GROUND DELAY PROGRAM PLANNING UNDER UNCERTAINTY BASED ON THE RATION-BY-DISTANCE PRINCIPLE

Michael O. Ball, Robert H. Smith School of Business and Institute for Systems Research, University of Maryland, College Park, mball@rhsmith.umd.edu

Robert Hoffman, Metron Aviation, Inc., Herndon, VA , hoffman@metronaviation.com

Avijit Mukherjee, University Affiliated Research Center, University of California (Santa Cruz), Moffett Field, CA, avijit@ucsc.edu

October 25, 2007

Abstract: This paper presents ration-by-distance (RBD), a new allocation method to be used in planning ground delay programs (GDPs) for Traffic Flow Management. It is shown that RBD minimizes total expected delay, under certain assumptions related to the manner in which GDP's are dynamically controlled. On the other hand, RBD taken to the extreme has poor characteristics with respect to the equity of the allocation it produces. To address this issue, we propose a constrained version of RBD as a practical alternative to allocation procedures used in operations today. It is shown that this algorithm has superior overall performance in terms of efficiency and equity relative to existing procedures.

1. Introduction

One of the primary responsibilities of the Federal Aviation Administration (FAA) air traffic managers is to formulate and evaluate initiatives to alleviate anticipated demand-capacity imbalances at airports, which typically occur when adverse weather conditions prevail for a prolonged period of time. When such an imbalance is expected at an airport, traffic managers apply pre-departure delays (typically known as ground delays) to flights bound for the troubled airport commensurate with the delays they would (theoretically) receive in an airborne queue. This prevents air traffic controllers, who have limited delay options once aircraft are airborne, from being inundated with unmanageable numbers of airborne aircraft. In effect, anticipated airborne delay is transferred back to the ground, where it can be managed in a safe and orderly manner.

The processes for imposing these ground delays are embodied in a traffic flow management initiative known as a *ground delay program* (GDP). In 2005, there were over *1,350* GDPs implemented in the US averaging six hours in length, which applied delays totaling more than *16.8 million* minutes, distributed over *530,000* flights (FAA 2006). Efficient and equitable execution of GDPs is a paramount concern for air traffic management.

Second only to safety, the principal concern in planning GDPs is the maximization of throughput into the airport (Wambsganss 1996). However, preserving equity among the competing airspace users has also emerged as a fundamental performance criterion (see

Vossen et al. 2003, Vossen and Ball 2006a). Under the Collaborative Decision Making (CDM) initiative, the *ration-by-schedule* (RBS) algorithm has been accepted as the standard for equitable allocation. Under RBS, flights are prioritized based on their scheduled time of arrival when arrival slots are allocated to them (Vossen and Ball 2006b). However, RBS is not applied in its pure form. Select flights are exempted from FAA-assigned ground delay. In addition to mandatory exemptions such as airborne or (non-Canadian) international flights, discretionary exemptions are made. Flights outside a certain radius from the airport are also exempted (Ball and Lulli 2004). There are multiple motivations for this discretionary exemption policy, but the scientific basis most germane to our analysis, is mitigation of capacity uncertainty. Flights originating farther from the GDP airport must serve their ground delay well in advance of their arrival at the airport. The amount of ground delay is based on predicted capacity reductions, i.e. adverse weather conditions, several hours into the future. Overly pessimistic forecasts lead to excessive ground delay (in hindsight). By assigning greater proportion of delays to shorter-haul flights, ground delay decisions can be reactively adjusted, and the overall delays can be reduced, based on near-term weather forecasts.

In this paper, we propose a stochastic model of the assignment of ground delays in the presence of weather uncertainty and show that a new allocation principle, *ration-by-distance (RBD)*, minimizes total expected delay. RBD is based on prioritizing flights by their travel distance while allocating constrained arrival time-slots. We demonstrate that RBD in its "pure" form generates inequities. To address this problem, we propose a constrained version of RBD that preserves a specified level of equity. We show that for any chosen equity level, constrained RBD produces a more efficient GDP policy than today's GDP policy, with respect to uncertainty of GDP end time. Furthermore, we show that in our algorithm we can vary the equity level more uniformly compared to the current practice of distance-based flight exemptions. Thus, constrained RBD provides a more flexible approach to planning and controlling a GDP.

2. Background

The GDP planning and control process is currently supported by the Flight Schedule Monitor (FSM) decision support tool (FAA 2006). FSM helps traffic managers determine and assign the appropriate amount of delay for flights involved in a GDP. The operational details of the tool and its use are too involved to cover in this paper. In this section, we provide just enough detail of operational practices to convey the content and benefits of our approach to GDP planning and control.

2.1 GDP Planning

A *GDP plan* requires the assignment of ground delays to an *included* set of flights bound for a single airport with a predicted capacity-demand imbalance. The included set is defined as those flights scheduled or estimated to arrive during the GDP planning horizon. Typically, the planning horizon coincides with a weather-induced period of reduced arrival capacity. Before assigning ground delays to flights, a set of time slots are created based on the predicted reduced airport acceptance rate (AAR) – a surrogate for airport capacity. The arrival slots and their times are commensurate with the capacity reduction, e.g. for an AAR of 30 aircraft per hour, 30 two-minute slots would be defined in each hour. We associate a single time with each slot and, for convenience, use s_j to refer to both the slot itself and its associated time. Of course, the AAR may vary over time. When stochastic planning models are used, the AAR is not known in advance with certainty, so a *planned AAR (PAAR)* is defined, based on stochastic information and models. Thus, the PAAR is quite likely to differ from the realized AAR and could even differ from the AAR predicted to be most likely.

Let $f_1,\ldots,f_n$ be the set of included flights. For each flight f_k, let d_k and a_k be the scheduled departure and arrival times, respectively. (For flights operating without a published schedule, estimated departure and arrival times are used as a surrogate.) When the GDP is planned, the FAA issues a *controlled time of departure (CTD)* and a *controlled time of arrival* (CTA) for each flight, denoted by d'_k and a'_k, respectively. The CTA issued to a flight corresponds to a time slot created based on the PAAR. Logically, we can view the process as starting by assigning a flight f_k to a slot s_j (this assignment implies a'_k is set equal to s_j). Assuming that the en route travel time L_k is known and deterministic, we can then set the CTD using $d'_k = a'_k - L_k$ and the *assigned ground delay* is given by $g_k = d'_k - d_k \geq 0$. Note that the arrival time, like the departure time, is shifted by the amount of the ground delay, i.e., $a'_k = a_k + g_k$.

Once the GDP planning horizon and the PAAR are set (not a topic of this paper), the planning problem reduces to choosing each a'_k from the slot times. This process can be modeled as an assignment problem (Terrab and Odoni 1993). Thus, on a basic level, the GDP planning problem is a very simple one. However, when one considers explicitly dynamic and stochastic aspects, the problem and its analysis become considerably more complex. Also, as mentioned before, equitable allocation of constrained airport capacity among competing airlines carries high significance to both the FAA and airlines. The procedure used in practice, the distance-based RBS algorithm (DB-RBS), is strongly motivated by both equity and efficiency considerations (see Ball and Lulli 2004). Under DB-RBS, flights originating at airports farther than a specified distance from the GDP airport are exempted. The RBS algorithm is then used to assign arrival slots (and delays) to the flights remaining in the included set. More formally, the DB-RBS algorithm is defined as follows.

DB-RBS Algorithm

Step 0. Choose a radius *r* around the GDP airport. For convenience, assume *r* is in minutes of flying time (rather than miles). Mark as exempt all flights f_k with estimated en route travel time greater than *r*, that is, $L_k > r$.

Step 1. Assign each airborne and exempt flight, f_k, to the slot closest to (but no earlier than) a_k. Let *F* be the list of remaining flights and let *S* be the list of remaining slots, sorted by increasing slot time and re-indexed by $j = 1,...,m$. Mark each $f \in F$ as unassigned.

Step 2. Process the slots in *S* as follows (ration by schedule). For $j = 1,...,m$, find the unassigned flight $f_k \in F$ with the least a_k such that $a_k \le s_j$. Assign f_k to slot s_j. (If no such flight exists, leave s_j empty.) Stop when all flights have been assigned a slot. (We assume a sufficient number of slots.)

End algorithm.

The two important features of DB-RBS to observe are that flights outside the radius *r* are exempt from delay (Step 1) and that flights inside the radius *r* receive slots according to earliest scheduled arrival time (Step 2).

After the initial GDP plan is developed and the CTD's are issued, various stochastic elements invoke changes in the PAAR and, therefore, positive and negative variations in the CTD's. One of the primary challenges of GDP control is to adjust the PAAR in order to match the actual airport capacity (i.e. AAR) as it changes over time. For instance, if weather conditions improve and the AAR increases, then flights can be released earlier or the GDP can be canceled altogether. Conversely, if reduced capacity persists or worsens, then the GDP needs to be extended and flights must be given later departure times.

