Scheduling Aircraft Landings—The Static Case

J. E. BEASLEY

The Management School, Imperial College, London SW7 2AZ, England

M. KRISHNAMOORTHY

CSIRO Mathematical and Information Sciences, Private Bag No. 10, Clayton South MDC, VIC 3169, Australia

Y. M. SHARAIHA

The Management School, Imperial College, London SW7 2AZ, England

D. ABRAMSON

Department of Digital Systems, Monash University, Clayton, VIC 3169, Australia

In this paper, we consider the problem of scheduling aircraft (plane) landings at an airport. This problem is one of deciding a landing time for each plane such that each plane lands within a predetermined time window and that separation criteria between the landing of a plane and the landing of all successive planes are respected. We present a mixed-integer zero–one formulation of the problem for the single runway case and extend it to the multiple runway case. We strengthen the linear programming relaxations of these formulations by introducing additional constraints. Throughout, we discuss how our formulations can be used to model a number of issues (choice of objective function, precedence restrictions, restricting the number of landings in a given time period, runway workload balancing) commonly encountered in practice. The problem is solved optimally using linear programming-based tree search. We also present an effective heuristic algorithm for the problem. Computational results for both the heuristic and the optimal algorithm are presented for a number of test problems involving up to 50 planes and four runways.

In this paper, we consider the problem of scheduling aircraft (plane) landings at an airport. This problem is one of deciding a landing time on a runway for each plane in a given set of planes such that each plane lands within a predetermined time window, and that separation criteria between the landing of a plane, and the landing of all successive planes, are respected.

This paper is organized as follows. In Section 1, we set the problem in context. In Section 2, we present a mixed-integer zero–one formulation of the problem for the single runway case. We then discuss previous work on the problem in Section 3. In Section 4, we extend the formulation to the multiple runway case, and, in Section 5, we strengthen the linear programming (LP) relaxations of these formulations by introducing additional constraints. These formulations can be solved using LP-based tree search. An effective heuristic for the problem (for any number of runways) is presented in Section 6. Computational results for both the heuristic and the optimal algorithm for a number of test problems involving up to 50 planes and four runways are presented in Section 7.

It is important to note here that, although throughout this paper we shall typically refer to

Transportation Science, © 2000 INFORMS
Vol. 34, No. 2, May 2000

0041-1655/00/3402-0180 $05.00
pp. 180–197, 1526-5447 electronic ISSN

planes landing, the models presented in this paper are applicable to problems involving just takeoffs only and to problems involving a mix of landings and takeoffs on the same runway.

We should also stress here that we are dealing only with the static case. In other words, we are dealing with the off-line case where we have complete knowledge of the set of planes that are going to land. The dynamic, or on-line, case, where decisions about the landing times for planes must be made as time passes and the situation changes (planes land, new planes appear, etc.) is the subject of a separate paper (BEASLEY et al., 1995).

1. PROBLEM CONTEXT

1.1 Basic Problem

Upon entering within the radar range (radar horizon) of air traffic control (ATC) at an airport, a plane requires ATC to assign it a landing time, sometimes known as the broadcast time, and also, if more than one runway is in use, assign it a runway on which to land. The landing time must lie within a specified time window, bounded by an earliest time and a latest time, these times being different for different planes. The earliest time represents the earliest a plane can land if it flies at its maximum airspeed. The latest time represents the latest a plane can land if it flies at its most fuel-efficient airspeed while also holding (circling) for the maximum allowable time.

Each plane has a most economical, preferred speed, referred to as the cruise speed. A plane is said to be assigned its preferred time, or target time, if it is required to fly in to land at its cruise speed. If ATC requires the plane to either slow down, hold, or speed up, a cost will be incurred. This cost will grow as the difference between the assigned landing time and the target landing time grows.

The time between a particular plane landing, and the landing of any successive plane, must be greater than a specified minimum (the separation time) which is dependent upon the planes involved. Separation times are bounded below by aerodynamic considerations. A Boeing 747, for example, generates a great deal of air turbulence (wake vortices) and a plane flying too close behind could lose aerodynamic stability. Indeed a number of aircraft accidents are believed to have been caused by this phenomenon (MULLINS, 1996). For safety reasons, therefore, landing a Boeing 747 necessitates a (relatively) large time delay before other planes can land. A light plane, by contrast, generates little air turbulence and, therefore, landing such a plane necessitates only a (relatively) small time delay before other planes can land. Planes taking off impose similar restrictions on successive operations.

1.2 Practical Complexities

As we might expect, the practical problem of scheduling aircraft landings within an ATC environment is more complex than the basic problem described above. Below, we consider a number of these complexities: control, separation times, latest times, runway allocation, and objective function, and indicate whether the mixed-integer zero–one formulations presented in this paper can deal with these complexities.

Control

The mixed-integer zero–one formulations presented in this paper are decision models. That is, they relate to making a decision as to the landing time for each plane to respect separation times at landing and optimize an appropriate objective. They leave to one side the associated control problem, namely, can the planes be flown (controlled) in such a manner as to enable the solution of the decision model to be implemented and, if this is possible, how might the solution to this control problem impact the costs assumed when solving the decision model?

Plainly the decision model and control problem are linked. However, as in many operations research models, we believe that there are benefits to be gained in separating out these two aspects of the problem. For example, in the UK, it has been reported that National Air Traffic Services (who operate ATC for London Heathrow and Gatwick) are using a mixed-integer model to gain insight into runway capacity (SIMONS, 1997). Such a strategic study can, by assuming that the control issue is solvable, provide an upper bound on runway capacity.

We would also comment here that the common ATC (ODONI, ROUSSEAU, and WILSON, 1994; MILAN, 1997) practice of scheduling aircraft to land in a first-come/first-served (FCFS) manner also effectively applies a (trivial) decision model to the problem while leaving the control problem for later resolution. Note too that it is possible to incorporate some control restrictions (for example, relating to overtaking or to trajectory segments) in our decision models in a simple manner (see the discussion of Additional Restrictions in Section 2.4).

Separation Times

To set mandatory minimum separation times, the appropriate aviation authorities (e.g., the Federal Aviation Administration in the USA, the Civil Aviation Authority in the UK) classify planes into a small (e.g., three or four) number of classes and specify the separation that must apply between each class. Based upon this, a number of papers that have appeared in the literature have assumed that the separation time between planes relates just to these classes. However, in practice, at any particular airport, the situation can be much more complex. At London Heathrow, for example, separation times on takeoff relate not only to the class of plane but also to the standard instrument departure route (SID) that the plane is to follow immediately after takeoff. The models presented in this paper, by explicitly allowing separation times to be plane dependent, cater for such situations.

Latest Times

The latest landing time is set (as indicated above) based on fuel considerations. A number of papers that have appeared in the literature assume (somewhat unrealistically) that this latest time is sufficiently large as to be of no consequence. The models presented in this paper however, by explicitly including a finite latest landing time, are more realistic. We present some results in Section 7 to investigate the computational effect upon our algorithm of a large latest time, as well as to investigate the ability of our algorithm to detect that the problem is infeasible if the earliest and latest times are close to one another.

Runway Allocation

Runway allocation deals with assigning an appropriate runway to a plane (if there is more than one runway available). The models presented in this paper specifically address the problem of runway allocation. In particular, these models are applicable irrespective of whether the runways are being operated in segregated-mode (only takeoffs or landings) or mixed-mode (takeoffs and landings mixed). The mode adopted can be policy dependent. For example, London Heathrow has two runways that, as a matter of policy, usually operate in segregated mode. London Gatwick, by contrast, has a single runway that (obviously) operates in mixed-mode. One of the benefits of the models presented in this paper is that they allow an explicit quantitative comparison to be made between alternative runway operating policies within an optimization framework.

Objective Function

In this paper, we shall assume that we are minimizing total cost, where the cost for any plane is linearly related to deviation from its target time. Figure 1 illustrates the variation in cost within the time window of a particular plane. Note here that, in

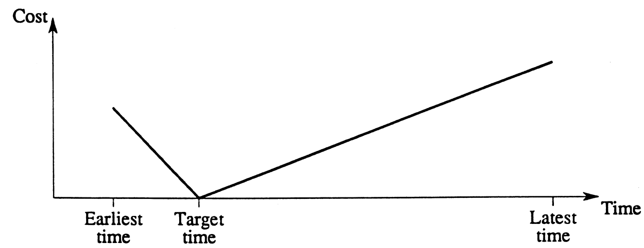

Fig. 1. Variation in cost for a plane within its time window.

Figure 1, we have a cost of zero for the plane landing at its target time. Any cost that the plane actually incurs in landing at its target time is (from the decision viewpoint) irrelevant because it is only the additional (marginal) cost over and above this that we can influence through an appropriate choice of landing time.

Note here that the objective function used is related to the issue of the viewpoint we adopt. If we are the airport operator, we may be interested in improving the long-term utilization of finite runway capacity by better scheduling. If we are an individual airline, we may be more interested in the individual plane costs incurred. In this paper, we implicitly assume, through the form of Figure 1, that we are interested in minimizing total plane costs.

There are a number of additional points to be made here:

1. Although the cost function shown in Figure 1 is nonlinear, the fact that it is composed of two linear portions enables us (see Sections 2.2 and 2.3) to linearize it and formulate the problem with a linear objective function.
2. The aircraft landing problem is a mixed-integer problem, hence (effectively) only by adopting a linearizable objective can such a model be solved numerically to optimality.
3. We indicate in Section 2.4 how any piecewise linear cost function that increases monotonically with respect to deviation from target time can be incorporated into our models.
4. We indicate in Section 5.4 how a linear formulation of the problem can be given that enables any (linear or nonlinear) cost function to be represented provided time can be discretized.
5. The cost function (Figure 1) we have adopted allows us to distinguish, in cost terms, both between landing before and after the target time and between different planes.
6. In some situations, a linear objective is perfectly reasonable (e.g., if using the models presented in this paper to land planes as fast as possible, see Section 2.4; or if using the multiple runway model

presented in this paper to balance workload between runways, see Section 4.3).

