarios $\mathbf{D}_k = [d_{11}^k, d_{12}^k, \ldots, d_{1|J_1|}^k, d_{21}^k, \ldots, d_{ij}^k, \ldots, d_{n|J_n|}^k](1 \le k \le Ns)$ are generated by random sampling, where d_{ij}^k indicates the sampling of the stochastic duration of operation O_{ij} according to its distribution function under the kth particular scenario. Then, the robust criteria are calculated by the average of Ns times of simulation with C^{PC} scheduling policy

$$\begin{aligned}
\Pr\left(\tilde{C}_{\max}(\mathbf{x}) \le C_{\max}{}^{*}\right) &= \frac{1}{Ns}\sum_{k=1}^{Ns}\frac{1}{2}\left(\operatorname{sgn}\left(C_{\max}{}^{*} + \varepsilon - C_{\max}(\mathbf{D}_k, \mathbf{x})\right) + 1\right), \\
EC_{\max}(\boldsymbol{x}) &= \frac{1}{Ns}\sum_{k=1}^{Ns} C_{\max}(\mathbf{D}_k, \mathbf{x}), \\
VC_{\max}(\boldsymbol{x}) &= \frac{1}{Ns}\sum_{k=1}^{Ns} \left(C_{\max}(\mathbf{D}_k, \mathbf{x}) - EC_{\max}(\mathbf{x})\right)^2,
\end{aligned} \tag{19}$$

where ε is a minimal value to make sure that when $C_{\max}(\mathbf{D}_k, \mathbf{x}) = C_{\max}{}^{*}$, the return of sgn $(\cdot)$ is still 1, which is in accordance with the definition of PCLM. Ashtiani et al. [63] showed that when computational effort matters, a rather low number of replications Ns ($Ns = 10$ for common cases) for evaluation is preferred during the search stage, which leads to lower accuracy but a higher number of scanned solutions and a better final outcome. In our problem, the accuracy of the final algorithm is obviously vital, and 3000 replications of simulation are conducted for the final evaluation.

To compare the fitness values of two individuals with the objectives in lexicographical order as shown in (1), the fitness values are integrated into a weighted sum

$$\min Rf = 1 - \Pr\left(\tilde{C}_{\max} \le C_{\max}{}^{*}\right) + \delta\left(EC_{\max} + \omega VC_{\max}\right), \tag{20}$$

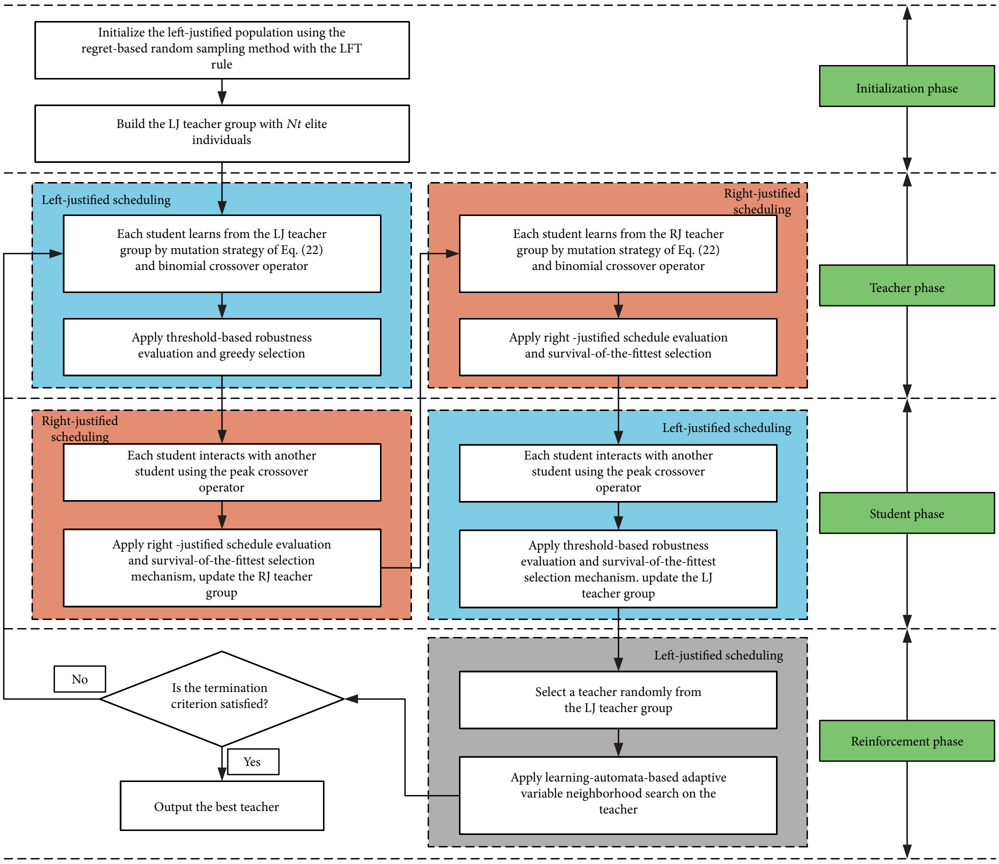

Initialize the left-justified population using the regret-based random sampling method with the LFT rule
Build the LJ teacher group with Nt elite individuals
Initialization phase
Left-justified scheduling
Each student learns from the LJ teacher group by mutation strategy of Eq. (22) and binomial crossover operator
Apply threshold-based robustness evaluation and greedy selection
Right-justified scheduling
Each student learns from the RJ teacher group by mutation strategy of Eq. (22) and binomial crossover operator
Apply right -justified schedule evaluation and survival-of-the-fittest selection
Teacher phase
Right-justified scheduling
Each student interacts with another student using the peak crossover operator
Apply right -justified schedule evaluation and survival-of-the-fittest selection mechanism, update the RJ teacher group
Left-justified scheduling
Each student interacts with another student using the peak crossover operator
Apply threshold-based robustness evaluation and survival-of-the-fittest selection mechanism. update the LJ teacher group
Student phase
No
Is the termination criterion satisfied?
Yes
Output the best teacher
Left-justified scheduling
Select a teacher randomly from the LJ teacher group
Apply learning-automata-based adaptive variable neighborhood search on the teacher
Reinforcement phase

Figure 6: The flowchart of HTLBO for the RSPFDO.

where δ is a minimum value to prioritize the two objectives in (1); here, we set $\delta = 10^{-7}$ to ensure that the value of PCLM dominates the fitness function.

In order to make a fair comparison among different algorithms, the termination condition of the algorithm is measured by the maximum number of generated schedules $N_{\max}^{gs}$. Note that making a baseline schedule or solving one scenario with the C^{PC} policy will be counted as one generated schedule.

To make the most of the limitative maximum number of generated schedules in this problem, two techniques are presented to concentrate on the largest computational effort on those promising solutions.

The first one is named as threshold-based evaluation. This technique is developed based on the following consideration. If the makespan of a baseline schedule is not within the limitative makespan $(C_{\max}(\mathbf{D}_0, \mathbf{x}) > C_{\max}^*)$, there is no need to continue robust resource allocation and evaluation with the multiple replications, where $\mathbf{D}_0$ represents the vector of expected durations used for baseline schedule generation.

The second one is designed for right-justified schedule evaluation. The purpose of the double justification is to reduce the makespan of flight deck operations. However, the final baseline schedule is a left-justified one in which the operation is started as soon as possible, so the evaluation with multiple replications can be neglected. More precisely, for any right-justified schedule or the opposite one whose makespan is beyond the limitative makespan with a given $\mathbf{x}$, set $\Pr(\tilde{C}_{\max}(\mathbf{x}) \le C_{\max}{}^*) = -1$, $EC_{\max}(\mathbf{x}) = C_{\max}(\mathbf{D}_0, \mathbf{x})$, $VC_{\max}(\mathbf{x}) = 0$, thus ensuring that the fitness value is always inferior to those of other promising solutions.

6.3. *Procedure of the Proposed HTLBO*

6.3. Procedure of the Proposed HTLBO. The flowchart of HTLBO for solving the RSPFDO is shown in Figure 6. The framework of the algorithm is based on the original TLBO which contains the teacher phase and the student phase. Compared to the conventional TLBO, there are three main improvements. Firstly, differential evolution operators and peak crossover operators are used for updating the population in the teacher phase and the student phase, respectively. Secondly, a reinforcement phase with learning-automata-based adaptive variable neighborhood search is incorporated into the algorithm. Thirdly, different from the double justification operated on individual solution, a population-orientated forward/backward scheduling is adopted, which means that after updating the population in a certain phase, the whole population will be scheduled in the way contrary to that in the last phase.

The overall procedure of HTLBO is illustrated as follows:

Step 1. Initialization phase. Initialize Np left-justified (LJ) students with the regret-based random sampling method with the latest finish time (LFT) rule, as discussed in Section 6.4. Then after threshold-based robustness evaluation of each student, a LJ teacher group with Nt LJ elite individuals of different representations is built. Set the generation $G = 1$.

Step 2. Teacher phase with left-justified scheduling. The framework of the teacher phase is based on the differential evolution (DE) algorithm, as discussed in Section 6.5. A teacher is chosen randomly from the LJ teacher group for each student, and then the teaching is conducted by mutation strategy of (22) with the binomial crossover operator. Afterwards, an LJ baseline schedule and resource allocation plan of each new individual is generated. At last, threshold-based robustness evaluation and greedy selection are applied to update the population.

Step 3. Student phase with right-justified (RJ) scheduling. For each student, another classmate is randomly selected from the population to form a pair. Then, two successors are created using the crossover operator, of which the baselines are generated by right-justified scheduling. At last, RJ schedule evaluation and survival-of-the-fittest selection mechanism are applied to update the population, and the RJ teacher group is updated with Nt RJ elite individuals, as discussed in Section 6.6.

Step 4. Teacher phase with right-justified scheduling. This step is executed in a similar way as that of step 2, except that the phase is based on right-justified scheduling, indicating that RJ baseline schedules are generated and RJ schedule evaluation are applied.

Step 5. Student phase with left-justified scheduling. This step is executed in a similar way as that of step 3, except that the phase is based on left-justified scheduling. In this step, LJ baseline schedules along with robust resource allocations are generated, and threshold-based robustness evaluation is applied. Besides, an LJ teacher group is updated at last.

Step 6. Reinforcement phase with left-justified scheduling. In this phase, a teacher is selected randomly from the LJ teacher group. Then, a learning-automata-based adaptive variable neighborhood search is applied to update the teacher's performance, as discussed in Section 6.7.

Step 7. Judge whether the termination criterion is satisfied. If the answer is yes, output the best teacher which includes the optimal robust baseline, personnel, and support equipment allocation plan; otherwise, go to step 2 to continue search with the generation $G = G + 1$.

6.4. Population Initialization.

To obtain good initial population at the beginning of search, individuals are generated by adopting the regret-based random sampling method with the latest finish time (LFT) rule [52]. With this method, operations with a small CPM-based latest finish time LFT have a higher chance of being selected. More precisely, the RK representation (starting time/completion time vector) is obtained from the initial schedule, which is generated by repeatedly selecting an unscheduled operation from the eligible operation set (denoted as E). An eligible operation is an unselected one for which all of the predecessors have already been selected. A roulette wheel selection strategy is applied to select operation in E, and the probability η_{ij} of being chosen for eligible operation O_{ij} based on the LFT rule is represented as

$$\eta_{ij} = \frac{\mu_{ij} + 1}{\sum_{(e,g)\in E}\left(\mu_{eg} + 1\right)}, \quad (21)$$

where $\mu_{eg} = \max_{(i,j)\in E}\mathrm{LFT}_{ij} - \mathrm{LFT}_{eg} + 1$ is the priority value of operation O_{eg}, and LFT_{ij} is the latest finish time of operation O_{ij}. Suppose that the population size is Np, each individual is generated with this procedure.

6.5. Teacher Phase.

In this phase, the primary goal is to utilize the knowledge of the elite individual (teacher) to improve the performance of each individual (student) and the level of the total population (class). Here, the operators of the DE algorithm are applied because it is simple and robust with fast convergence, especially in the area of scheduling [64].

In the basic DE, only the mutation strategies with the guide of the best individual, such as DE/best/1, DE/current-to-best/1, and DE/best/2, can imitate the teaching process. To overcome the limitation and increase the diversity of teaching ways, the mutation strategy for teaching is proposed, inspired by the mutation strategy of DE/current-to-gr_best/1 [65].

$$\mathbf{v}_{i,G} = \mathbf{x}_{i,G} + F_i \cdot \left(\mathbf{x}^{\mathrm{r}}_{t\mathrm{best},G} - \mathbf{x}_{i,G}\right) + F_i \cdot \left(\mathbf{x}_{r^i_1,G} - \mathbf{x}_{r^i_2,G}\right), \quad (22)$$

where $\mathbf{x}_{i,G}$ and $\mathbf{v}_{i,G}$ are the target vector and the donor vector at the Gth generation, respectively, and $\mathbf{x}^{\mathrm{r}}_{t\mathrm{best},G}$ is randomly selected from the teacher group which contains Nt different elite individuals. Note that $\mathbf{x}^{\mathrm{r}}_{t\mathrm{best},G}$, $\mathbf{x}_{r^i_1,G}$, and $\mathbf{x}_{r^i_2,G}$ are all randomly selected from the current population, and $i \neq r^i_1 \neq r^i_2$. In addition, none of them equals to $\mathbf{x}^{\mathrm{r}}_{t\mathrm{best},G}$. F_i is the scale factor of the ith individual target vector.

After the mutation operation, the binomial crossover operator and greedy selection of the basic DE are applied to update individuals with the guide of teachers. As for the parameter setting of scale factor F_i and crossover probability Cr_i, a parameter adaptation scheme is employed inspired by the work in [44, 65].

