STRATEGIC FLOW MANAGEMENT FOR AIR TRAFFIC CONTROL

MOSTAFA TERRAB
Rensselaer Polytechnic Institute, Troy, New York

AMEDEO R. ODONI
Massachusetts Institute of Technology, Cambridge, Massachusetts

(Received October 1990; revision received March 1992; accepted June 1992)

One of the most important functions of air traffic management systems is the assignment of ground-holding times to flights, i.e., the determination of whether and by how much the take-off of a particular aircraft headed for a congested part of the ATC system should be postponed to reduce the likelihood and extent of airborne delays. In this paper, we will present an analysis of the fundamental case in which flights from many origins must be scheduled for arrival at a single, congested airport. We will describe a set of approaches for addressing a deterministic and a stochastic version of the problem. A minimum cost flow algorithm can be used for the deterministic problem. Under a particular natural assumption regarding the functional form of delay costs, a very efficient, simple algorithm is also available. For the stochastic version, an exact dynamic programming formulation turns out to be impractical for typical instances of the problem and we present a number of heuristic approaches to it. The models and numerical results suggest the potential usefulness of formal decision support tools in developing effective ground-holding strategies. Many methodological and implementation issues, however, still require resolution.

Air traffic congestion has become a widespread phenomenon in the United States and Europe. The principal bottlenecks of the ATC system are the major commercial airports, of which at least a dozen in the United States and half as many in Europe currently operate near or above their point of saturation under even moderately adverse weather conditions. The ranks of heavily congested airports will undoubtedly swell before the end of this decade, if even modest rates of demand growth prevail, as widely anticipated.

There are long-, medium- and short-term approaches for dealing with this important problem. In the long term (5–15 years), the system's ability to handle increasing demand can be enhanced through improved ATC technologies, more *concrete* (new airports and additional runways at existing airports), use of larger aircraft and development of more attractive surface transportation systems for short-haul travel. In the medium term (6 months to a few years), available devices include: the adoption of airport user charges designed to modify temporal demand patterns; the imposition of (partial) restrictions on access to some airports (*slot systems*); and government pressure on airlines to modify the temporal or geographical concentrations of their flights. Finally, in the short term, i.e., on a daily basis and with a planning horizon of 12–24 hours, the best the ATC system can do is attempt to minimize the extent and impact (cost to users) of unavoidable delays for given demand and available-capacity levels. To accomplish this, the flow of air traffic can be controlled to match, as well as possible, demand with the available capacity over time and across the various components of the ATC and the airport's network. This flow-control task is often referred to as the ATC Flow Management Problem (**FMP**) (see Odoni 1987 for an extensive introduction to the **FMP** and a discussion of the interplay between its technical and policy aspects).

This paper will deal with a fundamental subproblem of the **FMP**, specifically strategies for determining the optimal take-off times for aircraft flying to congested airports. Such strategies may require delaying the take-offs of some aircraft beyond their scheduled departure time, even when these aircraft are otherwise ready to depart.

This type of action, familiar to most air travelers, is known as a *gate-hold* or *ground-hold* and is motivated by the simple fact that, as long as a delay in reaching the airport of destination is unavoidable, it is both less costly and safer to absorb this delay on the ground before take-off, rather than in the air. Unfortunately,

Subject classifications: Dynamic programming, applications: limited lookahead and heuristics. Transportation, models: air traffic control flow management.
Area of review: DISTRIBUTION, TRANSPORTATION AND LOGISTICS (SPECIAL ISSUE ON STOCHASTIC AND DYNAMIC MODELS IN TRANSPORTATION).

0030-364X/93/4101-0138 $01.25

this is complicated by delays that cannot be predicted accurately in advance because the capacity (or acceptance rate) of airports is a probabilistic quantity and difficult to predict in marginal weather. For example, on an overcast day in Boston, the capacity of Logan Airport for arrivals can be anywhere in the range of 25–60 per hour and predictions of this capacity, even a couple of hours in advance, are subject to large errors.

Widespread use of ground-holding/gate-holding did not begin in practice until after the 1981 air traffic controllers strike in the United States. Today, ground-holding in the United States ATC system is initiated and coordinated by the ATC System Command Center (ATCSCC), known until recently as the Central Flow Control Facility, in Washington, D.C. ATCSCC is equipped with outstanding and ever-improving information-acquisition capabilities, including regional and local weather data and forecasts, up-to-the-minute data on the location and status of airborne traffic throughout the country, as well as projections of traffic demand for every airport over a time horizon of several hours. The air traffic controllers at ATCSCC use this information to assign ground-holds to departing aircraft several hours in advance at a time.

Briefly, early each morning, ATCSCC receives an hour-by-hour arrival capacity forecast for every major commercial airport in the United States, as well as a demand forecast for arrivals at these airports. Arrival queues and delays are then projected for the day at each airport by essentially viewing each airport as a deterministic service system with a first-scheduled, first-served discipline. When severe delays are projected in this way for an airport, a ground-holding program is initiated for that airport: Those flights whose projected delay exceeds 15 minutes are held on the ground, prior to departure, for a period of time equal to the projected delay. Flights whose projected delay is between 0 and 15 minutes are either not held on the ground or are assigned a ground delay of 15 minutes—the decision is made on a judgmental basis by the expert controllers at ATCSCC. A program may be revised later in a day if information from an airport indicates either a major change in the day's capacity forecast, or significant differences between the capacity actually being achieved and the previously forecast capacity, or, less frequently, a major change in the day's demand forecast. However, since it is not instantly ascertainable in practice that such "deviations" have indeed taken place, these program revisions sometimes are effected only after costs have been incurred in the form of either high airborne delays at the airport of destination or of unnecessary ground-holding delays at the airports of origin (and unused airport capacity at the airport of destination).

There is widespread recognition of the need for algorithms and decision support systems to assist the ATCSCC controllers in managing more effectively the flow of air traffic and in devising strategies for ground-holding. This is the aim of the research reported here and of work currently being conducted under the Advanced Traffic Management (ATM) and other programs for the FAA/DOT (FAA 1989). Three areas of inquiry that seem particularly promising are: taking explicitly into consideration the large amount of uncertainty which is often associated with airport capacity forecasts; considering the potential cost savings that might be obtained by deviating from ordering flight arrivals strictly according to their original scheduled times; and developing strategies that account for the dynamic decision-making environment within which the ATC system operates. We will address the first two of these areas in this paper and make some remarks about the third in the final section.

The available technical literature on the ground-holding policy problem (**GHPP**) is sparse. A paper by Attwool (1977) is the first effort, to our knowledge, to examine ATC congestion from a network-wide perspective and it offers several valuable insights. Sokkapia (1985) specifically addressed the issue of ground-holding strategies and proposed a very simple heuristic for the problem that essentially consists of estimating, first, the expected delay that any flight F_i would suffer in traveling to a congested airport Z, and then assigning a ground-holding time to F_i which is equal to a fraction of the expected delay. That fraction is based on the identity of the airport of origin and the flight time to Z, but no specific method for determining this fraction was proposed. Andreatta and Romanin-Jacur (1987) seem to be the only ones to have investigated in-depth an algorithmic approach to determining ground-holding times, albeit for a simplified version of the problem. Specifically, they studied a problem with a single-time period (see subsection 3.1) and n flights to a single destination (airport) and developed a dynamic programming (DP) approach for obtaining an optimal ground-holding strategy to minimize total delay costs for the n flights. In subsection 3.2, we describe a DP algorithm which extends this approach to the multiperiod case. Finally, Wang (1991) has suggested a DP-based approach for framing the entire **FMP**.