The dynamic aspect of concern in this paper is the ability to keep pace with random changes in the AAR is affected by prior choices of CTD's and CTA's. The dependency arises because the airborne and ground-based "inventory" of flights at any given time is determined by the CTD's executed in the past. For instance, suppose that there is a significant capacity increase relative to the PAAR and the set of assigned arrival times a'_k's. To take advantage of this additional capacity, traffic managers would like to have additional flights arrive at the airport as quickly as possible. To do this, one must tap into the ground-based inventory of flights. Now if most of the short-haul flights have already departed (under the CTD's issued to them) then this inventory will consist mainly of long-haul flights. However, these long-haul flights will take considerable time to arrive at the airport thus leaving the additional capacity largely unused for a significant period of time. Of course, flight stage lengths do not fall neatly into two categories; we are merely highlighting a basic principle of inventory control.

The primary goal of this paper is to develop a strategy for GDP planning that allocates arrival slots among the included set of flights (i.e. assigns them CTA's and CTD's) so that in case of early termination of the program, the newly available AAR capacity can be used as efficiently as possible. Before proceeding to our proposed solution, we review some of the relevant research that addresses the problem of choosing various parameters of a GDP in the face of uncertainty.

2.2 Literature on the Stochastic Ground Holding Problem

GDP planning falls within the larger domain of air traffic flow management (Ball et al 2007). The problem of assigning ground delays to aircraft subject to airport capacity constraints in order to minimize a certain objective function, which is typically the expected total cost of both ground and airborne delays, is known as the ground holding problem (GHP; Odoni 1987). Airborne delays can result from situations where the actual flow of aircraft toward an airport's airspace exceeds the realized AAR. A particular variant of the GHP, known as the single airport ground holding problem (SAGHP), focuses on allocating arrival capacity (i.e., slots) at a single airport ignoring other constraints in the airspace. Richetta and Odoni (1993) were among the first to propose a stochastic linear integer programming (IP) formulation of the SAGHP that modeled AAR uncertainty. The uncertainty is represented by a finite set of scenarios of airport capacity, each with a specified probability of occurrence. Recent research has also addressed the problem of estimating capacity scenarios, which represent the time-varying profile of an AAR. Scenarios and their probabilities can be constructed from weather forecasts and/or by analyzing historical AAR evolution at an airport, e.g. see (Inniss and Ball 2004), (Liu et al. 2006) and (Wilson 2004). Ball et al. (2003)

proposed an aggregate version of the Richetta and Odoni model that directly sets the PAAR without assigning delays to specific flights. This model was designed for use in a CDM context where other processes that take equity into account and that accept airline preferences to determine the actual flight-to-slot assignment. The underlying IP has a constraint matrix that is "dual network" allowing fast solution by network flow methods. See Ahuja et al. (1993) for background. Kotnyek and Richetta (2006) extended both Richetta and Odoni results and the Ball et al. results by showing that a flight-to-slot stochastic IP with *marginally increasing* ground delay costs could be solved as a linear program. Further, the resulting flight-to-slot assignments were consistent with the CDM-based first-scheduled-first-served equity principle. All of these models can be used to determine a PAAR in face of uncertainty in the AAR. Still, these CDM-compatible models require a slot allocation procedure (such as the one we propose) to set flight-specific arrival and departure times.

Richetta and Odoni (1994) and Mukherjee and Hansen (2007) proposed models that simultaneously decide on the PAAR and slot assignment to individual (or sometimes groups of) aircraft. These are based on multi-stage stochastic programs (see Birge and Louveaux (1997) for background), where ground holding strategies can be revised based on updated forecasts. In addition to capacity scenarios, these models require as input a decision tree, whose branching points and branches reflect changing AAR's. A major challenge in using these models in practice, as highlighted by Liu et al. (2006), is the development of the required *scenario trees*. Further complicating the applicability is the fact that not only capacity scenarios, but also their branching points in time, must be predicted in advance, and provided as input to the models. Another limitation of the dynamic models, as acknowledged by Mukherjee and Hansen (2007), is that the decisions made based on a particular set of scenarios, provided as input, may no longer be optimal if the set of possible AAR profiles (i.e., scenarios) themselves change with time. Nevertheless, the dynamic models show significant improvement over the static models (Ball at al. 2003, Inniss and Ball 2004, Richetta and Odoni 1993) by adapting updated capacity forecasts into the decision making process. In light of the shortcomings of the scenario-based planning models, Liu and Hansen (2007) proposed a *scenario-free* sequential decision making problem, based on dynamic programming techniques, for the stochastic SAGHP. In order to reduce computational complexity associated with large-scale problems, they also proposed several heuristics.

All of the above models address uncertainty in the AAR, which is an instantiation of the general phenomenon of *capacity uncertainty* (see Ball et al (2007) for background). There has been a limited research, e.g. see Ball et al (2001) and Willemain (2004), directed at GDP planning under *demand uncertainty*. Demand uncertainty in this context refers to random deviations between the planned and actually arrival times of fights at the destination airport.

3. Research Contributions

In this paper, we propose an alternative to today's DB-RBS algorithm. Our algorithm, which we present in Section 5, employs the ration-by-distance principle together within a procedure that guarantees a specified level of equity. Our experimental results, presented in Section 6, show that this new algorithm outperforms the DB-RBS on both fronts – equity and efficiency. Thus, this is strong evidence of the practical usefulness of this approach.

On the theoretical side, we prove in Section 4 that a prioritization strategy based on scheduled flying times, which we term as *ration-by-distance* (RBD), minimizes total delay in the event the GDP cancels earlier than planned. We show that this result holds under two different models of GDP dynamics, which we argue cover a broad range of practical GDP scenarios. These results imply that the "pure" ration-by-distance approach minimizes expected delay *under all GDP cancellation time distributions*. We believe that this is a very strong, and perhaps somewhat surprising, property of this simple allocation procedure.

4. The Ration-by-Distance Algorithm and its Properties

As was discussed in Section 2.1 there is a higher level of flexibility associated with skewing ground delays to flights with shorter flying time. In fact, this intuition motivated the development and use of DB-RBS. This increase in flexibility can be viewed as a way of mitigating potential costs of weather uncertainty. In the real application of GDP's, the amount of ground delay taken by a flight is a random variable. Specifically, if the weather clears earlier than expected and the GDP is canceled, then many flights are allowed to depart earlier than their initially assigned CTD's. Further applying the logic behind DB-RBS, one might suspect that by prioritizing the allocation of ground delay based on distance (flight time), one could achieve good, or perhaps the best, performance under a stochastic GDP model. We now present such an approach, the ration-by-distance (RBD) algorithm for assigning flights to slots during a GDP.

RBD Algorithm

Step 1. Assign airborne flights to slots as in Step 1 of the DB-RBS algorithm. As in DB-RBS, define *S* to be the remaining slots and order these slots by increasing slot time.

Step 2. Process the remaining slots in *S* as follows. For $j = 1,...,m$, find the unassigned flight, f_k, with the largest flying time L_k such that $a_k \le s_j$. Assign f_k to slot s_j. That is, set $a'_k = s_j$. (If no such flight exists, leave s_j empty.)

End algorithm.

RBD differs from DB-RBS in that only airborne flights are exempted and the priority rule used in Step 2 changes from smallest a_k to largest L_k.

In Sections 4.1 and 4.2, we will show that under certain stochastic GDP models, RBD produces a GDP plan that minimizes total expected delay. To prove this result we must precisely define models of GDP dynamics and certain GDP cancellation policies. While these models certainly involve a set of assumptions, we argue later that they cover a very broad class of practical GDP instances.

We begin our theoretical results by abstracting relevant GDP dynamics. In the event of early GDP cancellation, the CTD's (and therefore CTA's) of flights not yet departed can be decreased. The set of flights and the amount of delay reduction depends on the time the GDP is canceled, T_c. Although T_c, is at the discretion of the FAA traffic managers, it directly depends on changes in weather conditions. Therefore, it can be modeled as a random variable. We associate a discrete probability distribution with the cancellation time $p_t = \Pr[T_c = t]$. In general, this distribution depends on weather characteristics. We see then that the actual arrival time of a flight and the ground delay that it absorbs are random variables that depend on T_c. However, these random variables also depend on the manner in which newly created arrival slots are allocated. Thus, to formulate a precise stochastic model we must prescribe how allocation is performed when capacity increases. We do this by defining two *GDP Cancellation Policies* (CP's), identified as CP1 and CP2. (We discuss a broader set of GDP dynamics in Section 4.3.)

CP1: At the time the GDP is cancelled, all flights that are currently being held on the ground are allowed to depart without any further delay. Further, it is assumed that when these flights arrive at the airport they are able to land without any additional (airborne) delay.

