

## Article

# A Dynamic Scheduling Method for Carrier Aircraft Support Operation under Uncertain Conditions Based on Rolling Horizon Strategy

Peilong Yuan <sup>1,2,\*</sup>, Wei Han <sup>1</sup>, Xichao Su <sup>1</sup>, Jie Liu <sup>1</sup> and Jingyu Song <sup>3</sup>

<sup>1</sup> Department of Airborne Vehicle Engineering, Naval Aviation University, Yantai 264001, China; hanwei70@yeah.net (W.H.); suxich@126.com (X.S.); liuyexiaobao@163.com (J.L.)

<sup>2</sup> Department of Astronautical Science and Engineering, National University of Defense Technology, Changsha 410073, China

<sup>3</sup> System Engineering Research Institute, China State Shipbuilding Corporation, Beijing 10094, China; songjy\_cs@163.com

\* Correspondence: yuanpeilong17@nudt.edu.cn; Tel.: +86-0535-6637477

Received: 13 July 2018; Accepted: 21 August 2018; Published: 3 September 2018



**Featured Application:** The dynamic scheduling research of carrier aircraft support operations in different missions involves multiple support resources (renewable resources include support operational crews and support equipment, non-renewable resources include oil, oxygen, nitrogen, hydraulic, power), complex scheduling process such as the operational activities that should satisfy serial and parallel constraint relationships, multiple constraints (renewable resources constraints, non-renewable resources constraints, operational space constraints). The efficient coordination of these resources can be described as a multi-resource constrained multi-project scheduling problem (MRCMPSP) under an uncertain environment. In this work, an integer programming mathematical model is established for carrier aircraft dynamic support scheduling to solve the non-deterministic polynomial-time hard (NP-hard) problem. A periodic and event-driven rolling horizon (RH) scheduling strategy that was inspired by the RH optimization method from predictive control technology is presented for the uncertain and dynamic environment. The RH strategy not only reduces the problem size, it also effectively adjusts the baseline scheduling in a reasonable computational time and avoids the unnecessary scheduling with effective resource allocation in the dynamic flight deck environment. The dual population genetic algorithm (DPGA) is designed to solve the large-scale scheduling problem. The computational results show that our approach can provide convenient guidance to carrier aircraft support and scheduling applications in the dynamic flight deck environment, and make better decisions regarding disruption on a real-time basis.

**Abstract:** The efficient scheduling of carrier aircraft support operations in the flight deck is important for battle performances. The supporting operations and maintenance processes involve multiple support resources, complex scheduling process, and multiple constraints; the efficient coordination of these processes can be considered a multi-resource constrained multi-project scheduling problem (MRCMPSP), which is a complex non-deterministic polynomial-time hard (NP-hard) problem. The renewable resources include the operational crews, resource stations, and operational spaces, and the non-renewable resources include oil, gas, weapons, and electric power. An integer programming mathematical model is established to solve this problem. A periodic and event-driven rolling horizon (RH) scheduling strategy inspired by the RH optimization method from predictive control technology is presented for the dynamic scheduling environment. The periodic horizon scheduling strategy can track the changes of the carrier aircraft supporting system, and the improved event-driven mechanism can avoid unnecessary scheduling with effective resource allocation under uncertain conditions. The dual population genetic algorithm (DPGA) is designed to solve the large-scale

scheduling problem. The activity list encoding method is proposed, and a new adaptive crossover and mutation strategy is designed to improve the global exploration ability. The double schedule for leftward and rightward populations is integrated into the genetic process of alternating iterations to improve the convergence speed and decrease the computation amount. The computational results show that our approach is effective at solving the scheduling problem in the dynamic environment, as well as making better decisions regarding disruption on a real-time basis.

**Keywords:** carrier aircraft; dynamic scheduling; support operation; rolling horizon strategy; dual population genetic algorithm

---

## 1. Introduction

The carrier aircraft is the main weapon used in large aircraft carriers. Before taking off, each carrier aircraft requires a series of certain supporting operations, such as inspection, oiling, gas supply, weapon mount, inertial alignment, etc.; several types of carrier aircrafts have to take off synergistically. The support operations involve multiple support resources (renewable resources include support operational crews and support equipment, non-renewable resources include oiling, oxygen, nitrogen, hydraulic, power), complex scheduling processes, as the operational activities should satisfy with serial and parallel constraint relationships—for instance, no oxygenation operation can be done at the time of refueling operation due to security considerations, and multiple constraints (renewable resources constraints, non-renewable resources constraints, operational space constraints, etc.). The main purpose of the synthetically complicated scheduling problem (it has been non-deterministic polynomial-time hard (NP-hard)) is about allocating the limited resources through optimization methods, and the research contents include multiple objects, multiple resource allocation, mission order management, an uncertain dynamic environment, etc. The research regarding aircraft carrier deck scheduling has considered resource-constrained project scheduling problems (RCPSP). Some uncertainties and dynamic disruptions exist in carrier aircraft scheduling, such as resource disruptions (resource station breakdown, the operators reduce) and time disruptions (the operational time for the equipment maintenance, which may become longer). The keys to achieving the carrier aircrafts' scheduling include different facilities, operational crews, and positions with set limitations in time, space, and resources.

RCPSP (which is from 1963) was firstly introduced by Kelley [1]. The objective of RCPSP is to minimize the makespan while satisfying the resource constraints and precedence relationships among the activities [2]. Blazewicz [3] showed that RCPSP is an NP-hard problem in 1983. The majority of the early research (Zhu. 2006 [4], Herroelen. 2005 [5]) aimed at developing better methods to minimize the makespan of the project and simultaneously set the starting times of the multi-project scheduling problem, in order to make all of the resource constraints meet the project scheduling [6]. Kyriakidis [7] went further by enlarging the search capacity for multi-resources and multi-projects. In recent years, RCPSP has gained widespread attention due to its practical importance and computational challenges [8]. Moreover, RCPSPs have current applications in some aspects regarding their time-based, cost-based, and resource-based objectives [9,10]. Less research has been done about carrier aircraft support scheduling, which is considered a RCPSP after incorporating some practical restrictions, especially regarding the dynamic scheduling.

A series of previous research studies about carrier aircraft scheduling problems were conducted, and some research papers have been published. In the early years, Ouija Board [11], an aircraft carrier deck simulation platform, was employed by the US Navy to monitor the arrangement and dispatch of the carrier aircraft; it sought an optimized scheduling strategy by artificial expertise. Johnston [12] provided a brief introduction of flight-deck operations for the carrier aircraft. Ryan [13,14] developed a deck operations course of action planner (DCAP) that was based on a human-computer interaction concept and utilized a conventional integer linear program-based planning algorithm. An agent-based

model [15] for an aircraft carrier flight deck was also developed, which was compared with that of the expert user heuristics over a set of experimental scenarios. Michini [16] applied the reverse reinforcement learning method and Dastidar [17] presented the queuing network optimizing strategy for aircraft scheduling problems. Zhi Zhang [18] designed a human–computer cooperation decision planning system (aircraft carrier deck operation planner (ACDOP) system). Feng Qiang [19] proposed a multi-agent based fleet maintenance personnel configuration method to solve the mission-oriented aircraft fleet maintenance personnel configuration problem. Lianfei Yu [20] viewed the flight deck scheduling problem as an extension of the flexible job shop scheduling problem (FJSP), and designed an improved differential evolution algorithm (IDE) to improve the computational efficiency. Our research team has also published some relevant papers for the carrier aircraft scheduling problem including the constraints in the scheduling [21], the multi-objective integrated supporting scheduling model, and the intelligence algorithm to solve this problem [22,23]. Compared with other land-based aircraft [24,25], the flight deck in an aircraft carrier presents a complex and uncertain environment in which time-critical scheduling and planning must be done to fulfill mission requirements and ensure the safety of operational crews and equipment [26]. However, most previous research has been aimed at developing better solution methods or scheduling strategies based on a static scheduling environment, and most research is oversimplified. The related literatures regarding managing and guiding the carrier aircraft operations and scheduling are very limited. The strong constraints in the operations usually result in a lot of dynamic scheduling disruptions. The research about the dynamic scheduling method of carrier aircraft support and operation is more important and complicated, but there has not been enough.

The carrier aircraft dynamic scheduling, which is a special RCPSP and considered as multi-resource constrained multi-project scheduling problem (MRCMPSP) under uncertain environments, may face a lot of significant disruptions due to resource unavailability, duration deviation, machine breakdowns, operator variation, and uncertain tasks. The traditional deterministic scheduling strategy must be revised and resolved in order to increase the resource utilization and respond to the changing flight deck environment. In order to find optimal solutions under uncertainty, some primary types of existing rescheduling strategies have been researched, such as the reactive scheduling strategy [27], the proactive scheduling strategy [28], and robust execution strategies [29]. In flight deck carrier aircraft scheduling research, Chao Qi [30] has researched aircraft carrier flight deck dynamic planning based on a dynamic hierarchical task network (HTN) planning process for a set of activities with resource constraints. However, the twofold network structure is simple, and still has some limitations. Gürkan [31] integrated the robust schedule design with fleet assignment and aircraft routing problems in a dynamic environment. This made it possible to improve the solution, since a routing decision could eliminate the necessity of inserting idle time or compressing cruise time, but it could not make a real-time decision when unpredictable disruptions occurred and the robust schedule needed to be rescheduled. Jürgen Kuster [32] described how the RCPSP can be used as a basis for real-time decision support in the disruption management of the aircraft turnaround, and argued that the RCPSP dynamic scheduling research for carrier aircraft is significant; however, it is only partial research that did not sufficiently deal with the uncertain random interferences. In other dynamic scheduling research, copious algorithms and methods can be found in the literature such as the branch and bound algorithm [33], the linear programming-based algorithm [34], and other exact algorithms [35], and many heuristic algorithms and metaheuristic algorithms have been also presented in the literature such as the genetic algorithm [36] (GA), hybrid GA [37], and differential evolution (DE) [38].

