Flight Sequencing in Airport Hub Operations

CHING CHANG AND PAUL SCHONFELD

Airlines operating hub and spoke networks (HSNs) can reduce aircraft costs and passenger transfer times at hubs through efficient sequencing of flights. Typically, batches of flights are processed during relatively brief time "slots." When aircraft differ significantly in sizes or loads, there is a considerable potential for reducing the delay costs through efficient flight sequencing. Sequencing bigger aircraft last in and first out (BLIFO) minimizes the costs of aircraft delays, gate usage, and passenger time. Sequencing smaller aircraft first in and first out (SFIFO) maximizes the gate utilization and terminal capacity. Therefore, BLIFO is preferable when airports are not busy and gate utilization is unimportant. SFIFO is preferable when airports are very busy. Some intermediate sequences might also minimize total cost, depending on the relative costs of aircraft delays, gates, and passenger time. BLIFO or SFIFO, whichever is lower, provides a very good initial solution in most cases. A sequential pairwise exchange algorithm can then improve this initial sequence until no further improvement is possible.

Hub and spoke networks (HSNs) have been widely adopted by U.S. domestic airlines because they can greatly reduce the cost of connecting a given number of cities and improve the service frequency. When compared with direct flights, the main disadvantages of the HSN routing are additional transfer times and costs at the hub. To minimize transfer times and costs, a batch of aircraft has to arrive and depart within a short "time slot," and all aircraft should be on the ground simultaneously for at least a short period so that transfers can be made. The common ground time window (GTW) for passenger and baggage redistribution is the time between arrival of the last flight and departure of the first flight. The size of aircraft is related to passenger loads on different flights, especially for long-run scheduling purposes. Large aircraft imply expensive aircraft and large passenger loads. If the sequencing allows larger aircraft to spend less time at the hub, the costs associated with the flight sequencing will be reduced. For instance, later arrivals and earlier departures for the larger aircraft would reduce average passenger delay time and aircraft ground time cost. Thus, total transfer passenger delay and aircraft cost would be reduced if larger aircraft were the last in and first out.

Extreme sequences such as last in first out (LIFO), first in first out (FIFO), and their variants can be shown to minimize certain cost factors. Under certain traffic conditions or when certain cost factors dominate sequencing decisions, it can be shown that particular extreme sequences are optimal. In more complex cases, where no factor dominates and several factors must be traded off, sequencing solutions are also more complex. In such cases we will take the least cost extreme solution as an initial solution and use a sequential pairwise exchange algorithm to improve that initial sequence until no further improvement is possible.

C. Chang, Industrial Management Department, Chung-Hua Polytechnic Institute, Hsin Chu, Taiwan 30067, R.O.C. P. Schonfeld, Civil Engineering Department, University of Maryland, College Park, Md. 20742.

Our flight sequencing problem is to find a sequence that minimizes the total costs of passenger transfer delay, aircraft ground time, and gates. It is difficult to optimize exactly the sequence for a batch of N arrivals and N departures at a hub because there are $(N!)^2$ possible sequences. An efficient heuristic method to solve this flight sequencing problem is proposed in this report.

The literature on flight sequencing to minimize the costs of the passenger transfer delay, aircraft ground time, and gate use is scarce. Previous studies mostly focus on the Aircraft Sequencing Problem (ASP). In each of these ASP models (*1,2*), a static problem is considered, in which N aircraft are already present on holding stacks outside the terminal area. Each aircraft can land at any time, and the problem is to find the sequence that maximizes runway capacity (or utilization) or, alternatively, minimizes delays. Dear (*1*) examined the dynamic case of the ASP in which the composition of the set of aircraft varies over time. Psaraftis (*3*) developed a dynamic programming (DP) approach for sequencing N groups of aircraft landing at an airport to minimize total passenger delays. Dear and Sherif (*4*) examined the constrained position shifting methodology, with simulation from the perspectives of both pilots and air traffic controllers, and later (*5*) developed a computer system to assist the sequencing and scheduling of terminal area operations. Bianco et al. (*2*) proposed a combinatorial optimization approach to the ASP, in which maximizing the runway capacity or utilization was modeled as an n job (landing or takeoff) and one-machine (runway) scheduling problem with non-zero ready times. Venkatakrishnan et al. (*6*) developed a statistical model for the landing time intervals between successive aircraft using data from Logan Airport in Boston. They found that reordering the sequence of landing aircraft could substantially reduce the landing time intervals and thereby increase runway capacity. Considering stochastic aircraft arrivals, Hall and Chong (*7*) developed a model for scheduling flight arrivals and departures to minimize delays for passengers connecting between aircraft at a hub terminal.

A review of the aforementioned studies indicates that deterministic models for optimizing runway capacity or utilization have received considerable attention. However, it is also important to consider the costs of passenger transfer delay, aircraft ground time, and gate use at hub airports. In addition, departure sequences are interrelated with arrival sequences (for instance, due to minimum ground time constraints, and desirability of replacing departing aircraft with similarly sized arriving aircraft to improve gate utilization) and should be determined jointly. The flight sequencing problem considered here is to find the arrival and departure sequences at a hub that minimizes those three costs.

SYSTEM DEFINITIONS

A system of hub airports is defined as follows.

Route Network

An HSN that has one hub airport and N spoke city airports (Figure 1) is considered here. All spoke routes connect at the hub. To travel from one spoke city to another, passengers must transfer at the hub. Nonstop travel is possible only if the origin or destination is at the hub. A more general system would have multiple hubs.

