An Airspace Planning Model for Selecting Flight-plans Under Workload, Safety, and Equity Considerations

Hanif D. Sherali • J. Cole Smith • Antonio A. Trani

Grado Department of Industrial and Systems Engineering (0118),
Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061
Department of Systems and Industrial Engineering, University of Arizona, Tucson, Arizona 85721
Charles Edward Via, Jr. Department of Civil and Environmental Engineering (0105),
Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061
hanifs@vt.edu • cole@sie.arizona.edu • vuela@vt.edu

In this paper, we present an airspace planning model (APM) that has been developed for use in both tactical and strategic planning contexts under various airspace scenarios. Given a set of flights for a particular time horizon, along with (possibly several) alternative flight-plans for each flight that are based on delays and diversions, due to special-use airspace (SUA) restrictions prompted by launches at spaceports or adverse weather conditions, this model prescribes a set of flight-plans to be implemented. The model formulation seeks to minimize and delay fuel-cost-based objective function, subject to the constraints that each flight is assigned one of the designated flight-plans, and that the resulting set of flight-plans satisfies certain specified workload, safety, and equity criteria. These requirements ensure that the workload for air-traffic controllers in each sector is held under a permissible limit, that any potential conflicts are routinely resolvable, and that the various airlines involved derive equitable levels of benefits from the overall implemented schedule. To solve the resulting 0–1 mixed-integer programming problem more effectively using commercial software (e.g., CPLEX-MIP), we explore the use of reformulation techniques designed to more closely approximate the convex hull of feasible solutions to the problem. We also prescribe a polynomial-time heuristic procedure that is demonstrated to provide solutions to the problem within 0.01% of optimality. Computational results are reported on several scenarios based on actual flight data obtained from the Federal Aviation Administration (FAA) to demonstrate the efficacy of the proposed approach for air-traffic management (ATM) purposes.

With impending future implementations of the free-flight paradigm for airline operations, and with the potential widespread use of the reusable launch vehicle (RLV) technology, many challenges arise in the context of airspace planning. The free-flight paradigm would permit airlines to compose more cost-effective routes for their flights rather than be restricted to traverse paths between designated way-points from various origins to different destinations. In addition,

it is anticipated that the RLV technology will add as many as 1,200 vehicle launches over the next decade (AST 1998), prompted by its ability to perform the same function as a space shuttle expedition at a fraction of the cost. Given that each such RLV launch involves the cordoning of a large section of airspace around the spaceport due to safety considerations, the result can be a severe disruption of commercial air traffic because flights routinely need to

be detoured around the ensuing so-called special-use airspace (SUA) restrictions.

Furthermore, a sharp increase in space transportation launches would induce increased delays of many flights. Even under present-day operations, delays and rerouting of air traffic are necessitated not only due to SUA delineations caused by commercial and military space launches, but also due to severe weather fronts or conditions, or zones of high turbulence. Events of this type create the need for the airlines and the Federal Aviation Administration (FAA) to jointly address how flights could be routed and scheduled in a collaborative decision-making framework that would prescribe an efficient and equitable management of delays.

Motivated by the foregoing considerations, and in response to the FAA's need for a collaborative decision-making tool, we develop a 0–1 mixed-integer programming model in this paper called the *airspace planning model* (APM). Given a set of flights to be scheduled over a planning horizon, along with several alternative flight-plans for each flight (referred to as *surrogates* for each flight), this model selects a set of flight-plans that minimize a delay and fuel-cost-based objective function, such that each flight is executed via a designated flight-plan, while satisfying certain specified workload, safety, and equity restrictions. The surrogates of each flight might represent alternative free-flight trajectories, or traversals between designated navigational fixes, that suitably differ in departure delay and time-space paths (including altitudes), while possibly detouring around SUAs or weather fronts. The model also requires that the selected mix of flight-plans should not impose any potential collision risk or conflict that is not routinely resolvable. Furthermore, the number of aircraft traversing each defined sector of the airspace at any point in time, as well as the accompanying conflict resolution requirements, should be limited to a manageable workload level conformant with the corresponding sector's air-traffic control (ATC) capabilities. Moreover, any delays and additional costs (operational as well as customer dissatisfaction penalties) should be equitably shared among the affected airline carriers in some acceptable, defined sense. Within the framework of generating this proposed model,

we utilize two particular submodels developed by Sherali et al. (2000). These two submodels are the *airspace sector occupancy model* (AOM), which determines the entering and exiting times of flights traveling through the various sectors that are characterized as three-dimensional, nonconvex polygonal regions of airspace, and the *aircraft encounter model* (AEM), which efficiently detects all possible pairwise conflicts between flight-plans, providing detailed statistics regarding the severity and geometry of each conflict. The information garnered from these two models is used to generate the various constraints for the airspace planning model developed herein.

Some seminal research contributions in the area of air-traffic management have been made by Lindsay et al. (1993) and Bertsimas and Patterson (1998, 2000). Lindsay et al. present a time assignment model for routing a set of flights through designated sequences of fixes. They discretize time into (15 minute) intervals and define binary variables to track the position of each flight during each time interval with respect to its specified sequence of fixes. The derived model seeks to minimize the total ground and air delay costs subject to capacity constraints for the fixes, along with other flight duration, continuity, and turnaround relationships. The authors mention (but do not implement) that equity among airlines could be incorporated within their model by suitably adjusting cost coefficients and/or by imposing additional constraints on the absolute or relative extents of delays that are assigned to each airline. Bertsimas and Patterson (1998) consider a similar ground and enroute control model to determine optimal release times for a set of aircraft and their speeds between fixes, while accounting for airspace capacities. They employ the same set of variables as Lindsay et al., but they construct tighter model representations to improve problem solvability. A significant model enhancement that considers the dynamic rerouting of flights under changing weather conditions was proposed by Bertsimas and Patterson (2000). Here a time-space network of airports and fixes is constructed based on assumed time-dependent flight durations between pairs of nodes. By aggregating flights between each origin-destination (OD) pair and representing these as a flow of a specific commodity, they model the