7. Although a linearizable cost function may not accurately describe cost, its use within an explicit optimizing mathematical approach may potentially lead to better decisions than heuristically solving the problem with a more accurate, but nonlinear, cost function.

Finally, we would note here that our experience has been that, in the field of aircraft scheduling, the issue of which objective function to adopt is the one that causes by far the most discussion. Arguments can convincingly be made for many different objective functions. We believe that different users will, for perfectly legitimate reasons, use different objective functions. The challenge for workers in this field is to develop solution approaches capable of dealing with a variety of such objectives. We believe that, in the light of the points made above, this paper represents a significant contribution toward developing such approaches.

2. SINGLE RUNWAY FORMULATION

IN THIS SECTION, we present an initial mixed-integer zero–one formulation of the static single runway aircraft landing problem.

2.1 Notation

Let

P = the number of planes
E_i = the earliest landing time for plane i ($i = 1, \ldots, P$)
L_i = the latest landing time for plane i ($i = 1, \ldots, P$)
T_i = the target (preferred) landing time for plane i ($i = 1, \ldots, P$)
S_{ij} = the required separation time ($\geqslant 0$) between plane i landing and plane j landing (where plane i lands before plane j), $i = 1, \ldots, P$; $j = 1, \ldots, P$; $i \neq j$
g_i = the penalty cost ($\geqslant 0$) per unit of time for landing before the target time T_i for plane i ($i = 1, \ldots, P$)
h_i = the penalty cost ($\geqslant 0$) per unit of time for landing after the target time T_i for plane i ($i = 1, \ldots, P$)

The time window for the landing of plane i is therefore $[E_i, L_i]$, where $E_i \leqslant T_i \leqslant L_i$. The variables are:

x_i = the landing time for plane i ($i = 1, \ldots, P$)
α_i = how soon plane i ($i = 1, \ldots, P$) lands before T_i
β_i = how soon plane i ($i = 1, \ldots, P$) lands after T_i

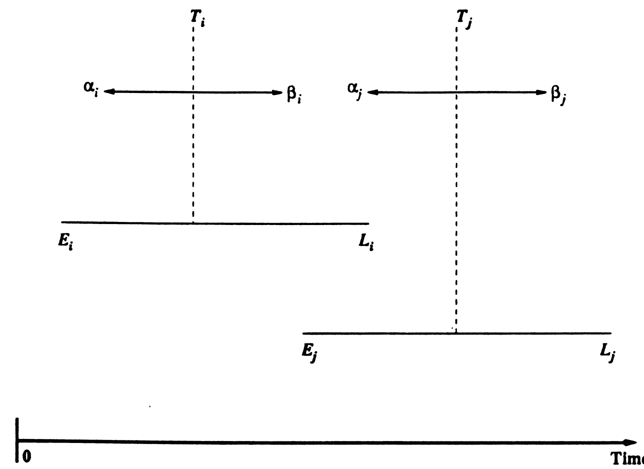

Fig. 2. An example of overlapping time windows.

$$\delta_{ij} = \begin{cases} 1 & \text{if plane } i \text{ lands before plane } j \ (i = 1, \ldots, P; \ j = 1, \ldots, P; \ i \neq j) \\ 0 & \text{otherwise} \end{cases}$$

Without significant loss of generality, we shall henceforth assume that the times E_i, L_i, and S_{ij} are integers.

2.2 Constraints

To clarify some of the constraints that will be given in this section, we provide a diagram (Figure 2), which depicts overlapping time windows for planes i and j. The first set of constraints are

$$E_i \leqslant x_i \leqslant L_i \quad i = 1, \ldots, P, \tag{1}$$

which ensure that each plane lands within its time window. Now, considering pairs (i, j) of planes we have that

$$\delta_{ij} + \delta_{ji} = 1 \quad i = 1, \ldots, P; \ j = 1, \ldots, P; \ j > i \tag{2}$$

In words, either plane i must land before plane j ($\delta_{ij} = 1$) or plane j must land before plane i ($\delta_{ji} = 1$).

It is trivial to see that, for certain pairs (i, j) of planes, we can immediately decide whether $\delta_{ij} = 1$ or whether $\delta_{ji} = 1$. For example, if the time windows for two planes i and j are [10, 50] and [70, 110], respectively, then it is clear that plane i must land first (i.e., that $\delta_{ij} = 1$). However, even if we know, for a pair of planes (i, j), the order in which they land, it does not mean that the separation constraint is automatically satisfied.

Continuing the example given above then, if the separation time $S_{ij} = 15$, the separation constraint is automatically satisfied, regardless of when within their respective time windows ([10, 50] and [70, 110]) planes i and j land. However, if $S_{ij} = 25$, the separation constraint is not automatically satisfied, i.e., there exist landing times for i and j (within their respective time windows) that violate the separation constraint. Hence we need to define three sets:

U = the set of pairs (i, j) of planes for which we are uncertain whether plane i lands before plane j;
V = the set of pairs (i, j) of planes for which i definitely lands before j (but for which the separation constraint is not automatically satisfied);
W = the set of pairs (i, j) of planes for which i definitely lands before j (and for which the separation constraint is automatically satisfied).

Then, we can define the set W by

$$W = [(i, j) | L_i < E_j \text{ and } L_i + S_{ij} \leqslant E_j \quad i = 1, \ldots, P; j = 1, \ldots, P; i \neq j]. \tag{3}$$

In words, i must land before j ($L_i < E_j$) and the separation constraint is automatically satisfied ($L_i + S_{ij} \leqslant E_j$).

We can define the set V by

$$V = [(i, j) | L_i < E_j \text{ and } L_i + S_{ij} > E_j \quad i = 1, \ldots, P; j = 1, \ldots, P; i \neq j]. \tag{4}$$

In words, i must land before j ($L_i < E_j$) but the separation constraint is not automatically satisfied ($L_i + S_{ij} > E_j$).

Some thought reveals that the pairs of planes (i, j) for which uncertainty exists with respect to which plane lands first must have overlapping time windows. Hence, we can define the set U as

$$U = [(i, j) | i = 1, \ldots, P; j = 1, \ldots, P; i \neq j; \quad E_j \leqslant E_i \leqslant L_j \text{ or } E_j \leqslant L_i \leqslant L_j \quad \text{or } E_i \leqslant E_j \leqslant L_i \text{ or } E_i \leqslant L_j \leqslant L_i]. \tag{5}$$

The definition of U means that one of the end points of the time window of one plane falls within the time window of the other.

We therefore have the constraint,

$$\delta_{ij} = 1 \quad \forall (i, j) \in \mathrm{W} \cup \mathrm{V}. \tag{6}$$

We need a separation constraint for pairs of planes in V, and this is

$$x_j \geqslant x_i + S_{ij} \quad \forall (i, j) \in V, \tag{7}$$

which ensures that a time S_{ij} must elapse after the landing of plane i at x_i before plane j can land at x_j.

We need a separation constraint for pairs of planes in U, and this is

$$x_j \geqslant x_i + S_{ij} - M\delta_{ji} \quad \forall (i, j) \in U, \tag{8}$$

where M is a large positive constant. There are two cases to consider here:

a. if $\delta_{ij} = 1$, then i lands before j and, from Eq. 2, we have that $\delta_{ji} = 0$. Therefore, Eq. 8 becomes

$$x_j \geqslant x_i + S_{ij}, \tag{9}$$

ensuring that separation is enforced.

b. if $\delta_{ij} = 0$, then j lands before i and, from Eq. 2, we have that $\delta_{ji} = 1$. Therefore, Eq. 8 becomes

$$x_j \geqslant x_i + S_{ij} - M, \tag{10}$$

i.e., $x_j \geqslant$ some large negative number, thereby ensuring that constraint is effectively inactive.

From the point of view of the LP relaxation of the problem, we would like M to be as small as possible, and this can be achieved by replacing M with $(L_i + S_{ij} - E_j)$ so that Eq. 8 becomes

$$x_j \geqslant x_i + S_{ij} - (L_i + S_{ij} - E_j)\delta_{ji} \quad \forall (i, j) \in U, \tag{11}$$

and, making use of Eq. 2, we can rewrite this as

$$x_j \geqslant x_i + S_{ij}\delta_{ij} - (L_i - E_j)\delta_{ji} \quad \forall (i, j) \in U. \tag{12}$$

To see that replacing M with $(L_i + S_{ij} - E_j)$ is valid, we need to recheck case (b) above. If $\delta_{ij} = 0$ ($\delta_{ji} = 1$), Eq. 12 becomes, after rearrangement,

$$x_j \geqslant E_j + (x_i - L_i). \tag{13}$$

Now we always have $(x_i - L_i) \leqslant 0$ (from Eq. 1), so Eq. 13 merely says $x_j \geqslant E_j +$ [some value $\leqslant 0$], a constraint that is always true.

Finally, we need constraints to relate the α_i, β_i, and x_i variables to each other. The variables α_i and β_i and the constraints presented below are necessary simply to ensure that we have a linear objective function and are:

$$\alpha_i \geqslant T_i - x_i \qquad i = 1, \ldots, P, \tag{14}$$

$$0 \leqslant \alpha_i \leqslant T_i - E_i \qquad i = 1, \ldots, P, \tag{15}$$

$$\beta_i \geqslant x_i - T_i \qquad i = 1, \ldots, P, \tag{16}$$

$$0 \leqslant \beta_i \leqslant L_i - T_i \qquad i = 1, \ldots, P, \tag{17}$$

$$x_i = T_i - \alpha_i + \beta_i \qquad i = 1, \ldots, P, \tag{18}$$

Equations 14 and 15 ensure that α_i is at least as big as zero and the time difference between T_i and x_i, and at most the time difference between T_i and E_i (see Figure 2). Equations 16 and 17 are similar equations for β_i. Equation 18 relates the landing time (x_i) to the time i lands before (α_i), or after (β_i), target (T_i).

