Optimizing Gate Assignment at Airport Terminals

Article *in* Transportation Science · May 1985

DOI: 10.1287/trsc.19.2.173

Rami S. Mangoubi
C.S. Draper Laboratory

Optimizing Gate Assignments At Airport Terminals.

Optimizing Gate Assignments at Airport Terminals

R. S. MANGOUBI
*Charles Stark Draper Laboratory, Inc., Cambridge, Massachusetts**

DENNIS F. X. MATHAISEL
Massachusetts Institute of Technology, Cambridge, Massachusetts
Babson College, Babson Park, Massachusetts

The airport flight-to-gate assignment problem is solved using two methods: (1) a linear programming relaxation of an integer program formulation and (2) a heuristic. The objective is to minimize passenger walking distances within the airport terminal area through a judicious gate assignment policy. An actual flight schedule for an average day at Toronto International Airport is used to compare existing walking distances, obtained from the original assignment, with results from the two methods. The results indicated that the original assignment had a 32% higher average per passenger walking distance than the minimum possible distance given by the LP solution. The heuristic's performance was near optimal; it gave an average walking distance which was only 3.9% greater than the minimum. Computation times for the heuristic are 3.4 CPU seconds per run, while the linear program consumes 386 seconds per run on an IBM 370/168. In addition, if the heuristic is solved first and its solution is used as an initial feasible basis for the LP relaxation of the IP, the total CPU used to obtain optimality is reduced to 42 seconds.

In planning for an effective utilization of aircraft gates at an airport terminal, one consideration should be the distance a passenger is required to walk inside the terminal to reach his departure gate, the baggage claim area, or his connecting flight. These distances can be reduced by improvements to the method in which scheduled flights are assigned to the

*This work was performed while the first author was at Dynatrend, Inc., Woburn, Massachusetts, and a graduate student at the Flight Transportation Laboratory, MIT, Cambridge, Massachusetts.

Transportation Science
Vol. 19, No. 2, May 1985

0041-1655/85/1902-0173 $01.25

airport terminal gates. In 1977 BRAAKSMA[1] demonstrated that this procedure can have a profound impact on passenger walking distances, without changing the layout of the terminal area. As an example, the mean walking distance per passenger at Terminal No. 2 of Toronto International Airport in Canada was reduced from 923 feet in 1973 to 800 feet in 1975. This improvement was a direct result of a change in gate assignment policy by Air Canada, the terminal's sole user. The decision represented a savings of over 100 feet per passenger.

A question arises. What is the best gate assignment policy? Specifically, what flight-to-gate assignment yields the minimum walking distances? The challenge is to develop an approach to the flight-to-gate assignment problem which yields the minimum total, or average per passenger, walking distance. The flight timetable should be assumed fixed, as should be the airport terminal layout. No changes would be required in the airline schedule or the airport terminal.

Two approaches to the assignment problem are presented. One makes use of a linear programming relaxation to an integer programming formulation, and the other employs a heuristic (see also MANGOUBI[2]). The heuristic approach is interesting because it provides a good solution to the congestion and walking distance problems, and it can be used as an initial solution to the LP relaxation of the integer program. Another integer programming formulation to the gate assignment problem, using a branch-and-bound technique, was presented by BABIC et al.[3] Finally, flight schedule data from the earlier Toronto study were used to compare the results from these two approaches with the Braaksma results. The paper also compares computational times of the two solution methods.

1. THE INTEGER PROGRAM FORMULATION

A BINARY variable x_{ij} is assigned for each possible flight-to-gate assignment:

$$x_{ij} = \begin{cases} 1 & \text{if flight } i \text{ is assigned to gate } j, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Furthermore, passengers are divided into three categories: arrival passengers, departure passengers, and transfer passengers. The terms p_i^a, p_i^d, and p_i^t will denote, respectively, the estimated number of arriving, departing, and transferring passengers using flight i. The walking distance is denoted by d_j^a for an arriving passenger using gate j, d_j^d for a departing passenger, and d_j^t for a transfer passenger. Let Z denote the total walking distance for all passengers, and let M be the number of flights in the schedule and N the number of gates. The assignment problem is expressed as:

$$\text{Min } Z = \textstyle\sum_{i=1}^{M} \sum_{j=1}^{N} (p_i^a d_j^a + p_i^d d_j^d + p_i^t d_j^t) x_{ij}. \tag{2}$$

The walking distance, d_j^a, for an arriving passenger using gate j is the measured distance between the airport gate and the baggage claim area. For a departing passenger, the term d_j^d is the distance between the check-in counter and the gate. The distance used for a transfer passenger is based on the assumption that a passenger arriving at gate j would be equally likely to board his next flight at any gate, including gate j. Transfer passenger walking distances, therefore, are determined from a uniform probability distribution of all intergate walking distances. If w_{jk} is the distance between gate j and gate k, then the expected walking distance for a transfer passenger arriving at gate j is:

$$d_j^t = 1/N \textstyle\sum_{k=1}^{N} w_{jk} \qquad \forall j = 1, \ldots, N. \tag{3}$$