problem of minimizing total ground holding and air delay costs subject to flow balance, continuity, and sector capacity constraints as a multicommodity integer network flow problem with side-constraints. A subsequent disaggregation of the resulting flows into specific flight trajectories is conducted using a randomization scheme. Observe that this assumes homogeneity among the overall mix of aircraft. Again, in contrast with our model, the authors only mention (and do not implement) the fairness or equity issue among airlines, suggesting that it could be modeled by expanding the commodities to be OD-pair as well as airline specific and then ensuring while executing the disaggregation routine that each airline receives no more than a given percentage of total delay. Observe also that none of these foregoing models are designed to consider free-flight trajectories, nor do they address collision-risk issues among the mix of prescribed flight trajectories, along with related conflict resolution workloads imposed on the various sectors. On the other hand, while our model considers all such specific aircraft characteristics, collision risk, sector workload, and equity issues, it requires the pregeneration of alternative flight-plans for each of the flights. Hence, our model can possibly benefit from the use of these other existing models to generate (perhaps in a dynamic column generation fashion) alternative flight-plans. This feature will be explored in future research developments.

Among other related works, Falker (1999a, b) provides an in-depth review of current National Airspace policies under which air and space traffic are kept separate. Future airspace scenarios are examined in which space transportation launches have significantly increased, and a discrete event simulation process is used to study several alternative modes of operations with respect to their safety, efficiency, equity, and ease of implementation. Detailed summaries of several other existing air-traffic management models may be found in Odoni et al. (1997), and in a concept paper by the Federal Aviation Administration/Eurocontrol (1998). Among these, the *analytic blunder risk model* (ABRM) is an analytic/probabilistic model that estimates the collision risk between two aircrafts when one aircraft strays from its intended

course and the other must avoid it. Another relevant tool is the *traffic organization and perturbation analyzer* (TOPAZ). This model evaluates the safety of any given operational concept by utilizing a suite of simulation software modules that differ from other fast-time simulation routines by allowing probabilistic deviations from normal operating conditions. The *national airspace resource investment model* (NARIM) is another prominent analytical tool that has been developed to assist the FAA and the National Aeronautics and Space Administration (NASA) in evaluating future airspace scenarios. NARIM uses several analytical procedures along with data collected from aircraft operations to provide a broad spectrum of information pertaining to optimal altitude selections, aircraft performance characteristics, and systemwide performance, under various alternative scenarios and new concepts of operations. NASA has also developed a set of decision support tools that comprise the system CTAS (*Center/TRACON Automation System*) to aid air-traffic controllers, and to enhance the efficiency and handling capacity at congested airports (see Erzberger et al. 1999 and McNally et al. 1998). The model presented in the present paper could be integrated within the foregoing existing arsenal of tools that are being utilized by the FAA and NASA for both tactical and strategic planning purposes. For example, CTAS has a conflict prediction and trial planning feature that permit the evaluation of a trial flight-plan with respect to the conflicts it induces when introduced within an existing mix of flight-plans. Our model could augment this feature by providing the capability to simultaneously analyze multiple flights and select among several surrogate flight-plans, while considering not only potential conflicts and their resolvability, but also their ensuing effects on sector workload and airline equity. In this manner, our model could be utilized to provide both operational and policy guidelines for safe, economical, and equitable routing and scheduling of commercial air traffic.

The remainder of this paper is organized as follows. In §1, we discuss details of the decision structure and the various modules pertaining to equity, workload, and collision risk considerations for developing the proposed airspace planning model. Two

alternative formulations of this problem are presented in §2, along with some preliminary results that lend insights into their relative characteristics and solvability. In §3, we develop a polynomial-time heuristic procedure that provides tight upper bounds to the problem. Computational results are provided in §4 to demonstrate the efficacy of the proposed exact and heuristic approaches using test problem instances based on actual airspace data that represent realistic scenarios of interest to the FAA. Finally, we conclude the paper in §5 by discussing various uses for this planning model and providing recommendations for future research.

1. Decision Structure and Modules for the Airspace Planning Model

Consider a planning horizon H, and suppose that we are given a set of flights $f = 1, \ldots, F$ covering this horizon that are relevant to a certain region of airspace. For each flight f, let

$P_f = \{$set of possible flight-plans p composed of departure and anticipated arrival times, along with time-space trajectories and cruising altitudes, while traversing designated routes between the corresponding origin-destination pair$\}$.

Note that there will typically exist some preferred departure time for each flight, along with some alternative (discrete) departure schedules or departure slots. For each departure time, given the existing time-dependent state of the special-use airspace restrictions or weather patterns, one or more flight-plans corresponding to different trajectories, aircraft velocities, and cruising altitudes could be generated using some commercial package, such as FliteStar (Jeppesen Sanderson, Inc. 1996) or OPGEN (NAS Concept Development 1998), for example. These models consider various details such as wind velocity and direction vectors, special-use airspaces, and terrain obstacles, while optimizing the flight paths. Note that each trajectory would be characterized as a (four-dimensional) time-space entity and could represent traversals between designated fixes or free-flight

paths that conform with the restricted airspace usage requirements. Given any such combination (f, p), $f = 1, \ldots, F, p \in P_f$, we can compute a cost factor c_{fp} for adopting plan p for flight f. This cost would reflect the related fuel expended, ground-holding, and air delay costs for the particular flight-plan. Note also that to ensure an equitable treatment of flights (as discussed subsequently in §1.1), the FAA could require each airline to submit plans for each of their flights (perhaps in a collaborative decision-making framework) that differ adequately with respect to departure times according to some established criteria. Such policies would need to be cognizant of any related gaming strategies on the part of airlines.

