



*Research article*

## **A multiple objective optimization model for aircraft arrival and departure scheduling on multiple runways**

**Ming Wei<sup>1,2</sup>, Bo Sun<sup>1,\*</sup>, Wei Wu<sup>1</sup> and Binbin Jing<sup>3</sup>**

<sup>1</sup> School of Air Traffic Management, Civil Aviation University of China, Tianjin 300300, China

<sup>2</sup> CAAC Key Laboratory of General Aviation Operation, Civil Aviation Management Institute of China, Beijing 102202, China

<sup>3</sup> School of Transportation, Nantong University, Nantong 226019, China

\* **Correspondence:** Email: bosun@cauc.edu.cn; Tel: +86-15022308929.

**Abstract:** This study proposes a multi-objective mixed integer linear programming (MOMILP) model for assigning a set of flights to different runways and determining their actual arrival and departure times. The proposed model envisages unique operation model of each runway (i.e., takeoff, landing, or mixed takeoff and landing). Further, interference in two flights between adjacent runways are also fully considered in this model. The work aims at reveal the optimal relationship between traffic stream characteristics, operation mode of each runway and flight scheduling to simultaneously minimizing flight delays and maximizing runway utilization. Since the problem of interest has a non-deterministic polynomial (NP-hard) complexity, a heuristic-based non-dominated sorting genetic algorithm (NSGA-II) is also presented to find Pareto-optimal solutions in a reasonable amount of time, where coding structure and heuristic algorithm for producing initial population are defined. Finally, a real-world example is provided to compare the difference in quality between the proposed and traditional models, and reveal changes in trends between delay time of flights and idle time of the runways, which can verify the correctness of the model.

**Keywords:** arrival and departure flight scheduling; multi-objective mixed integer linear programming; heuristic algorithm

---

### **1. Introduction**

Aircraft arrival and departure scheduling (AADS) aims at assigning a set of flights to different

runways and determining their actual arrival or departure times to improve operational efficiency. Due to different starting and occupation time of each flight, related with aircraft type (i.e., light, medium and heavy plane) and flight type (i.e., takeoff, landing), an unreasonable assignment of trips to runway may lead to idle runway and its queues for flights to take off or land at the same time [1]. Further, as the number of flights arriving exceeds the capacity of the airport's runway, it would cause an increase in delays. It is important to find optimal relation between the flight delay and improve runway utilization by reasonably scheduling flights of multiple runways [2–3]. Therefore, it has attracted wide attention from domestic and foreign scholars.

At present, most of the small airports have only one runway because of their low traffic flow, all planes of flights take off or land on the same runway [4]. Due to the origin or destination of each flight being different, some conflicts between their paths from the airport to in-out waypoints are existing, resulting in not only safety problems, but also efficiency problems. When the traffic flow increases, such problems of single-runway airport become more and more serious. Hence, some airports with large traffic flow have built and operated multiple runways for the landing and take-off of aircraft. In multiple runways of an airport, some runways are only used to taking off aircraft, part of them are dedicated to landing aircraft, to avoid the above phenomenon. According to the arrival rate of arrival and departure flights, how to determine the operation mode of each runway and its aircraft arrival and departure scheduling is a new research problem, which is much more complicated than traditional problems. However, few existing studies address this issue [2–5].

Another motivation for this paper is to study AADS with consideration of conflict between two aircrafts on adjacent runways. If the distance between two runways is short, they are not independent operation. When an aircraft takes off or lands on a runway, it makes some vortexes in aircraft wakes, which not only affects the takeoff or landing of the rear aircraft of the same runway, but also restricts the takeoff or landing of the aircraft of short-distance adjacent runway at same time. Compared with the traditional model, the study will reduce the capacity of the runway, but the operation is safe and practical. At present, domestic and foreign scholars also pay little attention to this issue [3,6–7].

The major contribution of this study is to propose a MOMILP model for AADS in order to reveal optimal relation between flight delays and runway utilization. The main contents in the paper are as follows: (1) Coordination of assignment of flights to runways and their arrival or departure time guidance process by considering type of each flight matching unique operation model of each runway and interference in two flights between adjacent runways; (2) a heuristic-based NSGA-II algorithm is developed by defining coding structure and heuristic algorithm for producing initial population to efficiently find a set of Pareto-optimal solutions. Finally, a numerical example is provided to illustrate the effectiveness of proposed model.

The rest of this study is described as follows. Section 2 introduces the latest literatures at home and abroad and summarizes their shortcomings. The research framework is described and the mathematical model is given in section 3. The detail of NSGA-II to resolving proposed model is given in section 4. A real numerical example is given to prove the applicability of this study in section 5. Some conclusions of my work, some critical remarks and possible future developments are discussed in section 6.

## 2. Review of literature

In the last decades, the AADS is a well-established and significant study problem of assigning

all flights to different runways in the landing and taking-off procedures [5]. There are many variants of the AADS, involving different objectives and constraints, include: cost, delay and fairness. However, those differences in variants of the AADS don't usually change the properties of the AADS, which lead to the use of existing formulations and solution methods to study them. Bianco et al. [6] has proved that the AADS with consideration of delay penalty costs being aircraft-dependent is a NP-hard problem. Further, Briskorn and Stolletz [7] has also proved that the problem is polynomial in the number of runway operations but exponential in the number of runways and operation classes under these assumptions of delay penalty costs being operation class-dependent.

In the early days, most airports had only one runway. The AADS with single runway (AADS-SR) was proposed by Dear [8] and Psaraftis [9], where few factors lead to that the research has been relatively mature. With the popularity of multi-runway airports in recent years, AADS-SR was then extended by Hansen [10], Pinol and Beasley [11], Salehipour et al. [12], and Liu [13] to consider multiple runways (AADS-MR). In AADS-MR, most of them has assumed independent runways to neglect runway-dependent earliest operation times, and few of them has considered interdependent runways by repeating some operation times for all separated pairs of runways [14–16]. However, none of the studies have considered interdependent runways in their numerical studies. These models were the static or the dynamic ones. In the static AADS, landing/departing flights are assigned to different runways, and sequenced by their start times in advance. In general, some mixed-integer models were developed for dealing with two cases of AADS-SR and AADS-MR, and their solutions were then solved using exact and heuristic algorithms [10,17–18]. In the dynamic AADS, an uncompleted set of landing/departing flights must be reassigned and sequenced using the First in First Out (FIFO) rule when a future incoming aircraft enter the system at every time [6,17–18]. All above studies adopted one of accurate formulations of the AADS, i.e., job shop scheduling model [16,21–22] and the alternative graph model [6]. When the AADS is generally viewed as a job shop scheduling plus additional constraints, each runway represents a machine and each flight denotes a job. Differently from AADS based on job shop scheduling, the microscopic representation of the alternative graph has taken air traffic regulations into account [3].