It must be pointed out that this uniform distribution assumption suffers from two inherent weaknesses. First, it will be shown from the results (Section 5) that both the heuristic and the LP solutions, by the nature of their objectives, will assign most flights to a few attractive gates. The less attractive gates will remain empty or nearly empty. The inference of this result is that the probability distribution of transfer passengers among the gates is not uniform. Second, the use of Equation 3 is likely to overestimate the walking distance for gate pairs that are heavily utilized. The reason is that the attractive gates tend to be clustered together, so that in addition to having a higher transfer probability, the transfer passengers' walking distance between these gates is shorter. As a result, the use of a uniform distribution and Equation 3 is likely to cause a bias in the objective function toward a greater weight placed on serving transfer passengers than perhaps should be placed on these individuals. In the absence of any a priori knowledge regarding transfer patterns, however, the use of this assumption to derive d_j^t is justifiable.

Two classes of constraints exist for the gate assignment problem at Toronto: those which are physical and inherent to the problem, and those which depend on the airport management or the airline using the terminal. The first class of constraints is necessary for the flight-to-gate assignment to meet the following two conditions.

1. Every flight must be assigned to one and only one gate:

$$\textstyle\sum_{j=1}^{N} x_{ij} = 1 \qquad \forall i = 1, \ldots, M. \tag{4}$$

The number of constraints of this type is equal to the number of flights, M.

2. No two aircraft may be assigned to the same gate concurrently:

$$\sum_{h \in L(i)} x_{hj} + x_{ij} \leq 1 \qquad \begin{matrix} \forall\, i = 1, \cdots, M \\ \forall\, j = 1, \cdots, N. \end{matrix} \tag{5}$$

There are $M \times N$ constraints of this type for M flights and N gates. $L(i)$ is the set of all flights h which landed before flight i and are still on the ground at the time flight i arrives. If the flights are indexed in order of their arrival time, then each set $L(i)$, here termed a "conflict set," can be defined as follows:

$$L(i) = \{h \mid t_h^d \geq t_i^a,\ h = 1, \cdots, i - 1\} \tag{6}$$

where t_h^d is the departure time of flight h and t_i^a is the arrival time of flight i. Set $L(i)$ is termed a "conflict set" because if any flight conflicts in time with flight i, it cannot be assigned to the same gate j. These constraints are inequalities in order to express the fact that some gates do not necessarily have to be used at all times.

The conflict set $L(i)$ can be defined in a recursive manner, which simplifies the formulation: of all flights preceding flight i, one needs to consider only those flights belonging to $L(i - 1)$ as well as flight $i - 1$ itself. This procedure works since the flights are indexed in order of their arrival time. The recursive definition of $L(i)$ is written:

$$L(i) = \{h \mid t_h^d \geq t_i^a,\quad h \in \{L(i - 1) \cup (i - 1)\}\}. \tag{7}$$

The total number of constraints in the integer program generated by the above formulation is $M \times (N + 1)$ for M flights and N gates. For example, there were 138 flights and 20 gates in the sample day's schedule at Toronto. The total number of constraints would be, accordingly, 2898. However, many of these constraints are redundant.

2. IDENTIFICATION OF REDUNDANT CONSTRAINTS

CONSIDER the following example. Suppose in a day's flight schedule there are only three aircraft on the ground at the time a pth aircraft is scheduled for arrival. Let these aircraft be, without loss of generality, the p-3rd, p-2nd, and p-1st arriving aircraft. Then the conflict set is $L(p) = \{p - 3, p - 2, p - 1\}$; and the conflict constraint for the pth flight and any gate j is:

$$\sum_{h \in L(p)} x_{hj} + x_{pj} = x_{p-3,j} + x_{p-2,j} + x_{p-1,j} + x_{pj} \leq 1. \tag{8}$$

Assume further that the $p + 1$st flight arrives, while neither p nor any of the three flights contained in $L(p)$ depart. For each gate:

$$L(p + 1) = \{p - 3, p - 2, p - 1, p\} = \{L(p) \cup p\}. \tag{9}$$

The conflict constraint for each gate j is:

$$\sum_{h \in L(p+1)} x_{hj} + x_{p+1,j} = x_{p-3,j} + x_{p-2,j} + x_{p-1,j} + x_{p,j} + x_{p+1,j} \leq 1. \tag{10}$$

Any solution satisfying (10) will automatically satisfy (8). As a result, the constraints generated by the pth flight are redundant and can be dropped. For an airport with 20 gates, the procedure can result in 20 fewer constraints.