Batch Arrivals and Batch Departures

A group of flights from various spoke cities arrive at the hub airport within a short time period, and then unload and load passengers and baggage during a common GTW; then, the same aircraft leave within a short departure period. If there are N arriving aircraft, there are also N departing aircraft.

Sequence

The sequence is the order of aircraft arrivals and departures. Two extreme sequences, FIFO and LIFO, are of special interest if aircraft are ranked by size or load. FIFO is the sequence in which aircraft depart in their order of arrival [Figure 2(a)]. LIFO is the sequence in which aircraft depart in their reverse arrival order [Figure 2(b)]. LIFO is interesting because it may allow larger aircraft and their passengers to arrive later and leave earlier, with considerable savings. FIFO is interesting because it can reduce slot durations. At busy airports in which gate utilization and terminal capacity are critical, a FIFO sequence can provide shorter intervals among successive batches of flights, as discussed later in this paper. Two extreme FIFO sequences are considered in which the aircraft order is by size. These are SFIFO, in which smaller aircraft are first, and BFIFO, in which bigger aircraft are first. Likewise, the LIFO options include SLIFO, in which smaller aircraft arrive last and depart first, and BLIFO, in which bigger aircraft arrive last and depart first.

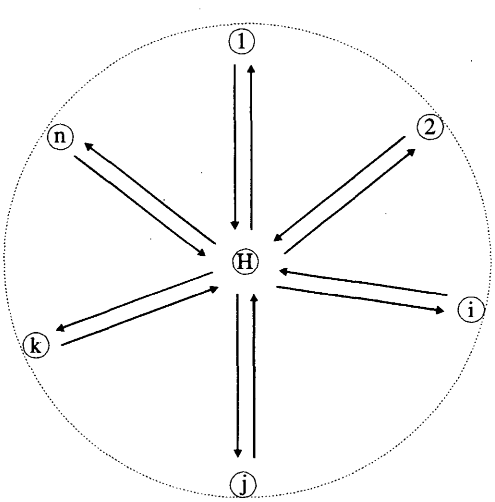

FIGURE 1 Hub and spoke network.

Cycle Time

The cycle time is the interval between the first arrival and the last departure in a batch of flights, along with the buffer separation time (Figure 3). The cycle time has four components:

1. The arrival period is the sum of all interarrival times, which is $\Sigma_{i=1}^{N-1} A_i$, where N is the total number of aircraft, and A_i is the interarrival time between the ith and $i + 1$st arrivals.
2. The GTW is the common time when all aircraft are simultaneously at the terminal, for transfer purposes. (However, transfer activities can start before the GTW and continue after its end.)
3. The departure period is the sum of all interdeparture times, which is $\Sigma_{i=1}^{N-1} D_i$, where D_i is the interdeparture time between the ith and $i + 1$st departures.
4. Buffer separation time, q, is the minimum separation time between successive slots, which is constrained by reliability considerations.

The first three time components are available for passengers, baggage, and cargo transfer activities. Aircraft are ready for departure after loading and servicing processes are completed.

Slot Sequences

Here, the time between the first aircraft arrivals of two consecutive batches is called a time slot. Figure 3(a) shows that if cycles do not overlap, the slot duration equals the cycle time. However, if cycles overlap [Figure 3(b)], the slot duration is smaller than the cycle time.

Figure 4 shows two types of slot sequences.

1. Overlapping cycles are possible if the departure sequence in the leading slot is similar to the arrival sequence in the trailing slot, as when:
   a. All slots are SFIFO or BFIFO [Figure 4(a)],
   b. Any pair of successive slots includes one SLIFO and BLIFO [Figure 4(b)].
2. Other nonoverlapping cycles can have
   a. Random sequences,
   b. Alternating SFIFO and BFIFO slots [Figure 4(a)],
   c. Similar LIFO sequences; that is, all SLIFO or all BLIFO slots [Figure 4(a)].

COST FUNCTIONS

Three cost components reflect the effects of different batch sequences. These are the total passenger transfer delay cost (C_p), the total aircraft ground time cost (C_a), and the total gate cost (C_g). Local passengers (originating or terminating at the hub) can be excluded in these total cost functions because there is no difference between HSN routing and direct flights. All these costs are computed in the same units ($/slot). The total relevant cost function is

$$C = C_p + C_a + C_g \tag{1}$$

where C = total cost per slot for the hub operation ($/slot).

We define these three component costs as follows.

