APEM journal

Advances in Production Engineering & Management
Volume 17 | Number 2 | June 2022 | pp 183–192
https://doi.org/10.14743/apem2022.2.429

ISSN 1854-6250
Journal home: apem-journal.org
Original scientific paper

A bi-objective optimization of airport ferry vehicle scheduling based on heuristic algorithm: A real data case study

Han, X.[a], **Zhao, P.X.**[a,*], **Kong, D.X.**[a]

[a]School of Management, Shandong University, Jinan, Shandong, P.R. China

ABSTRACT

The optimization of ferry vehicle scheduling is the key factor to improve the punctuality of flights and passenger satisfaction at airports. Based on the airport reality, a bi-objective mixed integer linear programming model for airport ferry vehicle scheduling is proposed in this paper, in which the first objective is to minimize the number of vehicles used, and the second objective is to minimize the maximum number of flights per ferry vehicle serving under the constraint that the first objective takes the optimal value. For the optimization model of the second objective, this paper designs three heuristic algorithms: strict equalization algorithm, relaxed equalization algorithm and transplantation algorithm, and integrates them into a main algorithm. The actual flight data of Beijing Capital International Airport are used for numerical examples, and all the examples tested can obtain the exact solution or high-quality approximate solution using the designed algorithm, which verifies the effectiveness of the algorithm. This study can be used to inform decisions on the efficient and balanced use of airport ferry vehicles. Despite the system presented in the paper is designed for airport, it can be applied to solve similar vehicle scheduling problems.

ARTICLE INFO

Keywords:
Ferry vehicle;
Vehicle routing;
Bi-objective optimization;
Heuristic algorithm;
Strict equalization algorithm;
Relaxed equalization algorithm;
Transplantation algorithm

**Corresponding author:*
pxzhao@sdu.edu.cn
(Zhao, P.X.)

Article history:
Received 10 May 2022
Revised 9 August 2022
Accepted 16 August 2022

1. Introduction

In recent years, the rapid development of the civil aviation industry has made the scheduling and management of the limited ground resources at airports, such as parking positions, runways and ground support vehicles, gradually become important and complex. Among them, the ground support vehicles are special vehicles that provide a series of ground services for the aircraft, such as refuelling, air catering and ferrying. However, at present, the scheduling of airport ground support vehicles is still mainly manual, which is inefficient and easy to cause flight delays [1]. Flights parked at remote stands need ferry vehicles to transport passengers, so the scheduling level of ferry vehicles not only affects the punctuality of flights, but also directly affects the experience of passengers. The ferry service is characterized by time-consuming and different number of vehicles required for flights, which makes the ferry vehicle resources even more tight. In addition to completing all ferry services on time with as few ferry vehicles as possible, balancing the workload of each ferry vehicle as far as possible also facilitates driver scheduling and vehicle maintenance.

The research on flight ground service scheduling mainly includes the scheduling optimization of ground service crew [2-8] and ground support vehicles. The ground support vehicle scheduling problem is a kind of vehicle routing problem, and the vehicle routing problem is widely used in the scheduling of vehicles such as electric vehicles [9], automated guided vehicles [10] and logistics vehicles [11]. For example, Norin *et al.* designed a greedy randomized adaptive search procedure to solve the de-icing vehicle scheduling problem [12]. Du *et al.* designed a column generation heuristic algorithm to solve the tractor scheduling problem [13]. Schyns designed an ant colony algorithm to solve the refuelling truck scheduling problem [14]. Padrón *et al.* proposed a decomposition framework and a sequence iterative method to solve the collaborative scheduling problem of multiple ground support vehicles [15]. Padrón and Guimarans [16] improved the algorithm proposed by Padrón *et al.* [15]. This paper focuses on the studies on ferry vehicle scheduling, as detailed in Table 1.

As can be seen from Table 1, although there are existing studies involving optimization objectives in terms of balancing the workload of ferry vehicles, they are all nonlinear objective functions, which are solved using solvers or meta-heuristic algorithms. Solvers have difficulty solving large-scale nonlinear integer programming problems, and meta-heuristic algorithms often have difficulty obtaining the accuracy of the resulting solutions. This paper constructs a bi-objective optimization model for ferry vehicle scheduling, in which the first objective is to minimize the number of ferry vehicles used, and the programming model is a two-index arc-flow model based on a directed acyclic network, which is easy to obtain the optimal solution. The second objective is to minimize the maximum number of flights served by a single ferry vehicle under the constraint of using the minimum number of ferry vehicles, and the programming model is a three-index mixed integer linear programming model. Since the model of the second optimization objective has large number of variables and is difficult to solve directly, three heuristic algorithms are designed and integrated into a main algorithm to solve the model. The analysis of numerical examples shows that the algorithm can solve 42 out of 60 examples to the optimum, and the Gap of the remaining examples is also very small.

The rest of the paper is organized as follows. Section 2 constructs a bi-objective optimization model for ferry vehicle scheduling. Section 3 presents several heuristic algorithms for solving the optimization model of the second objective. Section 4 uses the actual flight data of Beijing Capital International Airport for numerical examples to verify the effectiveness of the designed heuristic algorithms. Section 5 provides conclusions and some possible directions for future research.