Note that in the above case $L(p) \subset L(p + 1)$. Likewise, if k aircraft land consecutively with no departures in between, then the sets $L(p)$ through $L(p + k)$ are nested. That is,

$$L(p) \subset L(p + 1) \subset \cdots \subset L(p + k). \tag{11}$$

Relation (11) is useful since it helps in recognizing that the constraints generated by the flights $p, p + 1, \cdots, p + k - 1$ are redundant. To generalize: if two or more flights arrive with no departures in between, then only the constraints generated by the last one of these flights need to be considered.

In the case of the Toronto schedule, the original number of constraints was 2898. Of these, 138 constraints were of the first type (every flight assigned to only one gate), and 2760 were of the second type—the conflict constraints (no two aircraft are to be assigned to the same gate concurrently). The flight schedule showed, however, that there are 59 nonnested conflict sets, each of which generates 20 constraints—one for each gate at the terminal. As a result, the number of conflict constraints is reduced to $59 \times 20 = 1180$. The total number of constraints is then $1180 + 138 = 1318$, instead of 2898. The number of rows in the constraint matrix was reduced by 54%.

Finally, no integer programming algorithm was needed in this case, as the LP relaxation to the problem gave a 0, 1 optimal solution for the x_{ij}s. Further examination of the constraint matrix has led to the discovery of the possibility for a nonintegral basis in other cases, which means that the matrix is not totally unimodular, and the simplex algorithm will not in general offer an integral 0, 1 solution. If a noninteger solution does occur with any other case, some form of branch and bound approach would be needed.

3. ADDITIONAL CONSTRAINTS

IN ADDITION to the two types of constraints inherent to the assignment problem, other constraints which depend on the individual airport will usually exist. No such constraints were introduced into the current model, as they were not known to apply or exist for Toronto Terminal No. 2. However, a few possible constraints should be mentioned.

3.1. Subdivision of the Airport into Separate Airline Areas

Most U.S. airports are divided into several areas, where each area is reserved for the exclusive use of a particular carrier. If a total of S airlines are using the terminal, then the set I of all flights and the set J of all airport gates can each be subdivided into S subsets: $I = \{I_1, \cdots, I_S\}$ and $J = \{J_1, \cdots, J_S\}$. Such a subdivision would lead to S subterminals, and S smaller separate integer programs each similar in form to the formulation given in the previous sections. Proponents of shared airport terminal facilities might argue, however, that if walking distances are to be significantly reduced, the practice of the subdivision of the terminal areas should not be followed.

3.2. Excluding Some Aircraft Types from Some Gates

Some gates are often too small for wide-bodied aircraft. The constraint can be accommodated by setting the appropriate decision variables to zero. For example, let $x_{ij} = 0$ for all $i \in I$ and all $j \in J$, where I is the subset consisting of all flights with a wide-body aircraft, and J consists of all gates which are incompatible with such aircraft.

3.3. Flights Are to Be Assigned to Nearby Gates

If two flights, g and h, serve the same large number of transfers, it may be desirable to have them assigned near to each other. This assignment can be done by adding:

$$\sum_{z=1}^{N} \sum_{s=1}^{N} x_{gz} w_{zs} x_{hs} \le D_{gh}^{\max} \quad (12)$$

where w_{zs} is the intergate distance between gates z and s, and $D_{gh}^{\max}$ is the maximum allowable distance between flights g and h. However, this constraint is nonlinear and undesirable. One possible postoptimal method for dealing with such a constraint is presented in Section 5.

4. A HEURISTIC APPROACH

THE BURDEN of long walking distances on air travelers can also be eased by implementing simple, yet efficient, assignment procedures. For example, assign the best available gate to the first arriving flight. Many heuristic algorithms like this were tested,[4] and as a result one heuristic showed the best performance overall in reducing walking distances. The heuristic consists of assigning the aircraft/flight with the greatest number of on-board passengers to the gate with the corresponding shortest total walking distances. The algorithm proceeds as follows:

1. List the flights in order of decreasing passenger volume.
2. Consider the first flight in the list. Label it flight f.