A review of the relevant literature has shown that although the existing literature has obtained some results and provided theoretical support to improve the dynamic scheduling strategy for carrier aircraft support operations, there are still some limitations. Most carrier aircraft scheduling methods are based on a static flight deck environment, and need to have a number of parameters defined before a scheduling plan is started. The impact of each activity's scheduling time on the available equipment and operational crews were not considered, including the operating time of all of the activities, the allocation of operation resources (renewable resources and non-renewable resources),

and the number of aircraft that need to be supported. The time urgency may affect the selection of scheduling strategies, and the resource limitations can lead to dispatching conflict. The current research cannot meet the demand of real-time dynamic scheduling well for the carrier aircraft support operation. The proactive scheduling strategy can create a schedule that is robust with respect to dynamic disruptions, and make a plan regarding a temporal schedule and resource allocation, but it cannot generate rescheduling simultaneously when the dynamic disruption occurs. Some disruptions may cause a robust scheduling failure, which calls for a new schedule. The reactive strategy can repair a schedule that has been disrupted in terms of reducing disturbance from uncertainties, but it is difficult for these proposed algorithms to generate rescheduling in a short time, and they make little changes to the rescheduling sequence compared with the baseline scheduling. The relevant research studies have also shown that it is difficult to make static scheduling meet the needs of a dynamic deck scheduling environment, and these methods do not meet the requirements of real-life situations. In this paper, we consider actual dynamic deck environment factors, establish an integer programming mathematical model, and propose a dynamic rolling horizon (RH) scheduling strategy for carrier aircraft support operations under uncertain conditions. It is a very interesting work.

The structure of the paper is as follows. In the next section, the dynamic scheduling problem of carrier aircraft support operations is described in detail. Some definitions are proposed; then, the new integer programming model for the carrier aircraft support is established. In Section 3, the periodic and event-driven RH scheduling strategies are proposed to deal with the dynamic deck environment factors and different disruptions, and the associative recovery strategies are discussed in the creation of new support schedules. In Section 4, the improved genetic algorithm for the dynamic deck environment is illustrated in detail. The computational results and analysis are provided in Section 5. Finally, we provide the overall conclusions in Section 6.

## 2. Problem Description and Modeling

### 2.1. Problem Description

The scheduling problem of carrier aircraft support operations is about allocating the supporting resource (renewable resources and non-renewable resources) to complete the operations for preparing the aircraft well as soon as possible. It can be considered a MRCMPSP. Three kinds of constrained resource (renewable resources constraint, non-renewable resources constraint, operational space constraint) exist in the scheduling project. The renewable resources include support equipment resources (oiling station, oxygen and nitrogen station, hydraulic station, power station), the support resources of operational crews (machinery major, ordnance major, avionics major, and special equipment major), and the support space resources (cockpit and landing gear station). The resource station cannot support all of the aircraft in the deck because of the distribution position; a single station only can cover some of the settled carrier aircraft in the flight deck, but the different major operational crews can support all of the aircraft in the flight deck. The non-renewable resources, which are also called consuming resources, include oil, gas, weapons, etc. The support space resources such as the aircraft cockpit make up the support activity existing space constraints, so that the cockpit inspection of each major cannot proceed simultaneously.

There are some rules restricting the operational activities with serial and parallel constraint relationships. (i) No oxygenation operation can be done at the time of refueling operation due to security considerations; (ii) The operational crews for a single major operation can finish different activities with different priorities; for example, the carrier aircraft inspection of ordnance appearance is preferable to the inspection of ordnance cockpit; (iii) The sequence of some activities can be processed simultaneously; for example, the operations of different majors can be exchanged with parallel constraint relationships; (iv) A stationary resource station can only support some settled carrier aircraft; (v) The alignment operation of inertial navigation is generally carried out after all of the activities. In the scheduling problem, the supporting activities of each carrier aircraft include about 14 activities.

Besides, a virtual start activity and a virtual finish activity are set to combine the operational activities of each carrier aircraft to a multi-aircraft network flow. Figure 1 shows the support activities flow chart of a single-aircraft. The solid arrow indicates the serial and parallel constraint relationships of the operational activities. The dotted line indicates that the cockpit space constraint, which cannot proceed simultaneously in the four aircraft cockpit activities. The single-aircraft support process can be considered as a single project in the RCPSP.

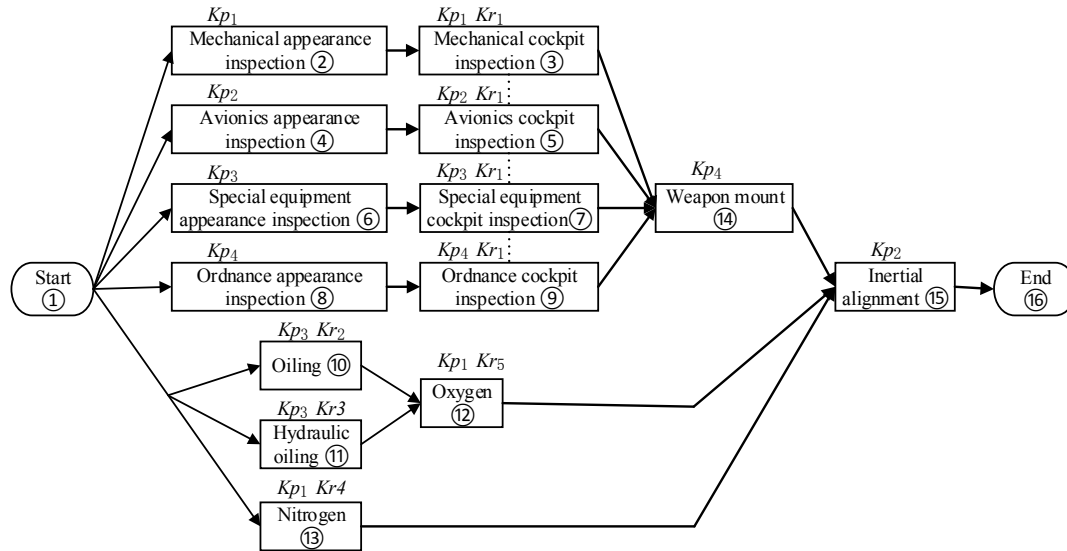


Figure 1. The support activity flow chart of each single carrier aircraft.

In the flight deck environment, some uncertainties and dynamic disruptions exist in the carrier aircraft scheduling, such as resource disruptions (resource station breakdown, the operators reduce), time disruptions (the operational time for the support equipment maintenance, which may become longer), and support tasks that are added, changed, or removed. Therefore, a proposal of a scheduling strategy for a dynamic port environment is required. The objective is to find a dynamic scheduling strategy that minimizes the makespan, improves the resource utilization, and reduces the influence of disruption. The mixed integer linear programming models are discussed next in this section.

## 2.2. Notation

- Sets

$I$  set of carrier aircrafts,  $i = 1, \dots, I$

$J_i$  set of activities for the  $i$ th carrier aircrafts,  $j = 0, 1, \dots, J_i$

$J_i^*$  set of the incomplete activities, including the affected activities in the same time window and activities that start after a disruption ends

$Kp$  set of different kinds of supporting major

$Ks$  set of different kinds of supporting space

$Kr$  set of different types of resource station

$Kw$  set of non-renewable resources upper limit

$Lr_k$  set of  $k$ th( $k \in Kr$ ) supporting resource station

$Lp_k$  set of  $k$ th( $k \in Kp$ ) supporting operational crew

$P_{ij}$  set of pre-constraint activities for the  $i$ th carrier aircraft  $j$ th activity

$R_{pkl}$  set of the carrier aircraft support range by the  $k$ th( $k \in Kp$ ) major  $l$ th( $l \in Lp_k$ ) operational crew

$Rekl$  set of the carrier aircraft support range by the  $k$ th( $k \in Kr$ ) type  $l$ th( $l \in Lr_k$ ) station

- Parameters

$d_{ij}$  duration of activity  $j$  for the  $i$ th carrier aircraft  
 $SX_i$  start time for supporting the  $i$ th carrier aircraft  
 $S_{ij}$  start time of activity  $j$  for the  $i$ th carrier aircraft  
 $T$  total planning horizon/upper bound of the project duration  
 $rp_{ijk}$  demand number of  $k$ th operational crew for the  $i$ th carrier aircraft  $j$ th activity  
 $rs_{ijk}$  demand number of  $k$ th space for the  $i$ th carrier aircraft  $j$ th activity  
 $re_{ijk}$  demand number of  $k$ th equipment for the  $i$ th carrier aircraft  $j$ th activity  
 $rw_{ijk}$  demand number of non-renewable resources for the  $i$ th carrier aircraft  $j$ th activity

- Decision variables

$C_{max}$  the maximum completion time of the activities

$Xp_{ijkl}$  1 if the  $k$ th station  $l$ th equipment is allocated to activity  $j$  for the  $i$ th carrier aircraft; 0 otherwise

$Xe_{ijkl}$  1 if the  $k$ th station  $l$ th equipment is allocated to activity  $j$  for the  $i$ th carrier aircraft; 0 otherwise

$x_{ijk}$  1 if the activity  $j$  for the  $i$ th carrier aircraft has been finished at the  $t$  moment; 0 otherwise

### 2.3. Basic Formulations about Objective Functions and Constraints

The objective is to minimize the completion time of the last support activity.

- Objective function.

$$\text{Min } F = \min (C_{max}) \quad (1)$$

- Constraints.

$$S_{i1} \geq SX_i, \forall i \in I \quad (2)$$

$$E_{ij} = S_{ij} + d_{ij}, \forall i \in I, \forall j \in J_i \quad (3)$$

$$S_{ij} \geq S_{ih} + d_{ih}, \forall (i, h) \in P_{ij}, \forall i \in I, \forall j \in J_i \quad (4)$$

$$\sum_{t=0}^T x_{ijt} = 1, \forall i \in I, j \in J_i \quad (5)$$

$$\sum_{O_{ij} \in I_{it}} rs_{ijk} \leq 1, \forall i \in I, \forall j \in J_i, \forall k \in Ks \quad (6)$$

$$\sum_{i=1}^n \sum_{j \in J(t)} rp_{ijk} \leq |Lp_k|, \forall k \in Kp, t \in [0, T] \quad (7)$$

$$\sum_{i=1}^n \sum_{j \in J(t)} re_{ijk} \leq |Lr_k|, \forall k \in Kr, t \in [0, T] \quad (8)$$

$$\sum_{i=1}^n \sum_{j \in J(t)} rw_{ijk} \leq |Lw_k^t|, \forall k \in Kw, t \in [0, T] \quad (9)$$

$$\sum_{l \in Lr_k} Xe_{ijkl} = re_{ijk}, \forall i \in I, \forall j \in J_i, \forall k \in Kr \quad (10)$$

$$\sum_{l \in Lp_k} Xp_{ijkl} = rp_{ijk}, \forall i \in I, \forall j \in J_i, \forall k \in Kp \quad (11)$$

$$\sum_{i \in I - re_{kl}} \sum_{j \in J_i} Xe_{ijkl} + \sum_{i \in I - rp_{k'l'}} \sum_{j \in J_i} Xp_{ijkl} = 0 \quad (12)$$

$$\forall j \in J_i, \forall k \in Kr, \forall l \in Lr_k, \forall k' \in Kp, \forall l' \in Lp_k$$

Constraint (2) ensures that the first support activity must start before the carrier aircraft is transported up to the flight deck and parked in a permanent position. Constraint (3) defines the



relationship between the finish time of each activity with the start time. Constraints (4) ensures the precedence relationship, which means that any activity cannot be started before the completion of its preceding activity. Constraint (5) ensures that each activity is executed only once from one exact period,  $t$ . Constraints (6), (7) and (8) represent the capacity constraints for renewable resources (operational space, operational crews, resource stations), respectively. Constraint (9) represents the capacity constraints for the non-renewable resources in each time period. Constraints (10) and (11) represent the allocation of the operational crews and resource stations, which need to be equal to the demand. Constraint (12) represents the support coverage of the operational crews and resource stations.