Table 1 Studies related to ferry vehicle scheduling

Literature	Optimization goal	Programming model	Solving method
[17]	Maximize robustness	ILP	Column generation
[18]	Minimize total costs, including vehicle usage costs and driving costs	LP	Shortest augmenting path algorithm
[19]	Minimize the variance of the number of flights per ferry vehicle serving	QP	Gurobi
[20]	Minimize the number of vehicles, the total vehicle mileage and the variance of the number of flights per ferry vehicle serving	Three-objective IP	Two-stage heuristic algorithm
[21]	Minimize the number of vehicles and the total deviation of the number of flights per ferry vehicle serving	Bi-objective MIP	Particle swarm optimization
[22]	Minimize the number of vehicles	MILP, LP	CPLEX
[1]	Minimize the number of vehicles, the total vehicle mileage and the difference between the vehicle arrival time and the earliest service time	Three-objective MIP	Non-dominated sorting genetic algorithm
[23]	Minimize the number of vehicles and the total vehicle idle time	Bi-objective MIP	Non-dominated sorting genetic algorithm
[24]	Minimize the number of vehicles	ILP, LP	Lingo

2. Problem description and model construction

Let there be $|N|$ flights requiring ferry services at an airport during a certain time period (including arriving flights and departing flights; if a flight arrives and then departs during this time period, it is considered as two flights), where N is the set of these flights. Ferry vehicles for arriving flights are required to transport passengers from the parking positions to the terminal, while ferry vehicles for departing flights do the opposite. Let the time required for the ferry vehicle to travel from the end position of the ferry service of flight $i \in N$ (the terminal if i is an arriving flight, otherwise the parking position of i) to the start position of the ferry service of flight $j \in N$ (the parking position of j if j is an arriving flight, otherwise the terminal) be t_{ij}. Let the ferry service start time window for flight $i \in N$ be $[a_i, b_i]$ and the required service time be $serv_i$, where a_i and b_i are determined according to the Scheduled Time of Arrival (STA) or Scheduled Time of Departure (STD) of flight i. Depending on the aircraft type, let the number of ferry vehicles required for flight $i \in N$ be D_i (if relevant data on the number of passengers on the flight are available, the number of passengers can be used instead of the aircraft type to determine the number of ferry vehicles required for the flight more accurately).

To facilitate the algorithm design, this paper transforms the research problem into a vehicle scheduling problem with only one ferry vehicle for each flight by setting up virtual flights. Let flight $i \in N$ correspond to D_i virtual flights with the same time window, service time and other relevant time parameters as i. Denote the set of all virtual flights as N^V [19, 22, 24]. We use the service sequence and service time compatibility information in the flight service time window to construct the underlying network $G = (V, A)$ of the ferry vehicle scheduling model, where $V = \{0, |N^V| + 1\} \cup \{i | i \in N^V\}$ (Nodes 0 and $|N^V| + 1$ can be regarded as ferry vehicle depot), $A = \{(i,j) | Ad_{ij} = 1, i, j \in V\}$. Ad_{ij} is an element of the adjacency matrix Ad of network G. Ad_{ij} is calculated as follows.

$$Ad_{ij} = \begin{cases} 1, \text{if } i = 0 \text{ and } j \in N^V \\ 1, \text{if } j = |N^V| + 1 \text{ and } i \in N^V \\ 1, \text{if } i, j \in N^V \text{ and } a_i + serv_i + t_{ij} \le b_j \\ 0, \text{otherwise} \end{cases} \tag{1}$$

For the ferry vehicle, the inequality $serv_i + t_{ij} \ge b_i - a_i$ holds for $\forall i, j \in N^V$, then G is a directed acyclic network [22, 25].

The first objective of the ferry vehicle scheduling optimization is to minimize the number of ferry vehicles used. The decision variable $x_{ij} \in \{0,1\}$ decides whether a ferry vehicle serves node j immediately after serving node i, and the decision variable $t_i \in [a_i, b_i]$ decides the service start time of flight i. The two-index mixed integer linear programming model for the first optimization objective is constructed as follows.

$$\min \sum_{(0,j) \in A} x_{0j} \tag{2}$$

$$\sum_{j|(i,j) \in A} x_{ij} = 1, \forall\, i \in N^V \tag{3}$$

$$\sum_{j|(j,i) \in A} x_{ji} = 1, \forall\, i \in N^V \tag{4}$$

$$a_i \le t_i \le b_i, \forall\, i \in N^V \tag{5}$$

$$t_i + serv_i + t_{ij} \le t_j + M(1 - x_{ij}), \forall (i,j) \in A, i, j \in N^V \tag{6}$$

$$x_{ij} \in \{0,1\}, \forall (i,j) \in A \tag{7}$$

The objective function Eq. 2 in the above model is to minimize the number of ferry vehicles dispatched. Constraints Eqs. 3 to 4 indicate that each virtual flight is served only once. Constraints Eqs. 5 to 6 are time window constraints.