For each generation G, the parameters of F_i and Cr_i for each individual target vector is generated independently as

$$\begin{cases} F_i = \mathrm{Cauchy}(M_{\mathrm{F}}, 0.1), \\ \mathrm{Cr}_i = \mathrm{Gaussian}(M_{\mathrm{Cr}}, 0.1), \end{cases} \quad (23)$$

where $\mathrm{Cauchy}(M_{\mathrm{F}}, 0.1)$ and $\mathrm{Gaussian}(M_{\mathrm{Cr}}, 0.1)$ are random values sampled from a Cauchy distribution with mean values M_{F} and M_{Cr} respectively, and variance of 0.1. At the beginning, both M_{F} and M_{Cr} are initialized as 0.5. The set of the successful scale factors and crossover probabilities are denoted as by S_{F} and S_{Cr}, respectively, and at the end of each generation, M_{F} and M_{Cr} are updated as

$$\begin{cases} M_{\mathrm{F}} = \omega_{\mathrm{F}} M_{\mathrm{F}} + (1-\omega_{\mathrm{F}})\mathrm{mean}_{WA}(S_{\mathrm{F}}), \\ M_{\mathrm{Cr}} = \omega_{\mathrm{Cr}} M_{\mathrm{Cr}} + (1-\omega_{\mathrm{Cr}})\mathrm{mean}_{WA}(S_{\mathrm{Cr}}), \end{cases} \tag{24}$$

where ω_{F} and ω_{Cr} are the weight factors conforming to uniform distribution (0.2, 0.8), and the mean of all successful scale factors and crossover probabilities is computed as

$$\begin{cases} \mathrm{mean}_{WA}(S_{\mathrm{F}}) = \sum_{m=1}^{|S_{\mathrm{F}}|} w_m S_{\mathrm{F},m}, \\ \mathrm{mean}_{WA}(S_{\mathrm{Cr}}) = \sum_{m=1}^{|S_{\mathrm{Cr}}|} w_m S_{\mathrm{Cr},m}, \end{cases} \tag{25}$$

where $w_m = \Delta Rf_m / \sum_{m=1}^{|S_{\mathrm{F}}|} \Delta Rf_m$, $|S_{\mathrm{F}}| = |S_{\mathrm{Cr}}|$, and the improvement of fitness $\Delta Rf_m = |Rf_{m,\mathrm{old}} - Rf_{m,\mathrm{new}}|$.

Note that the teacher phase is divided into two parts: right-justified teaching that the population is backward-scheduled and left-justified teaching that the population is forward-scheduled, and the parameter setting of each part is different. The same is true to the other phases.

6.6. Student Phase. Apart from learning from teachers, students always improve themselves by learning from each other. In this stage, the main purpose is to help the population making progress together with the interaction between individuals, and meanwhile avoiding falling into local extremum caused by the teacher phase so fast. The peak crossover operator that makes use of the resource utilization ratio (RUR) [66] is modified according to the resource conditions in flight deck operations and applied in the student phase.

Firstly, under a given schedule S, the integrated resource utilization ratio (IRUR) which measures the multiresource utilization at time unit t is defined as

$$\begin{aligned} \mathrm{IRUR}(t,S) = {} & \sum_{k\in Kp} \frac{\lambda_k^{\mathrm{p}}}{|Kp|} \sum_{O_{ij}\in A_t} \frac{rp_{ijk}}{|Lp_k|} + \sum_{k\in Ke} \frac{\lambda_k^{\mathrm{e}}}{|Ke|} \sum_{O_{ij}\in A_t} \frac{re_{ijk}}{|Le_k|} \\ & + \sum_{k\in Ks} \frac{\lambda_k^{\mathrm{s}}}{|Ks|} \sum_{O_{ij}\in A_t} \frac{rs_{ijk}}{Ni} + \sum_{k\in Kw} \frac{\lambda_k^{\mathrm{w}}}{|Kw|} \sum_{O_{ij}\in A_t} \frac{rw_{ijk}}{Lw_k^t}, \end{aligned} \tag{26}$$

where λ_k^{p}, λ_k^{e}, λ_k^{s}, and λ_k^{w} represent the weight of type k of personnel, support equipment, workstation space, and supply resource respectively, and they can be calculated by (take λ_k^{p} as an example)

$$\lambda_k^{\mathrm{p}} = \frac{\sum_{i\in I}\sum_{j\in J_i} d_{ij}\cdot \mathrm{sgn}\left(rp_{ijk}\right)}{\sum_{i\in I}\sum_{j\in J_i} d_{ij}}. \tag{27}$$

Further, the total integrated resource utilization can be denoted under a given start time t and time length l within the schedule S

$$\mathrm{TIRU}(t,l,S) = \sum_{\tau=t}^{t+l-1} \mathrm{IRUR}(\tau,S). \tag{28}$$

With the TIRU, the time intervals when the integrated resource utilization is high (namely, peaks) can be found, and they are exactly the bottlenecks of multi-aircraft flight deck operations. In the algorithm, only one peak, denoted by $[tp_1(S), tp_2(S)]$, is considered, and the start time $tp_1(S)$ of the peak with length l is calculated as

$$tp_1(S) = \underset{0\le t\le C_{\max}-l}{\arg\max}\,\mathrm{TIRU}(t,l,S). \tag{29}$$

As the peak contains good gens of individual, a two-point crossover operator is conducted to each pair of individuals to take advantage of the peak for interaction between individuals. For each individual $\mathbf{x}_i$, another individual $\mathbf{x}_j (j \neq i)$ is randomly selected from the population to form a pair. The pair takes turns to be the mentor $\mathbf{x}_{\mathrm{m}}$ and the learner $\mathbf{x}_{\mathrm{l}}$ to learn from each other. Let S_{m} and S_{l} be the schedule of mentor and learner, respectively, which are saved after evaluation in the last phase along with IRUR. At each turn, one successor $\mathbf{x}_{\mathrm{s}}$ is generated with the crossover of $\mathbf{x}_{\mathrm{m}}$ and $\mathbf{x}_{\mathrm{l}}$ as follows. Firstly, the length of the peak l is randomly sampled from the interval $[(1/4)C_{\max}(\mathbf{x}_{\mathrm{m}}), (3/4)C_{\max}(\mathbf{x}_{\mathrm{m}})]$, and the peak and start time tp_1 of the mentor $\mathbf{x}_{\mathrm{m}}$ are calculated by (26), (27), (28), and (29). Then, a two-point crossover [56] is applied to generate the successor $\mathbf{x}_{\mathrm{s}}$.

Case 1. If $x_{\mathrm{l},i} < tp_1(S_{\mathrm{m}})$, then $x_{\mathrm{s},i} := x_{\mathrm{l},i} - BM$.

Case 2. If $tp_1(S_{\mathrm{m}}) \le x_{\mathrm{l},i} \le tp_2(S_{\mathrm{m}})$, then $x_{\mathrm{s},i} := x_{\mathrm{m},i}$.

Case 3. If $x_{\mathrm{l},i} > tp_2(S_{\mathrm{m}})$, then $x_{\mathrm{s},i} := x_{\mathrm{l},i} + BM$.

Finally, after evaluations for the two successors, we select the one with better fitness and replace the old individual. However, to avoid loosening high-quality teacher, the individual with the best fitness will not be replaced unless the successors outperform it. This selection mechanism can be called as survival of the fittest which is commonly used in genetic algorithms.

6.7. Reinforcement Phase. As the teacher group serves as the guider of the class, its performance is crucial for both the evolution process and the output of the algorithm. To further improve the teaching level, a reinforcement phase is incorporated into the algorithm, which is realized by adaptive variable neighborhood search (AVNS).

In a nutshell, the AVNS starts from an individual selected from the teacher group, and then neighboring individuals are generated by means of several neighborhood operators and being tested for acceptance [67]. Different from the basic variable neighborhood search, the neighborhood operator is randomly selected in each iteration according to an updating probability. Since the neighborhood operator serves as a crucial function in search [68], five neighborhood operators covering different neighborhood scales are utilized according to the problem structure.

(1) Swap operations: swap two operations between which no precedence relation exists by exchanging their priority values.

(2) Shift operation: shift an operation to a new location in the schedule. Select an operation O_{ij} randomly, and the feasible range of shift is defined as $[\mathrm{es}_{ij}, \mathrm{ls}_{ij}]$, where es_{ij} and ls_{ij} are the earliest and latest start time of operation O_{ij} in the schedule. Then, the set of operations whose start times are within the feasible range is constructed, and the corresponding start time list in ascending order is calculated. Finally, an interval in the start time list is selected randomly, and the priority value of operation O_{ij} is set as a random number within the interval.

(3) Reshuffle operations within the aircraft. Firstly, an aircraft $i \in I$ is selected randomly, and Na operations corresponding to aircraft i are also selected randomly. Then each of the selected operation is shifted into a new location by means of a shifting operation operator. Here, the number of reshuffle operation is a random integer $Na \in Z[4, 9]$.

(4) Reshuffle operations with resource driven: the selection of operations is biased to those underutilizing the resources in time intervals. Contrary to the peak crossover which focuses on the time interval with high resource utilization, this operator aims at polishing the solution by addressing the inefficiencies in resource utilization. To be specific, $Na(Na \in Z[4, 9])$ operations are selected to shift into a new location by means of a shifting operation operator, and the selection probability for operation O_{ij} is defined by

$$p_{ij}^{\mathrm{r}} = \frac{1/\mathrm{IRUR}\left(S_{ij}\right)}{\sum_{e\in I}\sum_{g\in J_e} 1/\mathrm{IRUR}\left(S_{eg}\right)}. \quad (30)$$

(5) Swap neighbor aircraft: swap operations between two adjacent aircrafts in order. Here, the order of aircraft $I(S)$ is based on the concept of the "center of mass" [45], and center (i) for each aircraft $i \in I$ is computed as

$$\mathrm{centre}(i) = \frac{1}{|J_i|}\sum_{j\in J_i} S_{ij}. \quad (31)$$

The order of aircraft is sorted according to the centers of mass. Select $n(1 \le n < |I|)$ randomly and set $i_1 = I(S)_n$, $i_2 = I(S)_{n+1}$, with $\Delta\mathrm{centre} = |\mathrm{centre}(i_2) - \mathrm{centre}(i_1)|$, then the swap operation is conducted by $S_{i_1 j} = S_{i_1 j} + \Delta\mathrm{centre}, \forall j \in J_{i_1}$ and $S_{i_2 j'} = S_{i_2 j'} - \Delta\mathrm{centre}, \forall j' \in J_{i_2}$.

From the perspective of the neighborhood scale, swap operations and shift operation belong to the small neighborhood, and reshuffle operations within aircraft and reshuffle operations with resource driven belong to the medium neighborhood. Swap neighbor aircraft belong to the large neighborhood.

To select a suitable neighborhood in each iteration, an adaptive variable neighborhood strategy based on learning automata [69] is designed. Learning automata (LA) are simple devices of reinforcement learning that take actions in a feedback loop with single state environments. The objective of LA is to find an optimal state-oriented action and learn from the environmental feedback (reward or penalty) generated by past actions. At each time instant t, a single automaton selects an action according to its action probability vector, which is updated with a specific learning scheme, such as linear reward-penalty, linear reward-inaction, and linear reward-ε-penalty. The general update scheme for the adaptive variable neighborhood is given by

$$\begin{aligned} pl_i(g+1) = {} & pl_i(g) + \alpha_{\mathrm{reward}}\beta(g)(1 - pl_i(g)) \\ & - \alpha_{\mathrm{penalty}}(1 - \beta(g))pl_i(g), \end{aligned} \quad (32)$$

$$\begin{aligned} pl_j(g+1) = {} & pl_j(g) - \alpha_{\mathrm{reward}}\beta(g)pl_j(g) + \alpha_{\mathrm{penalty}}(1 - \beta(g)) \\ & \cdot \left[(Nl - 1)^{-1} - pl_j(g)\right], \end{aligned} \quad (33)$$

where $pl_i(g)$ denotes the probability of selecting the ith neighborhood at the gth iteration, which is initialized as $pl_i = 1/Nl$, $i = 1, 2, \dots, Nl$, the constants α_{reward} and $\alpha_{\mathrm{penalty}}$ are the reward and penalty parameters, Nl is the number of neighborhoods for selection, and $\beta(g)$ represents the reward received by the reinforcement feedback for a neighborhood selected at iteration g; here, we define it as

$$\beta(g) = \begin{cases} 1, & \text{if better result,} \\ \dfrac{1}{Nl}, & \text{if equal result,} \\ 0, & \text{if worse result.} \end{cases} \quad (34)$$

Equation (32) is used for updating the probability of the ith neighborhood at the gth iteration, while (33) is used to update the probabilities of other neighborhoods. When $\alpha_{\mathrm{reward}} = \alpha_{\mathrm{penalty}}$ or $\alpha_{\mathrm{penalty}} = 0$, the update scheme is referred as linear reward-penalty ($L_{\text{R-P}}$) or linear reward-inaction ($L_{\text{R-I}}$), respectively. When $\alpha_{\mathrm{penalty}}$ is small compared to α_{reward}, it is referred as linear reward-ε-penalty ($L_{\text{R-}\varepsilon\text{P}}$). In this algorithm, we adopt the $L_{\text{R-I}}$ scheme because of its ε-optimality in all stationary environments [70], then $\alpha_{\mathrm{penalty}}$ can be neglected in the algorithm.

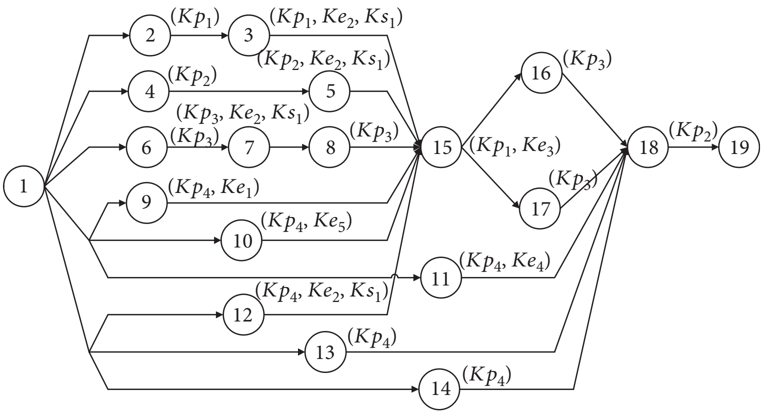


FIGURE 7: The general OONF network of a single aircraft.

Finally, the procedure of the learning-automata-based adaptive variable neighborhood strategy is summarized as follows:

Firstly, during the generation G, one individual is selected randomly from the teacher group to perform the reinforcement phase, for which the iteration number $N_{\text{iter}}(n_{\text{gs}})$ is determined according to the current number of generated schedules n_{gs}. As the algorithm focuses on the global exploration of solutions in the initial stage, the exploitation of local search increases with the evolution of the algorithm. $N_{\text{iter}}(n_{\text{gs}})$ is defined as a linearly increasing function

$$N_{\text{iter}}(n_{\text{gs}}) = N_{\text{iter}}^{\text{lb}} + \left(N_{\text{iter}}^{\text{ub}} - N_{\text{iter}}^{\text{lb}}\right) \frac{n_{\text{gs}}}{N_{\text{max}}^{\text{gs}}}. \tag{35}$$

where $N_{\text{iter}}^{\text{lb}}$ and $N_{\text{iter}}^{\text{ub}}$ represent the lower bound and upper bound of the iteration number N_{iter}. Ignoring the effect of local search in the initial stage, we set $N_{\text{iter}}^{\text{lb}} = 0$.