#### 2.4. Dynamic Disruption Rescheduling Model

Some disruptions may occur in the uncertain scheduling environment after some activities have been completed as planned, which may cause a scheduling delay or prescheduling failure. Then, the dynamic rescheduling model is established. The objective function of Equation (13) is to minimize the sum of the earliness and tardiness values of all of the activities' completion time cost, comparing the recovery schedule with the baseline schedule.

$$\delta = \min \sum_{i \in I} \sum_{j \in J_i^*} \left( \left| D_{ij}^* - D_{ij} \right| \right) \quad (13)$$

$D_{ij}^*$  represents the start time of the incomplete activity  $j$  of carrier aircraft  $i$  after disruption, and  $D_{ij}$  represents the start time of the incomplete activity  $j$  of carrier aircraft  $i$  before disruption.

This rescheduling model with disruptions differs from static scheduling. The dynamic scheduling problem is incorporated into the formulation in Equation (13) through optimizing the makespan objective function firstly, and then minimizing the sum of the earliness and tardiness values of all of the activities' completion time cost. The additional objective function is added to optimize the stability of the rescheduling activities, and obtain a more optimal new scheduling for the disruptions.

### 3. Rolling Horizon Dynamic Scheduling Strategy

The rolling horizon (RH) dynamic scheduling strategy is a specific manifestation principle of RH optimization in predictive control. Unlike the traditional optimization methods (proactive scheduling strategy [28], robust execution strategy [29]), the basic method of RH dynamic strategy involves dividing the support activities into multiple scheduling windows based on the arrival baseline scheduling sequence. The RH window can be continuously updated along with the scheduling time. A number of activities are selected and scheduled from activities waiting for processing in an activities window. The relevant operational information for carrier aircraft rescheduling is updated. Then, the other activities in the next RH window are selected. Then, activities in the RH window are scheduled again, and a new decision is made. The procedure is repeated until all of the support operations have been finished. In the RH dynamic scheduling strategy, the global scheduling optimization requirement is relaxed, and the local optimal scheduling can be implemented. The RH dynamic scheduling strategy has been researched in some problems, including a job shop scheduling problem [39], an Earth observing satellites scheduling problem [40], and a yard crane scheduling problem [41,42]. By comparing the results with the traditional method, these papers verified that the dynamic rolling horizon (RH) strategy can provide a positive contribution to improve the efficiency of dynamic scheduling and effectively adjust to the baseline schedule. However, there has not yet been any research on RCPSP, especially the carrier aircraft dynamic scheduling problem by the RH dynamic scheduling strategy.

The key to the RH dynamic scheduling strategy is selecting an appropriate rolling strategy. In the carrier aircraft dynamic scheduling problem, the periodic and event-driven RH scheduling strategies are designed to reschedule the update supporting activities window in a fixed period or whenever the activity that is about to occur gets disrupted. Machine breakdown, operators downsizing in operational crews, and support tasks that are added, changed, or removed are considered disruption

events. The time deviation of each incomplete activity after disruptions occur is checked, and whether to perform a rolling rescheduling is decided. The traditional event-driven dynamic scheduling strategy makes it easy to ignore the condition that the schedule can buffer partial disturbance and produce unnecessary rescheduling. The event-driven strategy is improved by building a profit and loss model as seen in Equation (13) to get the optimum rescheduling.

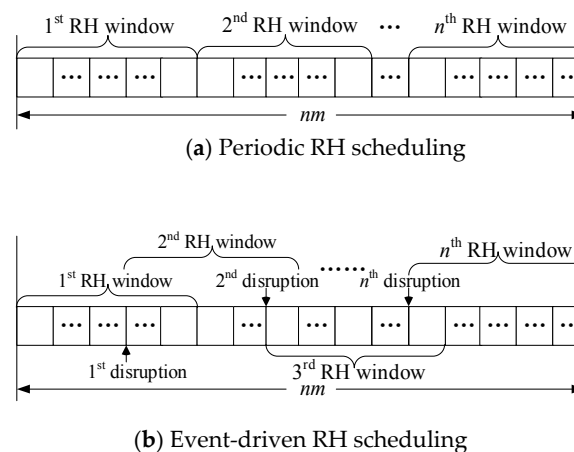
The optimization procedure of the RH dynamic scheduling modeling is described as follows.

Step 1. The proactive scheduling strategy is used to create a static schedule using the heuristic algorithm without considering the dynamic scheduling environment. The starting time and ending time can be predetermined in the proactive schedule.

Step 2. The time window is set to choose the activities. The RH strategy is applied. Multiple simple static schedules are generated by periodic RH scheduling strategy. The detail of the dynamic periodic RH scheduling strategy is shown in Figure 2a. In this step, if all of the activities in the current planning horizon are completed, it should be rolled to the next planning horizon. If several existing or new activities are generated simultaneously by different resource stations and operational crews, they should only be rolled into one period.

Step 3. The event-driven RH scheduling strategy is applied from the disruption start time. The new RH window will be set if the disruption occurs (e.g., machine breakdown) and rescheduling should be performed. In contrast, the latest plan completion time of all of the incomplete activities is set as the ending time. The details of the event-driven RH scheduling strategy are shown in Figure 2b.

Step 4. The rolling stoppage rule is set. If the latest completion time in the rescheduling of all of the activities occurs, the rolling process needs to be terminated.



**Figure 2.** Illustration of the dynamic rolling horizon (RH) strategy.

In an RH window, the scheduling procedure will go through four states based on the current scheduling time: (i) a new task; (ii) waiting tasks; (iii) running tasks; and (iv) finished tasks. In detail, a new task will be generated in the current RH window from the waiting tasks; it should be run and become the finished tasks. Then, another new task is scheduled again through the same procedure. The detail of the dynamic RH scheduling strategy is shown in Figure 3.

In the periodic RH scheduling strategy, the number of RH scheduling and the system adaptability to dynamic factors are determined by the size of time window  $\Delta T$ . If the  $\Delta T$  is too small, the amount of computation is reduced, but it considers less global information, and can lead to a decline in global performance. If  $\Delta T$  is too large, the computation time may increase by the heuristic algorithm, but the final performance may enhance the global solution. No fixed standard for the setting of  $\Delta T$  existed. In this paper, the  $\Delta T$  is specified according to the general characteristics of the scheduling problem and the scheduling algorithm efficiency. We will make an intensive discussion in the Section 5 to get an appropriate value in the RH scheduling strategy.



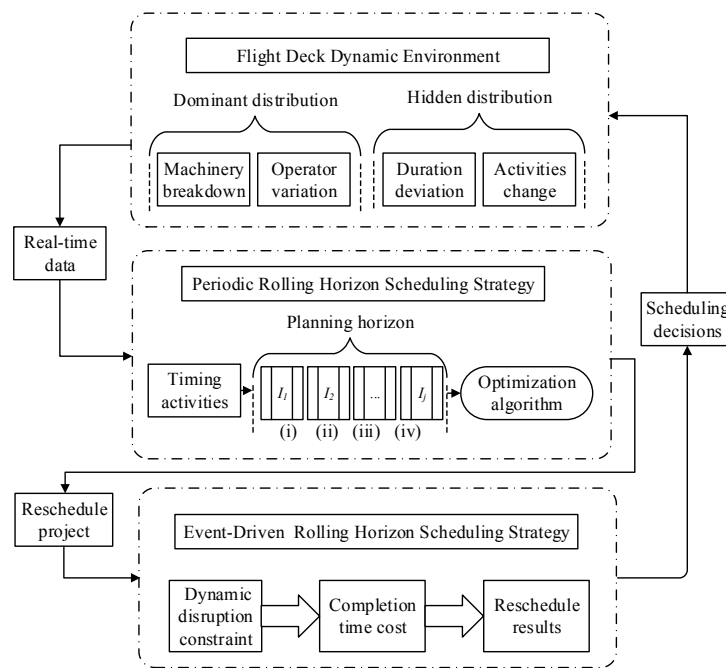


Figure 3. The flow chart of the dynamic rolling horizon scheduling strategy.

The pseudo-code of the RH dynamic scheduling strategy algorithm is described in Algorithm 1

---

**Algorithm 1.** Pseudo-code of rolling horizon dynamic scheduling strategy algorithm for carrier aircraft dynamic support scheduling under uncertain conditions.

---

**Input:** The object network data, priority of support operations, resource restraint, time window  $Tn$ , and disruption type.

**Output:** A dynamic scheduling project.

01: Generate a baseline static schedule by optimization algorithm. /\*The pseudo-code of the optimization algorithm is given in the following Algorithm 2\*/

02: **For** each activity of carrier aircraft  $I$  waiting for operation in set  $J_i$

03:  $RH \leftarrow \text{NULL}$ ;  $\text{Finished } S_n \leftarrow \text{NULL}$ ; /\*Initialization\*/

04: **For** time window  $Tn = 1$  to  $n$ , **do**

05: Select the timing of activities ( $An$ ) in time window  $Tn$ ; /\* $Tn$  is the  $n$ th time window\*/

06: **For**  $A_i \in An$ . Each activity that arrives before  $Tn$  **do**

07: **If** no disruption occurs

08: Add  $A_i$  to set  $RH$ ; /\* $RH$  is the rolling horizon\*/

09: Sort all of the activities in set  $RH$ , and schedule each activity by optimization algorithm; /\*Algorithm 2\*/

10: **Else if** dynamic disruption occurs

11: Select the timing of activities ( $An$ ) in a new time window for the set of the incomplete activities  $J_i^*$ ;

12: Add  $A_i$  to set  $RH$ ;

13: Sort all of the unprocessed activities in set  $RH$ , and reschedule each activity by optimization algorithm;

/\*The pseudo-code of the optimization algorithm is given in the following Algorithm 2\*/

14: **End if**

15: Add activity  $A_i$  to set of finished  $S_n$ ;

16: **End for**

17: Call Algorithm 2 in order to reschedule in  $RH$ ;

18: **End for**

19: Update the scheduling decisions;

20: **End for**

---

#### 4. Algorithm for Carrier Aircraft Dynamic Scheduling

In the RCPSP, when the resource constraints and number of projects are small, some exact algorithms (e.g., branch and bound algorithm, linear programming-based algorithm) can be employed to get the scheduling. However, the carrier aircraft support operation problem, which is considered a multi-resource constraint multi-project scheduling problem, involves plenty of carrier aircraft and activities. For this case, a heuristic algorithm has been developed. In this paper, we use the dual population genetic algorithm (DPGA), which has been used for adaptive diversity control [43], and combine the heuristic algorithm with the RH strategy to solve the carrier aircraft dynamic scheduling problem.