The second optimization objective is to minimize the maximum number of flights served by a single ferry vehicle under the constraint of using the minimum number of ferry vehicles. Let the optimal value of model 2 to 7 be K, that is, at least K ferry vehicles are needed to serve all flights on time. The second programming model uses the three-index decision variable x_{ij}^k to decide whether ferry vehicle k serves node j immediately after serving node i, and introduces a new decision variable z to represent the maximum number of flights served by a single ferry vehicle. The mixed integer linear programming model for the second optimization objective is constructed as follows.

$$\min z \tag{8}$$

$$\sum_{i,j|(i,j)\in A} x_{ij}^k \le z+1, \forall k \in \{1,2,\dots,K\} \tag{9}$$

$$\sum_{i|i\in N^V} x_{0i}^k = 1, \forall k \in \{1,2,\dots,K\} \tag{10}$$

$$\sum\nolimits_{k=1}^{K} \sum_{j|(i,j)\in A} x_{ij}^k = 1, \forall i \in N^V \tag{11}$$

$$\sum_{j|(i,j)\in A} x_{ij}^k = \sum_{j|(j,i)\in A} x_{ji}^k, \forall i \in N^V, k \in \{1,2,\dots,K\} \tag{12}$$

$$e_i \le t_i \le l_i, \forall i \in N^V \tag{13}$$

$$t_i + serv_i + t_{ij} \le t_j + M\left(1 - x_{ij}^k\right), \forall k \in \{1,2,\dots,K\}, (i,j) \in A \text{且} i,j \in N^V \tag{14}$$

$$x_{ij}^k \in \{0,1\}, \forall k \in \{1,2,\dots,K\}, (i,j) \in A \tag{15}$$

The objective function Eq. 8 and constraint Eq. 9 in the above model minimize the maximum number of flights served by a single ferry vehicle. Constraint Eq. 10 indicates that each ferry vehicle is dispatched only once. Constraint Eq. 11 indicates that each virtual flight is served only once. Constraint Eq. 12 is the flow balance condition of the virtual flight node. Constraints Eqs. 13 to 14 are time window constraints. Obviously, $\lceil \frac{|\boldsymbol{N^V}|}{\boldsymbol{K}} \rceil$ is a lower bound for the optimal value of model in Eqs. 8 to 15.

3. Proposed approach: A heuristic algorithm

Models in Eq. 2 to 7 can be solved quickly using the solver to obtain the exact solution [22]. Models 8 to 15 is a three-index arc-flow model with a large number of variables, which is difficult to solve directly using the solver for flight data with a 24-hour planning period. In this paper, three heuristic algorithms (see Algorithms 1, 2, and 3) are designed and integrated into a main algorithm (see Algorithm 0) to solve models in 8 to 15. The symbols involved in the algorithms are shown in Table 2.

Table 2 The symbols used in the algorithms

Symbol	Meaning
popsize	The number of solutions
$solu(u,i)$	The ferry vehicle serving flight i in solution u, $u = 1,2,\dots$, popsize, $i = 1,2,\dots,\lvert N^V\rvert$
$st(u,i)$	The service start time of flight i in solution u, $u = 1,2,\dots$, popsize, $i = 1,2,\dots,\lvert N^V\rvert$
$fv1(u,k)$	The last flight that ferry vehicle k has served in solution u, $u = 1,2,\dots$, popsize, $k = 1,2,\dots,K$
$fv2(u,k)$	The service start time of the last flight that ferry vehicle k has served in solution u, $u = 1,2,\dots$, popsize, $k = 1,2,\dots,K$
$fv3(u,k)$	The number of flights that ferry vehicle k has served in solution u, $u = 1,2,\dots$, popsize, $k = 1,2,\dots,K$
ava_i	The set of ferry vehicles that can serve flight i
ava_i'	The set of ferry vehicles whose number of served flights satisfies a specific condition among ferry vehicles available for flight i
$fea(u)$	The number of flights served in solution u
$opt(u)$	The maximum number of flights served by a single ferry vehicle in solution u
$solu0(i)$	The ferry vehicle serving flight i in the optimal solution of models 2 to 7
$st0(i)$	The service start time of flight i in the optimal solution of models 2 to 7
$solu'$	The set of solutions with the largest number of served flights
UF	The set of unserved flights in the solutions with the largest number of served flights
$UF^{\max}$	The last flight unserved in the solutions with the largest number of served flights

Firstly, all the flights in N^V are sorted in ascending order according to a_i and respectively numbered as flight $1,2,\dots,|N^V|$. Then, if two flights are served by the same ferry vehicle, the vehicle must serve the flight with the smaller number first. Algorithms 1-3 all ensure that the constraints on the time window and the number of ferry vehicles are not violated, then the resulting solution, if infeasible, is due to the fact that some flights are not served. Algorithm 2 has looser restrictions than Algorithm 1 in terms of the equilibrium degree of the ferry vehicle workload, so the main algorithm executes Algorithm 1 first, and if no feasible solution is obtained, Algorithm 2 is executed. If Algorithm 2 still fails to obtain a feasible solution, the time windows of the unserved flights are generally in the peak period of flight take-off and landing, during which there are fewer feasible vehicle scheduling schemes. Algorithm 3 transplants part of the ferry vehicle scheduling arrangement from the optimal solution of models 2 to 7, while taking into account the equilibrium degree of the ferry vehicle workload. If neither Algorithm 1 nor Algorithm 2 can get a feasible solution, Algorithm 3 can be executed.

