Research on Joint Optimization of Airport Surface Based on Improved A* Algorithm

Jinxin Liu¹, Sixu Feng²* and Xiaoyu Zeng³
¹-³ College of Civil Aviation, Nanjing, Jiangsu, 211106, China
*Corresponding author’s e-mail: 282761277@qq.com

Abstract. Joint optimization of runway and taxiway will help increase the usage rate of airport's existing hardware and software resources and ease flight delays. Firstly, the paper considers the relevant regulations of taxiing and the constraints of runway release interval, and takes the shortest total taxi time of each flight as the objective function to construct a joint optimization model based on the basic element layout of the airport. Secondly, the A* algorithm is improved for its characteristics and the actual situation of the airport surface, and the optimization model is solved by this improved method. Finally, taking Nanjing Lukou International Airport as an example, the optimal solution obtained by improved A* algorithm is compared with the actual running data to verify the applicability of the model.

1. Introduction

With the expansion of the airport and the rapid increase in the number of flights, how to solve the problem of taxi routing in complex airport scenes has become one of the focuses of many scholars at home and abroad. In 2001, Pesic B published the first paper on the optimization of airport surface operation, aimed at the shortest distance between the runway and the parking stand, and verified the timeliness of the algorithm by using Charles de Gaulle airport as an example[1]. In 2007, H. Balakrishnan studied the taxiway optimization operation in order to effectively reduce flight taxi time and queuing time[2]. An Active Routing (AR) framework is proposed in Chen et al. (2015b) aimed at more seamless integration of speed profiles into route and schedule optimisation[3]. Tancredi U first assigned taxiing paths to each aircraft and then used the conflict resolution mechanism to ensure safe and efficient operation of the airport surface[4]. The Roling PC built a MILP model based on a spatio-temporal network, allowing the aircraft to wait for collisions at the stand and special points, aimed at the minimum total taxi time and total waiting time, and ensuring that no conflicts occur within each planned time window[5]. Weiszer M considered the fuel consumption and environmental impact, introduces the aircraft speed curve, as well as adopts the external database method to reduce the calculation time to meet the real-time operation requirements of the airport[6]. Authors in Guepet et al.(2016) set the objective of taxi scheduling to be minimizing the emission by aircrafts’ surface movement which shows the concerning about the environmental protection[7]. Evertse C optimized the timed taxiing routes of all aircraft on an airport surface by minimizing the emissions that result from taxiing aircraft operations[8].

The above research lays a good foundation for the optimization of taxiing path, but there are few studies on the joint optimization of taxiway and runway in airport surface. Therefore, this paper attempts to establish a joint optimization model for taxiway and runway to achieve the goal of optimizing airport surface operation.
2. Joint optimization model for taxiway and runway of airport surface

Based on the research content, the following assumptions are made for the joint optimization model:

- Construct a network map, where $V$ represents a set of nodes, including intersections between taxiways, runway entrance nodes, and runway turn-off nodes. $E$ represents the taxiway set, the edge between any two nodes $v_1$ and $v_2$, $(v_1,v_2) \in E$. $(v_1,v_2)$ indicates that the aircraft’s taxi direction is $v_1$ to $v_2$.
- Assume that the aircraft is taxiing at a constant speed and is not allowed to stop or wait, that is, the taxiing process is continuous.
- The departure aircraft starts taxiing at its parking stand, and ends at the runway entrance; while the arrival aircraft starts taxiing at the runway turn-off, and ends at its parking position.

2.1. Definition of variables

The definitions of different variables and symbols in the model are as follows:

- $F^d$: Collection of all departure aircraft
- $F^a$: Collection of all arrival aircraft
- $F$: Collection of all departure and arrival aircraft, $F = \{f_1, f_2, f_3, \ldots, f_n\}$, $F^d \cup F^a = F$
- $N$: Collection of all nodes in airport network map, any node $n_m, n_n \in N$
- $N_r$: Collection of all runway entrance node, $N_r \subset N$
- $L_{mn}$: Distance between node $m$ and $n$ (Unit: km)
- $K_{mn}$: If there is an available taxiing path between nodes $m$ and $n$, $K_{mn} = 1$, otherwise $K_{mn} = 0$
- $T_{in}$: The moment when aircraft $f_i$ arrives at node $n_m$
- $T_i^0$: The theoretical earliest time when the aircraft $f_i$ started to taxi
- $B_{ij,m}$: If aircraft $f_i$ before aircraft $f_j$ arrives at node $n_m$
- $N_I$: Taxiing path of aircraft $f_i$, composed of a series of nodes $N_I = \{N^1_I, N^2_I \ldots N^k_I\}$
- $t_{ij}^0$: Minimum safe taxi time interval between aircraft $f_i$ and aircraft $f_j$
- $t_{ij}^0$: Minimum wake interval between aircraft $f_i$ and aircraft $f_j$
- $\Delta$: Runway occupancy time
- $ETA_I$: Estimated landing time of arrival aircraft $f_i$, that is the earliest time for aircraft $f_i$ started taxiing
- $ETA_P$: Estimated pushback time of departure aircraft $f_i$, that is the earliest time for aircraft $f_i$ started taxiing
- $ETD_I$: Estimated departure time of aircraft $f_i$
- $S$: Parking stand set

2.2. Objective function

At present, the allocation of the taxiing path of aircraft in large-scale airport is mostly determined by the air traffic controller, which means that the overall taxi time cannot be guaranteed to be optimal, which may cause the aircraft to taxi for too long. This will seriously affect the operation efficiency of the airport surface. Therefore, this paper establishes a mathematical model with the goal of minimizing the taxi time of all aircraft. The taxi time of each aircraft can be expressed by the difference between the time it taxis to the end node and the start node. The objective function is:

$$\min \sum_{f_i \in F} (T_{in}^m_n - T_{in}^1)$$

(1)

2.3. Constraints

2.3.1. Taxiing path constraint. It is feasible to ensure that the aircraft generates a taxiing path from the start node to the end node under the condition that the parking stand of the departing aircraft and the
runway entrance used, as well as the runway exit of the arrival aircraft and the assigned parking stand is unknown.

\[ W_{imn} \leq K_{mn} \quad \forall f_i \in F, \forall n_m, n_n \in N \]  

(2)

The constraint (2) ensures that the taxiing path of any aircraft \( f_i \) satisfies the taxiway capacity and physical availability, and ensures that the effective taxiing path of the aircraft \( f_i \) from the start to the end node is randomly generated.

2.3.2. Taxiing conflict constraint. Adopt a taxiing interval of 200m, that is, a safety interval of at least \( d = 200 \text{m} \) should be maintained between aircrafts. For aircraft with cross-conflict and transcendence conflicts, they must also be guaranteed a safety interval at each node in network map.

\[ l = \left( \frac{l_{mn}}{(T_{in} - T_{im})} \right) \cdot (T_{jm} - T_{im}) \]  

(3)

The constraint (3) is to ensure that the safety intervals of the nodes on the taxiing path are met when the two aircraft are taxiing on the same taxiway.

\[ B_{ijm}(l - d) \geq 0 \quad \forall m \in N^i \cap N^j, (m, n) \in N_{ij}, i \neq j \]  

(4)

The constraint (4) ensures a safe time interval between the two aircraft during the taxiing process and avoids cross-conflict on the nodes.

\[ B_{ijm} - B_{ijn} = 0 \quad \forall f_i, f_j \in F, \forall (m, n) \in N_i \cap N_j \]  

(5)

Assuming that each aircraft taxis at a constant speed, the constraint (5) can avoid the occurrence of over-conflict, that is, the rear aircraft does not allow to overtake the front aircraft during the taxiing process.

\[ B_{ijm} - B_{ijn} = 0 \quad \forall f_i, f_j \in F, \forall (m, n) \in N_i, (n, m) \in N_j \]  

(6)

Constraint (6) avoids head-on conflicts during taxiing, ensuring that aircraft taxiing on the same taxiway will not collide head-on.