Secondly, in each iteration, one of the five neighborhood operators is selected based on the roulette wheel selection method. After the conduct of the neighborhood operator and evaluation on the solution, the new individual is chosen if the solution quality is improved, and the probabilities of neighborhood selection are updated according to (32) and (33). The iterative process continues until the number of iteration reaches $N_{\text{iter}}(n_{\text{gs}})$, and the best individual replaces the original one in the teacher group.

7. Simulation Experiments

To investigate the superiority of the new model and the proactive robust optimization method for the RSPFDO, a set of simulation scenarios are conducted. Firstly, the proposed HTLBO algorithm is tested under the deterministic operation durations and is compared to other algorithms for the resource-constrained (multi)project scheduling problem (RC(M)PSP) to verify the advantage of HTLBO in baseline schedule generation. Then, the proposed proactive robust optimization method is used to solve the RSPFDO under the uncertainty of stochastic operation durations. Comparisons between the proposed method and other stochastic project scheduling algorithms are also conducted. Besides, the impacts of the proposed robust resource allocation schemes and C^{PC} policy are further tested and analyzed. All the experiments are performed on a personal computer with Intel Xeon E5 (3.0 gigahertz) processor using MATLAB 2017a under Windows 7.

7.1. Simulation Case Generation. The simulation case is generated based on the flight deck as shown in Figure 3. Two typical mission cases of flight deck operations are introduced, as they represent different levels of complexity in the operational environment.

(1) Mission case I: eight aircraft are required for pre-flight preparation before launch, including three ground attack aircraft, three fighter aircraft, one early warning aircraft, and one electronic warfare aircraft. There are about 136 operations to be scheduled, which make it a medium-size scheduling problem.

(2) Mission case II: twelve aircraft are required for pre-flight preparation before launch, including five ground attack aircraft, four fighter aircraft, one early warning aircraft, one electronic warfare aircraft, and one refueling aircraft. About 204 operations will be scheduled in this case, which make it a large-size scheduling problem.

Firstly, a general OONF network, formulated from the pre-flight preparation process of single aircraft, is shown in Figure 7. There are 17 operations to be executed for each single aircraft except for the dummy start operation (numbered 1) and the dummy end operation (numbered 19). $(Kp_k, Ke_{k'}, Ks_1, Kw_{k''},)$ which, adhering to the operation, represents the requirement of the operation for each type of resource. For example, $Kp_{k'}$ indicates that personnel with trade $k'(k' \in Kp)$ is required for the operation. Note that supply resource requirement is ignored in Figure 7 because there is a one-to-one correspondence between equipment and supply resource $k' = k''$, and only one kind of workstation space (cockpit) is considered. In addition, the number regarding the kinds of personnel, equipment,

Table 7: Operation durations for a team of aircraft.

Operation	Operation duration for baseline schedule (min)												Distribution	Distribution level I (min)			Distribution level II (min)		
no.	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}	I_{11}	I_{12}	type	d-LB	d-UB	AP	d-LB	d-UB	AP
2	4	4	6	4	4	3	3	4	3	5	4	3	Uniform	$d_{i2} - 0.8$	$d_{i2} + 0.8$	—	$d_{i2} - 1.5$	$d_{i2} + 1.5$	—
3	8	8	9	7	7	6	6	7	6	9	7	6	T-normal	$d_{i3} - 0.5$	$d_{i3} + 0.5$	$\sigma = 0.3$	$d_{i3} - 1.0$	$d_{i3} + 1.0$	$\sigma = 0.8$
4	3	3	5	3	3	4	4	3	4	5	5	3	Uniform	$d_{i4} - 0.8$	$d_{i4} + 0.8$	—	$d_{i4} - 1.5$	$d_{i4} + 1.5$	—
5	6	6	7	6	6	5	6	5	5	7	6	5	T-normal	$d_{i5} - 0.5$	$d_{i5} + 0.5$	$\sigma = 0.3$	$d_{i3} - 1.0$	$d_{i3} + 1.0$	$\sigma = 0.8$
6	5	5	—	4	4	3	3	—	5	4	8	5	Uniform	$d_{i6} - 0.8$	$d_{i6} + 0.8$	—	$d_{i6} - 1.5$	$d_{i6} + 1.5$	—
7	4	4	3	5	5	3	3	—	3	3	5	5	T-normal	$d_{i7} - 0.5$	$d_{i7} + 0.5$	$\sigma = 0.3$	$d_{i7} - 1.0$	$d_{i7} + 1.0$	$\sigma = 0.8$
8	—	5	—	6	8	6	8	6	9	8	8	6	T-normal	$d_{i8} - 1.0$	$d_{i8} + 1.0$	$\sigma = 0.6$	$d_{i8} - 1.2$	$d_{i8} + 1.2$	$\sigma = 1.2$
9	13	13	18	15	15	11	11	18	12	12	15	12	T-normal	$d_{i9} - 0.1$	$d_{i9} + 0.1$	$\sigma = 0.05$	$d_{i9} - 0.3$	$d_{i9} + 0.3$	$\sigma = 0.1$
10	6	6	—	—	4	4	5	5	4	4	5	4	T-normal	$d_{i10} - 0.1$	$d_{i10} + 0.1$	$\sigma = 0.05$	$d_{i10} - 0.3$	$d_{i10} + 0.3$	$\sigma = 0.1$
11	—	3	5	—	5	3	3	—	3	5	5	3	Bernoulli	0	d_{i11}	$p = 0.4$	0	d_{i11}	$p = 0.4$
12	3	3	3	4	4	4	3	3	4	4	4	3	T-normal	$d_{i12} - 1.0$	$d_{i12} + 1.0$	$\sigma = 0.5$	$d_{i12} - 1.5$	$d_{i12} + 1.5$	$\sigma = 0.8$
13	12	12	15	12	12	12	12	15	12	12	12	12	T-normal	$d_{i13} - 1.2$	$d_{i13} + 1.5$	$\sigma = 0.7$	$d_{i13} - 1.5$	$d_{i13} + 2.0$	$\sigma = 1.0$
14	8	8	10	8	8	8	8	10	8	8	8	8	T-normal	$d_{i14} - 1.0$	$d_{i14} + 1.5$	$\sigma = 0.8$	$d_{i14} - 1.5$	$d_{i14} + 2.0$	$\sigma = 1.2$
15	3	—	5	5	3	—	3	5	3	3	3	5	Bernoulli	0	d_{i15}	$p = 0.6$	0	d_{i15}	$p = 0.6$
16	10	10	—	12	12	8	9	—	8	6	12	8	T-normal	$d_{i16} - 0.8$	$d_{i16} + 1.2$	$\sigma = 0.5$	$d_{i16} - 1.5$	$d_{i16} + 1.8$	$\sigma = 1.0$
17	10	10	—	12	12	8	9	—	8	6	12	8	T-normal	$d_{i17} - 0.8$	$d_{i17} + 1.2$	$\sigma = 0.5$	$d_{i17} - 1.5$	$d_{i17} + 1.8$	$\sigma = 1.0$
18	7	7	10	7	7	6	6	6	8	8	7	6	T-normal	$d_{i18} - 1.0$	$d_{i18} + 1.0$	$\sigma = 0.7$	$d_{i18} - 1.6$	$d_{i18} + 1.6$	$\sigma = 1.1$

and supply resource is $|Kp| = 4$ and $|Ke| = |Kw| = 5$, respectively, and the required units of personnel and equipment for all the relevant operations are 1. Suppose that all the personnel can be allocated to all the aircraft, then $Rp_{kl} = I, \forall k \in Kp, \forall l \in Lp_k$.

Table 7 shows the operation durations for a team of aircraft. The operation duration for the baseline schedule is different among aircraft according to their types and missions. The dummy start operation and end operation, denoted as No. 1 and No. 19 are neglected in the table since there are no duration and resource requirement for them. The symbol "—“ in the table indicate that the corresponding operation is unnecessary for this aircraft. Three types of distribution are considered for the uncertainty, i.e., the uniform distribution, the truncated normal (abbreviated as T-normal) distribution with standard deviation σ, and the Bernoulli distribution with probability p. The operation duration for baseline schedule is set by the expected operation duration corresponding to the distribution except that the operation duration which follows Bernoulli distribution is set by the duration when the operation is executed. This is in line with the practice in flight deck operation scheduling that once an operation (e.g., oxygen filling and nitrogen charging) stands a certain chance of being executed, it will be considered as a deterministic event in the baseline schedule. As for the distribution level, the lower bound of duration (d-LB), the upper bound of duration (d-UB), and the addition parameter (AP) of distribution (e.g., σ and p) for each level are given. Note that distribution level I has relatively little variability, while distribution level II has larger variability, which are set for analysis of the effect of distribution level on the algorithm performance.

Table 8 shows the information related to different mission cases, where the parking spot and the release time for each aircraft, available number of personnel of each trade, and the limitative makespan for different mission cases are set.

Table 9 shows the reachability relation between parking spots and support equipments. In addition, the maximum number of supported aircraft corresponding to supply resource of each type is set as $[Lw_1^t, Lw_2^t, \ldots, Lw_5^t] = [5, 5, 2, 4, 2], \quad \forall t > 0$.

7.2. Comparison with Other Algorithms for Deterministic RC(M)PSP. Since the baseline schedule serves as the principal part of the flight deck operation plan, the optimization of its makespan is the foremost to make the makespan within the limitative value as much as possible. To verify the advantages of the proposed HTLBO algorithm in solving the RSPFDO under deterministic operation durations that can be viewed as RCMPSP, the objective of HTLBO is set as min $C_{\max}(\mathbf{D}_0, \mathbf{x})$ instead of the robust one as shown in (1), and the robust resource allocation and scheduling policy are neglected in those experiments.

For convenience, when SSGS and PSGS are used in the HTLBO algorithm, HTLBO-SS and HTLBO-PS are called respectively. In addition, 4 different algorithms, i.e., multimodal genetic algorithm (MMGA) [39], modified differential evolution algorithm (MDE) [41], improved particle swarm optimization algorithm (IPSO) [71], and hybrid estimation of distribution algorithm (HEDA) [42], are used to make a comparison with the HTLBO algorithm.

Table 8: Information related to different mission cases.

Mission case	Parking spot no. (released time/min)												Number of personnel				$C^*_{\max}$
	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}	I_{11}	I_{12}	$\vert Lp_1\vert$	$\vert Lp_2\vert$	$\vert Lp_3\vert$	$\vert Lp_4\vert$	
Mission case I	2 (11)	3 (6)	4 (0)	—	—	5 (0)	—	—	6 (5)	8 (0)	9 (0)	10 (12)	4	4	6	8	60
Mission case II	1 (17)	2 (6)	3 (0)	4 (0)	5 (0)	6 (22)	7 (18)	8 (12)	9 (0)	10 (0)	11 (7)	12 (11)	5	5	8	10	70

Table 9: Reachability relation between parking spots and support equipments.

Parking spot no.	Set of reachable support equipments no.				
	Ke_1	Ke_2	Ke_3	Ke_4	Ke_5
1	[1]	[1]	[1]	[1]	[1]
2	[1]	[1, 2]	[1]	[1]	[1, 2]
3	[1, 2]	[1, 2]	[1]	[1]	[1, 2]
4	[2]	[2, 3]	[1, 2]	[1, 2]	[2, 3]
5	[2]	[3, 4]	[2]	[2]	[3]
6	[3]	[3, 4]	[2]	[2]	[3]
7	[3, 4]	[5]	[3]	[3]	[4]
8	[4, 5]	[5, 6]	[4, 5]	[3, 4]	[4]
9	[4, 5]	[6, 7]	[4, 5]	[3, 4]	[4, 5]
10	[5]	[6, 7]	[5]	[4]	[5]
11	[6]	[8, 9]	[6]	[5]	[6]
12	[6]	[8, 9]	[6]	[5]	[6]

In terms of the parameter setting, the response of algorithm under different mission cases is integrated and is set firstly.

$$R\_C_{\max} = \sum_{j=1}^{N_{\mathrm{mc}}} p_j^{\mathrm{mc}} \cdot C_{\max}^j, \tag{36}$$

where N_{mc} is the number of mission case, p_j^{mc} is the probability of occurrence of the jth mission case, and $C_{\max}^j$ is the makespan of the jth mission case. In this experiment, $N_{\mathrm{mc}} = 2$, and the probability of occurrence for each mission case can be supposed as $p_j^{\mathrm{mc}} = 0.5$. As for the existing algorithms, several better groups of parameter setting from the corresponding reference [39, 41, 42, 71] are selected, and the best set of related parameters for each algorithm is determined after several preliminary experiments based on the response definition. As for the proposed HTLBO algorithm, the set of parameters is chosen through the Taguchi method, which will be discussed in detail in the next section. Table 10 shows the parameter settings for each algorithm. Note that there are several versions of MMGA, and GA-DCA-RK is selected because it outperforms the others in the preliminary experiments.

To evaluate the accuracy and stability, experiments using these algorithms are conducted under the two mission cases. In both cases, the maximum number of schedule is set as 5000 and 10,000 to investigate the difference of solutions, and the best result (best), average result (avg.), and standard deviation (std.) after 30 independent runs are shown in Table 11. The best results demonstrate the capacity of searching for the optimal solution, and the solution quality and the robustness of the algorithm can be evaluated by the average result and standard deviation.

Table 11 shows that the performances of HTLBO-SS and HTLBO-PS are better in both accuracy and stability in the two mission cases. Furthermore, the comparison between HTLBO-SS and HTLBO-PS is conducted, and the following conclusions can be reached. Firstly, the same best results are obtained by the two algorithms in both cases, and HTLBO-SS has more chance of finding better solutions according to the average result. Note that in mission case II, the success rate of finding optimal solutions is up to 100% with the HTLBO-SS algorithm after 10,000 iterations. Although the standard deviation of HTLBO-SS is slightly larger than that of HTLBO-PS in mission case I, it outperforms HTLBO-PS in general. The reason can be found in [57] that the SSGS generates active schedules while the PSGS creates nondelay ones. Therefore, the PSGS searches in a smaller solution space than the SSGS does. Based on the above results, the SSGS is chosen to generate the baseline schedule for the RSPFDO, and HTLBO-SS is actually used in the following experiments. As for the other four algorithms in Table 11, it is obvious that IPSO and HEDA outperform the MMGA and MDE in terms of best result and average result because the double justification is used for local search in IPSO and HEDA.