Algorithm 0: Main algorithm

Input: popsize, K, $|N^V|$, a_i, b_i, $serv_i$, t_{ij}, $solu0(i)$, $st0(i)$
Execute Algorithm 1, return $solu(u,i)$, $st(u,i)$, $fea(u)$, $opt(u)$;
If $\max_{u\in\{1,2,\dots,\text{popsize}\}}\{fea(u)\} < |N^V|$
  Execute Algorithm 2, return $solu(u,i)$, $st(u,i)$, $fea(u)$, $opt(u)$;
  If $\max_{u\in\{1,2,\dots,\text{popsize}\}}\{fea(u)\} < |N^V|$
    $solu' \leftarrow \left\{solu(u,i) \middle| i \in \{1,2,\dots,|N^V|\}, fea(u) = \max_{u\in\{1,2,\dots,\text{popsize}\}}\{fea(u)\}\right\}$;
    $UF \leftarrow \{i \in \{1,2,\dots,|N^V|\} | solu(u,i) \in solu', solu(u,i) = 0\}$;
    $UF^{\max} \leftarrow \max_{i\in UF}\{i\}$;
    Execute Algorithm 3, return $solu(u,i)$, $st(u,i)$, $fea(u)$, $opt(u)$;
Return $solu(u,i)$, $st(u,i)$, $fea(u)$, $opt(u)$

Algorithm 1 first makes ferry vehicles $1,2,\dots,K$ serve flights $1,2,\dots,K$ respectively (lines 1-7 of Algorithm 1). For flights $K+1, K+2, \dots, |N^V|$, the set ava_i of the ferry vehicles that can serve it is determined in turn, and from the set, a ferry vehicle that has served the least number of flights is randomly selected to serve it (lines 8-18 of Algorithm 1). The time complexity of Algorithm 1 is $O(\text{popsize} \times (|N^V| - K) \times K^2)$.

Algorithm 1: Strict equalization algorithm

Input: popsize, K, $|N^V|$, a_i, b_i, $serv_i$, t_{ij}
1 For $u \leftarrow 1,2,\dots,\text{popsize}$
2   For $i \leftarrow 1,2,\dots,K$
3     $solu(u,i) \leftarrow i$;
4     $st(u,i) \leftarrow a_i$;
5     $fv1(u,i) \leftarrow i$;
6     $fv2(u,i) \leftarrow a_i$;
7     $fv3(u,i) \leftarrow 1$;
8 For $u \leftarrow 1,2,\dots,\text{popsize}$
9   For $i \leftarrow K+1, K+2, \dots, |N^V|$
10     $ava_i \leftarrow \left\{k \in \{1,2,\dots,K\} \middle| fv2(u,k) + serv_{fv1(u,k)} + t_{fv1(u,k),i} \le b_i\right\}$;
11   If $ava_i \neq \emptyset$
12       $ava_i' \leftarrow \left\{k \in ava_i \middle| fv3(u,k) = \min_{k\in ava_i}\{fv3(u,k)\}\right\}$;
13   Randomly select a ferry vehicle in ava_i', denote as $\boldsymbol{k}$;
14       $solu(u,i) \leftarrow \boldsymbol{k}$;
15       $st(u,i) \leftarrow \max\{fv2(u,\boldsymbol{k}) + serv_{fv1(u,\boldsymbol{k})} + t_{fv1(u,\boldsymbol{k}),i}, a_i\}$;
16       $fv2(u,\boldsymbol{k}) \leftarrow \max\{fv2(u,\boldsymbol{k}) + serv_{fv1(u,\boldsymbol{k})} + t_{fv1(u,\boldsymbol{k}),i}, a_i\}$;
17       $fv1(u,\boldsymbol{k}) \leftarrow i$;
18       $fv3(u,\boldsymbol{k}) \leftarrow fv3(u,\boldsymbol{k}) + 1$;
19 $fea(u) \leftarrow \sum_{k=1}^{K} fv3(u,k)$; $opt(u) \leftarrow \max_{k\in\{1,2,\dots,K\}}\{fv3(u,k)\}$;
20 Return $solu(u,i)$, $st(u,i)$, $fea(u)$, $opt(u)$

The difference between Algorithm 2 and Algorithm 1 is that for flights $K+1, K+2, \dots, |N^V|$, if there are ferry vehicles with the number of served flights less than or equal to $\lfloor\frac{|N^V|}{K}\rfloor$ in set ava_i, then a ferry vehicle is randomly selected from them (from the ferry vehicles with the number of served flights less than or equal to $\lfloor\frac{|N^V|}{K}\rfloor$) to serve flight i. Otherwise, the factor of ferry vehicle task volume is no longer considered, and a ferry vehicle is randomly selected from ava_i to serve flight i (lines 9-27 of Algorithm 2). The time complexity of Algorithm 2 is $O(\text{popsize} \times (|N^V| - K) \times K)$.