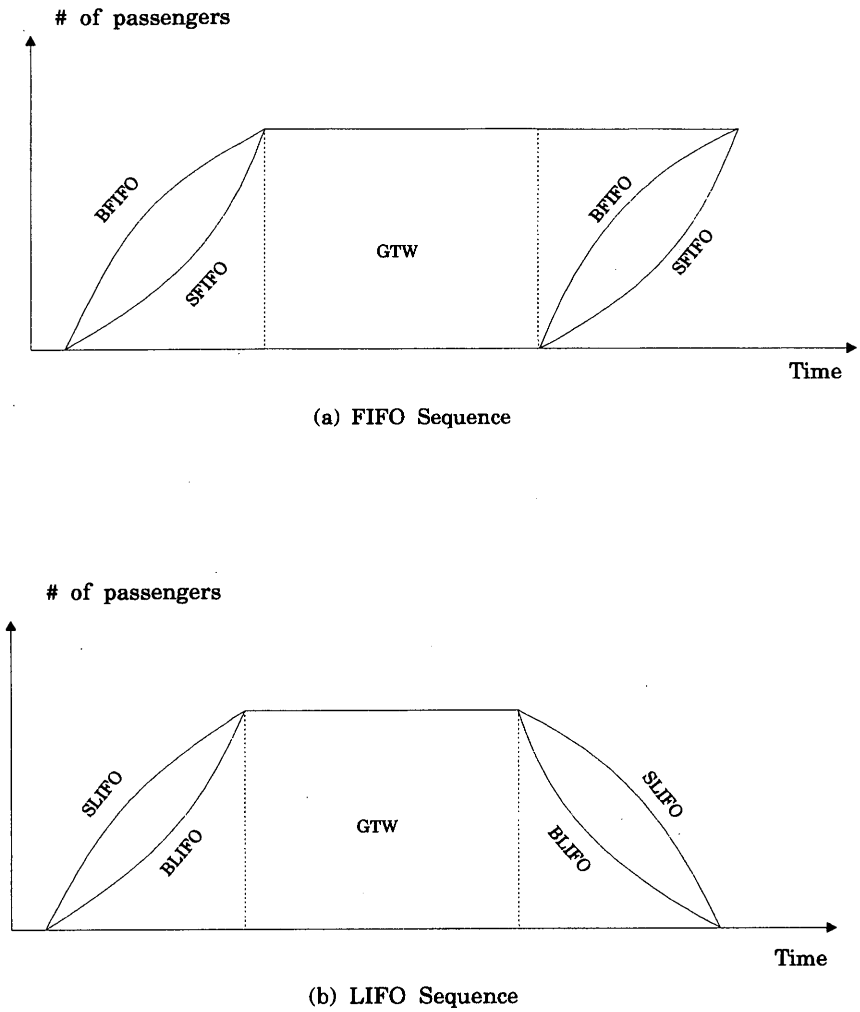

FIGURE 2 FIFO and LIFO sequences.

a. The passenger transfer delay is incurred by redistributing transfer passengers and their baggage from their original aircraft to their destination aircraft at the hub. The delay for each passenger is the difference between the passenger's departure time from the hub and the passenger's arrival time at the hub. Therefore, the total passenger transfer delay cost is the sum of delay costs for all transfer passengers in a batch of arrival and departure flights

$$C_p = V_p\left(\sum_{i=1}^{N} \sum_{j=1}^{N} P_{ij}\, t_{ij}\right) \tag{2}$$

where

C_p = total passenger transfer delay cost per slot (\$/slot),
v_p = time value of transfer passengers (\$/passenger hour),
P_{ij} = number of transfer passengers from arrival aircraft i to departure aircraft j (passengers), and
t_{ij} = the transfer delay time from arrival aircraft i to departure aircraft j (hours/slot).

b. The aircraft ground time is the time that an aircraft dwells at the hub. For HSNs, a batch of aircraft arrive and depart within a slot and all aircraft are on the ground simultaneously for at least a short period so that transfers can be made. An aircraft's ground time is determined by its arriving and departing times. The total aircraft ground time cost is the sum of ground time cost for all aircraft

$$C_a = \sum_{i=1}^{N} a_i\, v_{ai} \tag{3}$$

where

C_a = total ground time cost for all aircraft in a slot (\$/slot),
a_i = the time aircraft i is on the ground (hours/slot), and
v_{ai} = ground time value of aircraft i (\$/hour).

c. The total gate cost includes the hourly gate fixed cost and hourly gate usage cost. The fixed cost accounts for the gate construction and equipment installation. The usage cost is incurred when an aircraft parks and uses a gate. Because gates can have different characteristics in the same terminal, the gate fixed cost depends on the gate size and slot duration, and the gate usage cost depends on gate size and gate occupancy time

$$C_g = \sum_{i=1}^{N} (v_{ci}\, t_{gi} + v_{oi}\, t_{oi}) \tag{4}$$

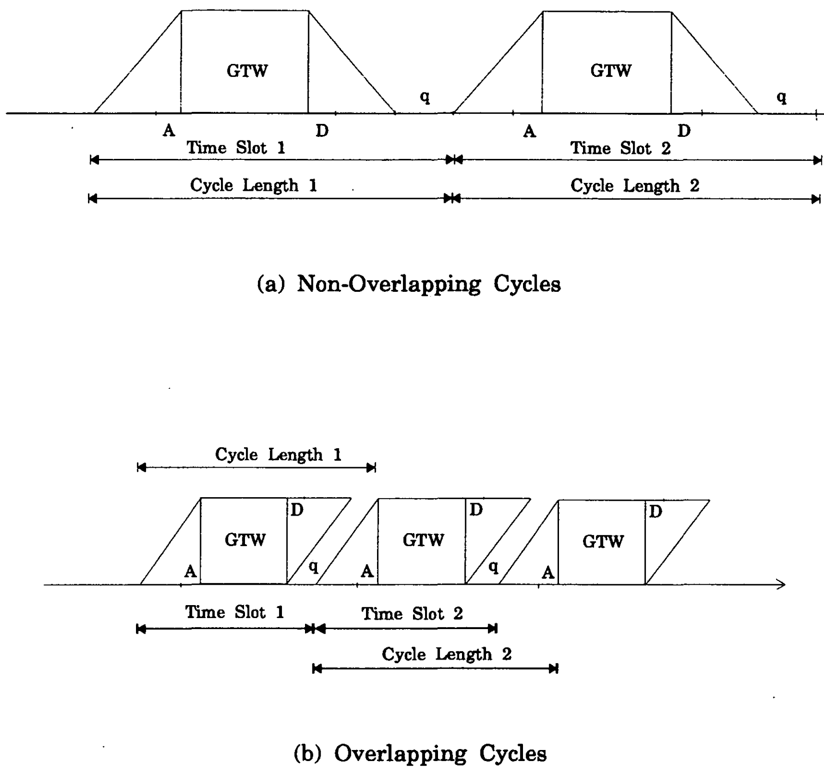

FIGURE 3 Overlapping and nonoverlapping cycles.