**Table 1.** Summary of relevant references.

Reference	Number of runways		Operation model		Operation environment	
	AADS-SR	AADS-MR	Interdependent	Independent	Static	Dynamic
[8]	□			□	□	
[9]		□		□	□	
[10–13]		□		□		
[14–16]		□	□		□	
[6,17–18]		□		□	□	□

Their solution methods could be fallen into three categories, i.e., mixed-integer programming (MIP), dynamic programming (DP), and branch-&-price (BP). The MIP approaches aims at building a mathematical optimization model for the AADS, which can be solved by using a standard solver, such as CPLEX. The most cited MIP model was proposed by Beasley et al. [18], and reformulated by, e.g., Farhadi et al. [21] and Ghoniem and Farhadi [23–24] to tighten these models. However, this method cannot yield the solutions for big problem instances quickly. DP approaches have been used by, e.g., Balakrishnan and Chandran [25], Harikiopoulo and Neogi [26], Furini et al. [22], Lieder et

al. [27] to deal with the AADS in fast computation times. All of them only study a DP approach for the time-indexed AADS-SR, except for AADS-MR of Lieder et al. [27]. However, they must follow some additional assumptions which is done to simplify the underlying models. In BP approaches, the underlying problem is divided into several subproblems which use single-runway problems as subproblems to generate columns [24]. Hence, this approach cannot be used to solve AADS with interdependent runways. Compared with BP approaches, DP approaches can offer a better chance for generalization of AADS with more new constraints such as general runway configurations.

A review of literature on AADS revealed the following critical issues, which deserve further investigations:

(1) Most studies have assumed that all operations of aircraft take-off or landing can be executed on all runways. From the safety perspective of aircraft entry and exit procedures in AADS-MR, some runways can be specially used for aircraft taking off or landing, and some runways can be used for both taking off and landing [2–3].

(2) Most studies have neglected heterogeneous or interdependent runways in AADS-MR [10-13,28]. In this case, there are at least a minimum diagonal separation delay between adjacent aircrafts on two-runway systems with interdependent runways.

(3) In AADS, flight delays minimization and runway utilization maximization are two favorable objectives. Since AADS is known as an NP-hard problem, a multiple-objective heuristic algorithm should be used to efficiently find a set of Pareto solutions to reveal optimal relation between them [29–30].

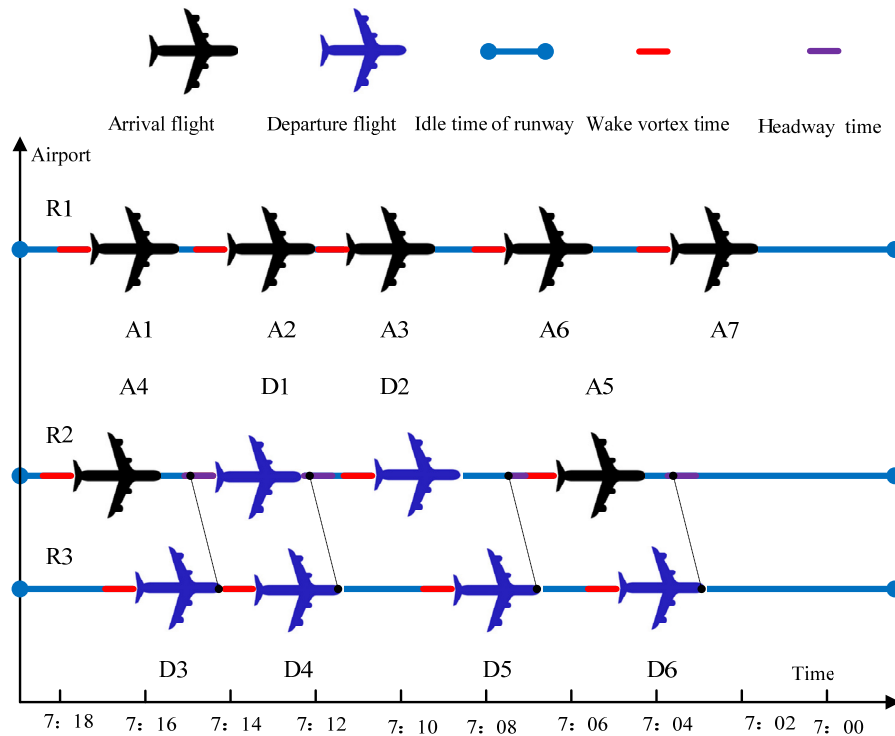
### 3. Methodology

#### 3.1. Research framework

An airport has multiple runways, where operation model of each runway is unique, i.e., one of three operation models, including takeoff, landing, or mixed takeoff and landing runway. There are some arrival and departure flights assigned to these different runways. Obviously, arrival flights must be assigned to one of landing and mixed runway, while departure flights should only be assigned to takeoff or mixed runway. Each flight has a planned starting and ending time occupying the runway, depending on the type of aircraft and the type of flight. If two adjacent flights are served by the same runway, the ending time of the previous flight plus wake vortex time of the front aircraft is less than the starting time of the next flight. Wake vortex time is decided by aircraft type of each flight and the type of takeoff or landing for these two flights. Further, if two flights from these runways take off or land at the same time, they affect each other, when the distance between adjacent runways is small. In this case, headway time between two aircraft must be greater than interval for the interference in these two runways. The main aim of this study is to assign a set of flights to different runways, route each runway to allow each flight to take off or land in sequence, and determining their actual arrival and departure times. A MOMILP model for this study is presented to simultaneously minimize flight delays and maximize runway utilization. To deal with real-life situations, this study will follow some assumptions:

- (1) Each flight must be only assigned to a runway, and all of them cannot be modified.
- (2) The wake time affected by the front plane to the rear plane can be estimated.
- (3) The difference in arrival and departure flights only lies in their maximum delay time.