Algorithm 2: Relaxed equalization algorithm

Input: popsize, K, $|N^V|$, a_i, b_i, $serv_i$, t_{ij}
For $\forall u \in \{1,2,\dots,\text{popsize}\}, i \in \{1,2,\dots,|N^V|\}$, set $solu(u,i) \leftarrow 0$;
For $u \leftarrow 1,2,\dots,$ popsize
For $i \leftarrow 1,2,\dots,K$
$solu(u,i) \leftarrow i$;
$st(u,i) \leftarrow a_i$;
$fv1(u,i) \leftarrow i$;
$fv2(u,i) \leftarrow a_i$;
$fv3(u,i) \leftarrow 1$;
For $u \leftarrow 1,2,\dots,$ popsize
For $i \leftarrow K+1, K+2,\dots,|N^V|$
$ava_i \leftarrow \{k \in \{1,2,\dots,K\} | fv2(u,k) + serv_{fv1(u,k)} + t_{fv1(u,k),i} \le b_i\}$;
If $ava_i \ne \emptyset$
$ava_i' \leftarrow \left\{k \in ava_i \middle| fv3(u,k) \le \lfloor\frac{|N^V|}{K}\rfloor\right\}$;
If $ava_i' \ne \emptyset$
Randomly select a ferry vehicle in ava_i', denote as $\Bbbk$;
$solu(u,i) \leftarrow \Bbbk$;
$st(u,i) \leftarrow \max\{fv2(u,\Bbbk) + serv_{fv1(u,\Bbbk)} + t_{fv1(u,\Bbbk),i}, a_i\}$;
$fv2(u,\Bbbk) \leftarrow \max\{fv2(u,\Bbbk) + serv_{fv1(u,\Bbbk)} + t_{fv1(u,\Bbbk),i}, a_i\}$;
$fv1(u,\Bbbk) \leftarrow i$;
$fv3(u,\Bbbk) \leftarrow fv3(u,\Bbbk) + 1$;
If $ava_i' = \emptyset$
Randomly select a ferry vehicle in ava_i, denote as $\Bbbk$;
$solu(u,i) \leftarrow \Bbbk$;
$st(u,i) \leftarrow \max\{fv2(u,\Bbbk) + serv_{fv1(u,\Bbbk)} + t_{fv1(u,\Bbbk),i}, a_i\}$;
$fv2(u,\Bbbk) \leftarrow \max\{fv2(u,\Bbbk) + serv_{fv1(u,\Bbbk)} + t_{fv1(u,\Bbbk),i}, a_i\}$;
$fv1(u,\Bbbk) \leftarrow i$;
$fv3(u,\Bbbk) \leftarrow fv3(u,\Bbbk) + 1$;
$fea(u) \leftarrow \sum_{k=1}^{K} fv3(u,k)$; $opt(u) \leftarrow \max_{k\in\{1,2,\dots,K\}}\{fv3(u,k)\}$;
Return $solu(u,i)$, $st(u,i)$, $fea(u)$, $opt(u)$

Algorithm 3 relies on the optimal solution of models 2 to 7 and the parameter $UF^{\max}$ determined by the solution obtained by Algorithm 2. For flights $1,2,\dots,UF^{\max}$, the results of the ferry vehicle assignment for these flights in the optimal solution of models 2 to 7 are directly transplanted (lines 2-8 of Algorithm 3). For flights $UF^{\max}+1, UF^{\max}+2,\dots,|N^V|$, the ferry vehicles continue to be assigned as in Algorithm 2. The time complexity of Algorithm 3 is $O(\text{popsize} \times (|N^V| - UF^{\max}) \times K)$.

Algorithm 3: Transplantation algorithm

Input: popsize, K, $|N^V|$, a_i, b_i, $serv_i$, t_{ij}, $UF^{\max}$, $solu0(i)$, $st0(i)$
1 For $\forall u \in \{1,2,\dots,\text{popsize}\}, k \in \{1,2,\dots,K\}$, set $fv3(u,k) \leftarrow 0$;
2 For $u \leftarrow 1,2,\dots,$ popsize
3 For $i \leftarrow 1,2,\dots,UF^{\max}$
4 $solu(u,i) \leftarrow solu0(i)$;
5 $st(u,i) \leftarrow st0(i)$;
6 $fv1(u, solu0(i)) \leftarrow i$;
7 $fv2(u, solu0(i)) \leftarrow st0(i)$;
8 $fv3(u, solu0(i)) \leftarrow fv3(u, solu0(i)) + 1$;
9 For $u \leftarrow 1,2,\dots,$ popsize
10 For $i \leftarrow UF^{\max}+1, UF^{\max}+2,\dots,|N^V|$
11 $ava_i \leftarrow \{k \in \{1,2,\dots,K\} | fv2(u,k) + serv_{fv1(u,k)} + t_{fv1(u,k),i} \le b_i\}$;
12 If $ava_i \ne \emptyset$
13 $ava_i' \leftarrow \left\{k \in ava_i \middle| fv3(u,k) \le \lfloor\frac{|N^V|}{K}\rfloor\right\}$;
14 If $ava_i' \ne \emptyset$
15 Randomly select a ferry vehicle in ava_i', denote as $\Bbbk$;
16 $solu(u,i) \leftarrow \Bbbk$;
17 $st(u,i) \leftarrow \max\{fv2(u,\Bbbk) + serv_{fv1(u,\Bbbk)} + t_{fv1(u,\Bbbk),i}, a_i\}$;
18 $fv2(u,\Bbbk) \leftarrow \max\{fv2(u,\Bbbk) + serv_{fv1(u,\Bbbk)} + t_{fv1(u,\Bbbk),i}, a_i\}$;
19 $fv1(u,\Bbbk) \leftarrow i$;
20 $fv3(u,\Bbbk) \leftarrow fv3(u,\Bbbk) + 1$;
21 If $ava_i' = \emptyset$