Accordingly, defining the *decision variables*

$$x_{fp} = \begin{cases} 1 & \text{if plan } p \in P_f \text{ is adopted for flight } f \\ 0 & \text{otherwise} \end{cases}$$
$$\forall f = 1, \ldots, F, p \in P_f, \tag{1}$$

we can formulate a total system-based objective function to

$$\text{minimize} \sum_{f=1}^{F} \sum_{p \in P_f} c_{fp} x_{fp}. \tag{2}$$

The constraints would include the selection of a plan for each flight as specified by

$$\sum_{p \in P_f} x_{fp} = 1 \quad \forall f = 1, \ldots, F, \tag{3}$$

as well as certain equity, workload, and conflict resolution restrictions, which are discussed next. Note that the mix of selected flight-plans that are ultimately prescribed by the model would entail certain *estimated* departure times and time-space trajectories. Naturally, in practice, these estimates are subject to uncertainty and variations in their values would need to be addressed dynamically in real-time by air-traffic controllers at airports. The proposed model could also be rerun in a rolling horizon fashion at suitable frequencies as, and when, updated information becomes available to revise decisions based on prior realizations.

1.1. Equity Constraints

Suppose that there are some $\bar{\alpha}$ airlines involved in the given mix of flights, indexed by $\alpha = 1, \dots, \bar{\alpha}$. In the process of selecting flight-plans based on (2) and (3) (in addition to workload and conflict resolution constraints as described in the subsequent sections), we would also like to achieve a degree of equity among these airline carriers. For each airline $\alpha = 1, \dots, \bar{\alpha}$, let us therefore define a *utility-based measure of effectiveness*

$$U_\alpha(x) \equiv \sum_{(f,p) \in \mathscr{A}_\alpha} u_{fp} x_{fp} \qquad (4a)$$

where

$$\mathscr{A}_\alpha \equiv \{(f,p), p \in P_f : \text{flight } f \text{ belongs to airline } \alpha\},$$
$$\alpha = 1, \dots, \bar{\alpha}, \qquad (4b)$$

and where the coefficients u_{fp} are suitably defined, as specified in Remark 2, for example. Accordingly, defining the equity variables x_l^e and x_u^e to respectively represent the lower and upper limits of the range of variation for the effectiveness measures $U_\alpha(x)$, $\alpha = 1, \dots, \bar{\alpha}$, where x_u^e is restricted to be no more than some specified value ν_e, we can model equity via the following mechanism.

Include the Following Restrictions Within the Constraints.

$$x_l^e \le U_\alpha(x) \le x_u^e \quad \forall \alpha = 1, \dots, \bar{\alpha}; \qquad (5)$$
$$x_l^e \ge 0, x_u^e \le \nu_e. \qquad (6)$$

Include the Following Terms Within the Objective Function.

$$(\text{Minimize}) \cdots + \mu_e(x_u^e - x_l^e) + \mu_u^e x_u^e \qquad (7)$$

where μ_e is a (commensurate) penalty per unit of variation in the measures $U_\alpha(x)$, $\alpha = 1, \dots, \bar{\alpha}$ and μ_u^e is a (commensurate) penalty for the maximum incurred measure of effectiveness. Note that we could take $\mu_u^e \equiv 0$ in case restricting $x_u^e \le \nu_e$ is itself sufficient. On the other hand, in order not to overly restrict the problem, ν_e could be taken as the maximum tolerable limit on any $U_\alpha(x)$ value, and then the penalty μ_u^e would serve to reduce x_u^e below ν_e to the extent

possible or desirable. The main idea here is to minimize the spread as well as the maximum measure of effectiveness over the airline carriers. (Taking $\mu_e = x_l^e = 0$ would correspond to an alternative pure minimax strategy.)

REMARK 1. The variables x_l^e and x_u^e can be fixed at their respective bounds of zero and ν_e if so desired. In this case, each measure $U_\alpha(x)$ would simply be restricted to be no greater than ν_e, for $\alpha = 1, \dots, \bar{\alpha}$, and the implied lower bounding constraints in (5) may be omitted.

REMARK 2. One possible measure of effectiveness $U_\alpha(x)$ (as used in the present study) is the average *excess* fuel and delay costs incurred by the airline α for each $\alpha = 1, \dots, \bar{\alpha}$. Letting m_α be the number of flights that belong to airline α, and denoting the excess fuel and delay costs over the preferred plan for each flight to be given by

$$d_{fp} \equiv c_{fp} - \min\{c_{fp} : p \in P_f\} \quad \forall f = 1, \dots, F, p \in P_f, \qquad (8)$$

we can define $U_\alpha(x)$ in (4a) by letting

$$u_{fp} \equiv d_{fp}/m_\alpha \quad \forall (f,p) \in \mathscr{A}_\alpha, \alpha = 1, \dots, \bar{\alpha}. \qquad (9)$$

Note that to ensure that an airline operating relatively more efficient equipment does not subsidize another airline that operates less efficient equipment in this scheme, we could consider the (excess) fuel-cost component used in (8) to be relative to some uniform aircraft efficiency characteristics. In either case, x_l^e and x_u^e can be treated as variables as in (5) and (6), or be fixed as mentioned in Remark 1.

1.2. Workload Constraints

Consider the total collection of flight-plans $\bigcup_{f=1,\dots,F} P_f$. Jointly, these plans involve traversals between certain pairs of fixes, as well as free-flight cruises between designated pairs of points/fixes, at various specified altitudes. Let us consider the segmentation of the national airspace into *sectors* as defined by the FAA (these are generally nonconvex polygons lifted into the vertical dimension and stacked at low, high, and super-high altitudes), and let

$$S = \{\text{set of all sectors involved with the collection}$$
$$\text{of the flight-plans } \bigcup_{f=1,\dots,F} P_f\}. \qquad (10)$$

Define the *workload* for any sector $s \in S$ at any point in time to be the number of aircraft that are residents within that sector at the given instant of time. To characterize this workload for each sector $s \in S$, we can examine the occupancy durations of the various flights within s over the horizon H. The model AOM of Sherali et al. (2000) provides this information by constructing a Gantt chart of flight-plan occupancy intervals for each sector. Note that a particular flight-plan that traverses through some sector s might be in conflict with another flight-plan that occupies a different neighboring sector s' as described in §1.3. While the latter flight-plan might not physically occupy sector s, this situation does need monitoring by both the sectors s and s'. Therefore, within the Gantt chart for sector s (and likewise for sector s'), we include the interval of duration for the flight-plan in sector s' over which the conflict with the identified flight-plan in sector s persists. Hence, the workload constraints accommodate such extraneous occupancy intervals as well. In practice, air-traffic control (ATC) operators routinely monitor several aircraft that are simultaneously traversing their sectors. Of course, when the workload (simultaneous occupancy of a number of aircraft) becomes too high, a potentially dangerous or untenable situation can arise. Hence, let us define the following entities.