(4) The effect of the capacity of the approach point on arrival and departure flight scheduling is neglected.



**Figure 1.** Graphical representation of proposed model.

To clearly explain the underlying principles of the proposed model, a small example consisting of three runways (R1-R3), seven arrival flights (A1-A7), and six departure flights (D1-D6) is shown in Figure 1. The runway R1 is only used for landing of an aircraft, the runway R2 is used for both of takeoff and landing of an aircraft, and the runway R3 is only used for takeoff of an aircraft. There are three generated flight tasks that takes off or lands on their runways in the optimization process, i.e., [A7-A6-A3-A2-A1], [A5-D2-D1-A4], and [D6-D5-D4-D3]. For example, flight A5 is completed on runway R2 at 7:06. After one-minute wake vortex and idle time of the preceding aircraft (A5) on the same runway and one-minute headway time of another aircraft (D5) on an affected runway (R3), flight D2 is started at 7:09. Similarly, the starting times of two flights D1 and A4 are 7:12 and 7:16, and their ending times are 7:17 and 7:18. In this case, each objective function of all runway tasks can be calculated easily.

### 3.2. Model formulation

#### 3.2.1. Notations

To facilitate model presentation, all definitions and notations used hereafter are summarized in Table 2.

**Table 2.** Parameters and variables in the proposed model.

<b>Indices:</b>	
$i, j$	Flight index
0	A virtual flight index that represents the first or last flight of a runway
$k, r$	Runway index
<b>Sets:</b>	
$F$	Set of all flights, $F = F_1 \cup F_2$
$F_1$	Set of arrival flights
$F_2$	Set of departure flights
$R_1$	Set of takeoff runways
$R_2$	Set of landing runways
$R_3$	Set of mixed landing and takeoff runways
$R$	Set of all runways, $R = R_1 \cup R_2 \cup R_3$
<b>Parameters:</b>	
$t_i^p$	Scheduled arrival or departure time of flight $i$ ; $\forall i \in F$
$d_i$	Maximum delay time of flight $i$ , related to the type of flight (arrival or departure); $\forall i \in F$
$t_i^h$	Occupation time of flight $i$ on a runway, related to its aircraft type; $\forall i \in F$
$o_i$	Aircraft type of flight $i$ , involving light, medium and heavy aircraft; $\forall i \in F$
$u_i$	Type of flight $i$ , involving arrival or departure flight; $\forall i \in F$
$T_{ij}^1$	Wake vortex time of adjacent flights $i$ and $j$ , related to its aircraft type and flight type; $\forall i, j \in F$
$T_{ij}^2$	Safe span time of adjacent flights $i$ and $j$ covered by a controller, related to safe distance; $\forall i, j \in F$
$T_{kr}$	Affected time between two flights on adjacent runways $k$ and $r$ , related to distance between these two runways. If there is no effect, $T_{kr} = 0$ ; $\forall k, r \in R$
$M$	A larger fixed value
<b>Decision Variables:</b>	
$x_i^r = \begin{cases} 0 \\ 1 \end{cases}$	Whether the flight $i$ is covered by the runway $r$ , or not; $\forall r \in R, \forall i \in F$
$z_{ij}^r = \begin{cases} 0 \\ 1 \end{cases}$	Whether the flight $i$ precedes flight $j$ on the runway $r$ , or not; $\forall r \in R, \forall i, j \in F$
$t_i^a$	Actual arrival or departure time of flight $i$ ; $\forall i \in F$
$U_{ir} = \begin{cases} 0 \\ 1 \end{cases}$	An auxiliary (real) variable for sub-tour elimination constraint in the runway $r$ ; $\forall r \in R$

### 3.2.2. Formulation

The proposed problem can be formulated as the following MOMILP model, which requires minimization of:

$$\min Z_2 = \sum_{\forall i \in F} [t_i^a - t_i^p] \quad (1)$$

$$\min Z_1 = \sum_{\forall r \in R} \sum_{\forall i, j \in F} z_{ij}^r * [t_j^a - t_i^a - \max(T_{ij}^1, T_{ij}^2) - t_i^h] \quad (2)$$

which is subject to:

$$\sum_{\forall r \in R_2 \cup R_3} x_i^r = 1, \forall i \in F_1 \quad (3)$$

$$\sum_{\forall r \in R_1 \cup R_3} x_i^r = 1, \forall i \in F_2 \quad (4)$$

$$0 \leq t_i^a - t_i^p \leq d_i, \forall i \in F \quad (5)$$

$$2z_{ij}^r \leq x_i^r + x_j^r, \forall i, j \in F \quad \forall r \in R \quad (6)$$

$$\sum_{\forall j \in F \cup \{0\}} z_{ij}^r = \sum_{\forall j \in F \cup \{0\}} z_{ji}^r = x_i^r, \forall i \in F, \forall r \in R \quad (7)$$

$$U_{ir} - U_{jr} + |F \cup \{0\}| \cdot x_{ij}^r \geq |F \cup \{0\}| - 1, \forall i, j \in F, \forall r \in R \quad (8)$$

$$t_i^a + t_i^h + \max(T_{ij}^1, T_{ij}^2) + (1 - z_{ij}^r) \cdot M \leq t_j^a, \forall i, j \in F, \forall r \in R \quad (9)$$

$$T_{ij}^1 = G(o_i, u_i, o_j, u_j), \forall i, j \in F \quad (10)$$

$$T_{kr} \leq |t_i^a - t_j^a| + (1 - x_i^r) * M + (1 - x_j^k) * M, \forall i, j \in F, \forall r, k \in R \quad (11)$$

In this formulation, the objective function (1) aims at minimizing total delay time for all flights. The objective functions (2) aim at minimizing total idle time for all runways. Constraint (3) guarantees that each arrival flight would be assigned landing to or mixed runway. Constraint (4) is used to assign each departure flight to takeoff to or mixed runway. Constraint (5) guarantees that the actual arrival or departure time of a flight does not exceed the maximum delay time. Constraint (6) and (7) set all flights (except virtual flight) being served by each runway to have the same incoming and outgoing arcs. Constraint (8) is used for the sub-tour elimination of flight tasks in the runway. Constraints (9) guarantees ending time of the previous flight plus the minimum safety interval time is not earlier than starting time of next flight. Constraint (10) is used for calculating wake vortex time of adjacent flights. Constraint (11) is used for avoiding flight conflicts between adjacent runways.

