International Journal of Control and Automation
Vol. 9, No. 5 (2016), pp. 161-172
http://dx.doi.org/10.14257/ijca.2016.9.5.16

Disruption Scheduling of Gate in Response to Temporary Airport Closure

Peng Yang[1], Wei Gao[2], Junqing Sun[1], Man Liang[2] and Yuanquan Wang[1]

[1] *School of Computer Science and Technology, Tianjin University of Technology, Tianjin 300384, China*
[2]*Civil Aviation University of China, Tianjin 300300, China*
E-mail: pqxx@sina.com

Abstract

Under serious conditions, airport will experience temporary shutdown, and while the airport reopen, the controller must make gate reassignment. In order to get intelligent use of the gate resource, the model was studied both considering operational smoothness and passenger satisfaction. The problem was formulated as a binary quadratic programming problem, and then, the model was transferred into dynamic programming (DP) model to solve by transformation, and the complexity of the DP algorithm is analyzed. Later, an improved Tabu Search algorithm (TS) was designed, and the greedy strategies were mixed to speed up the solving speed under large scale problem. At last, numerical simulation results demonstrate the effectiveness of the model and the improved TS algorithm.

Keywords: *Airlines, Transportation, Gate reassignment, Tabu search, Dynamic programming*

1. Introduction

Airport gate assignment is to appoint terminal or ramp position, called gates, for the arrival or departure flight and to ensure that the start and completion times of the processing of a flight is at its position. In practice, flight delays, severe weather, and many other unforeseen scenarios are all possible disturbing factors to airport operation. Under some serious conditions, the airport will experience temporary closures, which have a big negative impact on airport operation. They can lead widespread flight delays, which case both the directly affected and related flights need to be appropriately reassigned to alternate gates. If a disruption does happen in the near future, decisions must be made in a timely fashion to minimize the impact on the gates operation while keeping the customer satisfaction level. Because of the tightness of the flight schedules, the demand for effective reassignment will further increase in the following years.

The goal of this research is the development of a practical method by which gate reassignments can be made in response to the disruption, and make an improvement over the current limited manual process. The remainder of this paper is organized as follows. In Section 2, a brief introduction of the related literature is provided. In Section 3, a general description of the bushiness constraints of gate operation at airports is presented, then, the gate reassignment problem is formulated as an over constraint resource assignment problem. Section 4 contains the algorithmic approach description, first, a dynamic programming is formulated to get the optimal solution, and then a heuristic algorithm is created to speed up the solution process. In Section 5 is devoted to the experimental analysis of case studies of the model and algorithms. Finally, some conclusions are offered in Section 6.

ISSN: 2005-4297 IJCA

2. Literature Review

Gate assignment problems are explored in two ways, namely as planned assignment and as reassignment problems.

2.1. Planned Gate Assignment Problem

Planned Gate assignment problem has already been studied by many researchers, and this problem is considered as the constraint resource assignment problem where the airport gates are resources and flights are resource consumers. Much work has been centered on the gate assigning problem with the objective of minimizing distance cost[1], and this motivation is easy to understand. Later, many other objects were taken into consideration[2,3]. Yu *et al.*[4] established a multi-objective model. Both minimizing the total walking distance of all passengers and balancing the average walking distance of passengers among different airlines were taken into account. However, the scope of the model only concentrated on the domestic flights. Wei *et al.*[5] created a model, which both operational safety and efficiency were taken into consideration, proactive approach was adopted to avoid conflict of power-in vs. push-out based on the analysis of the operational process of aircrafts on the apron.

Flight schedule with flight arrival and departure times is subject to uncertainty and may change over time, so in recent years, the robustness of the gates schedule problem is also take into concern. Buffer time is the time when the gate is unoccupied by flight and it is the most popular strategy to apply buffer time to incorporate the robustness into gate scheduling. Zhao and Cheng [6] proposed a mixed integer model to formulate the robust assignment problem and the objective of the model is to find balance between efficiency and robustness. However, the application of buffer time is on the production of a good planned assignment rather than a good reassignment.

The basic gate assignment problem is quadratic assignment problem as shown to be NP-hard, so many heuristic algorithms are designed to this problem. For example, Xu and Bailey[7] provided a TS heuristic to exploit the special properties of different types of neighborhood moves, and created effective candidate list strategies. Wei and Liu[8] created a GATS algorithm combining genetic algorithms with tabu search to solve the proposed gate assignment model.