In Section 1, we describe the specific version of the ground-holding policy problem (**GHPP**) that is addressed here and present two models for analysis: one which assumes that airport capacities over the time period of interest can be forecast exactly (the deterministic model), and a second in which airport capacities are random variables described through a probability distribution (the stochastic model). In Section 2, we deal with the deterministic case and provide a very fast algorithm that can be used when the delay cost functions satisfy certain natural regularity conditions. Moreover, this algorithm serves as a building block for the heuristics we develop later for the probabilistic environment. In Section 3, we examine the stochastic model, develop a number of insights through an analysis of a case involving only a single period of time, and then present a DP-based algorithm for the general multiperiod case. Unfortunately, the DP algorithm is impractical for optimally solving problems of a size typical of what one might encounter at a major airport (400 or more arriving flights over a time period of roughly 15 hours). One is therefore forced to resort to heuristic approaches, of which we describe four general types in subsection 3.3. In Section 4, we give a brief summary of the findings of the extensive computational experiments we have carried out. The results are encouraging in suggesting an opportunity to achieve delay cost savings through increased use of decision support tools, utilizing exact or heuristic algorithms that address the combinatorial, stochastic and dynamic aspects of the problem. This should motivate further research on such questions as how strategies are best implemented in a dynamic setting, how uncertainty is explicitly recognized or how equitably delay costs are distributed among users. These issues are discussed briefly in the concluding section of the paper.

1. MODEL FORMULATION

We will examine a simplified, single-destination airport version of the **GHPP**. Specifically, we consider arrival operations at a given destination airport (airport Z) during a time interval $[0, T]$ for which we expect some congestion and during which N flights $F_1, F_2, \ldots, F_N$ are scheduled to land (Figure 1). The interval $[0, T]$ is subdivided into P-consecutive time periods $1, 2, \ldots, P$. (For instance, we may decide to subdivide a $T = 12$ hours period of high demand into $P = 72$ time intervals, each 10 minutes long.) The arrival capacity of the airport, i.e., the number of landings that can be served, during time period j will be denoted by K_j, $j = 1, \ldots, P$. For

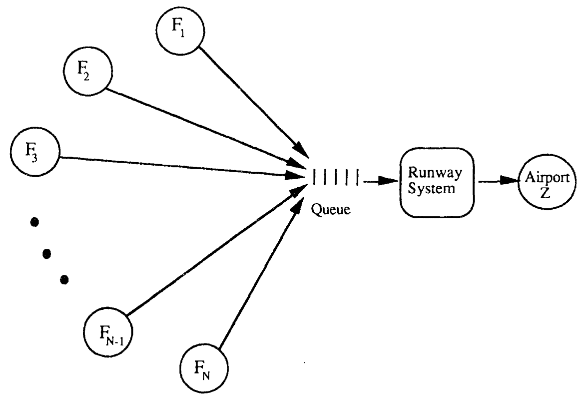

Figure 1. A single destination airport case.

each flight F_i we know:

- P_i, the index of the flight's scheduled landing period ($P_i = 1, \ldots, P$).
- $Cg_i(x)$, the cost of delaying flight F_i x time periods on the ground before take-off.
- $Ca_i(x, y)$, the cost of delaying flight F_i y time periods in the air before landing, if it has already been delayed x time periods on the ground.

Throughout the paper we make the following assumptions:

i. For any given x and y, $Ca_i(x, y) + Cg_i(x) > Cg_i(x + y)$.
ii. Travel times of aircraft between each origin and airport Z are deterministic and known in advance; delays can occur only as a result of congestion at Z.
iii. All flights that were not able to land during one of the time periods $1, 2, \ldots, P$ can do so during a final time period $P + 1$ (i.e., we assume that $K_{P+1} = \infty$).

Assumption i is clearly true in practice and provides much of the motivation for this work. Assumption ii is reasonable in view of the strategic nature of ground-holding decisions because, in practice, the amount of ground-holding is basically determined by heavy congestion at the major bottlenecks of the ATC system, i.e., the airports. Compared to airport-related capacity limitations and delays, such items as possible en-route sector congestion, and uncertainties in en-route flight times can be considered as second-order effects from the macroscopic viewpoint. Finally, assumption iii is technical and necessary to ensure that our problem formulations result in feasible solutions. In applications, arrivals assigned to time $P + 1$ are those whose landing will be scheduled beyond the time horizon of interest, e.g., until a period, such as early morning hours, when airport capacity always exceeds demand by a wide margin.

The **GHPP** is defined as the problem of finding the optimal amount of ground-hold to be imposed on each flight F_i so that the overall expected total delay cost (ground plus airborne delay costs) is minimized. A solution to the **GHPP** will be referred to as a Ground-Holding Policy (**GHP**) from here on.

We address the **GHPP** for two major cases: deterministic and probabilistic airport capacities.

1.1. Deterministic Model

In this case, we assume that the capacities $K_1, K_2, \ldots, K_P$ are deterministic and known before the earliest departure time of any flight $F_1, F_2, \ldots, F_N$. An important observation is that, given assumption i, it is never optimal under the deterministic model, to delay any flight in the air because it is possible to match demand with available capacity *exactly*. To see this, note that we can always improve on any ground-holding policy $\mathbf{GHP_0}$ that results in an airborne delay for a given flight F_i by considering the ground-holding policy $\mathbf{GHP_1}$ that differs from $\mathbf{GHP_0}$ only with respect to assigning an additional ground delay to F_i which is exactly equal to F_i's airborne delay under $\mathbf{GHP_0}$. Assumption i ensures that the total cost of $\mathbf{GHP_1}$ is strictly lower than that of $\mathbf{GHP_0}$.

In the deterministic case, the objective is therefore to find the **GHP** $X_1, X_2, \ldots, X_N$, which is feasible (does not violate the capacities K_i) and minimizes the total ground delay cost

$$TC = \sum_{i=1}^{N} Cg_i(X_i),$$

where X_i is the number of time periods flight F_i is delayed on the ground.

1.2. Stochastic Model

We now assume that the capacities $K_1, K_2, \ldots, K_P$ are random variables and that we are given a probabilistic forecast in the form of a joint probability mass function $P_{K_1}, \ldots, K_P$ for these capacities. It will be convenient to think of this probabilistic forecast as a number of scenarios $\mathscr{K}^1, \ldots, \mathscr{K}^C$, each scenario $\mathscr{K}^r$ representing a particular instance $\mathscr{K}^r = (k_1^r, \ldots, k_P^r)$ of the random capacity vector $K_1, K_2, \ldots, K_P$ with an associated probability of p^r.

Because of the uncertainty concerning the capacity of airport Z during $[0, T]$, we cannot predict anymore with certainty the amount of airborne delay that each flight would incur if it left on time. The **GHPP** is therefore now concerned with finding the amount of ground delay to impose on each flight to strike an optimal balance between the costs of these (known) ground delays and future (unknown) airborne delay costs based on available information concerning the capacity of airport Z.

We can compute, for the stochastic model, the total expected delay costs associated with a given **GHP** $(X_1, \ldots, X_N)$ by finding, for each capacity scenario $\mathscr{K}^r$, the airborne delay vector $(Y_1^r, \ldots, Y_N^r)$, where Y_i^r is the airborne delay to be imposed on flight F_i if scenario $\mathscr{K}^r$ materializes. The total expected delay cost to be minimized is given by

$$TC = \sum_{i=1}^{N} Cg_i(X_i) + \sum_{r=1}^{C} \left[p^r \sum_{i=1}^{N} Ca_i(X_i, Y_i^r) \right].$$

As the above expression suggests, delay costs for the stochastic model will normally consist of both ground delay *and* airborne delay costs, as the policy will attempt to balance *conservative* strategies (keeping airplanes on the ground too long and risking underutilizing the airport's capacity, if that capacity turns out to be on the "high side" of forecasts) against *optimistic* strategies (releasing airplanes from their airports of origin too early and thus risking incurring high airborne delays).

2. DETERMINISTIC CASE

In the deterministic case the **GHPP** reduces to a purely combinatorial problem for which standard solution methods are available. One motivation for looking at the deterministic case is that there are cases for which it is reasonable to assume that capacities can be forecast with little error. This is typically true for airports located in areas where there is little variability in weather conditions or for which changes in weather conditions are predictable and weather patterns remain stable for a long time period, once established. Another motivation is that we will be able to use deterministic solution methods to build approaches to the stochastic problem.