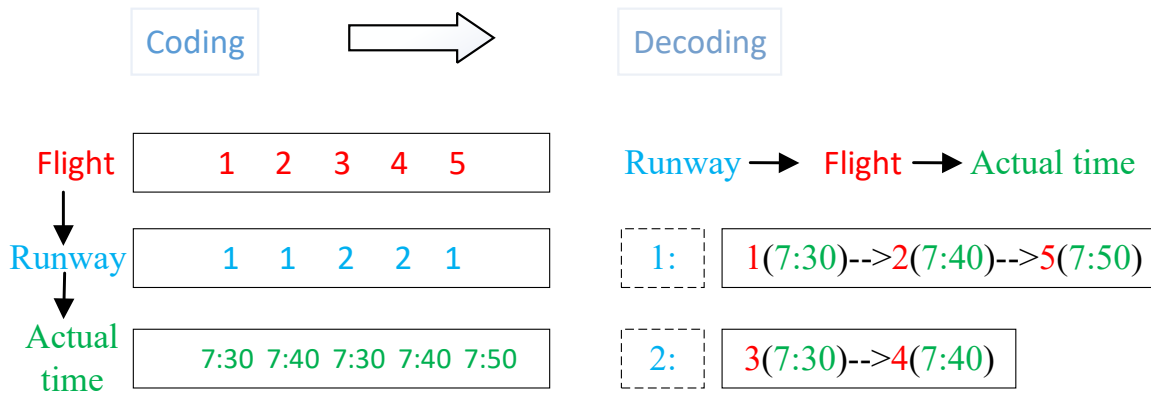
#### 4. Heuristic-based NSGA-II

Since this study is an NP-hard problem with two conflicting objectives, many evolutionary algorithms and metaheuristics [31–36] could be used to find a set of Pareto-optimal solutions in a short time, due to the exact algorithm not solving the large-scale problem quickly. The NSGA-II, as one of the most popular evolutionary multi-objective optimization algorithms, is a fast non-dominated sorting algorithms based on the NSGA, by introducing crowding of solutions for different targets, which improve the computational efficiency and robustness of the algorithm [29–30, 36–38]. In this study, a hybrid intelligent algorithm based on NSGA-II is designed to solve proposed model, which defines *chromosome code*, heuristic algorithm for generating initial population, etc. The details are described as follows.

##### 4.1. Chromosome coding

As above mentioned, the core of solving the problem lies in three variables  $x_i^r$ ,  $z_{ij}^r$  and  $t_i^a$ , where  $x_i^r$  and  $t_i^a$  determines  $z_{ij}^r$ . Using integer encoding, the two-dimensional vector  $U = (U_1, U_2)$  is used to represent each solution of the problem, where the element  $u_i$  of  $U_1 = (u_1, u_2, \dots, u_F)$  denotes assigned runway of flight  $i$ , ranging from 1 to  $R$ , the element  $u_i$  of  $U_2 =$

$(u_{F+1}, u_{F+2}, \dots, u_{2F})$  denotes actual arrival or departure time of flight  $i$ . For example, seen from Figure 3, one chromosome  $X = \{11221 \ 7:30 \ 7:40 \ 7:30 \ 7:40 \ 7:50\}$  of five flights and two runways can be decoded as: flights 1, 2 and 5 are assigned to runway 1 and their times are 7:30, 7:40 and 7:50; flights 3 and 4 are assigned to runway 2 and their times are 7:30 and 7:40.



**Figure 2.** An example of chromosome encoding.

#### 4.2. Heuristic algorithm for generating initial population

The quality of the initial population determines the solution quality. Due to the overlap of the time window of any two adjacent flights assigned to a runway, the probability of generating a feasible solution is low. Therefore, the heuristic algorithm is used to generate the feasible solution to create the initial population. The specific steps are as follows:

**Step 1:** All flights  $\forall i \in F$  are sorted according to  $t_i^p$ , and let the result be  $F'' = \{F_1'', F_2'', \dots, F_{|F|}''\}$ .

**Step 2:** Set the set of runways that could complete all flights  $F''$ . Let  $i = 1$ .

**Step 3:** Find set  $vR$  of feasible runways to complete flight  $F_i''$ . If  $vR \neq \emptyset$ , randomly select a runway  $\forall r \in vR$  to run flight  $F_i''$ , and turn to step4; otherwise, turn to step2.

**Step 3.1:** Let  $vR = \emptyset$ .

**Step 3.2:** Find any runway  $\forall r \in R$  that matches the flight type, and obtain its last flight  $lF(r)$ .

**Step 3.3:** Check whether the runway can continue to run flight  $F_i''$  after the completion of the flight  $lF(r)$ . If one of the following conditions is met: ①  $lF(r) = \emptyset$ , ②  $t_{lF(r)}^a + t_{lF(r)}^h +$

$\max(T_{lF(r)F_i''}^1, T_{lF(r)F_i''}^2) \leq t_{F_i''}^a$ , let  $vR = vR \cup \{r\}$ .

**Step 4:** Let  $i = i + 1$ , if  $i \leq |F|$ , turn to step3; otherwise, output the result and the algorithm is terminated.



### 4.3. Non-dominated sorting

#### 4.3.1. Non-dominated sorting

**Step 1:** Let the set and quantity of dominated individuals for a solution  $\exists x \in P$  be  $S_x = \emptyset$  and  $n_x = 0$ , respectively. For any solution  $\forall q \in P$ , if  $q \succ x$ ,  $S_x = S_x \cup \{q\}$ ; otherwise,  $n_x = n_x + 1$ .

**Step 2:** Let  $Q = \emptyset$ ,  $x_{\text{rank}} = 1$  and  $i = 1$ ,  $F_1 = \{x | n_x = 0, \forall x \in P\}$ .

**Step 3:** For  $\forall q \in S_x$  of  $\forall x \in F_i$ ,  $n_q = n_q - 1$ . If  $n_q = 0$ ,  $q_{\text{rank}} = i + 1$  and  $Q = Q \cup \{q\}$ .

