IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 7, 2018, accepted October 22, 2018, date of publication November 5, 2018, date of current version December 7, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2879503

A Robust Scheduling Optimization Method for Flight Deck Operations of Aircraft Carrier With Ternary Interval Durations

XI-CHAO SU[1], WEI HAN[1], YU WU[2], YONG ZHANG[1], AND JING-YU SONG[3]
[1]Naval Aviation University, Yantai 264001, China
[2]College of Aerospace Engineering, Chongqing University, Chongqing 400044, China
[3]System Engineering Research Institute, China State Shipbuilding Corporation, Beijing 10094, China

Corresponding author: Yu Wu (cquwuyu@cqu.edu.cn)

This work was supported by the Fundamental Research Funds for the Central University under Grant 106112016CDJRC000107.

ABSTRACT The sortie and recovery operations of the carrier-based aircraft fleet are carried out in a cyclic mode, and there are uncertainties during the flight deck operations, which is the main factor that influences the sustainability of cyclic operations. The robust scheduling problem for flight deck operations (RSPFDO) with uncertain duration is studied in this paper. First, the robust scheduling model of flight deck operations with ternary interval durations is established. In this model, the precedence and resource constraints including personnel, equipment, workstation space, and supply resource are taken into account, and the uncertain duration is modeled as a ternary interval number. Second, to make a proactive plan, serial schedule generation scheme is used to generate baseline schedules, and a robust personnel allocation scheme is designed. In terms of executing schedules, a pre-constrained scheduling policy is proposed to evaluate the ternary interval of makespan. Third, a double-population and self-adaptive differential evolution (DSDE) algorithm is presented to optimize the robust plan, which is embedded in a population-based double justification scheme, self-adaptive selection of mutation and crossover factors, and a chaotic catastrophe operator based on the scouts mode of artificial bee colony algorithm. The validity of the established model and the superiority of the DSDE algorithm are verified by simulations with mission cases and algorithm comparison. The results demonstrate that the proposed robust scheduling method can effectively improve the robustness of the flight deck operations.

INDEX TERMS Flight deck operations, ternary interval number, robust scheduling, differential evolution algorithm.

I. INTRODUCTION

The sortie generation capacity is a key indicator for the combat effectiveness of the carrier-based aircraft fleet. Different from ground operations of aircraft in the airport, sortie and recovery operations of carrier-based aircraft are executed in a cyclic mode, which means a fleet of aircraft will go through the pre-flight preparation, take-off process, return and landing on the flight deck within a certain cycle, and there is time overlapping between cycles of different fleets. To avoid the disorder and ensure the sustainability of cyclic operations, the duration of each cycle is regulated, which includes the modes of 1+00 (1 h 0 min), 1+15 (1 h 15 min), 1+30 (1 h 30 min) and 1+45 (1 h 45 min) [1]. During the cycle, operations on flight deck are time-critical, resource-constrained, and complicated in the execution procedure. What is more, uncertainties including the operation time, random failures of the aircraft and support equipment and real-time alteration of tasks may lead the delay and disturb the operating tempos. Therefore, a reasonable plan which contains the baseline schedule and allocation of resource in complex environments can improve the robustness of operation schedule, thus maintaining the sustainable sortie generation capacity [2].

Researches on uncertain scheduling for aircraft operations have been a focus for recent years, which can be categorized into two groups: reactive scheduling and stochastic scheduling. In terms of reactive scheduling, Ryan *et al.* [3] designed the deck operations course of action planner. The planner is embedded with the conventional integer linear program-based planning algorithm [4] to support dynamic model-based replanning, and simulation results show that the

human experience outperforms the automatic planner in most cases. Wu *et al.* [5] proposed a modeling and sequencing approach for landing a team of aircraft in which the possibility of failed-to-land aircraft is taken into consideration, and a dynamic sequencing method using ant colony algorithm is presented to solve the replanning problem. In reactive scheduling, the response time is an important index to measure the performance, and the agent-based model serves as an efficient framework for dynamic scheduling on flight deck in both manned [6] and manned-unmanned environments [7]. Feng *et al.* [8] extended the multi-agent system to the aircraft maintenance operations, and an integration optimization method for personnel configuration and scheduling was proposed. Besides, another efficient method called hierarchical task network (HTN) [9] planning was applied in this field, and the method is based on expert system in nature. As for the stochastic scheduling, the main advantage is that it can generate a best series of scheduling policies by offline simulation, and the scheduling policies can be applied to online scheduling without additional computation. Dastidar and Frazzoli [10] established a queueing network based approach for flight deck operations, which transformed the scheduling problem into a priority optimization of the network nodes. Feng *et al.* [11] built an improved direct graph to model the operation process and constraints, and designed an improved ant colony optimization algorithm to optimize the scheduling strategy. To cope with the multi-stage decision course under uncertainty on flight deck, the Markov decision process (MDP) [12] provides a tool for modeling. For example, Li *et al.* [13] cast the scheduling of flight deck operations in a MDP framework, and transformed the scheduling experience of experts into an intelligent scheduling strategy based on the reverse reinforcement learning.

In summary, all of these existing scheduling methods focused on the reactive scheduling and stochastic scheduling, but the plan of temporal schedule and resource allocation is ignored, which is needed in advance of execution. The plan of temporal schedule and resource allocation is crucial in terms of reducing disturbance from uncertainties, and this problem can be solved by robust scheduling methods. Note that uncertainties in the course of flight deck operations are comprised of operation-relevant and resource-relevant disturbances, and the resources disturbances such as equipment failure can be transformed approximately into the uncertainty of operation time according to the frequency of equipment failures and repairing time. Therefore, uncertain durations of operations can be seen as the main interference factor [14]. In this paper, the robust scheduling problem for flight deck operations (RSPFDO) with uncertainties in duration is studied.

In terms of problem modeling, The scheduling problem for flight deck operations was established as an extended flexible job shop scheduling model in [15], but more research work took it as a resource-constrained (multi-)project scheduling problem (RC(M)PSP) [2], [16], [17]. In the project scheduling with uncertainty, there are mainly three ways, i.e., random number [18]–[20], fuzzy number [21], [22] and interval number [23], to describe the activity duration uncertainties. The precise probability distribution function of operation duration is required when using the random number to denote the uncertainty. Similarly, the fuzzy membership function [24], [25] of operation duration is needed when using the fuzzy number. In the above two cases, massive sample data is required, but the corresponding data is often hard to be obtained in flight deck operations due to the lack of effective data collection tools [4]. Compared to the methods using the random number and the fuzzy number, methods based on interval number only require the upper and lower bound of operation duration, which are accessible in flight deck operations. Besides, the interval number is applied widely in the research of scheduling. Lei applied the interval number theory to production scheduling, and a population-based neighborhood search strategy [26] and an effective genetic algorithm were designed [27]. To solve the flow shop scheduling problem with interval processing time, Han *et al.* [28] formulated it as a multi-objective optimization problem by turning each interval objective into a real-valued. Ng *et al.* [29] proposed a graph-theoretic-based exact solution algorithm for interval scheduling of unrelated parallel machines. Xie and Chen [30] introduced another interval grey number to describe the processing time in flexible job shop scheduling problem, and new model and solving method based on the grey system theory are proposed. Drwal [31] considered the single machine scheduling problems with the interval due-date uncertainty, and proposed a robust scheduling method based on the concept of maximum regret.

As for the RC(M)PSP with uncertainty, the interval scheduling is seldom studied. Li *et al.* [23] proposed a proactive project scheduling model with interval activity time and cost, however, the constraints about the renewable resources are neglected in the model. Compared with the classical binary interval number, the ternary interval number [32] contains a most likely value of the uncertainty degree, which can extend the information needed for scheduling decisions. Different from the fuzzy scheduling with triangular fuzzy number which are associated with complex fuzzy operation for scheduling activities, and produces fuzzy schedule that are not intuitive to be the plan for flight deck operations, interval scheduling with ternary interval number adopts simple operation of interval number for scheduling activities. What is more, the most likely durations of activities can be used for generating basic schedule, which make it a better way to solve the RSPFDO with ternary interval durations.

This paper focuses on the robust scheduling problem for flight deck operations with ternary interval durations. Based on the description of resource and precedence constraints of the flight deck operations, a robust scheduling model for RSPFDO is established, with the robust objectives of maximizing the agreement index of makespan and deck operation cycle and minimizing the weighted sum of most likely makespan and variation range of makespan. To address the established model, a pre-constrained scheduling policy and an evaluation method for ternary interval of makespan

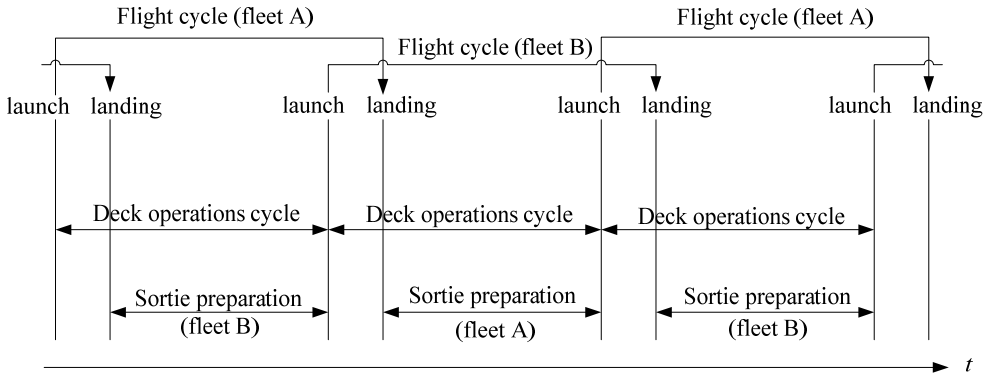

FIGURE 1. Sustainable cyclic operations of carrier-based aircraft.

are presented, and then a double-population self-adaptive differential evolution (DSDE) algorithm for optimizing the proactive schedule is proposed, which is incorporated with a population-based double justification, self-adaptive mutation and crossover factors and a chaotic catastrophe operator inspired by the scouts mode of artificial bee colony algorithm (ABC). Simulation experiments of mission cases with different scales are conducted to verify the feasibility of the model and efficiency of the algorithm.

The remainder of this paper is organized as follows. Section II describes the RSPFDO with ternary interval durations, and Section III presents a mathematical programming model for the problem; in Section IV, the robust scheme formulation and scheduling strategy are proposed; Section V describes the proposed DSDE, and Section VI reports and discusses the results of computational experiments; finally, this paper is concluded by Section VII.

II. PROBLEM DESCRIPTION

A. DESCRIPTION OF FLIGHT DECK OPERATIONS

As mentioned in Section 1, the carrier-based aircraft fleet executes missions in a cyclic way, as shown in Figure 1. During a complete operation cycle, each aircraft fleet should go through the processes of sortie preparation, launch, flight mission and landing. As the area of the flight deck is limited, only one fleet can make sortie preparation on the flight deck at the same period to avoid the jam. For example, when fleet A is on missions in the air and fleet B is doing preflight preparation on the flight deck, fleet B should take off before the recovery of fleet A to ensure the sustainable combat forces. Based on this cyclic operation mode, the deck operation cycle is set strictly in case of the time overlapping of two fleets happens. The same is true for the duration of sortie preparation. Therefore, the key to solving the problem is to schedule the flight deck operations with uncertainties to meet the time limit.

During the sortie preparation, an aircraft begins operations after being parked on a specific spot. Once chocked and chained, 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. The execution of the flight deck operations refers to the complicated scheduling of aircraft, support equipment and personnel. The problem consists of a set of aircraft $i \in I$, $I = \{1, 2, \cdots, n\}$, which is composed of a set of operations, denoted as J_i, a partition from all operations set $J = \{(i, j) \,|i \in I, j \in J_i\}$. Each aircraft $i \in I$ has a release time $\widetilde{Ex}_i$, denoted as a ternary interval, representing the time when aircraft i is parked on the spot p_i, and chocked and chained, which is the earliest starting time for the operations in J_i.

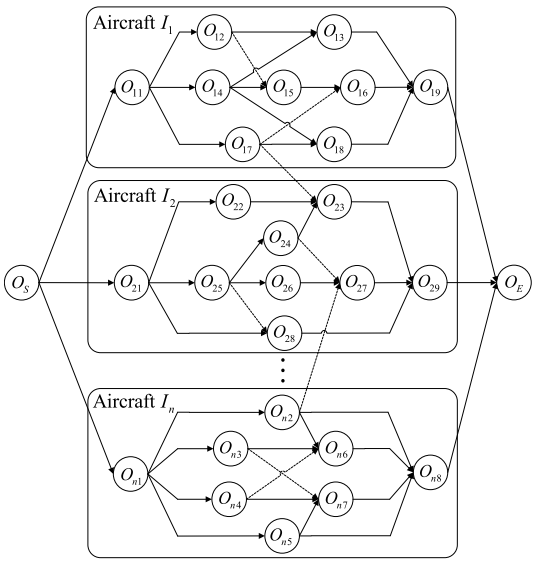

FIGURE 2. OON network model of sortie preparation process.

The operations are interrelated by two types of constraints: the precedence constraints and the resource constraints. As for the former, The precedence relationships are usually modeled as operation on node (ONN) network, let O_{ij} be the operation j of aircraft i, which cannot be started until all the operations in the immediate predecessor operation set P_{ij} are completed. An example is shown in Figure 2, where the solid arcs represent original precedence relations between operations, while the dashed arcs represent the extra

precedence relations imposed by priority assignment of personnel, equipments and workstation space after the plan is determined. The resource constraints are complex since four different types of the resources are taken into account: personnel, support equipment, workstation space and supply resource, and the set of the four types of resources are denoted as Kp, Ke, Ks and Kw respectively.