2.1. Standard Formulation

We set the assignment variables $x_{ij} = 1$ if flight F_i is assigned to land during period j, and $x_{ij} = 0$ otherwise. The x_{ij}'s are only defined for $j \geqslant P_i$.

We denote by C_{ij} the quantity $Cg_i(j - P_i)$, the cost of assigning flight F_i to land during time period T_j. This quantity is also only defined for $j \geqslant P_i$.

Using this notation, the solution of the following integer program (**IP**) yields the optimal policy for the deterministic **GHPP**:

$$\text{minimize } \sum_{i=1}^{N} \sum_{j=P_i}^{P+1} C_{ij} x_{ij} \tag{1}$$

subject to

$$\text{for all } (i, j): \quad x_{ij} = 0 \text{ or } 1 \tag{2}$$

$$\text{for all } i \ (= 1, \ldots, N): \quad \sum_{j=P_i}^{P+1} x_{ij} = 1 \tag{3}$$

$$\text{for all } j \ (= 1, \ldots, P): \quad \sum_{i=1}^{N} x_{ij} \leqslant K_j. \tag{4}$$

Figure 2 illustrates a capacitated network representation of (1)–(4). The numbers in brackets represent the costs per flow unit associated with the corresponding arc. When no number is indicated this cost is assumed to be zero. The letters u and l represent, respectively, the upper and lower limits for the flow on each arc. The default values are infinity for the upper bound and zero for the lower bound.

Each time period j is represented by an arc with the upper bound for the flow on this arc set to the capacity of the time period ($u = K_j$). Each flight F_i generates a node flight i in the above network. Each node flight i is connected to the nodes at the origin of the arcs representing all time periods with an index greater than or equal to P_i.

It is clear that the cost of any feasible flow through the network is

$$\sum_{i=1}^{N} Cg_i(j - P_i),$$

where j is the index of the time period corresponding to the nonzero flow out of the node flight i.

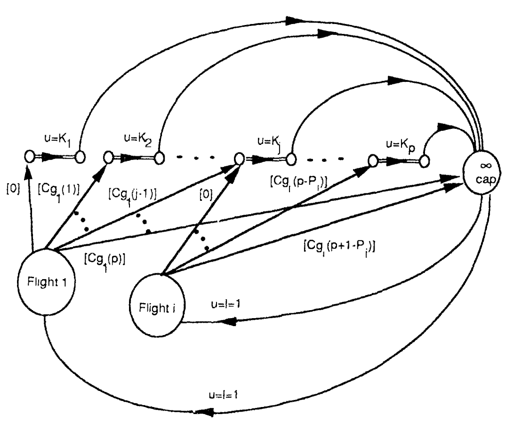

Figure 2. Capacitated network representation of deterministic **GHPP**.

The solution to (1)–(4) therefore corresponds to the minimum cost flow through this network.

2.2. Fast Algorithm

An algorithm that is even more efficient than the standard minimum cost flow algorithms can be devised to solve (1)–(4) when the cost functions $Cg_i(x)$ satisfy certain conditions which are reasonable for the particular ATC context under consideration.

Denote by $\partial_j Cg_i$ the marginal cost of delaying flight F_i during time period j ($j \geqslant P_i$), i.e.,

$$\partial_j Cg_i = Cg_i(j + 1 - P_i) - Cg_i(j - P_i).$$

Assume that the following two regularity conditions are satisfied by $Cg_i(x)$:

For any pair of flights (F_i, F_k):

i. If $\partial_j Cg_i > \partial_j Cg_k$ for some time period j, then for *any* $m > j$, $\partial_m Cg_i > \partial_m Cg_k$.

(Intuitively, this condition says that if it is more costly to delay flight F_i than it is to delay flight F_k during some time period j, then it is always cheaper to delay flight F_k rather than F_i later on.)

ii. If $\partial_j Cg_i = \partial_j Cg_k$ for some time period j, then for *any* $m > j$, $\partial_m Cg_i = \partial_m Cg_k$.

The following notation will be used to describe the very fast algorithm that can be used when the regularity conditions apply (or, equivalently, we will say that we are dealing with *regular* cost functions):

E_j = the set of indices of flights eligible for landing during time period j under the optimal policy;
OP_j = the set of indices of flights that actually are assigned to land during time period j under the optimal policy.

The following algorithm, which will be referred to throughout the remainder of this paper as the **Fast Algorithm**, determines an optimal policy $X_1, X_2, \ldots, X_N$:

STEP 1. **Initialize**

$E_0 = \varnothing$

$OP_0 = \varnothing$.

STEP 2. **For j = 1 to P do:**

$E_j = \{1 \leqslant i \leqslant N \mid P_i = j\} \cup E_{j-1} \backslash OP_{j-1}$

$OP_j = \{i \in E_j$ with the K_j highest $\partial_j Cg_i\}$.

STEP 3. **For $i = 1$ to N do:**

if $\exists j \in [1, \ldots, P]$ such that $i \in OP_j$, then set $X_i = j - P_i$; if no such j exists, then set $X_i = P + 1 - P_i$.

The algorithm thus consists of ordering, for each time period j, the candidate flights for landing according to their marginal cost of delay, and allowing the K_j flights with the highest marginal cost to land. The average-case complexity of the algorithm is therefore $O(N \log(N/P))$. The worst-case complexity of this algorithm is $O(PN \log N)$.

The correctness of the algorithm follows intuitively from the conditions imposed on the cost functions; they ensure that, when we consider candidate flights for a time period, these flights can be ordered unambiguously according to their potential costs for this and for all future time periods. Therefore, we can decide, in each time period, which flights to operate given the available capacity during that time period. A formal proof of the correctness of the algorithm (see Terrab 1990) requires more notation but also helps us to understand why the regularity conditions are needed.

Another algorithm with worst-case complexity of $O(N \log N)$ yields the same (optimal) policy. This algorithm consists of first ordering all flights according to their final marginal costs $\partial_{P+1}Cg_i$, and then, starting with time period 1, assigning flights to available capacity using this "final marginal cost ordering" to choose between two *eligible* flights that are competing for the same capacity (the flight with the highest final marginal cost gets priority).

3. STOCHASTIC CASE

We now turn to the stochastic case. In subsection 3.1, we present an interesting property of the single-time period case ($P = 1$) which will allow us to interpret some of the numerical results obtained for the multiple-time period case. In subsection 3.2, we present a dynamic program that solves the probabilistic version of the **GHPP** optimally when a fixed landing priority rule is given. In subsection 3.3 we describe several heuristics that will be evaluated in Section 4.

3.1. Single-Time Period Case

Consider a single-time period version of the **GHPP** (Figure 3) for which N flights, $F_1, \ldots, F_N$, are scheduled to land at airport Z during a single-time period T with capacity K (a random variable). Given a probabilistic forecast $P_K(k)$ for the capacity K, the **GHPP** consists of determining, for each flight F_i, whether F_i

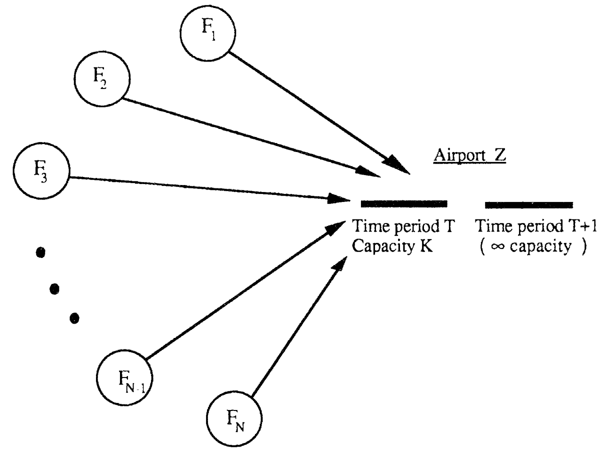

Figure 3. Single-time period case.

should be dispatched on-time to airport Z, in which case it may experience an airborne delay of at most one time period, or kept one time period on the ground, in which case it will be able to land without airborne delay during the next time period $T + 1$, which is assumed to have infinite capacity. This is the problem first addressed in Andreatta and Romanin-Jacur.