where

C_g = total gate cost per slot ($/slot),
v_{ci} = fixed cost of gate i ($/hour),
v_{oi} = usage cost of gate i ($/hour),
t_{oi} = gate i's occupancy time (hours/slot), and
t_{gi} = slot duration (hours/slot).

CONSTANT SLOT DURATION

A simplified case with constant slot duration is considered first, on the basis of these assumptions:

1. All transfer passengers considered arrive and depart within the same slot, and all transfer activities are completed within that slot.

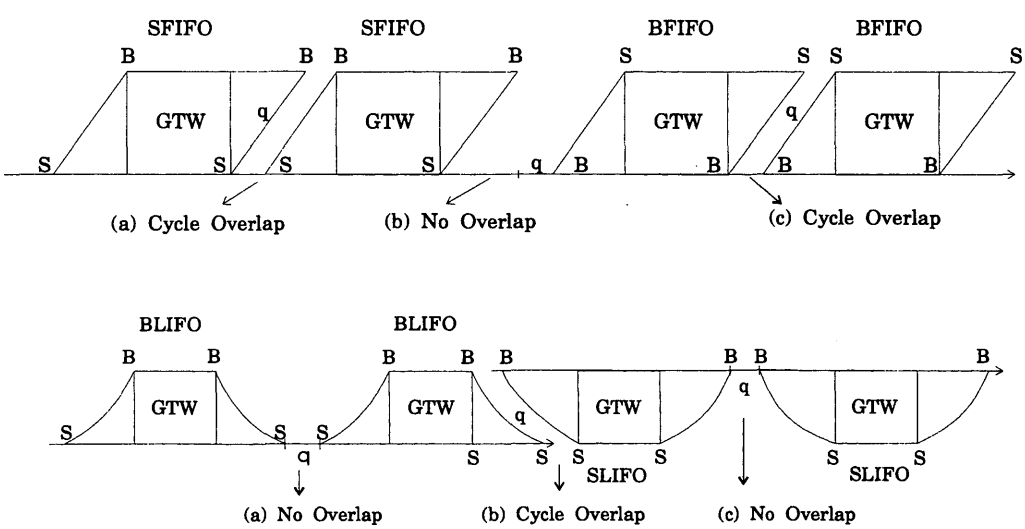

FIGURE 4 (*a*) Possible FIFO sequences; (*b*) LIFO sequences.

2. The number of gates, G, is greater than or equal to the number of aircraft, N.
3. The ground time window (T) is a constant.
4. Aircraft arrive punctually.

No other scheduling constraints limiting flight arrival and departure times should be considered in this problem. The ground activities of an aircraft include unloading passengers, baggage, and cargo; and cleaning, refueling, and loading passengers, baggage, and cargo. A, D, and T are fixed quantities and are assumed in this case to be independent of the sizes of aircraft. The buffer separation time between two time slots is q. Therefore, the slot duration is constant.

Analysis of Sequences

We first explore the extreme sequences LIFO and FIFO to determine in what situations they actually yield optimal solutions and then consider how more complex cases can be solved.

LIFO Sequence

In the LIFO sequence aircraft depart in their reverse order of arrival. Thus, a LIFO sequence benefits later arrivals and makes earlier arrivals a disadvantage. If larger aircraft (with higher costs per aircraft hour and passenger loads) are required to arrive later than smaller aircraft, such a BLIFO sequence minimizes the total passenger transfer delay cost, aircraft ground time cost, and gate usage cost. Figure 5 shows how BLIFO minimizes the total passenger transfer delay cost; the abscissa is time, and the ordinate is the cumulative number of passengers. The areas covered by the arrival curve, the (GTW), and departure curve in Figure 5 represent the total passenger transfer delay. Because the GTW has a fixed value, we only need to consider the areas covered by the arrival and departure curves. The slopes of arrival or departure curves represent the number of arriving or departing passengers per time unit; that is, the passenger departure rate. It can be observed in Figure 5 that the area under the BLIFO arrival and departure curves is smaller than areas for any other sequences. Thus, BLIFO minimizes the total passenger transfer delay cost. To minimize the total passenger transfer delay, aircraft should arrive in ascending order of their passenger arrival rates (the number of arriving passengers/runway time unit) and depart in descending order of their passenger departure rates.

Because larger aircraft may need more ground time than smaller ones, the BLIFO departure sequence must be modified to consider which aircraft are actually ready to leave. Smaller aircraft that are ready early need not wait for the unready larger aircraft. Accordingly, the aircraft departure sequence can be modified as follows.

Step 1. Check all aircraft to find which ones are ready to leave.

