Mathematical & Computational Applications, Vol.1, No. 1, pp. 9-14, 1996.

PROCEDURES FOR AIRCRAFT-GATE ASSIGNMENT

Ahmet Bolat[1] and Khaled As-Saifan[2]

[1]Department of Mechanical Engineering, King Saud University, Riyadh, Saudi Arabia
[2]Saudi Arabian Airlines, Riyadh, Saudi Arabia

ABSTRACT - The task of assigning arriving flights to the available gates at an airport can have a major impact on the efficiency with which the flight schedules are maintained and on the level of passenger satisfaction. In this paper, optimum and heuristic procedures are developed to assign flights in such a way that the idle periods, during which gates are not utilized, are distributed as uniformly as possible. Emprical study indicates that up to 60% improvements can be achieved over the current practice.

1. INTRODUCTION

The assignment of arriving aircraft to available gates in the apron area is a key activity in airline station operations. A proper assignment increases the ability to handle the large volumes of aircraft, passengers and baggages in a relatively short time. On the other hand, delays may disrupt the premade assignments and compound the difficulty of maintaining flight schedules efficiently. The importance of the gate assignment problem has attracted many researchers over the past 20 years. Initially, mathematical models are developed to minimize the average walking distance of the passengers inside the terminal building ([1]-[3]). Since these models fail to address the minor changes in the flight schedules, dynamic simulation models [4] and expert systems [5]-[8] are proposed to handle the uncertain information, and to consider additional performance criteria. Beside the difficulty in extracting the relevant knowledge, maintaining large complex systems, equipped with many hundreds or thousands of rules addressing most of the significant factors, becomes a critical factor.

In this work a mathematical model is developed to distribute the idle time periods, during which gates are not occupied by aircraft, as evenly as possible so that the assignments are flexible enough to cope with the minor changes. Several criteria, (including the passenger satisfaction level) can be incorporated through a gate-aircraft restriction matrix utilized in the model. Optimum and heuristic procedures are proposed to produce a master gate-assignments which can be revised whenever a major change disrupts the current assignments. A computational study is conducted over the randomly generated data and the real data obtained from the Saudi Arabian Airline (SA) over the domestic flights in the King Khaled International Airport (KKIA) in Riyadh.

2. PROBLEM FORMULATION

Consider N flights to be assigned to M gates, each one is available from HB_j till T unit time. The expected arrival and departure time of flight i (A_i and D_i) are known in advance and expressed in terms of unit time. The problem is to find an assignment Γ, where Γ_i represent the gate assigned for flight i, so that the range of the slack time (idle

Γ, respectively, then the corresponding range is $V^+(\Gamma)$ - $V^-(\Gamma)$. Let $V_i(\Gamma)$ be the slack time of flight i assigned in Γ, and B_{Γ_i} and E_{Γ_i} be the beginning and ending time of utilization period of gate Γ_i by flight i. By definition $V_i(\Gamma) = A_i - B_{\Gamma_i}$ and $E_{\Gamma_i} = D_i$. There is a relationship between B_{Γ_i} and the last flight used gate Γ_i before i. Let z* be this last flight, i.e., z* = max(z) so that $\Gamma_z = \Gamma_i$ and $1 \leq z < i$. In case i is the first assignment, then B_{Γ_i} is set to HB_{Γ_i}. Similar precaution is taken when i is the last flight in gate j; slack occurs until the end of the period ($V_{N+j} = T - E_j$).

Obviously, only feasible assignments are to be considered. The ground times of two flights, consequtively assigned to the same gate, should not overlap ($E_{\Gamma_{z*}} \leq A_i$). Secondly, only assignments satisfying managerial considerations and passenger expectations should be made, i.e., gate Γ_i can be assigned to flight i if $P_{i,\Gamma_i} = 1$. ($P_{i,j}$ is the gate restriction matrix equal to 0 if flight i is not allowed in gate j.) The following is the mathematical presentation of the problem described above:

P1	Find	Γ	
	so that	$V^+(\Gamma) - V^-(\Gamma)$ is minimized	
	where	$V^+(\Gamma) = \max\{V_i(\Gamma)\}$	$i = 1, \dots, N+M$
		$V^-(\Gamma) = \min\{V_i(\Gamma)\}$	$i = 1, \dots, N+M$
		$V_i(\Gamma) = A_i - B_{\Gamma_i}$	$i = 1, \dots, N$
		$B_{\Gamma_i} = \max\left(E_{z*}, HB_{\Gamma_i}\right)$	$i = 1, \dots, N$
		$z^* = \max(z)$ where $\Gamma_z = \Gamma_i$ and	$1 \leq z < i$
		$E_{\Gamma_i} = D_i$	$i = 1, \dots, N$
		$P_{i,\Gamma_i} = 1$	$i = 1, \dots, N$
		$B_{\Gamma_i} \leq A_i$	$i = 1, \dots, N$
	and	$V_{N+j} = T - E_j$	$j = 1, \dots, M$

It is clear that problem size grows drastically with the number of flights and gates. Below we propose an optimum algorithm which will be the basis for the heuristic to be utilized in solving practical instances of the problem.