For each sector $s \in S$, let $i = 1, \dots, M_s$ index the collection of maximal overlapping sets C_{si} of flight-plans (f, p), where an overlapping set of flight-plans is called maximal if it is not a strict subset of another overlapping set. For example, examining Figure 1, we have $M_s = 4$ maximal sets given by $C_{s1} = \{(f_1, p_1), (f_2, p_2), (f_3, p_3)\}$, $C_{s2} = \{(f_2, p_2), (f_3, p_3), (f_4, p_4)\}$, $C_{s3} = \{(f_3, p_3), (f_4, p_4), (f_6, p_6)\}$, and $C_{s4} = \{(f_5, p_5), (f_6, p_6)\}$. Hence, we have,

$$C_{si} = \{(f, p) : \text{flight-plan } (f, p) \text{ belongs to the } i\text{th}$$
$$\text{maximal overlapping set for sector } s\},$$
$$\forall i = 1, \dots, M_s, s \in S. \quad (11)$$

An efficient algorithm for determining these sets is described in Sherali and Brown (1994). Note that it is possible that if (f_1, p_1) and $(f_2, p_2) \in C_{si}$, for some s and i, then $f_1 = f_2$, i.e., this pair corresponds to the
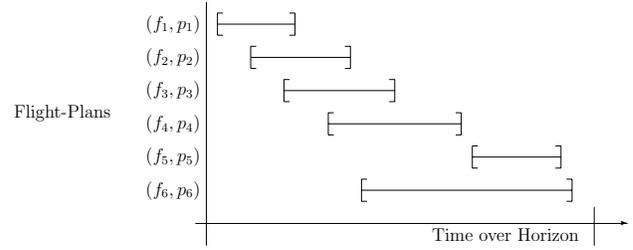

Figure 1   Gantt Chart for Formulating Workload Constraints

same flight, although in this case, the plans would be distinct.

Let us now define the variable n_s to represent the maximum number of overlapping flights within each sector $s \in S$. Note that n_s is given by the largest number of flight-plans selected from any of the maximal overlapping sets $C_{si}, i = 1, \dots, M_s$, i.e.,

$$n_s = \underset{i=1,\dots,M_s}{\text{maximum}} \left\{ \sum_{(f,p) \in C_{si}} x_{fp} \right\}, \quad (12)$$

because any other overlapping set is a subset of some maximal overlapping set. In our model, we bound this variable n_s on a suitable interval $[1, \bar{n}_s]$ and, furthermore, penalize its value in the objective function using a penalty factor that increases nonlinearly in an appropriate fashion with an increase in workload. The motivation here is that if the maximum number of aircrafts being simultaneously monitored in a sector increases from one to three, for example, the associated penalty should likely more than triple. Moreover, there should exist some absolute maximum number $\bar{n}_s$ of overlapping flights at any point in time, as determined by the capacity of sector s. We demonstrate below how this workload and associated penalty structure can be modeled without having to discretize time.

Toward this end, let us define the binary variables

$$y_{sn} = \begin{cases} 1 & \text{if the maximum workload in sector } s \text{ is } n \\ 0 & \text{otherwise} \end{cases}$$

$$\forall s \in S, n = 1, \dots, \bar{n}_s$$

and let μ_{sn} be the associated penalty for having $y_{sn} = 1$. We assume that

$$\mu_{s2} \geq \mu_{s1} \text{ and } \mu_{sj} \geq 2\mu_{s(j-1)} - \mu_{s(j-2)} \quad \forall j = 3, \dots, \bar{n}_s. \quad (13)$$
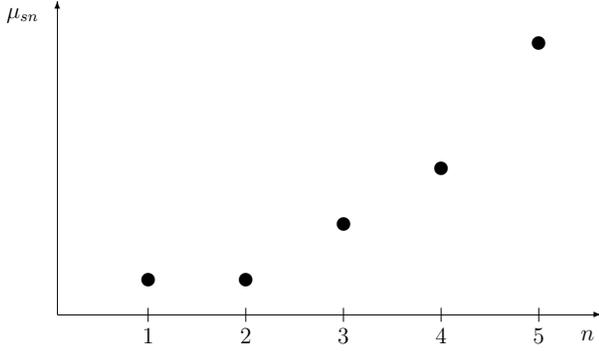
Figure 2   Illustration of a Convex Sector Workload Cost Structure

Note that the condition (13) implies that

$$0 \leq (\mu_{s2} - \mu_{s1}) \leq (\mu_{s3} - \mu_{s2}) \leq \cdots \leq (\mu_{s\bar{n}_s} - \mu_{s,\bar{n}_s-1}). \tag{14}$$

 Figure 2 illustrates the implied convex nature of this penalty structure. Observe that by enforcing $n_s \geq 1$, we always incur a workload cost of at least μ_{s1}, even when no aircrafts are being scanned over the horizon. This is reasonable because there always exists a fixed monitoring cost. More importantly, by avoiding a cost of zero corresponding to $n_s = 0$, we permit greater flexibility in considering practical workload costs that would satisfy (13). For example, we might have $0 < \mu_{s1} = \mu_{s2} = \cdots = \mu_{s\tau}$ for up to some threshold number τ of aircrafts being monitored, after which the costs might increase at an increasing rate as in (14). This is the cost structure that arises in practice, and we shall assume that it holds true. Moreover, as shown in Proposition 1, this structure precludes an explicit consideration of binariness on the y-variables, enabling us to equivalently treat these variables as continuous. This penalty structure for workload consideration may be incorporated into the model as follows, noting (12).