2.3.3. Runway operational constraints. Runway mode of operation has a significant impact on aircraft taxiing. Runway operational constraints are to ensure a safe separation between take-off and take-off, take-off and landing, and landing and landing aircraft.

\[ T_{ip} + t_{ij}^w - \left( 1 - Z_{ijp} \right)M \leq T_{jp}, \forall f_i, f_j \in F^d, \forall n_p \in N_r \]  

(7)

The constraint (7) ensures that the wake separation requirements are met between the take-off and take-off aircraft.

\[ T_{in_{ki}}^l + t_{ij}^w - \left( 1 - B_{ijr} \right)M \leq T_{jn_{kj}}^l - \Delta, \forall f_i \in F^d, \forall f_j \in F^a, \forall n_{ki}^l \in N_r \]  

(8)

The constraint (8) ensures that the wake separation requirements are met between two consecutive take-off-landing aircraft. The take-off aircraft must have left the runway before the landing aircraft arrives at the runway.

\[ T_{in_{k}}^l - \left( 1 - B_{ijr} \right)M \leq T_{jn_{kj}}^l - \Delta, \forall f_i \in F^a, \forall f_j \in F^b, \forall n_{kj}^l, n_{ki}^l \in N_r \]  

(9)

The constraint (9) ensures that in the case of a landing and take-off aircraft, the take-off aircraft can enter the runway after the landing aircraft leaves the runway.

\[ T_{in_{ki}}^l + t_{ij}^w - \left( 1 - B_{ijr} \right)M \leq T_{jn_{ki}}^l, \forall f_i, f_j \in F^a \]  

(10)

The constraint (10) ensures that the wake separation requirements are met between the landing and landing aircraft.
2.3.4. Taxi time constraints. The taxi time constraint is to ensure that the aircraft’s taxi time does not violate the provisions of the preset taxi schedule.

\[ T_{in1}^i \geq ETA_i, \forall f_i \in F^a \]  

(11)

The constraint (11) is used to ensure that the arrival aircraft begins to taxi no earlier than the estimated landing time.

\[ T_{in1}^i \geq ET P_i, \forall f_i \in F^d \]  

(12)

The constraint (12) is used to ensure that the departure aircraft begins to taxi no earlier than the estimated pushback time.

\[ T_{in1}^i \geq ET P_i, \forall f_i \in F^d \]  

(13)

The constraint (13) is used to ensure that the departure aircraft taxis to the runway threshold before the estimated departure time.

3. Optimal scheduling of taxiing path based on improved A* algorithm

For the taxiing path optimization scheduling problem, the heuristic function in the standard A* algorithm is:

\[ f(n) = g(n) + h(n) \]  

(14)

Where \( g(n) \) is the current distance, that is, the distance from the start node to the current node, and \( h(n) \) is an estimated metric, that is, the shortest path distance from the current node to the end node. In the literature on optimizing the taxiing path of aircraft using the standard A* algorithm, some literatures have problems with the algorithm's infinite loop, which is caused by the characteristics of the A* algorithm itself. Therefore, this paper improves the standard A* algorithm according to the actual situation: let the flight with infinite loop start a new round of pathfinding from the start point of the runway turn-off, that is, the arrival aircraft wait at the runway turn-off node, and re-select the taxiing path from it. The specific flow chart for improved A* algorithm is as follows:

![Figure 1. Flow chart for improved A* algorithm](image-url)
4. Empirical analysis

This paper selects Nanjing Lukou International Airport as an example for optimization verification. After simplifying the runway, taxiway, and terminal, abstract it into a network diagram consisting of nodes and taxiway segments, as shown in Figure 2:

![Figure 2. Simplified Nanjing Airport Scene Node Network Diagram](image)

The runway mode of Nanjing Airport is isolated operation, Runway 06 is used for take-off, and Runway 07 is used for landing. According to the airport operation rules, most of the arrival aircraft choose D3 fast turn-off to taxi off the runway, and a small part of the arrival aircraft choose D3 fast turn-off to taxi off the runway, while very few aircraft are off the track from the D4 fast turn-off. That is to say, the start node of the arrival aircraft is the intersection of the D3, D4, D5 fast turn-off and the runway 07, while the end node of the departure aircraft is the intersection of the taxiway K and the runway 06.

Depending on the true usage frequency of the fast turn-off, the node 110 is preferentially selected from the runway in programming, and if 110 is occupied, the node 113 is selected; if 113 is also occupied, the node 114 is selected. In the network diagram, there are a total of 172 nodes, which are represented by numbers 1-172. The taxiing path is represented by a node number, for example, the parking stand assigned to the arrival aircraft $f_i$ is 142, and its taxiing path can be expressed as 110-83-57-17-18-19-29-142. Based on actual operational experience, it is assumed that the aircraft is taxiing on the taxiway at a constant speed of 10 m/s.