The DPGA is a parallel genetic algorithm that is designed to improve the quality and diversity of the population. Firstly, encoding and decoding processes are set to get population and matching scheduling. The left and right populations proceed simultaneously with crossover, mutation, and local search processes. For the left population, right scheduling is made in the encoding and decoding processes. For the right population, left scheduling is made in the encoding and decoding processes. The original coding is corrected by decoding the end time of the scheduling solution, and the elite population is reserved. The pseudo-code of the DPGA with a local search strategy is described in Algorithm 2. The details are clearly expressed in the next section.

---

**Algorithm 2.** The pseudo-code of the dual population genetic algorithm for carrier aircraft dynamic support scheduling under uncertain conditions.

---

**Input:** The object network data

**Output:** A project scheduling

```

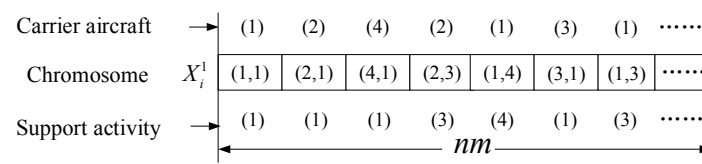
01: Initialize [pop_left, pop_right];
02: Decoding_left(chromosome[pop_left]), schedule_left([pop_left]);
03: Fitness[pop_left], evaluate(schedule_left[pop_left]);
04: For generation = 1 to GEN do
05: Select_parents(chromosome[pop_left], parents[pop_left]);
06: Crossover(parents[pop_left], children[pop_left]);
07: Mutation (children[pop_left]);
08: Decoding_left(chromosome[pop_left]), schedule_left([pop_left]);
09: Fitness[pop_left], evaluate(schedule_left[pop_left]);
10: Select local optimal individual[pop_left];
11: Local search for the left schedule;
12: Update chromosome[pop_right]. Save elite[pop_left]
13: Select_parents(chromosome[pop_right], parents[pop_right]);
14: Crossover(parents[pop_right], children[pop_right]);
15: Mutation (children[pop_right]);
16: Decoding_right(chromosome[pop_right]), schedule_right([pop_right]);
17: Fitness[pop_right], evaluate(schedule_right[pop_right]);
18: Select local optimal individual[pop_right];
19: Local search for the right schedule;
20: Update chromosome[pop_left] Save elite[pop_right];
21: End for
22: Decoding_left(chromosome[pop_left]), schedule_left([pop_left]);
23: Return a best schedule;

```

---

##### 4.1. Encoding and Decoding

In the RCPSP research, the task list encoding method is more efficient to get the scheduling results, and the chromosome with a two-dimensional matrix is designed.  $X = [x_1, x_2, x_3, \dots, x_k]$ ; here,  $x_k$  is  $(i, j)$ ,  $i$  represents the number of different carrier aircrafts ( $n$ ),  $j$  represents the activities ( $m$ ) satisfied to the procedure constraints, and  $m \times n$  represents the total activities (see Figure 4).



**Figure 4.** An illustration example of the chromosome encoding.

The chromosome coding example in Figure 4 shows the process of carrier aircraft scheduling. The decoding can make the chromosome into a corresponding scheduling solution in the algorithm. The start time, finish time, and resource allocation of different activities can be achieved through the decoding process. Two kinds of schedule generation schemes—the serial schedule generation scheme (SSGS) and the parallel schedule generation scheme (PSGS)—are used in the decoding process. Two allocation rules are set in this paper regarding the minimum accumulated processing time first (MAPTF) for the operational crews, and the minimum total processing time remaining (MTPTR) for the covering resource stations. In the first step, we use the SSGS method to allocate the resource; two allocation rules (MAPTF and MTPTR) are used for the operational crews and resource stations. The start time and finish time of each activity can be scheduled by the designed strategies. In the second step, the PSGS is used to minimize  $\delta$  after the disruption occurs by considering the start time and finish time of each activity. In each RH window, the parallel process can modify the reschedule close to the baseline schedule without disruption. Finally, the new schedule is generated for the resource allocation after the disruption occurs.

#### 4.2. Fitness Evaluation and Parent Selection

The fitness evaluation strategy is adopted to get smaller values of the objective functions corresponding to smaller values of the fitness function. The robustness of the solution is reduced by avoiding the improper selection of the fitness function. The fitness function in the optimization procedure is defined as follows:

$$f(s_i) = 2 - \Delta p + \frac{2(s_i - 1)(\Delta p - 1)}{N - 1} \quad (14)$$

Here,  $N$  represents the population size,  $s_i$  represents the location of chromosomes  $i$  in the population, and  $\Delta P$  represents the selective pressure which is linear ranking from one to two.

The parent selection strategy is aimed at creating the offspring for the next generation population to ensure the population diversity. A roulette approach is used in this paper through choosing different chromosomes among the current population.

$$P(x_j) = f(x_j) / \sum_{i=1}^n f(x_i), (j = 1, 2, 3, \dots, n) \quad (15)$$

Here,  $P(x_j)$  represents the selection probability of each individual, and  $f(x_j)$  represents the fitness of each individual. A series of random numbers (e.g.,  $i$ ) between 0 and 1 are generated. If the number  $i$  is located in the cumulative probabilities between two individuals, the individual with a larger fitness will be selected in the new population with higher cumulative probability.

#### 4.3. Crossover and Mutation

In order to avoid illegal results with procedure constraint, the one-point crossover strategy with a modification strategy is adopted. The crossover steps are described as follows.

Step 1. Generate the crossover position ( $a$ ) and select the parent chromosomes  $x_i^1$  and  $x_j^1$  through the crossover probability.

Step 2. Interchange the supporting sequence of the parent chromosomes  $x_i^1$  and  $x_j^1$  before the position ( $a$ ), and obtain the new  $x_i^2$  and  $x_j^2$  chromosomes.

Step 3. Choose the sequence gen of the  $x_j^1$  chromosomes after position ( $a$ ), and rearrange them according to the  $x_i^1$  order to get the new chromosome  $x_i^3$ . The  $x_j^3$  chromosome can be also obtained through the same method. The process of crossover is shown in Figure 5.

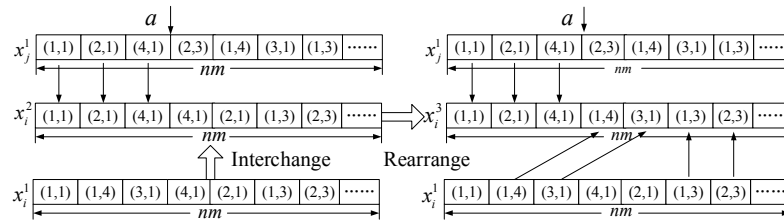


Figure 5. An illustrated example of crossover operation.

The mutation strategy is used to maintain the diversity of the population; the traditional strategies based on the insertion or interchange mutation strategy can only generate a smaller neighborhood mutation range. In this paper, a new neighborhood structure is designed to get a new adaptive mutation strategy, which is adapted for the mutation probability ( $P_i$ ).

$$P_i P_i^m - \frac{(P_i^m - P_i^n) G_i}{G_{\max}} \quad (16)$$

$$G_i = \frac{1}{N} \sum_{i=1}^N (C_{\max}^i - \overline{C_{\max}})^2 \quad (17)$$

Here,  $P_i^m$  and  $P_i^n$  represent the maximum and minimum value of  $P_i$ , while  $G_i$  represents the diversity of population,  $G_{\max} = \max\{G_1, G_2, \dots, G_n\}$ .

The mutation steps are described as follows.

Step 1. Select the carrier aircraft  $i$  through the selective probability  $P$  in the chromosomes ( $X$ ), and then get the activity sequence ( $A$ ) of the selected carrier aircraft in the chromosomes.

Step 2. Select a series support sequence ( $S$ ) in  $A$  and record the mutation location.

Step 3. Reschedule the selected support activities by the serial and parallel constraint, and get the new activity sequence ( $S'$ ).

Step 4. Insert the rescheduling sequence ( $S'$ ) into the original chromosome ( $X$ ). The inserted location is randomly in the mutation location.

#### 4.4. Local Search Strategy and Dual Population Genetic Algorithm

In order to enhance the global optimization speed of d, a local search strategy is introduced to improve the search efficiency and avoid premature optimization results. The local search strategy is added to get the local search results and select the local optimal individual by a simulated annealing strategy in the population after a global search for the crossover and mutation. The pseudo-code of the local search algorithm is described in Algorithm 3.

---

**Algorithm 3.** The pseudo code of local search algorithm in the dual population genetic algorithm (DPGA).

---

**Input:** The object network data,  
**Output:** Project scheduling  
01:  $s \leftarrow$  Generate initialized solution [pop];  
02:  $T \leftarrow T_0$ ;  
03: **While** termination conditions not met **do**;  
04:  $s' \leftarrow$  Pick random(pop(s))  
05:     **if**  $f(s') < f(s)$  **then**  
06:          $s \leftarrow s'$ ; /\*  $s'$  replace  $s$  \*/  
07:     **else**  
08:         Accept  $s'$  as a new solution with probability  $p$ ;  
09:     **end if**  
10: Update ( $T$ );  
11: **end while**

---

## 5. Experimental Results and Analysis

In this section, experiments of the RH scheduling strategy using the DPGA (Dwyer Instruments, Inc., Michigan, IN, USA) with a local search strategy for the aircraft carrier operations are described in detail. The experiments were performed using a PC with two Intel(R) Core(TM) i5-4258U CPU @ 2.4 GHz processors and 4 GB RAM (Intel cooperation, Santa Clara, CA, USA). The algorithm has been coded in MATLAB 2017(a) software (The MathWorks, Inc., Natick, MA, USA). All of the instances were solved by the general genetic algorithm (GA), improved genetic algorithm with local search named Memetic algorithm [22], different evaluation algorithm (DE) [23], and DPGA. The computational results were compared and analyzed to test the algorithms' performance. Finally, a comparison was conducted between the proposed approach RH scheduling strategy in this paper and the static reactive rescheduling strategy after distributions occurred in the support operations.