Include the Following Restrictions Within the Constraints.

$$\sum_{(f,p)\in C_{si}} x_{fp} - n_s \leq 0 \quad \forall i = 1, \ldots, M_s, s \in S \tag{15a}$$

$$n_s = \sum_{n=1}^{\bar{n}_s} n y_{sn} \quad \forall s \in S \tag{15b}$$

$$\sum_{n=1}^{\bar{n}_s} y_{sn} = 1 \quad \forall s \in S \tag{15c}$$

$$y_{sn} \geq 0 \quad \forall n = 1, \ldots, \bar{n}_s, s \in S. \tag{15d}$$

Include the Following Term Within the Objective Function.

$$\text{(Minimize)} \ \cdots + \sum_{s\in S} \sum_{n=1}^{\bar{n}_s} \mu_{sn} y_{sn}. \tag{15e}$$

Note that both n_s and y_{sn}, $\forall s, n$ have been declared as continuous variables in (15). As justified by the following result, the binary restrictions on y hold automatically at optimality in our model and, hence, so do the integrality and the bounding restrictions on the variables n_s $\forall s \in S$.

PROPOSITION 1. *Assume that* (13) *holds true. Consider any model that includes the term* (15e) *in the objective function and the restrictions* (15a–d) *in the constraints, where (among other stipulations) the x-variables are required to be binary valued. Then, given any x^* as part of an optimal solution to such a model, there exists an optimal completion for which $n_s = n_s^*$, $y_{sn_s^*} = 1$, and $y_{sn} = 0$ $\forall n \neq n_s^*$, where*

$$n_s^* \equiv \max\left\{1, \max_{i=1,\ldots,M_s}\left\{\sum_{(f,p)\in C_{si}} x_{fp}^*\right\}\right\}$$

is an integer in $[1, \bar{n}_s]$, for each $s \in S$. (16)

PROOF. Consider any $s \in S$. Define $f_s(n_s)$ as a piecewise linear function having breakpoints at $n = 1, \ldots, \bar{n}_s$ with $f_s(n) \equiv \mu_{sn}$ $\forall n = 1, \ldots, \bar{n}_s$. Note that by (14), f_s is nondecreasing and convex. Moreover, because the constraints (15b–d) compose n_s as a convex combination of the grid points $\{1, \ldots, \bar{n}_s\}$, and the nondecreasing nature of the penalties prefers n_s to be as small as possible, we can take $n_s = n_s^*$ as given by (16). By feasibility, $n_s^* \in [1, \bar{n}_s]$ is integer valued. Furthermore, by the convexity of f_s, we have that

$$\mu_{sn_s^*} \equiv f_s(n_s^*) \leq \sum_{n=1}^{\bar{n}_s} y_{sn} f_s(n) \equiv \sum_{n=1}^{\bar{n}_s} y_{sn}\mu_{sn} \tag{17}$$

for any feasible completion y to (15b–d), given $n_s = n_s^*$. Hence, we have that $y_{sn_s^*} = 1$, and $y_{sn} = 0$ $\forall n \neq n_s^*$, is

an optimal set of values for y_{sn}, $n = 1, \ldots, \bar{n}_s$, for each $s \in S$. This completes the proof. $\square$

REMARK 3. Observe that for each flight-plan combination (f, p), we can examine the number of times x_{fp} appears in the constraint set (15a) to assess the degree of workload being generated by this flight-plan. This indicator could be used to prompt the generation of alternative plans for a given flight f, based on the degrees of workload associated with the flight-plans in their current set P_f (in addition to related equity considerations). Furthermore, note that the foregoing concept of workload focuses on the critical notion concerning the maximum number of overlapping flights that are required to be monitored within each sector over the horizon. Several alternative measures of workload could be investigated in this context. For example, we could partition the horizon into suitable discrete time segments and evaluate and penalize the maximum number of overlapping flights that are required to be monitored within each sector for each time segment using a structure similar to (15)–(16) that is replicated for each such time segment. In addition, restrictions or penalties could be imposed based on the occupancy durations and relative headings of the flights within each particular sector. Such alternative considerations will be explored in future enhancements of the proposed model.

1.3. Conflict Constraints

For each sector, let us discretize the horizon into uniform time segments. The duration of each such segment for any sector would depend on the sector's conflict resolution capability. This capability is to be reflected within the conflict constraints developed later, which impose the restriction that for each sector the maximum number of (permissible) conflicts that need to be resolved for each time segment should not exceed one. (Note that this complements the foregoing workload restriction consideration.) For example, because air-traffic controllers monitoring a New York enroute sector are trained to be exposed to relatively larger workloads, and can handle a greater frequency of potential conflict resolutions, the duration of the time segments in the corresponding sector could be relatively smaller.

To develop these conflict constraints, we must first be able to evaluate each pair of flight trajectories for any potential conflicts. Furthermore, we must be able to determine the sector occupancy of each flight trajectory and, consequently, detect when and over which sectors intrusions leading to collision risk occur. This information is retrieved by running the airspace occupancy model (AOM) and the aircraft encounter model (AEM) as described in detail by Sherali et al. (2000). The sector occupancy information is derived from AOM as discussed in §1.2. The model AEM performs a conflict analysis based on the output of AOM. In a preprocessing step, AEM first identifies all possible situations in which conflicts might arise between pairs of flight-plans that are not surrogates of the same flight, along with the times during which such conflicts might occur. These potentially conflicting flight pairs are then examined in greater detail as they traverse between designated way-points (or breakpoints) along their trajectories during the identified critical intervals, to detect and characterize any conflicts that actually occur. AEM reports the severity of each such identified conflict, along with a detailed geometric analysis regarding the nature of the conflict.

