

# Fleet Assignment Problem Study Based on Branch-and-bound Algorithm

Wu Donghua

College of Continuing and Education  
Nanjing University of Aeronautics and Astronautics  
Nanjing, China  
wdh@nuaa.edu.cn

Xia Hongshan

College of Civil Aviation  
Nanjing University of Aeronautics and Astronautics  
Nanjing, China  
xhsca@nuaa.edu.cn

Fan Yongjun

College of Economics & Management  
Nanjing Agricultural University  
Nanjing, China  
Fyj8888@yahoo.com

Zhang Jinyuan

College of Continuing and Education  
Nanjing University of Aeronautics and Astronautics  
Nanjing, China  
nj-zhgy@126.com

**ABSTRACT:** In this paper, a strategy based on Branch-and-bound algorithm for an optimization problem was proposed, a scheduling method solving airliner fleet assignment problem was analyzed. By building a model, the requirement of balanced application of airplanes was transformed to optimizing schemes. The graph's depth-first search and Branch-and-bound algorithm were presented. The algorithm was used in an example. As the result, the time cost of the optimized fleet assignment for 70 flights a day with 12 A320 aircrafts was less than 0.3 second, which was one order of magnitude less than that of the genetic algorithm. The experiment results show that the method is an effective way for fleet assignment problem and has high practical application value.

**Keywords-** *Branch-and-bound algorithm; fleet assignment problem; genetic algorithm; building mode*

## I. INTRODUCTION

Branch-and-bound algorithm (B & B), an improved method based on the enumeration method can effectively solve some of larger issues. Since this method is flexible and easy to use in computers, it can not only be described as an integer linear programming (or mixed integer linear programming) problem, but also be used in almost all optimization problem. At present, the Branch-and-bound algorithm has been successfully used to solve production scheduling problems, traveling salesman problem, location problem, knapsack problem and the limited number of feasible solution of many other issues, and has been widely used in many fields, such as computer science, applied mathematics, management science and other related disciplines.

Solving the fleet assignment problem (FAP), due to the large number of flight scheduled each day, has always been a

challenging task for the airlines. Essentially it is a combinatorial optimization problem, which is well-known NP-hard problem. Reasonable scheduling of fleets not only helpful for the aircrafts and on-time running, but also can improve fleet utilization, and reduce operating and maintenance costs effectively.

In U.S., hub-and-spoke structure is adopted in most of the airline's route networks. The flight frequency among the hub and spoke cities is high, daily flights plan is almost same; the key constraint of the flight scheduling is the "4-day maintenance rule."<sup>[3]</sup> However, in China, the airline's operations is mainly based on a single point to point hub mode. Flight plans daily in a week are quite different. More restrictions should be considered in fleet assignment problem. [4]. Therefore, the present study overseas is not suitable for the situation in our country. We must find a feasible way, which can be used in the operation of our airlines.

Domestic algorithms for fleet assignment problem include simulated annealing algorithm, Ford Fulkerson algorithm and the Hungarian algorithm, proposed by Sun Hong, which were used to solve the fleet assignment problem based on the minimum fleet available, fleet balance application and fleet dispatching commands. An example of the above algorithms was illuminated in his "Mathematical programming model of aircraft assignment" with 20 aircraft, 35 flights and the result time is close to 23 seconds<sup>[5]</sup>. Zheng Yun etc. introduces ant algorithm into the course of solving the aircraft-arrangement problem in civil aviation<sup>[2]</sup>. Wang Wei, Wang Jinbiao discussed flight combination problem by partitions and refinement<sup>[6]</sup> etc. These algorithms provide the ideas for the solution of the problem, but most of the algorithms have some disadvantages, such as parameter setting is complex, convergence is slow, and just local

optimal solution can be accessed because of the limit of the algorithms.

The aim of this paper is to present a optimization model and there control strategies based on Branch-and-bound algorithm for the FAP as well as to suggest some future research directions in this arena. In the following example, 68 flights are assigned to 12 A320 aircraft, which the average time-consuming is less than 0.3 seconds. To compared with genetic algorithm, the control strategies based on Branch-and-bound algorithm can solve this kind of problem more efficiently.

## II. BRANCH-AND-BOUND ALGORITHM IDEAS AND CONTROL STRATEGIES

Branch-and-bound algorithm is equivalent to a search on a tree, which has a depth of  $n$ . Each node in the tree has a task of solving a linear programming problem. After each bound to the current search tree, the lower bound of all the leaf nodes should be compared to the lower bound to find the smallest node, this node shall be the next branch node. Branches will gradually find a feasible solution; the current feasible solution and its value will be stored. If the branch node's value is less than the lower bound of the value stored, branching will be continued, else the algorithm is terminated. Thus the stored solution is the most optimal one. It can be seen that Branch-and-bound algorithm is a proper search on all possible solution space of optimization problem with constraint condition. "Branch" creates the conditions for the emergence of the optimal solution, and "bound" can improve search efficiency.