3. PROCEDURES

Supressing the mathematical details for clarity of exposition, we first describe the Optimum Branch and Trim (OBT) procedure. The ith layer of the tree corresponds to the flight number under consideration and partial solution f at layer i-1 (Γ^f) is the set of gate assignments for flights 1, 2, ..., i-1. All (father) solutions in layer i-1 are considered to generate new (son) solutions in layer i. Note that at most M sons can be generated for each father depending on the gate restrictions and ground times of flights. If the resulting slack time of flight i at the assigned gate does not make the range of the solution worse than that of the incumbent, then a stronger dominance criteria can be applied.

We let $LBV^+(\hat{\Gamma}^s)$ be a lower bound on the maximum slack time in the solutions generated from son solution $\hat{\Gamma}^s$. The basic idea behind this bound is as follows: The earliest available gate is detemined. Say it is gate p and $E_p \le E_j$, j = 1,..., M. It is clear that some of the sons of $\hat{\Gamma}^s$ may assign gate p to the next flight i+1, some of them may assign a later flight k, and still others may not assign any flight at all. In the first case the next slack time is $A_{i+1} - E_p$, in the second case it is $A_k - E_p$ which is not less than $A_{i+1} - E_p$, and in the final case it is $T - E_p$. Hence, $LBV^+(\hat{\Gamma}^s) = \max\{A_{i+1} - E_p, V^+(\hat{\Gamma}^s)\}$. Solution $\hat{\Gamma}^s$ is kept as the sth new son solution if it passes this criteria too. Once all promising solutions are determined at the last layer, the slacks between the last flights and the end of horizon are evaluated. The solution with the lowest range is the optimal one. We now outline the procedure.

OBT Algorithm

Step 0. Find an incumbent solution Γ^* with the objective value $V^* = V^+(\Gamma^*) - V^-(\Gamma^*)$. Arrange flights in ascending order of their arrival times. Set $\Gamma^1 = \phi$ and F = 1.

Step 1. Consider each flight i in turn (i = 1,...,N). Set f = 1 and s = 1.

Step 2. Consider each gate j in turn (j = 1,..., M) for the current flight i. Check for feasibility: Assign flight i to gate j ($\hat{\Gamma}^s = \Gamma^f \cup j$) if $P_{ij} = 1$ and $A_i \ge B_{\Gamma_i^f}$.

Check for pruning. Find the resultant slack time $V_i(\hat{\Gamma}^s)$ and evaluate its effect on the maximum and minimum slack times, i.e., find $V^+(\hat{\Gamma}^s)$ and $V^-(\hat{\Gamma}^s)$. If the new range and the expected one are better than that of the incumbent then keep the new son, i.e., set s = s + 1, if $V^+(\hat{\Gamma}^s) - V^-(\hat{\Gamma}^s) < V^*$ and $LBV^+(\hat{\Gamma}^s) - V^-(\hat{\Gamma}^s) < V^*$.

Step 4 Arrange the new s son solutions in ascending order of their range, i.e., $V^+(\hat{\Gamma}^k) - V^-(\hat{\Gamma}^k) \le V^+(\hat{\Gamma}^{k+1}) - V^-(\hat{\Gamma}^k)$ where k = 1,2,..., s-1. Go to the next layer with $\hat{\Gamma}^1, \hat{\Gamma}^2, \ldots, \hat{\Gamma}^s$ as father solutions. Set F = s.

Step 5 Evaluate the end effect of horizon for each solution Γ^f. The one with the lowest range and still better than the incumbent is the optimal one.

OBT constructs N layers and a father solution may generate up to M new son solutions. In case neither of the criteria is able to prune any partial solution, the maximum number of complete solution may become M^N. (This exponential behaviour points out the importance of the trimming schemes and starting with a good incumbent solution.) Hence, a heuristic implementation of OBT, which discards some of the father solutions passed the both criteria, is adopted and called Truncated Branch Heuristic (TBH). Let the beam size be the maximum number of father nodes to be considered at the next layer. In the early layers the number of fathers may be less than the beam size, allowing the propagation of all possible branches. Truncation will start later when s ≥ Beam Size. The maximum number of solution at the final layer is bounded by Beam Size ×M. The outline of TBH is omitted because its differences from OBT are minimal.