1) **Personnel constraints.** Lp_k denotes the set of personnel with trade k ($k \in Kp$), $Lp_k = \{1, 2, \cdots, |Lp_k|\}$, rp_{ijk} denotes the number of personnel with trade k ($k \in Kp$) required for performing operation O_{ij}. It is obvious that the number of personnel is limited for executing operations at every moment. Besides, personnel are usually divided into group(s) according to the specific personnel organization mode, and each group is assigned to a certain set of aircraft, let Rp_{kl} be the set of aircraft that the lth personnel with trade k ($k \in Kp$) can be allocated to.
2) **Support equipment constraints.** Le_k represents the set of support equipments of the type k ($k \in Ke$), $Le_k = \{1, 2, \cdots, |Le_k|\}$, re_{ijk} represents the number of equipment of the type k ($k \in Ke$) required for performing operation O_{ij}. Support equipment constraints refer to not only the limitation of available quantity, but also the maximum range of their pipelines that stationary equipments can support aircraft. Let λ_{kl}^p be the coverage relation parameter of support equipment, and $\lambda_{kl}^p = 1$ if the lth support equipment of the type k ($k \in Ke$) can reach the pth parking spot, otherwise $\lambda_{kl}^p = 0$.
3) **Workstation space constraints.** This kind of constraint is considered only for those workstations with limited space for operations. Namely, when more than one operations are executed in the same workstation (e.g., the cockpit), they are scheduled with serial sequence due to the limit space. Set $rs_{ijk} = 1$ if operation O_{ij} is performed in workstation space of the type k ($k \in Ks$), otherwise $rs_{ijk} = 0$.
4) **Supply resource constraints.** The constraints mean the capacity of supplying aircraft concurrently. Lw_k denotes the maximum number of aircraft that the supply resource of type k ($k \in Kw$) can support at the same time, and set $rw_{ijk} = 1$ if the supply resource of type k ($k \in Kw$) is required for operation O_{ij}, otherwise $rw_{ijk} = 0$.

Besides, each operation O_{ij} has a duration $\tilde{d}_{ij}$, which is a ternary interval value as well. The objective of the problem is to make a proactive baseline schedule and allocate the relevant personnel and equipments for every operation, which makes the ternary interval of makespan of the multi-aircraft flight deck operations $\tilde{C}_{\max}$ robust to the due makespan $C_{\max}^d$.

B. DESCRIPTION OF TERNARY INTERVAL NUMBER

As an effective method to solve the decision-making problem with multi-attribute uncertainties, interval number is widely applied in many areas. As for binary interval number, only the lower and upper bound is considered, and the values between the bounds are taken equally. In this way. larger errors may be obtained when evaluating the overall result because the calculation result will enlarge the value range of interval number. Ternary interval number improves the reliability of decision-making by emphasizing the largest number of possibility values. Relevant definitions needed in this paper are listed as follows:

Definition 1: Suppose that $a, a^*, \bar{a}$ are all non-negative real numbers, $a, a^*, \bar{a} \in R$, and $a \le a^* \le \bar{a}$, then $\tilde{a} = [a, a^*, \bar{a}]$ is a ternary interval number, where $a, \bar{a}$ are the lower bound and upper bound of interval number respectively, and a^* is the value corresponding to the value of $\tilde{a}$ with the maximum probability, which is also called as the center of gravity. Specifically, when $a = a^* = \bar{a}$, the ternary interval number $\tilde{a}$ can be regarded as a real number.

Definition 2: For ternary interval $\tilde{a} = [a, a^*, \bar{a}]$ and $\tilde{b} = [b, b^*, \bar{b}]$, if $a \ge b$, $a^* \ge b^*$ and $\bar{a} \ge \bar{b}$, then $\tilde{a}$ is completely greater than $\tilde{b}$, denoted as. $\tilde{a} \underset{C}{\geq} \tilde{b}$.

Two specific operators used in this paper are listed as follows:

1)Addition operation:

$$\tilde{a} + \tilde{b} = [a + b, a^* + b^*, \bar{a} + \bar{b}]$$

2)Maximum operation:

$$\tilde{a} \vee \tilde{b} = [\max(a, b), \max(a^*, b^*), \max(\bar{a}, \bar{b})]$$

In reference to fuzzy scheduling [33], agreement index AI is defined to measure the intersection of a ternary interval number and some fixed intervals.

Definition 3: Suppose that $\tilde{a} = [a, a^*, \bar{a}]$ is a ternary interval number, with $a\bar{a}$ constructed as a hemline, and a^* is the foot point on the hemline. The area of triangle $\Delta(\tilde{a})$ is a constant number 1, and $b = [b_1, b_2]$ is a binary interval, then the agreement index is defined as $AI(\tilde{a}, b) = area(\tilde{a} \cap b)$, as shown in Figure 3.

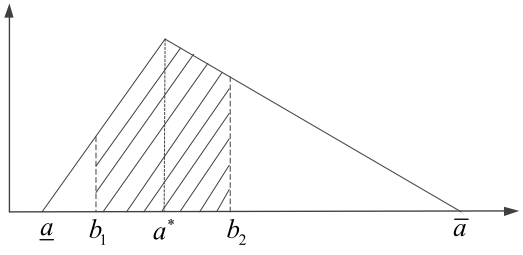

FIGURE 3. Agreement index of ternary interval number and binary interval number.

III. MATHEMATICAL FORMULATION

A. PROBLEM ASSUMPTIONS

The assumptions and conditions of RSPFDO are summarized as follows:

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

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

3) 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.

4) The distance between either two parking spots is large enough that there are no overlapping of workstation space between adjacent aircraft, and only the workstation space within a single aircraft is considered.

B. NOTATION

Apart from the notations given in Section 2.1, some other parameters are used as follows.

A_t set of operations which are active in period t

A_{it} set of operations of aircraft $i \in I$ which are active in period t

$\tilde{T}$ due interval of flight deck operations, and $\tilde{T} = [0, C^d_{\max}]$.

The following decision variables are used:

$\tilde{S}_{ij}$ ternary interval of start time of each operation O_{ij}

$$Xp_{ijkl} = \begin{cases} 1 & \text{if operation } O_{ij} \text{ is allocated to personnel} \\ & l\ (l \in Lp_k)\ \text{with trade } k\ (k \in Kp) \\ 0 & \text{otherwise} \end{cases}$$

$$Xe_{ijkl} = \begin{cases} 1 & \text{if operation } O_{ij} \text{ is allocated to equipment} \\ & l\ (l \in Le_k)\ \text{of the type } k\ (k \in Ke) \\ 0 & \text{otherwise} \end{cases}$$

$$Y_{ijeg} = \begin{cases} 1 & \text{if operation } O_{ij} \text{ precede operation } O_{eg} \\ & \text{when allocated to the same resources} \\ 0 & \text{otherwise} \end{cases}$$

C. MATHEMATICAL EXPRESSION

Under the uncertain operation durations, robustness of makespan can be divided into two levels. Firstly, the probability of completing the flight deck operations within the due interval is represented by the agreement index of the makespan and the due interval of flight deck operations; secondly, a comprehensive robust criterion is taken into account, and the definition of lexicographical robustness objectives is given as follows.

$$\min f = \left\{ G\left(\tilde{C}_{\max}\right) | \max AI\left(\tilde{C}_{\max}, \tilde{T}\right) \right\} \tag{1}$$

Where the ternary interval number of makespan is denoted by $\tilde{C}_{\max} = \left[\underline{C_{\max}}, C^*_{\max}, \overline{C_{\max}}\right]$. $AI\left(\tilde{C}_{\max}, \tilde{T}\right) = area\left(\tilde{C}_{\max} \cap \tilde{T}\right)$, where $AI\left(\tilde{C}_{\max}, \tilde{T}\right)$ is the agreement index of $\tilde{C}_{\max}$ and $\tilde{T}$. In Equation (1), the aim of the first level is to achieve a maximum agreement index of $\tilde{C}_{\max}$ and $\tilde{T}$, which means that the makespan of the aircraft fleet is as close to the due makespan as possible. The comprehensive robust criterion $G\left(\tilde{C}_{\max}\right)$ of expectation and variability of $\tilde{C}_{\max}$ is the goal in the second level.

$$G\left(\tilde{C}_{\max}\right) = C^*_{\max} + \alpha\left(\overline{C_{\max}} - C^*_{\max}\right) + \beta \underline{C_{\max}} \tag{2}$$

The first item in the objective is the gravity of $\tilde{C}_{\max}$, which represents the expectation of $\tilde{C}_{\max}$, and the second item is the deviation between the upper bound of $\tilde{C}_{\max}$ and the gravity; the last item is to shorten the completion time as much as possible. α and β are weigh coefficients, which can be set based on the scheduling commander's actual demands.

The constraints are as follows.

$$\tilde{S}_{i1} \underset{C}{\geq} \widetilde{Ex}_i, \quad \forall i \in I \tag{3}$$

$$\tilde{S}_{ij} \underset{C}{\geq} \tilde{S}_{ih} + \tilde{d}_{ih}, \quad \forall (i,h) \in P_{ij},\ \forall (i,j) \in J \tag{4}$$

$$\tilde{S}_{ij} + \tilde{d}_{ij} \underset{C}{\leq} \tilde{S}_{eg} + BM \cdot (1 - Y_{ijeg}), \quad \forall (i,j), (e,g) \in J \tag{5}$$

$$\sum_{i \in I} \sum_{j \in A_t} rp_{ijk} \leq |Lp_k|, \quad \forall k \in Kp,\ \forall t > 0 \tag{6}$$

$$\sum_{i \in I} \sum_{j \in A_t} re_{ijk} \leq |Le_k|, \quad \forall k \in Ke,\ \forall t > 0 \tag{7}$$

$$\sum_{j \in A_{it}} rs_{ijk} \leq 1, \quad \forall i \in I,\ \forall k \in Ks,\ \forall t > 0 \tag{8}$$

$$\sum_{i \in I} \operatorname{sgn}\left(\sum_{j \in A_t} rw_{ijk}\right) \leq Lw_k, \quad \forall k \in Kw,\ \forall t > 0 \tag{9}$$

$$\sum_{i \in I - Rp_{kl}} \sum_{j \in J_i} Xp_{ijkl} = 0, \quad \forall k \in Kp,\ \forall l \in Lp_k \tag{10}$$

$$\sum_{(i,j) \in J} \sum_{k \in Ke} \sum_{l \in Le_k} Xe_{ijkl} \cdot \left(1 - \lambda^{p_i}_{kl}\right) = 0, \quad \forall (i,j) \in J \tag{11}$$

$$\sum_{l \in Lp_k} Xp_{ijkl} = rp_{ijk}, \quad \forall (i,j),\ \forall k \in Kp \tag{12}$$

$$\sum_{l \in Le_k} Xe_{ijkl} = re_{ijk}, \quad \forall (i,j),\ \forall k \in Ke \tag{13}$$

Constraints (3) indicate that operations of aircraft $i \in I$ must be started after the release time $\widetilde{Ex}_i$. Constraints (4) represent the precedence relations of start time for each operation. Constraints (5) ensure that if two operation O_{ij} and O_{eg} require the same resources, the operation with higher priority starts first, where BM is a large number. Constraints (6) limit the number of support equipment. Similarly, constraints (7) limit the number of personnel of all trades. Constraints (8) guarantee that only one operation can be executed in each workstation space at a time. Constraints (9) indicate the capacity of supply resource of the type k $(k \in Kw)$ that can support aircraft concurrently. Constraints (10) and (11) ensure that personnel and support equipments can only be allocated to those aircraft within the range they can reach. Constraints (12) and (13) together

enforce that the assigned number of personnel and support equipment is matched to the demand.

IV. ROBUST SCHEME FORMULATION AND SCHEDULING POLICY

The coordinated scheduling of multiple carrier aircraft and many resources under uncertain durations in the flight deck operation is a NP-hard optimization problem with a large scale, and it is a resource-constrained multi-project scheduling problem (RCMPSP) in essence. The scheduling of flight deck operations is composed of three major elements: baseline schedule, personnel and equipment allocation plan and scheduling policy under uncertain durations. Correspondingly, the procedure of scheduling can be divided into three parts. Firstly, a baseline schedule is generated according to the priority of operations. Secondly, related personnel and support equipments are assigned in accordance with the baseline schedule. Lastly, dynamic scheduling is executed on the basis of scheduling policy under uncertain durations. Under the pre-constrained scheduling policy in combination with the baseline schedule and resource allocation plan, the ternary interval number of start and finish time of each operation and makespan can be derived with the upper and lower bound of each operation duration.