2.2. Gate Reassignment Problem

Another type of Airport Gate Assignment work is in the flight gate reassignment due to the flight delay or airport disruptions, i.e. to adjust the flight assignment due to the changes of the original daily flight schedule.

Although there has been much work done on planned assignment, there has so far been a few research on reassignment problems [9, 10], and the research are mainly focused on the case of temporary gate shortages with stochastic flight delays. For example, Gu and Chung[11] designed a genetic algorithm (GA) approach to the problem of gate reassignment problem, which is induced by flight delays. Validation for the GA is performed through comparison with manual results of gate managers.

Maharjan and Matis [12] formulated a binary integer program to the problem of gate reassignment in response to flight delays. The objective of the model was to minimize the walking distance of passengers between disrupted flights. Tang[13] designed a gate reassignment model to deal with temporary gate shortages and stochastic flight delays, and the disturbance values of the model are designed as a linear function. The objective function was designed to minimize the expected disturbance values of all scenarios, and the model would judge which flights needed to change gates or be delayed for system optimization.

These studies primarily deal with reassignment after real-time flight delays, rather than temporary airport closures. In comparison with the temporary gate shortages caused by flight delays, airport closure will incur longer perturbation time, more perturbed flights and gates, lack of contract gates, and there is still much need to do on this problem.

3. Problem Definition and Model

3.1. Problem Definition

Before the creation of the mathematical model, the gate reassignment purpose and the business constraints should be considered.

Business Constraints: Every flight must be assigned to one and only one gate or assigned to the apron, and it is prohibited that any two flights are assigned to the same gate simultaneously. If more than one activities overlap, they are not allowed to be assigned to the same gate.

Gate Compatibility: In practice, an airport usually has several different types of gates and each type of gates is restricted to be able to accommodate only certain types of planes. For example, there are small, medium, heavy gates at Tianjin Binhai International Airport. Large gates can be used for all types of flight, while small gates are only available for small flights.

Time Window Constraint: A departing flight should have enough time to get ground service prior to departure, such as fueling, passenger and luggage boarding. So the departure time minus the starting time of the flight should be bigger than the service time window and half of the buffer time.

Reassignment purpose: As the results of some unforeseen incidents, gate reassignment has to be made to obtain a new flight to gate scheme according to original assignment, which can guarantee the safe and smooth operation and better passenger services. To this end, the controller should regard minimizing the walking distance of passengers and the disruption of the original gate plan as the ultimate target.

Because the original assignment has already been created, the disruptions of the flight schedule that happen in the near future may affect the ground service providers' preparation at the target gates. While the authorities make flight gate reassignment during actual operations, they have to consider the disturbances to the original planned assignment and reduce the possible disturbances caused by such deviations, especially when flight schedules are tight.

To provide high-level services to the customers, the airport authority needs to assign the flights to gates so as to reduce the walking distance. However, it is possible and frequent for an airport to face the problem of not enough available gates after a relatively long closure. If too many flights arrived at the airport in a short-time slot, some of the flights may have to be assigned to the remote stands, and shuttle buses will be applied to pick the passengers to the terminal block.

As the number of flights exceeds the number of available gates, the reassignment problem is an over-constrained case. Under over constrained case, the distance from the remote gates or tarmac area to the terminal must be taken into consideration and the assignment to remote gates is undesirable.

In practice, two major types of disturbance may be caused by reassignment, namely space disturbances and time disturbances[14]. After temporary airport closures, the time disturbance will be inevitable, so only space deviation is considered in our model.

3.2. Model Assumption

To make the reassignment problem tractable and solvable, some simplifying assumptions will have to be made. However, this approach can often solve the realistic problem. The assumptions used in this issue for the gate reassignment model are listed as follows.

1) There are contract gates and remote stands at the airport. While there are not

contract gates available, the remote gates are sufficient. If a flight can not get a contract gate, it can definitely get a remote one. All the remote gates are defined as a dummy gate with infinite capacity in our model.

2) Flights are classified into 3 types: Heavy, Middle, and Little, so do the gates. Large flights normally are not allowed to park at a small gate or a middle gate, while a big gate is available for small flights. While it comes to the dummy gate, it can hold any types of flights.

3) The planned gate assignment and flight schedule indicating the flight arrival and departure time is known in advance.

4) For simplicity, the apron traffic constrains, such as push-out conflict and neighboring of flights, are not considered.