(Pruning scheme can be kept when an ad hoc assignment is available initially.) It is obvious that truncation may lead the TBH to miss the optimal solution. An inferior solution may become better in later steps or vice versa. For that reason, there is no quarantee that increasing beam size will yield better solutions. Until the algorithm reaches the steady state, smaller beam sizes may perform better than the larger ones.

4 COMPUTATIONAL EXPERIMENTS

Both procedures are coded in FORTRAN 77 and tested on an AMDAML 5880 computer. BTO performs fairly well over the randomly generated data with constant slack times. Slack time is treated as a parameter to determine the arival times of next flights. Ground times are generated uniformly between 7 and 48 unit time. (A unit time is equal to 5 minutes and the planning period is 288 unit time.) Schedules for the first 10 days are generated with one unit slack time indicating heaviest utilization. Then the next 10 days of schedules are generated with 2 unit slack time and so on. Finally schedules for days 111 - 120 result in 12 unit slack time indicating the lightest utilization. It is observed that BTO can assign 65 flights over 7 gates (with 3 unit slack time) within 96.74 seconds (at most 14880 nodes evaluation at one level). In the worst cases, it takes 135.48 seconds for assigning 55 flights over 7 gates (with 9 unit slack time). It evaluates maximum 16728 nodes while assigning 56 flights over 6 gates with 1 unit slack time. (To avoid memory problems 20,000 father solutions are allowed at most in the tree.) The most difficult schedules are observed when the slack times are 8 and 9 units; BTO can handle only 36 flights over 5 gates within 1553 and 882 seconds, respectively.

All of these results are achieved with the best incumbent solutions. By the construction of the data, the ranges of all optimal solutions are zero. The effect of the incumbent solution is studied over various starting points each defined by its range and called "best result" available. In the worst case, the best result has a range of 6 unit time, it is found that flights over 4 gates can be considered with an average of 34.4 flights. Figure 1 presents how the value of incumbent solution effects the average computation effort for assigning the flights optimally. It appears that this effect becomes significant when the incumbent solution results in 5 or 6 unit time slack range. The analysis of two-factor factorial design reveals that both the incumbent solution and the utilization level (optimal slack time) effect significantly the performance of BTO for 5% level. Furthermore, there is significant interaction between these factor, evident from the lack of parallelism of the lines in the Figure.

To study the effect of data type we have generated 10 days of flight schedules with varying slack times in each schedule. The length of the horizon is 200 unit time, and the ground times are uniformly distributed between 2 and 53 unit time. Slack times are also generated from a uniform distribution between 9 and 30 unit time. Now the maximum number of flights that can be handled reduces to 20 over 4 gates. Furthermore, the impact of the starting point is more evident: daily schedules 1 and 3 can be handled if the incumbent solutions are very close to the optimal solutions (6.25% deviations). The "easiest" schedule is set 5 for which BTO can find an optimal assignment even the incumbent solution is 66.66% far from the optimality. Overall the average required closeness to the optimality is 28.36% with a standard deviation of 19.64%. On the other hand, computational effort shows a wider spread: maximum 134.69 and minimum 0.65 seconds are required with the best starting points.

The TBH is tested over second group of data. (Note that it will yield the optimum solutions over the first group of data even the beam size is set to1). Table 1 presents the results of TBH solutions with various beam sizes. The performance of TBH can be categorized as follows: TBH reaches the steady state very quickly and finds an optimal solution (over sets 2, 4 and 6 with the beam sizes 1 or 2). Secondly, the steady state is observed after some times and initial optimal solutions can be missed for larger beam sizes (sets 8 and 10). Note that as large as 386 beam size is required to find the optimal solution second time. Thirdly, steady non decreasing improvements can be observed as in sets 5, 7 and 9. Finally, very slow improvements with large beam sizes (set 3) or no improvements at all (set 1) can be realized. In these cases, however, comparable low deviations result from the optimality even with smaller beam sizes. Over 10 sets, average deviation is approximately 3% when the beam size is around 400.

Finally, the TBH is tested over the real data sets collected from KKIA, Riyadh: Daily domestic flights of Saudi Arabian Airline are considered from 19 October 1993 to 25 October 1993. Contrary to the randomly generated flights, some of the aircrafts, such as AB 300, L1011, and B747, are not allowed to gates 1 and 8. (Since the flights can not be decomposed based on the gates, it is very difficult to trim the real data to fit the sizes that can be handled by the OBT algorithm.) We now turn to our primary objectives: The number of flights that are served in the remote area and that are towed momentarily from their assigned gates. The results are presented in Table 2. The current practice is a basic application of the FCFS policy. TBH assigns the flights within a seconds with 100 beam sizes. It seems that TBH solutions (generated based on the expected flight schedules and evaluated by using the actual flight schedules) require 7.67 - 60.00% less remote servings and towings combined with an average of 40.84%. The improvement over the number of remote servings is between 50 - 100%, with an average of 74.41%.