**Step 4:** If  $Q \neq \emptyset$ ,  $i = i + 1$  and  $F_i = Q$ , turn to step 3; otherwise, output the result and the algorithm is terminated.

#### 4.3.2. Crowding distance

For each objective function, the average edge length  $\varepsilon_m(x_i)$  of individual  $x_i$  between adjacent individuals  $x_{i-1}$  and  $x_{i+1}$  is calculated, according to the order of the objective value of all individuals.

Crowded distance of individual  $x_i$  is calculated by using  $\text{Distance}(x_i) = \sum_{m=1}^M \varepsilon_m(x_i)$  to depict the intensity of surrounding other individuals, where  $M$  is the number of target functions.

#### 4.3.3. Selection operator

The aims of selecting individuals is to distribute Pareto solutions evenly by following two rules:

① If and only if  $i_{\text{rank}} < j_{\text{rank}}$  and  $i \succ_n j$ , ②  $i_{\text{rank}} = j_{\text{rank}}$  and  $\text{Distance}(x_i) > \text{Distance}(x_j)$ .

For any two individuals, if their non-dominant order is different, take the individuals with low order, otherwise an individual with a small crowding distance would be selected.

### 4.4. Algorithm process

The detail steps of NSGA-II are shown as follows:

**Step 1:** Set the algorithm parameters, including: chromosome number  $N$ , maximum number of iterations  $N_{\text{iter}}$ , crossover and mutation rate.

**Step 2:** Let  $t=0$ , and randomly generate an initial population  $P_0$ .

**Step 3:** Selection, variation and crossover operations of population  $R_t$  are carried out to produce the descendant population  $Q_t$ . Let  $R_t = P_t \cup Q_t$ , turn to the step 4.

**Step 4:** Calculate the nondominant order and crowding distance of  $2N$  individuals of  $R_t$ , and a total of  $N$  individuals would be selected to generate a new population  $P_{t+1}$  according to their ranks.

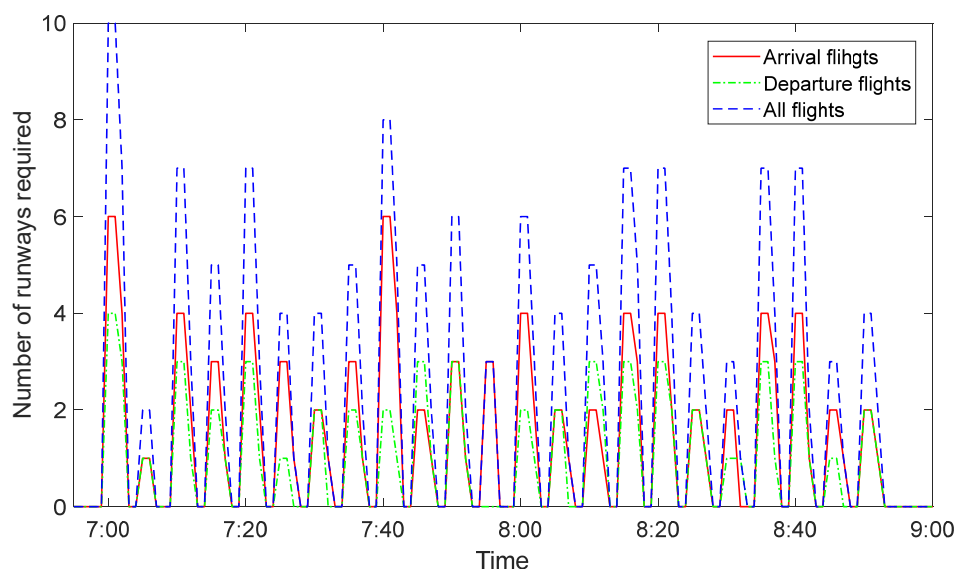
**Step 5:**  $t = t + 1$ . If  $t \geq N_{\text{iter}}$ , output the result and the algorithm is terminated; otherwise, turn to Step3.

## 5. Numerical example

### 5.1. Example description

In this section, a real-word example of flight scheduling in an airport in China during 7:00 and 9:00 is described to prove the applicability of the proposed model. There are 72 arrival flights and 51 departure flights during 7:00 and 9:00 in a day. Their aircraft types mainly involve light and medium - sized planes. According to the planned take-off or landing time of all flights, the number of runways required by these flights in different time periods is shown in Figure 3. Because the number of runways required in most of the time is greater than the actual number of runways at the airport, most flights must queue for takeoff or landing, which causes these flights to be delayed widely. In order to reduce the large area of flight delay, the aim of this study is to balance flight delays and runway utilization, by coordinating optimization process of the operation mode of each runway, assignment of flights to runways and their arrival and departure times. The solutions are obtained using NSGA-II. The key parameters used in the case study are given as follows:

- Maximum delay time of each flight:  $d_i=480\text{min}$ .
- Wake cortex time of adjacent flights:  $T_{ij}^1=1\text{min} (\forall i, j \in F)$
- Safe span time of adjacent flights covered by a controller:  $T_{ij}^2=2\text{min} (\forall i, j \in F)$
- Affected time between two flights on adjacent runways:  $T_{12} = 1\text{min}$ ,  $T_{13} = T_{23} = 0\text{min}$ .
- Related parameters of NSGA-II: maximum number of iterations 500, number of chromosomes 500, crossover probability 0.8, mutation probability 0.05 and shared radius 0.05.



**Figure 3.** Number of runways required for all flights at different time.

### 5.2. Results and discussion

One of the aims of the study is to reveal influence of different combination of runways operation modes on result. We analyze the difference between the traditional model and the present

model under different number of runways. The results are shown in Table 3, from which it can be seen that:

(1) With the increase in the number of runways, the delay time of flights is gradually decreased and the idle time of runways is also increased. When the number of runways is 1, the idle time of runways is 0 because the number of runways required in all time periods is less than the actual value. Similarly, when the number of runways is more than 1, it is greater than the number of runways required in some periods, resulting in runway idle phenomenon in some time periods.