Of the surface, this policy would seem to assume that after a GDP is cancelled arrival capacity is unlimited. However, we have observed that the natural spread of flight times and schedules allows traffic managers to use CP1 quite extensively in practice.

CP2: At the time the GDP is canceled, the existing slot set is augmented and a priority rule, based on a flight's current CTA, is used to reassign each flight to a potentially earlier slot. Further, it is assumed that there are no further changes in airport capacity so when these flights arrive at the airport they are able to land without any additional (airborne) delay.

Sections 4.1 and 4.2 treat policies CP1 and CP2, respectively. In each case, the proof makes use of an elementary slot exchange we call *a long-short (LS) swap*. Given any allocation of flights to slots, a LS swap is an exchange of the assigned slots between two flights, f_1 and f_2, such that $L_1 \geq L_2$, $a'_1 > a'_2$ and $a_1, a_2 \leq a'_2$. In other words, in the initial assignment, a longer-haul flight (f_1) has been assigned to a slot (a'_1) that is later than the slot (a'_2) to which a shorter-haul flight (f_2) has been assigned. An LS swap reverses this assignment.

4.1. Analysis of CP1

In this section, we prove that under CP1 RBD produces a solution that minimizes total expected delay. We start by defining for each flight k the random variable $A_k(i,t)$ as the arrival time (slot) assigned to flight k with $T_c = t$ and $a'_k = i$. We also define the random variable $G_k(i,t)$ to be the corresponding ground delay faced by that flight. Now, it is easy to see (via Figure 1) that $A_k(i,t)$ and $G_k(i,t)$ are given by the following expressions:

$$A_k(i,t) = Min\{i, Max\{t + L_k, a_k\}\} \tag{1}$$

$$G_k(i,t) = A_k(i,t) - a_k = Min\{i, Max\{t + L_k, a_k\}\} - a_k \tag{2}$$

Under the CP1 assumption, there is no airborne delay so the efficiency metric of interest is total ground delay. We define $GT(t)$ as the total ground delay incurred with $T_c = t$. Of course, $GT(t)$ depends on the GDP plan so that $GT(t) = \sum_k G_k(a'_k, t)$. A GDP plan that maximizes expected efficiency would be a set of valid a'_k variables that minimizes total expected ground delay, GT, which is computed via $GT = \sum_t p_t GT(t)$.

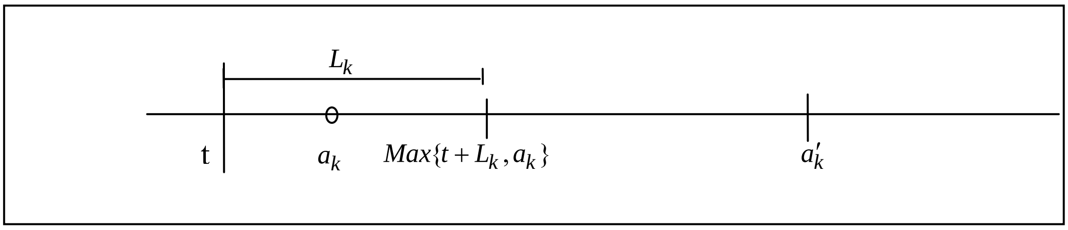


Figure 1: Arrival time of a flight for a given GDP cancellation time under CP1.

To prove our main results we need the following elementary inequality.

Lemma 1: *Let* u_1, u_2, v_1, v_2 *be real numbers with* $u_1 \le u_2$ *and* $v_1 \le v_2$. *Then we have the following inequality:*

$\min(u_1, v_2) + \min(u_2, v_1) \le \min(u_1, v_1) + \min(u_2, v_2)$.

Proof: Suppose that $u_1 \le v_1$. By hypothesis, we have $u_1 \le v_1 \le v_2$. Therefore, $\min(u_1, v_2) = \min(u_1, v_1)$. Since $v_1 \le v_2$, we also have that $\min(u_2, v_1) \le \min(u_2, v_2)$, thus proving the above inequality in this case. Now suppose that $u_1 \ge v_1$. The other case $u_1 > v_1$ can be shown by reversing the role of u 's and v 's.■

The next proposition shows that an LS swap cannot increase total delay.

Proposition 1: *Given two flights f*$_1$ *and f*$_2$*, with* $L_1 \ge L_2, i < j$ *and* $a_1, a_2 \le i$ *, then under CP1 delay is always minimized by assigning the longer flight, f*$_1$*, to the earlier slot, i.e.* $G_1(i,t) + G_2(j,t) \le G_1(j,t) + G_2(i,t)$.

Proof: From equation (2) we have:

$$G_1(i,t) + G_2(j,t) = \min\{i, \max\{t + L_1, a_1\}\} - a_1 + \min\{j, \max\{t + L_2, a_2\}\} - a_2 \quad (3)$$

Case 1: $t + L_2 \ge a_2$. In this case equation (3) can be rewritten as:

$$G_1(i,t) + G_2(j,t) = \min\{i, \max\{t + L_1, a_1\}\} - a_1 + \min\{j, t + L_2\} - a_2 \quad (4)$$

Since $L_1 \ge L_2$, $t + L_1 \ge t + L_2$ and it follows that $\max\{t + L_1, a_1\} \ge t + L_2$. Thus, equation (4) has the following form:

$\min\{u_1, v_2\} + \min\{u_2, v_1\} - a_1 - a_2$, where $u_1 \le u_2$ and $v_1 \le v_2$. It follows from the Lemma 1 that

$\min\{i, \max\{t + L_1, a_1\}\} + \min\{j, t + L_2\} \le \min\{j, \max\{t + L_1, a_1\}\} + \min\{i, t + L_2\}$, which completes the proof for this case.

Case 2: $t + L_2 < a_2$. In this case, equation (3) can be rewritten as:

$$G_1(i,t) + G_2(j,t) = \min\{i, \max\{t + L_1, a_1\}\} - a_1 + \min\{j, a_2\} - a_2 \quad (5)$$

By hypothesis, $a_2 < j$ so the last two terms of equation (4) sum to zero. Also by hypothesis, $i \le j$, so it easily follows that $\min\{i, \max\{t + L_1, a_1\}\} - a_1 \le \min\{j, \max\{t + L_1, a_1\}\} - a_1$, which completes the proof for case 2. ■

Figure 2 shows slot assignments for two flights before (top) and after (bottom) an LS swap. Given the cancellation time t as shown, both flights would serve full ground delay without the LS swap. However, if the LS swap were made, then flight f1 would recoup some of its ground delay. Figure 3 shows the amount of this savings Δ as a function of the cancellation time t. It is easy to verify that in general, the maximum delay saving from an LS swap under CP1 is given by $Min\{i - j, L_1 - L_2\}$.

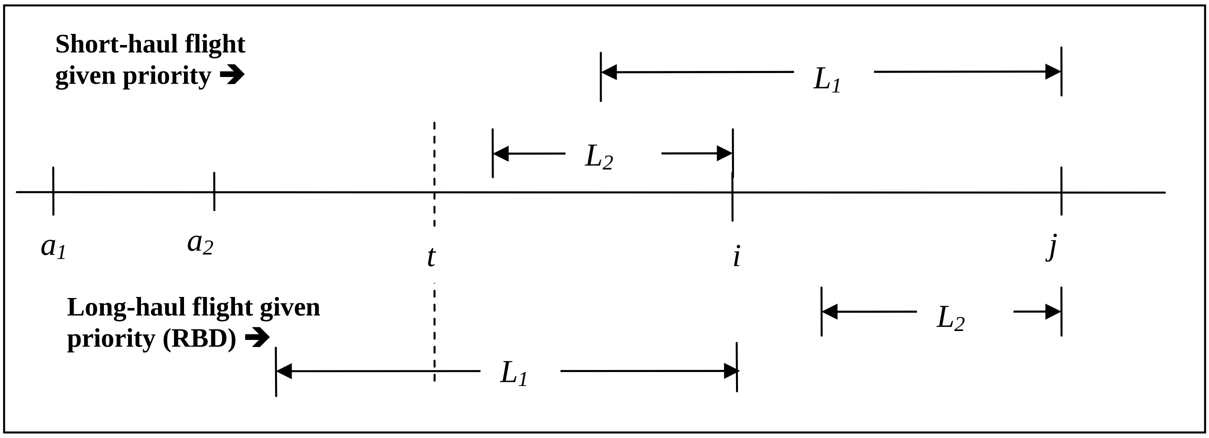


Figure 2: Example of a LS Swap (see Figure 3 for the resulting delay saving)