3. Store all gates which are available at the time of flight f's arrival in set S.
4. Assign flight f to gate $s \in S$, where s is the gate with the minimum total passenger walking distance for flight f:

$$p_f^a d_s^a + p_f^d d_s^d + p_f^t d_s^t = \min_{j \in S} \{p_f^a d_j^a + p_f^d d_j^d + p_f^t d_j^t\}. \quad (13)$$

5. If all flights have a gate assignment, terminate. Otherwise, consider the next flight f and go to Step 3.

Although assigning the aircraft with the greatest number of passengers to the best gate is a logical approach to the problem, this algorithm may not provide the best assignment: the assignment corresponding to the minimum per passenger walking distance. A simple example will illustrate why the heuristic can be suboptimal. Consider the sample flight schedule shown in Table I. A B-747 aircraft in flight 102 has a gate turn time of 3 hours, during which three B-727s are scheduled to be on the ground. There are two gates, labeled A and B, and the calculated walking distances are given in Table II. The heuristic assigns the largest aircraft, the B-747, to gate A because A has the shortest walking distances. The three B-727s are assigned to B. However, the best solution, which is given in Table III, calls for the 747 to be assigned to gate B and the three 727s assigned to A.

5. RESULTS

THIS SECTION compares the results of the two solution methods: the heuristic, and the linear program. Both are applied to a representative

TABLE I
Sample Flight Schedule

Flight	Aircraft	Arrival Time	Departure Time	On Board Passengers		
				Arrivals	Transfers	Departures
101	B-727	10:00	10:40	100	40	100
102	B-747	10:10	13:10	200	50	150
103	B-727	10:45	11:30	80	40	120
104	B-727	12:00	13:20	80	40	120

TABLE II
Passenger Walking Distances

Gates	Walking Distance per Passenger (feet)		
	Arrivals	Transfers	Departures
A	400	700	500
B	500	700	650

TABLE III

Gate Assignment Results

Flight	Aircraft	Heuristic Algorithm: Gate	Heuristic Algorithm: Walking distance per passenger (feet)	Optimization: Gate	Optimization: Walking distance per passenger (feet)
101	B-727	B	596	A	492
102	B-747	A	475	B	581
103	B-727	B	608	A	500
104	B-727	B	608	A	500

	Average Walking Distance per Passenger (feet)
Heuristic algorithm:	558
Optimization:	527

day's schedule at Terminal No. 2 of Toronto International Airport, and the results of both solution techniques are also compared against the existing walking distances obtained from the original flight-to-gate assignment.

The solution to the LP relaxation of the problem in this case is integral, and no additional work is necessary to make the solution integer, even though the constraint matrix is not totally unimodular. Although computational experience has indicated that the solution to the LP is highly likely to be integer, nontrivial cases exist where the optimal basis is not integral, and in these cases a branch and bound approach would be necessary.

Figure 1 shows the cumulative distribution of the weighted average walking distances for all categories of passengers (arriving, departing, and transferring). The distribution displays the results of the flight-to-gate assignment policy for the two solution methods. The curve on the extreme left, which is the result from the linear program, gives the minimum average walking distance. The solid line next to it is the result from the heuristic algorithm. Also displayed, on the extreme right, is the walking distance curve resulting from the original flight-to-gate assignment (the assignment which was in place prior to this study). The table adjacent to the graph indicates that the LP offers a mean walking distance of 608 feet per passenger, while the original airport assignment distance of 608 feet per passenger, while the original airport assignment gives a mean of 803 feet per passenger. The difference is 195 feet, which means that the original assignment produced an average walking distance per passenger which is 32% higher than the minimum possible distance given by the LP solution. The heuristic, on the other hand, offers an assignment with a mean of 632 feet per passenger, which is only 3.9% larger than the optimal answer. The near optimal performance of the heuristic is illustrated in Figure 1 by the fact that the LP distribution and the heuristic distribution are very close to each other.

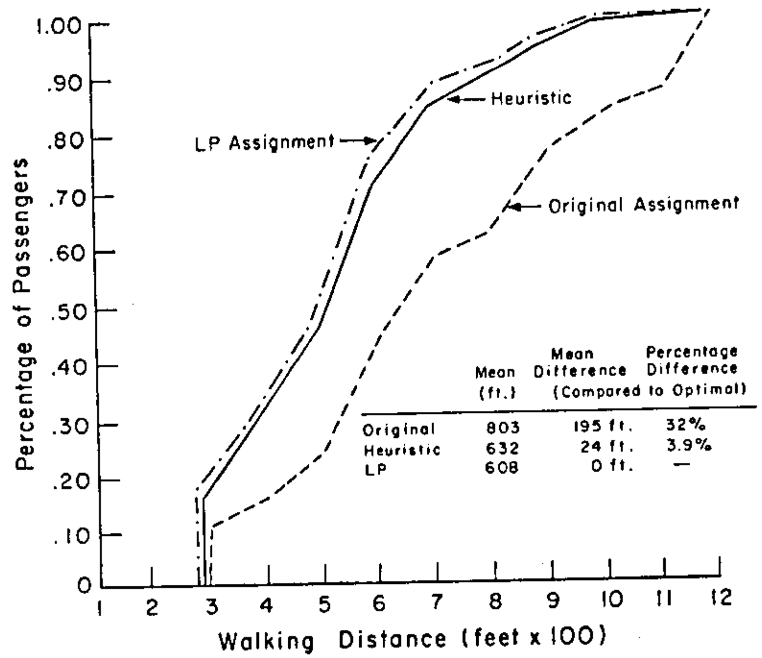

Fig. 1. Cumulative distributions of the overall mean walking distance for all passengers under each of the three assignment policies.