To further improve efficiency, control strategies are proposed in this paper. Firstly, we should analyze the practical problems, preprocess the raw data, and then search in the possible area. In this way, the search space can be narrowed. Secondly, in order to search for the optimal solution as soon as possible, an initial upper bound for the branch of the objective function value in integer programming is set, if the objective function value higher than the upper bound of the branch during the corresponding linear programming problem, it can be discarded as soon as possible. During the process of the branching, we do not solve the series of branching from the same point, but only first recorded one branch, then along the other branch the search continues, until all search on this branch is completed. If there is no feasible solution of the case, the smallest lower bound of branch points will be selected in the solution, and then continue searching along the branch. Finally, in the process of the branch, the largest integer variable is chosen in coefficient in the objective function, so integer optimal solution can be found as soon as possible.

## III. APPROACH FOR THE FLEET ASSIGNMENT PROBLEM

### A. Description of the fleet assignment problem

Fleet assignment problem is to decide which aircraft is assigned for each flight according to flight plan, maintenance and status information provided by the maintenance department. If the aircraft idle time is too long or works too busy will raise the airline operating costs and maintenance

costs. So under the premise considering the flight to meet the flight cover, maintenance requirements, and a balanced fleet, how to balance flying time of each aircraft is particularly important.

### B. Fleet assignment model

Fleet assignment model should abide by the following main constraints: (1) cover constraints so that each flight is assigned to exactly one aircraft; (2) balance constraints; and (3) aircraft availability constraints whereby the number of available aircraft of each type bounds their usage. According to the operational characteristics of airlines, the model aiming at aircraft balance can be established as following:

Set:

$F$  = flight set

$R$  = set of feasible flight loop

$M$  = maintenance base set

Subscript variable:

$j$  = subscript variable of feasible flight loop

$i$  = subscript variable of the flight

$$a_{i,j} = \begin{cases} 1, & i \in T_j \\ 0, & \text{else} \end{cases}$$

Parameters:.

$N$  = the number of fleet aircraft

$T_j$  = the flight time of the feasible flight loop  $j$

$Q$  = the total flight time of all aircrafts one day

Decision variables:

$$x_j = \begin{cases} 1, & T_j \text{ is chosen} \\ 0, & \text{else} \end{cases}$$

Objective function:

$$y = \min \sum_{j \in R} c_j x_j \quad (1)$$

Constraints:

$$\text{Cover: } \sum_{j \in R} a_{i,j} x_j = 1 \quad \forall (i \in F) \quad (2)$$

$$\text{Availability: } \sum_{j \in R} x_j \leq N \quad (3)$$

$$x_j \text{ binary} \quad (4)$$

The objective function,  $\min \sum_{j \in R} c_j x_j$ , means fleet balance

application.  $c_j = |T_j - Q/N|$  means using flight time of the flight loop minus the average flight time per aircraft at the same day, the absolute value of difference indicates that balance whether or not. Constraints (2) and (3) are the cover and aircraft availability constraints. Formula (4) is 0-1 integer requirements.

### C. Preprocessing data and steps

To reduce the algorithm's search space, we should find out the possible flight loop set  $R$  by preprocessing the flight data in advance in order to identify those flights meeting the

requirements of turn-time, total flight time and maintenance. We also should record flight time of each possible flight loop as  $T_j$  in the searching process. If there is no feasible solution, according to "the smallest lower bound first", the node is chosen from the record table to solve this problem in the next branch. In the branch, select the decision variables with max coefficient in the objective function.

The steps are described as following: Set (P) as the fleet assignment problem.

**Step 1** Solving the problem ( $\tilde{P}$ ) after giving up the 0-1 integer requirement. Put the optimal solution into  $\tilde{x}$  and minimum value into  $\tilde{y}_0$ .

If ( $\tilde{x}$  meet the 0-1 integer requirement)

Then end

Else  $\tilde{y}_0$  = lower bound of the (P)'s objective function value

**Step 2** Store (P) in  $\pi$ ,  $x^* = \tilde{x}$ ,  $y_0^* = \tilde{y}_0$

**Step 3** If ( $\pi = \phi$ )

Then  $x^*$  is the optimal solution of (P)

Else Continue

**Step 4** Take a sub-problem (Q) with the smallest lower bound value from  $\pi$ . Solve the relaxation problem (Q') after giving up the 0-1 integer requirement.