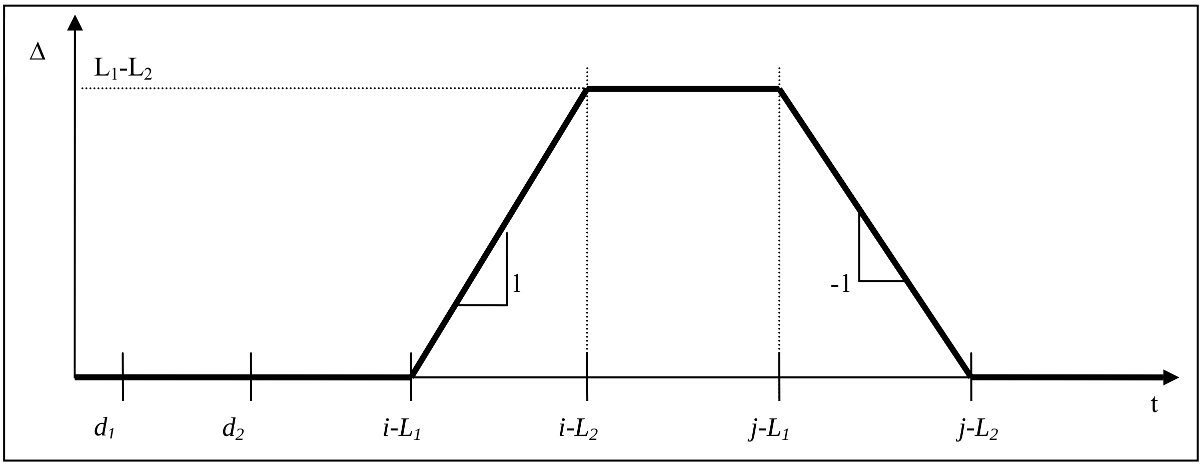


Figure 3: Delay savings from a LS swap as a function of GDP cancellation time.

We can now give the basic result concerning RBD.

Theorem 1: *Let* $\{a'_k*\}$*, the set of CTA's output by RBD, be used to compute* $GT(t)$*. Then,*

(i) for any given cancellation time t, $GT(t)$ *achieves its minimum value;*

(ii) for any given cancellation time distribution $\{p_t\}, GT$ *achieves its minimum value.*

Proof: We prove (i) by contradiction and, thus initially assume it is not true. Then there is an optimal CTA assignment, *AS**, different from the RBD assignment that has a smaller value of $GT(t)$. Proceeding from the earliest to the latest slot, consider the associated ordered list of flights in *AS**. Consider a similar list for the RBD solution. Since the solutions are different, there will be an earliest slot where they differ in flight assignment. Let f_2 be the flight that occupies that slot in *AS** and let f_1 be the flight that occupies it in the RBD solution. By the prioritization imposed by RBD, $L_1 \geq L_2$. Further, in *AS**, f_1 must be assigned to a later slot (otherwise, the first slot where the solutions differed would have occurred earlier). It follows that f_1 and f_2 can be interchanged in the optimal solution. Moreover, this is an LS swap so that by Proposition 1, it will leave the value $GT(t)$ the same or reduced. We can continue this process until the two solutions are the same. If at any step $GT(t)$ is reduced, we contradict the assumed optimality of *AS**. Otherwise, the process ends with a proof that the RBD solution has the same value of $GT(t)$ as *AS**, also a contradiction, thereby proving part (*i*).

Part (*ii*) follows directly from part (*i*) since the $GT(t)$ are the variable coefficients in the expression for GT. ■

4.2. Analysis of CP2

Under CP2, each flight requires a specific slot reassignment when the GDP is cancelled. To analyze CP2, we must develop and review certain results relating to the general problem of assigning flights to slots (see Vossen and Ball (2006a) and Ball et al (2007) for more details). Let S be a set of slots and F a set of flights. In addition to its scheduled arrival time a_k, each flight f_k has an earliest slot time $e_k \geq a_k$. In many cases, e_k equals a_k, but other factors sometimes force a difference. A feasible assignment of flights to slots is the association of a slot (index) $\sigma(k)$ with each flight f_k such that no two flights are assigned the same slot and $s_{\sigma(k)} \geq e_k$. In the subsequent analysis, we assume that the flights, slots and e_k values allow at least one feasible assignment of flights to slots. The delay for flight f_k is given by $s_{\sigma(k)}$ - a_k Associated with any given assignment is a set of *covered slots*, which are those slots with an assigned flight. Given a feasible assignment and its associated set of covered slots, *S'*, we note that:

$$\textit{Total flight delay} = \Sigma_k (s_{\sigma(k)} - a_k) = \Sigma_{j \in S'} s_j - \Sigma_k a_k. \quad (6)$$

The significance of this equation is that total flight delay depends only on *F* and the set of covered slots, *S'*.

We will now describe a simple priority-based algorithm for assigning flights to slots. The algorithm requires a priority rule, which is simply a way to order the flights.

REASSIGN

Step 1: Order the set of flights F based on the priority rule.

Step 2: For each $f \in F$ in order, permanently assign f to the earliest available slot s_j with $s_j \geq e_k$.

End Algorithm

We start by providing two basic properties of REASSIGN.

Proposition 2: Given a set of slots *S,* a set of flights *F,* for each flight $f_k \in F$, let a^o_k be its arrival time in some feasible assignment. Suppose that REASSIGN is applied to *S* and *F* where flights are ordered by increasing value of a^o_k. Then, REASSIGN will find a feasible assignment and the arrival time assigned to each flight will be less than or equal to a^o_k.

Proof: Assume that the flights are numbered according to the priority used (increasing value of a^o_k). Thus, f_1 is considered first and it can be assigned to the slot with time a^o_1 so the time it will receive is no greater than a^o_1 . Proceeding inductively it can be seen that when f_k is considered only slots with times less than a^o_k will have been assigned so the slot with time a^o_k will be available and the time assigned to f_k will be no greater than a^o_k . ■

Proposition 3: *Given a set of slots S, a set of flights F, REASSIGN will always produce the same set of covered slots (and total delay) no matter which priority rule is used.*

Proof: We prove this result by contradiction. Suppose that under two different priority rules REASSIGN generated two different final assignments, A_1 and A_2 and that A_1 and A_2 had two different sets of covered slots. The set of covered slots produced by A_1 and A_2 will match up to a certain point so that there will be an earliest slot there they do not match, i.e. an earliest slot s' that is not covered by one assignment, say A_1, that is covered by the other, say A_2. Denote by h the number of slots covered by A_2 that are earlier than s'. These h slots together with s' are covered by $h+1$ flights under assignment A_2. But under A_1, there are h flights covering slots earlier than s'. Thus, there must be at least one flight, f_k, that under A_1 is assigned to a slot later than s' (or assigned to no slot) that is assigned to a slot s' or earlier under A_2. But, it must be that the case that $e_k \leq s'$ so REASSIGN could have assigned f_k to s' but in A_1 but did not, which contradicts Step 2 of REASSIGN. Thus, the set of covered slots does not vary with the priority rule, and by applying (6) we see that total delay is constant. ■

Note that together, Propositions 2 and 3 imply that REASSIGN will always find a feasible solution if one exists and will always produce the same total delay independent of the priority rule used. These two properties now lead directly to another useful one.

Proposition 4: *Given a flight set F, slot set S, and set of earliest arrival times e_k, let TD be the total delay produced by REASSIGN. Let TD' by the total delay produced by REASSIGN when each e_k is replaced with a value $e'_k \leq e_k$. Then TD' $\leq$ TD.*

Proof: Let s_k be the slot time assigned to f_k when REASSIGN was applied to F and S with earliest flight assignment times e_k under policy P, and let TD be the total delay. Now consider the application of REASSIGN to F and S, with earliest flight assignment times e'_k, where the priority rule P' used is increasing value of s_k. By Proposition 2, the time assigned to flight f_k will be no greater than s_k, which immediately implies that $TD' \leq TD$. By Proposition 3, TD' does not vary with the policy P' that is used, so the result follows. ■

Having established properties of REASSIGN, we can more clearly state how CP2 is implemented. Recall that under CP2, when the GDP is canceled, an explicit assignment of flights to arrival slots is carried out to ease out of the program (as opposed to releasing all flights as in CP1). Our assumption is that under CP2, REASSIGN is used to find a new (possibly earlier) slot for each flight f_k. To facilitate our analysis we assume that REASSIGN is applied to all flights, both those airborne and those on the ground. By restricting the priority rule so that airborne flights are given highest priority, REASSIGN will assign each airborne flight to its original ETA, a'_k, so that airborne flights will not be assigned any additional delay (as is the standard policy). It can easily be seen that the earliest arrival time for flights on the ground at time t should be $e_k = Min\{a'_k, Max\{t + L_k, a_k\}\}$. On the other hand, for airborne flights we set $e_k = a'_k$.