A. BASELINE SCHEDULE GENERATION SCHEME

Schedule generation scheme (SGS) is widely used to generate baseline schedule in the traditional RCMPSP. To generate the baseline schedule, the gravity of ternary interval number of duration d_{ij}^{*} $(\forall (i,j) \in J)$ is taken as the baseline duration. Firstly, a serial schedule generation scheme (SSGS) is used to generate the baseline schedule $\left\{\left[S_{ij}^{\mathrm{b}}, E_{ij}^{\mathrm{b}}\right]\right\}$ $(\forall (i,j) \in J)$ and resource allocation plan on the basis of operation priority vector $\boldsymbol{x}$. In this method, only the number of personnel is taken into account, and the personnel allocation is not the focus. What make it different from the RCMPSP is that, the baseline schedule is generated along with a feasible equipment allocation plan since they are fixed to specific ranges respectively. The equipment is not the globally shareable resource, while the personnel can be replaced by the same trade in group, which can be allocated after the baseline schedule is generated. As to the equipment allocation, the rule called "the minimum total processing time remaining in covering area" (MTRCA) [16] is used at every decision point that the equipments are in demand. To be specific, let S_g be the set of scheduled operations at stage g, the set of eligible operations within the range of the first support equipment of the type k $(k \in Ke)$ is defined as $Je_{kl} = \left\{(i,j) \,\middle|\, (i,j) \in \bar{S}_g \,, re_{ijk} > 0, \lambda_{kl}^{p_i} = 1\right\}$, 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.

The schedule can be generated in the way of either forward (left-justified) scheduling or backward (right-justified) scheduling. Generally, the baseline schedule is a left-justified one [34]. To design the SSGS for the baseline schedule of flight deck operations, 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}

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

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

$\pi Ls_{ik}^{t} = 1$ if workstation space of the type k $(k \in Ks)$ for aircraft i in period t is occupied, $=0$ otherwise

ρLw_{k}^{t} 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

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

In the opposite way, the backward scheduling starts from the last operation and schedule operations according to a reverse direction of the original precedence relations as late as possible. The detailed procedure of backward scheduling of RCMPSP can be found in [34].

Under a given right-justified schedule, a baseline left-justified schedule can be generated based on the Algorithm 1, and the procedure is as follows Firstly, the priority of operations is set by the start times of the right-justified schedule, which indicate that the operation with earlier start time is scheduled to the left with higher priority. Secondly, the allocation plan of equipment remains unchanged since the equipment is not the globally sharable resource, as is explained before, and the left-justification without this precondition may output another different equipment allocation and degrade the makespan. Finally, given the above precondition as input, the left-justified schedule can be generated based on the Algorithm 1.

B. ROBUST PERSONNEL ALLOCATION SCHEME

The robust personnel allocation scheme we present is operation based, i.e., personnel allocation is executed for operations one by one in the order of start time sequence. A time vector Tb is obtained from the start time of baseline schedule placed in sort ascending. Suppose that operation O_{ij} requires personnel with trade k, i.e. $rp_{ijk} > 0$. Define that at time t $(t \in Tb)$, A_k^t $(k \in Kp)$ is the available personnel set with trade k. Define that $rcf_{ij} = LST_{ij} - EST_{ij}$ is the free slack of the operation O_{ij}, where LST_{ij} is the latest start time obtained from right-justification and EST_{ij} is the earliest start time obtained from left-justification [35]. The decrease

Algorithm 1 Algorithm of Serial Schedule Generation Scheme for Aircraft Deck Operations

Input: Priority of operations x_{ij}.

Output: Temporal schedule $\left\{\left[S^{\mathrm{b}}_{ij}, E^{\mathrm{b}}_{ij}\right]\right\}$, initial equipment allocation plan $\left\{Xe_{ijkl}\right\}$.

01: Initialize $\rho Lp^{t}_{ik}\pi Le^{t}_{ik}$, πLs^{t}_{ik}, ρLw^{t}_{k}, $S_g := \cup_{i\in I}(i,1)$, $S_{i1} := c$, $E_{i1} := Ex^{*}_{i}$.

02: **While** $\left|S_g\right| < |J|$

03: Calculate $D_g := \left\{(i,j)\,\middle|\,(i,j)\notin S_g,\ P_{ij}\subseteq S_g\right\}$.

04: $(i^*,j^*) := \min_{O_{ij}\in D_g}\left\{(i,j)\,\middle|\,x_{ij} = \inf_{O_{pq}\in D_g}\left(x_{pq}\right)\right\}$.

05: $ES_{i^*j^*} := \max\left\{E_{ij}\,\middle|\,(i,j)\in P_{i^*j^*}\right\}$, initialize $ERS_{i^*j^*} := ES_{i^*j^*}$.

06: **Repeat**

07: $SP_{i^*j^*} := \min\left\{t\,\middle|\,t\geq ERS_{i^*j^*}, rp_{i^*j^*k}\leq \rho Lp^{\tau}_{i^*k}, \tau = \left[t, t+d^{*}_{i^*j^*}\right), (\forall k\in Kp)\right\}$,

08: $SE_{i^*j^*} := \min\left\{t\,\middle|\,t\geq ERS_{i^*j^*}, re_{i^*j^*k'}\leq \left|\pi Le^{\tau}_{i^*k'}\right|, \tau = \left[t, t+d^{*}_{i^*j^*}\right), \left(\forall k'\in Ke\right)\right\}$,

09: $SS_{i^*j^*} := \min\left\{t\,\middle|\,t\geq ERS_{i^*j^*}, \pi Ls^{\tau}_{i^*k''}\cdot rs_{i^*j^*k''} = 0, \tau = \left[t, t+d^{*}_{i^*j^*}\right), \left(\forall k''\in Ks\right)\right\}$,

10: $SW_{i^*j^*} := \min\left\{t\,\middle|\,t\geq ERS_{i^*j^*}, rw_{i^*j^*k'''}\leq \rho Lw^{\tau}_{i^*k'''}, \tau = \left[t, t+d^{*}_{i^*j^*}\right), \left(\forall k'''\in Kw\right)\right\}$.

11: $ERS_{i^*j^*} := \max\left(SP_{i^*j^*}, SE_{i^*j^*}, SS_{i^*j^*}, SW_{i^*j^*}\right)$.

12: **Until** $SP_{i^*j^*} = SE_{i^*j^*} = SS_{i^*j^*} = SW_{i^*j^*}$.

13: $S_{i^*j^*} := ERS_{i^*j^*}$, $E_{i^*j^*} := ERS_{i^*j^*} + d^{*}_{i^*j^*}$.

14: **For** $\forall k\in Ke\wedge re_{i^*j^*k} > 0$

15: Select $l\in\pi Le^{t}_{i^*k}\left(t = S_{i^*j^*}\right)$ according to MTRCA rule,$Xe_{i^*j^*kl} := 1$.

16: **End For**

17: Update $\rho Lp^{t}_{ik}\pi Le^{t}_{ik'}$, $\pi Ls^{t}_{ik''}$, $\rho Lw^{t}_{k'''}$, $S_g := S_g\cup(i^*,j^*)$.

18: **End While**

of free slack indicates that the operation is more critical. When $rcf_{ij} = 0$, O_{ij} is on the critical chain. Suppose that $PI_{ij} = \left\{(e,g)\,\middle|\,(e,g)\in P_{ij}\wedge rp_{egk} > 0\right\}$ is the predecessor set of O_{ij} with demand of personnel with the same trade and $MI_{ij} = \left\{(e,g)\,\middle|\,E^{\mathrm{b}}_{eg}\leq S^{\mathrm{b}}_{ij}\wedge rp_{egk} > 0, (e,g)\notin P_{ij}\right\}$ is the prepositive set of O_{ij} with demand of personnel with the same trade. The robust allocation scheme based on time recursion is achieved in the following process.

Step 1: Sort the operations by S_{ij} in ascending order (tie-break: increasing RCF_{ij}) and obtain the sequence list of operations for allocation LO. Assign personnel for each operation according to the list order. $n = 1$.

Step 2: Select the nth operation $O_{ij} = LO(n)$, and get the start time $t = S_{ij}$. For the required trade k ($k\in Kp$), search for the latest operation finished by currently available personnel $\forall l\in A^{t}_{k}\cap\left\{l'\,|i\in Rp_{kl'}\right\}$ at time t, e.g.O_{eg}. If $O_{eg}\in PI_{ij}$, then personnel l is assigned to the operation O_{ij}, thus preventing adding extra resource flows to increase the risk of delay. If the required number is satisfied, turn to Step 4, otherwise, turn to Step 3.

Step 3: If the latest operation is finished by currently available personnel $\forall l\in A^{t}_{k}\cap\left\{l'\,|i\in Rp_{kl'}\right\}$ at time t, e.g. $O_{eg}\in MI_{ij}$,define the float time of person l as $PF_l = S^{\mathrm{b}}_{ij} - E^{\mathrm{b}}_{eg}$, i.e. the float time between operation O_{eg} and operation O_{ij}. Choose the available person with the longest float time $l' = \arg\max PF_l$ and assign him to operation O_{ij}, thus the effects of uncertainty can be eliminated as much as possible.

Step 4: If $n = |J|$, output the personnel allocation plan, otherwise $n = n+1$, and turn to Step 2.

C. PRE-CONSTRAINED SCHEDULING POLICY

Solving resource-constrained project scheduling with SSGS is usually based on determinate activity duration. However, under the interval activity duration, it is hard to determine the time interval when operation occupies the resources. In project scheduling under uncertainty, scheduling policy [19] is applied to turn the baseline schedule into executed schedule.

Definition 4: Suppose there is a scenario $\boldsymbol{d} = (d_{11}, d_{12}, \cdots, d_{ij}, \cdots, d_{n|J_n|})$, $d_{ij}\in\tilde{d}_{ij}$ under the given interval duration. With scheduling policy Π and operation priority Pri, define the executed schedule under the scenario $\boldsymbol{d}$ as a mapping $(\boldsymbol{S}, C_{\max}) = \boldsymbol{\Pi}(\boldsymbol{d}, Pri)$. The executed start time for each operation is $\boldsymbol{S} = \Pi^{1}(\boldsymbol{d}, Pri)$ and the executed finish time of multi-aircraft is $C_{\max} = \Pi^{2}(\boldsymbol{d}, Pri)$.

Currently, the common scheduling strategy contains activity strategy C^{AB} and resource strategy C^{RB}, where C^{AB} is a direct extension of SSGS, which schedules activities according to the order in a given activity list AL. For any activity i and activity j, when $i\prec_{AL} j$, there is $\Pi^{1}(\boldsymbol{d}, \mathrm{AL})_i\leq\Pi^{1}(\boldsymbol{d}, \mathrm{AL})_j$,which is also called the side constraint. C^{RB} is a direct extension of parallel schedule generation scheme (PSGS). In the scheduling of flight deck operations, the priority of stochastic scheduling strategy is based on the baseline schedule. Since the baseline schedule is generated by SSGS, direct application of C^{RB} will turns the active baseline schedule into a non-delay one, causing the degradation of the performance of solutions. C^{AB} is restricted to the side constraint of the AL, in which operation without constraints must delay until the operation with higher priority starts first.

Taken a comprehensive consideration, a pre-constrained policy (C^{PC}) for the stochastic scheduling of flight deck operations is designed. The policy is executed in the way of parallel scheduling to ensure that operations without constraints are scheduled as soon as possible. Before the execution, constraints of personnel, equipment, workstation space and supply resource are transformed into the precedence relationships in advance, which protects the active attribute of the baseline schedule. Constraints transformation mainly contains three parts. Firstly, executed sequence of operations for each personnel and equipment in the baseline scheduling BS is transformed into the finish-start precedence relations. That is, if operation O_{ij} is prior to O_{eg}, both of them is assigned to the same personnel or equipment, then $O_{ij} \prec O_{eg}$ are added to the initial precedence constraints. Secondly, the sequence of operations that occupy the same workstation space are converted into the finish-start precedence relations as the way of personnel and equipment. Therefore, the resource flows of personnel, equipment and workstation space is added to the original ONN and turns into precedence relations in ONN, thus generating an updated ONN' [20]. Thirdly, for operation sets which are related to the same type of supply resource, a start-start precedence relation is added to each pair of operations in the set. The start-start precedence relation state that, if the start times of operation O_{ij} and operation O_{eg} satisfies that $S^{\mathrm{b}}_{ij} \leq S^{\mathrm{b}}_{eg}$,then the practical start times of them satisfy the following relationship $\Pi^1(\boldsymbol{d}, \mathrm{BS})_{ij} \leq \Pi^1(\boldsymbol{d}, \mathrm{BS})_{eg}$.

Based on the pre-constrained policy, characters of scheduling mapping $\Pi_{PC}(\boldsymbol{d}, \mathrm{BS})$ are given as follow.

Definition 5: For a given baseline scheduling BS and an operation duration scenario $\boldsymbol{d}$, if the makespan obtained by the pre-constrained scheduling policy is the upper bound of the ternary interval number $\tilde{C}_{\max}$, i.e. $\bar{C}_{\max} = \Pi^2_{PC}(\boldsymbol{d}, \mathrm{BS})$, then $\boldsymbol{d}$ is the duration scenario that produces the upper bound of makespan, all of which compose the set $\bar{\boldsymbol{D}}$. In the same way, duration scenarios that produce the lower bound of makespan compose the set $\underline{\boldsymbol{D}}$.

Definition 6: Scenario with the upper bound of operation duration is represented as $\bar{\boldsymbol{d}}$, within which, the duration of each operation is the upper bound of the corresponding ternary interval number. Similarly, $\boldsymbol{d}^*$ and $\underline{\boldsymbol{d}}$ are scenarios with the gravity and lower bound of operation duration respectively.

Theorem 1: Under a given baseline schedule BS and pre-constrained scheduling policy, scenario with the upper bound and lower bound of operation duration belonging to the scenario set produces the upper bound and lower bound of makespan respectively, i.e. $\underline{\boldsymbol{d}} \in \underline{\boldsymbol{D}}$ and $\bar{\boldsymbol{d}} \in \bar{\boldsymbol{D}}$.