To make a clearer understanding of the results, a Gantt chart is introduced to show the baseline schedule and the resource allocation plan. For example, Figures 8 and 9 show the Gantt chart of optimal personnel allocation and support resource allocation respectively in deterministic mission case II obtained by the HTLBO-SS algorithm, and the personnel allocation is generated randomly according to the baseline schedule. In the two figures, Lp_k^l represents the lth personnel of the trade $k(k \in Ke)$, $Le_{k'}^{l}$ represents the l'th support equipment of the type $k'(k' \in Ke)$, and $I_i - j$ represents operation j of aircraft i. The makespan of the baseline schedule is $C_{\max} = 65$, which is consistent with the best result in Table 11.

7.3. Comparison with Other Algorithms of Stochastic Project Scheduling. In this section, the stochastic operation durations under different distribution levels are taken into consideration. Firstly, an optimal set of parameters is selected for the HTLBO algorithm through the Taguchi method. Then, based on the selected parameters, the HTLBO algorithm is compared with other algorithms of stochastic project scheduling.

There are five key parameters to be set in the HTLBO algorithm. They are the population size Np, teacher group

Table 10: Parameter settings of the algorithms.

Algorithm	Parameters
HTLBO-SS/PS	Population size $Np = 30$, teacher group size $Nt = 3$, upper bound of the iteration for enforcement phase $N_{\text{iter}}^{\text{ub}} = 10$, and reward parameter $\alpha_{\text{reward}} = 0.05$.
MMGA	Population size 50, crossover probability 0.8, and mutation probability 0.002
MDE	Population size 100, the scale factor $F1 = 0.7$, $F2 = 0.7$, and the crossover probability CR1 = 0.3, CR2 = 0.7.
IPSO	Number of particles $M = 50$, inertial weight $w = 0.2$, self-learning factor $c_1 = 4$, social-learning factor $c_2 = 2$, and selected time periods $q = 3$.
HEDA	Population size NP = 250, the number of selected individual to update the probability matrix $P = 20\%$ NP, the learning speed $\beta = 0.5$, the PBLS accept rate Pper = 0, and the descending rate $\lambda = 0.8$.

Table 11: Result comparison between TLBO and published algorithms (deterministic case).

Case set	$N_{\text{max}}^{\text{gs}}$	Performance measurement	HTLBO-SS	HTLBO-PS	MMGA	MDE	IPSO	HEDA
Mission case I	5000	Best	55	55	58	59	56	56
		Avg.	55.65	55.75	59.97	60.60	57.73	57.47
		Std.	0.24	0.20	0.51	0.73	1.23	0.53
	10,000	Best	55	55	57	58	56	56
		Avg.	55.40	55.85	58.80	59.7	57.57	56.84
		Std.	0.25	0.13	0.44	0.42	1.36	0.18
Mission case II	5000	Best	65	65	69	71	66	67
		Avg.	65.05	65.25	70.56	72.87	68.13	68.50
		Std.	0.05	0.20	0.73	0.74	2.18	0.72
	10,000	Best	65	65	68	70	66	67
		Avg.	65	65.15	69.20	71.6	67.87	67.86
		Std.	0	0.13	0.65	0.65	1.29	0.33

size Nt, upper bound of the iteration for enforcement phase $N_{\text{iter}}^{\text{ub}}$ (also denoted as Niu), reward parameter α_{reward} (also denoted as α), and number of scenario for robustness evaluation Ns. With the four levels for five parameters given in Table 12, an orthogonal array L_{16} (4^5) with 16 parameter combinations is obtained [71]. For each distribution level, the responses of PCLM and IRM for each parameter combination are represented by a combination result of different mission cases with probabilities of occurrence. Suppose that PCLM_{ij} and IRM_{ij} are the responses of PCLM and IRM under the ith distribution level and the jth mission case, p_j^{mc} is the probability of occurrence of the jth mission case, then the responses of the ith distribution PCL M_i and IRM_i are

$$\begin{cases} \text{PCLM}_i = \sum_{j=1}^{N_{\text{mc}}} p_j^{\text{mc}} \cdot \text{PCLM}_{ij}, \\ \text{IRM}_i = \sum_{j=1}^{N_{\text{mc}}} p_j^{\text{mc}} \cdot \text{IRM}_{ij}. \end{cases} \tag{37}$$

In this experiment, the value of N_{mc} and p_j^{mc} are equal to those in the experiment in Section 7.1. The weights between EC_{max} and VC_{max} are set to $\omega = 0.1$.

In this section, the maximum iteration is set as 25,000, and 30 independent runs are carried out for each parameter combination. The influence of the parameters on the solution robustness under the two distribution levels is shown in Figures 10 and 11, respectively. The vertical axis represents the PCLM in Figures 10(a) and 11(a), and IRM in Figures 10(b) and 11(b), and the horizontal axis represents the factor levels.

It can be seen that the parameter Ns has the greatest effect on the performance under both distribution levels. In distribution level I, $Ns = 10$ results in the best performance while in level II, $Ns = 20$ is the best choice. Since the calculation of evaluation for multiscenario is a part of the algorithm and accounts for a certain proportion of the maximum number of generated schedules, less variability of duration results in fewer scenarios used for evaluation, which is consistent with the conclusion obtained in [23]. In both distribution levels, the variations of parameters show opposite trends in the PCLM and the IRM, which conform to the inverse relationship between the PCLM and the IRM. Based on the experiment results, the optimal parameter combination is set to $Np = 30$, $Nt = 5$, $Niu = 70$, $\alpha = 0.2$, and $Ns = 10$ under distribution level I, and $Np = 30$, $Nt = 5$, $Niu = 25$, $\alpha = 0.1$, and $Ns = 20$ under distribution level II.

Based on the selected parameter combinations, the HTLBO algorithm is compared with the other algorithms for stochastic project scheduling, i.e., the genetic algorithm

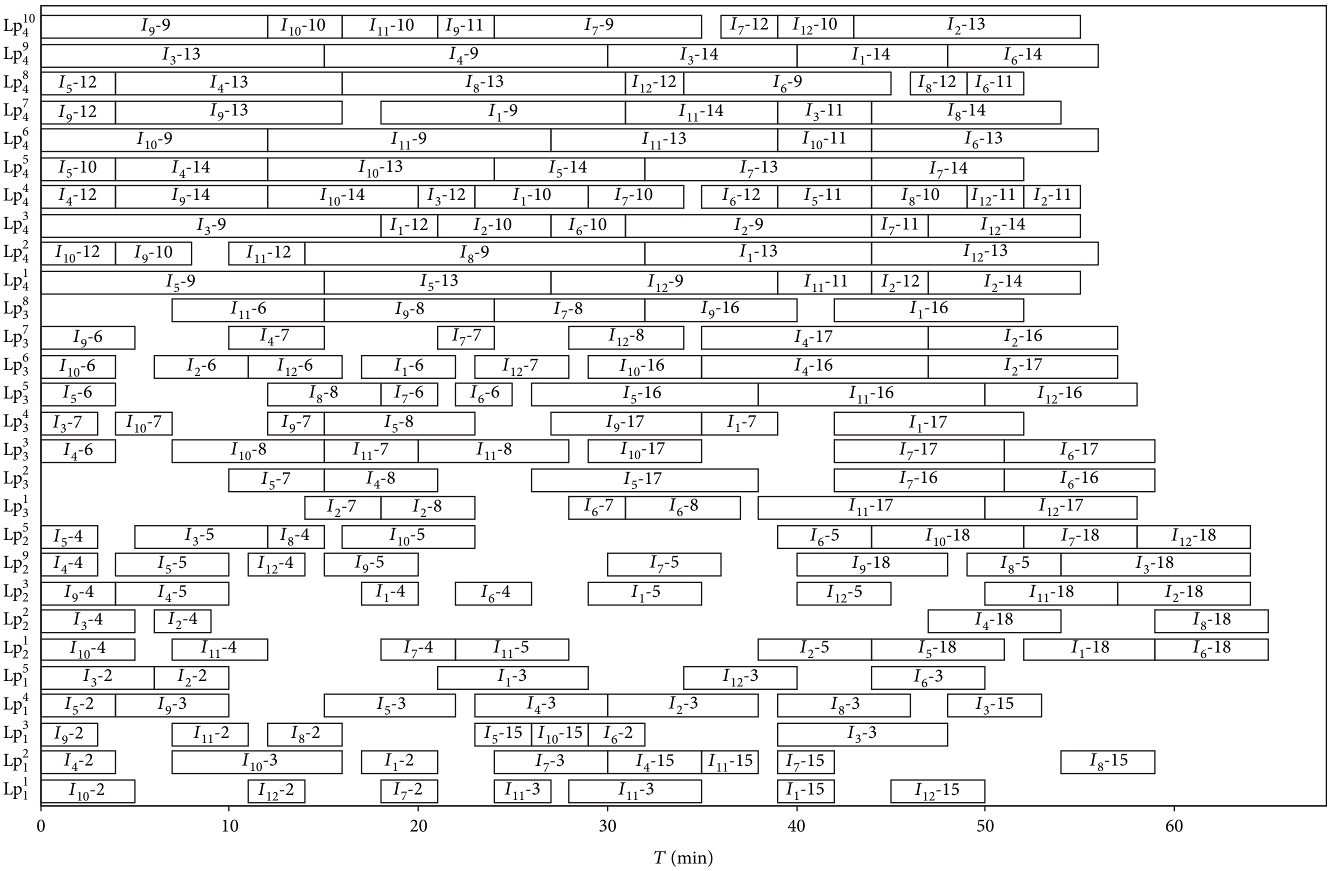


FIGURE 8: Gantt chart of optimal personnel allocation in deterministic mission case II.

with C^{AB} policy (denoted as ABGA [23]), the greedy randomized adaptive search procedure with C^{AB} policy (denoted as ABGR [72]), and the estimation of the distribution algorithm with C^{RB} policy (denoted as RBEDA [73]). The parameter settings of each algorithm are as follows: for ABGA, population size $nPob = 40$; for ABGR, the cardinality of the elite set is 80, the probability of selecting activity with latest finish time (LFT) $\text{pLFT} = 0.95$, the probability of selecting activity randomly $\text{pRandom} = 0.05$, minimum of nit $\text{nitmin} = 1$, and maximum of nit $\text{nitmax} = 30$; for the EDA, the population size of each generation $M = 250$, the size of the elite set $Q = 0.1\%\ M$, the PBLS acceptance rate $Pper = 0.1$, and the learning speed $\beta = 0.5$. For all the algorithms, numbers of scenario for robustness evaluation under the two distribution levels are set to 10.

With different termination criteria (10,000 and 25,000), the results are shown after 30 runs for each case (see Tables 13 and 14). In Tables 13 and 14, the results are depicted in the form of $[\overline{\Pr}(\tilde{C}_{\max} \le C_{\max}{}^{*}), \overline{EC}_{\max}, \overline{VC}_{\max}]$, which represent the average of $\Pr\ (\tilde{C}_{\max} \le C_{\max}{}^{*})$, $EC_{\max}$, and $VC_{\max}$ of 30 runs, respectively. The results in Tables 13 and 14 demonstrate that the HTLBO outperforms the other algorithms in terms of each criterion in both cases. In both mission cases under distribution level I, the probability of completing within the limitative makespan can reach 100%, and the probability can maintain above 94% under distribution level II with 25,000 schedules. The above results prove the effectiveness of the HTLBO algorithm in solving the RSPFDO.

As for the other algorithms, the ABGA algorithm performs the worst in general. Compared to the RBEDA algorithm, the ABGR algorithm works far better under distribution level I and slightly worse under distribution level II. This result is in line with the conclusion that the activity-based policies outperform the resource-based ones for the problems with small variability of activity durations [73]. In addition, the RBEDA with the C^{RB} policy has the far larger variance of makespan than others, which makes it not suitable for the RSPFDO.

Note that the proposed HTLBO algorithm is coded in MATLAB, the elapsed time of which can be about 500 times as that of C++ as researched by Fang et al. [73]. This will not have an influence on the performance of the HTLBO because the objective of this paper is to get results good enough rather than quick results. Table 15 shows the average CPU time under different cases and maximum number of generated schedules after 30 runs.

In general, it takes less than five minutes to solve the RSPFDO in any cases, which can satisfy the limitation of planning time. Actually, it will take less time to reach the optimal solution due to the fast convergence of the HTLBO algorithm. When comparing the average CPU time under deterministic durations with that under the two distribution