It needs to mention that during the period of closure (i.e. between the closing time and the reopening time) some flights that are originally scheduled to use gates at the airport may change their plan and fly to nearby airports. These kinds of flights are not considered in our model.

Parameters Definition

To illustrate the gate reassignment problem in a mathematical form, the following notation should be define, which would be use throughout this paper. It'd be noted that |X| means the cardinality of an arbitrary set X.

K Set of available contract gates to which the flight can be assigned.

N Set of the flights that have arrived at the airport before the closure and still stay at the gates.

M Set of flights that will arrive in the immediate time slot after the airport reopen.

K_i Set of gates can't be assigned to flight i for some particular reason, beside gate compatibility restriction.

VK Set of the remote gate, in our model all the remote gates are regarded as a dummy gate with infinite capacity. This denotes that |VK|=1.

AK Set of all the available gates at the airport. AK= $K \cup VK$

Parameters

A_i Expected arrival time of flight i.

S_i Ground service time of flight i plus the buffer time.

F_i Type of the flight, if F_i =1, then it is small flight; if F_i =2, then it is middle flight; if F_i =3, then it is heavy flight.

G_k Type of the contact gate, if Gk =1, then it can only hold small flight; if Gk =2, then it can serve both small flight and middle flight; if Gk =3, then it can serve all the flights. The remote stand can hold all types of flights.

$d_{kk'}$ Distance from gate k to gate k'.

b Buffer time between consecutive flights .

w_{ij} Walking distance for passenger from gate i to gate j, the travel distance from remote stand to any gates is set as twice of the max distance between gates.

$f_{i,j}$ Number of passengers needed to transfer from flight i to flight j.

H Sufficiently large coefficient.

$x_{ik}^{'}$: Binary variable, with $x_{ik}^{'}$=1 if flight i is assigned to the gate k, at the original scheme, otherwise $x_{ik}^{'}$=0.

Decision Variables

x_{ik} : Binary variable, with x_{ik} =1 if and only if flight i is assigned to the gate k, at the reassignment scheme, otherwise x_{ik} =0.

z_{ijk} : Binary variable, with z_{ijk} =1 if and only if flight i and flight j are assigned to the gate k and flight j immediately follows flight i at the reassignment scheme, otherwise z_{ijk} =0.

Mathematical model

Through the aforementioned discussion, the airport gate reassignment problem can be formulated as the over constraint resource assignment problem, where gates serve as the limited resources and flights play the role of resource consumers.

$$Min \quad f = \alpha_1 \sum_{i \in M} \sum_{j \in VK} x_{ij} + \alpha_2 \sum_{i \in M} \sum_{k \in AK} \sum_{k' \in AK, k \neq k'} \left(\delta_{ik}^{+} + \delta_{ik}^{-}\right) \cdot d_{kk'} + \alpha_3 \sum_{i \in M} \sum_{j \in AK} f_{i,j} \cdot w_{ij} \cdot d_{ij} \cdot x_{ij} \quad (1)$$

Subject to：

$$\sum_{j \in AK} x_{i,j} = 1, \quad \forall i \in M \quad (2)$$

$$A_j \geq A_i + S_i + H\left(x_{jk} + x_{ik} - 2\right) + b, \quad \forall i \in M \cup N, \forall j \in M, \forall k \in K$$
(3)

$$x_{ik} - \delta_{ik}^{+} + \delta_{ik}^{-} = x_{ik}^{'}, \quad \forall k \in AK, \forall i \in M \quad (4)$$

$$F_i \cdot x_{ik} \leq G_k, \quad \forall k \in K, \forall i \in M \quad (5)$$

$$x_{ik} = x_{ik}^{'}, \quad \forall i \in N, \forall k \in AK \quad (6)$$

$$x_{ik} = \sum_{i \in M} z_{ijk}, \quad \forall j \in M, \forall k \in K \quad (7)$$

$$x_{ik} = \sum_{j \in M} z_{ijk}, \quad \forall i \in M, \forall k \in K \quad (8)$$

$$x_{ik} = 0, \quad \forall i \in M, \forall k \in K_i \quad (9)$$

$$x_{ik}, x_{ik}^{'} \in \{0,1\}, \quad \forall i \in N \cup M, \forall k \in AK \quad (10)$$

$$\delta_{ik}^{+}, \delta_{ik}^{-} \in \{0,1\}, \quad \forall i \in N \cup M, \forall k \in AK \quad (11)$$

In the objective function (1), the first part is designed to minimize the flights that be assigned to the dummy gate. The second part is designed to minimize the diversion from the original plan. If a flight's original assignment decision has to be adjust, then the impaction to the ground service should be minimized. $d_{kk'}$ is applied in the model so that the flight find the alternate from nearby gates. The last part of the objective function is designed to directly minimize the overall passenger walking distance and guarantee the service level, that's to make sure the flights that assigned to remote stands carry the less number of passengers. The parameters α_1 α_2 and α_3 are introduced to balance the importance of the objectives in the objective function.