### 5.1. Problem Classification and Initial Data Set

During the daily operations, an aircraft has a fixed order of tasks. We take the experimental case study of the first wave attack of the typical mission for the carrier aircraft operation scheduling on the flight deck with a series of supporting activities. Five types of carrier aircraft, including the warning aircraft, antisubmarine aircraft, attack aircraft, electronic warfare aircraft, and refueling aircraft, are researched to be supported before takeoff in the flight deck. In order to describe the problems from the current carrier aircraft support operations, and define the scope of the study as well as clearly define the scheduling problem, the support operations were considered as MRCMPSP, and some assumptions were made in this section as follows: (i) All of the support operational crews and resource stations can be available at the beginning of all of the activities; (ii) The non-renewable resources (e.g., oil, gas, weapon, etc.) supplied by the resource station are sustainable supplies; (iii) The support activities of each carrier aircraft cannot be changed in the static scheduling; (iv) When disruptions occur in the operational process, the support time of the disruption activities becomes longer for the maintained process or operator; (v) The setup time of each activity and transferring time between adjacent activities are negligible.

Three kinds of resource constraint (space constraint, operational crews constraint, resource stations constraint) exist in the project. For the space constraint, we only consider the cockpit space constraints  $Ks1$  in the four kinds of cockpit inspection activities in this experiment. There are four kinds of operational crews, including the mechanical major ( $Kp_1$ ), avionics major ( $Kp_2$ ), special equipment major ( $Kp_3$ ), and ordnance major ( $Kp_4$ ), for the maintenance and service support operations of the carrier aircraft. The numbers of each professional operational crew are respectively  $|Kp_1| = 8$ ,  $|Kp_2| = 6$ ,  $|Kp_3| = 8$ , and  $|Kp_4| = 6$ . Five kinds of resource stations are laid out on the deck, including the power station ( $Kr_1$ ), oiling station ( $Kr_2$ ), hydraulic station ( $Kr_3$ ), nitrogen station ( $Kr_4$ ), and oxygen

station ( $Kr_5$ ) for the service support operations of non-renewable resources. The numbers of each resource station in the experiment are respectively  $|Kr_1| = 8$ ,  $|Kr_2| = 6$ ,  $|Kr_3| = 5$ ,  $|Kr_4| = 5$ , and  $|Kr_5| = 5$ . The renewable resources (operational crews and resource stations) are allocated according to the support activities in the different projects; that is, any kind of operational crew that has no support scope constraint can support all of the carrier aircraft, but the different resource stations can only provide service support for some of carrier aircraft in the fixed parking position. Table 1 shows the station set of each kind of resource station that covers the waiting support carrier aircraft in the fixed parking position. The supplied non-renewable resources (power, oil, hydraulic, nitrogen, oxygen, etc.) have unlimited total usage, but the ability of the crew to simultaneously provide the support service is limited.  $Kw$  represents the upper limit of non-renewable resources for the simultaneous support operation.  $Kw$  is set [6, 5, 2, 4, 2].

**Table 1.** Coverage relationship between resource station and carrier aircraft.

Carrier Aircraft	Resource Station Coverage Category				
	$Kr_1$	$Kr_2$	$Kr_3$	$Kr_4$	$Kr_5$
1	[1]	[1]	[1]	[1]	[1]
2	[1, 2]	[1]	[1]	[1]	[1]
3	[1, 2]	[1, 2]	[1, 2]	[1, 2]	[1, 2]
4	[2, 3]	[2]	[1, 2]	[1, 2]	[1, 2]
5	[3, 4]	[2]	[2]	[2]	[2]
6	[3, 4]	[3]	[2, 3]	[2, 3]	[2, 3]
7	[4, 5]	[3, 4]	[3]	[3]	[3]
8	[5, 6]	[3, 4, 5]	[3, 4]	[3, 4]	[3, 4]
9	[5, 6, 7]	[4, 5]	[3, 4]	[3, 4]	[3, 4]
10	[6, 7]	[5, 6]	[4, 5]	[4, 5]	[4, 5]
11	[7, 8]	[6]	[4, 5]	[4, 5]	[4, 5]
12	[8]	[6]	[5]	[5]	[5]
13	[8]	[6]	[5]	[5]	[5]

Four kinds of support operational crews and five types of resource stations make up the flexible and sustained carrier aircraft support activities in the flight deck. The support time (in minutes) of each activity in a single carrier aircraft are shown in the Table 2.

**Table 2.** Support time of each activity in a single carrier aircraft.

Carrier Aircraft	Support Activities													
	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩	⑪	⑫	⑬	⑭	⑮
1	12	3	5	7	6	9	0	0	18	4	5	5	0	10
2	13	4	3	6	4	7	9	4	13	5	3	3	12	7
3	13	4	3	6	4	7	9	4	13	5	3	3	12	7
4	13	4	3	6	4	7	9	4	13	5	3	3	12	7
5	13	4	3	6	4	7	9	4	10	5	3	3	12	7
6	13	4	3	6	4	7	9	4	10	5	3	3	12	7
7	14	5	4	5	3	6	10	5	12	5	3	3	11	6
8	14	5	4	5	3	6	10	5	12	5	3	3	11	6
9	14	5	4	5	3	6	10	5	10	5	3	3	11	6
10	14	5	4	5	3	6	10	5	10	5	3	3	11	6
11	14	5	4	5	3	6	10	5	10	5	3	3	11	6
12	12	3	5	7	5	10	7	3	12	4	4	5	9	6
13	12	5	3	5	4	7	0	0	18	5	5	5	0	6



## 5.2. Comparison with Other Heuristics Algorithms' Performance

### 5.2.1. Parameter Setting for the Heuristics Algorithm

A series of parameters for the heuristics algorithms were determined experimentally through several tests. The population size and iterations had been tested in the previous literature [18]. In this paper, the population size was set as 50. The maximum evolution and mutation generation was set as 100 generations. Based on the computation results of several tests, the crossover probability of the population was set as 0.8. The mutation probability of the population was set as 0.2. The local search mutation probability was 0.1. The general GA and DE algorithms were designed based on the above algorithm without a local search strategy and double schedule strategy. The Memetic algorithm is designed by improved genetic algorithm with a local search strategy.

### 5.2.2. Experimental Results Analysis for the Heuristics Algorithms

Due to the randomness of the results, each heuristics algorithm proposed in the experimental is run 30 times. Three instances are proposed in the research. Instance 1: Five carrier aircrafts (Nos. 2, 3, 9, 10, 12),  $n = 5$ . Instance 2: Nine carrier aircrafts (Nos. 2, 3, 4, 5, 8, 9, 10, 11, 12),  $n = 9$ . Instance 3: 13 carrier aircrafts (Nos. 1–13),  $n = 13$ . The objects are the makespan (M), average results (A), and *variance*, which are calculated with different wave carrier aircrafts. The final convergence gen (CG) can be also obtained. The results of the three instances are shown in Table 3.

**Table 3.** Comparison of the algorithm performance. DE: different evaluation algorithm; GA: genetic algorithm.

<i>n</i>	Algorithm	<i>M</i> (min)	<i>A</i> (min)	<i>Variance</i>	<i>CG</i>
5	GA	43	44.74	2.3739	6
	DE	43	44.68	2.4215	7
	Memetic	43	44.22	2.1045	6
	DPGA	43	43.74	1.3673	5
9	GA	56	61.60	4.3494	61
	DE	56	61.35	4.2587	66
	Memetic	55	60.44	4.1078	35
	DPGA	54	59.33	3.825	21
13	GA	69	73.44	5.182	67
	DE	69	73.56	5.0236	72
	Memetic	68	72.98	4.9045	52
	DPGA	64	68.28	4.0213	40

In the case of the smaller scale for the aircraft first wave attack (Instance 1), the four algorithms can achieve the same optimal results. When the scale increases, the difference among the four algorithms' performance increases. The advantages of DPGA are highlighted. The evolutionary convergence curves of the four algorithms for 13 carrier aircrafts and nine carrier aircrafts can be achieved in Figure 6.

From a comparative analysis of the results from Table 2 and Figure 6, it can be seen that the proposed DPGA algorithm not only has strong global searching ability, it also has a more stable and faster convergence speed to global optimal solutions. The local search strategies integrated in the Memetic algorithm can improve the diversity of the population and increase the search performance of the optimal results. When the number of carrier aircrafts increases, the results showed that the global optimal scheduling can be obtained faster and better by adopting the bidirectional schedule strategy in the DPGA algorithm by adaptively adjusting the crossover and mutation parameters.

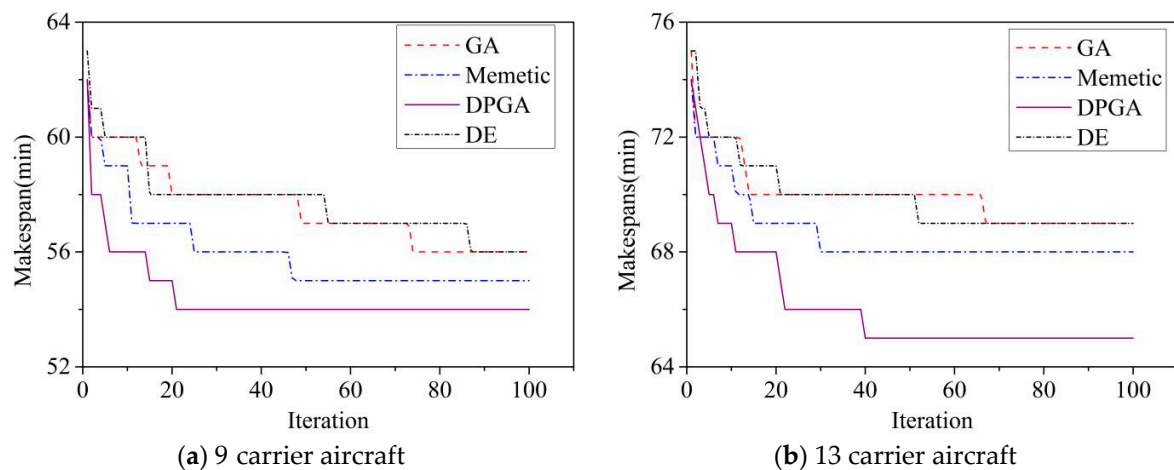


Figure 6. The convergence curves of the four algorithms.

### 5.2.3. Scheduling and Resource Allocation without RH Strategy

In the flight deck, the commanders most concerned with the operability and optimality of the schedule. The static schedules are calculated by the DPGA algorithm to obtain the support resource allocation without an RH strategy. The static scheduling Gantt charts of the operational crews and resource stations in Instance 3 are shown in Figures 7 and 8. The vertical axis represents the number of the supporting operational crew and resource station.  $L_{pk}^l$  represents the  $l$ th operational crew of the  $k$ th major,  $L_{rk}^l$  represents the  $l$ th support equipment of the  $k$ th resource station, and  $I_i - j$  represents the  $i$ th carrier aircraft number for the  $j$ th activity.