In this paper, Win64's Matlab2017b is used to optimize the taxiing path of Nanjing Airport by using the improved A* algorithm. Finally, the optimal taxiing path of all flights during the peak hours of a certain day is obtained. The over-the-station flight is disassembled into the arrival flight and departure flight to optimize respectively.

The optimization results show that the taxiing conflicts of all aircraft throughout the day are solved, and there is no dead loop. In order to more clearly show the distribution of the taxi time after the optimization of the 100 arrival and departure aircrafts, the flight number is the abscissa and the taxi time is the ordinate, and the line chart of the taxi time of the arrival and departure aircraft are drawn separately. As shown in Figure 3-4.

![Figure 3. Arrival aircraft taxi time distribution map](image)
It can be seen from Fig. 3 that the taxi time of each arrival aircraft is relatively uniform. Figure 4 shows the taxi time distribution of the departure aircraft. Except for the long taxi time of the number 10 flight, the taxi time of other flights is within the range of 200-500 seconds, and the distribution is also relatively uniform. The reason why the number 9 flight has a longer taxi time is because its parking stand is far from the runway entrance.

4.1. Comparative analysis before and after optimization

The optimized average taxi time of each aircraft in Nanjing Airport is compared with the average taxi time of each aircraft under actual operating conditions. The results obtained are shown in Table 1:

| Flight type  | Average taxi time under actual operation /s | Average taxi time under optimized operation /s |
|--------------|-------------------------------------------|-----------------------------------------------|
| Arrival      | 265.8                                     | 182.1                                         |
| Departure    | 432                                       | 352.9                                         |

As can be seen from the above table, after optimization using the improved A* algorithm, the average taxi time of the aircraft is much shorter than the actual operation’s, in which the average taxi time of the arrival aircraft is reduced by 83.7 seconds, and the average taxi time of the departure aircraft is reduced by 79.1 seconds, to a certain extent, the taxi time is minimized.

5. Conclusion

In this paper, the operation mode of the airport surface is summarized firstly. The shortest aircraft taxi time is taken as the objective function. Considering the taxiing conflict constraint between the aircraft and the runway operation constraints, a joint optimization model of the taxiway and the runway is established. Secondly, according to the characteristics of the standard A* algorithm and the actual situation of the taxiway scheduling problem, the algorithm is improved, and the running system and the parking space of Nanjing Lukou International Airport are simplified. The improved A* algorithm is used to optimize the taxiing path of 100 aircraft in a certain peak period of Nanjing Airport. In the optimization process, the conflict avoidance is considered, and the taxiing path after optimization of each aircraft is obtained. Finally, it is compared with the taxiing data under actual operating conditions to verify the superiority of the optimization model.

References

[1] Pesic B, Durand N, Alliot J M. Aircraft ground traffic optimisation using a genetic algorithm[C] //Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation. Morgan Kaufmann Publishers Inc., 2001: 1397-1404.
[2] BALAKRISHAN H, JUNG Y. A framework for coordinated surface operations planning at Dallas-Fort Worth International Airport[C]//AIAA Guidance, Navigational, Control Conf. Exh., Hilton Head, SC, 2007, Paper AIAA-2007-6553.

[3] Chen J, Weiszer M, Stewart P, et al. Toward a more realistic, cost-effective, and greener ground movement through active routing—part I: optimal speed profile generation[J]. IEEE Transactions on Intelligent Transportation Systems, 2015, 17(5): 1196-1209.

[4] Tancredi U, Accardo D, Fasano G, et al. An Algorithm for Managing Aircraft Movement on an Airport Surface[J]. Algorithms, 2013, 6(3): 494-511.

[5] Roling P C, Visser H G. Optimal Airport Surface Traffic Planning Using Mixed-Integer Linear Programming[J]. International Journal of Aerospace Engineering, 2008, 2008(1): 116-125.

[6] Weiszer M, Chen J, Stewart P. A Real-Time Active Routing Approach via a Database for Airport Surface Movement[J]. Transportation Research Part C: Emerging Technologies, 2015, 8(58): 127-145.

[7] Guépet J, Briant O, Gayon J P, et al. The aircraft ground routing problem: Analysis of industry punctuality indicators in a sustainable perspective[J]. European Journal of Operational Research, 2016, 248(3): 827-839.

[8] Evertse C, Visser H G. Real-time airport surface movement planning: Minimizing aircraft emissions[J]. Transportation Research Part C: Emerging Technologies, 2017, 79: 224-241.