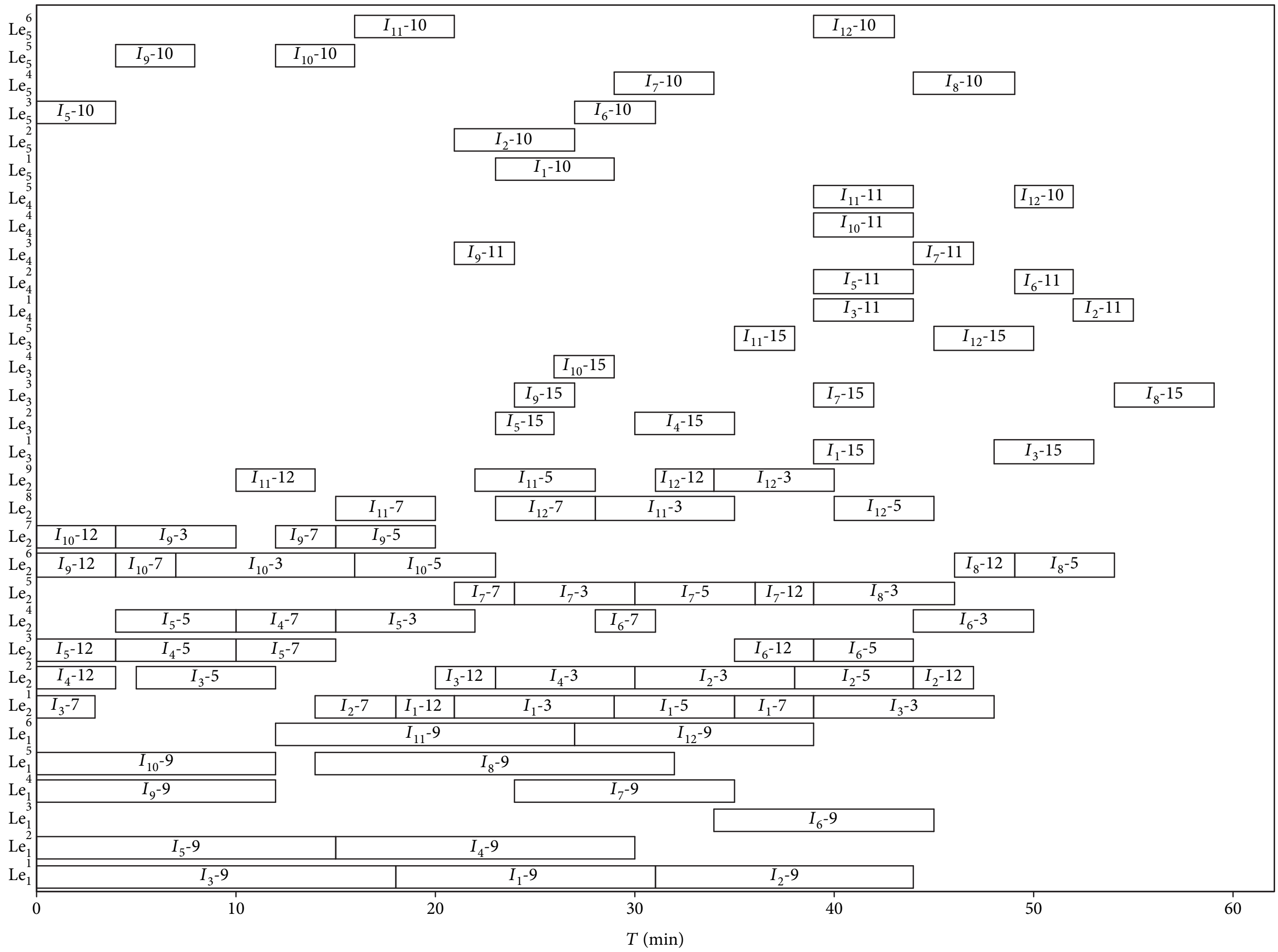


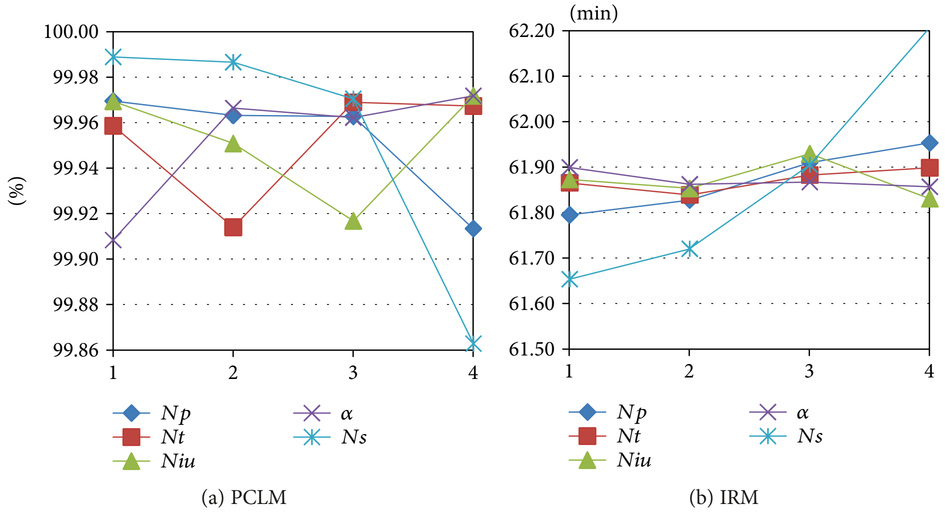
Figure 9: Gantt chart of optimal support resource allocation in deterministic mission case II.

Table 12: Combination of parameters values.

Parameters	Factor level			
	1	2	3	4
Np	30	50	80	100
Nt	1	3	5	7
Niu	15	25	50	70
α	0.01	0.05	0.1	0.2
Ns	10	20	40	70

levels, it can be found that it takes more time with the deterministic durations, and the greater the variability of durations is, the more time the algorithm will take. Under the deterministic durations, the computational time corresponds to the time required to generate $N_{\max}^{\mathrm{gs}}$ solutions with SSGS, while under the stochastic durations, a portion of time is required for robustness evaluation with the C^{PC} scheduling policy, which has turned the majority of resource constraints (except for the supply resources) into the precedence constraints, thus reducing the time of searching for and updating resource-feasible operations. Besides, when the variability of durations increases, the completion times of operations become more scattered, which results in the reduction of the number of resource-feasible operations at the same decision point t, thus increasing the time of searching and updating the resource-feasible operations.

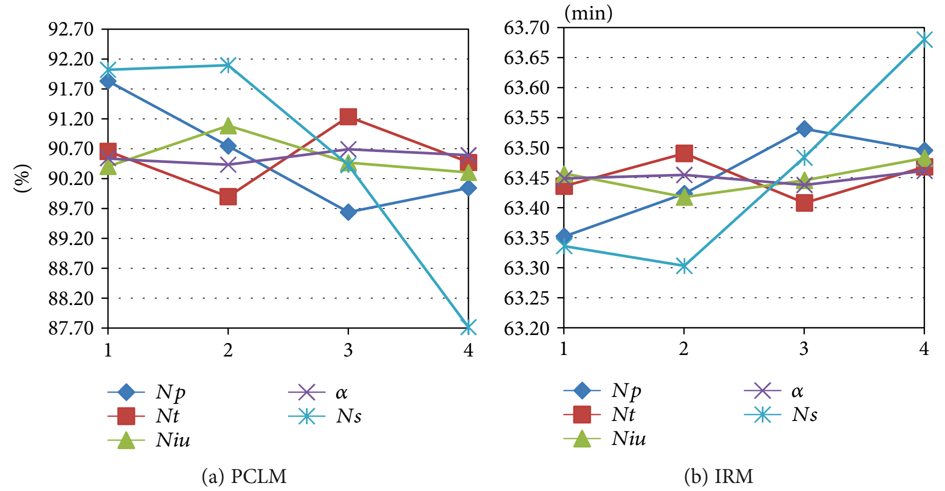
7.4. Experiment Results with Resource Allocation Schemes and Scheduling Policies. In this section, under the framework of the HTLBO algorithm, the results under different resource allocation scheme combinations and scheduling policies are compared and analyzed. The resource allocation scheme combinations are constructed according to whether the robust personnel allocation scheme and the robust equipment adjustment scheme are applied, and four kinds of combinations can be selected. They are "Yes-Yes," "Yes-No," "No-Yes," and "No-No," where "Yes" indicates that the corresponding resource allocation scheme is applied, while "No" indicates the opposite. If the robust personnel allocation scheme is not used, the personnel will be allocated randomly. Similarly, if the robust equipment adjustment scheme is not applied, the initial equipment allocation generated along with the baseline schedule by SSGS will be the final



(a) PCLM

(b) IRM

Figure 10: Influence of the parameters on the robust objectives under distribution level I.



(a) PCLM

(b) IRM

Figure 11: Influence of the parameters on the robust objectives under distribution level II.

decision plan. To verify the performance of the C^{PC} scheduling policy, another four scheduling policies, i.e., roadrunner, railway, C^{RB}, and C^{AB}, are used to make comparisons.

With the selected parameter combinations obtained in Section 7.3 and terminal condition of 25,000 schedules, the results of different resource allocation scheme combinations and scheduling policies are demonstrated in Table 16 after 30 runs for each case. Furthermore, to make a clear presentation, the trade-off relationship is depicted between PCLM and IRM of different resource allocation scheme combinations and scheduling policies, under different mission cases and distribution levels, as shown in Figures 12 and 13. The results of C^{RB} and C^{AB} are ignored in the figures because of their poor performance. In each figure, the x-axis represents the IRM and the y-axis denotes the PCLM, and the result of a certain resource allocation scheme combination along with a scheduling policy is remarked as "scheduling policy-Y/N-Y/N." For instance, "C^{PC}-Y-N" denotes the result obtained by the C^{PC} scheduling policy, the robust personnel allocation scheme, and not using of the robust equipment adjustment scheme. Combining the result in Table 16 with demonstration in Figures 12 and 13, the influences of different resource allocation scheme combinations and scheduling policies on the results are analyzed as follows.

The influence of the scheduling policy is explained firstly. The two online reactive scheduling policies, i.e., C^{RB} and C^{AB}, perform worse than the other policies. To be specific, the performances of C^{AB} are close to those of the other policies under distribution level I in both mission cases, but far worse under distribution level II with high variability. C^{RB} works even worse than C^{AB} except under distribution level II in mission case II. With the same resource allocation scheme combination, C^{PC} outperforms the roadrunner and railway in terms of both PCLM and IRM, while the results of roadrunner and railway vary under different distribution levels and mission cases. In most cases, railway is better than roadrunner. It also can be found in Table 16 that regardless of

Table 13: Results of the four algorithms (10,000 schedules).

Algorithm	Mission case I		Mission case II	
	Distribution level I	Distribution level II	Distribution level I	Distribution level II
HTLBO	[0.9998, 56.83, 0.65]	[0.9078, 58.27, 1.53]	[0.9998, 66.87, 0.64]	[0.8797, 68.48, 1.56]
ABGR	[0.9282, 58.73, 0.67]	[0.3677, 60.51, 1.58]	[0.9650, 68.35, 0.73]	[0.3810, 70.45, 1.65]
ABGA	[0.4638, 60.35, 0.66]	[0.1136, 61.85, 1.86]	[0.3129, 70.37, 0.81]	[0.0120, 73.60, 2.37]
RBEDA	[0.6360, 59.57, 3.52]	[0.4281, 60.59, 5.12]	[0.5732, 70.23, 4.56]	[0.4728, 70.51, 5.74]

Table 14: Result comparison with other algorithms (25,000 schedules).

Algorithm	Mission case I		Mission case II	
	Distribution level I	Distribution level II	Distribution level I	Distribution level II
HTBLO	[1.0000, 56.65, 0.61]	[0.9449, 57.94, 1.47]	[1.0000, 66.34, 0.53]	[0.9457, 67.89, 1.46]
ABGR	[0.9640, 58.43, 0.68]	[0.4503, 60.25, 1.59]	[0.9772, 68.07, 0.68]	[0.4484, 70.25, 1.50]
ABGA	[0.6385, 59.65, 0.62]	[0.1885, 61.23, 1.62]	[0.4216, 70.28, 0.57]	[0.0332, 72.16, 2.05]
RBEDA	[0.6825, 59.41, 3.13]	[0.4629, 60.31, 4.11]	[0.6866, 69.32, 3.79]	[0.5163, 70.31, 5.81]

Table 15: Comparison of average CPU time.

$N_{\max}^{\text{gs}}$	Mission case I			Mission case II		
	Deterministic durations (s)	Distribution level I (s)	Distribution level II (s)	Deterministic durations (s)	Distribution level I (s)	Distribution level II (s)
5000	50.9	42.35	42.40	79.68	62.51	62.75
10,000	98.63	83.30	84.59	171.13	125.83	126.48
25,000	250.16	174.18	198.07	388.76	276.12	298.92

the resource allocation scheme combination, railway works out the minimum variance of the makespan, while roadrunner performs the last but one, which precedes C^{RB}, which confirms the results of Tian and Demeulemeester [62]. The variance of the makespan of C^{PC} is between than that of railway and roadrunner and close to that of C^{AB}. To sum up, C^{PC} combines the advantages of all the other scheduling policies and works best in terms of optimizing PCLM and the average of makespan, and the variance of the makespan is moderate.

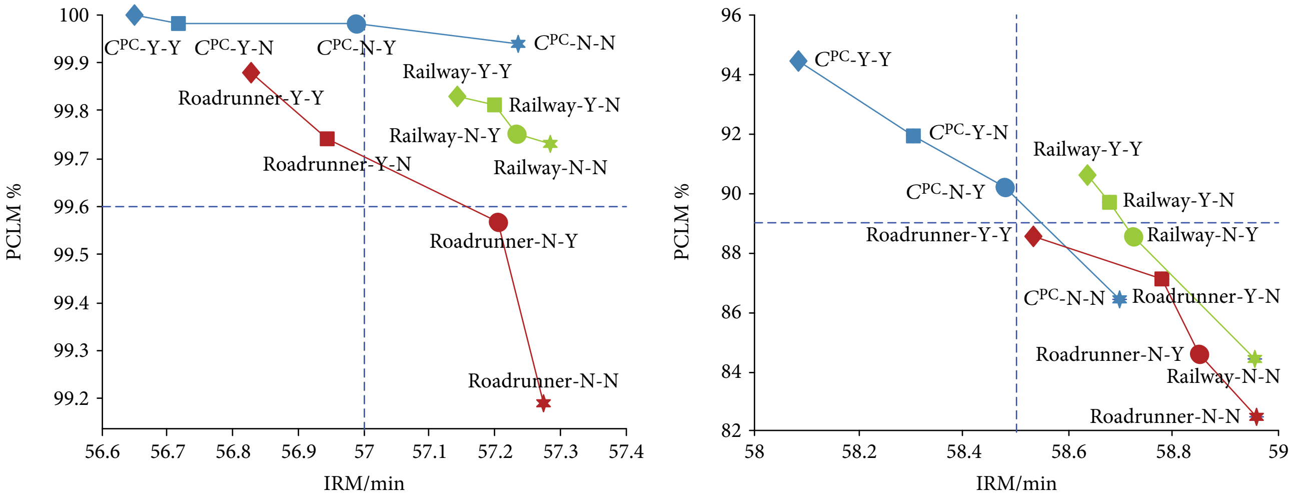
As for the influence of the resource allocation scheme, it is obvious that both PCLM and IRM improve with the application of the robust personnel allocation scheme and robust equipment adjustment scheme for all the scheduling policies except for C^{RB} and C^{AB} because the resource allocation is not needed in the two scheduling policies. Note that when only one of the two schemes is used, the improvement made by the robust personnel allocation scheme is greater than that by the robust equipment adjustment scheme for all the scheduling policies, and the underlying reason may stem from two aspects. Firstly, in the flight deck operations, the requirement for personnel is greater than that for support equipment, as shown in Figures 8 and 9. Therefore, the influence of the robust personnel allocation scheme is more comprehensive than the other. Secondly, since the robust equipment adjustment scheme is based on an initial allocation, it has limited the degree of adjusting the operations from the beginning under the range constraint of equipment, thus weakening the function of robust allocation. As for the personnel allocation, it is developed from nothing. In addition, from Figures 12 and 13, it reveals that the effects of resource allocation scheme combinations on the robustness criteria vary in different distribution levels and different policies. Firstly, under distribution level I, C^{PC} has less influence on the PCLM than the IRM. Railway has less influence on both PCLM and IRM. However, different from C^{PC}, the roadrunner has more influence on PCLM than IRM. Secondly, under distribution level II, the ranges of both PCLM and IRM become larger in any scheduling policy due to the increase in variability. In mission case I, C^{PC} has the largest ranges of both PCLM and IRM, while in mission case II, the roadrunner does. Furthermore, it can be seen from Table 16 that the variance of the makespan is reduced by the application of resource allocation schemes, but the effects are not so obvious.