Step 2. Sort all ready aircraft in descending order of their passenger departure rates, and let the aircraft with the largest passenger departure rate leave.

Step 3. Check the unready aircraft. If new aircraft become ready to leave, let them join the list of ready aircraft. Go to Step 2.

The way in which BLIFO minimizes total aircraft ground time cost can also be explained graphically. An aircraft's dwell time is the interval between its arrival time and departure time. Figure 5 shows that with BLIFO, larger aircraft have smaller dwell times. Since for BLIFO $a_1 \leq a_2 \leq \ldots \leq a_N$ and $v_{a1} \geq v_{a2} \geq \ldots \geq v_{aN}$, where a_i is the ith largest aircraft dwell time and v_{ai} is the ith largest aircraft's time value, BLIFO minimizes the total aircraft ground time cost (Equation 3).

A similar argument can be used to show that BLIFO minimizes total gate usage cost since in Equations 3 and 4 the total gate usage cost function has the same structure as the total aircraft ground time cost function. Consequently, BLIFO minimizes total passenger transfer delay cost, total aircraft ground cost, and total gate usage cost, but not the total fixed cost of gates.

FIFO Sequence

When gates differ in size and cannot all accommodate the largest aircraft, the sequence of flights depends on the order in which gates of different sizes become available after handling the previous batch of aircraft. With the gate-aircraft size compatibility restriction, a BLIFO slot cannot closely follow a preceding BLIFO slot. This reduces the gate utilization and terminal capacity, which are very

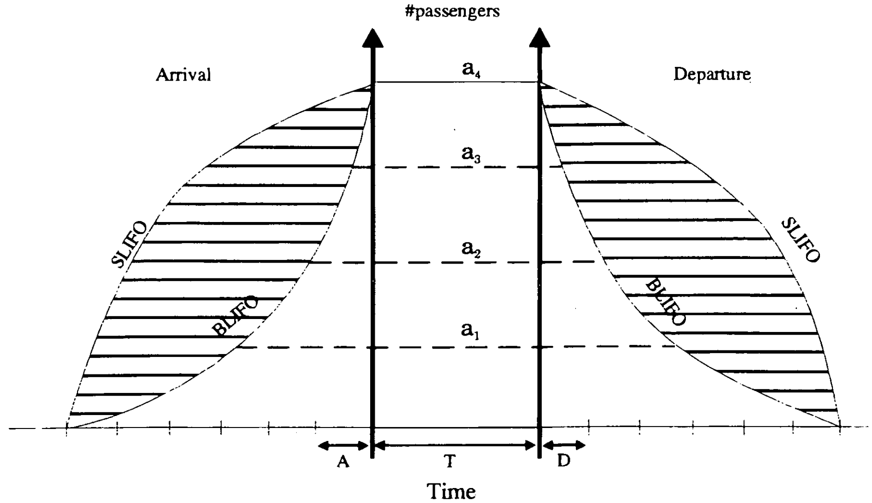

FIGURE 5 BLIFO and SLIFO sequences.

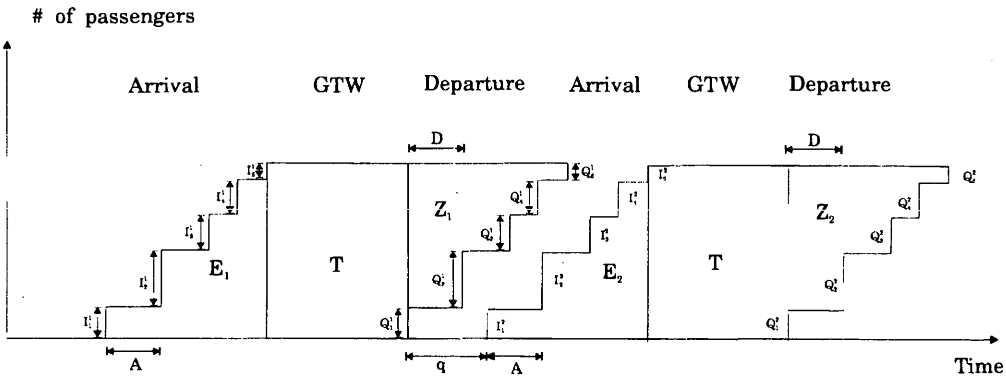

FIGURE 6 Overlapping slot sequences.

important at busy airports. To maximize the gate utilization and terminal capacity, the slots must overlap tightly. Two succeeding slots can overlap tightly if the departure sequence in the leading slot is similar to the arrival sequence in the trailing slot (Figure 6). FIFO yields tightly overlapping sequences for successive slots when the departure sequence in the earlier slot is the same as the arrival sequence in the later slot. Thus, FIFO can increase gate utilization and terminal capacity. Two extreme cases of FIFO, namely SFIFO and BFIFO, significantly affect the total passenger transfer delay when slots must overlap tightly. To minimize the total passenger transfer delay, the areas of Z_t and E_t in Figure 6, where t is the slot number, should be minimized. When interarrival times (A) and interdeparture times (D) have fixed values, the least transfer delay sequence minimizes areas $(E_1 + Z_1)$ in Figure 6, where $t = 1$. For instance, assume that there are five aircraft in each slot in Figure 6. Equation 7 represents the total passenger transfer delay. In minimizing total delay, subject to the overlapping slot constraint, the following results are obtained:

$$\text{Area } E_t = 4AI_1^t + 3AI_2^t + 2AI_3^t + AI_4^t \quad (5)$$

$$\text{Area } Z_t = DQ_2^t + 2DQ_3^t + 3DQ_4^t + 4DQ_5^t \quad (6)$$

$$\text{Min Area } (E_t + Z_t) = 4AI_1^t + (3AI_2^t + DQ_2^t) + 2AI_3^t + 2DQ_3^t) + AI_4^t + 3DQ_4^t) + 4DQ_5^t \quad (7)$$

where

I_m^t = the total number of transfer passengers on the mth arrival aircraft in slot t, and

Q_m^t = the total number of transfer passengers on the mth departure aircraft in slot t.