Proof: Absurdity is used for proving. Suppose that $\underline{\boldsymbol{d}} \notin \underline{\boldsymbol{D}}$, i.e. $\Pi^2_{PC}(\underline{\boldsymbol{d}}, \mathrm{BS}) > \underline{C_{\max}}$. For $\forall \boldsymbol{d} \in \underline{\boldsymbol{D}}$, there is always non-lower-bound operation set Λ, which means $d_{ij} \geq \underline{d}_{ij}, d_{ij} \in \boldsymbol{d}, (i,j) \in \Lambda$. Suppose that (i', j') is the earliest scheduled operation in Λ, i.e. $\Pi^1_{PC}(\underline{\boldsymbol{d}}, \mathrm{BS})_{i'j'} = \Pi^1_{PC}(\boldsymbol{d}, \mathrm{BS})_{i'j'}$. Based on the definition of C^{PC}, $\forall (e,g) \in H_{i'j'}$ where H_{ij} denotes the set of successors of operation O_{ij}, $\Pi^1_{PC}(\underline{\boldsymbol{d}}, \mathrm{BS})_{eg} \leq \Pi^1_{PC}(\boldsymbol{d}, \mathrm{BS})_{eg}$. Analogically, $\Pi^2_{PC}(\underline{\boldsymbol{d}}, \mathrm{BS}) \leq \Pi^2_{PC}(\boldsymbol{d}, \mathrm{BS}) = \underline{C_{\max}}$, which conflicts with $\Pi^2_{PC}(\underline{\boldsymbol{d}}, \mathrm{BS}) > \underline{C_{\max}}$. Hence $\underline{\boldsymbol{d}} \in \underline{\boldsymbol{D}}$. The same method can be used to prove $\bar{\boldsymbol{d}} \in \bar{\boldsymbol{D}}$.

In other words, Theorem 1 indicates that under a given baseline schedule BS and pre-constrained scheduling policy, scenario with the upper and lower bound of operation duration are mapped into the upper bound and lower bound of makespan $\tilde{C}_{\max}$, respectively.

Inference 1: Under a given baseline schedule BS and pre-constrained policy, for scenario with the upper and lower bound of operation duration $\bar{\boldsymbol{d}}$ and $\underline{\boldsymbol{d}}$, the two equations, i.e., $\Pi^1_{PC}(\underline{\boldsymbol{d}}, \mathrm{BS})_{ij} = S_{ij}$ and $\Pi^1_{PC}(\bar{\boldsymbol{d}}, \mathrm{BS})_{ij} = \bar{S}_{ij}$, are met for any operation.

Proof: For operation (i,j), construct an operation on node sub-network $ONS_{ij} \in ONN'$ and relevant sub-plan of baseline schedule $BS_{ij} \in BS$ based on its reachable predecessor operation set. Based on Theorem 1, $\Pi^2_{PC}(\underline{\boldsymbol{d}}, \mathrm{BS}_{ij}) = \underline{S}_{ij} + \underline{d}_{ij}$ and $\Pi^2_{PC}(\bar{\boldsymbol{d}}, \mathrm{BS}_{ij}) = \bar{S}_{ij} + \bar{d}_{ij}$, i.e. $\Pi^1_{PC}(\underline{\boldsymbol{d}}, \mathrm{BS}_{ij})_{ij} = S_{ij}$ and $\Pi^1_{PC}(\bar{\boldsymbol{d}}, \mathrm{BS}_{ij})_{ij} = \bar{S}_{ij}$. Since other operation $O_{eg} \in \overline{ONS_{ij}}$ outside the operation on node sub-network has no effect on (i,j), $\Pi^1_{PC}(\underline{\boldsymbol{d}}, \mathrm{BS})_{ij} = S_{ij}$ and $\Pi^1_{PC}(\bar{\boldsymbol{d}}, \mathrm{BS})_{ij} = \bar{S}_{ij}$ are likewise satisfied for all operations in baseline schedule.

Inference 2: Under a given baseline schedule BS and pre-constrained policy, for scenario with the gravity of operation duration $\boldsymbol{d}^*$, $\Pi^1_{PC}(\boldsymbol{d}^*, \mathrm{BS})_{ij} = S^*_{ij}$ and $S^*_{ij} = S^{\mathrm{b}}_{ij}$ are satisfied for any operation.

Proof Mathematical induction is used. For a dummy start operation $O_{i1}, \forall i \in I$, initialize the scheduling stage $g = 1$, and $\Pi^1_{PC}(\boldsymbol{d}^*, \mathrm{BS})_{i1} = Ex^*_i = S^*_{i1}$. In the baseline schedule which is generated based on SSGS, there is $S^{\mathrm{b}}_{i1} = Ex^*_i$. So the Inference is true at stage $g = 1$. Suppose that when $g = k, 1 \leq k \leq |J|$, the inference is also true. When $g = k+1$, according to the pre-constrained scheduling policy, suppose that under the scenario $\boldsymbol{d}^*$, the scheduling point is $t_g = \min\left\{S^*_{ij} + d^*_{ij} \middle| (i,j) \in A_{g-1}\right\}$, where A_{g-1} is the set of operations active in stage $g-1$ and $t_g = E^*_{ij} = E^{\mathrm{b}}_{ij}$, $(i,j) \in A_{g-1}$. Obviously, the set of eligible operations D_g satisfied with precedence and supply resources constraints corresponding to the operations which is started at time t_g in the baseline schedule. Consequently $\forall (e,f) \in D_g$, $\Pi^1_{PC}(\boldsymbol{d}^*, \mathrm{BS})_{ef} = S^{\mathrm{b}}_{ef}$. According to the definition of addition and maximum operation of ternary interval number, the gravity of the earliest start time of operation O_{ef} which is satisfied with precedence constraints is $EST^*_{ef} = \max\left\{S^*_{ij} + d^*_{ij} \middle| O_{ij} \in P'_{ij}\right\} = t_g$, where P'_{ij} is the precedent operation set of O_{ij} in ONN', and at time t_g,operations in P'_{ij} satisfy the constraints of supply resource. Then $S^*_{eg} = t_g = \Pi^1_{PC}(\boldsymbol{d}^*, \mathrm{BS})_{eg}$. Hence the inference is true in stage $g = k+1$. Above all, under scenario $\boldsymbol{d}^*$ with the pre-constrained policy, the execution procedure can follow the baseline schedule in any stage, and the start time of each operation in baseline schedule is the gravity of start time.

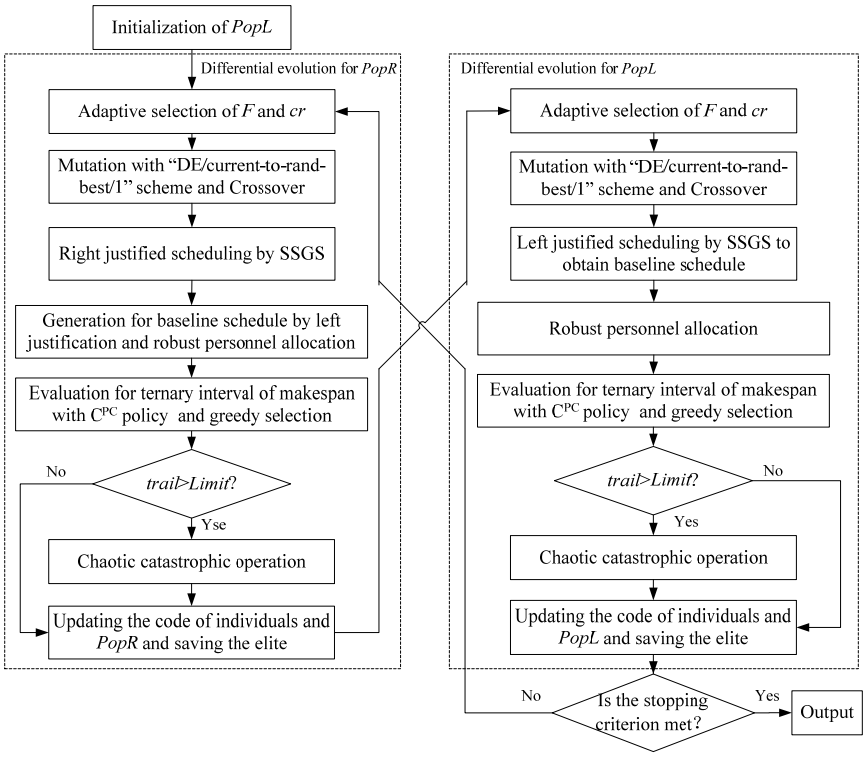

FIGURE 4. The flowchart of DSDE for the RSPFDO.

All in all, based on the baseline schedule generated with the gravity duration of operation in combination with pre-constrained scheduling policy, the ternary interval number of start time of each operation and makespan can be calculated according to scenarios with the upper bound $\underline{\boldsymbol{d}}$, lower bound $\bar{\boldsymbol{d}}$ and gravity of operation duration $\boldsymbol{d}^*$. In this way, the difficulty in searching for the ternary interval start time of each operation with complex precedence constraints and resource constraints is avoided, even though an evaluation for a robust schedule will take three times of generating schedules.

V. THE PROPOSED DSDE ALGORITHM

In Section 4, the function that maps a priority vector of operations to the proactive baseline schedule and resource allocation plan, and the ternary intervals of start and finish times of operations and makespan has been constructed, then the priority vector of operations can be optimized with the robustness objectives in Equation (1). Since the RSPFDO can be classified as a variant of RCPSP, a variety of optimization algorithms coping with the RCPSPs (variations of the RCPSP) can be referred to. Among these algorithms, differential evolution (DE) algorithm is a relatively effective meta-heuristic algorithm and has been applied in the classical RCPSP [36], multi-mode RCPSP [37] and RCMPSP [38] with outstanding performances, and the combinations with other algorithms or techniques, such as fuzzy clustering and chaotic technique in FCDE [36], and genetic algorithm in COA [39], often produce better results.

To cope with this problem, a double-population self-adaptive differential evolution (DSDE) algorithm is proposed. Inspired by the double justification which has been proved to be an efficient technique for reducing the makespan [34], a population-based double justification strategy is used. To be specific, two populations are maintained in the algorithm: the right-justified population (RJ-population) that generates right-justified baseline schedules, and the left-justified population (LJ-population) that generates left-justified baseline schedules. Besides, differential evolution algorithm (DE) is adopted to update the populations with self-adaptive mutation and crossover factors. What is more, to avoid falling into the local optimums, a chaotic catastrophe operator inspired by the scouts mode of artificial bee colony algorithm (ABC) is incorporated into the DSDE.

A. PROCEDURE OF THE PROPOSED DSDE

The flowchart of DSDE for solving the RSPFDO is shown in Figure 4.

Step 1: Generate an initial left-justified population PopL based on Logistic chaotic mapping.

Step 2: The RJ-population PopR is updated by feeding it with the relevant individuals taken from PopL and PopR, and new individuals are generated through a differential evolution operator. Each new individual is generated by the following sub-step:

Step 2.1: Adaptive scale factor F and adaptive crossover probability cr are generated based on the diversity of PopR.

Step 2.2: For any individual in PopR, apply the "DE/current-to-rand-best/1" strategy and binomial crossover operator to generate new individual.

Step 2.3: On the basis of SSGS, a new individual is decoded by right-justified scheduling.

Step 2.4: Generate baseline schedule and equipment allocation plan according to the left-justification method in Section 4.1, and generate personnel allocation plan according to the scheme in Section 4.2.

Step 2.5: Evaluate $\tilde{C}_{\max}$, compare the new individual with the original one according to robustness objectives and select the best individual.

Step 2.6: Execute chaotic catastrophe. If the condition for catastrophe is satisfied, initialize the individual with chaos initialization.

Step 2.7: The original code is modified by start/finish time of decoded scheduling to generate a new right-justified population PopR, whose elites remain.

Step 3: PopL is updated through a differential evolution operator in a process similar to Step 2.1-2.7. The scheduling direction for PopL is opposite to that for PopR, and is decoded by left-justified scheduling to generate the baseline schedule.

Step 4: If the maximum number of generated schedules (Ns) is achieved, end the algorithm; otherwise, turn to Step 2.

B. ENCODING AND DECODING SCHEMES

Random-key (RK) representation is used in this algorithm. When a new solution is obtained, a new schedule can be generated based on the SSGS, and the start/finish time of each operation is transformed into a random-key. Compared with encoding in the activity-list representation, encoding in random-key (RK) representation is simpler since it is not constrained by the precedence constraints. Besides, encoding in RK with start/finish time can be mapped into schedules in a biunique way. To fully utilize the double justification, this algorithm adopts a population-based double justification strategy, and the representations for individuals in the LJ population and RJ population are defined respectively as follows.

PopL: $x = x^{\mathrm{L}} = [S^{\mathrm{b}}_{11}, S^{\mathrm{b}}_{12}, \cdots, S^{\mathrm{b}}_{1|J_1|}, S^{\mathrm{b}}_{21}, \cdots, S^{\mathrm{b}}_{ij}, \cdots, S^{\mathrm{b}}_{n|J_n|}]$ which is used for generating a left-justified schedule.

PopR: $x = x^{\mathrm{R}} = [E^{\mathrm{b}}_{11}, E^{\mathrm{b}}_{12}, \cdots, E^{\mathrm{b}}_{1|J_1|}, E^{\mathrm{b}}_{21}, \cdots, E^{\mathrm{b}}_{ij}, \cdots, E^{\mathrm{b}}_{n|J_n|}]$ which is used for generating a right-justified schedule.