Note that the above comparisons and analysis of different resource allocation scheme combinations and scheduling policies are based on the framework of the proposed proactive robust optimization method. To further compare and analyze the effects of different resource allocation scheme combinations and scheduling policies at a micro level, the specific baseline schedule and resource allocation is adopted, as shown in Figures 8 and 9.

Firstly, as the personnel allocation is generated randomly in Figure 8, the robust personnel allocation scheme is applied to reallocate the personnel for each operation according to the baseline schedule, and the new personnel allocation is shown in Figure 14. Likewise, based on the initial equipment allocation plan in Figure 9, the robust equipment adjustment scheme is applied to update the allocation of equipment.

Table 16: Results of resource allocation scheme combinations and scheduling policies (25,000 schedule).

Scheduling policy	Robust personnel allocation	Robust equipment adjustment	Mission case I		Mission case II	
			Distribution level I	Distribution level II	Distribution level I	Distribution level II
C^{PC}	Yes	Yes	[1.0000, 56.59, 0.61]	[0.9449, 57.94, 1.47]	[1.0000, 66.34, 0.53]	[0.9457, 67.89, 1.46]
	No	Yes	[0.9998, 56.92, 0.68]	[0.9022, 58.32, 1.59]	[0.9999, 66.77, 0.59]	[0.8712, 68.56, 1.47]
	Yes	No	[0.9998, 56.65, 0.67]	[0.9198, 58.15, 1.56]	[0.9999, 66.48, 0.60]	[0.9359, 68.09, 1.46]
	No	No	[0.9994, 57.17, 0.67]	[0.8644, 58.54, 1.58]	[0.9995, 66.93, 0.58]	[0.8607, 68.69, 1.50]
Roadrunner	Yes	Yes	[0.9988, 56.75, 0.79]	[0.8857, 58.36, 1.74]	[0.9948, 66.89, 0.85]	[0.9184, 68.09, 1.76]
	No	Yes	[0.9957, 57.12, 0.85]	[0.8462, 58.66, 1.91]	[0.9869, 67.28, 1.02]	[0.8261, 68.68, 1.89]
	Yes	No	[0.9974, 56.86, 0.84]	[0.8715, 58.60, 1.78]	[0.9911, 67.00, 0.99]	[0.8963, 68.24, 1.78]
	No	No	[0.9919, 57.19, 0.85]	[0.8248, 58.76, 2.01]	[0.9826, 67.32, 1.07]	[0.8024, 69.06, 2.09]
Railway	Yes	Yes	[0.9983, 57.09, 0.54]	[0.9061, 58.52, 1.17]	[0.9988, 66.91, 0.45]	[0.9324, 68.24, 1.17]
	No	Yes	[0.9975, 57.18, 0.54]	[0.8856, 58.60, 1.25]	[0.9975, 67.03, 0.50]	[0.8780, 68.66, 1.27]
	Yes	No	[0.9981, 57.15, 0.50]	[0.8972, 58.56, 1.19]	[0.9983, 66.95, 0.53]	[0.9257, 68.34, 1.19]
	No	No	[0.9973, 57.23, 0.55]	[0.8444, 58.83, 1.27]	[0.9963, 67.12, 0.54]	[0.8369, 68.85, 1.42]
C^{RB}	—	—	[0.7153, 59.11, 3.15]	[0.5011, 60.16, 3.96]	[0.7963, 68.62, 3.62]	[0.6730, 69.32, 5.52]
C^{AB}	—	—	[0.9926, 57.71, 0.63]	[0.7343, 59.35, 1.47]	[0.9915, 67.62, 0.55]	[0.5524, 69.94, 1.46]



(a) Distribution level I

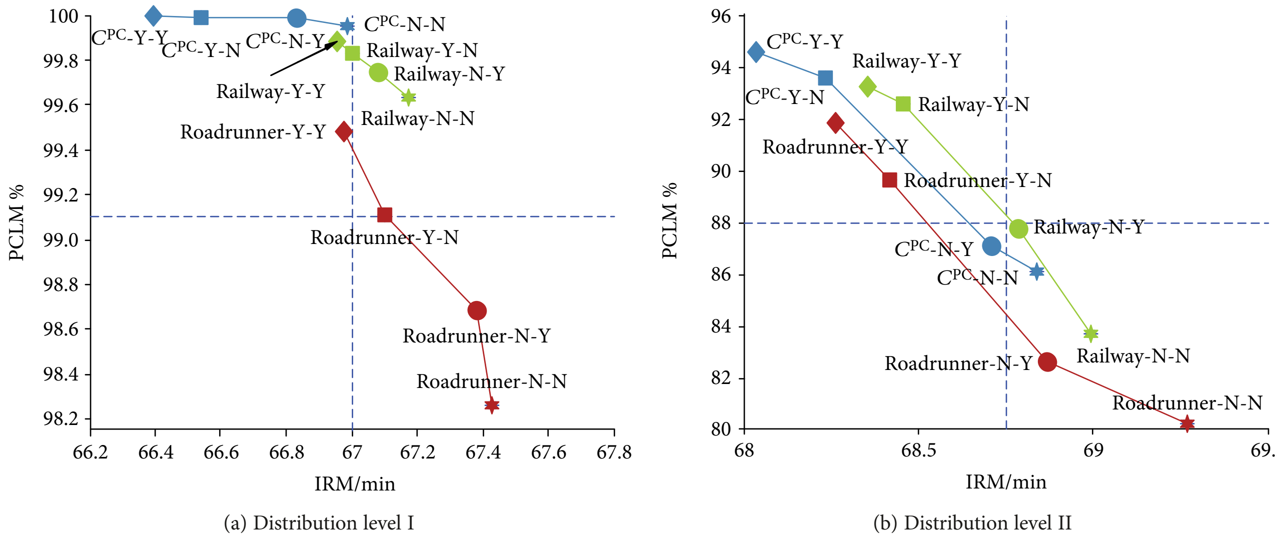
(b) Distribution level II

Figure 12: Results of resource allocation scheme combinations and scheduling policies under mission case I.

Considering the fact that some kinds of equipments are used for only one operation, little adjustment is made on them. The Gantt chart of robust support resource allocation of the 2nd type which is required most frequently is depicted, as shown in Figure 15. From Figures 14 and 15, it is noteworthy that more operations with precedence relation are allocated to the same personnel or equipment compared to the origin allocations.

Then, based on the baseline schedule and robust personnel and equipment allocation, the five scheduling policies are applied for simulation experiment under the two distribution levels with 3000 replications for each policy. Figure 16 shows the resulting distributions of the makespan with different scheduling policies. The boxplots show the median value as a horizontal dashed line, and the black diagonal lines emanating from the median denote the 95% confidence interval of the median. The edges of each box refer to the 25th (Q1) and 75th (Q3) percentiles while the red asterisks represent the outliers of the makespan. The horizontal black lines (connected by the dashed line whiskers) signifies extreme nonoutlier values and are subdivided into upper adjacent value (UAV) and lower adjacent value (LAV).

As is revealed in Figures 16(a) and 16(b), the minimum median, which means the best $EC_{\max}$, is obtained under C^{PC}. The minimum Q1 and UAV imply a larger PCLM, and the minimum Q3 and LAV indicate the capacity of completing the operations as soon as possible. The advantage of C^{PC} is independent of the distribution level. However, railway has the minimum interquartile range (IQR, equal to Q3-Q1), which measures the stability of the makespan.



(a) Distribution level I

(b) Distribution level II

Figure 13: Results of resource allocation scheme combinations and scheduling policies under mission case II.

C^{RB} still performs the worst, and C^{AB} is competitive under distribution level I but falls behind the other policies except that C^{AB} performs better than C^{RB} under distribution level II, as is similarly shown in Table 16.

Finally, comparison in resource allocation scheme combinations is conducted, and four combinations are applied for simulation experiment under the two distribution levels with 3000 replications by the C^{PC} policy. Figures 16(b) and 17(a) demonstrate the distributions of the makespan with different resource allocation scheme combinations. In Figures 17(a) and 17(b), the average of the makespan is improved with the application of the robust personnel allocation scheme and robust equipment adjustment scheme, and the improvement made by the robust personnel allocation scheme is greater than that by the robust equipment adjustment scheme. The improvement is independent of the distribution level as well. The difference between the IQR of each scheme combination is not much obvious, which is consistent with the results shown in Table 16, which indicates that the application of the robust resource allocation scheme exhibits a relatively little effect on the stability of the makespan.

On the basis of the above computational results, some conclusions and managerial insights are summarized as follows.

(1) The generation of schedule with smaller expectation of the makespan is an important content of the robust optimization for the RSPFDO. Under the deterministic expectation of durations, the proposed HTLBO with both SSGS and PSGS outperforms some state-of-the-art algorithms for deterministic RC(M)PSP in both accuracy and stability, and the SSGS performs the best due to its active nature. The results indicate the superiority of HTLBO in deterministic RC(M)PSP.

(2) In terms of resource allocation, both the proposed robust personnel allocation scheme and the robust equipment adjustment scheme can help improve the robustness of the plan. Since the requirement for equipments is far less than that of personnel in the flight deck operations, the improvement made by the robust personnel allocation scheme is greater than that by the robust equipment adjustment scheme.

(3) Under the stochastic operation durations with either higher or lower variability, the proposed HTLBO performs much better than some main algorithms for stochastic project scheduling, and the proposed preconstraint policy (C^{PC}) which considers all kinds of constraints in the RSPFDO outperforms the main existing scheduling policies in terms of both PCLM and IRM. The advantages of C^{PC} become more obvious when dealing with higher variability. In general, the proposed HTLBO is proved to be an effective method for the RSPFDO.

8. Conclusions and Future Work

In this paper, the robust scheduling problem for flight deck operations (RSPFDO) under duration uncertainty is studied. Literature investigation shows that the RSPFDO is of great significance and is hard to solve because of complicated constraints. To address this challenge, a proactive robust optimization method for the problem is proposed. Firstly, the procedures for flight deck operations, various kinds of constraints of resource, and expression of duration uncertainty are described. On this basis, a comprehensive mathematical formulation for RSPFDO is established with the robust objectives of maximizing the probability of completing within the limitative makespan (PCLM) and minimizing the weighted sum of expected makespan and variance of makespan (IRM). The optimal solution for the RSPFDO contains a baseline schedule and a personnel and support equipment allocation plan. Subsequently, to generate a robust solution, both serial and parallel schedule generation

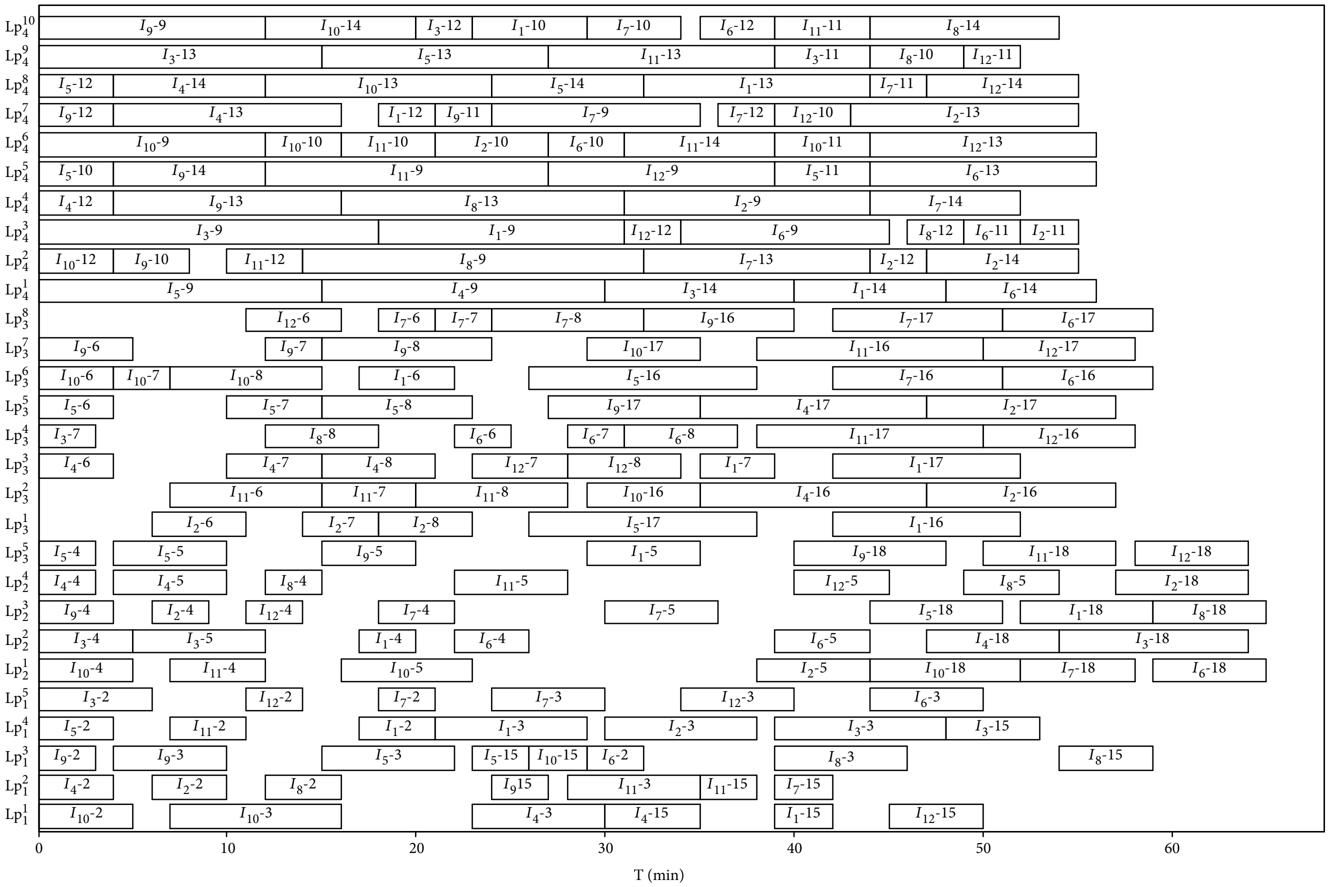


Figure 14: Gantt chart of robust personnel allocation.