If any detected conflict is declared to be *fatal* in this analysis by virtue of an aircraft penetrating the inviolable airspace surrounding another aircraft, we would immediately impose a constraint in the present model that permits the selection of at most one such flight-plan. Denoting FC as the set of such *fatally conflicting* pairs of flight-plans $P \equiv (f_1, p_1)$ and $Q \equiv (f_2, p_2)$, we begin by stipulating that

$$x_P + x_Q \leq 1 \quad \text{for all } (P, Q) \in \text{FC}. \tag{18}$$

Other nonfatal conflicting situations are permitted to exist, provided that they can be resolved by the ATC in the particular sector in which they occur. To reflect this resolution capability, we formulate the following set of additional conflict constraints. Suppose that we construct a *conflict graph* $G_{st}(N_{st}, A_{st})$ for each sector s and time segment t, where N_{st} is the set of nodes that represent all the flight-plans (f, p) that reside in sector s during time segment t, and A_{st} is the set of edges
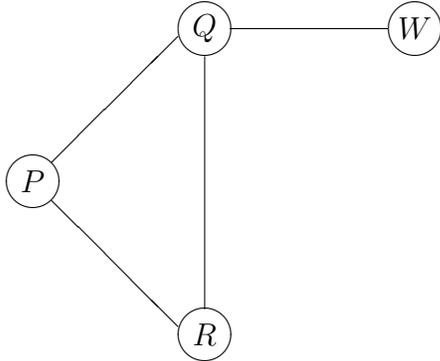
Figure 3   **Illustration of a Conflict Graph** $G_{st}(N_{st}, A_{st})$

such that if flight-plans P and Q are in conflict in sector s during this time segment t, then A_{st} includes an edge joining these corresponding nodes. If a flight-plan residing in an adjacent sector conflicts with a flight-plan residing within s during the time segment t, then we include such a flight-plan in N_{st}, with the corresponding conflict edge being incorporated within A_{st}. Because we have explicitly excluded non-permissible conflicts via (18) above, we can restrict our attention to recording via A_{st} just the *permissible conflicts*; that is, conflicts that can be resolved based on ATC capabilities. The resulting graph G_{st} would typically be a collection of (disjoint) components.

Figure 3 illustrates an instance of a conflict graph for some sector s and time segment t. During this time segment, each pair of flight-plans (P, Q), (P, R), (Q, R), and (Q, W) conflict (at least one flight-plan of each pair penetrates the protective shell of the other). Note that this could occur, as noted above for example, in a situation where flight-plans P, Q, and R are residing within sector s, whereas W is crossing a neighboring sector s', while penetrating the protective shell of the flight-plan Q.

We now impose the constraint that no more than one permissible conflict should occur for each sector during each time segment. To model these constraints, for each sector, consider the edges in A_{st} taken two at a time, and for each pair k let S_k be the set of nodes (representing flight-plans) at which this pair of edges is incident. $|S_k|$ equals three or four, depending on whether the pair of edges is adjacent (i.e., shares a

common end-node) or not. The imposed constraint would then be

$$\sum_{P \in S_k} x_P \le |S_k| - 1, \tag{19}$$

so that the identified pair of edges would be precluded from coexisting. As we are selecting any two edges in A_{st} at a time, there are $|A_{st}|(|A_{st}| - 1)/2$ inequalities of the type (19) for each sector s, for each time segment t. For notational convenience, let us assume that the index k runs contiguously for $k = 1, \ldots, K$ over the constraints (19) for all s, t. Observe that there will likely be several redundant constraints established via this process. In particular, the following result holds true.

PROPOSITION 2. *Consider a pair of constraints of the type* (19) *corresponding to sets* S_{k_1} *and* S_{k_2}, *such that* $S_{k_1} \subseteq S_{k_2}$. *Then* (19) *for* S_{k_2} *is redundant (even in the continuous sense) and can therefore be deleted.*

PROOF. Let us show that (19) for S_{k_1} implies that (19) holds true for S_{k_2}. Given that (19) holds for S_{k_1}, we have that

$$\sum_{P \in S_{k_2}} x_P = \sum_{P \in S_{k_1}} x_P + \sum_{P \in S_{k_2} - S_{k_1}} x_P$$

$$\le |S_{k_1}| - 1 + |S_{k_2} - S_{k_1}| = |S_{k_2}| - 1.$$

This completes the proof. □

To illustrate (19) and Proposition 2, consider the example of Figure 3. For this conflict graph, the six pairs of edges yield the following inequalities.

k	Edge Pair	S_k	Corresponding Inequality (19)	
1	(P, Q), (Q, R)	$\{P, Q, R\}$	$x_P + x_Q + x_R \le 2$	(20a)
2	(P, Q), (P, R)	$\{P, Q, R\}$	$x_P + x_Q + x_R \le 2$	(20b)
3	(Q, R), (P, R)	$\{P, Q, R\}$	$x_P + x_Q + x_R \le 2$	(20c)
4	(P, Q), (Q, W)	$\{P, Q, W\}$	$x_P + x_Q + x_W \le 2$	(20d)
5	(Q, R), (Q, W)	$\{Q, R, W\}$	$x_Q + x_R + x_W \le 2$	(20e)
6	(P, R), (Q, W)	$\{P, Q, R, W\}$	$x_P + x_Q + x_R + x_W \le 3$	(20f)

Observe that the clique formed by the nodes $\{P, Q, R\}$ produces the same inequality and that, as predicated by Proposition 2 and because $S_1 \subseteq S_6$, (20f) is implied by (20a). Hence, the nonredundant inequalities produced by this conflict graph are given by (20a), (20d), and (20e).

In general, we have devised an efficient implementation scheme wherein constraints (19) are first generated for all adjacent pairs of edges, recognizing that if a node-triplet forms a clique, then the corresponding pairs of adjacent edges would generate duplicate constraints as in (20a–c). The implemented procedure automatically detects and avoids the generation of such duplicates. For all remaining pairs of edges, a constraint is generated only if it is not already implied by the constraints generated from the adjacent pairs of edges based on Proposition 2. Furthermore, we also ensure that different conflict graphs do not produce duplicated constraints, or constraints that are implied by previously generated inequalities of type (19). This is accomplished by constructing the *overall conflict graph* $G(N, A)$, where N is the set of nodes representing all the flight-plans, and A is the set of arcs such that if flight-plans P and Q conflict (with resolvable severity) at any point in time in the horizon, then we have an edge connecting P and Q. By identifying pairs of edges in G and recording whether or not an inequality of type (19) has been already generated or is implied for each such pair of edges, we efficiently derive a set of nonredundant inequalities of this type. Let $K_{NR} \subseteq K$ be the set of indices k for which such nonredundant members of (19) for the corresponding sets S_k have been explicitly generated and added to the model via this procedure.