Based on the encoding, left-justified and right -justified scheduling operates alternately in two populations.

In terms of decoding, firstly, the baseline schedule and resource allocation plan are generated by SSGS and robust personnel allocation scheme with the priority represented in code of individual. Secondly, on the basis of proof in Section 4.3, the $\underline{\boldsymbol{d}}$, $\boldsymbol{d}^*$ and $\bar{\boldsymbol{d}}$ can be mapped to $\underline{C_{\max}}$, $C^*_{\max}$ and $\overline{C_{\max}}$ respectively according to the pre-constrained policy. Finally, the fitness of the individual can be calculated by Equation (1) and (2). Note that since the baseline schedule is left-justified, the right-justified schedules should be turned into the left-justified ones by left justification before evaluation for the plan.

C. ADAPTIVE MUTATION AND CROSSOVER

To overcome the limitations of less reliable convergence performance of DE/current-to-best/1 scheme, the DE/current-to-rand-best/1 scheme [40] is adopted. For any target vector i in the G population, the mutation equation is as follow:

$$\boldsymbol{v}_{i,G} = \boldsymbol{x}^{\mathrm{o}}_{i,G} + F \cdot \left(\boldsymbol{x}^{rand}_{best,G} - \boldsymbol{x}^{\mathrm{o}}_{i,G}\right) + F \cdot \left(\boldsymbol{x}^{\mathrm{o}}_{r1,G} - \boldsymbol{x}^{\mathrm{o}}_{r2,G}\right) \tag{14}$$

Where, $\boldsymbol{x}^{\mathrm{o}}_{i,G}$ is the ith individual received from other population, $\boldsymbol{x}^{rand}_{best,G}$ is the best of the 10% vectors randomly chosen from the current population, and $\boldsymbol{x}^{\mathrm{o}}_{r1,G}$, $\boldsymbol{x}^{\mathrm{o}}_{r2,G}$ $(i \neq r^i_1 \neq r^i_2)$ are two individuals randomly chosen from another population. F is the scale factor. Such operator using random subpopulation rather than the best in the whole population as directors can search for more potential good solutions and accelerate the convergence rate. Besides, the diversity of populations is ensured, and the probably of falling into local optimums is avoided.

To further enhance the diversity of populations, crossover is operated between the aberrant individual $\boldsymbol{v}_{i,G}$ and the original individual $\boldsymbol{x}^{\mathrm{o}}_{i,G}$ to generate a new one $\boldsymbol{u}_{i,G}$. $u_{ij,G}$ can be obtained by

$$u_{ij,G} = \begin{cases} v_{ij,G}, & rand \leq cr \ or \ j = j_{rand} \\ x_{ij,G}, & otherwise \end{cases} \tag{15}$$

where *rand* is a random number within the interval [0,1], cr is the possibility of crossover.

F and cr are two important and sensitive parameters. F can control the extension of mutation and cr can control the extension of accepting new individuals. It is hard and imprecise to give their values based on transcendent knowledge, thus an adaptive parameter selection strategy based on Cauchy distribution is given as follows:

$$F = \begin{cases} \mu_F + \sigma \times C\,(0,1) \\ \mu_F = F_l + \dfrac{(F_u - F_l)}{1 + \exp\left(A\left(D_G / D_{\max} - 0.5\right)\right)} \end{cases} \tag{16}$$

$$cr = \begin{cases} \mu_{CR} + \sigma \times C\,(0,1) \\ \mu_{CR} = cr_l + \dfrac{(cr_u - cr_l)}{1 + \exp\left(A\left(D_G / D_{\max} - 0.5\right)\right)} \end{cases} \tag{17}$$

where F_u and F_l is the upper and lower bound of F respectively, cr_u and cr_l is the upper and lower bound of cr respectively. $C\,(0,1)$ denotes the standard Cauchy distribution, and σ is the given standard deviation, usually with the value of 0.1. A determines the curvature of variation curves. When A is smaller, the closer variation curves is to the straight line. D_G is the measure function of population diversity, whose

definition is as follow:

$$D_G = \frac{1}{Np}\sum_{i=1}^{Np}\sqrt{\frac{1}{|J_n|}\sum_{j=1}^{|J_n|}\left(x_{ij,G}-\bar{x}_{j,G}\right)^2} \tag{18}$$

where Np is the size of population, $\bar{x}_{j,G}$ is the vector of jth dimension of the popularity average centre in G iteration. When D_G decreases, the difference among individuals is smaller, which indicates that the population diversity is ruined. $D_{\max} = \max\left\{D_{G'}, G' = 1, 2, \cdots, G\right\}$

From Equation (16) and (17), F and cr can be adaptively adjusted with the variation of population diversity. When the population is of poor diversity, F and cr will increase to enhance the exploration ability and to avoid local optimums. When the population is of rich diversity, F and cr will decrease to enhance the exploitation ability to accelerate convergence of the algorithm. The disturbance exerted by the Cauchy distribution enlarges the exploration and helps to jump out of the local optimums.

D. GREEDY SELECTION AND CHAOTIC CATASTROPHE

In traditional DE algorithm, the fitness value $f\left(\boldsymbol{u}_{i,G}\right)$ of $\boldsymbol{u}_{i,G}$ and the fitness value $f\left(\boldsymbol{x}_{i,G}\right)$ of $\boldsymbol{x}_{i,G}$ are compared based on the greedy selection strategy. Then individual i is replaced by the better one. Suppose the optimization problem is $\min f\left(\boldsymbol{x}\right)$, the greedy selection is operated by the following equation.

$$\boldsymbol{x}_{i,G+1} = \begin{cases} \boldsymbol{u}_{i,G}, & f\left(\boldsymbol{u}_{i,G}\right) < f\left(\boldsymbol{x}_{i,G}\right) \\ \boldsymbol{x}_{i,G}, & \text{otherwise} \end{cases} \tag{19}$$

Based on the greedy selection, the scout mode in artificial bee colony algorithm (ABC) is adopted, and the chaotic catastrophe strategy is introduced to avoid local extremum. Set a counter *trial* for each individual to track the unchanged times. If the fitness value is improved, trial is set to zero. Otherwise, $trial = trial + 1$. When the maximum searching times *Limit* is achieved, chaos initialization is applied and trial is set to zero.

For lexicographical robustness objectives given in Equation (1), comparison between fitness values of individuals $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ is as follows.

$$\begin{cases} f\left(\boldsymbol{x}_1\right) < f\left(\boldsymbol{x}_2\right), & \text{if } AI\left(\boldsymbol{x}_1\right) < AI\left(\boldsymbol{x}_2\right) \\ f\left(\boldsymbol{x}_1\right) < f\left(\boldsymbol{x}_2\right), & \text{if } AI\left(\boldsymbol{x}_1\right) = AI\left(\boldsymbol{x}_2\right) \wedge G\left(\boldsymbol{x}_1\right) < G\left(\boldsymbol{x}_2\right) \\ f\left(\boldsymbol{x}_1\right) = f\left(\boldsymbol{x}_2\right), & \text{if } AI\left(\boldsymbol{x}_1\right) = AI\left(\boldsymbol{x}_2\right) \wedge G\left(\boldsymbol{x}_1\right) = G\left(\boldsymbol{x}_2\right) \\ f\left(\boldsymbol{x}_1\right) > f\left(\boldsymbol{x}_2\right), & \text{otherwise} \end{cases}$$

VI. SIMULATION EXPERIMENTS AND DISCUSSION

A. SIMULATION CASE GENERATION

In order to test the performance of the new model and the proactive robust optimization method for the RSPFDO, a set of simulation scenarios are generated based on the Admiral Kuznetsov aircraft carrier. Three typical mission cases of flight deck operations are introduced. For all the mission cases, $C^d_{\max} = 70$.

Mission case I: six aircraft are assigned for sortie preparation, which contains about 102 operations, making it be a small-size scheduling problem.

Mission case II: nine aircraft are assigned for sortie preparation, which contains about 153 operations, making it be a medium-size scheduling problem.

Mission case III: twelve aircraft are assigned for sortie preparation, which contains about 204 operations, making it be a large-size scheduling problem.

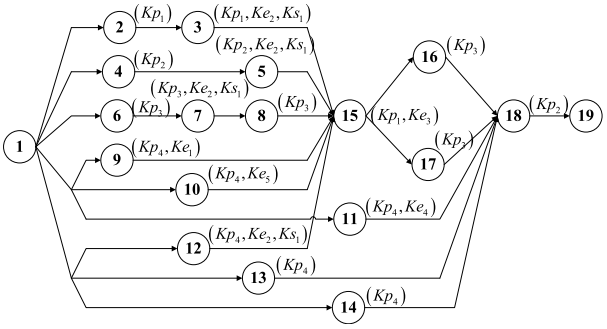

FIGURE 5. The general OON network of single aircraft.

A general ONN, constructed from the precedence relations of single aircraft operations, is shown in Figure 5. In the ONN, $(Kp_k, Ke_{k'}, Ks_1, Kw_{k''})$ that adhering to the operation denotes the requirement of the operation for each type of resource. Note that supply resource requirement is ignored in Figure 5 because there is a one-to-one match between equipment and supply resource $k' = k''$. The required units of personnel and equipment for all the relevant operations are 1. Four trades are considered in the cases, including special equipment, avionics, ordnance and machinery. As for the workstation space, only one kind of this resource (cockpit) is considered. 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 1 shows the ternary interval of durations for twelve aircraft. The operation duration for baseline schedule is different among aircraft according to their types and missions. The symbol '–' in the table indicates that the operation is unnecessary for execution. Table 2 shows the parking spot allocation and released time of aircraft for each mission case. Table 3 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 $[Lw_1, Lw_2, \cdots, Lw_5] = [5, 5, 2, 4, 2]$. Since the equipment and the supply capacity are fixed on the flight deck, only the impact of personnel strength can be analyzed. The average personnel strength of trade k ($k \in Kp$) for a single aircraft is defined as

$$PS_k = \frac{|Lp_k|}{\left(n \cdot \sum_{i\in I}\sum_{j\in J_i} rp_{ijk}\right) \Big/ \sum_{i\in I}|J_i|} \tag{20}$$

Besides, The proposed DSDE algorithm is implemented in MATLAB version 2017, and all the tests are performed

TABLE 1. Ternary interval of durations for the aircraft.

Operation No.	d^* (min)												$\underline{d}$ (min)	$\overline{d}$ (min)
	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}	I_{11}	I_{12}		
2	4	4	6	4	4	3	3	4	3	5	4	3	$d^*_{i2}-0.3$	$d^*_{i2}+0.3$
3	8	8	9	7	7	6	6	7	6	9	7	6	$d^*_{i3}-0.2$	$d^*_{i3}+0.2$
4	3	3	5	3	3	4	4	3	4	5	5	3	$d^*_{i4}-0.3$	$d^*_{i4}+0.3$
5	6	6	7	6	6	5	6	5	5	7	6	5	$d^*_{i5}-0.2$	$d^*_{i5}+0.2$
6	5	5	--	4	4	3	3	--	5	4	8	5	$d^*_{i6}-0.3$	$d^*_{i6}+0.3$
7	4	4	3	5	5	3	3	--	3	3	5	5	$d^*_{i7}-0.2$	$d^*_{i7}+0.2$
8	--	5	--	6	8	6	8	6	9	8	8	6	$d^*_{i8}-0.5$	$d^*_{i8}+0.5$
9	13	13	18	15	15	11	11	18	12	12	15	12	$d^*_{i9}-0.1$	$d^*_{i9}+0.1$
10	6	6	--	--	4	4	5	5	4	4	5	4	$d^*_{i10}-0.1$	$d^*_{i10}+0.1$
11	--	3	5	--	5	3	3	--	3	5	5	3	d^*_{i11}	$d^*_{i11}+0.3$
12	3	3	3	4	4	4	3	3	4	4	4	3	$d^*_{i12}-0.2$	$d^*_{i12}+0.2$
13	12	12	15	12	12	12	12	15	12	12	12	12	$d^*_{i13}-0.4$	$d^*_{i13}+0.8$
14	8	8	10	8	8	8	8	10	8	8	8	8	$d^*_{i14}-0.4$	$d^*_{i14}+0.8$
15	3	--	5	5	3	--	3	5	3	3	3	5	d^*_{i15}	$d^*_{i15}+0.3$
16	10	10	--	12	12	8	9	--	8	6	12	8	$d^*_{i16}-0.4$	$d^*_{i16}+0.6$
17	10	10	--	12	12	8	9	--	8	6	12	8	$d^*_{i17}-0.4$	$d^*_{i17}+0.6$
18	7	7	10	7	7	6	6	6	8	8	7	6	$d^*_{i18}-0.3$	$d^*_{i18}+0.3$

TABLE 2. Parking spot allocation and released time of aircraft for each mission case.

Mission case	Parking spot No. (release time/min)											
	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}	I_{11}	I_{12}
Mission case I	--	2 (0)	3 (0)	4 (0)	5 (0)	--	--	--	9 (0)	10 (0)	--	--
Mission case II	2 (11)	3 (6)	4 (0)	--	--	5 (0)	--	1 (0)	6 (5)	8 (0)	9 (0)	10 (12)
Mission case III	1 (17)	2 (6)	3 (0)	4 (0)	5 (0)	6 (22)	7 (18)	8 (12)	9 (0)	10 (0)	11 (7)	12 (11)

TABLE 3. 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]

on a personal computer with Intel Xeon E5 (3.0 gigahertz) processor and 32 gigabyte RAM. After several preliminary experiments, suitable sets of parameters for DSDE are determined as follows: $Np = 30$, $F_l = 0.2$, $F_u = 0.5$, $cr_l = 0.6$, $cr_u = 0.9$, $A = 10$ and $Limit = 15$.