Constraint (2) is the flight constraint, namely one flight must get a gate to be served.
Constraint (3) indicates the gate constrain. H is introduced here to make sure that no more than one flight can be assigned to one contact gate. While it comes to the dummy gate, it proposes unrestricted capacity, and even two overlapped flight activities can be

assigned to the dummy gate.

Constraint (4) is the balance constrain. Although only x_{ij} is choose as the control variable, but δ_{ik}^{+} and δ_{ik}^{-} will be influenced by x_{ij} directly.

Constraint (5) is the gate compatibility constrain, which makes sure that a small gate will not hold big flight.

Constraint (6) means that the assignment that have already realized can't be changed over time.

Constraint(7) and constraint (8) stipulate that any flight can have no more than one flight immediately following and at most one immediately preceding flight, at one gate.

Constraint(9) indicates that a flight cannot be assigned to a gates that don't satisfy operation restrictions.

Constraint (10) and constraint (11) specifies the binary requirement for the control variables and state variables.

It should be mentioned that there are some additional constraints in the real operations are out of consideration in the current formulated model. However, such preference constraints can easily be added to the model. In addition, some of the original flights may be delayed or canceled, and the flight delay times due to schedule changes can be calculated outside the model.

4. Solution

As the proposed model is a quadratic assignment problem (QAP), it is one of the great challenges in combinatorial optimization. It is well known that QAP is NP-hard [15]. It was also proved that finding an approximate solution of QAP is also NP-hard and is difficult to solve in short time with an exact method, such as dynamic programming (DP).

4.1. Dynamic Programming

DP is a powerful technique that can be used to solve many problems. To solve our problem through DP, the problem has to be broken into a reasonable number of sub-problems, in such a way that optimal solutions can be used to the smaller sub-problems, which are typically overlapped, to give us optimal solutions to the larger ones.

The most important items of DP are the stages and the states. To solve our model through DP, the following components are created.

Stages: Without loss of generality, the set of newly arrival flights M are sorted in ascending order concerning their A_i. If $A_i = A_j$, $i \neq j$ then it is arbitrarily chosen which flight is first. The arrival times are not necessarily equidistant. There are |N|-1 time periods defined by starting time A_i and A_{i+1}. Hereby period i defines the time span between time points i and i+1, because the randomness of the flight.

The time periods might be of different length and in general they do not correspond to the ground times of the flight activities.

1) A decision function for each period, whose codomain depends on the state of the period. At each stage, the assignment of flight has to be decided, to minimize the sub-problem.

2) An objective function for each period. This corresponds to the preference value of the flight activity that has to be assigned in the according period. The sum of the period objective functions is the overall objective function. Each period objective function only depends on the period decision function.

3) For each period, there is a state variable, which only depends on the decision and the state of the previous period, i.e. a transformation function describes the transition from state i-1 to i.

State variables: state of gate j at stage i can be defined as c_{ij}, if the gate is busy at stage i then c_{ij}=1, otherwise c_{ij}=0. So, the airport state at stage i can be defined as $C_i = \left(c_{i1}, c_{i2}, \cdots, c_{i|VK|}\right)$. The state C_i only depend on the decision and state of the previous state, it describes which and how long gates are blocked for activity i, considering that previously assigned flight activities might block some gates.

Decision variable at a stage: $X_i = \left(0, \cdots, 0, 1, 0, \cdots, 0\right)$ is the decision variable, which means that the flight is assigned to the very gate at stage i. In this way, the x_{ik} can be decided directly, and z_{ijk} can be determined implicitly.