There is one technical subtlety here, namely that Eqs. 14–18 are not sufficient to uniquely define α_i and β_i. To see that this is so, note that, if α_i and β_i

satisfy these equations, so too do $\alpha_i + K$ and $\beta_i + K$ for any constant K satisfying $0 < K \leqslant \min[(T_i - E_i) - \alpha_i, (L_i - T_i) - \beta_i]$. The uniqueness of α_i and β_i is only guaranteed by the fact that adding K to both of them increases the objective function (Eq. 19), provided that $g_i + h_i > 0$. Hence, in any minimum cost solution, at least one of Eqs. 14 and 16 will be satisfied with equality, thereby defining α_i and β_i appropriately.

2.3 Objective

We now need only to set up the objective function, minimize the cost of deviation from the target times (T_i), and this is

$$\text{minimize} \sum_{i=1}^{P} (g_i\alpha_i + h_i\beta_i). \tag{19}$$

The complete formulation (model) of the single runway problem is therefore to minimize function 19 subject to Eqs. 1, 2, 6, 7, 12, and 14–18.

This formulation is a mixed-integer zero–one program involving $3P$ continuous variables, at most $P(P - 1)$ binary (zero–one) variables and at most $[3P + 3P(P - 1)/2]$ constraints (excluding bounds on variables). However, as the computational results given in Section 7 show, the actual size of the problem (in terms of variables and constraints) can be much less than this.

2.4 Comment

There are three aspects of the model we should comment on here, namely: objective function, incorporating takeoffs, and additional restrictions. These are dealt with below.

Objective Function

Note first here that the objective of minimizing the average landing time, minimize $(\sum_{i=1}^{P} x_i)/P$, which is equivalent to minimize $\sum_{i=1}^{P} x_i$, is a special case of the objective we have adopted above with $T_i = E_i$ and $h_i = 1$. This is because, with these values and making use of Eqs. 15 and 18, we have that Eq. 19 becomes minimize $\sum_{i=1}^{P} (x_i - E_i)$, which is equivalent to minimize $\sum_{i=1}^{P} x_i$.

We have chosen to specify the objective (Eq. 19) as relating to deviation from target times. For example, if we were using the model in a tactical sense during the course of a single day's operations to decide landing times for planes, then this objective would seem reasonable. This is because it effectively enables each plane to express a view as to its preferred target landing time, which is explicitly considered in arriving at a solution (cf. the FCFS rule that enables no such consideration of preferences).

However, it is clear that alternative objectives are possible. For example, if we were using the model in a strategic sense to gain some measure of runway capacity, we might well adopt a minimax objective such as

$$\text{minimize} \max[x_i | i = 1, \ldots, P] \tag{20}$$

to land the last plane as soon as possible. Note here that this objective is easily linearizable by: adding a variable Z_{last} (representing the landing time of the last plane to land); adding constraints $Z_{\text{last}} \geqslant x_i$ $i = 1, \ldots, P$; and changing the objective to minimize Z_{last}. Hence, with this objective function, the solution approach adopted below is applicable. Note too that we might well, in practice, vary our objective over time, e.g., in busy periods move from a cost-based objective (Eq. 19) to a throughput-based objective (Eq. 20).

With respect to indicating how any piecewise linear cost function that increases monotonically with respect to deviation from target time can be incorporated into our model, suppose, for ease of illustration, that we just consider deviations after target time. If the time interval $[T_i, L_i]$ contains a single breakpoint b_i (where $T_i < b_i < L_i$) such that the incremental penalty cost per unit of time for landing after b_i is H_i ($\geqslant 0$), i.e., each unit of time landed after b_i costs $h_i + H_i$ in total, then this can be incorporated into the model by introducing variables $\theta_i \geqslant 0$ where

$$\theta_i \geqslant x_i - b_i \quad i = 1, \ldots, P \tag{21}$$

and adding $\sum_{i=1}^{P} H_i\theta_i$ to the objective function.

Note here that this is a standard approach for representing piecewise linear monotonically increasing functions and easily generalizes to more breakpoints (before or after target). Note too that this approach can be applied to any nonlinear cost function that can be approximated by a piecewise linear monotonic function.

Incorporating Takeoffs

We said before that the model presented in this paper is applicable to problems involving takeoffs only and to problems involving a mix of landings and takeoffs on the same runway. To see that this is so, reflect that all we have really done is build a model to decide times for planes to respect time window constraints and separation time constraints. This is precisely the situation we have both for takeoffs and for a mix of landings and takeoffs—just a set of planes with time windows and separation times, where we know, for any particular pair (i, j) of planes, whether i is landing or taking off, and whether j is landing or taking off, and so can set the separation time S_{ij} accordingly.

Additional Restrictions

In any particular situation, there may be additional restrictions upon what is possible. Such restrictions (depending upon their nature) can often be incorporated into the model we have presented above.

When considering takeoffs, taxiway restrictions may impose precedence relations on the order in which planes may take off. Such precedence relations are easily incorporated into the model through setting the δ_{ij} variables appropriately.

When considering landings, precedence relationships relating to the landing sequence may arise. These would typically be required to ensure that, in terms of implementing the solution from the decision model (the control issue, see Section 1.2), a) "fast" planes do not have to overtake "slow" planes as they come into land, and b) planes already in a standard trajectory segment (e.g., see ROBINSON, DAVIS, and ISAACSON, 1997) land in the order they are already in. Again, such precedence relations are easily incorporated into the model through setting the δ_{ij} variables appropriately.

With respect to restrictions on the number of landings within a given time period (for example because of gate availability), the situation is more complex, but such restrictions can be incorporated into the model. We will illustrate this for just one time period $[t_1, t_2]$ and suppose that we wish at most Q landings in this time period. Let G be the set of planes such that

$$G = [i|[t_1, t_2] \cap [E_i, L_i] \neq \emptyset \; i = 1, \ldots, P], \quad (22)$$

i.e., G is the set of planes that have a possible landing time within the time period $[t_1, t_2]$. Then (assuming for ease of illustration that all times are integer), any plane $i \in G$ must land in one of the three time segments $[E_i, \max(E_i, t_1) - 1]$, $[\max(E_i, t_1), \min(L_i, t_2)]$, and $[\min(L_i, t_2) + 1, L_i]$. Note here that some of these time segments may be non-existent, e.g., if $[t_1, t_2] = [5, 10]$ and $[E_i, L_i] = [2, 7]$ these segments are [2, 4], [5, 7], and [8, 7]. Hence let

$$\gamma_{ij} = \begin{cases} 1 & \text{if plane } i \text{ lands in time segment } j (j = 1, 2, 3) \\ 0 & \text{otherwise,} \end{cases}$$

where, obviously, γ_{ij} is set to zero if the segment is non-existent. Then we need to add to the model

$$\sum_{i \in G} \gamma_{i2} \leqslant Q, \quad (23)$$

$$\gamma_{i1} + \gamma_{i2} + \gamma_{i3} = 1 \quad \forall i \in G, \quad (24)$$

$$x_i \leqslant [\max(E_i, t_1) - 1]\gamma_{i1} + L_i(1 - \gamma_{i1}) \quad \forall i \in G, \quad (25)$$

$$\max(E_i, t_1)\gamma_{i2} \leqslant x_i \leqslant \min(L_i, t_2)\gamma_{i2} + L_i(1 - \gamma_{i2}) \quad \forall i \in G, \quad (26)$$

$$x_i \geqslant [\min(L_i, t_2) + 1]\gamma_{i3} \quad \forall i \in G. \quad (27)$$

Equation 23 restricts the number of landings in the specified time period $[t_1, t_2]$ and Eq. 24 ensures that exactly one of γ_{i1}, γ_{i2}, and γ_{i3} is one. Equations 25–27 relate the landing time x_i to the appropriate time segment. Note here that, because $\Sigma_{i \in G} \gamma_{i2}$ is the number of landings in the specified time period, we could, if we wish, add a cost term relating to this to our objective function. Note too that these constraints (Eqs. 23–27) could be applied to any subset of planes (e.g., we might wish to restrict the number of planes from a particular airline that land in a specified time period).

We would comment here that, in restricting landings for reasons such as gate availability, we are essentially forcing planes to "wait" in the air. Thought, therefore, needs to be given to the issue of whether it might not be better to land the planes and allow them to "wait" on the ground. Finally, note here that setting $Q = 0$ ensures that there are no landings in the specified time period $[t_1, t_2]$ and so the constraints given above also enable us to deal with what are known as blocked intervals and reserved time slots (ERZBERGER, 1995).

3. PREVIOUS WORK

IN THIS SECTION, we review the work that has been reported in the literature on the problem of scheduling aircraft landings. Readers interested in the wider problems that occur in the management of air traffic are referred to BIANCO and ODONI (1993), Odoni et al. (1994), WINTER and NÜSSER (1994).

It is clear that, with the problem of increasing congestion at airports, the efficient and effective scheduling of plane takeoffs and landings is an important one. However, we were surprised to find that, from the Operations Research viewpoint, relatively little has been written. Moreover, the work that has been done is widely scattered among internal reports, conference proceedings, books, and journals with the result that it is (in our experience) difficult to access.

ANDREUSSI, BIANCO, and RICCIARDELLI (1981) referred to the problem as the aircraft sequencing problem and presented a paper concerned with developing a discrete-event simulation model to evaluate different sequencing strategies. Computational results were presented for a number of simulated scenarios.

DEAR and SHERIF (1989, 1991) discussed both the static and dynamic aircraft landing problems and presented a heuristic algorithm for the (single runway) dynamic aircraft landing problem based upon a technique they refer to as *constrained position shifting*. This involves finding, for a small set of planes, the best possible positions for them in the landing queue subject to the constraint that no plane can be moved more than a pre-specified number of positions away from the position it had in the landing queue based on FCFS (see also DEAR, 1976). Computational results were presented for three simulated scenarios involving 500 planes. In Dear (1976) and Dear and Sherif (1989, 1991), the separation constraint only applies to successive plane landings. In other words, if i, j, and k land one after each other such that i lands before j, which lands before k, then both $x_j \geqslant x_i + S_{ij}$ and $x_k \geqslant x_j + S_{jk}$ are guaranteed to hold, but it may be that $x_k < x_i + S_{ik}$. We shall refer to this situation by saying that only *successive separation* is enforced. This contrasts with the problem as we have defined it where separation is enforced between all pairs of planes (which we shall refer to as *complete separation*). Note here that, if the triangle inequality $S_{ik} \leqslant S_{ij} + S_{jk}$ ($\forall i$, $\forall k \neq i$, $\forall j \neq i, k$) holds, then successive separation is sufficient to ensure complete separation.

