Hindawi
Journal of Advanced Transportation
Volume 2023, Article ID 4484202, 10 pages
https://doi.org/10.1155/2023/4484202

Research Article

How to Improve the Efficiency of Check-In Counter: A Counter-Sharing Method

Shao-Peng Yang,[1] Tie-Qiao Tang ,[1] Tao Wang,[2] and Liang Chen[3]

[1]*School of Transportation Science and Engineering, Beihang University, Beijing 100191, China*
[2]*School of Vehicle and Mobility, Tsinghua University, Beijing, China*
[3]*Beijing Key Laboratory of Traffic Engineering, Beijing University of Technology, Beijing 100124, China*

Correspondence should be addressed to Tie-Qiao Tang; tieqiaotang@buaa.edu.cn

Received 18 April 2023; Revised 31 May 2023; Accepted 13 June 2023; Published 21 June 2023

Academic Editor: Yuan Gao

Improving the efficiency of check-in counters is a significant concern for airport operation management departments and airline companies. However, the existing studies mainly optimize the counter allocation based on the number of counters and airport demands, which needs more exploration on improving the utilization rate of check-in counters under the condition of unchanged resource allocation. In this paper, we propose a counter-sharing method to improve check-in efficiency by sharing the idle check-in counters in the adjacent check-in area. In the proposed counter-sharing method, we first define the passenger's total walking distance and waiting time for queues as the metrics, then reassign the check-in areas and adjust the internal departure sequence of flights to maximize the counter-sharing rate. The numerical results indicate that the proposed counter-sharing method can improve the efficiency of check-in counters and reduce the passenger's total walking distance and waiting time in the queue during the check-in process, which can enhance the airport's operation efficiency and competitiveness.

1. Introduction

Aviation transportation has rapidly developed in recent years. The Airbus Global Market Forecast for 2019–2038 indicated that the annual air traffic growth rate would be 4.3% in the next 20 years [1]. However, the growth rate of airport facilities could hardly reach the same speed as air traffic since the expansion was a long-term and expensive project [2]. When the airport facilities cannot be added to enhance the airport service capacity, the increased aviation transportation will cause many problems (e.g., congestion, long waiting times for passengers in queues) in airport terminals.

Each flight has a check-in counter demand, and the efficiency of check-in counters significantly impacts the airport's level of service and system performance [3]. Therefore, the check-in counter assignment problem has risen, and researchers have proposed many models and methods to study the management of check-in counters. For example, Ahn and Park [4] employed the passenger arrival distribution pattern to propose an optimization model for calculating the most appropriate number of check-in counters and the corresponding duration of each counter. The model could offer airlines a means of operating check-in counters with greater cost-effectiveness, thus enhancing customer service. Yan et al. [5] developed an integer programming model to assign the check-in counters and designed a heuristic method for solving their model. They later improved this study and formulated one 0-1 programming method to minimize the total inconsistencies in common-use counter assignments [6]. Van Dijk and Van der Sluis [7] proposed a method combining simulation with integer programming to minimize the number and opening hours of check-in counters. In order to balance the quality of service stipulated by the airport authority and the optimal allocation of resources to the check-in counters, Parlar and

Sharafali [8] developed one stochastic dynamic programming model, which could optimize the number of check-in counters for opening the time window specified.

Tang [9] developed a new network model for the optimization of common-use check-in counter assignments, which could minimize the number of counters required for daily operations. Hsu et al. [10] developed a model for the dynamic allocation of check-in counters with the target of minimizing the waiting time for passengers, and the feasibility of the developed model is validated by comparing actual data from the free selection of check-in counters by passengers and the dynamic assignment of passengers to check-in counters. Gao et al. [11] proposed a target optimization model for the purpose of putting a minimum of the equipment quantity and shortening passengers' onboarding time and employed the improved NSGA-II algorithm to solve the model and get the allocation plan of check-in counters. Lalita et al. [12] proposed an exact integer linear programming model for allocating variable check-in counters in airports, which solves the check-in counter allocation problem with deterministic inputs and variable counter allocation. Liu et al. [13] developed an associative decision integer programming model to quantitatively describe the total time of handling luggage in the collaborative work system, which could generate various allocation schemes of check-in counters to reduce the waiting time of passengers in queues. The above studies provide practical methods for the efficient operation of check-in counter assignments and valuable means of developing effective longer-term solutions to the problem of passenger terminal congestion and delays.

Recently, passenger's requirements for airport service quality have increased quickly, which has led researchers and aviation agencies to pay more attention to the airport service quality and passenger satisfaction. For example, Airport Council International defined the overall service quality as the overall level of passenger satisfaction measured by survey responses [14]. Fodness and Murray [15] empirically investigated passengers' expectations of service at airport terminals and found that passengers do not expect long waiting times for queues or long walking distances. Bezerra and Gomes [16] used the partial least squares structural equation modeling to analyze the drivers of passenger loyalty to the airport and found that passenger experience was critical to their loyalty to the airport. Kayapınar and Erginel [17] developed a bi-criteria cost function that involves the costs of opening check-in counters and the costs of modeling passengers' waiting time. Adacher and Flamini [18] considered passenger satisfaction and proposed a bi-criteria objective function to minimize operational costs and the passengers' discomfort in terms of waiting time in line. Batouei et al. [19] delved into passenger experience by analyzing the data of 377 passengers and found that a good passenger experience will bring an excellent reputation to airports and attract more passengers.