The cumulative distributions for each of the three categories of passengers (arriving, departing, transferring) are shown in Figures 2, 3, and 4, respectively. The greatest savings in walking distance is experienced by the departing passengers. When compared to the LP assignment, the original average walking distance per passenger was 51% higher. The heuristic's performance trails behind the LP by only 4.1%. This difference in savings among the types of passengers is due to the fact that departing passengers comprise the largest category (50% of the entire passenger population in this case).

Transfer passengers, on the other hand, do not gain any savings as a result of a change in assignment policy. In fact, the original solution offers a 1.6% shorter average transfer distance compared to the LP solution. The heuristic yields a solution which is 2.7% worse than optimal, which means that the original assignment offers a 4.3% shorter transfer distance than the heuristic. The most likely explanation for this poor performance in both solution methods is that, in the Toronto case, transfer passengers comprise only 15% of the total number of passengers, and their transfer distances are not assigned as great a weight in the objective function (Equation 2). It is not surprising that this one small component of the objective function increases. It simply identifies a

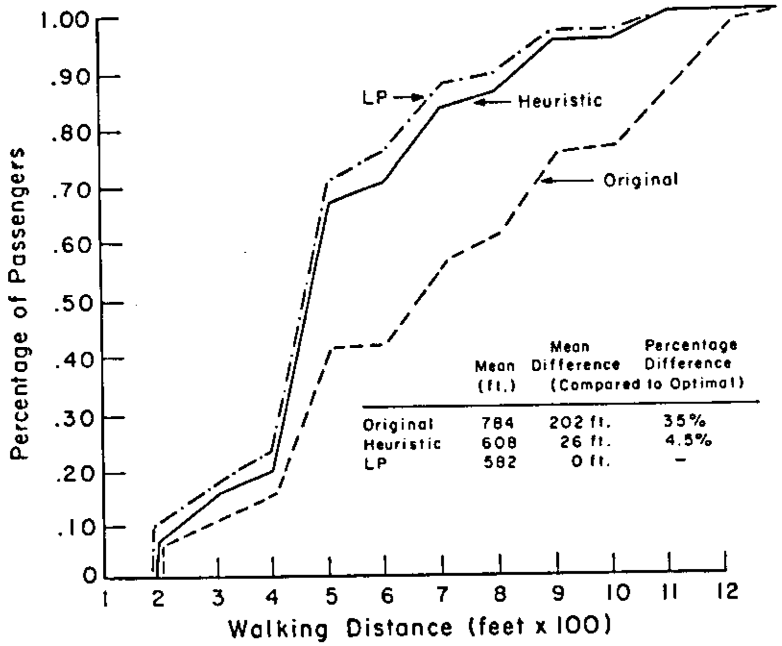

Fig. 2. Cumulative distributions of the expected walking distance for *arriving* passengers under each of the three assignment policies.

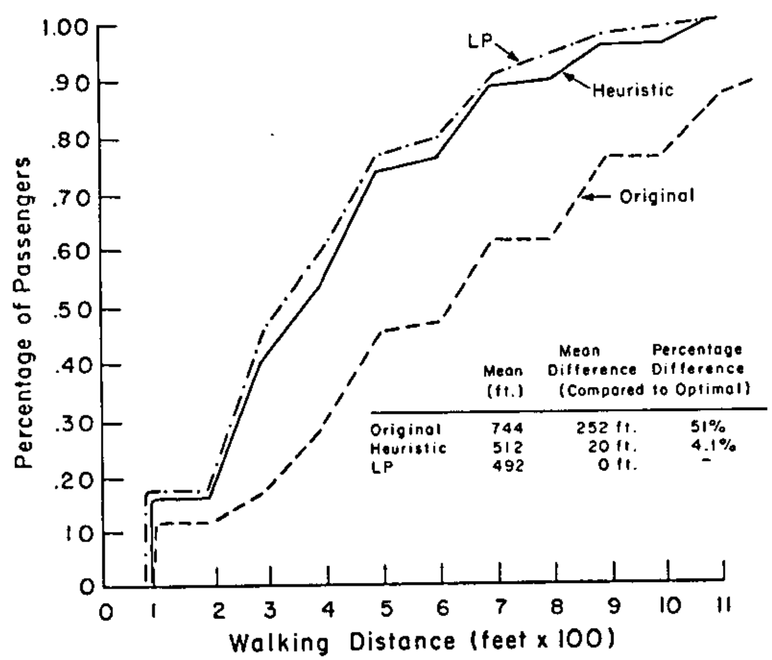

Fig. 3. Cumulative distribution of the expected walking distance for *departing* passengers under each of the three assignment policies.