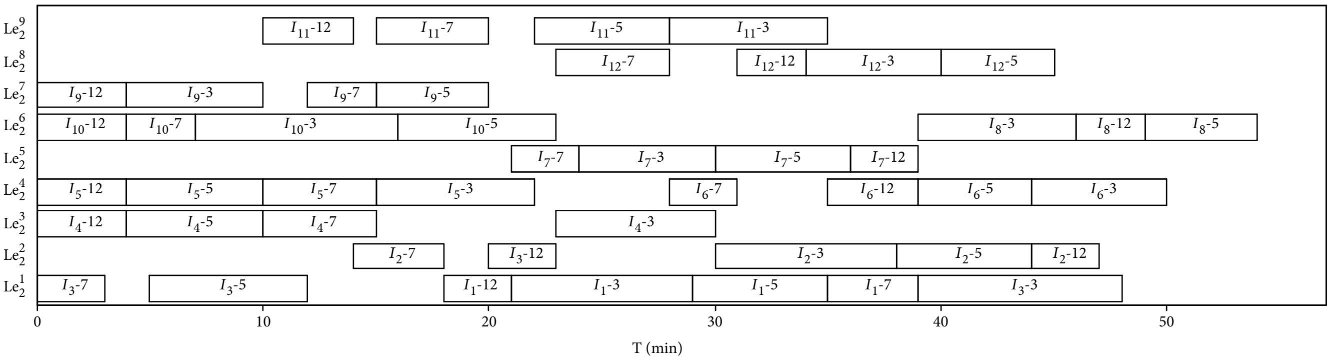
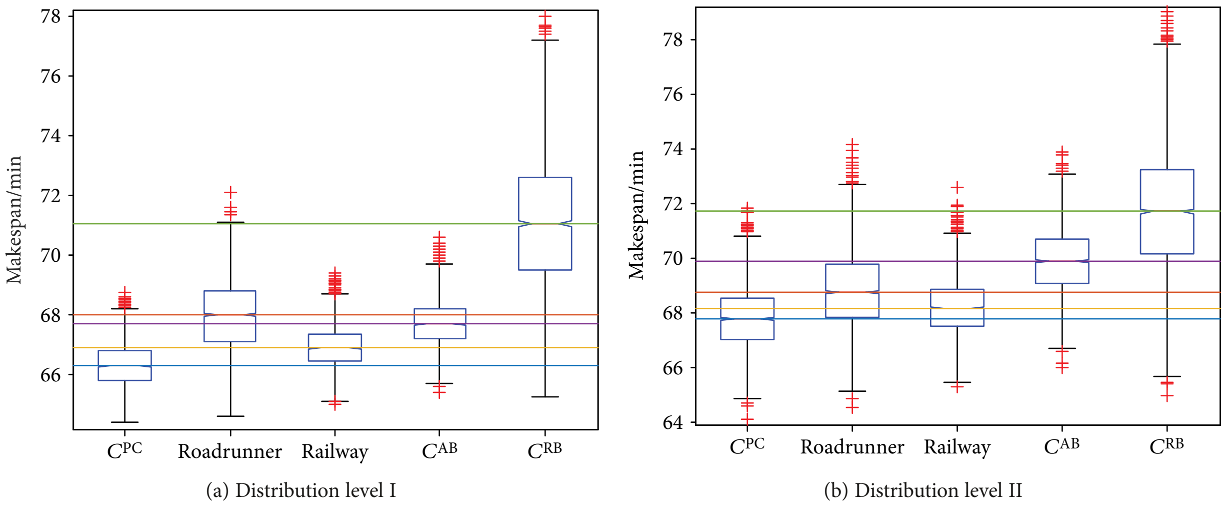


Figure 15: Gantt chart of robust support resource allocation of the second type.

schemes for baseline schedule are presented, and a robust personnel allocation scheme and equipment allocation adjustment scheme are designed accordingly for resource allocation. Furthermore, in terms of scheduling implementation, an RSPFDO-oriented preconstraint scheduling policy (C^{PC}) is developed to turn baseline schedules into executed schedules. Besides, a hybrid teaching-learning-based optimization (HTLBO) algorithm for proactive robust scheduling optimization of the RSPFDO is proposed, with which the baseline schedule along with the personnel and equipment allocation can be optimized continuously.

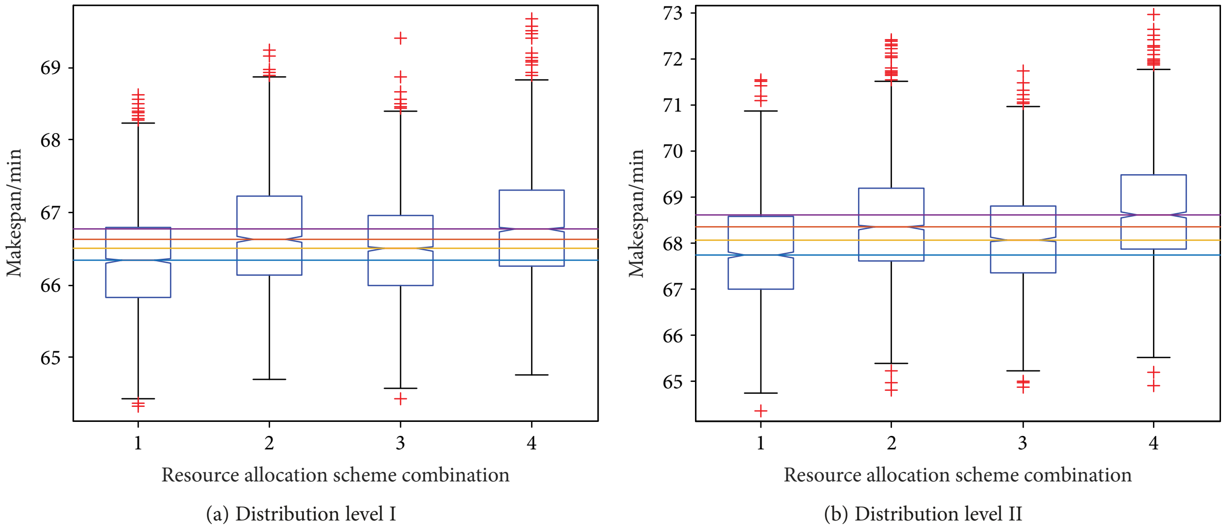
To investigate the validity and superiority of the proposed method for solving the RSPFDO, a series of experiments are conducted. According to the experimental results, the following conclusions can be made. Firstly, under the deterministic durations, HTLBO is competitive compared to some state-of-the-art algorithms for deterministic RC(M)PSP, and HTLBO with serial schedule generation scheme (SSGS) outperforms that with the parallel schedule generation scheme (PSGS). Secondly, considering the stochastic operation durations, the proposed HTLBO performs much better than some main algorithms for stochastic



(a) Distribution level I

(b) Distribution level II

FIGURE 16: Boxplot of makespan with different scheduling policies.



(a) Distribution level I

(b) Distribution level II

FIGURE 17: Boxplot of makespan with different resource allocation scheme combinations (1 for "C^{PC}-Y-Y", 2 for "C^{PC}-N-Y", 3 for "C^{PC}-Y-N", and 4 for "C^{PC}-N-N").

project scheduling. Thirdly, the objective of the RSPFDO is improved with the application of the robust personnel allocation scheme and robust equipment adjustment scheme, and the proposed C^{PC} outperforms the main existing scheduling policies in terms of both PCLM and IRM. In general, the proposed HTLBO is proved to be an effective method for the RSPFDO.

In the future, an improved mathematical model for the RSPFDO is expected. For example, the setup times and transfer times of personnel or support equipments between operations will be considered, and the stages of respot and turnaround for launching will be incorporated. The second is that major disruptions such as breakdowns of aircraft or equipments should be taken into consideration; thus, the scheduling problem for flight deck operations under uncertainty can be solved in virtue of proactive-reactive scheduling. Besides, the influence of different resources on the flight deck operations will be analyzed to further optimize the resource configuration, especially the number of personnel of each trade and the arrangement of stationary deck resources which are the two key factors.

Data Availability

The data about the raw data and the experimental data and analysis results has been presented in the article.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the National Natural Science Foundation of China NO. 51375490. The authors highly appreciate the valuable comments provided by the reviewers.

References

[1] A. Liu and K. Liu, “Advances in carrier-based aircraft deck operation scheduling,” *Systems Engineering - Theory & Practice*, vol. 37, no. 1, pp. 49–60, 2017.

[2] US Naval Air Systems Command, *CVN Flight-Hanger Deck NATOPS Manual*, NACYAIR 00-80T-120, Washington, DC, USA, 2008.

[3] A. Jewell, M. A. Wigge, C. M. Gagnon, M. Colleen, L. A. Lynn, and K. M. Kirk, *USS Nimitz and Carrier Airwing Nine Surge Demonstration*, US Navy: Center for Naval Analyses, 1998.

[4] A. Jewell, *Sortie generation capacity of embarked airwings*, US Navy: Center for Naval Analyses, 1998.

[5] T. Jiang, X. Su, and W. Han, “Optimization of support scheduling on deck of carrier aircraft based on improved differential evolution algorithm,” in *2017 3rd IEEE International Conference on Control Science and Systems Engineering (ICCSSE)*, pp. 136–140, Beijing, China, 2017.

[6] E. Demeulemeester, “Robust project scheduling,” *Foundations and Trends® in Technology, Information and Operations Management*, vol. 3, no. 3-4, pp. 201–376, 2009.

[7] R. G. Dastidar and E. Frazzoli, “A queueing network based approach to distributed aircraft carrier deck scheduling,” in *Infotech@Aerospace 2011*, St. Louis, MO, USA, 2011.

[8] L. Yu, C. Zhu, J. Shi, and W. Zhang, “An extended flexible job shop scheduling model for flight deck scheduling with priority, parallel operations, and sequence flexibility,” *Scientific Programming*, vol. 2017, Article ID 2463252, 15 pages, 2017.

[9] W. W. Shi, W. Han, and W. C. Si, “Optimization of direct flight line maintenance process for multi-carrier planes,” *Computer Engineering and Design*, vol. 34, no. 12, pp. 4214–4219, 2013.

[10] J. C. Ryan, A. G. Banerjee, M. L. Cummings, and N. Roy, “Comparing the performance of expert user heuristics and an integer linear program in aircraft carrier deck operations,” *IEEE Transactions on Cybernetics*, vol. 44, no. 6, pp. 761–773, 2014.

[11] X. C. Su, W. Han, W. Xiao, and T. T. Jiang, “Pit-stop support scheduling on deck of carrier plane based on memetic algorithm,” *Systems Engineering and Electronics*, vol. 38, no. 10, pp. 2303–2309, 2016.

[12] S. Tao, C. Wu, Z. Sheng, and X. Wang, “Space-time repetitive project scheduling considering location and congestion,” *Journal of Computing in Civil Engineering*, vol. 32, no. 3, 2018.

[13] Y. Wu, X. Pan, R. Kang, C. He, and L. Gong, “Multi-parameters uncertainty analysis of logistic support process based on GERT,” *Journal of Systems Engineering and Electronics*, vol. 25, no. 6, pp. 1011–1019, 2014.

[14] J. C. Ryan, M. L. Cummings, N. Roy, A. Banerjee, and A. Schulte, “Designing an interactive local and global decision support system for aircraft carrier deck scheduling,” in *Infotech@Aerospace 2011*, pp. 1–12, St. Louis, MO, USA, 2011.

[15] Q. Feng, S. K. Zeng, and R. Kang, “A MAS-based model for dynamic scheduling of carrier aircraft,” *Acta Aeronautica Et Astronautica Sinica*, vol. 30, no. 11, pp. 2119–2125, 2009.

[16] Q. Feng, S. Li, and B. Sun, “A multi-agent based intelligent configuration method for aircraft fleet maintenance personnel,” *Chinese Journal of Aeronautics*, vol. 27, no. 2, pp. 280–290, 2014.

[17] J. C. Ryan and M. L. Cummings, “A systems analysis of the introduction of unmanned aircraft into aircraft carrier operations,” *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 2, pp. 209–220, 2016.

[18] C. Qi and D. Wang, “Dynamic aircraft carrier flight deck task planning based on HTN,” *IFAC-PapersOnLine*, vol. 49, no. 12, pp. 1608–1613, 2016.

[19] Y. Wu and X. Qu, “Obstacle avoidance and path planning for carrier aircraft launching,” *Chinese Journal of Aeronautics*, vol. 28, no. 3, pp. 695–703, 2015.

[20] B. Michini and J. P. How, “A human-interactive course of action planner for aircraft carrier deck operations,” in *Infotech@Aerospace 2011*, pp. 1–11, St. Louis, MO, USA, 2011.

[21] Q. Feng, W. Bi, B. Sun, and Y. Ren, “Dynamic scheduling of carrier aircraft based on improved ant colony algorithm under disruption and strong constraint,” in *2017 Second International Conference on Reliability Systems Engineering (ICRSE)*, pp. 1–9, Beijing, China, 2017.

[22] O. Lambrechts, E. Demeulemeester, and W. Herroelen, “Time slack-based techniques for robust project scheduling subject to resource uncertainty,” *Annals of Operations Research*, vol. 186, no. 1, pp. 443–464, 2011.

[23] F. Ballestín, “When it is worthwhile to work with the stochastic RCPSP?,” *Journal of Scheduling*, vol. 10, no. 3, pp. 153–166, 2007.

[24] M. Masmoudi and A. Haït, “Project scheduling under uncertainty using fuzzy modelling and solving techniques,” *Engineering Applications of Artificial Intelligence*, vol. 26, no. 1, pp. 135–149, 2013.

[25] H. Li, Z. Xu, L. Xiong, and Y. Liu, “Robust proactive project scheduling model for the stochastic discrete time/cost tradeoff problem,” *Discrete Dynamics in Nature and Society*, vol. 2015, Article ID 586087, 10 pages, 2015.

[26] W. X. Wang, X. Wang, X. L. Ge, and L. Deng, “Multi-objective optimization model for multi-project scheduling on critical chain,” *Advances in Engineering Software*, vol. 68, no. 1, pp. 33–39, 2014.

[27] S. Van de Vonder, E. Demeulemeester, and W. Herroelen, “Proactive heuristic procedures for robust project scheduling: an experimental analysis,” *European Journal of Operational Research*, vol. 189, no. 3, pp. 723–733, 2008.

[28] F. Deblaere, E. Demeulemeester, W. Herroelen, and S. van de Vonder, “Robust resource allocation decisions in resource-constrained projects,” *Decision Sciences*, vol. 38, no. 1, pp. 5–37, 2007.

[29] N. Pang, H. Su, and Y. Shi, “Project robust scheduling based on the scattered buffer technology,” *Applied Sciences*, vol. 8, no. 4, p. 541, 2018.