The above studies indicate that developing effective passenger-oriented (the waiting time in queues and walking distances to access airport facilities) facility management strategies could increase passengers' satisfaction and spread positive word of mouth, which helps promote the airport's high-quality development.

Sharing is a new mode in the current social environment, and it has been applied in many commercial projects, such as bike sharing, portable battery sharing, and car sharing. Sharing can reasonably distribute resources and improve a system's efficiency. As the optimization theories and airport hardware advanced, it became feasible to manage the check-in counters optimally using the sharing mode. In this paper, a counter-sharing method is developed to enhance the utilization rate of check-in counters by sharing the idle counters in the adjacent check-in areas. Integrating the sharing concept into the management of check-in counters in airports can enhance check-in efficiency without additional operational costs. In the proposed counter-sharing method, the passenger's total waiting time and walking distances are taken as the metrics to evaluate the reassignment of check-in areas based on the departure flight schedule. The reassignment of check-in areas based on the departure flight schedule needs to transform the solutions into vectors and improve the solutions. The general idea behind the differential evolution algorithm is the representation of a solution as a vector of decision variables, which fits our problem very well. Thus, we use the differential evolution algorithm to solve the problem.

Compared with the existing studies, this paper has two contributions: (i) We explore the planning of check-in counters from the standpoint of airline companies and develop a counter-sharing method with the target of enhancing the efficiency of check-in counters by sharing idle the check-in counters in the adjacent check-in areas; (ii) we reassign the check-in areas and internally adjust the departure sequence of airline flights to minimize the passengers' waiting time for queues and walking distances during check-in, which are important metrics reported in the literature. The management of check-in counters from available research does not consider sharing mode explicitly, and we incorporate the concept of sharing into the management of check-in counters and provide exact solutions to real-world problems. The feasibility of the counter-sharing method is validated by case analysis. The results of the case analysis evidenced the superiority of the counter-sharing method in terms of shorter passengers' waiting times and walking distances and better utilization of check-in counters. The remainder of this paper is organized as follows: the implementation of the counter-sharing method and related assumptions are introduced in detail in Section 2; then, case studies, which are employed to validate the proposed method, are presented in Section 3; and some conclusions are summarized in Section 4.

2. Problem Statement

In this section, we introduce some rules for check-in and propose the counter-sharing method and its implementation.

2.1. Rules for Check-In. Today, the check-in process can be achieved in various ways: online, via self-service kiosks at the airport, and via the traditional check-in counters where airline representatives serve the passengers. The traditional check-in counters where airline representatives serve the passengers are still the first choice for passengers [20]. Thus, most passengers will first visit check-in counters to get a check-in service when traveling by air; they use check-in counters to check luggage and choose, buy, or change a seat. Besides, each airline occupies a check-in area in the airport terminal to place check-in counters and rents the check-in counters in this area to serve all of its flights daily.

The check-in counter system has the following three primary rules [5]:

(1) It is an exclusive-use system with multiple counters and queues of passengers. These check-in counters will service only passengers booked for a specific flight.
(2) The finite value of the number of confirmed passengers for each flight is known a priori.
(3) The check-in counters typically open two hours before the scheduled flight departure and must close 30 min before the boarding gate closes, irrespective of whether all passengers show up at the counters.

The check-in counters should meet the following rules:

(1) Each counter has the same service rate.
(2) One check-in counter must be open for every 45 passengers.
(3) The passenger's waiting cost, in units of time, is the same at each check-in counter.

It should be noted that rule 2 of the check-in counters is the hard rule of some airports, which has been formulated to provide a good quality of service to the passengers during the check-in process [21].

Furthermore, we suppose that we have the following information:

(1) The layout of the terminal area
(2) The number of passengers on each flight and each flight's departure time
(3) The distances of passengers moving to the check-in areas
(4) The number of check-in counters that each airline sets at its check-in area.

2.2. Implementation of the Counter-Sharing Method. The utilization of the airport check-in counters is crucial since it determines the check-in efficiency to a certain extent [22, 23]. Sharing is a new mode in the current social environment, which could reasonably distribute resources and improve a system's efficiency. Therefore, we propose a check-in counter management method based on sharing to improve the utilization of check-in counters, which could enhance check-in efficiency.

To illustrate this counter-sharing method, we assume that we have a discrete search space X and a function F that assigns a value to each one of the elements in the search space. The problem can be formulated as follows:

$$\text{Min } \mathrm{F}(\mathrm{S_i}), \quad \mathrm{S_i} \in X, \tag{1}$$

where S_i is feasible solution in the discrete search space X.

According to the previous studies [24–27], the passengers' walking distances and waiting time for queues should be taken as factors that determine the final value assigned by the function, since these two factors can better reflect practical needs for improving airport operations in real life. Therefore, we construct a function based on passengers' walking distances and waiting time in queues. To unify the units of the two objective function values, we divide the passengers' walking distances (unit: meters) by 1.0 m/s and transform them into the passenger walking time (unit: seconds).