If ((Q') has no allowed solution or the minimum value)

Then go to step 3

Else If (the optimal solution of (Q') is also the admissible solutions of (Q))

Then go to step 6

Else Continue

**Step 5** In the branch, select the decision variables with max coefficient in the objective function. Branch is split into two sub-problems as (Q1) and (Q2), depending on 0 or 1, and give them a new lower bound as  $\tilde{y}_0$ . (Q1) and (Q2) are recorded in  $\pi$ . Go to step 4

**Step 6** If ( $\tilde{y}_0 \leq y_0^*$ )

Then  $\tilde{x} \rightarrow x^*$ ,  $\tilde{y}_0 \rightarrow y_0^*$ , Go to Step 3

Else go to step 3

#### IV. EXAMPLE

##### A. Parameter setting

Eastern Airlines need to fly 68 flights weekly by A320 aircrafts, thus constituted a flight set F ( $i = 1, \dots, 68$ ).  $M = \{\text{Wuxi, Nanjing, Pudong Airport}\}$ .  $N = 12$ . Assume the average total flight time of every plane per day is less than or equal to 14 hours, Turn-time is more than or equal to 40 minutes.  $Q$  = the total flight time of this day. 233 possible flight loops ( $j = 1, \dots, 233$ ) being searched out in advance by using depth-first search algorithm from the set of feasible flight loop R, part of them are shown in Figure 1.

$X = [x_1, x_2, \dots, x_{233}]$ .  $T_j$  = flight time of each possible flight loop recorded in the searching process. Set  $c_j = |T_j - Q/N|$  in the objective function. In addition to the availability constraint, there are total 69 constraints.

航班环号	由航班序号组成的航班环					
1	1	13	25	38		
2	1	13	25	38	44	57
3	1	13	25	38	44	61
4	1	13	25	38	46	54
5	1	13	25	38	46	66
6	1	13	25	38	47	58
7	1	13	25	38	51	61
8	1	13	25	38	59	67
9	1	13	25	38	63	68
10	1	13	62	66		
11	1	38				
12	1	38	44	57		
13	1	38	44	61		
14	1	38	46	54	62	66
15	1	38	46	66		
16	1	38	47	58		
17	1	38	51	61		
18	1	38	59	67		
19	1	38	63	68		
20	2	16				
21	2	16	28	40		
22	2	16	28	40	51	61
23	2	16	28	40	59	67
24	2	16	28	40	63	68
25	2	16	28	57		
26	2	16	28	61		
27	2	16	29	41	52	60
28	2	16	29	41	55	64
29	2	16	29	41	56	65
30	2	16	29	60		
31	2	16	31	45	55	64

Figure 1. Part of the flight loops.

##### B. Results

This experiment was completed in the environment including the computer with 2.60GHz CPU, 2 GB Memory, 500G Hard Disk, and MATLAB 7.0 in it. The data pre-processing searching time for possible flights set R is 0.133 second. We iterate 58 times to solve the aircraft assignment problem with Branch-and-bound method, which costs 0.097 second. The total time is 0.23 second. Figure 2 shows the matching results between 12 aircraft and 68 flights serial number.

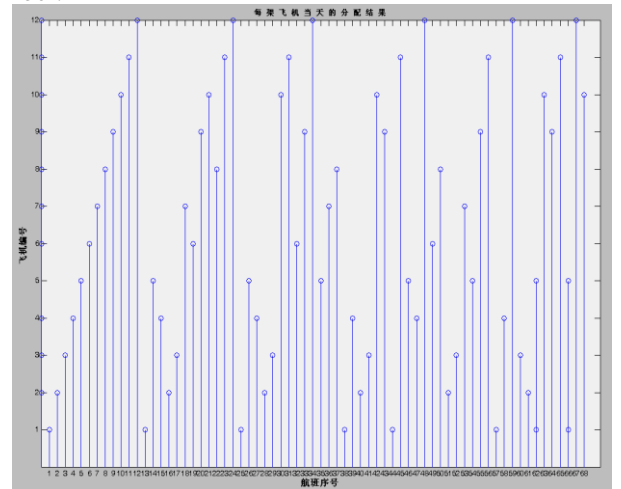


Figure 2. Fleet assignment results.

Table 1 describes the time-consuming situation of flights scheduling carried out in 7 days per week. Optimization runs more than 10 times every day. The time-consuming for searching feasible flight loops per day is average about 0.152 second. Fleet assignment problem solving takes 0.11 seconds on average, the total average time-consuming is less than 0.3 seconds. If we can assign the everyday results by using week as unit shift, then the fleet assignment table in a month or even a quarter can be gotten.