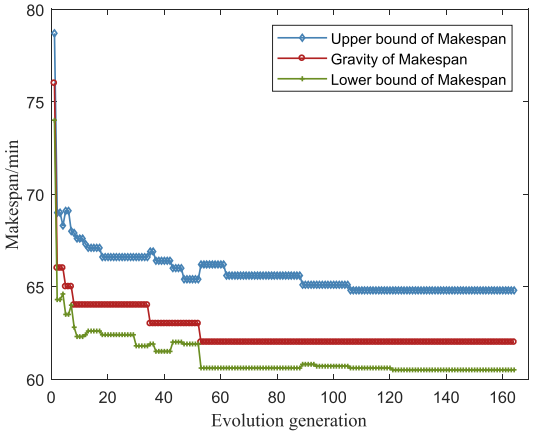

FIGURE 6. Convergence curve of ternary interval of makespan.

B. ONE CASE STUDY

To illustrate the output of the proposed robust scheduling optimization method, the Mission case I with minimum

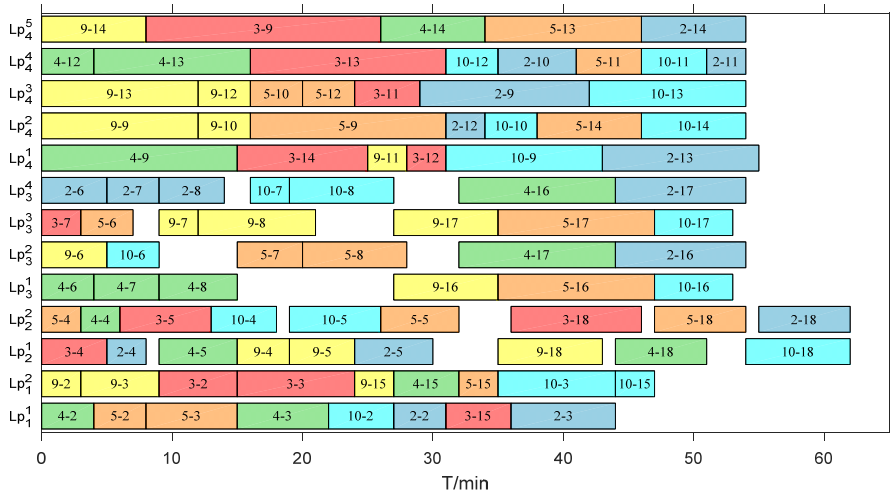

FIGURE 7. Gantt chart of optimal personnel in Mission case I.

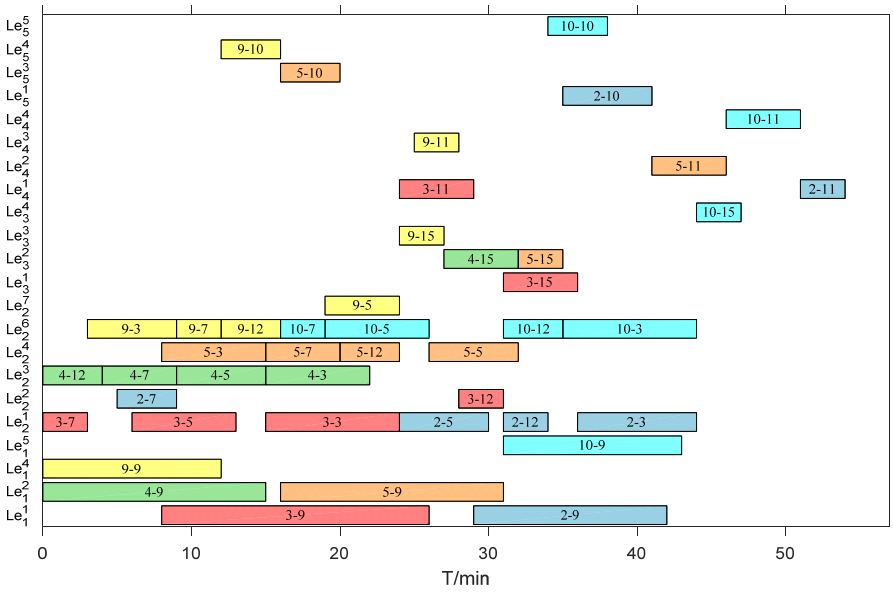

FIGURE 8. Gantt chart of optimal equipment in Mission case I.

scheduling scale is selected for simulation. The average personnel strength of each trade is set as $PS = 2.6$, and according to the formula (20), personnel number of each trade is $[|Lp_k|]_{1\times 4} = [2, 2, 4, 5]$. The maximum number of schedule is set as Ns=10,000. After calculation with the DSDE algorithm, the final ternary interval of the makespan is $\tilde{C}_{\max} = [60.5, 62.0, 64.8]$, and the convergence curve of the upper bound, gravity and lower bound of makespan are shown in Figure 6. It can be seen from Figure 6 that the convergence trends of the three criterions are in accordance with the definition in formula (1) and (2). The result reaches the best agreement index $AI = 1$ quickly, and the comprehensive robust criterion is updated continuously, within which the first-ranking criterion is descending, the other two criterions become smaller when the upper level criterion remains unchanged.

The Gantt charts of the optimal robust baseline schedule and resource allocation plan are shown in Figure 7 and Figure 8. 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' $\left(k' \in Ke\right)$, and $i-j$ represents operation O_{ij}.

To illustrate the interval of the start/finish time of operations, a trapezoidal Gantt chart is proposed. Figure 9 and Figure 10 show the trapezoidal Gantt chart of optimal personnel and equipment respectively. As is shown in the figures, each irregular trapezoid represents the start/finish time interval of certain operation, where the left sideline represents the start time interval and the right sideline represents the

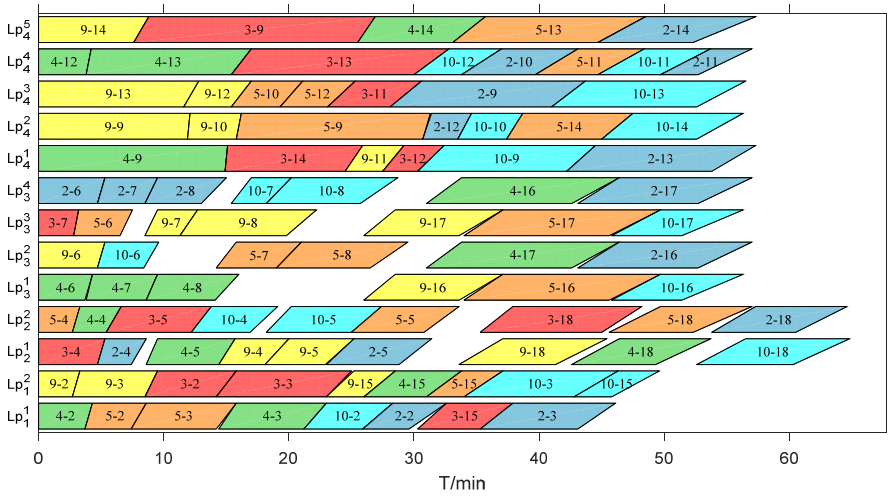

FIGURE 9. Trapezoidal Gantt chart of optimal personnel in Mission case I.

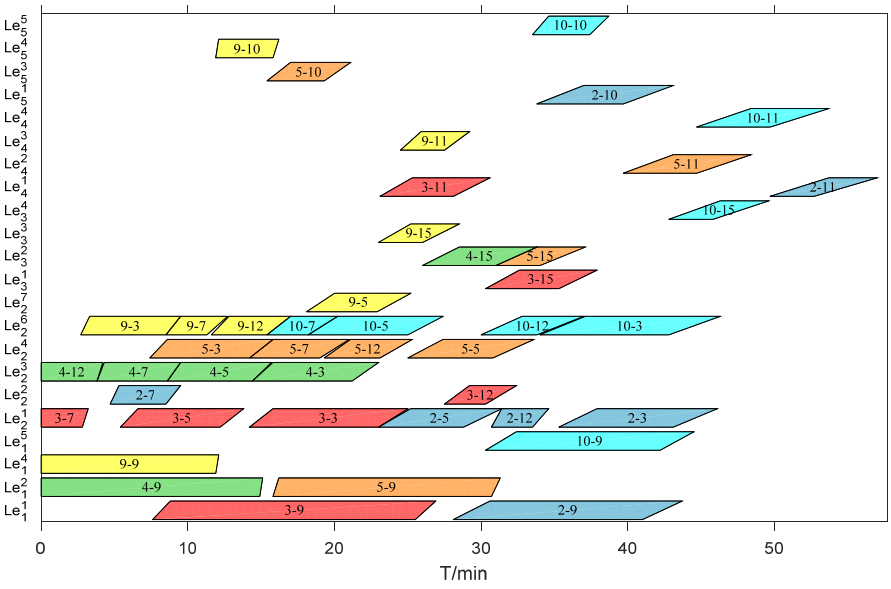

FIGURE 10. Trapezoidal Gantt chart of optimal equipment in Mission case I.

finish time interval. Furthermore, the upper extreme point and the lower extreme point of the left sideline denote the upper bound and lower bound of the start time respectively, while the upper extreme point and the lower extreme point of the right sideline denote the upper bound and lower bound of the finish time respectively.

The gravity of operations is shown in baseline schedule in Figure 7 and Figure 8 according to the Inference 2, so the ternary interval of the start/finish time of operations can be shown completely with the typical Gantt chart of baseline and the trapezoidal Gantt chart.

C. VERIFICATION OF THE EVALUATION METHOD FOR MAKESPAN

As is illustrated in Section 4.3, the ternary interval of makespan is calculated according to scenarios with the upper bound $\underline{\boldsymbol{d}}$, lower bound $\bar{\boldsymbol{d}}$ and gravity of operation duration $\boldsymbol{d}^*$ with the pre-constrained scheduling policy. To verify the evaluation method, the case in Section 6.2 is taken as an text example, which produces the ternary interval number of makespan $\tilde{C}_{\max} = [60.5, 62.0, 64.8]$. The verification method is as follows. Firstly, according to the pre-constrained scheduling policy, the resource flows of personnel, equipment, workstation space and supply resource in the baseline schedule and resource allocation shown in Figure 7 and Figure 8 are converted into extra precedence relations. Secondly, to evaluate the exact distribution of the makespan by simulation with a certain replications, the truncated normal distribution $N\left(\boldsymbol{d}^*, \boldsymbol{\sigma}, \underline{\boldsymbol{d}}, \bar{\boldsymbol{d}}\right)$ is selected to represent the ternary interval number of durations, and five levels of variance are set to represent different levels of distribution that $\sigma = 0.01, 0.1, 0.5, 1, +\infty$. Finally, with the pre-constrained

scheduling policy and random sampling from the truncated normal distribution, Monte-Carlo simulation is applied under five levels of variance, and the number of simulation is set to 5,000. The results are shown in Figure 11.

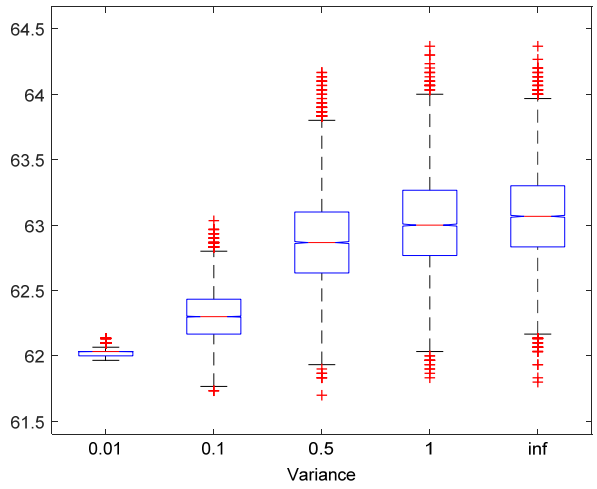

FIGURE 11. Boxplot of makespan with different levels of variance.

It can be seen from Figure 11 that under each level of variance, the distributions of the makespan are all within the interval [60.5, 64.8], and there is even a certain distance from the bounds of the interval. The average makespan and distribution range become larger with the increase of the variance. When the variance is small ($\sigma = 0.01$), the average makespan is close to the gravity of the makespan with little deviation. While with the infinite variance, it can be seen as a uniform distribution, and the average makespan reaches the maximum (63.1). The result of the experiment verifies the Inference 1 and Inference 2, and the evaluation method for makespan is proved feasible. In addition, the bounds of the interval generated by this method are conservative since the distributions of the makespan are quite concentrated compared with the bounds of the interval.