The function based on passengers' walking distances and waiting time for queues can be defined as follows:

$$F(S_i) = \alpha_1 * T^{\text{total}} + \alpha_2 * D^{\text{total}}, \tag{2}$$

where T^{total} is a factor that measures passengers' waiting time for queues in the solution; D^{total} is a factor that evaluates passengers' walking distances in the solution; and α_1, α_2 are the weights of the corresponding factors.

D^{total} represents the sum of the walking distances of passengers on all flights to the corresponding check-in areas. Thus, we can define D^{total} as follows:

$$D^{\text{total}} = \sum_{i=1}^{M} P_i D_i, \tag{3}$$

where M is the number of flights; P_i is the number of the ith flight passengers; and D_i is the distance of the ith flight passengers moving to the corresponding check-in area.

T^{total} represents the sum of the queue time of all flight passengers at the check-in counters, and we can define T^{total} as follows:

$$T^{\text{total}} = \sum_{i=1}^{M} T_i, \tag{4}$$

where T_i is the total queue time of the ith flight passengers.

The airlines will open the corresponding number of check-in counters according to the constraint that one check-in desk must be open for every 45 passengers, and the maximum number of check-in counters in a check-in area is typically 5 [21]. Thus, the passengers' total queue time at check-in counters is different for flights with different numbers of passengers. For flights with no more than 225 passengers, the passengers' total queue time should have two features: (i) when the number of passengers is balanced with the number of check-in counters, the passengers' total queue time equals the product of a fixed constant and the number of flight passengers; (ii) when the number of passengers is unbalanced with the number of check-in counters, the passengers' total queue time is less than that of balance and increases as the number of flight passengers increases. Therefore, for $P_i \le 225$, the $T_{i,1}$ can be defined as follows:

$$T_{i,1} = 45 * T^{\text{average}} * \left[\left(I_i + 1\right) + \left(\frac{P_i}{45} - \left\lfloor \frac{P_i}{45} \right\rfloor \right)^{I_i+1} \right], \quad (5)$$

where T^{average} is the average queue time of each passenger; I_i is an integer calculated based on the number of flight passengers, which represents the flight size and can be defined as follows:

$$I_i = \left\lfloor \left(P_i - P^{\min}\right)/45 + 1 \right\rfloor, \quad (6)$$

where $P^{\min}$ is the minimum flight parameter, which is a constant.

For flights with more than 225 passengers, the passengers' total queue time shall equal the total queue time when the number of passengers and the number of check-in counters are balanced, plus additional delay time. Therefore, for $P_i > 225$, the $T_{i,2}$ can be defined as follows:

$$T_{i,2} = P_i * T^{\text{average}} + \left[\left(P_i - 225\right) * K_{i,1} - \frac{5 * K_{i,1}\left(K_{i,1} - 1\right)}{2} \right] * T^{\text{service}}, \quad (7)$$

where T^{service} is the service time of check-in counters; $K_{i,1}$ is the number of extra queues and can be defined as follows:

$$K_{i,1} = \left\lceil \frac{\left(P_i - 225\right)}{5} \right\rceil. \quad (8)$$

The counter-sharing method is intended to improve the efficiency of check-in counters by sharing idle counters between airlines in adjacent check-in areas. The actual operation of the counter-sharing method is as follows: Suppose the check-in areas of two flights are adjacent, and the check-in counters' opening hours for the two flights overlap. In that case, the flight with more passengers (more than 225 passengers) can borrow extra check-in counters from the flight with fewer passengers (fewer than 180 passengers) during the check-in counters' overlapping opening time. Therefore, the total queue time of the ith flight passengers under the counter-sharing method T_i^{sharing} can be defined as follows:

For $P_i \le 225$,

$$T_i^{\text{sharing}} = T_{i,1}, \quad (9)$$

For $P_i > 225 \wedge P_i^{\text{sharing}} \le 225$,

$$T_i^{\text{sharing}} = \rho_i * T_{i,1} + \left(1 - \rho_i\right) * T_{i,2}, \quad (10)$$

For $P_i^{\text{sharing}} > 225$,

$$T_i^{\text{sharing}} = \rho_i * P_i * T^{\text{average}} + \rho_i * \left[\left(P_i^{\text{sharing}} - 225\right) * K_{i,2} - \frac{\left(5 + B_i\right) * K_{i,2}\left(K_{i,2} - 1\right)}{2} \right] * T^{\text{service}} + \left(1 - \rho_i\right) * T_{i,2}, \quad (11)$$

where P_i^{sharing} is the number of the ith flight passengers under the counter-sharing method; B_i is the maximum number of check-in counters that adjacent check-in areas of the ith flight can share. The quantitative relationship between P_i^1 and B_i can be defined as follows:

$$P_i^{sharing} = P_i - 45 * B_i. \quad (12)$$

ρ_i is the period that could share check-in counters, which can be defined as follows:

$$\rho_i = \frac{t_i}{90}, \quad (13)$$

where t_i is the intersection of the opening time of the check-in counter of the ith flight and that of the adjacent check-in area.

$K_{i,2}$ is the number of extra queues under the counter-sharing method and can be defined as follows:

$$K_{i,2} = \left\lceil \frac{\left(P_i^{sharing} - 225\right)}{5 + B_i} \right\rceil. \quad (14)$$

To help better understand the counter-sharing method, we take an example as follows.