We denote by Cg_i the cost of delaying flight F_i one time period on the ground, and by Ca_i the cost of delaying it one time period in the air. We assume that the flights $F_1, \ldots, F_N$ have been indexed such that $Ca_{i+1} \geq Ca_i$.

It is obvious that under these conditions there exists an optimal landing priority rule which consists of giving landing priority to flights with higher indices when they show up for landing during time period T (a precise definition of a landing priority rule for the multiperiod case will be given in the next section). We will assume that this optimal landing priority rule is in effect in the remainder of this discussion of the single-time period **GHPP**.

We will denote by C the number of capacity cases $k_1, k_2, \ldots, k_C$ that the probabilistic forecast $P_K(k)$ distinguishes; we also denote by p_r the probability $P_K(k_r) \equiv \text{Prob}(K = k_r)$ with $\sum_{r=1}^{C} p_r = 1$. We will assume that $C > 1$ and that the capacities have been indexed such that $k_1 < k_2 < \ldots < k_C$.

The following interesting property of the single-time period **GHPP** is proved in Terrab, where it is referred to as Claim 3.

Dominating Capacity Property of the Optimal Solution

If we assume that $Ca_i = \gamma \cdot Cg_i$ for all $i = 1, \ldots, N$, where γ is a coefficient ($\gamma > 1$) *that does not depend* *on the index i*, then:

1. If there exists *no* index u ($1 \leqslant u \leqslant C$) such that $\gamma \cdot (p_1 + p_2 + \ldots + p_u) = 1$, then:

   i. If $\gamma \cdot p_1 > 1$, the policy that sends the k "most costly" flights $(F_{N-k+1}, \ldots, F_N)$ on time, where k is defined by $k = \min\{k_1, N\}$, is optimal.

   ii. Else, there must exist an index, s, such that $\gamma \cdot (p_1 + p_2 + \ldots + p_s) < 1$ and $\gamma \cdot (p_1 + p_2 + \ldots + p_{s+1}) > 1$ (since we know that $\gamma \cdot (p_1 + p_2 + \ldots + p_C) = \gamma > 1$), and the policy that sends the flights $(F_{N-k+1}, \ldots, F_N)$ on time, where $k = \min\{k_{s+1}, N\}$, is optimal.

2. If there exists an index s such that $\gamma \cdot (p_1 + p_2 + \ldots + p_s) = 1$, and if we define $k = \min\{k_s, N\}$ and $k' = \min\{k_{s+1}, N\}$, then any policy that sends the flights $(F_{N-k'-m+1}, \ldots, F_N)$ on-time is optimal for any m such that $0 \leqslant m \leqslant k' - k$.

The above property indicates that when $N \geqslant k_C$ (i.e., under certainly congested conditions), there always exists an optimal **GHP** which sends *exactly* k_r flights on-time to airport Z, for some index $r = 1, \ldots, C$, which corresponds to the dominating capacity case. This observation results in the following very simple procedure for finding an optimal **GHP** for the single-time period case:

- Find the smallest integer $u = u(P_K)$ such that $\gamma \cdot (p_1 + p_2 + \ldots + p_u) > 1$ (such an integer has to exist because $\gamma > 1$ by condition i of Section 1).
- Set $K_{\text{opt}}(P_K, N) = \min\{u(P_K), N\}$.
- The **GHP** P_{opt} that keeps the first $N - K_{\text{opt}}(P_K, N)$ flights on the ground is optimal.

3.2. Exact Dynamic Program

3.2.1. Description

The dynamic program (**DP**) we will develop is an extension of the DP presented in Andreatta and Romanin-Jacur for the single-time period case to cases for which we have a probabilistic forecast for the capacities $K_1, K_2, \ldots, K_P$ of P consecutive time periods $1, 2, \ldots, P$.

An exact optimal solution to the stochastic **GHPP** can be found through a dynamic program under the assumption that a fixed landing priority rule $\Pi_1, \Pi_2, \ldots, \Pi_N$ has been specified for the flights $F_1, F_2, \ldots, F_N$. By a fixed priority rule we mean that, if two flights F_i and F_k are airborne and are candidates for landing during the same time period, F_i will not be cleared for landing before F_k if $\Pi_i < \Pi_k$. For the remainder of this subsection we assume that the flights have been reordered, so that $\Pi_{i+1} > \Pi_i$ for all i.

The **DP** algorithm is based on two observations:

1. The expected (ground plus air) delay cost for a flight F_i, $C_i(X_i, \mathbf{Hi})$ depends only on two quantities:

   i. X_i = the ground-hold for flight F_i.

   ii. The vector $\mathbf{Hi} = (H_i^1, H_i^2, \ldots, H_i^P)$, where H_i^j represents the number of flights with priority greater than Π_i that are assigned to arrive at airport Z during time period j.

2. The optimal (minimum) value of the expected delay costs for the first i flights $(F_1, F_2, \ldots, F_i)$, $G_i(\mathbf{Hi})$, depends only on $\mathbf{Hi}$ and is given by the recursion formula:

$$G_i(\mathbf{Hi}) = \operatorname*{Min}_{X_i} \{C_i(X_i, \mathbf{Hi}) + G_{i-1}[\mathbf{H(i-1)}(\mathbf{Hi}, X_i)]\}, \tag{5}$$

where the functional relationship $\mathbf{H(i-1)}(\mathbf{Hi}, X_i)$ is defined as follows.

- For time period $P_i + X_i$ to which flight F_i is reassigned set

  $$H_{i-1}^{P_i+X_i} = H_i^{P_i+X_i} + 1.$$

  This states that the number of flights with higher priority than flight F_{i-1} scheduled during time period $P_i + X_i$ has to be increased by one unit if flight F_i is reassigned to that time period.

- For all other time periods T_j with $j \neq P_i + X_i$ set $H_{i-1}^j = H_i^j$, stating that the number of flights with higher priority than flight F_{i-1} scheduled during time period flight j does not change if F_i is reassigned to another time period.

The **DP** that determines the ground-holds $X_1^*, X_2^*, \ldots, X_N^*$ which minimize the total expected cost $\sum_{i=1}^N C_i(X_i, \mathbf{Hi})$ consists of two passes (for details see Terrab):

Pass 1 goes through all flights in *ascending order* (of landing priority) and finds, for all **Hi**, the $X_i(\mathbf{Hi})$ that minimizes $G_i(\mathbf{Hi})$ according to the recursive formula (5).

Pass 2 goes through all flights in *descending order* of priority and uses the results of pass 1 to retrieve the optimal ground-holds X_i^*, by computing successively the **Hi***'s that correspond to this optimal solution.

3.2.2. Complexity

The **DP** provides an *exact* solution to the (static) probabilistic formulation even if we do not assume "regular" cost functions. Unfortunately, we are constrained in the use of this **DP** algorithm both by its time and memory space complexity. Let us define:

M = the maximum airport capacity for any single-time period. It is reasonable to assume that M is a number of order N/P, i.e., of an order similar to the typical demand per time period at a congested airport;

C = the number of capacity cases considered in the probabilistic forecast.

We can then show (Terrab) that the time complexity of the **DP** is $O[NCP^2(M+1)^P]$ while its storage requirements are $O[N(M+1)^P]$.

The dominating factor in these expressions is $(M+1)^P \approx (N/P+1)^P$. A typical instance of the **GHPP** could involve several hundred flights and a total time span of several hours. For $N = 500$, $T = 12.5$ hours and time periods of 15 minutes each, we have $P = 50$ periods and $(N/P+1)^P = 11^{50}$. The practicality of the exact **DP** approach is therefore as those described in the next subsection. Before doing this, however, it is important to note that the proposed **DP** algorithm is well suited for parallelization as it consists of a large number of quasi-independent steps that can be carried out simultaneously on separate processors. More details can be found in Terrab.