D. COMPARISON AND DISCUSSION

To evaluate the performance of proposed DSDE algorithm, experimental evaluation and comparisons among different mission cases are conducted. Although the RSPFDO has not been studied, it can be regarded as RCMPSP essentially, and four different algorithms for the RC(M)PSP, i.e., multi-modal genetic algorithm (MMGA) [41], modified differential evolution algorithm (MDE) [38], improved particle swarm optimization algorithm (IPSO) [34], and hybrid estimation of distribution algorithm (HEDA) [42] are used to make a comparison with the DSDE algorithm. Based on several preliminary experiments, suitable sets of parameters for each algorithm are determined as follows. For the MMGA, population size, crossover probability and mutation probability are set as 50, 0.8, 0.002, respectively. For the MDE, population size is 100, the scale factor F1=0.7, F2=0.7, and the crossover probability CR1=0.3, CR2=0.7. For the IPSO, number of particles M=50, inertial weight w=0.2, self-learning factor c1=4, social-learning factor c2=2, and selected time periods q=3. For the HEDA, population size NP=250, the number of selected individual to update the probability matrix P=20% NP, the learning speed=0.5, the PBLS accept rate Pper=0, and the descending rate=0.8. Besides, to analyze the impact of personnel strength, three levels of personnel strength are set: 2.6, 3.2 and 3.9 respectively. After 30 independent runs for each algorithm with the maximum number of schedule Ns=10000, the results are shown in Table 4. The performance is measured by the best result (best), average result (avg.), and the worst result (worst).

In Table 4, for all mission cases and PS levels, DSDE can obtain better average result and the best result, and the worst result is better than those of the other algorithms, which means that the DSDE performs better in searching for good solutions. In addition, the span between average result and the worst result is smaller compared with the other algorithms, indicating the robustness of the proposed DSDE. Especially, all the agreement indexes are 1, indicating the DSDE can satisfy the requirement for the due makespan well.

Figure 12 along with Figure 6 shows the convergence curve of ternary interval of makespan for three mission cases with three levels of PS. It is obvious that as the PS increases from 2.6 to 3.9, both the best result, average result and the worst result get improved for all the mission cases. When the PS is larger, the personnel for operations are more sufficient, and the optimal solution are easier to be found, which can be verified in Mission case I and II that when PS=3.9, the optimal result is found in the first several steps in Figure 12(h). When the scale of scheduling becomes larger, the difficulty of searching for the best solution increases, therefore, the difference between the best solution and the worst solution is larger under Mission case III compared with the other two mission cases. Therefore, the personnel strength is a key factor for efficiency and robustness of the flight deck operations. When the scale of aircraft fleet is large, or the due makespan is tight, more personnel are needed.

As for the other four algorithms, only when the value of PS is large (PS=3.9), the four algorithms can find the best result in mission case I, and HEDA and IPSO obtain the best result as well in mission case II. MMGA performs better in mission case III, while in other mission cases HEDA and IPSO which are imbedded in double justification outperform the others. Note that the proposed DSDE performs far better than the MDE, which can be explained by three main reasons. Firstly, a population-based double justification is used, which can help reducing the makespan of the baseline schedule. Typically, the double justification is applied to individuals as a local search scheme. The population-based double justification, however, incorporate this scheme into the course of population evolution, and the computational cost can be reduced. Secondly, a self-adaptive mutation and crossover factors are applied to keep the balance of exploration and exploitation. Last but not the least, a chaotic catastrophe operator based

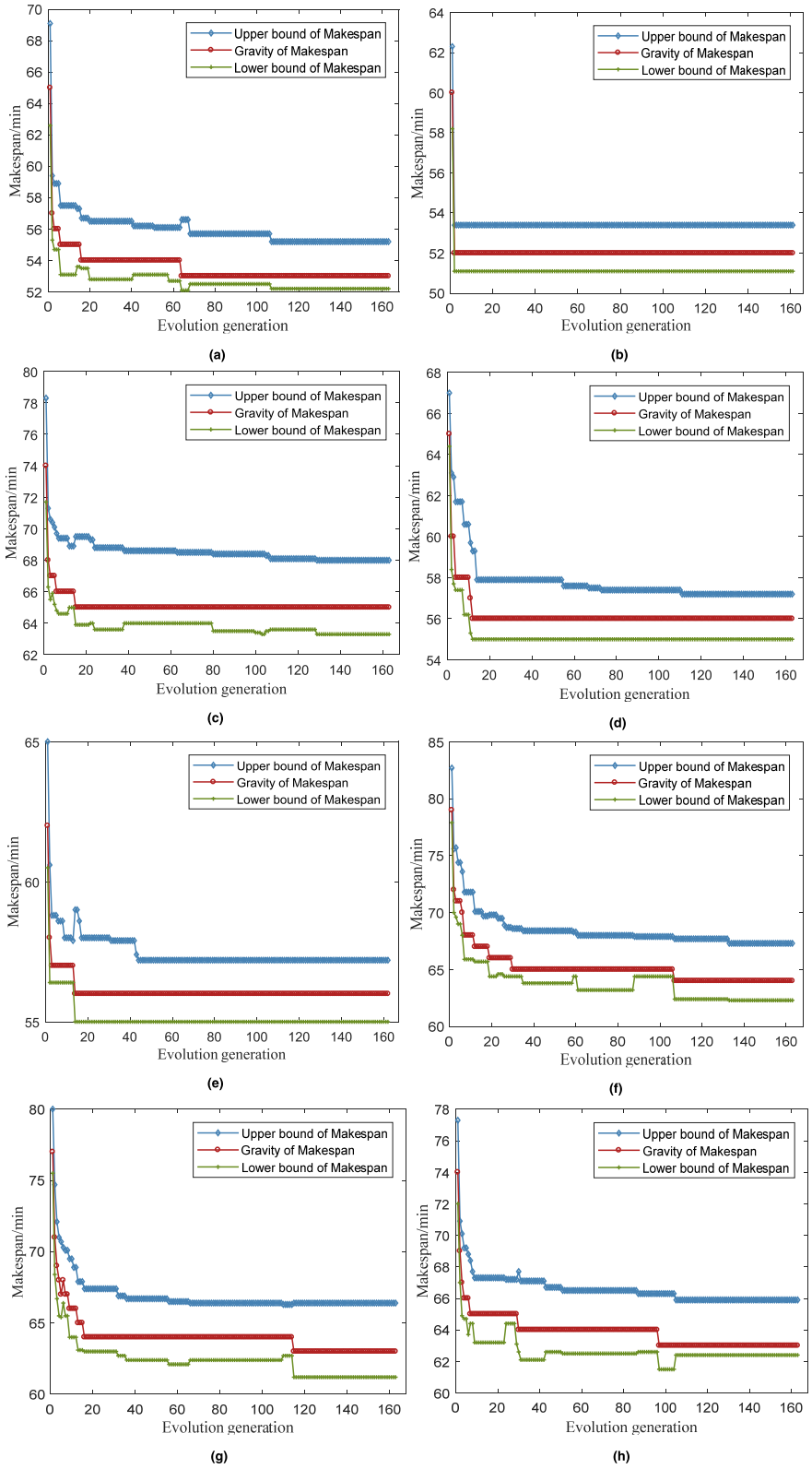

FIGURE 12. Convergence curve of ternary interval of makespan for three mission cases. (a) Mission case I with PS=3.2. (b) Mission case I with PS=3.9. (c) Mission case II with PS=2.6. (d) Mission case II with PS=3.2. (e) Mission case II with PS=3.9. (f) Mission case III with PS=2.6. (g) Mission case III with PS=3.2. (h) Mission case III with PS=3.9.

TABLE 4. Result comparison between DSDE and published algorithms.

Case Set	PS	Performance measurement	DSDE		MMGA		MDE		IPSO		HEDA	
			$\tilde{C}_{\max}$	*AI*	$\tilde{C}_{\max}$	*AI*	$\tilde{C}_{\max}$	*AI*	$\tilde{C}_{\max}$	*AI*	$\tilde{C}_{\max}$	*AI*
Mission case I	2.6	Best	**[60.5, 62.0, 64.8]**	1.00	[63.3, 65.0, 68.1]	1.00	[63.7, 65.0, 69.0]	1.00	[61.5, 63.0, 65.8]	1.00	[61.5, 63.0, 66.5]	1.00
		Avg.	**[60.9, 62.5, 65.3]**	1.00	[64.5, 66.0, 69.3]	0.99	[65.0, 66.8, 69.8]	0.96	[62.3, 63.9, 66.9]	1.00	[62.5, 63.9, 67.2]	1.00
		Worst	**[61.8, 63.0, 65.7]**	1.00	[66.6, 68.0, 70.3]	0.92	[66.5, 68.0, 71.9]	0.65	[62.9, 65.0, 68.0]	1.00	[63.3, 65.0, 67.7]	1.00
	3.2	Best	**[52.2, 53.0, 55.2]**	1.00	[53.2, 55.0, 57.7]	1.00	[53.8, 55.0, 58.2]	1.00	[52.2, 53.0, 56.0]	1.00	[53.1, 54.0, 57.2]	1.00
		Avg.	**[52.4, 53.2, 55.8]**	1.00	[54.4, 55.8, 58.4]	1.00	[55.3, 56.7, 59.5]	1.00	[53.3, 54.8, 57.6]	1.00	[53.9, 55.1, 58.2]	1.00
		Worst	**[53.2, 54.0, 56.0]**	1.00	[56.0, 57.0, 59.1]	1.00	[56.2, 58.0, 60.9]	1.00	[54.1, 56.0, 59.0]	1.00	[54.1, 56.0, 58.6]	1.00
	3.9	Best	**[51.1, 52.0, 53.4]**	1.00	**[51.1, 52.0, 53.4]**	1.00	**[51.1, 52.0, 53.4]**	1.00	**[51.1, 52.0, 53.4]**	1.00	**[51.1, 52.0, 53.4]**	1.00
		Avg.	**[51.1, 52.0, 53.4]**	1.00	[51.1, 52.1, 53.7]	1.00	[51.4, 52.4, 54.5]	1.00	[51.3, 52.0, 54.3]	1.00	[51.1, 52.0, 53.5]	1.00
		Worst	**[51.1, 52.0, 53.4]**	1.00	[51.9, 53.0, 55.1]	1.00	[53.1, 54.0, 55.4]	1.00	[51.1, 52.0, 55.4]	1.00	[51.1, 52.0, 54.3]	1.00
Mission case II	2.6	Best	**[63.2, 65.0, 68.0]**	1.00	[63.6, 66.7, 69.0]	1.00	[64.5, 67.0, 70.8]	0.87	[63.6, 65.0, 68.1]	1.00	[63.8, 65.0, 69.1]	1.00
		Avg.	**[63.5, 65.0, 68.3]**	1.00	[65.3, 66.7, 69.5]	0.99	[66.5, 68.1, 71.8]	0.67	[64.5, 65.8, 69.0]	1.00	[64.6, 65.9, 69.4]	1.00
		Worst	**[63.6, 65.0, 68.6]**	1.00	[66.5, 68.0, 70.1]	0.97	[67.3, 69.0, 73.3]	0.45	[66.9, 68.0, 70.1]	0.97	[64.7, 66.0, 69.8]	1.00
	3.2	Best	**[55.0, 56.0, 57.2]**	1.00	[56.0, 57.0, 59.8]	1.00	[57.9, 59.0, 61.7]	1.00	[55.0, 56.0, 59.0]	1.00	[55.0, 56.0, 58.8]	1.00
		Avg.	**[55.1, 56.1, 57.4]**	1.00	[57.4, 58.6, 61.5]	1.00	[58.5, 59.8, 62.6]	1.00	[57.8, 58.7, 60.7]	1.00	[55.2, 56.4, 59.5]	1.00
		Worst	**[56.4, 57.0, 57.9]**	1.00	[59.2, 60.0, 63.0]	1.00	[59.6, 61.0, 63.7]	1.00	[61.0, 62.0, 63.2]	1.00	[55.9, 57.0, 59.9]	1.00
	3.9	Best	**[55.0, 56.0, 57.2]**	1.00	[55.0, 56.0, 57.3]	1.00	[55.0, 56.0, 58.3]	1.00	**[55.0, 56.0, 57.2]**	1.00	**[55.0, 56.0, 57.2]**	1.00
		Avg.	**[55.1, 56.1, 57.4]**	1.00	[55.2, 56.2, 58.6]	1.00	[55.5, 56.4, 59.1]	1.00	[55.0, 56.1, 57.8]	1.00	[55.0, 56.0, 58.4]	1.00
		Worst	**[56.4, 57.0, 57.9]**	1.00	[56.4, 57.0, 60.0]	1.00	[55.9, 57.0, 60.4]	1.00	[55.0, 57.0, 59.7]	1.00	[55.0, 56.0, 59.0]	1.00
Mission case III	2.6	Best	**[62.3, 64.0, 67.3]**	1.00	[65.2, 67.0, 69.3]	1.00	[67.9, 70.0, 74.5]	0.32	[63.8, 66.0, 68.8]	1.00	[64.9, 67.0, 70.4]	0.93
		Avg.	**[63.1, 64.9, 67.8]**	1.00	[66.8, 68.5, 72.0]	0.63	[69.2, 71.1, 75.0]	0.15	[66.4, 68.3, 71.5]	0.68	[66.2, 68.1, 71.7]	0.70
		Worst	**[63.4, 65.0, 68.1]**	1.00	[67.2, 69.0, 72.9]	0.49	[70.5, 72.0, 75.6]	0.00	[69.0, 70.0, 72.7]	0.27	[67.2, 69.0, 72.5]	0.53
	3.2	Best	**[61.2, 63.0, 66.3]**	1.00	[63.4, 65.0, 68.1]	1.00	[64.8, 66.0, 69.6]	1.00	[63.4, 65.0, 67.9]	1.00	[64.3, 65.0, 68.3]	1.00
		Avg.	**[62.3, 64.1, 66.6]**	1.00	[63.8, 65.5, 68.7]	1.00	[65.6, 67.4, 70.6]	0.88	[64.2, 66.2, 69.1]	0.98	[64.5, 66.3, 69.6]	0.98
		Worst	**[63.1, 65.0, 67.4]**	1.00	[63.8, 66.0, 69.2]	1.00	[66.4, 69.0, 72.4]	0.62	[65.6, 68.0, 71.3]	0.77	[65.2, 67.0, 70.6]	0.89
	3.9	Best	**[62.4, 63.0, 65.9]**	1.00	[62.4, 64.0, 66.5]	1.00	[63.9, 65.0, 67.5]	1.00	[62.4, 64.0, 67.0]	1.00	[63.0, 64.0, 68.2]	1.00
		Avg.	**[62.2, 63.7, 66.4]**	1.00	[62.5, 64.2, 67.1]	1.00	[64.2, 66.0, 68.9]	1.00	[63.8, 65.6, 68.4]	0.99	[63.2, 65.3, 68.3]	1.00
		Worst	**[62.5, 64.0, 66.7]**	1.00	[63.1, 65.0, 67.9]	1.00	[65.2, 67.0, 70.4]	0.92	[65.8, 68.0, 71.5]	0.78	[64.0, 66.0, 69.3]	1.00

on the scout mode of ABC algorithm is adopted to avoid falling into local optimum and increase the diversity of both the population.