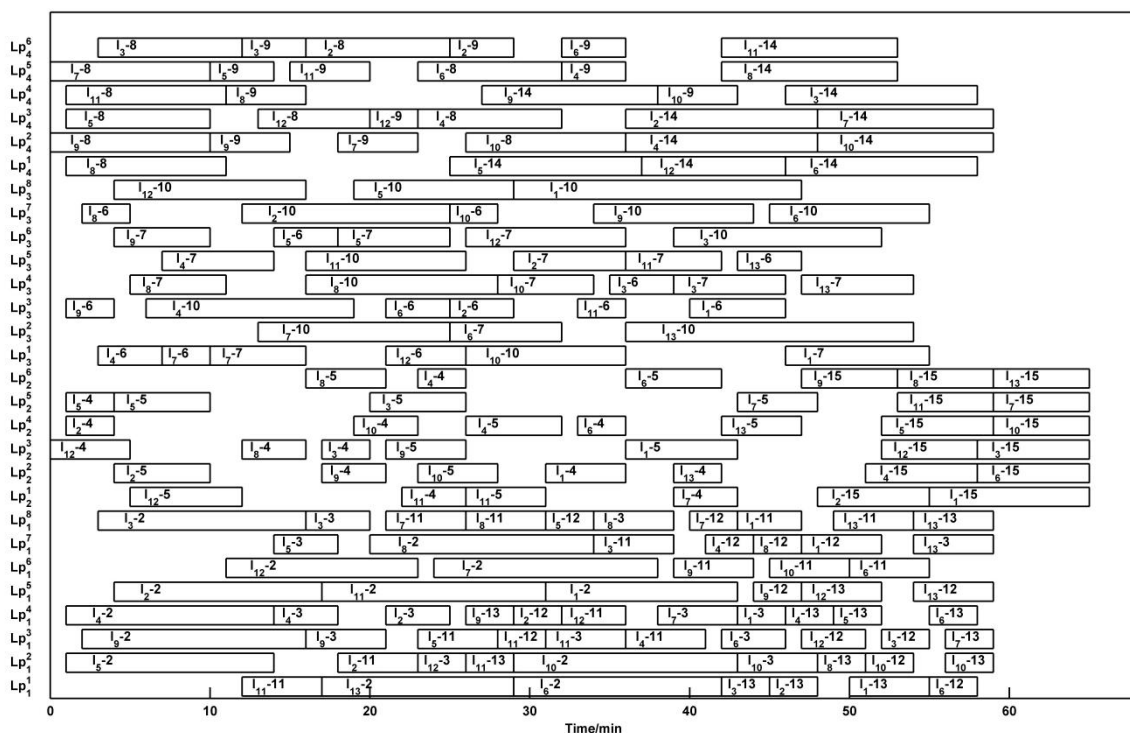
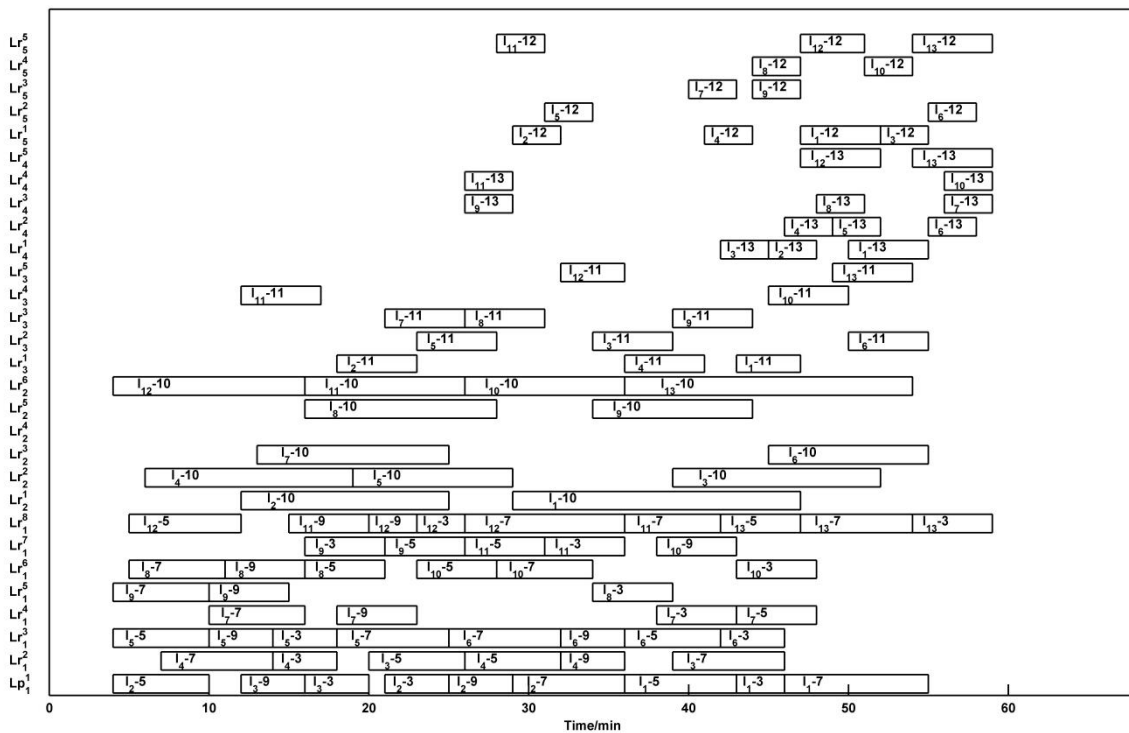


Figure 7. Gantt chart of optimal support personnel allocation in the static scheduling.



**Figure 8.** Gantt chart of optimal support equipment allocation in the static scheduling.

### 5.3. Dynamic Scheduling Parameters Analysis of Periodic RH Strategy

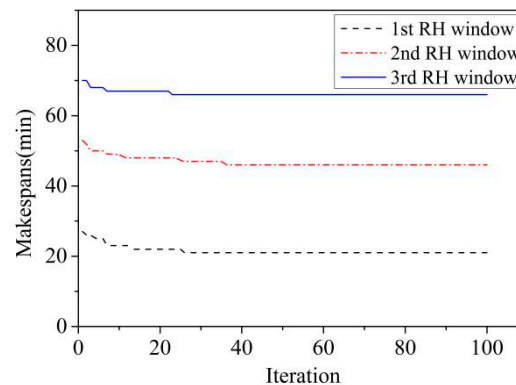
In Section 3, the size of the RH window was discussed, which was closely related to the carrier aircraft scheduling problems, and had a great influence on the RH scheduling results. It was necessary to select the appropriate size of the RH window before the rolling mechanism was triggered. In order to get a better window size, we selected a different size of the time window for Instance 3. Each size of the time window in the periodic RH scheduling strategy was calculated 10 times in each experiment. We then recorded the window size (WS), roll times (R), makespan (M), each window's computational time (WT), the total computational time (TT), the average computational time (AT), and finally, each window's convergence gen (CG). Table 4 shows the optimum computational results.

**Table 4.** Results of window size analysis. WT: each window's computational time; TT: the total computational time; AT: the average computational time; CG: each window's convergence gen.

WS (min)	R	M (min)	WT (s)	TT (s)	AT (s)	CG
15	5	77	12.92, 25.69, 20.98, 31.95, 2.37	93.91	18.78	1, 1, 1, 1, 1
20	4	72	24.19, 36.22, 41.65, 2.25	104.31	26.08	2, 5, 7, 10
25	3	66	29.78, 49.56, 22.10	101.44	33.81	26, 35, 23
30	3	67	46.08, 60.94, 2.14	109.16	36.39	19, 30, 3
35	2	66	54.97, 51.92	106.89	53.45	42, 62

Table 4 shows the relationships between the window size and the computational results. When the time of the WS is too little, R increases. In each RH window, only a small portion of the activities are considered to get the local optimal scheduling. When the WS was set as 15 min, the convergence results can be obtained in the first gen because of its small computation scales. The final results considered less global optimal scheduling. When the WS increases, the number of activities involved in the waiting scheduling window increases, and the computational scale and time increased. A large rolling WS can reduce the number of rolling windows. An excessive number of activities in a window could increase the computation time of a single window schedule, and the final convergence gen

also appeared later, but there was not an obvious variation in the total computational time. In order to achieve a more efficient dynamic schedule, the WS as 25 min was selected to balance the rolling times ( $R$ ) and each window's computational time ( $WT$ ). Figure 9 shows the convergence curves in the different RH windows.



**Figure 9.** The convergence curves of the dual population genetic algorithm (DPGA) in different RH windows.

Figure 9 shows that the periodic RH strategy can make the calculation easier to converge, because the number of waiting scheduling activities in a single RH window is reduced, which is less than the static scheduling. Different numbers of carrier aircraft were set to make a comparison and obtain the support operational results under the same conditions by the DPGA algorithm. Table 5 shows the comparison results with and without periodic RH scheduling for different waves of carrier aircraft.

**Table 5.** Comparison with and without RH scheduling for different waves of carrier aircraft.

$n$	No Rolling			Rolling				
	$M$ (min)	$TT$ (s)	$CG$	$M$ (min)	$TT$ (s)	$WT$ (s)	$AT$ (s)	$CG$
5	43	57.12	18	48	36.42	18.25, 18.17	18.21	14, 5
9	54	106.75	21	55	71.45	26.30, 42.85, 2.30	23.82	31, 65, 1
13	65	171.55	40	66	101.44	29.78, 49.56, 22.10	33.81	26, 35, 23

The experimental results in Table 5 show that the total computational time ( $TT$ ) of the periodic RH dynamic scheduling is a little shorter than the static scheduling results. However, the computational time of each RH window ( $WT$ ) decreased greatly. The reduced computational time makes the scheduling easier and more able to cope with the dynamic environment. It was found that the total computational time ( $TT$ ), each window's computational time ( $WT$ ), and finally the convergence gen ( $CG$ ) of the three experiments obtained from the proposed approach were better than those obtained from the static rolling horizon approach. The global optimal scheduling may be lost by the RH dynamic scheduling strategy. Moreover, the makespan and computational time of the proposed approach was acceptable. The results show that the periodic RH dynamic scheduling strategy is superior for solving large-scale carrier aircraft support scheduling problems, and the RH dynamic scheduling strategy, which closely approached an actual situation, was proven to be suitable for carrier aircraft scheduling.