[30] P. Lamas and E. Demeulemeester, “A purely proactive scheduling procedure for the resource-constrained project scheduling problem with stochastic activity durations,” *Journal of Scheduling*, vol. 19, no. 4, pp. 409–428, 2016.

[31] M. E. Bruni, L. di Puglia Pugliese, P. Beraldi, and F. Guerriero, “An adjustable robust optimization model for the resource-constrained project scheduling problem with uncertain activity durations,” *Omega*, vol. 71, pp. 66–84, 2017.

[32] P. Ghoddousi, R. Ansari, and A. Makui, "An improved robust buffer allocation method for the project scheduling problem," *Engineering Optimization*, vol. 49, no. 4, pp. 718–731, 2017.

[33] R. K. Chakrabortty, R. A. Sarker, and D. L. Essam, "Resource constrained project scheduling with uncertain activity durations," *Computers & Industrial Engineering*, vol. 112, pp. 537–550, 2017.

[34] S. Rostami, S. Creemers, and R. Leus, "New strategies for stochastic resource-constrained project scheduling," *Journal of Scheduling*, vol. 21, no. 3, pp. 349–365, 2018.

[35] S. Tao, C. Wu, Z. Sheng, and X. Wang, "Stochastic project scheduling with hierarchical alternatives," *Applied Mathematical Modelling*, vol. 58, pp. 181–202, 2018.

[36] H. Li and N. K. Womer, "Solving stochastic resource-constrained project scheduling problems by closed-loop approximate dynamic programming," *European Journal of Operational Research*, vol. 246, no. 1, pp. 20–33, 2015.

[37] D. Morillo Torres, L. F. Moreno Velasquez, and F. J. Díaz Serna, "A branch and bound hybrid algorithm with four deterministic heuristics for the resource constrained project scheduling problem (RCPSP)," *DYNA*, vol. 82, no. 190, pp. 198–207, 2015.

[38] S. Chand, Q. Huynh, H. Singh, T. Ray, and M. Wagner, "On the use of genetic programming to evolve priority rules for resource constrained project scheduling problems," *Information Sciences*, vol. 432, pp. 146–163, 2018.

[39] J. F. Gonçalves, J. J. M. Mendes, and M. G. C. Resende, "A genetic algorithm for the resource constrained multi-project scheduling problem," *European Journal of Operational Research*, vol. 189, no. 3, pp. 1171–1190, 2008.

[40] E. Pérez, M. Posada, and A. Lorenzana, "Taking advantage of solving the resource constrained multi-project scheduling problems using multi-modal genetic algorithms," *Soft Computing*, vol. 20, no. 5, pp. 1879–1896, 2016.

[41] R. Yan, W. Li, P. Jiang, Y. Zhou, and G. Wu, "A modified differential evolution algorithm for resource constrained multi-project scheduling problem," *Journal of Computers*, vol. 9, no. 8, p. 1922, 2014.

[42] L. Wang and C. Fang, "A hybrid estimation of distribution algorithm for solving the resource-constrained project scheduling problem," *Expert Systems with Applications*, vol. 39, no. 3, pp. 2451–2460, 2012.

[43] J. Tian, X. Hao, and T. Murata, "Robust optimization method based on hybridization of GA and MMEDA for resource constraint project scheduling with uncertainty," *IEEJ Transactions on Electronics, Information and Systems*, vol. 137, no. 7, pp. 957–966, 2017.

[44] S. Elsayed, R. Sarker, T. Ray, and C. C. Coello, "Consolidated optimization algorithm for resource-constrained project scheduling problems," *Information Sciences*, vol. 418-419, pp. 346–362, 2017.

[45] S. Asta, D. Karapetyan, A. Kheiri, E. Özcan, and A. J. Parkes, "Combining Monte-Carlo and hyper-heuristic methods for the multi-mode resource-constrained multi-project scheduling problem," *Information Sciences*, vol. 373, pp. 476–498, 2016.

[46] R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems," *Computer-Aided Design*, vol. 43, no. 3, pp. 303–315, 2011.

[47] W. Shao, D. Pi, and Z. Shao, "An extended teaching-learning based optimization algorithm for solving no-wait flow shop scheduling problem," *Applied Soft Computing*, vol. 61, pp. 193–210, 2017.

[48] Y. Xu, L. Wang, S. Y. Wang, and M. Liu, "An effective teaching–learning-based optimization algorithm for the flexible job-shop scheduling problem with fuzzy processing time," *Neurocomputing*, vol. 148, pp. 260–268, 2015.

[49] P. Kumar Roy, A. Sur, and D. K. Pradhan, "Optimal short-term hydro-thermal scheduling using quasi-oppositional teaching learning based optimization," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 10, pp. 2516–2524, 2013.

[50] W. Q. Ma, C. Y. Zhang, Q. H. Tang, and Y. Jia, "Steelmaking and continuous casting scheduling based on hybrid teaching-learning-based optimization algorithm," *Computer Integrated Manufacturing Systems*, vol. 21, no. 5, pp. 1271–1278, 2015.

[51] H. Zheng and L. Wang, "An effective teaching–learning-based optimisation algorithm for RCPSP with ordinal interval numbers," *International Journal of Production Research*, vol. 53, no. 6, pp. 1777–1790, 2015.

[52] H.-y. Zheng, L. Wang, and S.-y. Wang, "A co-evolutionary teaching-learning-based optimization algorithm for stochastic RCPSP," in *2014 IEEE Congress on Evolutionary Computation (CEC)*, pp. 587–594, Beijing, China, 2014.

[53] H. Y. Zheng, L. Wang, and X. L. Zheng, "Teaching–learning-based optimization algorithm for multi-skill resource constrained project scheduling problem," *Soft Computing*, vol. 21, no. 6, pp. 1537–1548, 2017.

[54] Z. Zhang, S. Lin, R. Dong, and Q. Zhu, "Designing a human-computer cooperation decision planning system for aircraft carrier deck scheduling," in *AIAA Infotech @ Aerospace*, pp. 1–9, Kissimmee, FL, USA, 2015.

[55] L. S. Cardona-Meza and G. Olivar-Tost, "Modeling and simulation of project management through the PMBOK® standard using complex networks," *Complexity*, vol. 2017, Article ID 4791635, 12 pages, 2017.

[56] D. Debels and M. Vanhoucke, "A decomposition-based genetic algorithm for the resource-constrained project-scheduling problem," *Operations Research*, vol. 55, no. 3, pp. 457–469, 2007.

[57] R. Kolisch, "Serial and parallel resource-constrained project scheduling methods revisited: theory and computation," *European Journal of Operational Research*, vol. 90, no. 2, pp. 320–333, 1996.

[58] T. A. M. Toffolo, H. G. Santos, M. A. M. Carvalho, and J. A. Soares, "An integer programming approach to the multimode resource-constrained multiproject scheduling problem," *Journal of Scheduling*, vol. 19, no. 3, pp. 295–307, 2016.

[59] V. Valls, F. BallestíN, and S. Quintanilla, "Justification and RCPSP: a technique that pays," *European Journal of Operational Research*, vol. 165, no. 2, pp. 375–386, 2005.

[60] O. Lambrechts, E. Demeulemeester, and W. Herroelen, "A tabu search procedure for developing robust predictive project schedules," *International Journal of Production Economics*, vol. 111, no. 2, pp. 493–508, 2008.

[61] O. I. Tukel, W. O. Rom, and S. D. Eksioglu, "An investigation of buffer sizing techniques in critical chain scheduling," *European Journal of Operational Research*, vol. 172, no. 2, pp. 401–416, 2006.

[62] W. Tian and E. Demeulemeester, "On the interaction between railway scheduling and resource flow networks," *SSRN Electronic Journal*, vol. 25, no. 1-2, pp. 1–25, 2010.

[63] B. Ashtiani, R. Leus, and M. B. Aryanezhad, "New competitive results for the stochastic resource-constrained project scheduling problem: exploring the benefits of pre-processing," *Journal of Scheduling*, vol. 14, no. 2, pp. 157–171, 2011.

[64] S. Das and P. N. Suganthan, "Differential evolution: a survey of the state-of-the-art," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, 2011.

[65] S. M. Islam, S. Das, S. Ghosh, S. Roy, and P. N. Suganthan, "An adaptive differential evolution algorithm with novel mutation and crossover strategies for global numerical optimization," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 2, pp. 482–500, 2012.

[66] V. Valls, F. Ballestín, and S. Quintanilla, "A hybrid genetic algorithm for the resource-constrained project scheduling problem," *European Journal of Operational Research*, vol. 185, no. 2, pp. 495–508, 2008.

[67] M. J. Geiger, "A multi-threaded local search algorithm and computer implementation for the multi-mode, resource-constrained multi-project scheduling problem," *European Journal of Operational Research*, vol. 256, no. 3, pp. 729–741, 2017.

[68] Y. Niu, Z. Yang, P. Chen, and J. Xiao, "A hybrid tabu search algorithm for a real-world open vehicle routing problem involving fuel consumption constraints," *Complexity*, vol. 2018, Article ID 5754908, 12 pages, 2018.

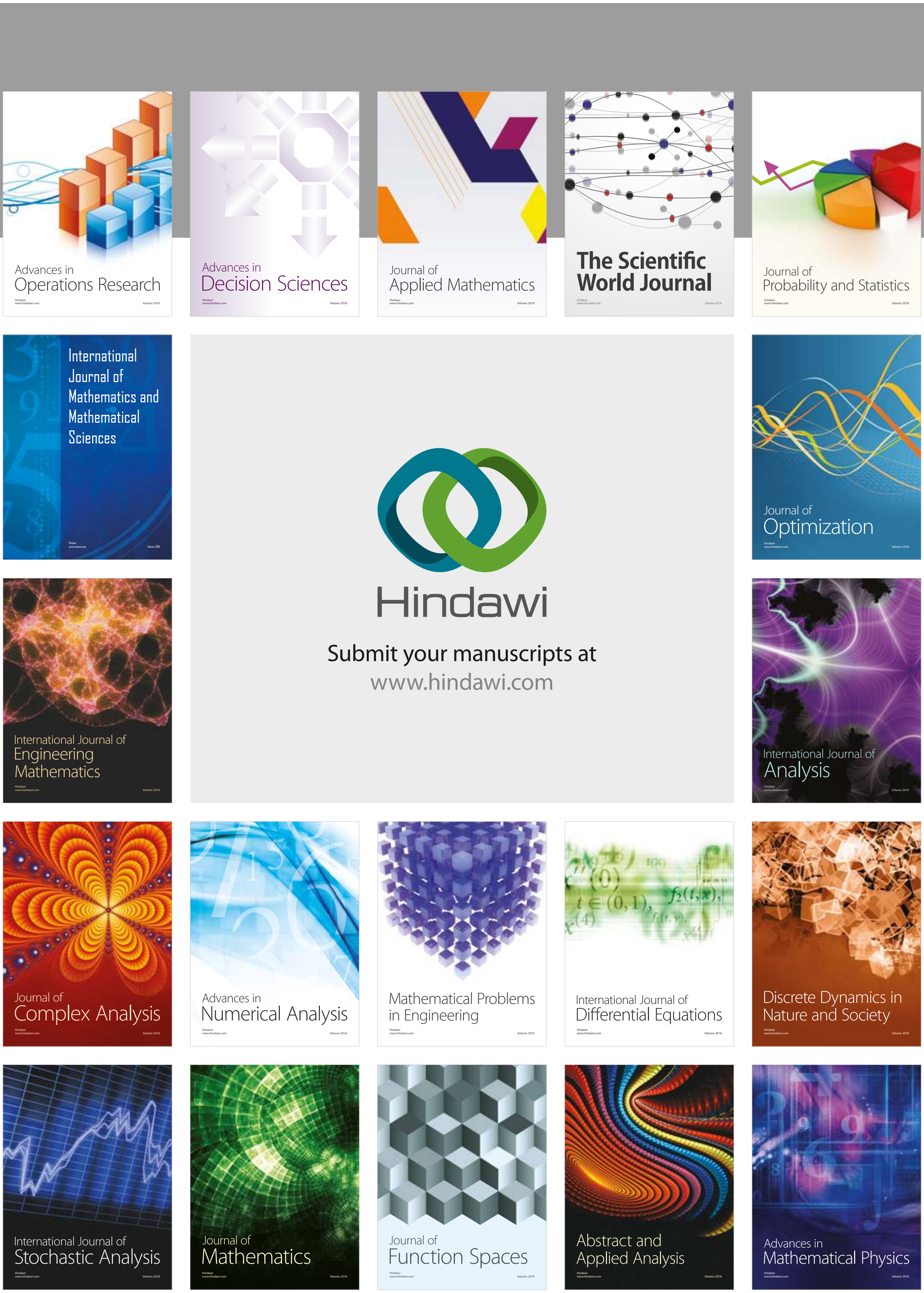
[69] T. Wauters, K. Verbeeck, P. de Causmaecker, and G. vanden Berghe, "A learning-based optimization approach to multi-project scheduling," *Journal of Scheduling*, vol. 18, no. 1, pp. 61–74, 2015.

[70] T. Wauters, K. Verbeeck, G. V. Berghe, and P. de Causmaecker, "Learning agents for the multi-mode project scheduling problem," *Journal of the Operational Research Society*, vol. 62, no. 2, pp. 281–290, 2011.

[71] Q. Jia and Y. Seo, "An improved particle swarm optimization for the resource-constrained project scheduling problem," *The International Journal of Advanced Manufacturing Technology*, vol. 67, no. 9–12, pp. 2627–2638, 2013.

[72] F. Ballestin and R. Leus, "Resource-constrained project scheduling for timely project completion with stochastic activity durations," *Production and Operations Management*, vol. 18, no. 4, pp. 459–474, 2009.

[73] C. Fang, R. Kolisch, L. Wang, and C. Mu, "An estimation of distribution algorithm and new computational results for the stochastic resource-constrained project scheduling problem," *Flexible Services and Manufacturing Journal*, vol. 27, no. 4, pp. 585–605, 2015.


Advances in Operations Research
Advances in Decision Sciences
Journal of Applied Mathematics
The Scientific World Journal
Journal of Probability and Statistics
International Journal of Mathematics and Mathematical Sciences
Journal of Optimization
International Journal of Engineering Mathematics
International Journal of Analysis
Journal of Complex Analysis
Advances in Numerical Analysis
Mathematical Problems in Engineering
International Journal of Differential Equations
Discrete Dynamics in Nature and Society
International Journal of Stochastic Analysis
Journal of Mathematics
Journal of Function Spaces
Abstract and Applied Analysis
Advances in Mathematical Physics