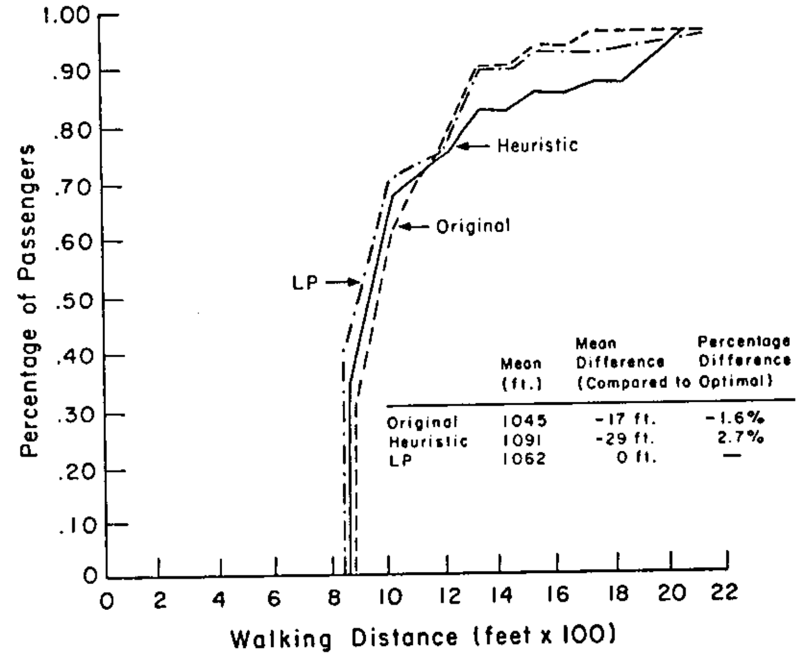

Fig. 4. Cumulative distributions of the expected walking distance for *transfer* passengers under each of the three assignment policies.

trade-off between serving a large number of originating and terminating passengers well and a small number of transfering passengers poorly.

It is often the case that two flights serve the same large numbers of transfer passengers. For instance, in the United States many flights arrive from northern cities with passengers connecting in Atlanta and destined for the South. If, in general, flights g and h are two such flights, then the problem can be handled in the following manner. First, drop the transfer walking distance contribution in the coefficients of the objective function for x_{gs} and x_{hs} for all gates $s = 1$ to N; and solve the problem. Then, if flight g is assigned to a gate, say gate z, fix $x_{gz} = 1$ and add the following constraint:

$$\textstyle\sum_{s=1}^{N} w_{zs} x_{hs} \leq D_{gh}^{\max} \tag{14}$$

$D_{gh}^{\max}$ is the maximum allowable walking distance permitted between flights g and h. Obtaining a new optimal basis should not require many additional iterations, and flight h would be assigned a gate within a distance $D_{gh}^{\max}$ of flight g.

Likewise, if it is known that a group of transfer passengers comprise the majority of passengers in two or more flights, then these flights could be assigned to gates which are near to each other. (The solution may be undesirable from the point of view of arriving and departing passengers.) There is no one best way of dealing with transfers. Each airport should adopt an appropriate solution depending on the situation at hand. If traffic flow and distance data are desired, one can collect these data using the "time-stamping" method of Braaksma,[5] where pedestrians are given cards which are time-stamped at various check points in the airport terminal. Once such information is known, one can then insure that connecting flights are positioned at nearby gates by the addition of constraints similar to those described in (14).

Another approach to the problem of transfer passengers would be to use the following formulation[1] for the gate assignment problem.

$$\text{Min } \sum_{i=1}^{M} \sum_{j=1}^{N} (p_i^a d_j^a + p_i^d d_j^d) x_{ij} + \sum_{i=1}^{M} \sum_{j=1}^{N} \sum_{g=1}^{M} \sum_{k=1}^{N} (p_{ig}^t w_{jk}) y_{ijgk} \quad (15)$$

subject to:

$$\sum_{j=1}^{N} x_{ij} = 1 \qquad \forall\, i = 1, \cdots, M \quad (16)$$

$$\sum_{h \in L(i)} x_{hj} + x_{ij} \le 1 \qquad \forall\, i = 1, \cdots, M;\ \forall\, j = 1, \cdots, N \quad (17)$$

$$\sum_{g=1}^{M} \sum_{k=1}^{N} y_{ijgk} < Q x_{ij} \qquad \forall\, i, j \quad (18)$$

$$\sum_{i=1}^{M} \sum_{j=1}^{N} y_{ijgk} < Q x_{gk} \qquad \forall\, g, k \quad (19)$$

$$\sum_{j=1}^{N} \sum_{k=1}^{N} y_{ijgk} = 1 \qquad \forall\, i, g. \quad (20)$$

All variables are as defined before, and:

p_{ig}^t = expected number of passengers transfering between flights i and g

Q = a very large constant

$$y_{ijgk} = \begin{cases} 1 & \text{if flight } i \text{ is assigned to gate } j \text{ and flight } g \text{ is assigned to gate } k \\ 0 & \text{otherwise.} \end{cases}$$