3.3. Heuristics

Motivated by the complexity of the exact **DP** approach we now consider heuristics for the stochastic version of the **GHPP**. We have identified four groups of heuristics for evaluation: Limited Lookahead; Maximum Marginal Return; Equivalent Capacity; and Equivalent Policy. These are described below.

3.3.1. Limited Lookahead

Limited Lookahead consists of subdividing the set of P time periods into smaller sets (of three time periods, for example) and running the **DP** algorithm on each subproblem consecutively, taking into account the flights *left over* from the previous subproblem (i.e., those landing in the infinite capacity final period relative to the previous subproblem). The motivation for doing this is a reduction in time complexity as well as storage space requirements. Assume that we use Q such subproblems of R-time periods each ($P = Q \cdot R$). The implementation goes as follows.

STEP 1. Apply an R-time period version of the **DP** described in the previous subsection to the first R-time periods $1, 2, \ldots, R$ for all flights scheduled to land at airport Z during $1, 2, \ldots, R$, using $P_{K_1,K_2,\ldots,K_R}$, and assuming that $K_{R+1} = \infty$. The resulting assignments of flights to time periods $1, 2, \ldots, R$ are final; flights assigned to time period $R+1$ are carried over to the next subproblem.

STEP 2. Apply the same size **DP** to time periods $R+1, R+2, \ldots, 2R$ for all flights scheduled during $R+1, R+2, \ldots, 2R$ as well as any flights that were assigned to time period $R+1$ by the previous suboptimization.

STEP 3. Continue this sequence of suboptimizations until the final subproblem which corresponds to time periods $P-R+1, P-R+2, \ldots, P$.

As is the case for the general **DP** formulation, **Limited Lookahead** can be used with general cost functions and assumes that a fixed landing priority rule is in effect. (A referee has suggested a less "myopic" version of the **Limited Lookahead** heuristic. Under this, only flights assigned to time period 1 become final at the end of Step 1. In Step 2, the **DP** for time periods $2, 3, \ldots, R+1$ are then solved and flights assigned to period 2 become final. This continues until the final **DP** which involves periods $P-R+1, P-R+2, \ldots, P$. We have not experimented with this version because, as indicated in Section 4, **Limited Lookahead** turned out to be much slower than the other heuristics proposed below and the new version would be even slower because it repeats the **DP** $P-R$ times. On the other hand, the quality of the solutions provided by this modified version would probably be better than that of the original.)

3.3.2. Maximum Marginal Return Heuristic

The Maximum Marginal Return heuristic (**MMR**) works with general cost functions and assumes that a fixed landing priority rule is in effect. The basic idea for this heuristic comes from observation 1 of subsection 3.2.1, namely that if we have a fixed landing priority rule, the expected cost for flight F_i, $C_i(X_i, \mathbf{Hi})$, depends only on X_i and the status of flights of higher priority represented by the vector **Hi**.

Suppose that flights have been indexed so that $\Pi_{i+1} > \Pi_i$ for all i. By definition, the flight with highest priority F_N is such that $\mathbf{HN} = \mathbf{0}$. Therefore, the expected (ground plus air delay) cost for flight F_N, $C_N(X_N, \mathbf{0})$, does not depend on the status of any other flight; it depends only on the ground-hold X_N that we impose on flight F_N. Thus, we can find the optimal ground-hold X_N^* that leads to the lowest expected delay cost for F_N. Once we have X_N^*, we can compute $\mathbf{H(N-1)}^*[\mathbf{0}, X_N^*]$ through the procedure outlined in subsection 3.2.1 and we can therefore also compute, for each possible ground-hold X_{N-1}, the expected cost for flight F_{N-1}, $C_{N-1}(X_{N-1}, \mathbf{H(N-1)^*[0}, X_N^*])$. Again this allows us to find the optimal ground-hold X_{N-1}^* for flight F_{N-1} as the one that yields the lowest expected cost for that flight. This procedure is repeated until we have computed the optimal ground-hold for the flight with the lowest landing priority F_1.

The **MMR** therefore computes ground-holds by minimizing the expected cost for each flight individually, starting with the flight with the highest priority. More precisely, the N steps of the algorithm are:

STEP 1. Find

$X_N^* = \arg\min C_N(X_N, \mathbf{0})$.

Compute $\mathbf{H(N-1)^*[0}, X_N^*]$ (using the procedure outlined in subsection 3.2.1).

STEP 2. Find

$X_{N-1}^* = \arg\min C_{N-1}(X_{N-1}, \mathbf{H(N-1)^*[0}, X_N^*])$.

Compute

$\mathbf{HN-2^*[H(N-1)^*}, \mathrm{N}_{N-1}^*]$.

:

STEP N + 2 − i. Find

$X_{i-1}^* = \arg\min C_{i-1}(X_{i-1}, \mathbf{H(i-1)^*[Hi^*}, X_i^*])$.

Compute

$\mathbf{Hi-2^*[H(i-1)^*}, X_{i-1}^*]$.

:

STEP N. Find

$X_1^* = \arg\min C_1(X_1, \mathbf{H1^*[H2^*}, X_2^*])$.

A major component of this algorithm is the computation, for a given flight F_i, of the expected cost associated with a given ground-hold X_i, and a given vector **Hi**. The time complexity of **MMR** is $O[NCP^2]$.

Note that if we use regular (ground and air-delay) cost functions (as, for example, those that will be described in Section 4) and use the resulting optimal landing priority rule, **MMR** translates into a "myopic" strategy that consists of going after the lowest possible marginal cost increase at each step of the algorithm (thereby the appellation maximum marginal return).

The next two heuristics are intended to take advantage of the availability of efficient algorithms for the deterministic case (as, for example, the **Fast Algorithm** (subsection 2.2) developed for the deterministic version of the **GHPP** with regular cost functions).

3.3.3. Equivalent Capacity Heuristics

The strategy for this family of heuristics is to reduce the stochastic problem to a deterministic one by reducing the probabilistic forecast $P_{K_1,K_2,\ldots,K_P}$ to a single set of equivalent deterministic capacities $EK_1, EK_2, \ldots, EK_P$ and then solve the **GHPP** by using one of the approaches discussed in Section 2.

The most obvious is the **Expected Capacity** heuristic which consists of using $EXK_1, EXK_2, \ldots, EXK_P$ as the equivalent deterministic capacities, where EXK_u is the expected capacity for time period u computed from $P_{K_1,K_2,\ldots,K_P}$ in the following manner:

$$EXK_u = \sum_{r=1}^{C} k_u^r \cdot p^r.$$

It is important to realize that this heuristic is "blind" to the magnitude of airborne-delay costs; it determines ground delays based solely on ground costs $Cg_i(x)$. (For example, if we use the regular cost function described in Section 4, we would obtain through this heuristic the same **GHP** independently of the value of the coefficient γ.) Since airborne-delay costs per unit of time are typically higher than ground costs we can expect this heuristic to yield solutions that are too optimistic in the sense that they do not impose enough ground delays to minimize the total expected cost. We have therefore investigated another heuristic in this same family that is intended to compensate for this bias: the Weighted Capacity Heuristic (**WEK**) consists of using equivalent capacities $WK_1, WK_2, \ldots, WK_P$ where:

$$WK_u = \sum_{r=1}^{C} k_u^r \cdot w^r \qquad \left(\sum_{r=1}^{C} w^r = 1\right).$$

The weights w^r allow hedging against unfavorable capacity cases by giving more weight (compared to the actual probabilities p^r) to these cases. The extreme case is to set $w^e = 1$ for the lowest capacity case e, resulting in a very pessimistic strategy. This policy can be optimal if the airborne-delay costs per unit of time are very high compared to ground-delay costs.