(2) For the same number of runways, as the number of arrival flights is greater than the number of departure flights, the more the number of mixed and special landing runways, the less the delay time of the flight and the idle time of the runway (i.e., the runway utilization efficiency is high). Therefore, the operation mode of each runway should be reasonably determined by flight type and arrival rate of all flights to reduce their delays, excepting for considering flights conflicts. According to the flight type and space distribution of the airport, the operation mode of the runway should be reasonably determined to reduce the flight delay.

**Table 3.** Model performance of proposed AADS under different number of runways and their operation models.

Number of runways	Operation mode	Delay time of flights(min)	Idle time of runways(min)
1	Mixed landing and take-off	11602	0
	Take-off Mixed	7502	125
2	Landing Mixed	7479	147
	Mixed Mixed	6250	102
	Landing Landing Mixed	2813	96
3	Take-off Take-off Mixed	1858	79
	Take-off Landing Mixed	1144	62
	Mixed Mixed Mixed	851	59

Another aim of this study is to reveal influence of flight conflicts between adjacent runways on result. The difference in model performance between the traditional model and the present model is also analyzed under different number of runways. The results are shown in Table 4, from which it can be seen that:

(1) When there is only one runway, the results of these two models are consistent because there is no interference between adjacent runways.

(2) As the number of runways more than two increases, since the landing and taking-off procedures of all flights in a runway are not interfered with by other runways, the runway idle time of the traditional model is always kept at zero. Meanwhile, the increase in the number of runways has led to fewer queues for flights to take off or land, so delays of the traditional model have been gradually reduced.

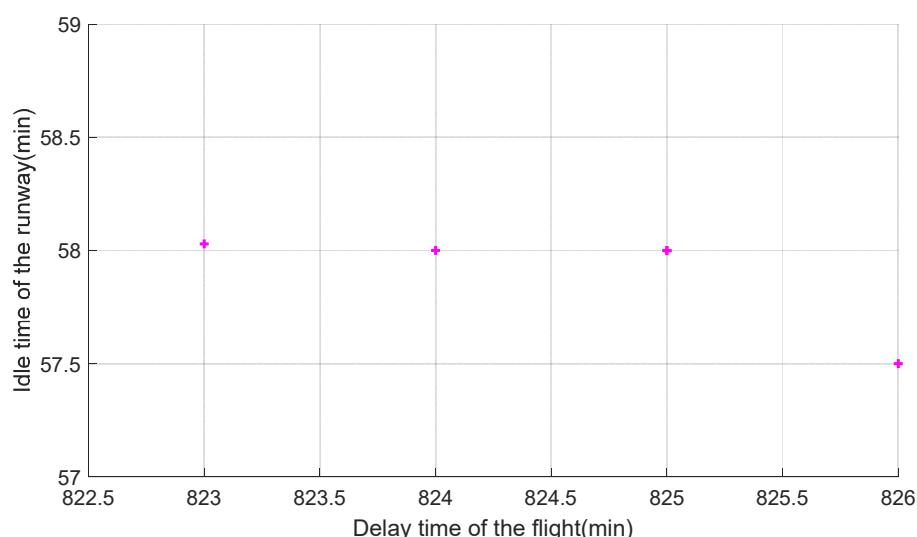
(3) As the number of runways more than two increases, since the runway idle time of the improved model contains interference time, it is not zero. In addition, an increase in the number of runways has led to a decrease in the number of flights allocated to the runways, which resulted in a gradual decrease in the runway idle time of the proposed model.

(4) When adjacent runways are close to each other, in order to ensure the safe operation of each

aircraft, disturbed runways require more idle time to avoid aircraft interference, which takes more time to complete the takeoff or landing of all flights, resulting in increased flight delays. Although the delay time of proposed model is more than traditional one, it is more in line with the actual control operation process.

**Table 4.** Result comparison of proposed and traditional AADS under different runway numbers.

Number of runways	AADS with consideration of interference in adjacent runways		AADS without consideration of interference in adjacent runways	
	Delay time of flights(min)	Idle time of runways(min)	Delay time of flights(min)	Idle time of runways(min)
1	11602	0	11602	0
2	6250	102	2447	0
3	851	59	452	33



**Figure 4.** Changing relationship between flight delay and runway utilization in the Pareto solution.

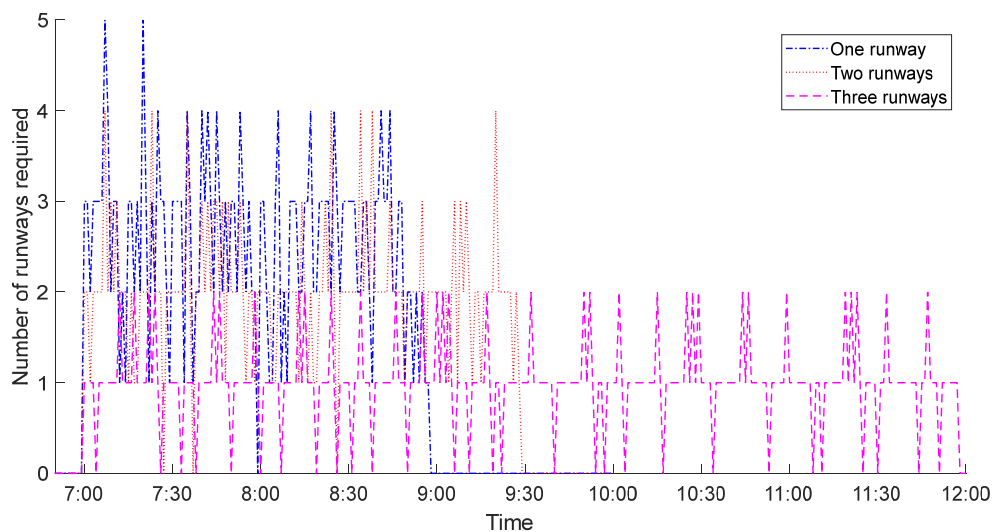
As above mentioned, the total delay times of all flights are the least, when all the runways are mixed takeoff and landing mode. Taking three mixed runways as an example, four Pareto solution is sought to reveal the trends in change relationship between flight delay time and runway idle time. As shown in Figure 4, the upper and lower limits of flight delay time are 823 min and 826 min, and the upper and lower limits of runway idle time are 57.5 min and 58.2 min. The reduced idle time of the runways will gradually increase the delay time of the flights. This is because the runway's idle time will be reduced due to the further shortage of runway resources. In this case, there may be more flights queuing for takeoff or landing. The calculation results are in line with the intuitive analysis.