#### 5.4. Results for the Disruptions by Event-Driven RH Scheduling Strategy

In this section, the event-driven RH scheduling strategy was selected to generate a recovery scheduling for the disruptions. The size of the recovery time window was also set as 25 min. The DPGA algorithm was used to reoptimize the recovery schedule in each RH window. Every time a disruption

occurred, the RH window was selected again, and generated a recovery scheduling by the DPGA algorithm with the same computer configuration and similar parameters. If another disruption occurred in the recovery time window, the recovery scheduling should be revised again.

For a single disruption of the second carrier aircraft, the eighth activity was chosen randomly to analyze the possible impacts of the disruption. The disruption made the resource unavailable due to resource breakdown or the operators reducing in one operational crew, and prolonged the support time of the disruption activity to 19 min. The event-driven RH scheduling strategy was triggered. The DPGA algorithm was used to calculate the recovery makespan and obtain the new resource allocation. The Gantt chart of the operational crews by periodic RH scheduling strategy without disruption was shown in Figure 10. The recovery scheduling of the operational crews for the disruption by event-driven RH scheduling strategy is shown in Figure 11.

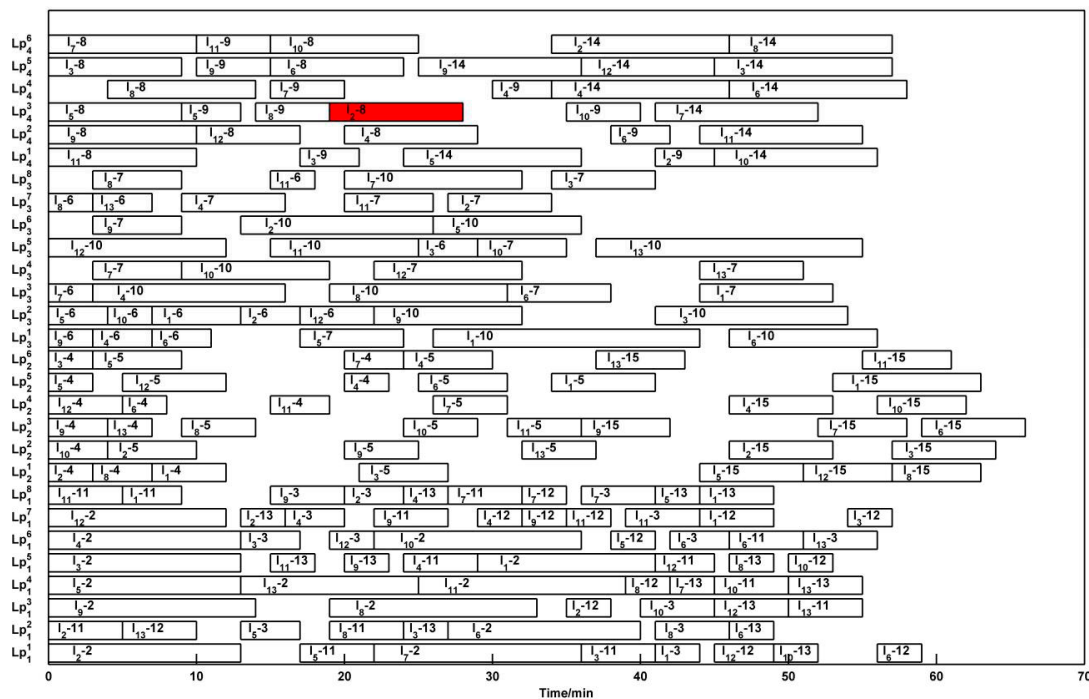


Figure 10. Gantt chart of optimal support personnel allocation by periodic RH scheduling strategy.

A comparison between Figures 10 and 11 show that the priorities of all of the activities are satisfied. The event-driven RH scheduling strategy can generate a recovery scheduling by the DPGA algorithm. When a disruption occurred, another method can be selected to obtain a complete rescheduling result by the reactive rescheduling strategy. In order to analyze the performance of the event-driven RH scheduling strategy, a comparison between the event-driven RH scheduling strategy and the reactive rescheduling strategy was made. Three kinds of disruptions instances were set. Instance 1: A single disruption (the support time of the eighth activity of the second carrier aircraft is prolonged to 19 min). Instance 2: Two disruptions (the support time of the eighth activity of the second carrier aircraft is prolonged to 19 min, and the support time of the 14th activity of the ninth carrier aircraft is prolonged to 15 min with the increase of weapon number). Instance 3: A series of disruptions make the inertial alignment support time of all of the aircraft increase 3 min because of the flight deck environment. Table 6 shows the comparison of makespan ( $M$ ), total computation time ( $TT$ ), and completion time cost ( $\delta$ ) calculated by different scheduling strategies.

In Table 6, the reactive rescheduling strategy can also get rescheduled. It considered more regarding the global optimization, but also increased the computation time and obtained more complexity in the activity variation. The event-driven RH rescheduling result could obtain a smaller rescheduling

makespan, and the new strategy gave more consideration to the earliness and lateness penalty of the supporting activities. Comparative analysis proved that the event-driven RH rescheduling strategy could obtain a better balance between the computation and the scheduling objectives. It could also obtain a better response to different disruption conditions (e.g., machine breakdown, operator change, dynamic deck environment) and reduce the computation time for recovering the schedules in the event of disruptions illustrated by numerical experiments. The Gantt chart of the scheduling activities by RH recovery scheduling strategy can provide a convenient guidance to the carrier aircraft support operations, and make the rescheduling more suitable for the dynamic operational environment of the carrier flight deck.

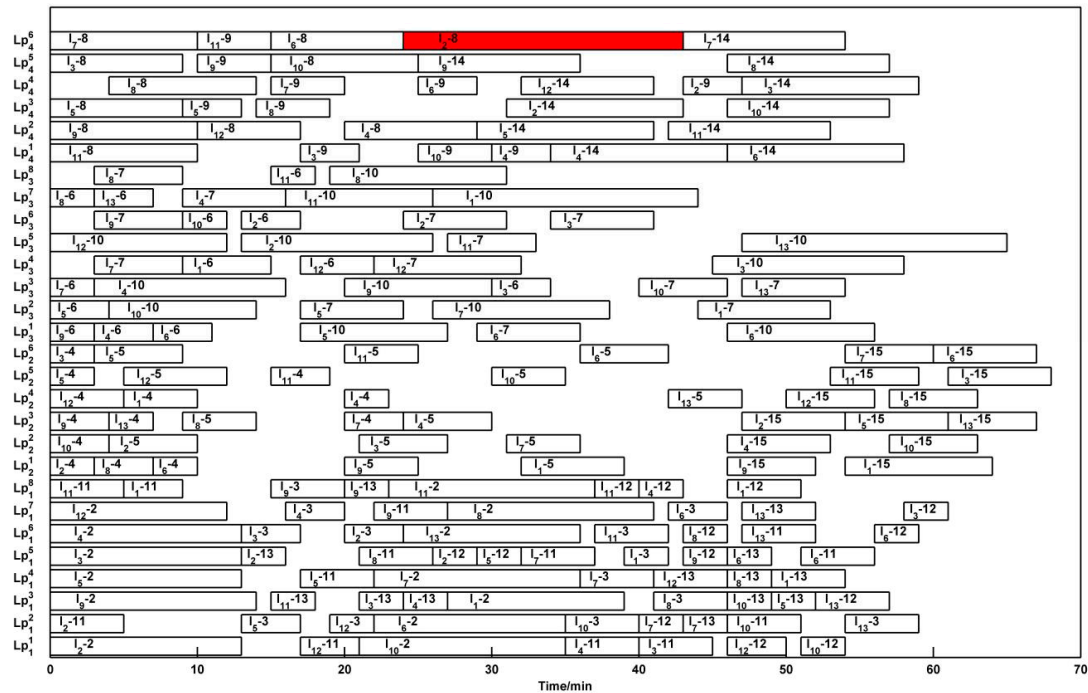


Figure 11. Gantt chart of optimal support personnel allocation by event-driven RH scheduling strategy.

Table 6. Comparison results of different scheduling strategies.

Disruption Instance	<i>n</i>	RH Scheduling			Reactive Rescheduling		
		<i>M</i> (min)	<i>TT</i> (s)	$\delta$	<i>M</i> (min)	<i>TT</i> (s)	$\delta$
A single disruption	5	48	37.33	79	43	46.81	525
	9	55	42.16	229	55	73.10	1057
	13	68	77.07	504	68	127.28	2419
Two disruptions	5	48	20.19	34	45	40.44	566
	9	56	42.06	221	56	70.825	1186
	13	69	79.53	512	68	128.67	2405
A series of disruptions	5	51	19.28	42	48	32.84	526
	9	60	45.17	256	60	71.52	1001
	13	70	30.09	310	71	129.91	2402

## 6. Conclusions

The purpose of this paper was to get a real-time disruption recovery solution to deal with the dynamic scheduling problem for carrier aircraft in the dynamic and complex flight deck environment. The support operation for carrier aircraft is considered as MRCMPSP. An integer programming model was constructed by considering the independent operational time and resource constraint of each



activity. A periodic and event-driven RH scheduling strategy combined with multiple heuristic algorithms (GA, DE, Memetic, DPGA) was presented to solve the MRCMPSP. The approach developed in this paper can enable better decisions for real-time disruption recovery scheduling.

The RH scheduling strategy inspired by RH predictive control technology for a continuous manufacture system was presented to adapt the dynamic flight deck environment. The appropriate size of the RH window could balance the rolling times and computational time. The RH dynamic scheduling method, which was better than the reactive rescheduling, not only reduced the problem size, it could also effectively adjust the baseline scheduling in a reasonable computational time and avoid the unnecessary scheduling with effective resource allocation.

A DPGA algorithm was proposed. The activity list encoding method was set in the chromosome with a two-dimensional matrix. A new adaptive crossover and mutation strategy based on the subtopology structure was designed to improve the global search ability. A double schedule for the leftward and rightward populations was integrated with alternating iterations to improve the convergence speed and obtain a more efficient solution in the search space. The computational results have shown that our approach was effective in solving the scheduling problem. The method in this paper can also be extended to other RCPSP such as aircraft, ship, and job shops. In the future, we can make an important extension regarding more developed dynamic scheduling methods for large projects and multimodal research. Studies may further consider more multi-objective optimization methods when different disruptions occur within carrier aircraft scheduling problems.