Forward recursive formula: We have the state transition equation that $C_{i+1} = t\left(C_i, X_i\right)$, which means that the state of stage i+1 are only affected by the previous state and the decision made at stage i. The stage cost function at stage i can be expressed as $r_i\left(C_i, X_i\right)$. If an optimal solution to the problem contain within it optimal solution to sub-problems, the forward recursive formula of the reassignment problem can be expressed as

$$f_{i+1}(X_{i+1}) = \min\left[r_i\left(C_i, X_i\right), f_{i-1}(X_{i-1})\right]$$

Algorithm complexity: The complexity of DP for this problem depends on the number of gates in all periods. In each period, for a given state, the objective function can be evaluated in linear time. Because the objective function has to be determined for each possible state, it has to be determined at most $\left(|AK|-1\right)^{\left(|AK|-1\right)}$ times in each period. Thus, the complexity of the algorithm comes down to $\Theta\left(|M| \cdot |AK|^{|AK|}\right)$. As can be seen from the results, a high number of available gates increases the runtime drastically.

4.2. Improved Tabu Search Algorithm

As is the case for the DP algorithm, the overall problem is simplified by considering a subset of the full solution space, but convergence of the algorithm to an optimal solution requires a very large number of subset optimal solutions. If an airport with a big number of gates is considered, the performance of DP algorithm will turns to be poor, even with relatively less flights, although it can delivers exact solutions. In practice, a hub airport may accommodate more than 100 flights at the same time. To that end, heuristic method should be applied to get the near optimal solution within a reasonable computing time.

The tabu search (TS) algorithm is a meta-heuristic algorithm introduced by Glover, and is known to be an effective tool for combinatorial optimization problems. In our work, two kinds of exchange moves are created in QAP, internal exchange move and apron exchange move.

Internal exchange move: Internal exchange move means the flights that park at contract gates, exchange their position. It need to be mention that under some condition no single flights exchange or consecutive flight pairs exchange can provide feasible solutions. To get feasibly solution at each internal exchange move, the exchange rule should be modified. If no single to single flight move can create a feasibly solution, then many to many move will be take into consideration, which makes moves to be more variable, and more diverse neighborhoods can be search to generate good quality solutions easier.

Apron exchange move: Apron Exchange Move is used to deal with the flights assigned to the apron. In each move, one of the flights, which is currently assigned to the remote gate, exchange its position with a flight that is assigned to a contract gate.

The TS searches the neighborhood of the current solution using short term memory. At each iteration, feasible neighboring moves are evaluated, and a move that improves objective value is used to find the next solution. In each iteration of TS, no less than one move is generated. The move can be one of the above-mentioned moves with a given probability. The neighborhood move is accepted when the objective function value can be improved. If there is no improvement, the solutions can also be accepted with probabilistic criterion.

Greedy strategy: Greedy is a strategy that works well on optimization problems. It follows the problem solving heuristic of making the locally optimal choice at each stage with the hope of finding a global optimum. It is shortsighted in their approach in the sense that they take decisions on the basis of information at hand without worrying about the effect these decisions may have in the future.

Greedy strategy for initial: A good idea is to find a near optimal initial solution to begin the search iteration. A greedy method introduced by Ding *et al.* [16] is applied to get the initial solution. It sorts flights in accordance with the original schedule and current condition and minimize the size of the flights set that are assigned to the dummy gate. There are two cases should be considered.

1) According to the original plan, the flights that be assigned to the remote aprons are assigned to the dummy gate;

2) According to the original schedule, if a gate assigned to a flight is idle, then the flight can take the gate, otherwise the flight will be assigned to the dummy gate.

The greedy solution can not only give us a feasible initial solution, but also help us to speed up the search process, which will be used in the heuristic algorithm.

Greedy strategy in exchange move: In consideration the reassignment purpose, it is most often desirable to make reassignments within the same terminal as which the original assignment was made to avoid extremely big impact on the ground operation. Regardless of the nature of the restriction, the potential optimal solution of this mathematical program may be restricted to a nearby gates subset of the original gate where the best reassignments strategy should be made within the subset, so we concentrate our search on the subset. The choice made by a greedy algorithm may depend on choices made so far but not on future choices or all the solutions to the sub-problem. In other words, a greedy algorithm never reconsiders its choices. This is the main difference from dynamic programming, which is exhaustive and is guaranteed to find the solution. By the way, the minimal number of flights left to the remote gate is determined by the greedy algorithm, the “ many− many” exchange is unnecessary to apron exchange move.

In order to escape from possible local optima, a random search at other regions is also need to be performed which not only accepts changes that decrease the objective function, but also some changes that increase it. The random search and acceptance is controlled by a probability p.