The key property for CP2, which is analogous to Proposition 1, is given by Proposition 5. Its proof requires Lemma 2, which is a slight generalization of Lemma 1.

Lemma 2: *Let* u_1, u_2, v_1, v_2 *be real numbers such that* $u_1 \le u_2$ *and* $v_1 \le v_2$*. Then each of the sets* $\{s \in S : s \ge \min(u_1, v_2)\}$ *and* $\{s \in S : s \ge \min(u_2, v_1)\}$ *forms a superset of at least one of the following sets:* $\{s \in S : s \ge \min(u_1, v_1)\}$ *and* $\{s \in S : s \ge \min(u_2, v_2)\}$.

Proof: Analogous to Lemma 1. ■

Proposition 5: *If CP2 is applied before and after an LS swap, then for any cancellation time, t, the delay obtained after the LS swap is no more than the delay obtained before.*

Proof: We prove this Proposition, by showing that the net effect of an LS swap is to revise the earliest arrival times so that in aggregate Proposition 4 applies and, therefore, delay cannot increase. As discussed above, in this analysis we ignore the difference between airborne and non-airborne flights but designate that airborne flights have $e_k = a'_k$.

As before, we define the parameters associated with the LS swap: let f_1 and f_2, such that $L_1 > L_2$, $a_1' > a_2'$ and $a_1, a_2 \le a_2'$. Also, as before, let $a_2' = i$ and $a_1' = j$.

Case 1: $t + L_2 \ge a_2$. Since $L_1 > L_2$, it follows that $t + L_1 > t + L_2$ and hence $Max\{t + L_1, a_1\} \ge t + L_2$. Therefore, before L-S swap, we have $e_1 = Min\{j, Max\{t + L_1, a_1\}$ and $e_2 = Min\{i, t + L_2\}$. After the L-S swap, we have $e_1 = Min\{i, Max\{t + L_1, a_1\}$ and $e_2 = Min\{j, t + L_2\}$. By Lemma 2 (letting $u_1 = i, u_2 = j, v_1 = t + L_2, v_2 = \max(t + L_1, a_1)$) the feasible sets formed after LS swap are supersets of those before the swap. It follows from Proposition 3 that the delay obtained after the LS swap is no greater than that before.

Case 2: $t + L_2 < a_2$. Before the LS swap we have $e_1 = Min\{j, Max\{t + L_1, a_1\}$ and $e_2 = Min\{i, a_2\}$ and after the LS swap we have $e_1 = Min\{i, Max\{t + L_1, a_1\}$ and $e_2 = Min\{j, a_2\}$. Now, if $Max\{t + L_1, a_1\} \ge a_2$, then the same logic applies as in Case 1 above. On the other hand, if $Max\{t + L_1, a_1\} < a_2$, then by assumption, $Max\{t + L_1, a_1\} < a_2 < i < j$. Therefore the feasible sets before and after the LS swap are identical in this case, which completes the proof for Case 2. ■

As in the case of CP1, we now can derive the desired property.

Theorem 2: *Using cancellation policy, CP2, the solution produced by RBD minimizes total delay for each cancellation time t and hence minimizes total expected delay.*

Proof: Proof is similar to that of Theorem 1. ■

It is important to note that under CP2, it is possible (at least in somewhat "pathological cases") for CP2 to assign a flight a revised slot time that is actually later than its original CTA, a'_k. The following proposition shows that under a very reasonable priority rule, this cannot occur.

Proposition 6: *If the priority rule used in within REASSIGN under policy CP2 is according to increasing value of a'_k, then each flight will be assigned a slot time no later than a'_k after a GDP is cancelled.*

Proof: This follows directly from Proposition 2. ■

4.3. GDP Extension

An early cancellation is not the only possible deviation from an original GDP plan that can occur. Sometimes, weather does not clear as forecasted causing a longer than anticipated period of reduced capacity. Weather severity may also cause an unanticipated drop in the AAR causing the PAAR to exceed the (actual) AAR. In some cases, GDP planners do not make any changes in assigned ground delay and the situation resolves itself through forced airborne holding. However, when the problem is severe enough the GDP is extended. A program extension occurs when the planned end time is pushed further into the future relative to the originally planned end time. Most often, the PAAR remains the same as the value specified for the original program. An extension largely involves the assignment of CTA's to flights that originally were scheduled to arrive after the planned end time. In fact, since these flights had previously received no assigned delays, the ability to assign delays to them is totally unaffected by the choice between DB-RBS and RBD. Additionally, flights that had received CTA's are sometimes assigned additional delay. The choice between DB-RBS and RBD does impact which, and how many, of these flights are on the ground when a decision to extend is made (a flight must be on the ground in order to receive additional ground delay).

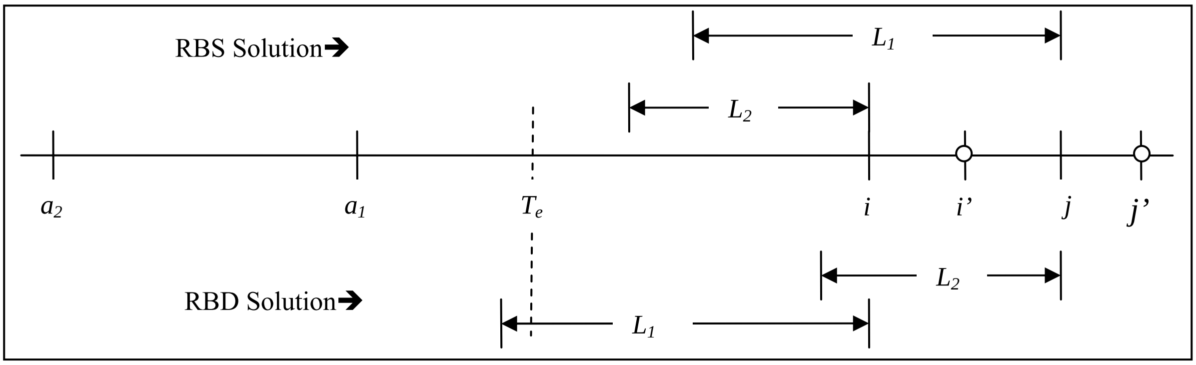


Figure 4: RBD vs. RBS slot availability under GDP Extension

Figure 4 demonstrates the difference between the two extreme solutions (RBD and RBS) in case of a GDP extension. The results depend on when the GDP extension notice is served, i.e. the time T_e at which it becomes known that the GDP will be extended beyond the initially planned end time. Let i' and j' be the new slots after the GDP extends, available for flights that were initially assigned to i and j respectively. If $T_e < i\text{-}L_1$ then under both RBD and RBS the flights are reassigned to their new slots i' and j', and therefore the two algorithms produce the same results. Similarly, it is easy to see that when $i\text{-}L_2 < T_e < j\text{-}L_1$, or when $T_e > j\text{-}L_2$, both RBS and RBD yields the same amount of airborne holding. The cases when the two algorithms yield different results are as follows. When $i\text{-}L_1 < T_e < i\text{-}L_2$ airborne delay in the amount of $i' - i$ is faced under RBD, whereas under RBS, this delay is converted to ground holding. On the other hand, when $j\text{-}L_1 < T_e < j\text{-}L_2$ RBS results in an amount of airborne delay of $j' - j$, over and above any encountered under RBD. Thus, when a GDP is extended, airborne holding may be necessary. However, depending on T_e, RBD and RBS may perform equally well, while there are cases where one outperforms the other. Further insight into the relative performance of RBD and RBS under extensions is provided in Section 6 when we give the results of related computational experiments.

5. Equity Considerations and a Practical Approach

The results of Section 4 indicate that RBD optimizes a system-wide efficiency metric but says nothing about equity, the second important performance criterion. Consider a situation where a flight with one of the smallest value of L_k has a scheduled arrival time early within a 4-hour program. Such a flight will have the lowest priority throughout the allocation process and will very likely receive a very long delay, e.g. close to four hours. In practice, such a situation would certainly be deemed highly inequitable by carriers. Equity is of great importance and motivated us to create a more practical constrained version of RBD.

To effectively address equity considerations, we define an equity metric and associated constraint to maintain equity within the RBD process. We follow the approach of Vossen et al. (2003), where the inequity of a given allocation is defined as its deviation from an ideal allocation, which we define as the pure RBS allocation, meaning RBS without any discretionary exemptions. Vossen et al. (2003) and Vossen and Ball (2006a) provide strong justification for this choice. For each flight f_k, let a''_k, denote the RBS slot assignment. Our equity metric is defined as the maximum positive deviation between any flight's assigned slot and its RBS slot.