The constraints of Equations 16 and 17 are exactly the same as those of Equations 4 and 5 respectively, while (18) and (19) merely define y_{ijgk} in terms of x_{ij} and x_{gk}, respectively. The constraint in (18) states that if flight i is not assigned to gate j, then the sum over all possible assignments of flights g to gate k in y_{ijgk} must be zero. Otherwise, the sum is effectively unconstrained. Equation 19 is similar to Equation 18, but with respect to x_{gk}. Equation 20 says that y_{ijgk}, summed over all gates j and k, must equal 1 for all flights i and g. That is, for all flight pairs i and g, exactly one y_{ijgk}, of all the possible combinations of values for j and k, must equal 1.

[1] The authors would like to thank one of the referees who suggested that this formulation, although admittedly hard to solve, may well be an avenue for future research.

6. COMPUTATIONAL EXPERIENCE

ALTHOUGH the results for both the heuristic and the LP are comparable, the difference in their respective computational cost is substantial, as shown in Table IV. For the heuristic, the CPU time required was 3.4 seconds. The LP, using no initial basic feasible solution, consumed 386 CPU seconds. If the heuristic results are used as an initial basic feasible solution to start the LP, the computation time for the LP is reduced to 42 CPU seconds. A smaller test case consisting of 40 flights and 20 gates was also used and the results are shown in Table IV. This case has 600 variables and 280 constraints. The CPU time consumed was 2 seconds for the heuristic, 20 seconds for the LP, and 8 seconds if the heuristic solution is used as an initial basis. In both cases, but especially in the full scale case, the use of the heuristic solution as an initial basis was found to be very worthwhile as it brought about large savings in CPU times: 60% in the small case and 89% in the large case. All results were obtained on an IBM 370/168 in the Conversational Monitor System (CMS). There is no guarantee, however, that the above excellent performance of the heuristic is reproducible for other cases.

7. IMPLEMENTATION OF THE SOLUTION

TWO ISSUES should be discussed concerning the implementation of the solution. The first concerns aircraft gate utilization. Figure 5 is a histogram of the number of flights assigned to each gate. More than one-half of the flights are assigned to only 6 of the 20 gates. As a result, some

TABLE IV
Computation Results on an IBM 370/168

	Heuristic Algorithm	Optimization Without an Initial Basis	Optimization With an Initial Basis from Heuristic
Full Scale Case:			
Central Processing Unit (CPU time) (seconds)	3.4	386.0	42.0
Cost ($)	3.15	87.42	11.02
Test Case:			
Central Processing Unit time (seconds)	2.0	20.0	8.0
Cost ($)	1.33	4.28	1.73

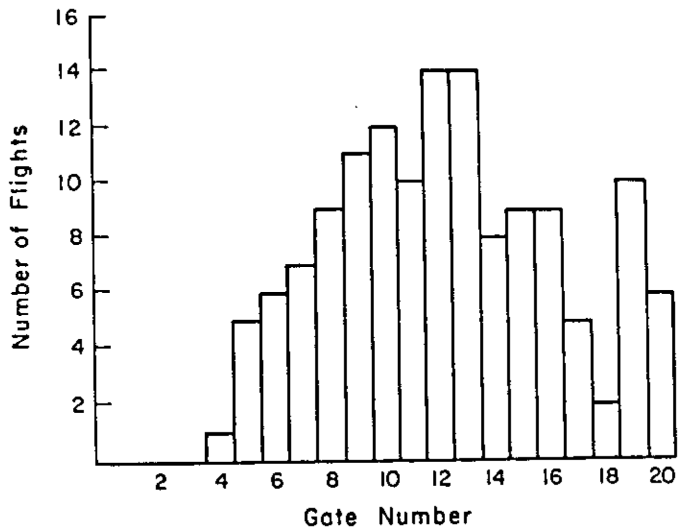

Fig. 5. Gate utilization.