Randomly select a ferry vehicle in ava_i, denote as k;
$solu(u,i) \leftarrow k$;
$st(u,i) \leftarrow \max\{fv2(u,k) + serv_{fv1(u,k)} + t_{fv1(u,k),i}, a_i\}$;
$fv2(u,k) \leftarrow \max\{fv2(u,k) + serv_{fv1(u,k)} + t_{fv1(u,k),i}, a_i\}$;
$fv1(u,k) \leftarrow i$;
$fv3(u,k) \leftarrow fv3(u,k) + 1$;
$fea(u) \leftarrow \sum_{k=1}^{K} fv3(u,k)$; $opt(u) \leftarrow \max_{k\in\{1,2,\dots,K\}}\{fv3(u,k)\}$;
Return $solu(u,i)$, $st(u,i)$, $fea(u)$, $opt(u)$

4. Numerical examples: Results and discussion

This section takes 24 hours (0:00-23:59) as the planning period, and uses the flight data of Beijing Capital International Airport for 60 days from February 1 to April 1, 2018 as the numerical examples. The average number of virtual flights for these 60 data sets is 900. Models 2 to 7 can be solved directly using the CPLEX solver, and the results are shown in Table 3. The running conditions are a 2.7 GHz PC (Intel® Core™ i7-7500U CPU), Windows 7 operating system, running with 8 GB RAM, and using CPLEX 12.9 solver.

Table 3 Results of solving models 2 to 7

Serial number	Number of virtual flights	Solving time (s)	Objective value	Serial number	Number of virtual flights	Solving time (s)	Objective value
1	957	34.04	76	31	905	68.16	66
2	815	64.27	58	32	844	94.07	67
3	878	133.89	61	33	889	131.92	72
4	870	102.51	74	34	876	38.92	70
5	856	45.35	60	35	884	129.76	64
6	901	137.72	72	36	968	152.71	78
7	885	216.34	61	37	844	120.32	54
8	912	105.71	75	38	898	59.25	72
9	968	34.63	76	39	839	21.14	68
10	894	121.24	62	40	1029	180.03	70
11	746	42.99	55	41	1040	226.7	77
12	825	91.09	64	42	1054	269.23	70
13	855	89.62	57	43	878	117.8	59
14	876	81.09	72	44	832	72.57	64
15	832	19.94	74	45	947	32.96	80
16	878	124.16	62	46	856	99.97	71
17	856	80.29	64	47	904	104.13	80
18	888	113.24	72	48	902	27.81	64
19	859	102.35	54	49	770	226.68	57
20	997	278.51	72	50	979	203.46	67
21	933	170.15	65	51	877	107.53	63
22	990	214.33	64	52	862	106.35	57
23	1016	271.08	62	53	845	105.75	56
24	931	183.39	64	54	888	135.85	63
25	923	157.87	64	55	880	130.92	58
26	884	132.13	56	56	1012	175.3	78
27	838	102.57	66	57	1068	181.98	70
28	897	106.77	66	58	1033	177.51	74
29	858	125.05	62	59	791	1806.15	49
30	841	75.93	60	60	939	4276.94	54

As can be seen from Table 3, for models 2 to 7, the optimal solutions can be obtained for all 60 groups of data, and the average time to solve the problem is 200 seconds. The optimal solutions for 58 groups of data can be obtained within 5 minutes. Algorithm 0 designed in Section 3 is implemented using MATLAB R2017b and used for these 60 groups of data. The solving results of models 8 to 15 are obtained as shown in Table 4 (popsize is set to 300).

As can be seen from Table 4, the heuristic algorithms designed are very suitable for solving models 8 to 15. Among these 60 groups of data, the exact optimal solutions can be obtained for 42 groups of data (all obtained by Algorithm 1), and for the other 18 groups of data, the objective value differs from the lower bound $\lceil\frac{|N^V|}{K}\rceil$ of the model by only 1. There are 11 groups of data that need to use Algorithm 2 and only 4 groups of data that need to use Algorithm 3, all of which can obtain the approximate optimal solution, and the maximum Gap with the lower bound of the model is only 7.69 %.

Take the 60th group of data as an example to show the results of ferry vehicle scheduling. This group of data has a total of 939 virtual flights using 54 ferry vehicles with the $\lceil\frac{|N^V|}{K}\rceil$ value of 18. The 60th group of data uses Algorithm 3 to obtain the final solution with the $UF^{\max}$ value of 255, and a single ferry vehicle serves up to 19 virtual flights. The number of flights served by each ferry vehicle is shown in Table 5.