PSARAFTIS (1980) incorporated constrained position shifting (see Dear, 1976; Dear and Sherif, 1989, 1991) within a dynamic programming recursion (with successive separation) and considered the single-runway static problem. He viewed the aircraft landing problem as comprising groups of identical planes waiting to land. This contrasts with the approach adopted in this paper where (potentially) all planes are different. Computational results were presented for a single problem involving three groups of planes (15 planes in total). The work presented in Psaraftis (1980) is based on PSARAFTIS (1978), which also considers the two-runway problem and (essentially) builds upon the single-runway solution approach (Psaraftis, 1980) by enumerating all possible partitions of the groups of planes between runways. Psaraftis presented computational results for six two-runway (static) problems involving up to 15 planes (three groups of planes in all cases).

BRINTON (1992) presented a depth-first tree search algorithm based on enumerating all possible aircraft sequences. Branches in the tree were discarded when the cost of a partially constructed sequence exceeded the best-known feasible solution. His approach can also be extended to include runway allocation. For computational reasons, he proposed applying his enumeration algorithm in a heuristic manner via a moving window approach (order the set of planes, sequence the first F (unfrozen) planes, freeze the landing of the first plane in this ordered set, and repeat).

ABELA et al. (1993) presented a mixed-integer zero–one formulation of the single-runway aircraft landing problem together with a heuristic based upon a genetic algorithm. Computational results were presented for a number of problems involving up to 20 planes. In this paper, we present a stronger formulation of the problem than the one presented in Abela et al. (1993). Furthermore, we also solve the multiple-runway case.

VENKATAKRISHNAN, BARNETT, and ODONI (1993) observed separation times adopted on landing at Logan Airport, Boston. Using these observed separation times, they applied the work of Psaraftis (1978, 1980), which they modified in a heuristic manner to take account of earliest/latest times, to see the improvement that could result from better sequencing. Milan (1997) considered the problem of assigning priorities to aircraft waiting to land from a queuing theory viewpoint.

A number of papers have appeared that view the aircraft landing problem as a job shop scheduling problem, as indeed did Psaraftis (1978, 1980). The runways represent identical machines and the planes represent jobs. The earliest time associated with each plane (job) is the *ready time* (sometimes called the *release time*) of the job. Typically such papers assume the latest time (which we consider explicitly in this paper) to be sufficiently large to be of no consequence.

The processing time of a particular job (plane) on a particular machine (runway) is then dependent upon just the job following it on the same machine (successive separation), or all the other jobs that will follow it on the same machine (complete separation). This is because the processing time, or, alternatively, the setup time, for a particular job on a particular machine must be sufficient to ensure that the following jobs (planes) are not started before the appropriate separation time has elapsed.

In other words, the problem of scheduling plane landings can be viewed as either a job shop scheduling problem with release times and sequence-dependent processing times (but zero setup times), or a job shop scheduling problem with release times and sequence-dependent setup times (but zero processing times). Typically, papers in the literature have taken as their objective function the minimization of the maximum landing time (Eq. 20), in job shop scheduling terms, the minimization of makespan, so that the aircraft landing problem is a $P/R/E_i$/seq-dep/$C_{\max}$ job shop scheduling problem, where R is the number of runways. Note here that, as a consequence of this, the aircraft landing problem is NP-hard.

BIANCO, NICOLETTI, and RICCIARDELLI (1978) adopted the job shop scheduling view (but with constant separation times) for the single-runway problem. A branch-and-bound algorithm was presented but no detailed computational results were given. BIANCO, RINALDI, and SASSANO (1987), BIANCO et al. (1988) and BIANCO and BIELLI (1993) adopted the job shop scheduling view (with successive separation) for the single-runway problem. They presented a mixed-integer zero–one formulation of the problem together with a tree search algorithm based upon a Lagrangean lower bound, a lower bound derived from scheduling theory and a heuristic procedure. Computational results were presented for a number of test problems involving up to 15 planes, and for three larger test problems. Of these three larger problems, one involved 20 planes, Bianco et al. (1987, 1988); the other two involved 30 and 44 planes, Bianco et al. (1987), Bianco and Bielli (1993).

For more on the job shop scheduling problem with sequence-dependent processing/setup times see LOCKETT and MUHLEMANN (1972), WHITE and WILSON (1977), SO (1990), GUINET (1991, 1993), SIMCHI-LEVI and BERMAN (1991), OVACIK and UZSOY (1993, 1994, 1995), JORDAN and DREXL (1995), LOW (1995), RUBIN and RAGATZ (1995), BRUCKER and THIELE (1996), FRANCA et al. (1996), LIAO and YU (1996), VAN DER VEEN and ZHANG (1996), HWANG and SUN (1997), LEE, BHASKARAN, and PINEDO (1997), LEE and PINEDO (1997), TAN and NARASIMHAN (1997).

Note here that, if successive separation applies, then the aircraft landing problem can be viewed as an open traveling salesman problem (TSP) with time windows, each city in the TSP being a plane. The difficulty with this approach lies in representing the objective function. BIANCO, MINGOZZI, and RICCIARDELLI (1993) adopted this approach and presented a dynamic programming algorithm for the TSP with cumulative costs. Computational results were presented for the solution of problems involving up to 35 cities (optimally) and 60 cities (heuristically). This problem is equivalent to the (single-runway) aircraft landing problem with no time windows and the objective of minimizing the average landing time ($\sum_{i=1}^{P} x_i/P$). BIANCO et al. (1999) also adopted this approach and presented a dynamic programming formulation, lower bounds, and two heuristic algorithms. Computational results were presented for a number of randomly generated problems and for two (single-runway) aircraft landing problems involving 30 and 44 planes.

The TSP problem with cumulative costs is sometimes also called the deliveryman problem. LUCENA (1990) presented an algorithm for this problem based upon Lagrangean relaxation and presented computational results for the (optimal) solution of problems involving up to 30 cities. FISCHETTI, LAPORTE, and MARTELLO (1993) presented an algorithm based upon dual ascent and gave computational results for the (optimal) solution of problems involving up to 60 cities. For earlier work on this problem, see PICARD and QUEYRANNE (1978), FOX et al. (1980).

As mentioned in Section 2.4, one use of the model we have developed is to gain some measure of runway capacity. Early work on the problem of runway capacity was given by BLUMSTEIN (1959). More recently, STEWART and SHORTREED (1993) have explored via simulation the trade-off among risk, separation, and capacity. For an overview of the work that has been done relating to runway capacity, see Odoni, Rousseau, and Wilson (1994).

There are a number of software systems available to help in airport aircraft management. These systems include, in part, modules to schedule landings and takeoffs:

- COMPAS (VÖLCKERS 1986, 1987, 1990; PLATZ and BROKOF, 1994), which takes an initial heuristic schedule and applies an enumeration procedure designed to eliminate separation-time conflicts
- MAESTRO (GARCIA, 1990), where the solution procedure is unclear (cf. Garcia, 1990; Venkatakrishnan, Barnett, and Odoni, 1993; Robinson, Davis, and Isaacson, 1997)
- OASIS (LJUNGBERG and LUCAS, 1992; LUCAS et al., 1994), which uses an A*-based (NILSSON, 1980) search procedure.

Probably the most extensive software system currently existing is CTAS (Center TRACON (terminal radar approach control) Automation System) developed at NASA Ames Research Center. CTAS contains two modules addressing the problem considered in this paper: a) a sequencer and scheduler module that sequences, using a simple constructive heuristic based on merging partial sequences, planes to land at a single runway, and schedules using the expression: scheduled landing time = max[scheduled landing time of previous plane + required separation, earliest possible landing time]; and b) a runway allocation module that allocates aircraft to runways in a heuristic fashion. See Erzberger (1995), LEE and DAVIS (1996), DAVIS et al. (1997), ISAACSON, DAVIS, and ROBINSON (1997), Robinson et al. (1997), and *http://www.ctas.arc.nasa.gov/project_description/fast.html.*

Finally, we would note here that there is currently much interest within Europe, and a number of ongoing projects, concerned with improving arrival and departure sequencing, which is the name by which the problem is commonly known in Europe (BEASLEY, 1996b).

4. MULTIPLE RUNWAY FORMULATION

IN THIS SECTION, we extend the formulation of the single-runway problem to the multiple-runway case.

4.1 Formulation

Most busy international airports have at least two runways and some have three or more. Thus, in situations where there is more than one runway to choose from, we need to determine the appropriate runway for planes to land on, as well as deciding a landing time for each plane.

If we have planes landing on different runways, we have the issue of the separation time required between such planes to consider. In the USA, for example (Venkatakrishnan, Barnett, and Odoni, 1993), planes can land simultaneously on parallel runways provided that their centerlines are more than F feet apart, where, under visual flight rules, $F = 1200$, but under instrument flight rules $F = 4300$. We shall henceforth assume that the runways are situated such that:

s_{ij} = the required separation time ($0 \leqslant s_{ij} \leqslant S_{ij}$) between plane i landing and plane j landing (where plane i lands before plane j and they land on different runways), $i = 1, \ldots, P$; $j = 1, \ldots, P$; $i \neq j$

Essentially, here we have assumed that the separation time between planes landing on different runways is runway independent. Let:

R = the number of runways

$$z_{ij} = \begin{cases} 1 & \text{if planes } i \text{ and } j \text{ land on the same runway } (i = 1, \ldots, P;\ j = 1, \ldots, P;\ i \neq j) \\ 0 & \text{otherwise} \end{cases}$$

$$y_{ir} = \begin{cases} 1 & \text{if plane } i\ (i = 1, \ldots, P) \text{ lands on runway } r (r = 1, \ldots, R) \\ 0 & \text{otherwise} \end{cases}$$

We then introduce the following constraints into the formulation of the problem given previously:

$$\sum_{r=1}^{R} y_{ir} = 1 \quad i = 1, \ldots, P \tag{28}$$

$$z_{ij} = z_{ji} \quad i = 1, \ldots, P;\ j = 1, \ldots, P;\ j > i \tag{29}$$

$$z_{ij} \geqslant y_{ir} + y_{jr} - 1 \quad i = 1, \ldots, P;\ j = 1, \ldots, P;\ j > i;\ r = 1, \ldots, R. \tag{30}$$

Equation 28 ensures that each plane lands on exactly one runway whereas Eq. 29 is a symmetry constraint (if i and j land on the same runway so do j and i). Equation 30 ensures that, if there is any runway r for which y_{ir} and y_{jr} are both one, then we force z_{ij} to be one (i and j land on the same runway). If $z_{ij} = 0$, then Eq. 30 ensures that planes i and j cannot land on the same runway.