5. Numerical Illustration

In this numerical illustration, we use simulation to determine the validity of the proposed model and demonstrate the efficiency of the designed algorithm, which can deliver an appropriate solution in relatively short runs. We implement DP algorithm and heuristic algorithm in JAVA and run them on an Intel P4 2.6G with 2 GB RAM in the environment of Microsoft Windows XP.

The example problem is considered in this case study to conduct simulation although the model being developed is independent on the problem structure. The example is drawn partially with minor modifications from a hub airport layout structure. The airport is one of the busiest airport in total passenger traffic in China. The airport has 120 gates and handles more than 1300 daily flights, and only 40 gates are considered in the simulation, one dummy remote stand and 3 contract gates.

Passenger behavior has dynamic pattern in each flight at each day, so we make realistic approximations to simplify the problem to simulate the passenger performance.

Random distributions, obtained from history data, are applied to create passenger transfer between flights, and the passengers are assumed to only walk horizontally or vertically. The distances between gates that used in this example are accurate, yet the number of gates at remote stands is only reflective of the true condition, which is used as for the evaluation of model.

Suppose the airport is shut down at 8:0 AM and reopen at 9AM, for serious weather condition, and it's hypothesized that upon the reopen of the airport, the set of information of flights become known. In different cases, the quantities of flights that need to be assigned vary from 175 to 245, and the original gate assignment theme turns infeasible, and reassignment of the gates turns inevitable. The actual arrival time of flight i is randomly generated in the interval [A_i , A_i +10]. This interval models a flight landing every 10 min and the average service time of different kinds of flights are showed in Table 1. Such kind of service performance is very typical at large airports of China.

Table 1. Service Time

Type	Service time(min)
Heave flights	60
Large flights	50
Small flights	40

Results and Analysis

Evaluation of the proposed model given in this issue is different from the validation of classical models, brute-force method cannot be regarded as a standard. An acceptable model is one that can create better solution than current manual way, while the validation objective of other classical models is to determine the model represents reality. To validate the proposed mathematical model, we also use the airport manual method to create the reassignment scheme. For the simulation test, several scenarios are generated and the scenarios are not only imputed to the model and solved by DP algorithm and improved TS algorithm, but also presented to a group gate manager and solved manually. To judge the quality of the reassignment scheme, we evaluate the solutions through the proposed target function(1), as can be seen in Table 2. As is showed in the table, the max relative error of improved TS algorithm is 3.29% and the average relative error is 1.30%, while it comes to the manual results, the max relative error is 17.90% and the average relative error is 10.58%. Although the improved TS algorithm cannot get the optimal solution as DP, the results of this algorithm can also generate better results than the experienced gate managers. Most of results we get under different cases, the relative error converged to within 1.5%, which is good enough in terms of solution quality.

Table 2. Results of Objective Function

Experiment	Quantity of flights	Manual results (r_M)	DP results(r_D)	Improved TS results(r_I)	Difference between r_M and r_D (D_D=r_D - r_M)	Difference between r_M and r_I (D_I= r_I - r_M)
1	175	31.84	28.73	29.32	3.11	0.59
2	180	31.54	29.15	30.11	2.39	0.96
3	185	38.57	30.14	30.78	8.43	0.64
4	190	32.95	31.72	31.78	1.23	0.06
5	195	36.71	32.53	33.17	4.18	0.64
6	200	36.88	34.61	34.61	2.27	0
7	205	38.17	35.41	36.13	2.76	0.72
8	210	41.64	36.42	37.04	5.22	0.62
9	215	39.31	36.72	37.14	2.59	0.42
10	220	39.87	37.31	37.91	2.56	0.6
11	225	45.43	42.75	43.1	2.68	0.35