Any algorithm that can solve the deterministic version of the **GHP** can be used as an equivalent capacity heuristic, once the equivalent capacities have been selected. The time complexity of the algorithm is then equal to the complexity of the deterministic algorithm used. For the case of regular cost functions, the **Fast Algorithm** can be used and then the time complexity of the **Equivalent Capacity** Heuristic is $O[N \log N]$. For the case of general cost functions, we can use the minimum cost flow formulation (ses subsection 2.1).

3.3.4. Equivalent Policy Heuristics

The strategy for this family of heuristics is to compute ground-holding policies for each capacity case

$\mathscr{K}^r = (k_1^r, \ldots, k_P^r)$ separately (using any deterministic algorithm) and then to aggregate these policies into a single one. Again, the most obvious is the **Expected Policy Heuristic** which consists of:

STEP 1. For each capacity case $\mathscr{K}^r = (k_1^r, \ldots, k_P^r)$, compute the deterministic policy $(X_1^r, X_2^r, \ldots, X_N^r)$, where X_i^r is the optimal ground-hold for flight F_i if the capacity sequence $\mathscr{K}^r$ were to materialize.

STEP 2. Set the ground-holding time X_i for flight F_i equal to the "expected" ground-hold for flight F_i:

$$EX_i = \sum_{r=1}^{C} X_i^r \cdot p^r.$$

The same observation as for the equivalent capacity heuristics about the optimistic character of this policy can be made. We can, however, use a similar modification (a weighted policy heuristic) to try to compensate for this, namely use adjusted weights w^j instead of the actual probabilities p^j in the computation of the final ground-holds:

$$WX_i = \sum_{r=1}^{C} X_i^r \cdot w^r.$$

The time complexity of the algorithms in this family is $O[CN \log N]$ if regular cost functions are used.

Finally, note that the **Weighted Capacity** heuristic is equivalent to the **Weighted Policy** heuristic when we set one of the weights to 1 (and all others to 0).

4. EXPERIMENTAL RESULTS

4.1. Introduction

We have investigated the performance of both exact and heuristic algorithms through extensive numerical experiments whose findings we only summarize in this section. For details the reader is referred to Terrab.

The numerical experiments were conducted for quite realistic cases modeled on arrivals at Boston's Logan Airport, the country's 10th busiest, using a MacII cx®. This particular choice of a destination airport allowed us to consider cases that involve as many as 700 arrivals during the congested time span of the day. The delay costs assigned to individual aircraft are representative of actual aircraft costs (i.e., the sum of fuel costs, aircraft depreciation and maintenance costs, and crew costs); they do not include delay costs to passengers. The experimental setting is as follows.

We consider operations at a given airport Z during a given day according to a schedule that resembles Logan Airport during a typical day, using 1987 data. Since 95% of the total daily operations occur between 7 a.m. and 11 p.m., we restrict our analysis to this time span. The total number of scheduled landings for each hour of the day is an input. A random number generator determines the exact scheduled landing time for each individual flight using a uniform distribution within each hour. (This is equivalent to simulating the instants of Poisson arrivals, *given* the number of arrivals per hour.) We distinguish three types of aircraft on the basis of their ground delay costs. A random number generator assigns each fight to one of the three categories according to a prespecified flight mix. The delay costs for the first period of ground delay are based on direct operating costs per hour equal to \$400 for aircraft type 1 (general aviation and small commuter aircraft), \$1,200 for aircraft type 2 (narrow-body jets), and \$2,000 for aircraft type 3 (wide-body jets); the mix is 40% of aircraft type 1, 40% of aircraft type 2, and 20% of aircraft type 3. Airborne-delay costs per hour are set higher, reflecting fuel costs as well as a premium for safety, increased ATC workload, etc. (see also subsection 4.3).

4.2. Deterministic Case

Almost all the numerical examples in this paper will assume that the marginal ground costs (see subsection 2.2) are given by

$$\partial_j Cg_i = C_i(1 + \alpha)^{j-P_i},$$

where C_i is the cost of delaying flight F_i for one time period on the ground ($C_i = Cg_i(1)$), and α is a coefficient that does not depend on the index i. It is a simple matter to verify that these cost functions satisfy the regularity conditions of subsection 2.2. Since flights that belong to the same aircraft type are assumed to have the same delay cost functions, we use only three values of C_i based on our classification of aircraft into three categories, as explained in subsection 4.1.

The coefficient α will play an important role in these numerical examples because, as we will see, the magnitude of α affects the distribution of delays among classes of aircraft. For example, setting α to 0, which corresponds to assuming linear cost functions, can be expected to assign a disproportionate amount of delays to smaller aircraft (aircraft type 1); on the other hand, a very high α will result in assigning aircraft to available capacity on a roughly FCFS basis, as is usually done under present practice. For an

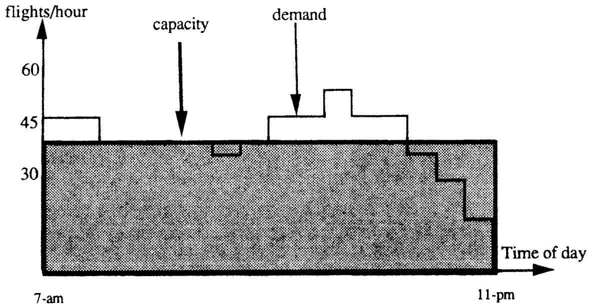

Figure 4. Example of demand and capacity profile for arrivals at Logan Airport.

interpretation of α we note that

$$\alpha = \frac{\partial_j Cg_i - \partial_{j-1} Cg_i}{\partial_{j-1} Cg_i},$$

which is the *fractional increase in cost* due to holding a flight on the ground for an additional time period.

The following numerical example is intended to illustrate the impact of different values of α on the costs as well as on the distribution of ground-holds among aircraft types. For comparison, these results are contrasted with those obtained for the same sample problem when using an FCFS policy. Furthermore, even when the **GHPs** are obtained using $\alpha \neq 0$, the costs we compare are computed assuming linear costs ($\alpha = 0$) to allow comparison with previous cases.

Figure 4 shows the relationship of demand to capacity assumed in this numerical example. We use the **Fast Algorithm** to determine the optimal **GHP** over a 16-hour time span using 10-minute long individual time periods ($P = 96$). A capacity of 40 landings per hour is assumed throughout the day. This capacity scenario is typical of Logan Airport under day-long nonideal weather conditions.

Table I summarizes the results obtained for different values of α. The numbers under the columns with the heading N15 represent, for each aircraft type, the number of flights delayed less than 15 minutes, the numbers under N30 represent the number of flights delayed between 15 and 30 minutes, and so on, until N2UP which corresponds to the number of flights delayed more than 2 hours. It is clear from these results that if linear costs ($\alpha = 0$) are used to determine the ground-holding policies, the algorithm will generally give priority to types 2 and 3 aircraft. This raises questions of equitability because one category of users (namely aircraft type 1 in this case) will always be penalized. The results also show that this state of affairs can be remedied, if a less than optimal solution is accepted, through a judicious choice of α. Such a choice of α (see, e.g., the results for $\alpha = 4$) can often achieve significant total cost savings, while distributing delay costs among users in a more equitable fashion. Some of the differences in total delay costs may seem very large but this is due to the fact that we chose a

Table I
Sensitivity of Solution to the Parameter α

Flight Type	# Flights Per Type	Total Flight-Hour Delay Per Type	N15	N30	N45	N60	N90	N120	N2UP
				$\alpha = 0$					
Total delay cost for **fast algorithm** = $76,130									
Total delay cost for first-come first-served policy = $191,190									
1	263	181	67	64	39	12	29	52	0
2	222	3	222	0	0	0	0	0	0
3	129	0	129	0	0	0	0	0	0
				$\alpha = 2$					
Total delay cost for **fast algorithm** = $88,397									
Total delay cost for first-come first-served policy = $191,190									
1	263	166	67	64	39	12	62	19	0
2	222	17	187	26	9	0	0	0	0
3	129	0	129	0	0	0	0	0	0
				$\alpha = 4$					
Total delay cost for **fast algorithm** = $106,600									
Total delay cost for first-come first-served policy = $191,190									
1	263	149	67	64	39	17	76	0	0
2	222	29	165	26	31	0	0	0	0
3	129	6	116	13	0	0	0	0	0

highly congested example (a one-third reduction from the maximum landing capacity throughout the day) to clearly illustrate the combinatorial aspect.