Table 4 Results of solving models 8 to 15

Serial number	$\lceil\frac{\lvert N^V\rvert}{K}\rceil$	Algorithm used	Objective value	Gap (%)	Serial number	$\lceil\frac{\lvert N^V\rvert}{K}\rceil$	Algorithm used	Objective value	Gap (%)
1	13	1	13	0.00	31	14	1	14	0.00
2	15	1	15	0.00	32	13	1	13	0.00
3	15	1	15	0.00	33	13	1	13	0.00
4	12	1	12	0.00	34	13	1	13	0.00
5	15	1	15	0.00	35	14	1	14	0.00
6	13	1	13	0.00	36	13	1	13	0.00
7	15	1	15	0.00	37	16	1	16	0.00
8	13	1	13	0.00	38	13	1	13	0.00
9	13	1	13	0.00	39	13	1	13	0.00
10	15	1	15	0.00	40	15	1	15	0.00
11	14	1	14	0.00	41	14	1	14	0.00
12	13	1	13	0.00	42	16	1	16	0.00
13	15	1	15	0.00	43	15	1	16	6.25
14	13	1	13	0.00	44	13	1	14	7.14
15	12	1	12	0.00	45	12	1	13	7.69
16	15	1	15	0.00	46	13	2	14	7.14
17	14	1	14	0.00	47	12	2	13	7.69
18	13	1	13	0.00	48	15	2	16	6.25
19	16	1	16	0.00	49	14	2	15	6.67
20	14	1	14	0.00	50	15	2	16	6.25
21	15	1	15	0.00	51	14	2	15	6.67
22	16	1	16	0.00	52	16	2	17	5.88
23	17	1	17	0.00	53	16	2	17	5.88
24	15	1	15	0.00	54	15	2	16	6.25
25	15	1	15	0.00	55	16	2	17	5.88
26	16	1	16	0.00	56	13	2	14	7.14
27	13	1	13	0.00	57	16	3	17	5.88
28	14	1	14	0.00	58	14	3	15	6.67
29	14	1	14	0.00	59	17	3	18	5.56
30	15	1	15	0.00	60	18	3	19	5.26

Table 5 Results of ferry vehicle scheduling on a certain day

Ferry vehicle	Number of flights served	Ferry vehicle	Number of flights served	Ferry vehicle	Number of flights served
1	15	19	18	37	18
2	18	20	13	38	16
3	17	21	18	39	19
4	19	22	18	40	18
5	19	23	14	41	15
6	19	24	19	42	17
7	18	25	18	43	19
8	15	26	18	44	16
9	14	27	17	45	15
10	18	28	18	46	17
11	17	29	18	47	16
12	19	30	19	48	19
13	19	31	19	49	16
14	19	32	19	50	15
15	18	33	19	51	17
16	19	34	15	52	15
17	19	35	17	53	18
18	19	36	19	54	16

5. Conclusion

This paper proposes a bi-objective optimization model for airport ferry vehicle scheduling to optimize the number of vehicles used and the equilibrium degree of the vehicle workload, in which the objective function designed for the second objective is to minimize the maximum number of flights served by a single ferry vehicle. For the second optimization objective, three concise and efficient heuristic algorithms are designed and tested using the actual flight data from an airport. The analysis of numerical examples shows that Algorithms 1 and 2, which do not depend on the optimal solution of the first optimization objective, have been able to solve most of the examples, and the effectiveness of these algorithms is verified by the fact that all three algorithms can obtain exact solutions or high-quality approximate solutions.

Although the bi-objective optimization in this paper is designed for airports, its algorithms can also be applied to solve similar vehicle scheduling problems that balance the workload of vehicles. Possible future research directions also include the problem of real-time scheduling of ferry vehicles considering uncertain and unexpected conditions.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (grant numbers 72071122, 72134004); the Natural Science Foundation of Shandong Province (grant number ZR2020MG002); and the Social Science Planning Research Project of Shandong Province (grant number 20CGLJ11).

References

[1] Zeng, Z.H., Jia, L.Q. (2020). Research on airport special vehicle scheduling problem, In: *Proceedings of 2020 International Conference on Virtual Reality and Intelligent Systems (ICVRIS),* Zhangjiajie, China, 1076-1078.

[2] Lv, H., Qin, Y., Tang, R., Luo, C. (2011). Multi-population co-evolutionary memetic algorithm for multi-objective airport ground services scheduling problem, In: *Proceedings of 2011 3rd International Workshop on Intelligent Systems and Applications,* Wuhan, China, 1-4, doi: 10.1109/ISA.2011.5873292.

[3] Ip, W.H., Wang, D., Cho, V. (2013). Aircraft ground service scheduling problems and their genetic algorithm with hybrid assignment and sequence encoding scheme, *IEEE Systems Journal*, Vol. 7, No. 4, 649-657, doi: 10.1109/JSYST.2012.2196229.

[4] Trabelsi, S.F., Mora-Camino, F., Padrón Astorga, S.P. (2013). A decentralized approach for ground handling fleet management at airports, In: *Proceedings of 2013 International Conference on Advanced Logistics and Transport,* Sousse, Tunisia, 302-307, doi: 10.1109/ICAdLT.2013.6568476.

[5] Tang, F., Liu, S., Dong, X., Cui, B. (2017). Aircraft ground service scheduling problems and Partheno-genetic algorithm with hybrid heuristic rule, In: *Proceedings of 2017 IEEE 7th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, Honolulu, USA, 551-555, doi: 10.1109/CYBER.2017.8446518.

[6] Fink, M., Desaulniers, G., Frey, M., Kiermaier, F., Kolisch, R., Soumis, F. (2019). Column generation for vehicle routing problems with multiple synchronization constraints, *European Journal of Operational Research*, Vol. 272, No. 2, 699-711, doi: 10.1016/j.ejor.2018.06.046.

[7] Gök, Y.S., Guimarans, D., Stuckey, P.J., Tomasella, M., Ozturk, C. (2020). Robust resource planning for aircraft ground operations, In: Hebrard, E., Musliu, N. (eds.), *Integration of constraint programming, artificial intelligence, and operations research, CPAIOR 2020, Lecture notes in computer science,* Vol. 12296, Springer, Cham, Switzerland, 222-238, doi: 10.1007/978-3-030-58942-4_15.