If $I_m^t \approx Q_m^t$ (the number of the transfers on mth arrival aircraft is similar to the number of the transfers on the mth departure aircraft), for all m, the following is true:

a. If $A > D \Rightarrow \{I_1^t < I_2^t < I_3^t < I_4^t < I_5^t\}$ and $\{Q_1^t < Q_2^t < Q_3^t < Q_4^t < Q_5^t\}$ minimizes areas of $(E_t + Z_t)$. This sequence is SFIFO.

b. If $A < D \Rightarrow \{I_1^t > I_2^t > I_3^t > I_4^t > I_5^t\}$ and $\{Q_1^t > Q_2^t > Q_3^t > Q_4^t > Q_5^t\}$ minimizes areas of $(E_t + Z_t)$. This sequence is BFIFO.

c. If $A = D \Rightarrow$ all FIFO sequences have the same transfer delay.

Accordingly, either the SFIFO or BFIFO flight sequence minimizes total passenger transfer delay when slots must overlap tightly and $I_m^t \approx Q_m^t$, for all m. However, if $I_m^t \not\approx Q_m^t$, for all m, neither SFIFO nor BFIFO guarantees the minimum total passenger transfer delay.

Similarly, it is easy to find a sequence that minimizes total aircraft ground cost and gate usage cost since both costs are related to aircraft sizes and their dwell times. Figure 6 shows that the following properties are true when slots must overlap tightly. (It should be noted that here I_m^t need not be equal to Q_m^t, for all m, since both costs are not related to the passenger loads.):

d. If $A > D$, SFIFO minimizes total aircraft ground time cost and total gate usage cost.

e. If $A < D$, BFIFO minimizes total aircraft ground time cost and total gate usage cost.

f. If $A = D$, all FIFO sequences have the same total aircraft ground time costs and total gate usage costs.

For this simple case when slots must overlap tightly, either SFIFO or BFIFO minimizes total aircraft ground time cost and gate usage cost.

When slots overlap tightly and $I_m^t \approx Q_m^t$, for all m, the least total cost sequence is:

g. SFIFO, if $A > D$.

h. BFIFO, if $A < D$.

i. All FIFO sequences have the same total costs, if $A = D$.

If $I_m^t \not\approx Q_m^t$, for all m, the above results may not be true. However, (*d*), (*e*), and (*f*) are still true when slots must overlap tightly. If neither SFIFO nor BFIFO minimizes total passenger transfer delay, the sequence which minimizes total passenger transfer delay has higher total costs of aircraft ground time and gate usage. Therefore, the total cost of SFIFO, if $A > D$, or BFIFO if $A < D$, is very close to the minimum total cost when slots must overlap tightly (*8*).

3.2. Preferable Sequence

When an airport is not busy and the gate utilization (gate fixed cost) can be ignored, BLIFO is preferable because it minimizes the total passenger transfer delay, aircraft ground time, and gate usage cost. If aircraft are not ready to leave as soon as BLIFO sequence requires, the BLIFO departure sequence should be modified as in the LIFO sequence already described.

When an airport is very busy and slots must overlap tightly, a FIFO sequence (specifically SFIFO if $A > D$ and BFIFO otherwise) maximizes gate utilization as well as terminal capacity and is the least total cost sequence if $I_m^t \approx Q_m^t$, for all m. When $I_m^t \not\approx Q_m^t$, for all m, neither SFIFO nor BFIFO may be the least total cost overlapping sequence. However, either SFIFO or BFIFO is still preferable because the total cost of SFIFO or BFIFO is very close to the minimum total cost.

When an airport's condition is moderately busy, trade-offs among passenger time, aircraft costs, and gate cost may lead to a least total cost sequence in between extreme sequences such as BLIFO, BFIFO, or SFIFO. Moreover, the time values of passengers, aircraft, and gates vary in different times and places. In such cases, the least total cost sequence may be found by starting from some initial solution and using the sequential pairwise exchange algorithm to try swapping aircraft positions in the sequence until no further improvement is possible. We can choose the best extreme solution (i.e., BLIFO or SFIFO if $A > D$ and BFIFO otherwise) as our initial solution and then improve it with a systematic exchange algorithm. The total number of exchanges is $N(N - 1)/2$, where N is the total number of aircraft, for example, [1,2], [1,3], . . . , [1,N], [2,3], [2,4], . . . , [$N - 2$,$N - 1$], [$N - 2$,N], [$N - 1$,N]. For instance, assume that $A > D$. Our sequential pairwise exchange algorithm to improve the flight sequencing is as follows.

Step 1. Compute the total costs of SFIFO and BLIFO. The one with the lower total cost is the initial solution. Store its total cost.

Step 2. Sequentially choose a pair of aircraft and exchange their arrival orders. Compute the new total cost.