The requirement that the separation time is S_{ij} for planes landing on the same runway but s_{ij} for planes landing on different runways can be easily dealt with. Redefine the sets W and V by replacing S_{ij} in Eqs. 3 and 4 by $\max(S_{ij}, s_{ij})$ and replace S_{ij} in the separation constraints (Eqs. 7 and 8) by $(S_{ij}z_{ij} + s_{ij}(1 - z_{ij}))$.

Hence, Eq. 7 becomes

$$x_j \geqslant x_i + S_{ij}z_{ij} + s_{ij}(1 - z_{ij}) \quad \forall (i, j) \in V \tag{31}$$

and Eq. 8 becomes

$$x_j \geqslant x_i + S_{ij}z_{ij} + s_{ij}(1 - z_{ij}) - M\delta_{ji} \quad \forall (i, j) \in U, \tag{32}$$

where M can be replaced by $(L_i + \max(S_{ij}, s_{ij}) - E_j)$ to give

$$x_j \geqslant x_i + S_{ij}z_{ij} + s_{ij}(1 - z_{ij}) - (L_i + \max(S_{ij}, s_{ij}) - E_j)\delta_{ji} \quad \forall (i, j) \in U. \tag{33}$$

4.2 Overview

The complete formulation (model) of the multiple runway problem is, therefore, to minimize function 19 subject to Eqs. 1, 2, 6, 14–18, 28–31, and 33. This formulation is a mixed-integer zero–one program involving $3P$ continuous variables, at most $[2P(P - 1) + PR]$ binary (zero–one) variables and at most $[4P + (4 + R)P(P - 1)/2]$ constraints (excluding bounds on variables). However, as the computational results given in Section 7 show, the actual size of the problem (in terms of variables and constraints) can be much less than this.

4.3 Workload

In some situations with multiple runways, it may be that the issue of runway workload needs to be considered (e.g., workload becomes important if different air traffic controllers deal with different runways). Let w_{ir} (≥ 0) represent a measure of the workload involved in landing plane i on runway r. Then, the amount of work involved for each runway r is given by $\sum_{i=1}^{P} w_{ir} y_{ir}$. Obviously, therefore, we could, if we wished, introduce explicit constraints on the workload allowed on each runway into our model.

Alternatively, it may be that our objective becomes one of balancing workload between runways. This can be easily formulated. Letting $Z_{\min}$ represent the workload on the most lightly loaded runway, and $Z_{\max}$ represent the workload on the most heavily loaded runway, we can formulate the problem of balancing workload by amending the objective of the multiple-runway model given above to be

$$\text{minimize } Z_{\max} - Z_{\min}, \tag{34}$$

and adding to the model the constraints

$$Z_{\max} \geq \sum_{i=1}^{P} w_{ir} y_{ir} \quad r = 1, \ldots, R \tag{35}$$

$$Z_{\min} \leq \sum_{i=1}^{P} w_{ir} y_{ir} \quad r = 1, \ldots, R. \tag{36}$$

Equation 35 ensures that $Z_{\max}$ is at least as big as the largest workload, and Eq. 36 ensures that $Z_{\min}$ is at least as small as the smallest workload. Minimizing $Z_{\max} - Z_{\min}$ ensures that we minimize the difference between these workloads, i.e., the runways are as balanced as possible.

5. STRENGTHENING THE RELAXED FORMULATIONS

ALTHOUGH THE formulations given above for both the single- and multiple-runway cases are sufficient to describe the problems, we intend solving them numerically through the use of LP-based tree search. This technique involves relaxing the zero–one variables δ_{ij}, y_{ir}, and z_{ij} to continuous (fractional) variables. If this is done, then there are a number of additional valid constraints that we can add to the problem, which are redundant in zero–one space; but which strengthen (improve) the value of the LP relaxation in continuous space. In this section, we discuss how the time windows for each plane can be tightened, present the additional constraints we have developed, and present an alternative formulation of the problem.

5.1 Time Window Tightening

Let Z_{UB} be any upper bound on the optimal solution to the problem. Then, it is possible to limit the deviation from target for each plane. Specifically, for plane i, if we assume that all other planes make a zero contribution to the objective function (Eq. 19) value, we can update E_i using

$$E_i = \max[E_i,\ T_i - Z_{\text{UB}}/g_i] \quad i = 1, \ldots, P, \tag{37}$$

because, if we land more than Z_{UB}/g_i time units before target, we would exceed the upper bound on the optimal solution. Similarly we have that

$$L_i = \min[L_i,\ T_i + Z_{\text{UB}}/h_i] \quad i = 1, \ldots, P, \tag{38}$$

because, if we land more than Z_{UB}/h_i time units after target, we would exceed the upper bound on the optimal solution.

Using Eqs. 37 and 38, the time window $[E_i, L_i]$ for each plane i ($i = 1, \ldots, P$) can be tightened in a preprocessing step. The benefit of tightening (closing) the time windows is that (potentially) the sets U and V can be reduced in size, thereby giving a smaller problem to solve.

Note here that, if we adopt the minimax objective (Eq. 20) discussed above, Eqs. 37 and 38 are not valid. Instead, any upper bound Z_{UB} limits the latest time at which any plane can land (see Eq. 20) and so, in this case, we can update L_i using

$$L_i = \min[L_i,\ Z_{\text{UB}}] \quad i = 1, \ldots, P. \tag{39}$$

5.2 Additional Constraints

Delta Setting

Consider

$$\delta_{ij} \geq (x_j - x_i)/(L_j - E_i) \quad \forall (i, j) \in U. \tag{40}$$

If $(x_j - x_i) > 0$, then plane j lands after plane i (i lands before j) and, hence, we need to enforce $\delta_{ij} = 1$. In the above constraint, the $(L_j - E_i)$ term is a scaling factor and represents the maximum value $(x_j - x_i)$ can take. This constraint is redundant if $R = 1$ (because, in that case, it is dominated by Eq. 12), but not if $R \geq 2$.

Delta Sum

Consider the δ_{ij} variables. The first plane to land lands before $(P - 1)$ other planes, the second plane to land lands before $(P - 2)$ other planes, etc. Thus, we must have that the δ_{ij} variables sum to $(P -$

1) + $(P-2) + \cdots + 1$, i.e., to $P(P-1)/2$. Hence, we have that

$$\sum_{i=1}^{P} \sum_{j=1, j\neq i}^{P} \delta_{ij} = P(P-1)/2. \tag{41}$$

Gap Closing

Suppose we have two planes i and j with $T_i < T_j$ (i.e., i would prefer to land before j). Then the gap $(T_j - T_i)$ between the target times of i and j can only be closed by a combined movement of $(\beta_i + \alpha_j)$ (see Figure 2). Hence, we have that

$$\delta_{ij} \geqslant 1 - \frac{\beta_i + \alpha_j}{T_j - T_i} \quad \forall (i, j) \in U \quad \text{with} \quad T_i < T_j, \tag{42}$$

because, if $(\beta_i + \alpha_j)$ is insufficient to close the gap, we must have that i lands before j, i.e., that $\delta_{ij} = 1$.

Minimum Deviation

Suppose we have two planes i and j with $T_i < T_j$ and $(T_j - T_i) < S_{ij}$, i.e., i would prefer to land before j but their target times do not satisfy separation. So, if both planes land on the same runway, some movement from target (for one or both planes) must occur.

Assume for the moment that $z_{ij} = 1$, so both planes do land on the same runway. Referring to Figure 2, if $\delta_{ij} = 1$ then i lands before j and this movement must be at least $[S_{ij} - (T_j - T_i)]$. However, if j lands before i ($\delta_{ji} = 1$) then this movement must be at least $[(T_j - T_i) + S_{ji}]$. Hence, we have that

$$\begin{aligned}(\alpha_i + \beta_i) &+ (\alpha_j + \beta_j) \\ &\geqslant [S_{ij} - (T_j - T_i)]\delta_{ij} + [(T_j - T_i) + S_{ji}]\delta_{ji} \\ &\quad - \max\{[S_{ij} - (T_j - T_i)], [(T_j - T_i) + S_{ji}]\}(1 - z_{ij}) \\ &\forall (i, j) \in U \quad \text{with } T_i < T_j \text{ and } (T_j - T_i) < S_{ij}\end{aligned} \tag{43}$$

where the term involving $(1 - z_{ij})$ renders the constraint inactive if $z_{ij} = 0$. Obviously, for the single-runway situation, this term can be ignored (because, in that case, i and j must land on the same runway). A similar constraint can also be derived for planes i and j with $T_i < T_j$ and $(T_j - T_i) < s_{ij}$.

Order Deciding

Consider the set U. It may contain two overlapping time windows from which we can deduce, using the separation times, which plane lands first (assuming they land on the same runway). For example, if the time window for plane i is [10, 50] and the time window for plane j is [40, 70] then $(i, j) \in U$. However, if $S_{ji} = 15$, we can deduce that i can never land after j (i.e., that $\delta_{ji} = 0$) assuming i and j land on the same runway (i.e., assuming that $z_{ij} = 1$). Hence, define

$$U^* = [(i, j) | (i, j) \in U \text{ and } E_j + S_{ji} > L_i], \tag{44}$$

i.e., the set U^* contains pairs (i, j) of planes for which it is impossible for j to land before i (assuming they land on the same runway). Hence, we have

$$\delta_{ji} + z_{ij} \leqslant 1 \quad \forall (i, j) \in U^*. \tag{45}$$

Obviously, for the single-runway situation, the z_{ij} term is one (because, in that case, i and j must land on the same runway).