2. Two Alternative Model Formulations for the Airspace Planning Problem

In this section, we construct and computationally investigate two alternative model formulations for the described airspace planning problem. The first of these, referred to as the airspace planning model 1 (APM1), incorporates the foregoing equity, workload, and conflict constraints, along with the proposed

objective cost terms, to yield the following mixed-integer 0–1 programming problem.

APM1: Minimize $\displaystyle\sum_{f=1}^{F}\sum_{p \in P_f} c_{fp} x_{fp} + \sum_{s \in S}\sum_{n=1}^{\bar{n}_s} \mu_{sn} y_{sn}$

$$+ \mu_e (x_u^e - x_l^e) + \mu_u^e x_u^e \quad (21a)$$

subject to $\displaystyle\sum_{p \in P_f} x_{fp} = 1 \quad \forall\, f = 1, \dots, F \quad (21b)$

$$\sum_{(f,\,p) \in C_{si}} x_{fp} - n_s \leq 0$$
$$\forall\, i = 1, \dots, M_s, s \in S \quad (21c)$$

$$n_s = \sum_{n=1}^{\bar{n}_s} n y_{sn} \quad \forall\, s \in S \quad (21d)$$

$$\sum_{n=1}^{\bar{n}_s} y_{sn} = 1 \quad \forall\, s \in S \quad (21e)$$

$$x_l^e \leq U_\alpha(x) \leq x_u^e \quad \forall\, \alpha = 1, \dots, \bar{\alpha} \quad (21f)$$

$$x_P + x_Q \leq 1 \quad \text{for all } (P, Q) \in \text{FC} \quad (21g)$$

$$\sum_{P \in S_k} x_P \leq |S_k| - 1 \quad \text{for each } k \in K_{NR} \quad (21h)$$

$$x \text{ binary}, y \geq 0, x_l^e \geq 0, x_u^e \leq \nu_e. \quad (21i)$$

2.1. An Alternative Representation of Conflict Constraints: Model APM2

We can alternatively model the conflict constraints by defining a variable z_{PQ} for each distinct edge (P, Q) in the conflict graph G_{st} for each sector s and time segment t, which takes on a value of 1 if this conflict is permitted and 0 otherwise. Then, we would have a single conflict constraint for each (s, t) that requires

$$\sum_{(P,Q) \in A_{st}} z_{PQ} \leq 1 \quad \forall\, s \in S, \ \forall\, \text{time segments } t. \quad (22a)$$

The new z-variables would then need to be related to the original x-variables via the following constraints:

$$z_{PQ} \geq x_P + x_Q - 1, z_{PQ} \geq 0, \quad \forall (P, Q) \in A_{st},$$

$$\forall s \in S, \text{ and time segments } t. \qquad (22b)$$

This, in effect, would create a linearized version of an essentially a quadratic model based on equating $z_{PQ} = x_P x_Q$ and would also permit the penalizing of different types of conflicts differently in the objective function, if necessary. Note that we do not need the other constraints of the type $z_{PQ} \leq x_P$ and $z_{PQ} \leq x_Q$ that are usually incorporated in such a linearization scheme because in the present context, we simply wish to enforce that $z_{PQ} = 1$ whenever $x_P = x_Q = 1$, and that no more than one z-variable takes on a value of 1. Although reformulation techniques of the type described in Sherali et al. (1998) could be used to further strengthen the representation of the resulting model based on (22), it would also increase its size, so we did not pursue this option at this stage. Let us refer to this alternative formulation in which the constraints (21h) in APM1 are replaced by (22a,b) as model APM2.

2.2. Preliminary Computational Comparison of APM1 and APM2

Naturally, we expect model APM2 to become more favorable as the number of nonfatal conflicts in the model increases. To demonstrate the effect of the density of conflict graphs on the performance of APM1 versus APM2, we generated the following two test sets. The first, P75, contains 15 flights, each having five surrogate flight-plans, traversing the Miami and Jacksonville (ZMA-ZJX) *air route traffic control centers* (ARTCC). The second scenario, P260, contains 52 flights, each having five surrogate flight-plans, traversing the Salt Lake City ARTCC (ZLC). For scenario P75, the run turned out to examine 290 conflict

subgraphs, typically containing between 10 and 130 edges each, while for scenario P260, only six conflict graphs were explored having two to five edges each. Each of these scenarios was solved on a SUN Ultra-1 workstation using CPLEX-MIP 6.5 (CPLEX Optimization, Inc. 1999). Table 1 demonstrates the relative effectiveness of model APM1 versus APM2 for each of these cases. Note that for P75, APM2 efficiently handles the dense conflict constraints and provides a significant computational advantage over APM1. However, for P260, where the conflict constraints are not as restrictive, APM2 is not as effective as APM1. This observation was evident in several other similar test cases. In §4, we provide some additional insights into the efficacy of using APM1 versus APM2 in solving the underlying airspace planning problem.

3. Heuristic Solution Procedure

In this section, we prescribe a polynomial-time heuristic procedure for generating good quality approximate solutions for the airspace planning problem. This heuristic is based on Geoffrion's (1967, 1969) scheme for implementing a depth-first implicit enumeration procedure via a partial solution list *PS* that records the ordered sequence of restrictions imposed on the enumeration tree branches along the path from the root node to the current active node. Whenever the complement of any of these restrictions has been explored and fathomed, the particular restriction is flagged to indicate this fact. In our implementation, we employ a Boolean variable to flag this occurrence, setting it to FALSE when flagged, and TRUE otherwise. Within this framework, we perform customized branching decisions, feasibility analyses, and logical tests to exploit the structure of our problem. By imposing a fixed upper bound N on the number of backtracking steps performed in this implicit enumeration process, we ensure only a polynomial growth

Table 1   Comparison of APM1 Versus APM2 for Two Different Conflict Density Scenarios

	APM1		APM2	
Scenario	Branch-and-Bound Nodes	CPU Time (seconds)	Branch-and-Bound Nodes	CPU Time (seconds)
P75	44	14.76	35	4.23
P260	2271	55.41	2515	63.96

Init: Set backtracks = 0, max_backtracks = N = number of flights. Let the partial solution list PS be empty. Each item in PS will be a pair of elements: a variable and a Boolean value.