We now define a constrained version of RBD that enforces an upper bound – δ – on this inequity metric (i.e., $Max_k(a'_k - a''_k) \le \delta$). We call this allocation method *equity-based RBD (E-RBD).* Enforcing this upper bound requires some significant changes to RBD. Rather than directly creating an allocation, E-RBD first gives each flight a temporary slot assignment based on the application of RBS. A set of assignment exchanges is executed where each assignment-exchange gives a permanent slot assignment to one flight and adjusts the temporary assignments of one or more others. The assignment-exchanges are identified by choosing flights in order of decreasing value of stage length (L_k) and then executing an operation that assigns the chosen flight the earliest feasible slot. Before describing the algorithm it is useful to illustrate the assignment/exchange operation.

The most elementary form of the assignment-exchange moves the identified flight to an earlier slot and then "bumps forward" each of the intermediate flights to "make room" for the move. Such an operation is illustrated in Figure 5A. In this example, the delay on the targeted flight f_4, is reduced by the width of 3 slots and the delay on each of the 3 intermediate flights (f_1, f_2, f_3) is increased by the width of 1 slot. This exchange would give f_4 a permanent slot assignment and adjust the temporary slot assignments of the other 3 flights. Figure 5B illustrates a more complex assignment-exchange. In this case, the existing assignment of one of the intermediate flights, f_2, is permanent. Therefore, adjustments to the temporary assignment of f_1 must take this into account. In this case the delay of f_1 increases by the width of 2 slots. In general, we define an f_k*-to-*s_j *assignment/exchange* as the operation that removes f_k from its currently assigned slot, reassigns it to an earlier slot s_j and simultaneously makes the minimal reassignments to the intermediate flights in a way consistent with Figure 5. We say that an f_k-to-s_j assignment/exchange is *δ-feasible* provided that: (a) the current flight assignment for s_j is temporary, (b) $a_k \le s_j$, and (c) none of the delay increases to the intermediate flights induces a violation to the equity constraint with a right hand side of δ. We can now define E-RBD.

E-RBD Algorithm

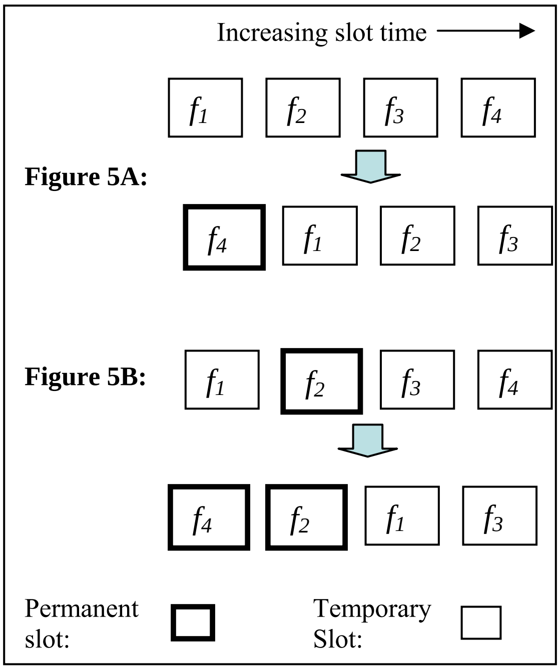
***Step* 0**. Choose an equity deviation limit δ.

***Step* 1.** Assign each airborne flight, f_k, to the slot closest to (but no later than) a_k and remove these flights and slots from the respective lists. For each of the remaining *m* included flights, f_k, give f_k a temporary slot assignment by setting a'_k to its (unconstrained) RBS slot.

***Step* 2.** Order the *m* included flights by decreasing value of L_k. For $k = 1,\ldots,m$:
find the earliest slot s_j such that the f_k-to-s_j assignment/exchange is δ-feasible;
execute this exchange and permanently assign f_k to s_j.

***End Algorithm*.**

It should be noted that in the later executions of Step 2, it will typically be the case that the earliest slot identified will be the one to which the flight is temporarily assigned. Thus, in such cases, the net effect of the Step 2 iteration will be to make the existing temporary assignment permanent. Of course, the early executions of Step 2 will implement the types of operations illustrated in Figure 5.



Figures 5A and 5B: Examples of Assignment Exchange

6. Performance Comparison of Algorithms

We conducted a set of experiments to gain insight in the differences between the three rationing policies: DB-RBS, RBD and E-RBD.

6.1 Test Data and Scenarios

We constructed a test data set based on demand data from the FAA's Aviation System Performance Metrics (ASPM) database for San Francisco International airport (SFO) on August 11, 2005. Our scenario mimics a typical SFO morning GDP induced by the late burn off of marine stratus. The marine stratus conditions effectively eliminate closely-spaced parallel approach, thereby reducing the AAR from approximately 60 flights per hour to 30 flights per hour. Our demand data exceeded 30 flights per hour for three consecutive hours, from 9:00 to 12:00 local time. To accommodate this imbalance, we planned a 4-hour GDP from 9:00 to 13:00. The control times in the 12:00 hour were necessary to accommodate the pent-up demand from earlier hours. We evaluated five GDP cancellation times, one for the top of each hour during the program (9:00, 10:00,…, 13:00). Each of the three algorithms – DB-RBS, RBD, and E-RBD -- was evaluated under the five cancellation scenarios.

6.2. Experimental Results

Figure 6 shows the equity and efficiency evaluation of DB-RBS. The horizontal axis specifies the exemption distances (for DB-RBS) in nautical miles, with higher values (i.e.

smaller number of exempt flights) to the left. The vertical axis has two scales, the left one being for efficiency, measured as total minutes of delay, and the right vertical axis being for equity, measured as maximum deviation from the RBS allocation over all flights. The right vertical axis gives the scale for the bars on the graph, which measure of equity. Note that the equity deviation ranges from nearly zero minutes (leftmost bar) to just over 160 minutes (rightmost bar). The dramatic increase in deviation (inequity) is to be expected, since as exemption distance increases, the total amount of "required" delay must be absorbed by a shrinking pool of (non-exempt) flights. The rate of decline is essentially quadratic (for reasons we don't entirely understand). The plateaus correspond to ranges of distance in which no new flights bound for the GDP airport are encountered.

The efficiency of DB-RBS is measured by the five line plots – one for each cancellation time (denoted as "Cnx" in the figure). Note that these are vertically stacked with the latest cancellation time being on top. This means that for any fixed exemption distance, the delay minutes drop each time the program is cancelled earlier. This makes intuitive sense, since more flights can be released earlier than their controlled times, hence reducing the total amount of delay.

Scanning any of the line plots left to right, we see that as the distance parameter decreases (thereby exempting longer-haul flights), the total delay decreases also, or levels off, in a nearly linear manner. This phenomenon confirms the fundamental principle of ration-by-distance: delay can be saved under early GDP cancellation by assigning a greater proportion of delays to the short haul flights as compared to long haul flights; the earlier the cancellation time, the greater the savings. Note also the equity-efficiency tradeoff in Figure 6: as equity goes up, efficiency goes down. This is the tradeoff associated with the distance parameter. It becomes more pronounced for earlier cancellation times.

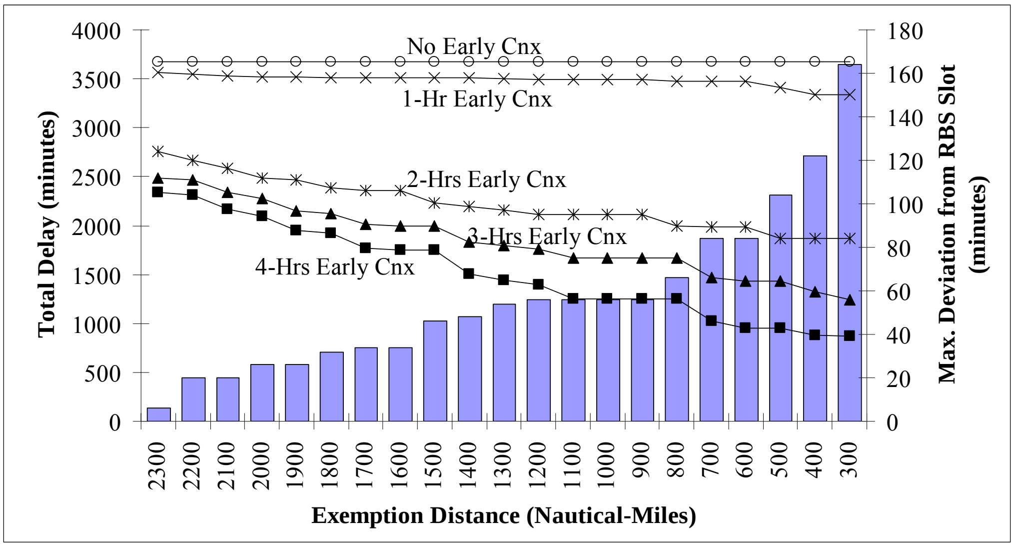


Figure 6: Performance Statistics for Distance-based Ration-by-Schedule (DB-RBS) Algorithm