Figures 5 and 6 shows a comparative analysis of the number of runways occupied and delay time of all flights under different number of runways, respectively. The result shows:

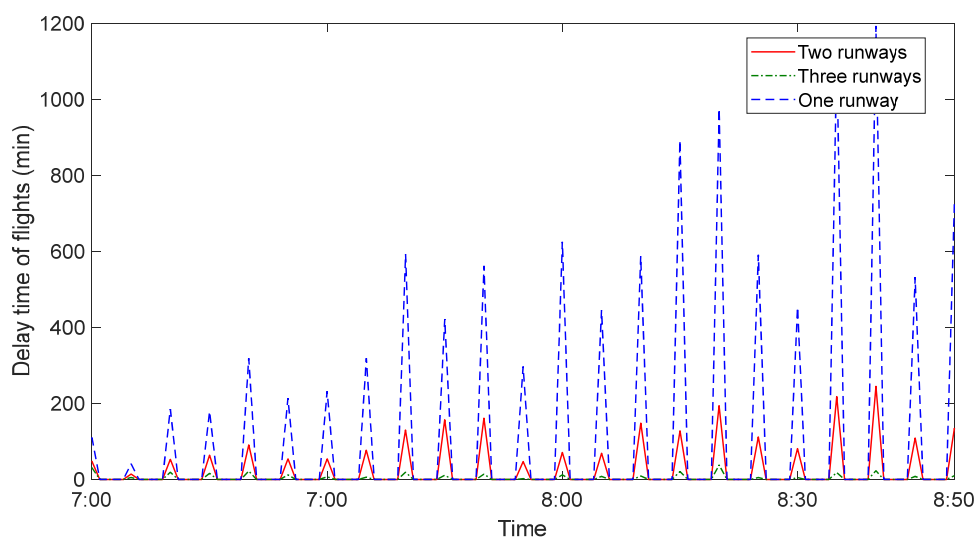
(1) The number of runways required at different time periods is different. They are insufficient in some time periods, but wasted in other time periods. It is necessary to comprehensively optimize the flight schedule to make them take off or land evenly in each time period to reduce the delays.

(2) As the number of runways increases, the number of flights queuing for takeoff or landing decreases, so the total delay time of all flights is reduced;

(3) If the number of runways required is always greater than the actual number of runways at any time, the total delay time of flights will gradually spread as time goes on.



**Figure 5.** Comparison of the number of runways required in at any period under different number of runways.



**Figure 6.** Comparison of flight delay time in at any period under different number of runways.

## 6. Conclusions

In this paper, a novel MOMILP model is proposed for aircraft arrival and departure scheduling to reveal optimal relation between flight delays and runway utilization. Proposed model has features in different operating modes of multiple runways and flight conflicts between adjacent runways. Furthermore, wake vortex time and safe span time of a controller are incorporated into the proposed model in order to represent real-world conditions. Our model is infinitely close to reality, and the generated scheduling scheme can be applied in practice. A heuristic-based NSGA-II is also designed to find a set of the meta-optimal solutions in a short time. The main findings can be summarized as follows:

(1) For the same number of runways, the reduction in total idle time of the runways will lead to an increase in total delay time of the flights.

(2) As the number of runways increases, the number of flights queuing for takeoff or landing decreases, which lead to increase of total idle time of runways and reduction in the total delay time of flights.

(3) If the operation modes of multiples runways should not match the spatial and temporal distribution of arrival and departure flights, the demand for some arrival or departure flights requesting for runways for taking off or landing aircrafts has not been met.

(4) When some flight conflicts between adjacent runways are considered, runway capacity would be decreased and flight delays would be also increased.

It can be seen from the above that our conclusion is in line with the value analysis, which is of great help to improve the operation efficiency of the runway. Note that the premise of this study is to a given fixed route from the runway to the airway point/gate, and ignores the influence of their route design on model performance. On the one hand, route design from the gate to a runway determines time of the plane arriving at the runway, which may cause planes to line up on the runway for takeoff or landing; on the other hand, route design from the runway to the airway point determines time of the plane passing through the airway point, which may cause the aircraft to hover in the air waiting for takeoff or landing. In this sense, the integrated operation of this study and routes design pay an important role in reducing flight delays and improving airport capacity. As a result, extending this model to simultaneously select routes from the runway to the airway point/gate is worth further study.

## Acknowledgements

This study was financially supported by the Humanities and Social Sciences Foundation of the Ministry of Education of China (20YJCZH176); the Central College Basic Scientific Research Operating Expenses Fund of the Civil Aviation University of China (3122020079).

## Conflict of interests

The authors declared that they have no conflicts of interest to this work.

## References

1. B. Sun, M. Wei, W. Wu, B. B. Jing. A novel group decision making method for airport operational risk management, *Math. Biosci. Eng.*, **17** (2020), 2402–2417.