We can derive a similar constraint for planes that land on different runways. Define

$$U^{**} = [(i, j) | (i, j) \in U \text{ and } E_j + s_{ji} > L_i], \tag{46}$$

then we have

$$\delta_{ji} + (1 - z_{ij}) \leqslant 1 \quad \forall (i, j) \in U^{**}. \tag{47}$$

Number on Runway

The multiple runway formulation of the problem can be strengthened if we develop a lower bound on the number of pairs of planes that can land on the same runway. In other words, a constraint of the form

$$\sum_{i=1}^{P} \sum_{j=1, j\neq i}^{P} z_{ij} \geqslant K, \tag{48}$$

where K is derived by examining the different ways of landing P planes on R runways. Suppose m_1 planes land on runway 1, m_2 on runway 2, ..., m_R on runway R. Then,

$$\begin{aligned}\sum_{i=1}^{P} \sum_{j=1, j\neq i}^{P} z_{ij} &= m_1(m_1 - 1) + m_2(m_2 - 1) + \cdots \\ &\quad + m_R(m_R - 1) \qquad (49) \\ &= \sum_{r=1}^{R} (m_r)^2 - \sum_{r=1}^{R} m_r = \sum_{r=1}^{R} (m_r)^2 - P.\end{aligned}$$

Hence, an appropriate value for K is the optimal solution value of

$$\text{minimize} \sum_{r=1}^{R} (m_r)^2 - P \tag{50}$$

$$\text{subject to} \sum_{r=1}^{R} m_r = P \tag{51}$$

$$m_1, m_2, \ldots, m_R \text{ integers } (\geqslant 0). \tag{52}$$

This is an integer quadratic program that can be solved (in the worst case) by total enumeration in $O(RP^{R-1})$ operations (possible because R is small in practice). For example, if $P = 50$ and $R = 2$, the optimal solution to the above problem has $m_1 = m_2 = 25$ so that $K = 1200$.

5.3 Complete Formulation

The complete (strengthened) single/multiple-runway formulation (model) is therefore to minimize function 19 subject to Eqs. 1, 2, 6, 14–18, 28–31, 33, 40–43, 45, 47, and 48. Note here that we did develop a number of other additional constraints that can be used to strengthen the LP relaxations of the original formulations of the problem. However, we have only presented those additional constraints that we found to be of computational benefit.

5.4 An Alternative Formulation

It is clear that the formulation of the problem given above is relatively complex. We did experiment with a simpler alternative formulation that can be easily derived if we discretize time (require all times, and in particular the landing time, to be integer). Specifically, define:

C_{itr} = the cost of plane i landing at time t on runway r

$$X_{itr} = \begin{cases} 1 & \text{if plane } i \text{ lands at time } t \text{ on runway } r \\ 0 & \text{otherwise,} \end{cases}$$

then the formulation is

$$\text{minimize} \sum_{i=1}^{P} \sum_{t=E_i}^{L_i} \sum_{r=1}^{R} C_{itr} X_{itr} \tag{53}$$

subject to (54)

$$\sum_{t=E_i}^{L_i} \sum_{r=1}^{R} X_{itr} = 1 \quad i = 1, \ldots, P$$

$$X_{itr} + X_{j\tau r} \leqslant 1$$

$$i = 1, \ldots, P;\ j = 1, \ldots, P;$$
$$i \neq j;\ r = 1, \ldots, R;$$
$$\forall t \in [E_i, L_i]; \quad \forall \tau \in [E_j, L_j] \cap [t, t + S_{ij} - 1] \tag{55}$$

$$X_{itr} + X_{j\tau u} \leqslant 1$$

$$i = 1, \ldots, P; \quad j = 1, \ldots, P;$$
$$i \neq j;\ r = 1, \ldots, R; \quad u = 1, \ldots, R; \quad u \neq r;$$
$$\forall t \in [E_i, L_i];\ \forall \tau \in [E_j, L_j] \cap [t, t + s_{ij} - 1] \tag{56}$$

$$X_{itr} \in (0, 1)$$

$$i = 1, \ldots, P;\ \forall t \in [E_i, L_i];$$
$$r = 1, \ldots, R. \tag{57}$$

Equation 54 ensures that each plane lands exactly once (at some time on some runway). Equation 55 ensures that, if plane i lands at time t on runway r, no other plane j (landing in $[E_j, L_j]$) can land on runway r at a time τ, which lies within the time period $[t, t + S_{ij} - 1]$, i.e., this is the separation constraint for planes that land on the same runway. Equation 56 is the separation constraint for planes that land on different runways.

This formulation of the problem can be linked to the previous formulations using the constraints

$$x_i = \sum_{t=E_i}^{L_i} \sum_{r=1}^{R} tX_{itr} \quad i = 1, \ldots, P \tag{58}$$

$$y_{ir} = \sum_{t=E_i}^{L_i} X_{itr} \quad i = 1, \ldots, P;\ r = 1, \ldots, R. \tag{59}$$

One advantage of this discrete-time formulation of the problem is that C_{itr} can be quite general, so we are not restricted to a linear function for deviations from target (unlike the continuous-time formulations presented above). One disadvantage of this discrete time formulation is a relatively large number of variables and constraints (in particular, Eqs. 55 and 56). Moreover, limited computational experience with the LP relaxation of this formulation was disappointing, so it was not explored further.

6. HEURISTIC UPPER BOUND AND RESTARTING

6.1 Upper Bound

If we can find an upper bound Z_{UB} on the optimal solution to the problem, then we can tighten the time windows for each plane as indicated above (Eqs. 37 and 38). In addition, such an upper bound can be used to curtail the LP-based tree search. We used the following heuristic to find an upper bound (a feasible solution) for the problem:

1. Let A_r ($r = 1, \ldots, R$) be the ordered set of planes landing on runway r, where initially $A_r = \varnothing\ \forall r$.

2. Consider the planes in ascending target (T_j) order (ties broken arbitrarily) and, for each such plane j in turn:
   a. compute for each runway r the value of

$$B_r = \max[T_j, \max[x_k + S_{kj} | \forall k \in A_r], \max[x_k + s_{kj} | u = 1, \ldots, R\ u \neq r\ \ \forall k \in A_u]] \quad (60)$$

   this expression gives the best (least cost) time B_r that j can land on runway r, given the earlier landings on that runway and the earlier landings on all other runways
   b. let γ be the runway corresponding to $B_\gamma = \min[B_r \mid r = 1, \ldots, R]$ (ties broken arbitrarily). Add j to A_γ and set $x_j = B_\gamma$.

   In words, what we are doing here is adding each plane in turn to a particular runway (at a particular landing time) depending upon the best (least cost) possible landing time for it, given the other planes landing on the runways.
3. One feature of the above (Steps 1 and 2) is that we never land before target, always at or after target. To improve the solution obtained, we, in this step, recalculate the landing times, but with the runways for each plane and the landing order fixed as decided at Steps 1 and 2. This can be simply done because we have effectively made a heuristic choice of all the integer variables in our formulation of the problem, and we need only now solve the resulting LP to decide an optimal set of landing times (optimal with respect to the given heuristic integer choices).

6.2 Restarting

Although the heuristic presented above gave good results, computational experience indicated that the LP-based tree search occasionally found an improved feasible solution (upper bound) early in the tree search. Because the advantage of an improved feasible solution is that we can tighten the time windows for each plane, thereby tightening the formulation, it seemed appropriate to restart the problem each time an improved feasible solution was found (provided it was found early in the tree search).

In the computational results presented below, we restart the problem if we find an improved feasible solution less than P (the number of planes) seconds after starting the tree search. Hence our solution approach is:

1. apply the heuristic to generate an upper bound Z_{UB}
2. use Z_{UB} to tighten the time windows
3. use tree search to resolve the LP relaxation of the problem:
   a. if an improved feasible solution is found within P seconds of starting the tree search, then terminate the tree search, update Z_{UB} using this improved feasible solution and restart the problem (i.e., go to Step 2);
   b. otherwise, continue the tree search until normal termination (when an integer solution will have been found and proved to be optimal).

Note here that this solution approach means that we may restart the problem a number of times, each time with an improved feasible solution.

7. COMPUTATIONAL RESULTS

THE ALGORITHM PRESENTED in this paper was programmed in FORTRAN and run on a DEC 3000/700 (200 Mhz Alpha chip) for a number of test problems involving up to 50 planes. To solve the mixed-integer zero–one formulations of the problem to optimality using LP-based tree search, we used the CPLEX software package (CPLEX, 1994). All of the test problems solved in this paper are publicly available from OR-Library (BEASLEY, 1990, 1996a), E-mail the message *airlandinfo* to *o.rlibrary@ic.ac.uk* or see *http://mscmga.ms.ic.ac.uk/jeb/orlib/airlandinfo.html*.

Each test problem was solved with an increasing number of runways until the optimal solution value dropped to zero (indicating that we had sufficient runways to enable all planes to land on target). Note here that, for the multiple-runway case, we assumed that (s_{ij}) was zero.

Table I shows the computational results. In that table, we give, for each problem, the number of planes; the number of runways; the heuristic solution value and the time taken; the number of restarts, the solution value at each restart, and the time taken up to each restart; CPLEX statistics (number of variables, number of constraints, and the LP relaxation value) at the final restart (i.e., at the initial tree node before branching); and the optimal solution value, the total number of tree nodes and the total time taken. All computer times are in DEC 3000/700 seconds. Note here that, in Table I, if the heuristic gives a solution value of zero, we automatically know that this is the optimal solution (i.e., all planes land on target).

It is clear from Table I that the heuristic presented in this paper is able to find good quality solutions quickly. In addition, good (if not optimal) feasible solutions are found early in the tree search for a number of the problems for which an optimal solution has not been found by the heuristic.