As shown in Figure 1, flight X in check-in area A has 140 passengers, and its boarding time is 10 : 00; for the rules to be respected, three check-in counters should be opened from 8 : 00 to 9 : 30. Flight Y in check-in area B has 315 passengers, and its boarding time is 10 : 35. For the rules to be respected, five check-in desks should be opened from 8 : 35 to 10 : 05. Flight Z in check-in area C has 280 passengers, and its boarding time is 14 : 00; therefore, five check-in desks should be opened from 12 : 00 to 13 : 30. In this case, between 8 : 35 and 9 : 30, Flight Y can borrow the two extra check-in counters from Area A to improve check-in efficiency. Since there is no overlapping time between Flight X and Flight Z, Flight Z cannot borrow check-in counters from Flight X. The "Y" in Figure 1 means Flight Y can borrow check-in counters from Area A, and the "N" in Figure 1 means Flight Y cannot borrow check-in counters from Area C.

The counter-sharing method could (i) enable airlines with rich check-in resources to share their check-in counters to save costs; (ii) make airlines lacking check-in counters

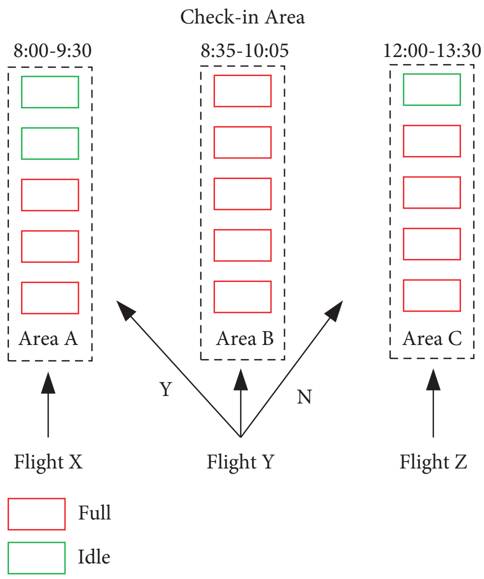

Figure 1: A diagrammatic sketch of the counter-sharing method.

borrow additional check-in counters to reduce passenger waiting time in queues. Besides, the above weights could be given different values to assign a different priority to the parameters. Depending on the airport's requirements, these priorities will affect the selection of various feasible solutions. Furthermore, the functions used by this approach are not restricted to only two values; they could be extended to include more parameters depending on the particular case of research in question.

2.3. Restraint Condition. Based on the above discussions, the counter-sharing method can be defined as a static optimization problem based on airports' check-in areas. Theoretically, the check-in areas can be reassigned based on the departure flight schedule to maximize the sharing of check-in counters among airlines. Here, the reassignment of the check-in areas based on the departure flight schedule refers to internally adjusting the departure sequence of airline flights which would alter the corresponding relationship between the original check-in areas and airlines. Reassigning the check-in areas based on the departure flight schedule can make the opening hours of check-in counters for flights in adjacent check-in areas overlap as much as possible and ensure maximizing the counter-sharing rate. Therefore, it is necessary to combine the reassignment of check-in areas based on the departure flight schedule with the counter-sharing method to maximize the utilization of check-in counters.

When we perform the redistribution of the check-in areas, we should consider the following constraints:

(1) Check-in counters are opened in advance 2 h before the flight takes off.

(2) Check-in counters are closed in advance 30 min before the flight takes off.

(3) The number of open check-in counters is calculated based on the rule that one check-in desk must be open for every 45 passengers.

For all flights of m airlines, the vector used to represent the information can be defined as $s = [a_i, N_1, N_2, \ldots N_m]$, where a_i is an integer in $(1, 2, \ldots\alpha)$, which is used to indicate an assignment sequence of check-in areas; α is the check-in areas' total number of assignment sequences, which is equal to the numerical value of check-in areas' permutations $A(m, m)$. The vector representing the flight information for the jth airline with n_j flights is $N_j = [b_j, t_1^o, t_2^o, \ldots t_{n_j}^o]$, where b_j is an integer within $(1, 2, \ldots\beta)$, which is used to indicate a departure flight's sequence; β is the airline's total number of departure flight sequences, which is equal to the numerical value of flights' permutations $A(n_j, n_j)$; t_l^o represents the check-in counters' opening times for the lth flight.

The reassignment of the check-in areas corresponds to the airlines, after one check-in area is assigned to an airline, the remaining check-in areas can only be assigned to other airlines. The departure flight schedules are established sequentially in time slots, which take one flight at a time and look for the corresponding available time slot that satisfies the restrictions; once allocated, the schedules continue with the next flight on the assignment list. After all the flights and check-in areas are assigned, an initial solution is obtained.

2.4. Solution Algorithm. Based on the above discussions, redistributing check-in areas based on the departure flight schedule need to transform the solutions into vectors and improve the solutions. The differential evolution algorithm, developed by Storn and Price [28], is a heuristic search algorithm based on population, and each individual in the group corresponds to a solution vector. The general idea behind an evolutionary algorithm is the representation of a solution in the form of a vector of decision variables. Transforming the decision variables into a vector-like representation is an interesting and challenging problem. Once the decision variables have been represented in a vector, the optimization problem can be specified. Thus, the differential evolution algorithm is suitable for solving our problems. Table 1 shows the pseudocode of the differential evolution algorithm.