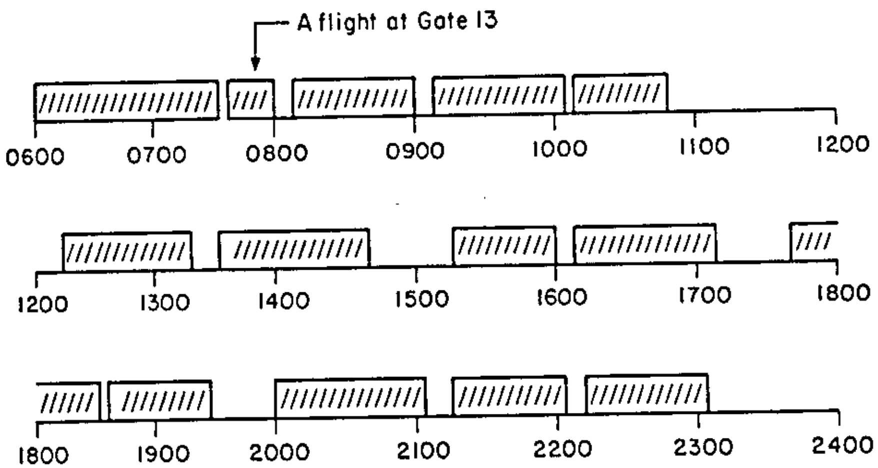

Fig. 6. Schedule for gate No. 13.

gates receive high utilization with little free time between flights. An example is gate 13 shown in Figure 6. Such an assignment may cause equipment and passenger congestion. Odier,[6] for example, discusses the situation in which apron congestion exists from aircraft parking at the Charles de Gaulle International Airport in France. His problem is resolved by setting up apron constraints. The model has been used in daily operations for Air France at Charles de Gaulle since April 1982.

The second implementation issue is schedule reliability. The problem of gate utilization becomes even more difficult when one considers that airline schedules are sometimes unreliable due to weather, air traffic control delays, etc. From the results of Toronto 2, Figure 6 shows that for gate 13 the time between the departure of one flight and the arrival of the next often does not exceed 5 minutes. A flight which leaves the gate more than 5 minutes late causes queueing problems. One solution might be to provide an additional buffer of time between flights. The assumption would generate different conflict sets than before, but the structure of the constraint matrix and the nature of the heuristic would remain the same.

8. CONCLUSIONS

THE STUDY addressed the problem of assigning flights to gates at airport terminals in such a way as to minimize passenger walking distances within the terminal. Two approaches were developed: an (integer) linear program formulation; and a heuristic. Both methods were shown, in a case study of Canada's Toronto International Airport, to produce considerable improvements over the original flight-to-gate assignment. The existing average walking distances, obtained from the original assignment, were shown to be 32% higher than the minimum possible distance given by the LP solution. The heuristic's performance was near optimal; it gave an average walking distance which was only 3.9% greater than the minimum. These results indicate that a judicious flight-to-gate assignment policy, without changes in terminal layout and without changes in flight schedules, can have a positive impact on passenger walking distances at airport terminals. Computation times for the heuristic are 3.4 CPU seconds per run, while the linear program consumes 386 seconds per run on an IBM 370/168. In addition, if the heuristic is solved first and its solution is used as an initial feasible basis for the LP relaxation of the IP, the total CPU time used to obtain optimality is reduced to 42 seconds.

These results point to the value of developing and using heuristics to obtain approximate, but very close to optimal, as well as computationally efficient solutions to problems modeled as linear programs. Even if an exact optimal solution is desired, the heuristic solution can be used as an initial basis in starting the LP for sharp reductions in computation time and storage.

REFERENCES

1. J. P. BRAAKSMA. "Reducing Walking Distances at Existing Airports," *Airport Forum*, No. 4 (August), 135–142 (1977).
2. R. S. MANGOUBI. A Linear Programming Solution to the Gate Assignment Problem at Airport Terminals, Flight Transportation Laboratory Report R80-1, pp. 41–44, Department of Aeronautics and Astronautics, MIT (June), 1980.
3. O. BABIC, C. TEODOROVIC AND V. TOSIC. "Aircraft Stand Assignment to Minimize Walking," *J. Trans. Engineer.* **110** (No. 1, January), 1984.
4. R. S. MANGOUBI. Testing Gate Assignment Algorithms at Airport Terminals, B. S. thesis, Department of Mechanical Engineering, MIT, 1978.
5. J. P. BRAAKSMA. "Time Stamping: A New Way to Survey Pedestrian Traffic in Airport Terminals," *Trans. Res. Rec.* **588**, pp. 27–34.
6. E. ODIER. Rossini Model: Optimization of Aircraft Assignment to Parking Positions. Formulation and Application to Charles de Gaulle Terminal 2, unpublished, Service de Recherche Operationnelle, Air France, 1 Square Max Hymans, 75757 Paris (May), 1983.