TABLE I
COMPUTATIONAL RESULTS

			Heuristic		Tree Search								
Problem Number	Number of Planes	Number of Runways	Value	Time (secs)	Number of Restarts	Solution Value at Each Restart	Time at Each Restart (secs)	Number of Variables at Final Restart	Number of Constraints at Final Restart	LP Value at Final Restart	Optimal Value	Number of Tree Nodes	Total Time (secs)
1	10	1	optimal	0.1	—	—	—	68	173	321.16	700	49	0.4
		2	optimal	0.1	—	—	—	90	158	0	90	91	0.6
		3	optimal	0.1	—	—	—	—	—	—	0	—	—
2	15	1	1500	0.1	1	1480	1.9	157	412	430.00	1480	454	5.2
		2	optimal	0.1	—	—	—	196	386	0	210	115	1.8
		3	optimal	0.1	—	—	—	—	—	—	0	—	—
3	20	1	1380	0.1	1	820	0.6	194	525	449.40	820	42	2.7
		2	optimal	0.1	—	—	—	277	489	0	60	142	3.8
		3	optimal	0.2	—	—	—	—	—	—	0	—	—
4	20	1	optimal	0.1	—	—	—	336	929	924.37	2520	20002	220.4
		2	optimal	0.1	—	—	—	389	870	0	640	193319	1919.9
		3	optimal	0.1	—	—	—	345	837	0	130	39901	2299.2
		4	optimal	0.2	—	—	—	—	—	—	0	—	—
5	20	1	5420	0.1	—	—	—	426	1181	964.83	3100	50745	922.0
		2	1070	0.1	—	—	—	452	1076	0	650	282160	11510.4
		3	240	0.1	—	—	—	361	890	0	170	20035	1655.3
		4	optimal	0.2	—	—	—	—	—	—	0	—	—
6	30	1	optimal	0.1	—	—	—	80	222	5393.25	24442	10806	33.1
		2	882	0.1	—	—	—	630	1266	0	554	25316	1568.1
		3	optimal	0.2	—	—	—	—	—	—	0	—	—
7	44	1	optimal	0.1	—	—	—	174	341	184.00	1550	2192	10.6
		2	optimal	0.2	—	—	—	—	—	—	0	—	—
8	50	1	2690	0.4	7	2480; 2285; 2245; 2080; 2040; 1990; 1950	5.4; 11.3; 16.8; 24.3; 29.9; 41.4; 51.2	1318	3785	1547.86	1950	1114	111.9
		2	255	0.2	—	—	—	1562	3154	0	135	9020	3450.6
		3	optimal	0.6	—	—	—	—	—	—	0	—	—

To examine the computational effect of the latest landing time upon the solution procedure, we resolved the largest problem shown in Table I (Problem 8, with $R = 1$ and 2). To make a more valid comparison, we set the upper bound Z_{UB} equal to the optimal solution value shown in Table I and solved using both the original latest landing time and with the latest landing time increased by a factor of 100 (i.e., $L_i = 100L_i$). The result was that increasing the latest landing time by a factor of 100 increased the solution time by only 1.4%. That this effect is so small is principally due to the explicit setting of L_i via Eq. 38.

To examine the issue of infeasibility, we resolved the largest problem shown in Table I (Problem 8) with the earliest landing time as before, but with the latest landing time L_i now equal to $E_i + 2$ and the target landing time T_i now equal to $E_i + 1$. The result was that the algorithm proved that, for $R = 1$ and for $R = 2$ the problems were infeasible in 3.7 and 6.7 seconds, respectively. For $R = 3$, a feasible solution was found in 0.2 seconds.

One feature of Table I on which we should comment is the LP value at the final restart. It will be seen that all the LP values are zero in the case of multiple runways ($R \geqslant 2$). To see why this is so, consider Eq. 28 with $R = 2$. When the integrality requirement on the y_{ir} is relaxed, each plane i can be assigned values $y_{i1} = y_{i2} = 1/2$. This leads, in Eq. 30, to $z_{ij} \geqslant 0$, satisfied by $z_{ij} = 0$ and effectively saying (Eqs. 31 and 32) that all separation times are zero (recall here that $s_{ij} = 0$ for our test problems).

Because producing an LP lower bound of zero for the multiple-runway case appears, at first sight, to be a trivial achievement, the question therefore arises as to the value of the formulation given in this paper for the multiple-runway case. We believe that the multiple-runway formulation we have presented has merit, not because it produces a good lower bound at the initial tree node, but because using it in an LP-based tree search does enable us to solve, to optimality, multiple runway problems of moderate size. As far as we are aware this is the first time such problems have been so solved in the literature.

8. CONCLUSIONS

IN THIS PAPER, we have presented formulations of the static aircraft landing problem, for both the single- and multiple-runway cases, as mixed-integer zero–one programs. The LP relaxation of these formulations was strengthened by the introduction of additional constraints. Computational results were presented for a number of test problems involving up to 50 planes and four runways. Throughout this paper, we have tried to indicate how our formulations can be used to model a number of issues commonly encountered in practice, such as choice of objective function, precedence restrictions, restricting the number of landings in a given time period, runway workload balancing. Finally we would comment that we believe the problem of effectively scheduling aircraft takeoffs and landings is an important problem that has (for various reasons) not achieved the prominence it deserves in the Operations Research literature. We hope that this paper will help to highlight the problem and encourage others to work on it.

ACKNOWLEDGMENTS

J.E.B. WOULD LIKE TO acknowledge the financial support of the Commonwealth Scientific and Industrial Research Organization, Australia.

REFERENCES

J. ABELA, D. ABRAMSON, M. KRISHNAMOORTHY, A. DE SILVA, AND G. MILLS, "Computing Optimal Schedules for Landing Aircraft," in *Proceedings of the 12th National ASOR Conference,* Adelaide, Australia, 71–90 (1993).

A. ANDREUSSI, L. BIANCO, AND S. RICCIARDELLI, "A Simulation Model for Aircraft Sequencing in the Near Terminal Area," *Eur. J. Oper. Res.* **8,** 345–354 (1981).

J. E. BEASLEY, "OR-Library: Distributing Test Problems by Electronic Mail," *J. Oper. Res. Soc.* **41,** 1069–1072 (1990).

J. E. BEASLEY, "Obtaining Test Problems via Internet," *J. Global Optim.* **8,** 429–433 (1996a).

J. E. BEASLEY, Private communication, based on attendance at the *APATSI Workshop on Combined Arrival and Departure Management* held at Eurocontrol, Brussels (1996b).

J. E. BEASLEY, M. KRISHNAMOORTHY, Y. M. SHARAIHA, AND D. ABRAMSON, "Dynamically Scheduling Aircraft Landings—the Displacement Problem," Working Paper, available from the first author at The Management School, Imperial College, London, England, 1995.

L. BIANCO AND M. BIELLI, "System Aspects and Optimization Models in ATC Planning," in *Large Scale Computation and Information Processing in Air Traffic Control,* L. Bianco and A. R. Odoni (eds.), 47–99, Springer-Verlag, Berlin (1993).

L. BIANCO, P. DELL'OLMO, AND S. GIORDANI, "Minimizing Total Completion Time Subject to Release Dates and Sequence Dependent Processing Times," In *Advances in Combinatorial Optimization,* Y. M. Sharaiha and J. E. Beasley (eds.), Annals of Operations Research **86,** 393–415 (1999).

L. BIANCO, A. MINGOZZI, AND S. RICCIARDELLI, "The Travelling Salesman Problem with Cumulative Costs," *Networks,* **23,** 81–91 (1993).

L. BIANCO, B. NICOLETTI, AND S. RICCIARDELLI, "An Algorithm for Optimal Sequencing of Aircraft in the Near Terminal Area," in *Optimization Techniques: Proceedings of the 8th IFIP Conference on Optimization Techniques Würzburg, September 5–9, 1977,* Part 2, J. Stoer (ed.), Lecture Notes in Control and Information Sciences **7,** 443–453, Springer-Verlag, Berlin, (1978).

L. BIANCO AND A. R. ODONI (EDS.), "*Large Scale Computation and Information Processing in Air Traffic Control,*" Springer-Verlag, Berlin, (1993).

L. BIANCO, S. RICCIARDELLI, G. RINALDI, AND A. SASSANO, "Scheduling Tasks with Sequence-Dependent Processing Times," *Naval Res. Logist.* **35,** 177–184 (1988).

L. BIANCO, G. RINALDI, AND A. SASSANO, "A Combinatorial Optimization Approach to Aircraft Sequencing Problem," in *Flow Control of Congested Networks,* A. R. Odoni, L. Bianco, and G. Szegö (eds.), NATO ASI Series, Series F: Computer and Systems Science **38,** 323–339, Springer-Verlag, Berlin, (1987).

A. BLUMSTEIN, "The Landing Capacity of a Runway," *Opns. Res.* **7,** 752–763 (1959).

C. R. BRINTON, "An Implicit Enumeration Algorithm for Arrival Aircraft Scheduling," in *Proceedings of the 11th IEEE/AIAA Digital Avionics Systems Conference,* 268–274, IEEE, Piscatatway, NJ, (1992).

P. BRUCKER AND O. THIELE, "A Branch & Bound Method for the General-Shop Problem with Sequence Dependent Setup-times," *OR Spektrum* **18,** 145–161 (1996).

CPLEX OPTIMIZATION INC., "Using the CPLEX Callable Library (version 3.0)," CPLEX Optimization Inc., Incline Valley, NV, 1994.

T. J. DAVIS, D. R. ISAACSON, J. E. ROBINSON, III, W. DEN BRAVEN, K. K. LEE AND B. SANFORD, "Operational Test Results of the Passive Final Approach Spacing Tool," Presented at the *IFAC 8th Symposium on Transportation Systems 97,* Chania, Greece (1997). Available from *http://www.ctas.arc.nasa.gov/publications/*

R. G. DEAR, "The Dynamic Scheduling of Aircraft in the Near Terminal Area," Report R76-9, Flight Transportation Laboratory, MIT, Cambridge, MA, 1976.