4.3. Probabilistic Case

In the probabilistic case, an optimal policy may and usually will also include airborne delays. Therefore, in addition to ground-holding costs, we need airborne-delay costs for each flight. Although all the algorithms presented in Section 3 can work with general cost functions, most of our examples will assume that both the ground-delay and the airborne-delay cost functions satisfy the regularity conditions. More specifically, we will assume that $Ca_i(x, y)$ is such that the marginal cost of delaying flight F_i during time period j ($j \geq P_i + x$) in the air $\partial_j Ca_i$ is given by:

$$\partial_j Ca_i \equiv Ca_i(x, j - P_i - x + 1) - Ca_i(x, j - P_i - x)$$
$$= \gamma C_i(1 + \alpha)^{x-1}(1 + \beta)^{j-P_i-x+1}.$$

The coefficient β plays the same role for airborne-delay costs as does α for ground delay costs. We note that

$$\beta = \frac{\partial_j Ca_i - \partial_{j-1} Ca_i}{\partial_{j-1} Ca_i},$$

which is the fractional increase in cost due to holding a flight in the air for an additional time period.

When we assume that $\alpha = \beta$ (as was done in many of our numerical examples) we get $\partial_j Ca_i = \gamma C_i (1 + \alpha)^{j-P_i}$, in which case, γ can be interpreted as a multiplicative coefficient intended to reflect the ratio between the direct airborne operating costs of aircraft and the direct costs of keeping them on the ground (recall that $C_i \equiv Cg_i(1)$). The coefficient γ was typically set to 2 (i.e., marginal airborne delays were assumed to be twice as costly as marginal ground delays).

We performed a large number of experiments comparing the algorithms described in Section 3. Here we present the main conclusions drawn from these comparisons as well as a few typical experimental results that support these conclusions. The reader interested in additional details is referred to Terrab. These conclusions can be summarized in four major points:

1. More sophisticated methods (such as those typified by our heuristics) which take uncertainty into account can generate solutions to the **GHPP** that provide significant savings in delay costs when compared with strategies that disregard uncertainty. Figure 5 shows an example involving a 9-hour time span and three possible capacity scenarios, $KAP1$, $KAP2$, and $KAP3$, with associated probabilities of, respectively, 0.4, 0.4, and 0.2. For one typical run, **MMR** led to a total (ground + airborne) expected delay cost of $152,300, versus $309,220 for an FCFS strategy with a deterministic capacity forecast which consists of the expected capacity for each time period (this run used linear cost functions).

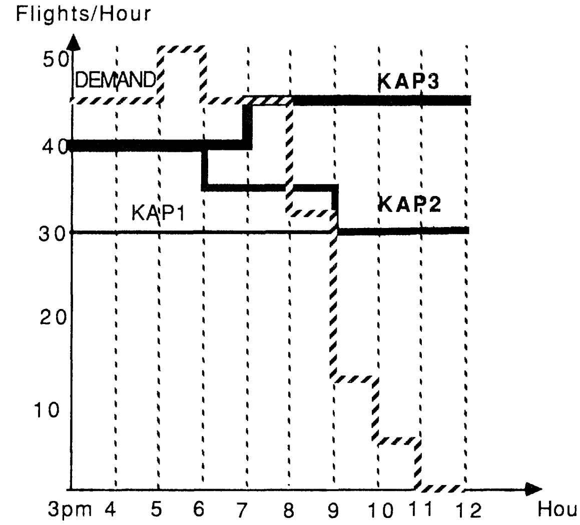

Figure 5. Capacity cases and demand profile.

2. **MMR** and **Limited Lookahead** seem to perform better than the expected capacity or the expected policy heuristics when we use regular cost functions (**Limited Lookahead**, however, runs significantly more slowly than **MMR**). The results of another example (Table II) illustrate these points; it involves 72 flights and three capacity cases $KAP1$, $KAP2$, and $KAP3$ (Figure 6), with respective probabilities, 0.3, 0.5, and 0.2. (**MMR** ran in a few seconds versus about 20 minutes for **Limited Lookahead** for problems of this size.)

3. The exact **DP** approach can outperform all our heuristics by a considerable margin when we consider general cost functions. However, it is very demanding computationally even for small instances of the

Table II
Comparison of Heuristics

Heuristic	Ground Costs ($)	Total (Ground + Airborne) Expected Costs ($)
MMR	6,606	10,661
Limited Lookahead	6,606	10,661
Expected Capacity	6,045	11,222
Expected Policy	6,849	14,837

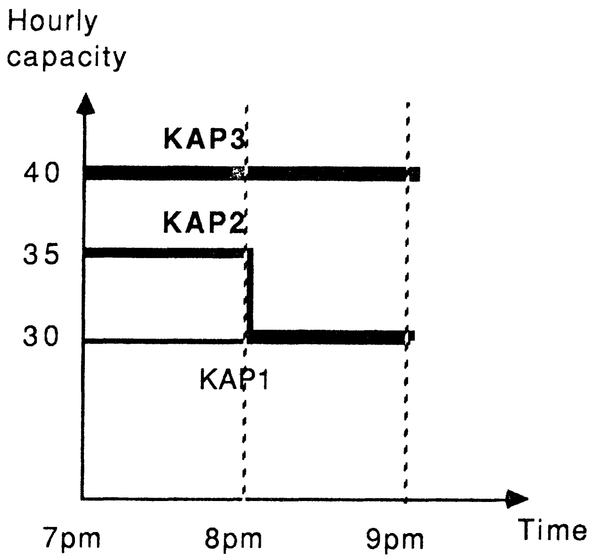

Figure 6. Capacity cases.

GHPP. This was demonstrated on several small numerical examples (we were limited in the size of the sample problems by the time and space complexity of the exact **DP** approach as they related to the particular hardware used) for which we introduced cost functions that, for some flights, increased sharply after a given time threshold (and thus violated the regularity conditions). The exact **DP** approach typically resulted in policies with total expected delay costs that were as much as 20% lower than those obtained using the heuristic approaches. The exact **DP** approach required almost all five megabytes of available memory and several minutes of run time for problems involving only twelve flights and three time periods.

4. Good solutions can be found with a proper choice of weights using the **WEK** heuristic. Furthermore, with less than four capacity scenarios, the simple criterion developed for the selection of the dominating capacity scenario in the single-time period case can be used to correctly select the weight that should be set to 1 for the multiperiod case (so we can think of this result as an extension of the dominating capacity property to the multiperiod **GHPP**, when capacity scenarios can be ordered according to capacity magnitudes as is the case in our numerical examples). However, these results do not hold when there exist four or more capacity scenarios in the probabilistic forecast.

The following examples illustrate these points. Table III shows the results for a run that used the capacity cases shown in Figure 6 and probabilities for the capacity cases $KAP1$, $KAP2$, and $KAP3$ set to, respectively, 0.3, 0.4, and 0.3. It indicates that when the weight of capacity case $KAP2$ is set to 1 heuristic **WEK** performs as well as **MMR**. Table IV shows the results when a fourth capacity case, which corresponds to a capacity of 35 flights per hour between 3 p.m. and midnight, is considered (the probabilities for the four capacity cases are set to 0.35, 0.25, 0.25, and 0.15). None of the policies obtained from **WEK** by setting one of the weights to 1 performs as well as **MMR**.