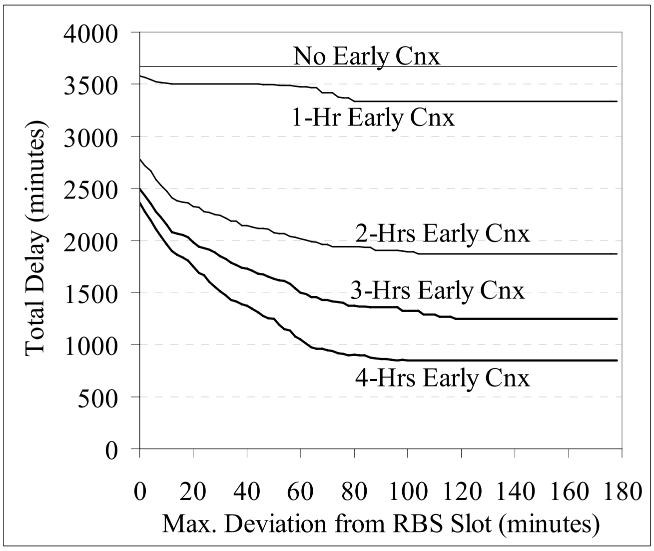


Figure 7: Performance of E-RBD

Figure 7 shows the performance of E-RBD: the horizontal axis is the maximum deviation parameter, δ, while the vertical axis is the total delay, in minutes. We did not 'plot' the equity of E-RBD (as we did in Figure 2) because, by our metric, it is directly controlled by parameter δ. The five line plots in Figure 7 correspond to the five possible GDP cancellation times. Note that the total delay decreases as δ increases. This is because higher values of δ increase the number of feasible f_k-to-s_j assignment/exchanges. Also, as the program cancellation time moves earlier, the total delay decreases. The line plots in Figure 7 are very much like those of Figure 6, with two important exceptions: in Figure 7, they are nearly quadratic, while in Figure 6, they were nearly linear. (This is because the horizontal axis of Figure 3 is a quadratic translation of the horizontal axis of Figure 2.) Also, the total delay resulting from E-RBD is generally lower than that of DB-RBS (more on this later). Figure 7 shows the same equity-efficiency tradeoff that we saw in Figure 6.

We do not have a separate figure to evaluate the performance of RBD, because its performance is imbedded in Figure 7 as an extreme case of E-RBD. The total delay resulting from RBD can be found at the far right of each cancellation-time line plot.

It is instructive to further compare the performance of RBS, RBD and E-RBD from another equity perspective. In Figure 8, we show the sum over all flights, the squared deviation from RBS under four different allocation strategies. Thus, the units are minutes squared. For E-RBD, we chose two representative values of the δ parameter: 20 minutes and 80 minutes. (These are values where the marginal decrease in delay became small as δ increased further.)

Figure 8 yields several interesting results. First, when the program is not cancelled, RBS has perfect equity (zero deviation). But when the program is cancelled early, RBS deviates from perfect equity because the RBS allocation is based on the planned 4-hour AAR reduction, and not the AAR that results from early cancellation of the GDP.

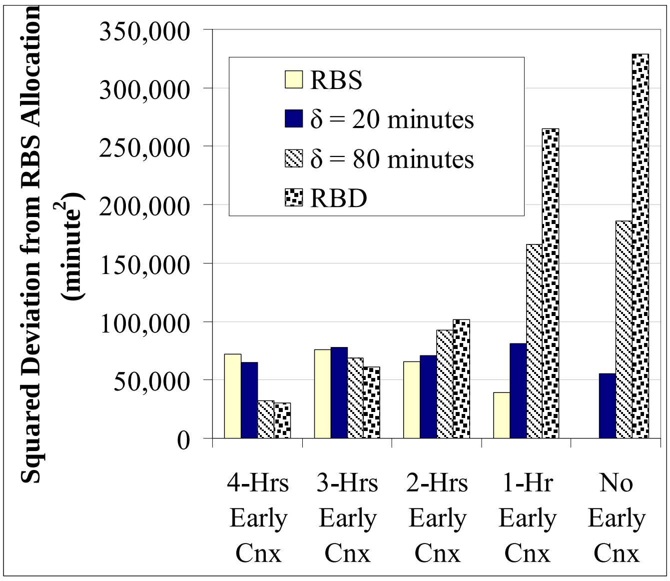


Figure 8: Equity Measures for RBS, RBD, and E-RBD

Second, as expected, RBD has very poor equity when the program is not cancelled; in hindsight, short-haul flights have been penalized excessively. On the other hand, since RBD saves significant delay under early cancellation times, it also registers less deviation from the ideal allocation. This is because more flights are allowed to depart closer to their scheduled departure times. Third, as δ increases from 20 to 80 min under E-RBD, equity worsens, if the GDP is not canceled earlier.

Figures 7 and 8 provide an assessment of the efficiency and equity tradeoffs under E-RBD. If the GDP is canceled two hours earlier than planned, E-RBD with δ=0 (i.e. RBS), δ=20 minutes, and δ=80 minutes, yields 49%, 25%, and 4% additional total delay compared to the RBD allocation. The equity gained from the three allocation policies, in lieu of the loss in efficiency, are 35%, 30%, and 9% respectively, compared to the RBD allocation. Clearly, under RBS, the 35% gain in equity, compared to RBD, is outweighed by the 49% loss in efficiency, if the GDP cancels two hours earlier than anticipated. Whereas, setting the parameter δ to a value of 80 minutes produces less percentage loss in efficiency than the gain in equity metric. In case of even earlier cancellation of the ground delay program, the RBD algorithm produces the most efficient and equitable allocation of slots to flights.

Figure 9 illustrates the fundamental differences between E-RBD and DB-RBS most clearly. It compares the two procedures with respect to both equity and efficiency, for a 2-hour early cancellation time (chosen because it is typical in GDPs). Each point shows the total delay (vertical axis) for a given level of max deviation from RBS (horizontal axis). More efficient points sit lower on the graph, while more equitable points lie farther to the left. This "criterion space" plot is often used in multi-objective optimization to visually distinguish between dominated and non-dominated solutions and to allow decision makers to make tradeoffs among candidate solutions. We can see that the E-RBD curve dominates the DB-RBS curve. Thus, there are several cases where one can move from a DB-RBS solution to an E-RBD solution and simultaneously improve both equity and efficiency. Delay savings in such cases are on the order of 10%.

Also illustrated in Figure 9, is the capability under E-RBD to produce a wide range of solutions by varying the parameter δ. On the other hand, DB-RBS is only able to produce a much smaller set. This results because airports are not evenly distributed over distance and also because a slight increase or decrease in the exemption distance often results in the inclusion or exclusion, respectively, of flights from several airports.

We have computed comparable results for the other four cancellation times as well (charts omitted for sake of brevity). The delay savings generated by E-RBD range from 0 to 19%, with the greatest savings occurring at the 4-hour cancellation time.

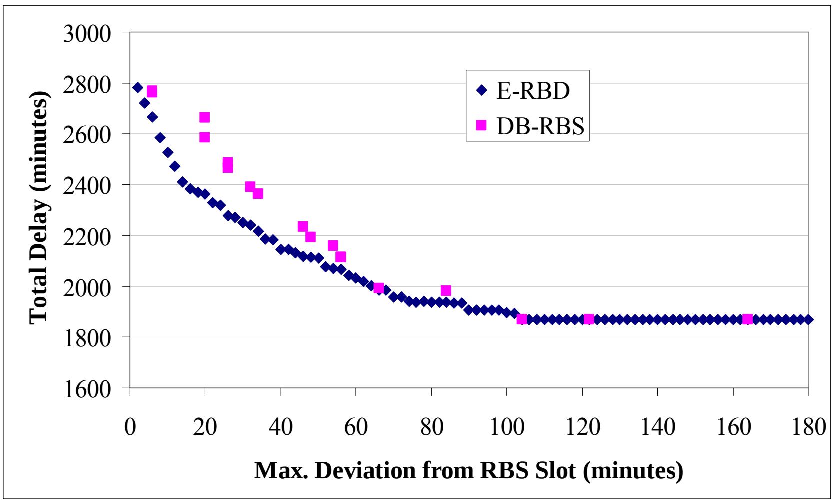


Figure 9: Efficient Frontiers for E-RBD and DB-RBS

Finally, to complement the results of Section 4.3, we compare the performance of RBD and RBS algorithms when the GDP is extended by two hours. The results are presented in Figure 10. In Section 4.3, we concluded that the amount of airborne holding under GDP extension depends on how far in advance that information becomes available. Figure 10 clearly substantiates this – with a 30 minute advance notice, the amount of airborne holding that can be eliminated by adjusting CTA's is 49% and 25% under RBD and RBS allocation respectively.