REFERENCES

1. J.P. Braksma. "Reducing Walking Distances at Existing Airports", *Airport Forum*, No. 4 (August) 135 - 142 (1977).
2. R.S. Mangoubi. "Optimizing Gate Assignments at Airport Terminals," *Transportation Science*, Vol. 19, No. 2 (May)173 - 188 (1985).
3. R.A. Bihr. "A Conceptual Solution to the Aircraft Gate Assignment Problem Using 0, 1 Linear Programming", *Computers Ind. Engng*, Vol. 19, Nos. i-4, 280-284, (1990).
4. S.G. Hamzawi, "Microcomputer Simulation Aids Gate Assignment", *Airport Forum* No. 5, 66-70 (1986).
5. R.P. Brazile and K.M. Swigger, "GATES, An Airline Gate Assignment and Tracking Expert System", *IEEE Expert* (Summer) 33 - 39 (1988).
6. G.D. Gosling. "Design of an Expert System for Aircraft Gate Assignment," *Transportation Research*, Vol. 24 A, No. 1, 59-69 (1990).
7. K. Srihari and R. Muthukrishnan. "An Expert System Methodology for Aircraft-Gate Assignment", *Computers Ind. Engng*. Vol. 21, No. 1-4, 101-105 (1991).
8. R.P. Brazil and K.M. Swigger, "Generalized Heuristics for the Gate Assignment Problem," *Control and Computers*, Vol. 19, No. 1, 27-32 (1991).

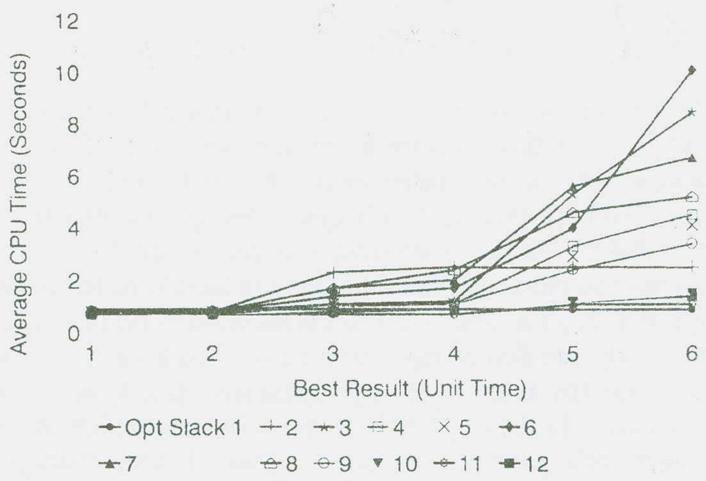

Figure 1: The computational effort of BTO over different slack times and incumbent solutions.

Allowed Nodes	DATA SET 1	2	3	4	5	6	7	8	9	10	% deviation Average from optimality
1	20	16	21	14*	169	16*	119	21	117	54	222.34
2	20	15*	21	14*	169	16*	119	19*	117	18*	200.62
5	20	15*	17	14*	169	16*	119	19*	117	18*	198.12
7	20	15*	17	14*	169	16*	119	75	117	171	312.60
10	20	15*	17	14*	130	16*	18*	75	117	171	234.82
29	20	15*	17	14*	130	16*	18*	75	91	171	220.38
30	20	15*	17	14*	130	16*	18*	75	18*	171	179.82
97	20	15*	17	14*	130	16*	18*	19*	18*	171	150.35
142	20	15*	17	14*	18*	16*	18*	19*	18*	171	88.12
385	20	15*	17	14*	18*	16*	18*	19*	18*	54	23.12
386	20	15*	17	14*	18*	16*	18*	19*	18*	18*	3.13
2801	20	15*	16*	14*	18*	16*	18*	19*	18*	18*	2.50
6689	20	15*	16*	14*	18*	16*	18*	19*	18*	18*	2.50

Table 1: Results of the TBH solutions over random data (* indicates optimality)

Date	No. of Flights	TBH Solutions Remote	TBH Solutions Towed	Practice of KKIA Remote	Practice of KKIA Towed	% of Improvement Remote	% of Improvement Towed	% of Improvement Total
9 October 1933	72	3	3	6	6	50.00	50.00	50.00
10 October 1993	77	4	8	8	5	50.00	-60.00	7.69
11 October 1993	81	1	8	11	6	90.90	-33.33	47.06
12 October 1993	72	1	3	5	3	80.00	0.00	50.00
13 October 1993	82	1	7	6	3	83.33	-133.33	11.11
14 October 1993	64	1	1	3	2	66.67	50.00	60.00
15 October 1993	79	0	4	7	3	100.00	-33.33	60.00

Table 2: Results of the TBH and the current practice in KKI Airport, Riyadh over real data.