[8] Gök, Y.S., Tomasella, M., Guimarans, D., Ozturk, C. (2020). A simheuristic approach for robust scheduling of airport turnaround teams, In: *Proceedings of 2020 Winter Simulation Conference (WSC),* Orlando, USA, 1336-1347, doi: 10.1109/WSC48552.2020.9383947.

[9] Huang, A.Q., Zhang, Y.Q., He, Z.F., Hua, G.W., Shi, X.L. (2021). Recharging and transportation scheduling for electric vehicle battery under the swapping mode, *Advances in Production Engineering & Management*, Vol. 16, No. 3, 359-371, doi: 10.14743/apem2021.3.406.

[10] Shi, W., Tang, D.B., Zou, P. (2021). Multi-objective automated guided vehicle scheduling based on MapReduce framework, *Advances in Production Engineering & Management*, Vol. 16, No. 1, 37-46, doi: 10.14743/apem2021.1.383.

[11] Zhao, P.X., Luo, W.H., Han, X. (2019). Time-dependent and bi-objective vehicle routing problem with time windows, *Advances in Production Engineering & Management*, Vol. 14, No. 2, 201-212, doi: 10.14743/apem2019.2.322.

[12] Norin, A., Yuan, D., Granberg, T.A., Värbrand, P. (2012). Scheduling de-icing vehicles within airport logistics: A heuristic algorithm and performance evaluation, *Journal of the Operational Research Society*, Vol. 63, No. 8, 1116-1125, doi: 10.1057/jors.2011.100.

[13] Du, J.Y., Brunner, J.O., Kolisch, R. (2014). Planning towing processes at airports more efficiently, *Transportation Research Part E: Logistics and Transportation Review*, Vol. 70, 293-304, doi: 10.1016/j.tre.2014.07.008.
[14] Schyns, M. (2015). An ant colony system for responsive dynamic vehicle routing, *European Journal of Operational Research*, Vol. 245, No. 3, 704-718, doi: 10.1016/j.ejor.2015.04.009.
[15] Padrón, S., Guimarans, D., Ramos, J.J., Fitouri-Trabelsi, S. (2016). A bi-objective approach for scheduling ground-handling vehicles in airports, *Computers & Operations Research,* Vol. 71, 34-53, doi: 10.1016/j.cor.2015.12.010.
[16] Padrón, S., Guimarans, D. (2019). An improved method for scheduling aircraft ground handling operations from a global perspective, *Asia-Pacific Journal of Operational Research*, Vol. 36, No. 4, Article No. 1950020, doi: 10.1142/S0217595919500209.
[17] Diepen, G., Pieters, B.F.I., van den Akker, J.M., Hoogeveen, J.A. (2013). Robust planning of airport platform buses, *Computers & Operations Research*, Vol. 40, No. 3, 747-757, doi: 10.1016/j.cor.2011.08.002.
[18] Weiszer, M., Chen, J., Locatelli, G. (2015). An integrated optimisation approach to airport ground operations to foster sustainability in the aviation sector, *Applied Energy*, Vol. 157, 567-582, doi: 10.1016/j.apenergy.2015.04.039.
[19] Li, Q., Bi, J., Li, Z. (2017). Research on ferry vehicle scheduling problem within airport operations, In: *Proceedings of 2017 10th International Symposium on Computational Intelligence and Design (ISCID),* Hangzhou, China, 248-251, doi: 10.1109/ISCID.2017.153.
[20] Liu, Y., Zhang, J., Ding, C., Bi, J. (2019). Modeling and heuristic algorithm of ground ferry vehicle scheduling in large airports, In: *Proceedings of the 19th COTA International Conference of Transportation Professionals*, Nanjing, China, 159-170, doi: 10.1061/9780784482292.015.
[21] Zhao, P.X., Gao, W.Q., Han, X., Luo, W.H. (2019). Bi-objective collaborative scheduling optimization of airport ferry vehicle and tractor, *International Journal of Simulation Modelling*, Vol. 18, No. 2, 355-365, doi: 10.2507/IJSIMM18(2)CO9.
[22] Han, X., Zhao, P., Meng, Q., Yin, S., Wan, D. (2020). Optimal scheduling of airport ferry vehicles based on capacity network, *Annals of Operations Research*, Vol. 295, No. 1, 163-182, doi: 10.1007/s10479-020-03743-0.
[23] Liu, Y., Wu, J., Tang, J., Wang, W., Wang, X. (2022). Scheduling optimisation of multi-type special vehicles in an airport, *Transportmetrica B: Transport Dynamics*, Vol. 10, No. 1, 954-970, doi: 10.1080/21680566.2021.1983484.
[24] Zhao, P., Han, X., Wan, D. (2021). Evaluation of the airport ferry vehicle scheduling based on network maximum flow model, *Omega,* Vol. 99, Article No. 102178, doi: 10.1016/j.omega.2019.102178.
[25] Bertsimas, D., Jaillet, P., Martin, S. (2019). Online vehicle routing: The edge of optimization in large-scale applications, *Operations Research,* Vol. 67, No. 1, 143-162, doi: 10.1287/opre.2018.1763.