Step 3. If the new total cost is below the previous one, substitute it and store the new arrival sequence. Otherwise, keep the previous sequence. Go to Step 2.

This algorithm was used by Chang (*8*) and had a reasonable computation time.

VARIABLE SLOT DURATION

When the interarrival and interdeparture times are variable and the GTW is constant, slot duration differs for various flight sequences. If an airline accounts for a significant fraction of the flights at an airport, the runway capacity directly affects the interarrival and interdeparture times of an airline's batch of connecting flights. One key factor that can affect the interarrival and interdeparture times is the minimum separation required by FAA to guard against wake-vortex turbulence (*9*). The wake-vortex separation depends on weights of the leading and following aircraft. Three weight classes of aircraft (heavy, large, and small) must be considered.

Minimum Separation Requirement

Let A_{ij} be the interarrival time between two successive landing aircraft i and j, and D_{ij} be the interdeparture time between two successive take-off aircraft i and j, where both i and j are aircraft size indices. Aircraft are ordered and labeled according to decreasing size; for example, {1, 2, 3, 4} are heavy aircraft, {5, 6, 7, . . . , 10} are large aircraft, and {11, 12, . . . , N} are small aircraft. Let the time period between the first arrival and the last arrival be called total arrival time, and the time period between the first departure and last departure be called total departure time. Based on the FAA's minimum separation regulation, the following properties exist:

a. $A_{ij} \leq A_{ji}$, if $i \geq j$,
b. $D_{ij} \leq D_{ji}$, if $i \geq j$,
c. $A_{ij} \geq D_{ij}$, for all ij pairs.

Overlapping Sequence

In order to maximize the gate utilization and terminal capacity, slots should overlap tightly. When interarrival times and interdeparture times are dependent on the relative weight classes of two successive landing and takeoff aircraft, FIFO can still increase the gate utilization. Assume that departure processes are fixed. Based on these properties, if aircraft arrive in the order of {Small, Large, and Heavy}, the minimum total arrival time is obtained. Due to property (*a*), A_{ij} would be smaller than A_{ji} if $i \geq j$. In order to minimize total arrival time, small aircraft should land before large aircraft. Similarly, for a fixed arrival process, in order to minimize total departure time, smaller aircraft should take off before larger aircraft. Accordingly, SFIFO minimizes cycle length and slot duration since SFIFO has the smallest total arrival and departure times. Therefore, when slots must overlap tightly, SFIFO is the overlapping slot sequence that maximizes the gate utilization and terminal capacity.

Because the interdeparture time is slightly shorter than the interarrival time, SFIFO benefits larger aircraft. This implies that SFIFO is the overlapping sequence that minimizes the total cost of aircraft ground time.

SFIFO has the smallest gate time and can minimize total gate fixed cost because the shortest slot duration sequence yields the highest gate utilization. In addition, SFIFO minimizes the total aircraft ground time. Therefore, SFIFO also minimizes total gate usage cost. Consequently, SFIFO is the overlapping sequence with the least total gate cost.

On the basis of the results of the constant slot duration case, if interarrival time, A_{ij}, is greater than interdeparture time, D_{ij}, for all i and j, and $I_m^t \approx Q_m^t$ (the number of the transfers on mth arrival aircraft is similar to the number of the transfers on the mth departure aircraft), for all m, SFIFO is the overlapping sequence with the least total passenger transfer delay.

With SFIFO, similarly sized aircraft arrive or depart together, consistent with the principle of grouping takeoffs and landings of similarly sized aircraft (*6*). Similarly sized aircraft land or take off together and average interarrival time and interdeparture time are minimized. Therefore, SFIFO maximizes runway capacity in such hub operations.

Thus, SFIFO is the least total cost overlapping sequence if $I_m^t \approx Q_m^t$, for all m. Otherwise, SFIFO may not minimize total passenger transfer delay. The SFIFO Sequence section of this paper also indicates that the total passenger transfer delay of SFIFO is close to the optimal value. Moreover, SFIFO still minimizes total aircraft ground time cost and total gate cost (including gate usage and gate fixed costs) when slots must overlap tightly. Thus, SFIFO is a near-optimal overlapping sequence because its total cost is close to the minimum total cost when slots must overlap tightly.

4.3. Nonoverlapping Sequence

When an airport is not busy and gate utilization is unimportant, BLIFO is still preferable. BLIFO is the sequence in which aircraft arrive in ascending order of their passenger arrival rates (the number of arriving passengers per runway time unit) and depart in descending order of their passenger departure rates. This always benefits large aircraft and reduces total cost significantly. However, BLIFO may have a longer slot duration than SFIFO. The total interarrival time for BLIFO is the same as that for SFIFO, but the total interdeparture time for BLIFO is greater than that for SFIFO. For instance, assume {1, 2, 3,4} are heavy aircraft, {5, 6, 7, 8} are large aircraft, and {9, 10} are small aircraft. Let the departure sequence of SFIFO be (10, 9, 8, 7, 6, 5, 4, 3, 2, 1). The departure sequence of BLIFO is (1, 2, 3, 4, 5, 6, 7, 8, 9, 10). Therefore, the difference of the slot durations between BLIFO and SFIFO is the difference of the separations, that is, $\sigma_{i=1}^{9} (D_{i,i+1} - D_{i+1,i}) = D_{45} - D_{54} + D_{89} - D_{98}$, divided by the take-off speed. Other interdeparture times are the same (e.g., $D_{12} = D_{21}$) since interdeparture times for SFIFO and BLIFO are equal if two successive takeoff aircraft are in the same weight class. For instance, if all aircraft are in the same weight class, BLIFO and SFIFO have the same slot duration. Since FAA defines only three weight classes and BLIFO is also consistent with the principle of grouping landings or takeoffs of similarly sized aircraft, the difference in total departure times between SFIFO and BLIFO is small. This difference can be ignored if the number of aircraft in a slot is large. BLIFO has a very small cycle time. The arguments used in the section LIFO Sequence can also be used to show that BLIFO with variable interarrival and interdeparture times minimizes the costs of total passenger transfer delay, total aircraft ground time, and total gate usage. The BLIFO departure sequence can again be modified to deal with unready flights with the procedures described in the LIFO Sequence section.