12	230	47.67	44.16	44.23	3.51	0.07

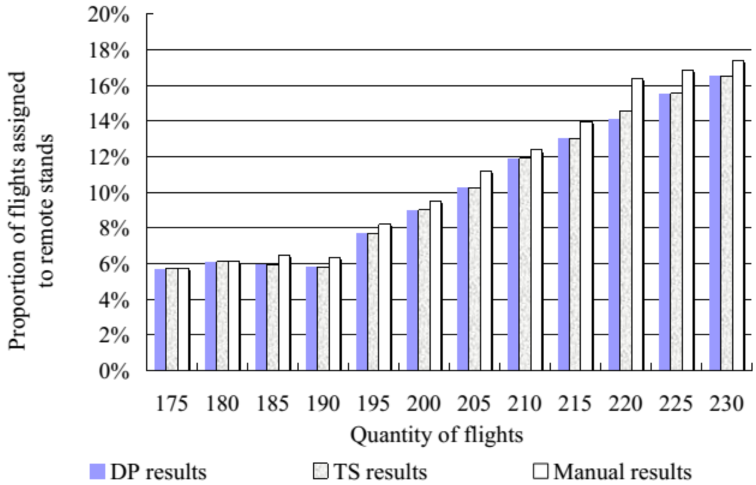

Figure 1. Proportion of Flights Assigned to Remote Apron

As can be seen from Figure 1, in comparing with the optimal results that we get through DP algorithm, the results of improved TS algorithm is rather promoting. In most cases, the performance of the improved TS algorithm is exactly equals to that of optimal solution. The max relative error is only 3.2%, and the average relative error is 0.4%. When it comes to the manual method, the max relative error and average relative error are 10.0% and 6.9%. With the increase of the flight quantity, after the reopen of the airport, more flights will be assigned to remote apron, but the performance of the algorithm outperform the manual counterpart.

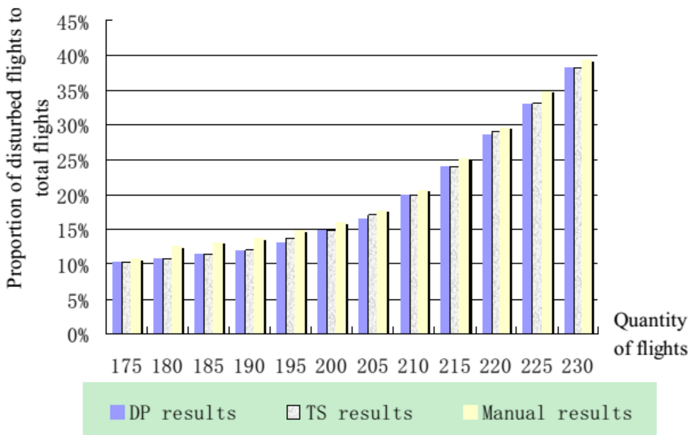

Figure 2. Ratio of Disturbed Flights to Total Flights

With the increase of the flight quantity, which means the airport experience longer closure, in these cases, even the optimal solution considerably deviates from the planned gate assignment scheme. As can be seen from Figure 3, the proportion of flights, reassigned by DP algorithm, which need to be reassigned raises from 10.2% to 53.7%, and we can get similar results through improved TS algorithm. As size of the problem increase, the gap between the optimal results and the manual method is obviously, which means that there are difference between the object and the optimal solution. In average, the manual method yields 3.1 unnecessary gate changes on each reassignment. The tendency of overall passenger walking distance are exactly same, so the detail results are not provided.

A validated method should be able to yield an appropriate scheme in reasonable runs, but the exact time cost of reassignment process conducted by experienced gate managers is hard to get, because the managers often need to get the flights information in advance, and nobody can judge at which time point do they reach an agreement on the final solution. So, we only present the results of improved TS algorithm and the original TS algorithm, which don't apply greedy strategies.

Table 3. Results of Time Costs

Experiment	Number of flights	Number of gates	Improved TS time costs(s)	original TS time costs(s)
1	100	15	7.79	7.88
2	150	20	21.34	21.43
3	200	25	51.36	45.67
4	250	30	85.07	76.51
5	300	35	167.66	121.92
6	350	40	297.65	212.78
7	400	45	472.69	312.25
8	450	50	683.12	407.55
9	500	55	992.70	531.83
10	550	60	1349.84	714.80

As can be seen from the Table 3, in this set of test cases, when the size of the problem is small, the improved TS algorithm does not get better results than the original TS algorithm, which does not apply greedy strategy, and the greedy strategy even leads relatively longer time costs. When it comes to problem with large scale, the application of greedy does make sense, which speeds up the search process, and the improved TS algorithm shows an obvious shorter running time. For the DP algorithm, as the size of the problem increase, especially the increase of gates, the time costs increase drastically, and can't provide a solution in reasonable runs, so, the DP approach still have trouble to be applied to arbitrarily large airports with many gates and busy traffic.

From these results, we conclude that the gate reassignment model offered a significant improvement over the current manual method, making it very efficient for real-time operations. These results also show that the model can be efficiently solved using improved TS algorithm.