2. S. L. Wu, P. C. Chen, K. Y. Chang, C. C. Huang, Robust gain-scheduled control for vertical/short take-off and landing aircraft in hovering with time-varying mass and moment of inertia, *Proc. Inst. Mech. Eng. Part G.*, **222** (2008), 473–482.
3. H. Nazini, T. Sasikala, Simulating aircraft landing and takeoff scheduling in distributed framework environment using Hadoop file system, *Cluster. Comput.*, **22** (2019), 13463–13471.
4. Y. Ding, J. Valasek, Aircraft landing scheduling optimization for single runway noncontrolled airports: Static Case, *J. Guid. Control Dynam.*, **30** (2007), 252–255.
5. M. Ahmed, S. Alam, M. Barlow, A cooperative co-evolutionary optimization model for best-fit aircraft sequence and feasible runway configuration in a multi-runway airport, *Aerospace*, **5** (2018), 345–353.
6. L. Bianco, P. Dell’Olmo, S. Giordani, Scheduling models for air traffic control in terminal areas, *J. Scheduling*, **9** (2006), 180–197.
7. D. Briskorn, R. Stolletz, A dynamic programming approach for the aircraft landing problem with aircraft classes, *Eur. J. Oper. Res.*, **43** (2015), 61–69.
8. R. G. Dear, *The dynamic scheduling of aircraft in the near terminal area*, MIT Libraries, (1976).
9. H. N. Psaraftis, A dynamic programming approach for sequencing identical groups of jobs, *Oper. Res.*, **28** (1980), 1347–1359.
10. V. J. Hansen, Genetic search methods in air traffic control, *Comput. Oper. Res.*, **31** (2004), 445–459.
11. H. Pinol, J. E. Beasley, Scatter search and bionomic algorithms for the aircraft landing problem, *Eur. J. Oper. Res.*, **171** (2006), 439–462.
12. A. Salehipour, L. M. Naeni, H. Kazemipoor, Scheduling aircraft landings by applying a variable neighborhood descent algorithm: runway-dependent landing time case, *J. Appl. Oper. Res.*, **1** (2009), 39–49.
13. Y. H. Liu, A genetic local search algorithm with a threshold accepting mechanism for solving the runway dependent aircraft landing problem, *Optim. Lett.*, **5** (2011), 229–245.
14. G. Bencheikh, J. Boukachour, A. E. H. Alaoui, Improved ant colony algorithm to solve the aircraft landing problem, *Int. J. Comput. Theory Eng.*, **3** (2011), 224–233.
15. A. Salehipour, M. Modarres, L. Moslemi Naeni, An efficient hybrid meta-heuristic for aircraft landing problem, *Comput. Oper. Res.*, **40** (2013), 207–213.
16. A. Faye, Solving the aircraft landing problem with time discretization approach, *Eur. J. Oper. Res.*, **242** (2015), 1028–1038.
17. A. T. Ernst, M. Krishnamoorthy, R. H. Storer, Heuristic and exact algorithms for scheduling aircraft landings, *Networks*, **34** (1999), 229–241.
18. J. E. Beasley, M. Krishnamoorthy, Y. M. Sharaiha, D. Abramson, Scheduling aircraft landings – The static case, *Transport. Sci.*, **34** (2000), 180–197.
19. H. N. Psaraftis, A dynamic programming approach for sequencing groups of identical jobs, *Oper. Res.*, **28** (1980), 1347–1359.
20. C. S. Venkatakrishnan, A. Barnett, A. M. Odoni, Landings at Logan airport: Describing and increasing airport capacity, *Transport. Sci.*, **27** (1993), 211–227.
21. F. Farhadi, A. Ghoniem, M. Al-Salem, Runway capacity management – an empirical study with application to Doha international airport, *Transp. Res. Part E: Logist. Transp. Rev.*, **68** (2014), 53–63.
22. F. Furini, M. P. Kidd, C. A. Persiani, P. Toth, State space reduced dynamic programming for the aircraft sequencing problem with constrained position shifting, *Int. Symp. Comb. Optim. (ISCO)*, **2014** (2014), 267–279.

23. A. Ghoniem, F. Farhadi, M. Reihaneh, An accelerated branch-and-price algorithm for multiple-runway aircraft sequencing problems, *Eur. J. Oper. Res.*, **246** (2015), 34–43.
24. A. Ghoniem, F. Farhadi, A column generation approach for aircraft sequencing problems: A computational study, *J. Oper. Res. Soc.*, **66** (2015), 1717–1729.
25. H. Balakrishnan, B. Chandran, Algorithms for scheduling runway operations under constrained position shifting, *Oper. Res.*, **58** (2010), 1650–1665.
26. D. Harikiopoulo, N. Neogi, Polynomial-time feasibility condition for multiclass aircraft sequencing on a single-runway airport, *IEEE Trans. Intell. Transp. Syst.*, **12** (2011), 2–14.
27. A. Lieder, R. Stolletz, Scheduling aircraft take-offs and landings on interdependent and heterogeneous runways, *Transport. Res. E-Log.*, **88** (2016), 67–188.
28. M. Samà, A. D'Ariano, F. Corman, D. Pacciarelli, Coordination of scheduling decisions in the management of airport airspace and taxiway operations. *Transport. Res. Pro.*, **23** (2017), 246–262.
29. J. Jemai, M. Zekri, K. Mellouli, An NSGA-II algorithm for the green vehicle routing problem, *Evo. Comput. Com. Opt.*, **2012** (2012), 37–48.
30. M. Wei, T. Liu, B. Sun, B. B. Jing, Optimal integrated model for feeder transit route design and frequency-setting problem with stop selection, *J. Adv. Transport.*, **2020** (2020).
31. A. Slowik, H. Kwasnicka, Nature inspired methods and their industry Applications-Swarm intelligence algorithms, *IEEE T. Ind. Inform.*, **14** (2018), 1004–1015.
32. M. A. Dulebenets, A comprehensive evaluation of weak and strong mutation mechanisms in evolutionary algorithms for truck scheduling at cross-docking terminals, *IEEE Access*, **6** (2018).
33. X. Zhao, C. Wang, J. Su, J. Wang, Research and application based on the swarm intelligence algorithm and artificial intelligence for wind farm decision system, *Renew. Energ.*, **134** (2019), 681–697.
34. L. Brezonik, I. Fister, V. Podgorelec, Swarm intelligence algorithms for feature selection: A review, *Appl. Sci.*, **8** (2018), 1521.
35. M. A. Dulebenets, A delayed start parallel evolutionary algorithm for just-in-time truck scheduling at a cross-docking facility, *Int. J. Prod. Econ.*, **212** (2019), 236–258.
36. H. Anandakumar, K. Umamaheswari, A bio-inspired swarm intelligence technique for social aware cognitive radio handovers, *Comput. Electr. Eng.*, **71** (2018), 925–937.
37. T. Li, G. Kou, Y. Peng, Y. Shi, Classifying with adaptive hyper-spheres: An incremental classifier based on competitive learning, *IEEE Trans. Syst. Man Cybern. Syst.*, **50** (2020), 1218–1229.
38. G. Kou, C. S. Lin, A cosine maximization method for the priority vector derivation in AHP, *Eur. J. Oper. Res.*, **235** (2014), 225–232.



AIMS Press

©2020 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)