Based upon these results, the proposed evaluation method for ternary interval of makespan is verified and proved to be a simple way to evaluate the influence of ternary interval durations on the scheduling of flight deck operations. As for the optimization part of the robust scheduling method, the proposed DSDE algorithm can improve the robustness of flight deck operation with due makespan more effectively compared with other algorithms studied above.

From the simulations and cases study shown above, three main advantages of using ternary interval durations can be concluded as follows. Firstly, the method has a stronger practical operability which requires only the upper, most likely value and lower bound of the operation duration that are relatively easily accessible in the flight deck operations. Compared to the fuzzy schedule and classical binary interval schedules, this method can offer a baseline schedule that are intuitive and operable for execution, as are shown in Figure 8 and Figure 9. Secondly, the evaluation method proposed in Section 4.3 and verified in Section 6.3 takes three times of scheduling with deterministic durations respectively to calculate the ternary interval of makespan, it is simpler to be operated compared with the fuzzy scheduling operator, classical binary interval scheduling operator, and Monte-Carlo simulation for scheduling with stochastic durations. Thirdly, on the basis of the binary interval scheduling, the ternary interval number contains the gravity of durations, which can help commander mastering the most likely schedule under the interval uncertainty, and the parameter can offer important criterion for further plan decisions.

VII. CONCLUSIONS

In this paper, the robust scheduling problem for flight deck operations (RSPFDO) under ternary interval uncertainty of duration is studied to ensure the sustainability of the sortie and recovery operations of carrier-based aircraft fleet. A robust scheduling model for RSPFDO is established, which takes the precedence constraints and resource constraints including personnel, equipment, workstation space and supply resource constraints into consideration. A double-population self-adaptive differential evolution (DSDE) algorithm is proposed to generate robust baseline schedule and resource allocation plan. According to the experimental results, the following conclusions can be made.

Firstly, the proposed scheduling scheme and evaluation method for ternary interval of makespan is verified by Monte-Carlo simulation. Secondly, the proposed DSDE algorithm performs better compared to some state-of-the-art algorithms for the RC(M)PSP under different scales of scheduling mission and different levels of personnel strength, and the results demonstrate that the operations can all be finished within the limitative deck cycle. Compared with the MDE, the population-based double justification, self-adaptive selection of mutation and crossover factors and chaotic catastrophic operator are verified to improve the performance of the basic DE algorithm. To sum up, the proposed DSDE is suitable for solving the RSPFDO under ternary interval uncertainty of duration.

In the future, we will focus on the reactive scheduling when major disruptions such as breakdowns of aircraft or equipments happen during the execution, and more efficient optimal methods are expected to solve the dynamic scheduling for flight deck operations.

REFERENCES

[1] A. Jewell, "Sortie generation capacity of embarked airwings," Center for Naval Analyses, Alexandria, VA, USA, Tech. Rep. C-N00014-96-D-0001, Dec. 1998.

[2] A. Liu and K. Liu, "Advances in carrier-based aircraft deck operation scheduling," *Syst. Eng. Theory Pract.*, vol. 37, no. 1, pp. 49–60, 2017.

[3] J. Ryan, M. Cummings, N. Roy, A. Banerjee, and A. Schulte, "Designing an interactive local and global decision support system for aircraft carrier deck scheduling," in *Proc. AIAA Infotech Aerospace*, St. Louis, MO, USA, 2011, p. 1516.

[4] 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 Trans. Cybern.*, vol. 44, no. 6, pp. 761–773, Jun. 2014.

[5] Y. Wu, L. Sun, and X. Qu, "A sequencing model for a team of aircraft landing on the carrier," *Aerosp. Sci. Technol.*, vol. 54, pp. 72–87, Jul. 2016.

[6] J. C. Ryan and M. L. Cummings, "A systems analysis of the introduction of unmanned aircraft into aircraft carrier operations," *IEEE Trans. Human-Mach. Syst.*, vol. 46, no. 2, pp. 209–220, Apr. 2016.

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

[8] Q. Feng, S. Li, and B. Sun, "A multi-agent based intelligent configuration method for aircraft fleet maintenance personnel," *Chin. J. Aeronaut.*, vol. 27, no. 2, pp. 280–290, 2014.

[9] 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.

[10] R. G. Dastidar and E. Frazzoli, "A queueing network based approach to distributed aircraft carrier deck scheduling," in *Proc. AIAA Infotech Aerosp.*, St. Louis, MO, USA, 2011, p. 1514.

[11] 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 *Proc. 2nd Int. Conf. Rel. Syst. Eng.*, Jul. 2017, pp. 1–9.

[12] B. Michini and J. P. How, "A human-interactive course of action planner for aircraft carrier deck operations," in *Proc. AIAA Infotech Aerosp.*, St. Louis, MO, USA, 2011, p. 1515.

[13] Y. Li, Y. Zhu, F. Yang, and Q. Jia, "Inverse reinforcement learning based optimal schedule generation approach for carrier aircraft on flight deck," *J. Nat. Univ. Defense Technol.*, vol. 35, no. 4, pp. 171–175, 2013.

[14] Y. Wu, X. Pan, R. Kang, C. He, and L. Gong, "Multi-parameters uncertainty analysis of logistic support process based on GERT," *J. Syst. Eng. Electron.*, vol. 25, no. 6, pp. 1011–1019, Dec. 2014.

[15] 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," *Sci. Program.*, vol. 2017, no. 1, 2017, Art. no. 2463252.

[16] X. C. Su, W. Han, W. Xiao, and T. Jiang, "Pit-stop support scheduling on deck of carrier plane based on memetic algorithm," *Syst. Eng. Electron.*, vol. 38, no. 10, pp. 2303–2309, 2016.

[17] X. C. Su, W. Han, Y. Zhang, J. Y. Song, and Z. Y. Zhao, "Scheduling method for maintenance and service support of carrier-based aircraft on flight deck with different man-aircraft matching patterns," *Acta Aeronautica et Astronautica Sinica*, to be published, doi: 10.7527/S1000-6893.2018.22314.

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

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

[20] N. Pang, H. Su, and Y. Shi, "Project robust scheduling based on the scattered buffer technology," *Appl. Sci.*, vol. 8, no. 4, p. 541, 2018.

[21] M. Masmoudi and A. Haït, "Project scheduling under uncertainty using fuzzy modelling and solving techniques," *Eng. Appl. Artif. Intell.*, vol. 26, no. 1, pp. 135–149, 2013.

[22] Y. Alipouri, M. H. Sebt, A. Ardeshir, and M. H. F. Zarandi, "A mixed-integer linear programming model for solving fuzzy stochastic resource constrained project scheduling problem," *Oper. Res.*, vol. 2017, no. 1, pp. 1–21, 2017.

[23] H. Li, Z. Xu, L. Xiong, and Y. Liu, "Robust proactive project scheduling model for the stochastic discrete time/cost trade-off problem," *Discrete Dyn. Nature Soc.*, vol. 2015., no. 2, 2015, Art. no. 586087.

[24] S. Wang, G. Liu, and S. Gao, "A hybrid discrete imperialist competition algorithm for fuzzy job-shop scheduling problems," *IEEE Access*, vol. 4, pp. 9320–9331, 2016.

[25] X. Su, Y. Wu, J. Song, and P. Yuan, "A fuzzy path selection strategy for aircraft landing on a carrier," *Appl. Sci.*, vol. 8, p. 779, May 2018.

[26] D. Lei, "Population-based neighborhood search for job shop scheduling with interval processing time," *Comput. Ind. Eng.*, vol. 61, pp. 1200–1208, Nov. 2011.

[27] D. Lei, "Interval job shop scheduling problems," *Int. J. Adv. Manuf. Technol.*, vol. 60, pp. 291–301, Apr. 2012.

[28] Y. Han, D. Gong, Y. Jin, and Q.-K. Pan, "Evolutionary multi-objective blocking lot-streaming flow shop scheduling with interval processing time," *Appl. Soft Comput.*, vol. 42, pp. 229–245, May 2016.

[29] C. T. Ng, T. C. E. Cheng, A. M. Bandalouski, M. Y. Kovalyov, and S. S. Lam, "A graph-theoretic approach to interval scheduling on dedicated unrelated parallel machines," *J. Oper. Res. Soc.*, vol. 65, no. 10, pp. 1571–1579, 2014.

[30] N. Xie and N. Chen, "Flexible job shop scheduling problem with interval grey processing time," *Appl. Soft Comput.*, vol. 70, pp. 513–524, Aug. 2018.

[31] M. Drwal, "Robust scheduling to minimize the weighted number of late jobs with interval due-date uncertainty," *Comput. Oper. Res.*, vol. 91, pp. 13–20, Mar. 2018.

[32] D. Luo and X. Wang, "The multi-attribute grey target decision method for attribute value within three-parameter interval grey number," *Appl. Math. Model.*, vol. 36, no. 5, pp. 1957–1963, 2012.

[33] S. Abdullah and M. Abdolrazzagh-Nezhad, "Fuzzy job-shop scheduling problems: A review," *Inf. Sci.*, vol. 278, pp. 380–407. Sep. 2014.

[34] Q. Jia and Y. Seo, "An improved particle swarm optimization for the resource-constrained project scheduling problem," *Int. J. Adv. Manuf. Technol.*, vol. 67, nos. 9–12, pp. 2627–2638, 2013.

[35] O. I. Tukela, W. O. Rom, and S. D. Eksioglu, "An investigation of buffer sizing techniques in critical chain scheduling," *Eur. J. Oper. Res.*, vol. 172, no. 2, pp. 401–416, 2006.

[36] M.-Y. Cheng, D.-H. Tran, and Y.-W. Wu, "Using a fuzzy clustering chaotic-based differential evolution with serial method to solve resource-constrained project scheduling problems," *Autom. Construct.*, vol. 37, no. 1, pp. 88–97, 2011.

[37] N. Damak, B. Jarboui, P. Siarry, and T. Loukil, "Differential evolution for solving multi-mode resource-constrained project scheduling problems," *Comput. Oper. Res.*, vol. 36, no. 9, pp. 2653–2659, 2009.

[38] R. Yan, W. Li, P. Jiang, Y. Zhou, and G. Wu, "A modified differential evolution algorithm for resource constrained multi-project scheduling problem," *J. Comput.*, vol. 9, no. 8, pp. 1922–1927, 2014.

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

[40] L.-B. Deng, S. Wang, L.-Y. Qiao, and B.-Q. Zhang, "DE-RCO: Rotating crossover operator with multiangle searching strategy for adaptive differential evolution," *IEEE Access*, vol. 6, pp. 2970–2983, 2018.

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

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

XI-CHAO SU received the B.S. degree in aircraft system and engineering and the M.S. degree in aeronautical and astronautical science and technology from Naval Aviation University, Yantai, China, in 2012, where he is currently pursuing the Ph.D. degree in aeronautical and astronautical science and technology. His research interests include operation scheduling for aircraft and intelligent computation.

WEI HAN received the B.S. degree in aircraft and engine engineering and the M.S. degree in aeronautical and astronautical science and technology from Naval Aviation University (NAU), Yantai, China, in 1992 and 1996, respectively, and the Ph.D. degree in solid mechanics from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2003. He is currently a Professor with NAU. His research interests include operation scheduling for aircraft and flight dynamics.

YU WU received the B.S. degree in aircraft design and engineering from the Nanjing University of Aeronautics and Astronautics in 2011, and the Ph.D. degree in aerospace engineering from Beihang University in 2016. He is currently a Lecturer with the College of Aerospace Engineering, Chongqing University. His main research interests are optimization theory, and planning and decision-making on aircraft operation.

YONG ZHANG received the B.S. degree in aircraft and engine engineering and the M.S. degree in aeronautical and astronautical science and technology from Naval Aviation University (NAU), Yantai, China, in 2000 and 2008, respectively. He is currently with NAU. His research interests cover the general area of the process industry control and automation, as well as related fields such as aerodynamics.

JING-YU SONG received the B.S. degree in electric engineering from Xi'an Jiaotong University, Xian, China, in 2004, and the M.S. degree in electric engineering from Xi'an Jiaotong University in 2007. He is currently an R&D Engineer with the China State Shipbuilding Corporation System Engineering Research Institute. His research interests include aviation system engineering.