6. Conclusion

The paper presents a new idea for the airport gate reassignment problem in response to temporary airport closure. Through our primary objective is to minimize the overall impact on the ground operation that reduce the overall distance between the original scheme and the new schedule, we have also attempted to improve the passenger's satisfaction by minimizing the overall walking distance. First the problem is formulated as an over constraint resource assignment problem, then it is reformulated as a DP problem and the complexity of the algorithm is evaluated. To solve the large scale problems within a reasonable time bound, a hybrid heuristic algorithm is designed through mixing of greedy strategies and TS algorithm. At last, numerical simulation results demonstrate the effectiveness and efficiency of the model and the improved TS algorithm, the results can be applied in real time. According to the test results, we can draw some conclusions. (1) Compared to the manual assignment, this model can reduce the whole passengers' walking distances and improve the fairness between airlines at the same time. (2)Although the DP can get exact results, it cannot be applied to large airport with many gates. (3)The designed greedy TS algorithm can reduce computational complexity without much loss of accuracy.

The model proposed in this paper only addressed disturbances that were due to temporary closure, rather than other types of disturbances. Therefore, adopting the model to fit other types of disturbances can be a direction of future research. The research scope of the paper does not consider the impact on cargo operation. How to combine with effective resource schedule can be further researched.

Acknowledgements

This work is financially supported by National Nature Science Foundation of China under the Grants 61039001, and National Science technology Support Plan Projects under the Grants 2011BAH24B10.

References

[1] U. Dorndorf, A. Drexl, Y. Nikulin and E. Pesch, "Flight gate scheduling: state-of-the-art and recent developments", Omega, vol.35, **(2007).**
[2] F. Cheng, M. Hu and Z. Zhao, "A new optimization model of airport gate assignment", Journal of Transportation Systems Engineering and Information Technology, vol. 12, **(2012)**.
[3] X. Jie and Z. Chen, "Airport gate assignment with airplane taxiing cost analysis", Journal of Transportation Systems Engineering and Information Technology, vol. 10, **(2010)**.
[4] J. Yu, L. Zeng, and Y. Luo, "Multiobjective Gate Assignment Based on Passenger Walking Distance and Fairness", Mathematical Problems in Engineering, vol. 2013, **(2013)**.
[5] D. Wei, C. Liu and X. Li, "Airport gate assignment problem with push-out conflict-avoidance", Mathematics in Practice and Theory, vol. 42, **(2012)**.
[6] H. Zhao and L. Cheng, "Ant Colony Algorithm and Simulation for Robust Airport Gate Assignment", Mathematical Problems in Engineering, vol. 2014, **(2014)**.
[7] J. Xu and G. Bailey, "The airport gate assignment problem: mathematical model and a Tabu search algorithm", Proceedings of the 34th Hawaii international conference on system sciences, Maui, USA, **(2001)**, January3-6.
[8] D. Wei and C. Liu, "Multi-objective model and optimization for airport gate assignment problem", Proceedings of the 2012 International Conference on Information Technology and Software Engineering Lecture Notes in Electrical Engineering, **(2013)**.
[9] T. Pan and C. Wey, "An efficient gate reassignment algorithm in post technology mapping", Proceedings of the 39th Midwest Symposium on Circuits and Systems, **(1996)**, Lowa, SA, August 18-21.
[10] T. Pan and C. Wey, "GRASS: an efficient gate re-assignment algorithm for inverter minimization in post technology mapping", IEEE Proceedings, Computers and Digital Techniques. vol. 144, **(1997)**.
[11] G. Yu and A. Chung, "Genetic algorithm approach to flight gate reassignment problem", Journal of Transportation Engineering, vol. 125, **(1999)**.
[12] B. Maharjan and T. I Matis, "An optimization model for gate reassignment in response to flight delays", Journal of Air Transport Management, vol. 17, **(2011)**.
[13] C. H. Tang, "A gate reassignment model for the taiwan taoyuan airport under temporary gate shortages and stochastic flight delays", Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transaction, vol. 41, **(2011)**.
[14] B. Maharjan and T. I. Matis, "An optimization model for gate reassignment in response to flight delays", Journal of Air Transport Management, vol. 17, **(2011)**.
[15] Y. Xia, "Improved Gilmore-Lawler bound for quadratic assignment problems", Chinese Journal of Engineering Mathematics, vol. 24, **(2007)**.
[16] H. Ding, A. Lim, B. Rodrigues and Y. Zhu, "The Over-constrained Airport Gate Assignment Problem", Computers and Operations Research, vol. 32, **(2005)**.