One of the critical tasks in the differential evolution algorithm is to properly represent the information with vectors, which will significantly influence the algorithm's performance. In Section 2.3, we used vectors to represent the identifier of each flight, the number of passengers on each flight, and the check-in counters' opening times. Another task of the evolutionary algorithm is the crossover, which is the main operation for improving the current solutions. The crossing is performed between the elements of two solutions (solution A (SolA) and solution B (SolB)). This study's check-in areas and departure flight schedules must be optimized. Hence, SolA and SolB should contain vectors representing the check-in areas and departure times of flights. The algorithm will take one element from SolA and

Table 1: The pseudocode of the differential evolution algorithm.

Algorithm 1: DE algorithm

Input: Population: M; Dimension: D; Genetation: T
Output: The best vector (solution) $-\Delta$
$t \longleftarrow 1$ (initialization);
For $i = 1$ to M do
  For $j = 1$ to D do
    $x_{i,t}^{j} = x_{\min}^{j} + rand(0,1) \bullet (x_{\max}^{j} - x_{\min}^{j})$;
  end
end
While $(|f(\Delta)| \geq \varepsilon)$ or $(t \leq T)$ do
  For $i = 1$ to M do
    ‣ (*Mutation and Crossover*)
    For $j = 1$ to D do
      $v_{i,t}^{j} = \text{mutation}(x_{i,t}^{j})$
      $u_{i,t}^{j} = Crossover(x_{i,t}^{j}, v_{i,t}^{j})$
    end
    ‣ (*Greedy Selection*)
    If $f(u_{i,t}) < f(x_{i,t})$ then
      $x_{i,t} \longleftarrow u_{i,t}$
      If $f(x_{i,t}) < f(\Delta)$ then
        $\Delta \longleftarrow x_{i,t}$;
      end
    else
      $x_{i,t} \longleftarrow x_{i,t}$
    end
  end
  $t \longleftarrow t + 1$;
end
Return the best vector Δ;

randomly choose another one from SolB for the crossover. Figure 2 illustrates the crossing process; the light blue color denotes the elements of the solutions that the crossover operators have changed. During the execution of the algorithm, once a feasible solution is generated, it must be evaluated on a static basis using an objective function to improve the solution.

To maintain the consistency of the generated solutions, the algorithm will verify two aspects of the new solution:

(1) The corresponding relations between the flights and the check-in areas should be ensured.

Here, we use $[a_i, N_1, N_2, \ldots N_m]$ to indicate an assignment sequence of check-in areas. The flight information for the jth airline N_j can be defined as a vector $[b_j, t_1^o, t_2^o, \ldots t_{n_j}^o]$. When the value range of b_j is $(1, 2, \ldots\beta)$, it could include all the departure flight sequences for the ith check-in area. Therefore, the corresponding relations between the flights and the check-in areas can be ensured.

(2) Constraints should be ensured to meet consistency.

The crossover will evaluate whether the solution violates the constraints in Section 2.3. If it does not, it will be kept as a feasible solution, which will be evaluated later with other feasible solutions.

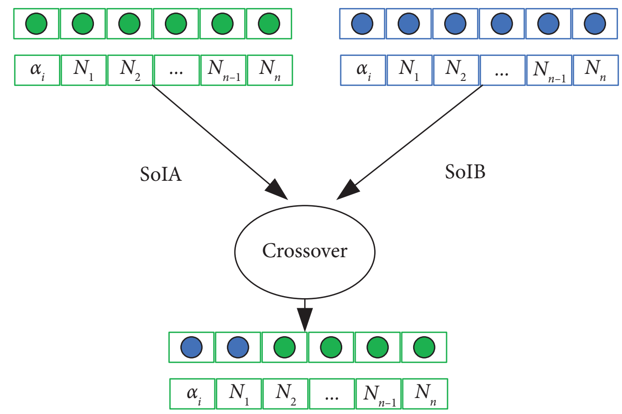

Figure 2: Crossover between solutions.

Table 2: The values of parameters.

Parameter name	Value
D_A, D_F (m)	190
D_B, D_E (m)	170
D_C, D_D (m)	150
$P^{\min}$	90
T^{average} (s)	160
T^{server} (s)	65

3. Case Study

In this section, we use a case study with six check-in areas (*A*, *B*, *C*, *D*, *E*, and *F*) and twenty-four flights to test the counter-sharing method implementation and then perform sensitivity analyses of the weighting vectors.

3.1. Model Testing and Results. The operating aircraft mainly include the *B*737–500, *A*320, *B*737–900T1, *B*757–200, *B*766–300 ER, *A*340–300, *A*330–200, and *B*777–200, which are common aircraft types. The capacities of these aircraft are 130, 150, 165, 200, 235, 255, 280, and 315 passengers, respectively. Besides, we assume that the check-in areas A, B, C, D, E, and F are symmetrically arranged, and the terminal entrance is at the symmetrical center of these check-in areas. The walking distances of passengers accessing these check-in areas are $D_A, D_B, D_C, D_D, D_E, D_F$, respectively. For simplicity, we set the weighting vector of the passengers' walking distances and waiting time for queues to (1.1). Some parameter values can be defined in Table 2.