R. G. DEAR AND Y. S. SHERIF, "The Dynamic Scheduling of Aircraft in High Density Terminal Areas," *Microelect. Reliab.* **29,** 743–749 (1989).

R. G. DEAR AND Y. S. SHERIF, "An Algorithm for Computer Assisted Sequencing and Scheduling of Terminal Area Operations," *Transp. Res. Part A, Policy and Practice* **25,** 129–139 (1991).

H. ERZBERGER, "Design Principles and Algorithms for Automated Air Traffic Management," in *Knowledge-based*

Functions in Aerospace Systems, AGARD Lecture Series no. 200, 7:1–7:31, NATO (1995).

M. FISCHETTI, G. LAPORTE, AND S. MARTELLO, "The Delivery Man Problem and Cumulative Matroids," *Opns. Res.* **41,** 1055–1064 (1993).

K. R. FOX, B. GAVISH, AND S. C. GRAVES, "A n-Constraint Formulation of the (Time-Dependent) Traveling Salesman Problem," *Opns. Res.* **28,** 1018–1021 (1980).

P. M. FRANCA, M. GENDREAU, G. LAPORTE, AND F. M. MÜLLER, "A Tabu Search Heuristic for the Multiprocessor Scheduling Problem with Sequence Dependent Setup Times," *Int. J. Product. Econ.* **43,** 79–89 (1996).

J.-L. GARCIA, "MAESTRO—A Metering and Spacing Tool," *Proceedings of the 1990 American Control Conference,* 502–507, IEEE, Piscataway, NJ, (1990).

A. GUINET, "Textile Production Systems: A Succession of Non-Identical Parallel Processor Shops," *J. Oper. Res. Soc.* **42,** 655–671 (1991).

A. GUINET, "Scheduling Sequence-Dependent Jobs on Identical Parallel Machines to Minimize Completion Time Criteria," *Int. J. Product. Res.* **31,** 1579–1594 (1993).

H. HWANG AND J. U. SUN, "Production Sequencing Problem with Reentrant Work Flows and Sequence Dependent Setup Times," *Comput. Industr. Eng.* **33,** 773–776 (1997).

D. R. ISAACSON, T. J. DAVIS, AND J. E. ROBINSON, III, "Knowledge-Based Runway Assignment for Arrival Aircraft in the Terminal Area," Presented at the *AIAA Guidance, Navigation and Control Conference,* New Orleans, LA (1997). Available from *http://www.ctas.arc.nasa.gov/publications/*

C. JORDAN AND A. DREXL, "A Comparison of Constraint and Mixed-Integer Programming Solvers for Batch Sequencing with Sequence-Dependent Setups," *ORSA J. Comput.* **7,** 160–165 (1995).

Y. H. LEE, K. BHASKARAN, AND M. PINEDO, "A Heuristic to Minimize the Total Weighted Tardiness with Sequence-Dependent Setups," *IIE Trans.* **29,** 45–52 (1997).

K. K. LEE AND T. J. DAVIS, "The Development of the Final Approach Spacing Tool (FAST): A Cooperative Controller-Engineer Design Approach," *Control Eng. Pract.* **4,** 1161–1168 (1996).

Y. H. LEE AND M. PINEDO, "Scheduling Jobs on Parallel Machines with Sequence-Dependent Setup-Times," *Eur. J. Oper. Res.* **100,** 464–474 (1997).

C. J. LIAO AND W. C. YU, "Sequencing Heuristics for Dependent Setups in a Continuous Process Industry," *Omega,* **24,** 649–659 (1996).

M. LJUNGBERG AND A. LUCAS, "The OASIS Air Traffic Management System," in *Proceedings of the 2nd Pacific Rim International Conference on Artificial Intelligence,* Seoul, Korea, 1003–1009 (1992).

A. G. LOCKETT AND A. P. MUHLEMANN, "A Scheduling Problem Involving Sequence Dependent Changeover Times," *Opns. Res.* **20,** 895–902 (1972).

C. Y. LOW, "Job Shop Scheduling Heuristics for Sequence Dependent Setups," *Comput. Industr. Eng.* **29,** 279–283 (1995).

A. LUCAS, M. LJUNGBERG, R. EVERTSZ, G. TIDHAR, R. GOLDIE, AND P. MAISANO, "New Techniques for Air Traffic Management for Single and Multiple Airports," Technical note 51, Australian Artificial Intelligence Institute, Melbourne, Australia, 1994.

A. LUCENA, "Time-Dependent Traveling Salesman Problem—The Deliveryman Case," *Networks,* **20,** 753–763 (1990).

J. MILAN, "The Flow Management Problem in Air Traffic Control: A Model of Assigning Priorities for Landings at a Congested Airport," *Transp. Plan. Technol.* **20,** 131–162 (1997).

J. MULLINS, "Trails of Destruction," *New Scientist,* **2056,** 28–31 (1996).

N. J. NILSSON, *Principles of Artificial Intelligence,* Tioga Publishing Company, Palo Alto, CA, 1980.

A. R. ODONI, J.-M. ROUSSEAU AND N. H. M. WILSON, "Models in Urban and Air Transportation," in *Operations Research and the Public Sector: Handbooks in Operations Research and Management Science,* Vol. 6, S. M. Pollock, M. H. Rothkopf, and A. Barnett (eds.), 107–150, Elsevier Science, Amsterdam, 1994.

I. M. OVACIK AND R. UZSOY, "Worst-Case Error Bounds for Parallel Machine Scheduling Problems with Bounded Sequence-Dependent Setup Times," *Opns. Res. Lett.* **14,** 251–256 (1993).

I. M. OVACIK AND R. UZSOY, "Rolling Horizon Algorithms for a Single-Machine Dynamic Scheduling Problem with Sequence-Dependent Setup Times," *Int. J. Prod. Res.* **32,** 1243–1263 (1994).

I. M. OVACIK AND R. UZSOY, "Rolling Horizon Procedures for Dynamic Parallel Machine Scheduling with Sequence-Dependent Setup Times," *Int. J. Prod. Res.* **33,** 3173–3192 (1995).

J. C. PICARD AND M. QUEYRANNE, "The Time-Dependent Traveling Salesman Problem and its Application to the Tardiness Problem in One-Machine Scheduling," *Opns. Res.* **26,** 86–110 (1978).

K. PLATZ AND U. BROKOF, "Optimising Air Traffic Flow at Airports," in *Advanced Technologies for Air Traffic Flow Management,* H. Winter and H.-G. Nüsser (eds.), 153–190, Springer-Verlag, Berlin (1994).

H. N. PSARAFTIS, "A Dynamic Programming Approach to the Aircraft Sequencing Problem," Report R78-4, Flight Transportation Laboratory, MIT, Cambridge, MA, 1978.

H. N. PSARAFTIS, "A Dynamic Programming Approach for Sequencing Groups of Identical Jobs," *Opns. Res.* **28,** 1347–1359 (1980).

J. E. ROBINSON, III, T. J. DAVIS, AND D. R. ISAACSON, "Fuzzy Reasoning-Based Sequencing of Arrival Aircraft in the Terminal Area," Presented at the *AIAA Guidance, Navigation and Control Conference,* New Orleans, LA, 1997. Available from *http://www.ctas.arc.nasa.gov/publications/*

P. A. RUBIN AND G. L. RAGATZ, "Scheduling in a Sequence Dependent Setup Environment with Genetic Search," *Comput. Opns. Res.* **22,** 85–99 (1995).

D. SIMCHI-LEVI AND O. BERMAN, "Minimizing the Total

Flow Time of n Jobs on a Network," *IIE Trans.,* **23,** 236–244 (1991).

R. SIMONS, "Making Best Use of the World's Favourite Runways," *OR Newsletter* September 19–21, (1997).

K. T. SO, "Some Heuristics for Scheduling Jobs on Parallel Machines with Setups," *Management Sci.* **36,** 467–475 (1990).

J. A. STEWART AND J. H. SHORTREED, "Risk and Capacity Impacts of ATC Separation Rules," *J. Transp. Eng.—Am. Soc. Civil Eng.* **119,** 705–722 (1993).

K. C. TAN AND R. NARASIMHAN, "Minimizing Tardiness on a Single Processor with Sequence-Dependent Setup Times: A Simulated Annealing Approach," *Omega,* **25,** 619–634 (1997).

J. A. A. VAN DER VEEN AND S. ZHANG, "Low-Complexity Algorithms for Sequencing Jobs with a Fixed Number of Job-Classes," *Comput. Opns. Res.* **23,** 1059–1067 (1996).

C. S. VENKATAKRISHNAN, A. BARNETT, AND A. R. ODONI, "Landings at Logan Airport: Describing and Increasing Airport Capacity," *Transp. Sci.* **27,** 211–227 (1993).

U. VÖLCKERS, "Dynamic Planning and Time-Conflict Resolution in Air Traffic Management," in *Lecture Notes in Control and Information Sciences 80: Artificial Intelligence and Man-Machine Systems,* H. Winter (ed.), 175–197, Springer-Verlag, Berlin, 1986.

U. VÖLCKERS, "A Rule-Based System for Arrival Sequencing and Scheduling in Air Traffic Control," in *Knowledge Based Concepts and Artificial Intelligence: Applications to Guidance and Control,* AGARD Lecture Series no. 155, 9:1–9:12, NATO, 1987.

U. VÖLCKERS, "Arrival Planning and Sequencing with COMPAS-OP at the Frankfurt ATC-center," in *Proceedings of the 1990 American Control Conference,* 496–501, IEEE, Piscataway, NJ, 1990.

C. H. WHITE AND R. C. WILSON, "Sequence Dependent Set-Up Times and Job Sequencing," *Int. J. Product. Res.* **15,** 191–202 (1977).

H. WINTER AND H.-G. NÜSSER (EDS.), "*Advanced Technologies for Air Traffic Flow Management,*" Springer-Verlag, Berlin, 1994.

(Received, November 1996; revisions received: February 1998, October 1998; accepted January 1999)