CONCLUSIONS

The flight-sequencing problem considered here is to seek an efficient flight sequence that minimizes the total costs of passenger transfer delay, total aircraft ground time, and gates. When aircraft differ significantly in size or load, there is considerable potential for reducing the costs through efficient flight sequencing. In addition, aircraft landings and takeoffs must satisfy the minimum separation requirement. The interarrival times and interdeparture times depend on the weight classes of two successive aircraft landings or takeoffs. The flight-sequencing disciplines that favor large aircraft such as SFIFO and BLIFO may minimize the considered total cost under some circumstances. Even if SFIFO or BLIFO does not minimize the total cost, one of them (the one with the lower total cost) will be a good initial solution for the flight sequence, which can then be improved with the sequential pairwise exchange algorithm.

When an airport is not busy, the gate utilization is less important and gate-fixed cost can be neglected. BLIFO is then preferable since it minimizes the costs of total passenger transfer delay, total aircraft ground time, and total gate usage. When an airport becomes busy, the gate utilization and terminal capacity become more critical and slots should overlap tightly. SFIFO is the least total cost overlapping sequence if $I'_m \approx Q'_m$, for all m. However, if $I'_m \not\approx Q'_m$, for all m, then SFIFO may not minimize total passenger transfer delay. When SFIFO does not minimize total passenger transfer delay, the sequence that minimizes total passenger transfer delay has higher total costs of aircraft ground time and gate usage because SFIFO minimizes these two costs. Besides, total passenger transfer delay of SFIFO is close to the optimal value. Without $I'_m \approx Q'_m$, for all m, SFIFO may not be the optimal overlapping sequence but is still near-optimal.

When an airport is moderately busy, neither BLIFO nor SFIFO may be the optimal sequence. In addition, the time values of passengers, aircraft, and gates vary in different times and places. As the time value of the gates increases relative to other costs, SFIFO is increasingly preferable to BLIFO. To find the optimal sequence, BLIFO or SFIFO, whichever has the lower total cost, is used to be the initial solution and improved by the sequential pairwise exchange algorithm until no further improvement is possible. However, this improved sequence may not be the exact optimal sequence since the flight sequencing problem is an NP-hard problem (*2*) when an airport is moderately busy.

In this report, the GTW is assumed to be independent of flight sequence, even though the minimum ground times of smaller and larger aircraft are considered. Improved models should explicitly consider how the GTW is affected by a flight sequencing. In addition, the flight sequencing and gate assignment are interdependent. In a previous report Chang (*8*) has analyzed a more realistic flight sequencing problem with a variable ground time window and combined gate assignment, and also has provided extensive numerical results.

REFERENCES

1. Dear, R.G. *The Dynamic Scheduling of Aircraft in the Near Terminal Area.* MIT Flight Transportation Laboratory Report R-76-9, MIT, Cambridge, Mass., 1976.
2. Bianco, L., G. Rinaldi, and A. Sassano. A Combinatorial Optimization Approach to Aircraft Sequencing Problem. *Proc., NATO Advanced Research Workshop on Flow Control of Congested Networks*, Capri, Italy, October 12–18, 1986, pp. 323–339.
3. Psaraftis, H.N. A Dynamic Programming Approach for Sequencing Groups of Identical Jobs. *Operations Research*, Vol. 28, No. 6, 1980, pp. 1247–1359.
4. Dear, R.G., and Y.S. Sherif. The Dynamic Scheduling of Aircraft in High Density Terminal Area. *Microelectronics and Reliability*, Vol. 29, No. 5, 1989, pp. 743–749.
5. Dear, R.G., and Y.S. Sherif. An Algorithm for Computer Assisted Sequencing and Scheduling of Terminal Area Operations. *Transportation Research*, Part A, Vol. 25, Nos. 2/3, 1991, pp. 129–139.
6. Venkatakrishnan, C. S., A. Barnett, and A.R. Odoni. Landings at Logan Airport: Describing and Increasing Airport Capacity. *Transportation Science*, Vol. 27, No. 3, 1993, pp. 211–227.
7. Hall, R.W., and C. Chong. Scheduling Timed Transfers at Hub Terminals. *Transportation and Traffic Theory* (C.F. Daganzo, ed.), 1993, pp. 217–236.
8. Chang, C. *Flight Sequencing and Gate Assignment at Airport Hubs.* Ph.D. Dissertation, University of Maryland at College Park, 1994.
9. *1991–92 Aviation System Capacity Plan*, Federal Aviation Administration, Washington, D.C.

Publication of this report sponsored by Committee on Aviation Economics and Forecasting.