Tables 3 and 4 represent the initial and optimal solutions, respectively. The two departure schedules display each flight's identifier, the number of passengers on each flight,

Table 3: The initial departure flight schedule.

Check-in area	Flight ID	The number of passengers	Flight departure time
A	FI0001	315	7:45
A	FI0002	200	10:20
A	FI0003	255	13:15
A	FI0004	165	16:05
B	FI0005	150	8:10
B	FI0006	200	10:35
B	FI0007	235	12:50
B	FI0008	165	15:45
C	FI0009	235	7:20
C	FI0010	130	9:55
C	FI0011	200	12:30
C	FI0012	165	15:25
D	FI0013	150	8:30
D	FI0014	165	11:20
D	FI0015	150	12:35
D	FI0016	235	16:00
E	FI0017	235	7:40
E	FI0018	130	10:30
E	FI0019	200	13:35
E	FI0020	165	16:30
F	FI0021	200	8:45
F	FI0022	255	11:05
F	FI0023	235	13:50
F	FI0024	280	16:15

Table 4: The optimal departure flight schedule.

Check-in area	Flight ID	The number of passengers	Flight departure time
A	FI0004	165	7:45
A	FI0001	315	10:20
A	FI0003	255	13:15
A	FI0002	200	16:05
B	FI0019	200	8:10
B	FI0018	130	10:35
B	FI0020	165	12:50
B	FI0017	235	15:45
C	FI0016	235	7:20
C	FI0013	150	9:55
C	FI0015	150	12:30
C	FI0014	165	15:25
D	FI0007	235	8:30
D	FI0006	200	11:20
D	FI0005	150	12:35
D	FI0008	165	16:00
E	FI0011	200	7:40
E	FI0010	130	10:30
E	FI0012	165	13:35
E	FI0009	235	16:30
F	FI0024	280	8:45
F	FI0022	255	11:05
F	FI0023	235	13:50
F	FI0021	200	16:15

the check-in area of each flight, and the departure time of each flight. Although Tables 3 and 4 are not the actual flight plan, they are sufficient to test and validate the method presented in this work. From Tables 3 and 4, we can find that the flight departure time of the adjacent check-in areas in the optimal solution is closer than the initial solution, and the difference between the number of flight passengers in the adjacent check-in areas in the same period becomes prominent. For example, to minimize the passengers' waiting time for queues and walking distances during check-in, the flight departure sequence of check-in area A has changed from 1, 2, 3, and 4 to 4, 1, 3, and 2; the flight

Table 5: The numerical results.

Initial solution (s)	Optimal integer solution (s)	Initial queue time (s)	Optimal queue time (s)	Initial walking time (s)	Optimal walking time (s)
2, 085, 912	1, 980, 298	1, 205, 562	1, 105, 948	880, 350	874, 350

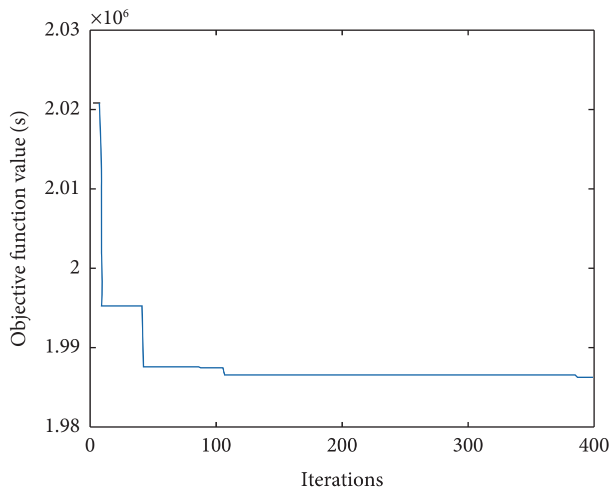

Figure 3: The convergence of the objective function.

departure sequence of check-in area F has changed from 1, 2, 3, and 4 to 4, 2, 3, and 1. Besides, the airline in check-in area B has moved to check-in area D, the airline in check-in area C has moved to check-in area E, the airline in check-in area D has moved to check-in area C, and the airline in check-in area E has moved to check-in area B.

Table 5 presents the numerical results of the optimal solution and initial solution. In the optimal solution, the total passenger walking time is 874,350 s, and the total passenger queue time is 1,105,948 s. The weighted objective value, equal to the total walking time plus the total queue time, is 1,980,298 s. In the initial solution, the total passenger walking time is 880,350 s, with a gap of 6,000 s from the optimal solution; the passenger queue time is 1,205,562 s, with a gap of 99,614 s from the optimal solution; and the weighted objective value of the initial solution is 2,085,912 s, with a gap of 105,614 s from the optimal solution. The numerical results show the effectiveness of the counter-sharing method, which could reduce the weighted objective value of passenger walking time and queue time and improve check-in efficiency.

We test the convergence speed of the evolution algorithm. Figure 3 illustrates the evolution of the function value versus the number of iterations. It can be found that the evolution algorithm efficiently solves our problem, which converges with a few iterations.

The above results indicate that with our counter-sharing method, a high-quality assignment of check-in areas based on the departure flight schedule can be found, which could improve the utilization of check-in counters and reduce the passengers' walking distances and waiting time for queues.