**Author Contributions:** P.Y. contributed significantly to establish an integer programming mathematical model and design a periodic and event-driven rolling horizon scheduling strategy for the carrier aircraft dynamic scheduling. W.H. contributed to determine the topic of this paper and analyze the results of the experiments; X.S. contributed to propose the solution for the carrier aircraft scheduling problems and design the dual population genetic algorithm. J.L. contributed to collect the data and improve the writing of paper. J.S. contributed to provide the background of this study and search the literature.

**Funding:** This research was funded by the National Natural Science Foundation of China NO. 51375490.

**Acknowledgments:** The authors highly appreciate the valuable comments provided by the reviewers, whose comments contributed significantly to improve the paper.

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

## References

1. Kelly, J.E., Jr. The critical path method, resource planning and scheduling. In *Industrial Scheduling*; John, F., Gerald, L.T., Eds.; Prentice Hall: Upper Saddle River, NJ, USA, 1963; pp. 347–365.
2. Hartmann, S.N.; Briskorn, D. A survey of variants and extensions of the resource-constrained project scheduling problem. *Eur. J. Oper. Res.* **2010**, *207*, 1–14. [[CrossRef](#)]
3. Blazewicz, J.; Lenstra, J.K.; Kan, A.R. Scheduling subject to resource constraints: Classification and complexity. *Discret. Appl. Math.* **1983**, *5*, 11–24. [[CrossRef](#)]
4. Zhu, G.; Bard, J.F.; Yu, G.A. branch-and-cut procedure for the multimode resource-constrained project-scheduling problem. *Inf. J. Comput.* **2006**, *18*, 377–390. [[CrossRef](#)]
5. Herroelen, W.; Leus, R. Project scheduling under uncertainty: Survey and research potentials. *Eur. J. Oper. Res.* **2005**, *165*, 289–306. [[CrossRef](#)]
6. Zapata, J.C.; Hodge, B.M.; Reklaitis, G.V. The multimode resource constrained multiproject scheduling problem: Alternative formulations. *AIChE J.* **2008**, *54*, 2101–2119. [[CrossRef](#)]
7. Kyriakidis, T.S.; Kopanos, G.M.; Georgiadis, M.C. Milp formulations for single-and multi-mode resource-constrained project scheduling problems. *Comput. Chem. Eng.* **2012**, *36*, 369–385. [[CrossRef](#)]
8. Lamas, P.; Demeulemeester, E. A purely proactive scheduling procedure for the resource-constrained project scheduling problem with stochastic activity durations. *J. Sched.* **2016**, *19*, 409–428. [[CrossRef](#)]
9. Xiong, J.; Liu, J.; Chen, Y.; Abbass, H.A. A knowledge-based evolutionary multiobjective approach for stochastic extended resource investment project scheduling problems. *IEEE Trans. Evol. Comput.* **2014**, *18*, 742–763. [[CrossRef](#)]

10. Wang, W.-X.; Wang, X.; Ge, X.-L.; Deng, L. Multi-objective optimization model for multi-project scheduling on critical chain. *Adv. Eng. Softw.* **2014**, *68*, 33–39. [[CrossRef](#)]
11. Johnston, J.S. A Feasibility Study of a Persistent Monitoring System for the Flight Deck of U.S. Navy Aircraft Carrier. Master's Thesis, Air Force Institute of Technology-Graduate School of Engineering & Management, Dayton, OH, USA, 2009.
12. Johnston, J.S.; Swenson, E.D. Feasibility study of global-positioning-system-based aircraft-carrier flight-deck persistent monitoring system. *J. Aircr.* **2010**, *47*, 1624–1635. [[CrossRef](#)]
13. Ryan, J.C.; Cummings, M.; Roy, N.; Banerjee, A.; Schulte, A. Designing an interactive local and global decision support system for aircraft carrier deck scheduling. In Proceedings of the Infotech@Aerospace Conferences and Exhibit 2011 (AIAA), St. Louis, MO, USA, 29–31 March 2011.
14. Ryan, J.C.; Banerjee, A.G.; Cummings, M.L.; Roy, N. Comparing the performance of expert user heuristics and an integer linear program in aircraft carrier deck operations. *IEEE Trans. Cybern.* **2014**, *44*, 761–773. [[CrossRef](#)] [[PubMed](#)]
15. Ryan, J.; Cummings, M.L. Development of an agent-based model for aircraft carrier flight deck operations. *M&S J.* **2014**, *2014*, 5–15.
16. Michini, B.; How, J.P. A human-interactive course of action planner for aircraft carrier deck operations. In Proceedings of the Infotech@Aerospace Conferences and Exhibit 2011 (AIAA), St. Louis, MO, USA, 29–31 March 2011.
17. Dastidar, R.G.; Frazzoli, E. A queueing network based approach to distributed aircraft carrier deck scheduling. In Proceedings of the Infotech@Aerospace Conferences and Exhibit 2011 (AIAA), St. Louis, MO, USA, 29–31 March 2011.
18. Zhang, Z.; Lin, S.; Dong, R.; Zhu, Q. Designing a human-computer cooperation decision planning system for aircraft carrier deck scheduling. In Proceedings of the Infotech@Aerospace Conferences and Exhibit 2015 (AIAA), Kissimmee, FL, USA, 5–9 January 2015.
19. Feng, Q.; Li, S.; Sun, B. A multi-agent based intelligent configuration method for aircraft fleet maintenance personnel. *Chin. J. Aeronaut.* **2014**, *27*, 280–290. [[CrossRef](#)]
20. Yu, L.; Zhu, C.; Shi, J.; Zhang, W. An extended flexible job shop scheduling model for flight deck scheduling with priority, parallel operations, and sequence flexibility. *Sci. Program.* **2017**, *2017*. [[CrossRef](#)]
21. Han, W.; Su, X.-C.; Chen, J.-F. Integrated maintenance support scheduling method of multi carrier aircrafts. *Syst. Eng. Electron* **2015**, *37*, 809–816.
22. Su, X.-C.; Han, W.; Xiao, W.; Jiang, T. Pit-stop support scheduling on deck of carrier plane based on memetic algorithm. *Syst. Eng. Electron* **2016**, *38*, 2303–2309.
23. Jiang, T.; Su, X.; Han, W. Optimization of support scheduling on deck of carrier aircraft based on improved differential evolution algorithm. In Proceedings of the 2017 3rd IEEE International Conference on Control Science and Systems Engineering (ICCSSE), Beijing, China, 17–19 August 2017; pp. 136–140.
24. Bennell, J.A.; Mesgarpour, M.; Potts, C.N. Dynamic scheduling of aircraft landings. *Eur. J. Oper. Res.* **2017**, *258*, 315–327. [[CrossRef](#)]
25. Moser, I.; Hendtlass, T. Solving dynamic single-runway aircraft landing problems with extremal optimisation. In Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Scheduling, Honolulu, HI, USA, 1–5 April 2007; pp. 206–211.
26. Rochlin, G.I.; La Porte, T.R.; Roberts, K.H. The self-designing high-reliability organization: Aircraft carrier flight operations at sea. *Nav. War Coll. Rev.* **1987**, *40*, 76–90.
27. Deblaere, F.; Demeulemeester, E.; Herroelen, W. Reactive scheduling in the multi-mode rcpsp. *Comput. Oper. Res.* **2011**, *38*, 63–74. [[CrossRef](#)]
28. Herroelen, W.; Leus, R. Robust and reactive project scheduling: A review and classification of procedures. *Int. J. Prod. Res.* **2004**, *42*, 1599–1620. [[CrossRef](#)]
29. Fu, N.; Lau, H.C.; Varakantham, P. Robust execution strategies for project scheduling with unreliable resources and stochastic durations. *J. Sched.* **2015**, *18*, 607–622. [[CrossRef](#)]
30. Qi, C.; Wang, D. Dynamic aircraft carrier flight deck task planning based on HTN. *IFAC-PapOnLine* **2016**, *49*, 1608–1613. [[CrossRef](#)]
31. Gürkan, H. An Integrated Approach for Robust Airline Scheduling, Aircraft Fleeting and Routing with Cruise Speed Control. Master's Thesis, Bilkent University, Ankara, Turkey, July 2014.

32. Kuster, J.; Jannach, D. Handling airport ground processes based on resource-constrained project scheduling. In Proceedings of the International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, Annecy, France, 27–30 June 2006; pp. 166–176.
33. Sprecher, A.; Drexel, A. Multi-mode resource-constrained project scheduling by a simple, general and powerful sequencing algorithm. *Eur. J. Oper. Res.* **1998**, *107*, 431–450. [[CrossRef](#)]
34. Kopanos, G.M.; Kyriakidis, T.S.; Georgiadis, M.C. New continuous-time and discrete-time mathematical formulations for resource-constrained project scheduling problems. *Comput. Chem. Eng.* **2014**, *68*, 96–106. [[CrossRef](#)]
35. Hartmann, S.n.; Drexel, A. Project scheduling with multiple modes: A comparison of exact algorithms. *Networks* **1998**, *32*, 283–297. [[CrossRef](#)]
36. Van Peteghem, V.; Vanhoucke, M. A genetic algorithm for the preemptive and non-preemptive multi-mode resource-constrained project scheduling problem. *Eur. J. Oper. Res.* **2010**, *201*, 409–418. [[CrossRef](#)]
37. Ozdamar, L. A genetic algorithm approach to a general category project scheduling problem. *IEEE Trans. Syst. Man Cybern. C* **1999**, *29*, 44–59. [[CrossRef](#)]
38. Damak, N.; Jarboui, B.; Siarry, P.; Loukil, T. Differential evolution for solving multi-mode resource-constrained project scheduling problems. *Comput. Oper. Res.* **2009**, *36*, 2653–2659. [[CrossRef](#)]
39. Fang, J.; Xi, Y. A rolling horizon job shop rescheduling strategy in the dynamic environment. *Int. J. Adv. Manuf. Technol.* **1997**, *13*, 227–232. [[CrossRef](#)]
40. Dishan, Q.; Chuan, H.; Jin, L.; Manhao, M. A dynamic scheduling method of earth-observing satellites by employing rolling horizon strategy. *Sci. World J.* **2013**, *2013*. [[CrossRef](#)] [[PubMed](#)]
41. Chang, D.; Jiang, Z.; Yan, W.; He, J. Developing a dynamic rolling-horizon decision strategy for yard crane scheduling. *Adv. Eng. Inform.* **2011**, *25*, 485–494. [[CrossRef](#)]
42. Chang, D.; Fang, T.; Fan, Y. Dynamic rolling strategy for multi-vessel quay crane scheduling. *Adv. Eng. Inform.* **2017**, *34*, 60–69. [[CrossRef](#)]
43. Park, T.; Ryu, K.R. A dual-population genetic algorithm for adaptive diversity control. *IEEE Trans. Evol. Comput.* **2010**, *14*, 865–884. [[CrossRef](#)]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).