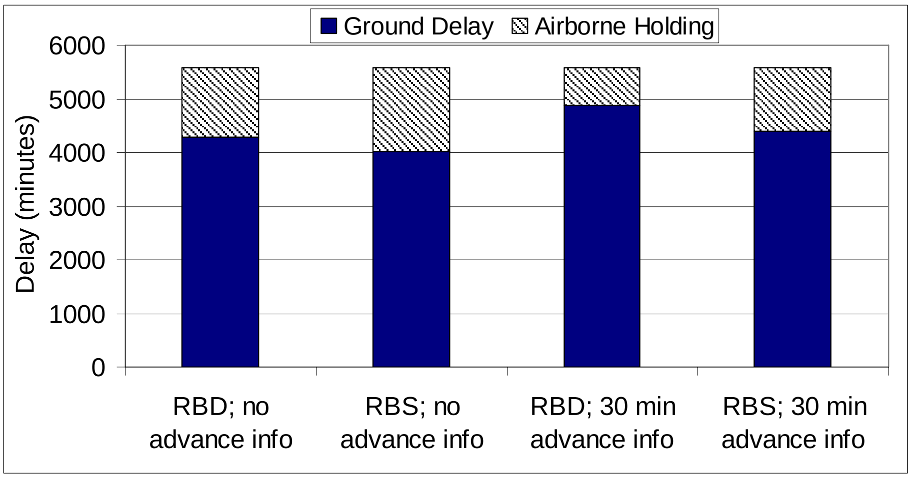


Figure 10: Performance Comparison of RBD and RBS Algorithm under GDP Extension by 2 Hours

Note that the amount of airborne holding resulting from the inability to adjust CTA's of airborne flights is also lower under RBD allocation, as revealed by Figure 10. In the case when no advance information on the GDP extension is available, nearly 16% less airborne holding occurs if the initial slot allocation was done by RBD rather than RBS. This reduction, which is even greater with 30 minutes of advance notice (41% lower in this case), results because the emphasis given to short-haul flights provides added flexibility when extending a program just as it did when canceling a program early. Generally speaking, under RBD, there are fewer flights on the ground in the early stages of a GDP while there are more on the ground in the later stages of a GDP. This makes it possible to adjust the ground delays of short-haul flights and prevent excess airborne holding. We maintain that the decision to extend programs is usually made in the later stages of a GDP and thus RBD in fact improves the flexibility and the options available in planning an extension.

7. Conclusions

We have described a new GDP slot rationing scheme, RBD, and we have shown that it minimizes total expected delay under very broadly applicable early termination models. We have further described a second rationing scheme, E-RBD, that is practical in the sense that, unlike (pure) RBD, it takes into account both equity and efficiency factors. Our computational experiments show that not only is E-RBD comparable to the DB-RBS algorithm used in practice today, but in fact, it provides an efficiency advantage.

E-RBD has a second important advantage over DB-RBS. DB-RBS is driven by its distance parameter. As this parameter is increased, additional airports fall into the scope of the GDP, meaning that flights departing those airports must share in the total assigned ground delay. Often times, a slight change in the distance parameter can affect a large number of flights of one airline (e.g. when the airport is a hub for one airline). This sensitivity of the distance parameter motivates airlines to argue for or against specific distance parameters on a daily basis. In contrast, our E-RBD policy is driven by a very natural parameter (δ = maximum deviation from RBS), with a clear performance interpretation: a measure of equity. As such, it could be set based on objective standards or a well-defined FAA policy, thus reducing the temptation for parochial considerations to creep into FAA decision making.

In addition, as it is changed, the impact on delays allocated to flights should be less abrupt than the impact of changing the DB-RBS distance parameter. Thus, the use of E-RBS admits a more scientific, and less political, basis for GDP planning. While a wider range of experiments and scenarios are certainly needed, our proven principles and demonstrated results provide a strong case for the adoption of E-RBD.

8. Acknowledgements

This work was supported by the National Center of Excellence for Aviation Operations Research (NEXTOR), under FAA cooperative agreement number 01CUMD1. Opinions expressed herein do not necessarily reflect those of the FAA or the U.S. Department of Transportation. Part of this research was carried out by the first author while visiting the Institute of Transportation Studies (ITS) at the University of California, Berkeley. The support and hospitality of ITS and UC Berkeley are gratefully acknowledged.

9. References

Ahuja, R. K., T. L. Magnanti, and J. B. Orlin. 1993. Network flows: theory, algorithms, and applications, Prentice Hall, New Jersey.

Ball, M.O., C. Barnhart, G. Nemhauser and A. R. Odoni, 2007, "Air Transportation: Irregular Operations and Control", *Handbook of Operations Research and Management Science: Transportation,* C. Barnhart and G. Laporte, eds, Elsevier, Amsterdam, pp 1-67.

Ball, M.O., G. Dahl and T. Vossen, 2007. Matchings in connection with ground delay program planning, submitted for publication.

Ball, M.O., R. Hoffman, A. Odoni, and R. Rifkin. 2003. A stochastic integer program with dual network structure and its application to the ground-holding problem. *Operations Research*, **51**, pp. 167 – 171.

Ball, M.O. and G. Lulli. 2004 Ground delay programs: Optimizing over the included flight set based on distance. *Air Traffic Control Quarterly* **12**, pp. 1 – 25.

Ball, M.O., T. Vossen. and R. Hoffman. 2001, Analysis of demand uncertainty in ground delay programs, in *Proceedings of 4th USA/Europe Air Traffic Management R&D Seminar*, 2001.

Birge, J.R. and F. Louveaux. 1998. Introduction to stochastic programming. Springer, New York.

Federal Aviation Administration, 2006, CDM Web Site: http://cdm.fly.faa.gov/index.html.

Hoffman, R., M.O. Ball, A. Mukherjee. 2007. Ration-by-distance with equity guarantees: a new approach to ground delay program planning and control. In *Proceedings of the 7th USA/Europe Air Traffic Management R&D Seminar,* Barcelona.

Inniss, T. and Ball, M.O., 2004, Estimating one-parameter airport arrival capacity distributions for air traffic flow management, *Air Traffic Control Quarterly,* **12**, 223-252.

Kotnyek , B. and O. Richetta. 2006. Equitable models for the stochastic ground holding problem under collaborative decision making. *Transportation Science* **40**, pp. 133-146.

Liu, P., M. Hansen, and A. Mukherjee. 2006. Scenario-based air traffic flow management: From theory to practice. *Transportation Research Record* **1951**, pp. 113-121.

Liu, P. and M. Hansen. 2007. Scenario-free sequential decision model for the single airport ground holding problem. In *Proceedings of the 7th USA/Europe Air Traffic Management R&D Seminar*, Barcelona.

Mukherjee, A. and M. Hansen. 2007. Dynamic stochastic model for a single airport ground-holding problem. To appear in *Transportation Science*.

Odoni, A. R. 1987. The flow management problem in air traffic control. In A. R. Odoni, L. Bianco, and G. Szego, Editors, *Flow Control of Congested Networks*, Springer-Verlag, Berlin, pp. 269-288.

Richetta, O. and A. R. Odoni. 1993. Solving optimally the static ground-holding problem in air traffic control. *Transportation Science* **27**, pp. 228-238.

Richetra, O. and A. R. Odoni. 1994. Dynamic solution to the ground holding problem in air traffic control. *Transportation Research* **28A**, pp. 167-185.

Terrab, M. and A. R. Odoni. 1993. Strategic flow management for air traffic control. *Operations Research* **41**, pp. 138-152.

Vossen, T., M. O. Ball, R. Hoffman, and M. Wambsganss. 2003. A general approach to equity in traffic flow management and its application to mitigating exemption bias in ground delay programs. *Air Traffic Control Quarterly*, **11**, pp. 277-292. (a preprint was published in *Proceedings of 5th USA/Europe Air Traffic Management R&D Seminar*, 2003).

Vossen, T. and M. O. Ball. 2006a. Optimization and mediated bartering models for ground delay programs. *Naval Research Logistics*, **53**, pp. 75 – 90.

Vossen, T. and M. O. Ball. 2006b. Slot trading opportunities in collaborative ground delay programs. *Transportation Science* **40**, pp. 29-43.

Wambsganss, M., 1996, Collaborative decision making through dynamic information transfer, *Air Traffic Control Quarterly*, **4**, pp. 107-123.

Willemain, T.R. 2002. Contingencies and cancellations in ground delay programs. *Air Traffic Control Quarterly* **10**, pp. 43-64.

Wilson, F.W. 2004. A stochastic air traffic management model that incorporates probabilistic forecasts. *Proceedings of 20th International Conference on Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology*.