Table III
Three Capacity Cases

	Weights for $KAP1$, $KAP2$, $KAP3$ Used in WEK			
	0.4, 0.5, 0.1		0, 1, 0	
Heuristic	Ground Costs ($)	Total Expected Costs ($)	Ground Costs ($)	Total Expected Costs ($)
MMR	62,100	130,740	62,100	130,740
WEK	96,900	146,260	62,100	130,740

Finally, we have conducted experiments designed to evaluate how the benefits from implementing ground-holding policies generated by **MMR** change with the level of utilization of the destination airport. This was done by running cases with different total numbers of scheduled flights and comparing the delay costs per scheduled flight obtained with **MMR** to those obtained from using an FCFS strategy with a deterministic capacity forecast which consists of the most likely capacity scenario. As can be expected, the experiments show that the cost savings increase significantly as the destination airport becomes more congested.

5. DISCUSSION

As the computational experiments with both deterministic and stochastic problems indicate, well-designed ground-holding policies have excellent

Table IV
Four Capacity Cases

Heuristic	Ground Costs ($)	Total (Ground + Airborne) Expected Costs ($)
MMR	102,300	167,230
WEK with weights 1, 0, 0, 0	176,500	176,500
WEK with weights 0, 1, 0, 0	62,100	175,400
WEK with weights 0, 0, 1, 0	35,200	198,940
WEK with weights 0, 0, 0, 1	97,300	169,720

potential for achieving large savings in total delay costs. It is clear therefore that this is a promising area for further research. The most important area for such work involves the dynamic aspects of the stochastic **GHPP**. In the stochastic model we have formulated, the probability p^r of each possible capacity scenario $\mathscr{K}^r$ is assumed to remain fixed (static) over time. Thus, an optimal **GHP** can be determined *once and forever*, in this case, before the earliest departure time of flights to airport Z. This optimal **GHP** minimizes the here and now total expected delay costs.

However, **GHP**s thus obtained may not be optimal under a dynamic decision-making scenario. In this latter case, the optimal **GHP** has to be expressed as a *rule* in which the ground-hold X_i also depends on the history of airport Z's capacity up to the time of the departure of each flight F_i. This can be seen through a simple example.

Consider two flights F_1 and F_2 scheduled to land at airport Z during the same time period. We assume that airport Z's capacity K during that period is probabilistic and can take only two values: $K = 2$ with probability 0.3 and $K = 1$ with probability 0.7 ($P_K(2) = 0.3$; $P_K(1) = 0.7$). The capacity of Z in the next time period is assumed to be infinite, so the largest delay that can be incurred is one time period.

Flight F_1 is a long-haul flight with a scheduled departure time t_1 and costs $Cg_1(1) = \$1{,}000$ and $Ca_1(1) = \$2{,}000$ (see Figure 7). Flight F_2 is a short-haul flight with scheduled departure time $t_2 > t_1$ and costs $Cg_2(1) = \$1{,}200$ and $Ca_2(1) = \$2{,}400$. Furthermore, we assume that the uncertainty about K will be resolved by time t_2, the time of departure of F_2 so that 70% of the time we know at time t_2 that $K = 1$ and 30% of the time we know that $K = 2$.

If we consider only a static formulation of this single-time period problem at time $t = t_1$, i.e., if we consider only the probabilistic forecast P_K at $t = t_1$, it is not difficult to see that the optimal solution is to keep F_1 one time period on the ground and to send

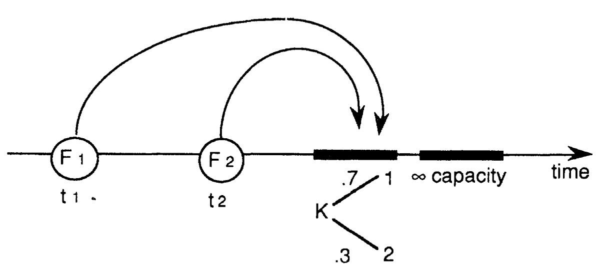

Figure 7. Dynamic two-flight example.

F_2 to airport Z on time. The total expected cost of this policy is \$1,000.

Now suppose that we want to find the optimal dynamic solution by recognizing that the uncertainty in K will be resolved at time t_2. In this case, the optimal solution is to send F_1 on-time and decide what to do with flight F_2 at time t_2 for an expected cost of $(0.7 \times 1{,}200 + 0.3 \times 0) = \$840$. We have used the assumption that K is deterministic at time t_2 only to illustrate the point more vividly; the same argument can be made by using any properly selected conditional probabilistic forecast at t_2.

Despite its extreme simplicity, this example serves to suggest that dynamic **GHP** models differ from static ones in some fundamental respects that go beyond the requirements to simply update strategies over time. First, in addition to scheduled arrival times, scheduled departure times must also be considered explicitly in developing dynamic strategies. Second, a new set of initial conditions must be considered whenever strategies are updated. These initial conditions refer to aircraft already delayed on the ground or already airborne due to actions taken earlier. Third, in the case of long-haul flights which may already be airborne at the time when strategies are updated, some tactical decisions (such as slowing down or speeding up the airplane on its way to its destination) may affect overall ground-hold policies in important ways.

In addition to illustrating how static and dynamic information-updating scenarios can result in different ground-holding policies, our example also points to what is indeed a systematic bias of optimal policies in the dynamic case. Namely, optimal policies favor long-range flights over short-range ones, in the sense that long-range flights are more likely to be allowed to take-off on-time (i.e., with no or little ground-holding). Intuitively, good dynamic ground-holding policies tend to be more active with short-range flights (i.e., impose more ground-holds on them) in order to take advantage of the improved state of knowledge at the time when short-range flights are scheduled to depart. Indeed, current practice in the ATC system partly reflects this tendency: For example, flights to the United States from Europe or nonstop coast-to-coast flights are exempt from ground-holding, in all but exceptional circumstances.

Many difficult practical issues need to be addressed before models and decision support tools such as those described here can be adopted for ATC flow management. For one, systematic biases in ground-holding policies that may result from such tools (e.g., the bias in favor of long-range flights we just discussed or the bias in favor of larger airplanes exhibited in the example of subsection 4.2) would have to be recognized and dealt with, possibly through constrained optimization or through the kind of adjustment indicated in subsection 4.2. The policy of FCFS after takeoff (with some adjustments) for all users is a fact of life in the United States, but not necessarily in the rest of the world. Thus, whenever an approach may favor large commercial aircraft systematically, it may be difficult to implement and thus achieve the aggregate cost savings shown here.

A second set of issues concerns appropriate ways to introduce the explicit consideration of uncertainty regarding future weather and future airport capacities and demand. This has to be done in a manner acceptable to air traffic controllers through carefully designed interfaces with automation aids and decision support tools. Currently, because of the essentially "manual" nature of the decision making process, uncertainty is only implicitly considered.

Finally, there is a host of questions related to the hierarchy, distribution and timing of decision making responsibilities regarding ground-holds, especially in a dynamic context.

ACKNOWLEDGMENT

The authors wish to thank the editors, the associate editor, and two referees for helpful suggestions on improving this paper. The research for this paper was supported in part by the Charles Stark Draper Laboratory, Inc.

REFERENCES

ANDREATTA, G., AND G. ROMANIN-JACUR. 1987. Aircraft Flow Management Under Congestion. *Trans. Sci.* **21,** 249–253.

ATTWOOL, V. W. 1977. Some Mathematical Aspects of Air Traffic Systems. *J. Inst. Navig.* **30,** 394–411.

Federal Aviation Administration. 1989. *Plan for Research, Engineering and Development*, Vols. I and II, Washington, D.C. (January).

ODONI, A. R. 1987. The Flow Management Problem in Air Traffic Control. In *Flow Control of Congested Networks*, A. R. Odoni, L. Bianco and G. Szego (eds). Springer-Verlag, Berlin, 269–288.

SOKKAPIA, B. G. 1985. *Arrival Flow Management as a Feedback Control System.* The MITRE Corporation, Washington, D.C.

TERRAB, M. 1990. *Ground Holding Strategies for Air Traffic Control.* Ph.D. Dissertation, Operations Research Center, Massachusetts Institute of Technology, Cambridge, Mass.

WANG, H. 1991. An Exact Dynamic Programming Approach to the Overall Flow Control Problem in Air Traffic Management. *Trans. Sci.* **25,** 308–313.