TABLE I. TIME-CONSUMING IN A WEEK

week	Mon	Tue	Wed	Thu	Fri	Sat	Sun
Flights	68	68	70	70	70	66	72
Flight loops	233	272	280	227	277	237	269
Iterate times	58	37	40	44	43	47	43
Total time-consuming (s)	0.23	0.315	0.28	0.228	0.271	0.235	0.27
Average (s)	0.261						

The total time-consuming of the resolutions based on Branch-and-bound algorithm and genetic algorithm for solving the same problem is different. They were compared in table 2. Based on genetic algorithm, the average time-consuming is 4.803s. This is almost the same time of Hong Sun. In his "flight scheduling mathematical programming model", he gets a satisfactory time scheduling program close to 23 seconds in an example of 20 aircraft, 35 flights<sup>[5]</sup>. It can be seen that the time consuming of using a Branch-and-bound algorithm with a control strategy is faster than the genetic algorithm for more than an order of magnitude.

## REFERENCES

- [1] Ma zhongfan, "Mathematical foundations of linear integer programming", China: Science Press, 1998.
- [2] Zhang Yun, Wang Jinbiao, Wang Yuankun, "Application of Ant Algorithm in Aircraft-arrangement Problem in Civil Aviation", Computer Engineering, vol:13, PP:7-9, 2005..
- [3] Massoud Bazargan, "Airline operations and scheduling"[M]. USA: Ashgate Publishing Limited, 2006..
- [4] Sun Hong, Du Wen, "Airliner aircraft dispatching problem: Heuristic algorithm based on stage-assignment "[J]. Journal of Systems Engineering, vol:18(2), PP:168~172, 2003.
- [5] Sun Hong, Du Wen, "Mathematical programming model of aircraft assignment", Journal of Traffic and Transportation Engineering, vol:4(3), PP:117~120, 2004.
- [6] WANG Wei, WANG Jinbiao, "A multipartite graph model for aircraft arrangement of CAAC", Computer and Communications, vol:26(4), pp:112-115, 2008.

TABLE II. COMPARISON OF TWO ALGORITHMS TIME-CONSUMING

Week	Flights	Flight loops	B&B (seconds)		GA (seconds)	
			time-consuming	Average	time-consuming	Average
Mon	68	233	0.230	0.261	4.189	4.803
Tue	68	272	0.315		5.505	
Wed	70	280	0.280		4.802	
Thu	70	227	0.228		3.734	
Fri	70	277	0.272		5.832	
Sat	66	237	0.232		3.823	
Sun	72	269	0.273		5.733	

## V. CONCLUSION

In this paper, the problem of aircraft assignment is solved efficiently with the Branch-and-bound method. In order to improve the efficiency of branch and bound method, three search strategies are given. Firstly, if there is no feasible solution in the event, the smallest lower bound of the branch points is always chosen. Secondly, there are several candidates to the branch node, select the integer variable with the largest coefficient in objective function as next branch. Thirdly, the preprocessing of data is necessary in the process of solving specific fleet assignment problem, so that the search with branch and bound algorithm is developed only in the possible field, so the computation is far less than that in exhaustive method. To compare with the genetic algorithm, the algorithm strategy is more feasible and effective in the concrete example.

To further improve the efficiency, we can combine Branch-and-bound algorithm with the modern heuristic algorithms for special flight scheduling and real-time dispatch of flight, which will be further discussed and researched in the fleet assignment arena.

## ACKNOWLEDGMENT

This paper is supported by National Natural Science Foundation Project (60672167).

- [7] Wang Zhuzhu, Cheng Jiaying, "An algorithm on solving combinatorial optimization problems-branch-and-bound method"[J], Journal of Anhui University, vol:28(1), PP:10214, 2004.
- [8] Hanif D. Sherali, Ebru K. Bish, Xiaomei Zhu, "Airline fleet assignment concepts, models, and algorithms", European Journal of Operational Research, vol:172, PP:1-30, 2006.
- [9] Mercier, A., Soumis, F., "An integrated aircraft routing, crew scheduling and flight retiming model", Computers and Operations Research, vol:34, PP:2251-2265, 2007.
- [10] Mohamed Haouari, Najla Aissaoui, Farah Zeghal Mansour, "Network flow-based approaches for integrated aircraft fleet and routing", European Journal of Operational Research, Vol: 193, PP:591-599, 2009.
- [11] Arif Imran, Said Salhi, Niaz A. Wassen, "A variable neighborhood-based heuristic for the heterogeneous fleet vehicle routing problem", European Journal of Operational Research, Vol:197, PP:509-518, 2009.
- [12] Andrew J. Woodcock, John M. Wilson, "A hybrid tabu search/branch & bound approach to solving the generalized assignment

problem”, European Journal of Operational Research, Vol: 207, PP:566-578, 2010.

[13] Jonathan Dumas, Fati Aithnard, François Soumis ,”Improving the objective function of the fleet assignment problem”, Transportation Research Part B: Methodological, Vol: 43,PP:466-475, 2009.