Besides, we perform a sensitivity analysis on the weighting vectors, which are the basic inputs. Sensitivity analyses of other factors may be performed in a similar manner if needed.

As mentioned above, the weighting vector reflects the relative importance of two objective functions (e.g., passengers' walking time and queue time). To evaluate the effects of various weighting vectors on the assignment of check-in areas and departure flight schedules, we test four scenarios with weighting vectors of (1/2.1), (1/10.1), (1.1/2), and (1.1/10). The results are displayed in Table 6.

According to Table 6, when the weighting vector is (1/2 : 1), there is a weighted objective value of 1,434,292 s; when the weighting vector is (1/10 : 1), there is a weighted objective value of 991,022 s; when the weighting vector is (1 : 1/2), there is a weighted objective value of 1,570,355 s; and when the weighting vector is (1 : 1/10), there is a weighted objective value of 1,265,193 s. These results confirm the performance and stability of our proposed method. When the weight of the queue time decreases, the allocation of check-in areas and departure flight schedules tends to reduce the objective function value of passengers' walking time. When the weight of the walking time decreases, the allocation of check-in areas and departure flight schedules tends to reduce the objective function value of passengers' queue time.

It might be not credible and particle without considering the proportion of self-service check-ins and passengers that do not require luggage check-in, which needs to be further discussed. Therefore, we test four scenarios with proportions of self-service check-in and passengers that do not require luggage check-in of 10%, 20%, 30%, and 40%. The results are displayed in Table 7.

Table 6: The results of different weighting vectors.

Weighting vector (walking time: queue time)	Total queue time	Total walking time	Weighted objective values
1/2 : 1	1, 017, 472	830, 633	1, 434, 292
1/10 : 1	1, 106, 728	778, 172	991, 022
1 : 1/2	1, 133, 255	874, 200	1,570, 355
1 : 1/10	1, 179, 948	852, 450	1, 265, 193

Table 7: The results of different proportions of passengers who do not require check-in counters.

Proportions (%)	Total queue time	Total walking time	Weighted objective values
5	1, 008, 175	836, 333	1, 844, 508
10	962, 174	778, 172	1, 740, 346
15	873, 699	725, 711	1, 569, 410
20	785, 223	673, 250	1, 458, 473

According to Table 7, when the proportion of passengers who do not require check-in counters is 5%, the total passenger walking time is 1,008,175 s, and the total passenger queue time is 836,333 s. When the proportion of passengers who do not require check-in counters is 10%, the total passenger walking time is 962,174 s, and the total passenger queue time is 778,172 s. When the proportion of passengers who do not require check-in counters is 5%, the total passenger walking time is 873,699 s, and the total passenger queue time is 725,711 s. When the proportion of passengers who do not require check-in counters is 10%, the total passenger walking time is 785,223 s, and the total passenger queue time is 673,250 s. These results confirm that the counter-sharing method still performs well even if some passengers self-service check-in services or do not require check-in counters. When the proportion of passengers who do not require check-in counters increases, the weighted objective value decreases, and the decrease in passengers' total queue time is more significant than that of passengers' walking time.

4. Conclusions

The aviation industry is expected to grow at a high pace in the coming future. Therefore, it is necessary to take resource management technology to support the rising demand for airport facility resources. In this paper, we develop a counter-sharing method to improve check-in counters' efficiency by sharing idle counters between airlines in adjacent check-in areas, reassign the check-in areas and departure flight schedules to maximize check-in counter sharing, and take the passengers' waiting time for queues and walking distances as the metric. To solve the problem in a reasonable time, we use a differential evolution. Through numerical tests, our method is flexible enough to include a variety of constraints in the evolutionary algorithm to provide solutions that align with the objectives of airport terminals. The results are effective in the field of airport operations, which could help airport operators achieve a competitive advantage in marketing.

Nevertheless, this paper still has the following limitations:

(1) We lack a real-life case study to verify the counter-sharing method
(2) The proposed evaluation function cannot completely reflect all the demands of airport passengers
(3) We do not compare the performance of the proposed algorithm with that of other algorithms.

Given these limitations, we will, in the future, do the following studies:

(1) Collect data and conduct a real-life case study to verify the counter-sharing method
(2) Incorporate more factors that passengers are concerned about and develop a more reasonable evaluation function
(3) Compare our proposed algorithm with other algorithms regarding the computation time and numerical results.

Data Availability

The data displayed in this paper are simulated by the first author. If some readers need the data, contact the corresponding author via the e-mail listed in this paper.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (72231001).

References

[1] S. A. S. Airbus, *Global Market Forecast*, Airbus, Blagnac, France, 2019.
[2] M. M. Pandey, "Evaluating the service quality of airports in Thailand using fuzzy multi-criteria decision making method," *Journal of Air Transport Management*, vol. 57, pp. 241–249, 2016.

[3] M. A. Ornek, C. Ozturk, and I. Sugut, "Model-based heuristic for counter assignment problem with operational constrains: a case study," *Journal of Air Transport Management*, vol. 77, pp. 57–64, 2019.

[4] Y. Park and S. B. Ahn, "Optimal assignment for check-in counters based on passenger arrival behaviour at an airport," *Transportation Planning and Technology*, vol. 26, no. 5, pp. 397–416, 2003.

[5] S. Y. Yan, C. H. Tang, and M. Chen, "A model and a solution algorithm for airport common use check-in counter assignments," *Transportation Research Part A: Policy and Practice*, vol. 38, no. 2, pp. 101–125, 2004.

[6] S. Y. Yan, K. C. Chang, and C. H. Tang, "Minimizing inconsistencies in airport common-use checking counter assignments with a variable number of counters," *Journal of Air Transport Management*, vol. 11, no. 2, pp. 107–116, 2005.

[7] N. M. van Dijk and E. van der Sluis, "Check-in computation and optimization by simulation and IP in combination," *European Journal of Operational Research*, vol. 171, no. 3, pp. 1152–1168, 2006.

[8] M. Parlar and M. Sharafali, "Dynamic allocation of airline check-in counters: a queueing optimization approach," *Management Science*, vol. 54, no. 8, pp. 1410–1424, 2008.

[9] C. H. Tang, "A network model for airport common use check-in counter assignments," *Journal of the Operational Research Society*, vol. 61, no. 11, pp. 1607–1618, 2010.

[10] C. I. Hsu, C. C. Chao, and K. Y. Shih, "Dynamic allocation of check-in facilities and dynamic assignment of passengers at air terminals," *Computers & Industrial Engineering*, vol. 63, no. 2, pp. 410–417, 2012.

[11] Q. Gao, B. Li, B. X. Wei et al., "Juvenile hormone regulates photoperiod-mediated male reproductive diapause via the methoprene-tolerant gene in the ladybeetle *Harmonia axyridis*," *Insect Science*, vol. 29, no. 1, pp. 139–150, 2022.

[12] T. R. Lalita, D. K. Manna, and G. Murthy, "Mathematical formulations for large scale check-in counter allocation problem," *Journal of Air Transport Management*, vol. 85, Article ID 101796, 2020.

[13] Y. Liu, X. Q. Yang, Y. Xiang, Y. Chen, G. Mao, and X. Z. Zhou, "Allocation and optimization of shared self-service check-in system based on integer programming model," *Journal of Combinatorial Optimization*, vol. 44, no. 1, pp. 532–556, 2022.

[14] ACI, *Guide to Airport Performance*, Measures Airports Council international, Montreal, Canada, 2012.

[15] D. Fodness and B. Murray, "Passengers' expectations of airport service quality," *Journal of Services Marketing*, vol. 21, no. 7, pp. 492–506, 2007.

[16] G. Bezerra and C. F. Gomes, "Determinants of passenger loyalty in multi-airport regions: implications for tourism destination," *Tourism Management Perspectives*, vol. 31, pp. 145–158, 2019.

[17] S. Kayapınar and N. Erginel, "Designing the airport service with fuzzy QFD based on SERVQUAL integrated with a fuzzy multi-objective decision model," *Total Quality Management and Business Excellence*, vol. 30, no. 13-14, pp. 1429–1448, 2019.

[18] L. Adacher and M. Flamini, "Optimization of airport check-in service quality focused on operational costs and passengers' satisfaction," *PLoS One*, vol. 16, no. 8, Article ID e0253586, 2021.

[19] A. Batouei, M. Iranmanesh, H. Mustafa, D. Nikbin, and T. A. Ping, "Components of airport experience and their roles in eliciting passengers' satisfaction and behavioural intentions," *Research in Transportation Business & Management*, vol. 37, Article ID 100585, 2020.

[20] S. P. Yang, T. Q. Tang, and T. Wang, "How to raise the check-in efficiency at an airport terminal: a route guidance strategy," *IEEE Access*, vol. 10, pp. 97692–97700, 2022.

[21] M. Mujica Mota, "Check-in allocation improvements through the use of a simulation–optimization approach," *Transportation Research Part A: Policy and Practice*, vol. 77, pp. 320–335, 2015.

[22] R. M. Wang and J. F. Zhu, "The operation optimization research of airport check-in counter," *Journal of Civil Aviation Flight University of China*, vol. 25, pp. 53–56, 2014.

[23] G. Bruno, A. Diglio, A. Genovese, and C. Piccolo, "A decision support system to improve performances of airport check-in services," *Soft Computing*, vol. 23, no. 9, pp. 2877–2886, 2019.

[24] A. R. Correia, S. C. Wirasinghe, and A. G. de Barros, "Overall level of service measures for airport passenger terminals," *Transportation Research Part A: Policy and Practice*, vol. 42, no. 2, pp. 330–346, 2008.

[25] E. D. Zidarova and K. G. Zografos, "Measuring quality of service in airport passenger terminals," *Transportation Research Record*, vol. 2214, no. 1, pp. 69–76, 2011.

[26] T. L. Hwang, C. R. Jeng, and S. S. Wang, "Airport check-in counter assignment: a proposed solution," *International Journal of Aviation Management*, vol. 1, no. 4, pp. 257–270, 2012.

[27] D. Chonsalasin, S. Jomnonkwao, and V. Ratanavaraha, "Measurement model of passengers' expectations of airport service quality," *International Journal of Transportation Science and Technology*, vol. 10, no. 4, pp. 342–352, 2021.

[28] R. Storn and K. Price, "Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.