Solve the LP relaxation; obtain the solution $\bar{x}$ and the reduced costs $\bar{c}$. Set $\bar{x}_{fp} \leftarrow \lfloor \bar{x}_{fp} \rfloor \ \forall \ f, p$. Let $\mathcal{F} = \{$flight $f : \sum_p \bar{x}_{fp} = 0\}$. For each $f \in \mathcal{F}$, let $L_f = \{$permissible surrogates of flight $f\}$, and for all (f, p) such that $\bar{x}_{fp} = 1$, place (x_{fp}, TRUE) in PS in order of nondecreasing $\bar{c}_{fp}$.

Stop: return solution $\bar{x}$. —Y— $\mathcal{F} = \emptyset$?

Perform logical tests on L_f for all $f \in \mathcal{F}$ adding all new variables fixed at 0 to the end of PS with a FALSE Boolean value.

N

Let $p(f) \in \text{argmin}\{\bar{c}_{fp} : p \in L_f\} \ \forall \ f \in \mathcal{F}$, $f^* \in \text{argmin}\{\bar{c}_{fp(f)}, f \in \mathcal{F}\}$. Set $\bar{x}_{f^*p(f^*)} = 1$, remove f^* from $\mathcal{F}$, and add $(x_{f^*p(f^*)}, \text{TRUE})$ to the end of PS.

N

$L_f = \emptyset$ for any $f \in \mathcal{F}$? —N— $|L_f| = 1$ for any $f \in \mathcal{F}$?

Call Subroutine SUB (Figure 5)

Y

Y

For each f such that $|L_f| = 1$, say $L_f = \{p(f)\}$, put $\bar{x}_{fp(f)} = 1$ and add $(x_{fp(f)}, \text{FALSE})$ to the end of PS and remove f from $\mathcal{F}$.
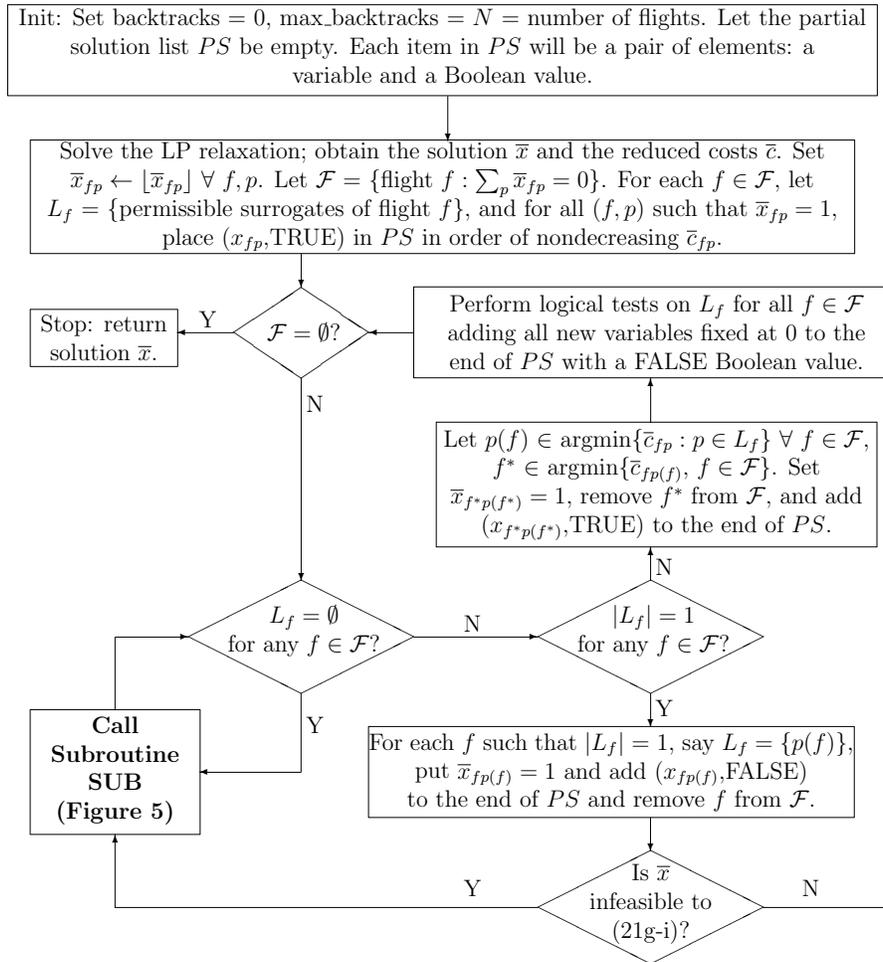
Y— Is $\bar{x}$ infeasible to (21g-i)? —N

Figure 4   MAIN Routine for the Heuristic Procedure

in the overall effort with respect to the size of the problem. In essence, this heuristic is a prematurely terminated customized implicit enumeration procedure, and in concept, can be similarly applied to other classes of problems that are modeled as 0–1 mixed-integer programs using general related techniques as discussed in Nemhauser and Wolsey (1988), for example. A flowchart for the main routine executed by this heuristic is given in Figure 4, in which flight-plans are iteratively selected for each flight, ensuring at each step that feasibility with respect to conflict and equity constraints is maintained. The subroutine SUB, depicted in Figure 5, handles the backtracking details for this procedure.

To initialize the heuristic, we select a maximum allowable number of backtracking steps, N, and create a *partial solution list* (PS) for recording the history of pertinent actions executed during the solution process. Each item in this list contains two elements. The first element is an x-variable that is currently being fixed by the heuristic. The second element is a flag that describes whether the variable was selected by a defined process to be set equal to 1 (TRUE), or if it was necessarily set to 0 or 1 due to feasibility considerations based on previous restrictions (FALSE). To provide an advanced start, the linear programming (LP) relaxation to APM1 (or APM2) is solved. Let $\bar{c}_{fp}$ be the reduced cost coefficient for the variable x_{fp} in
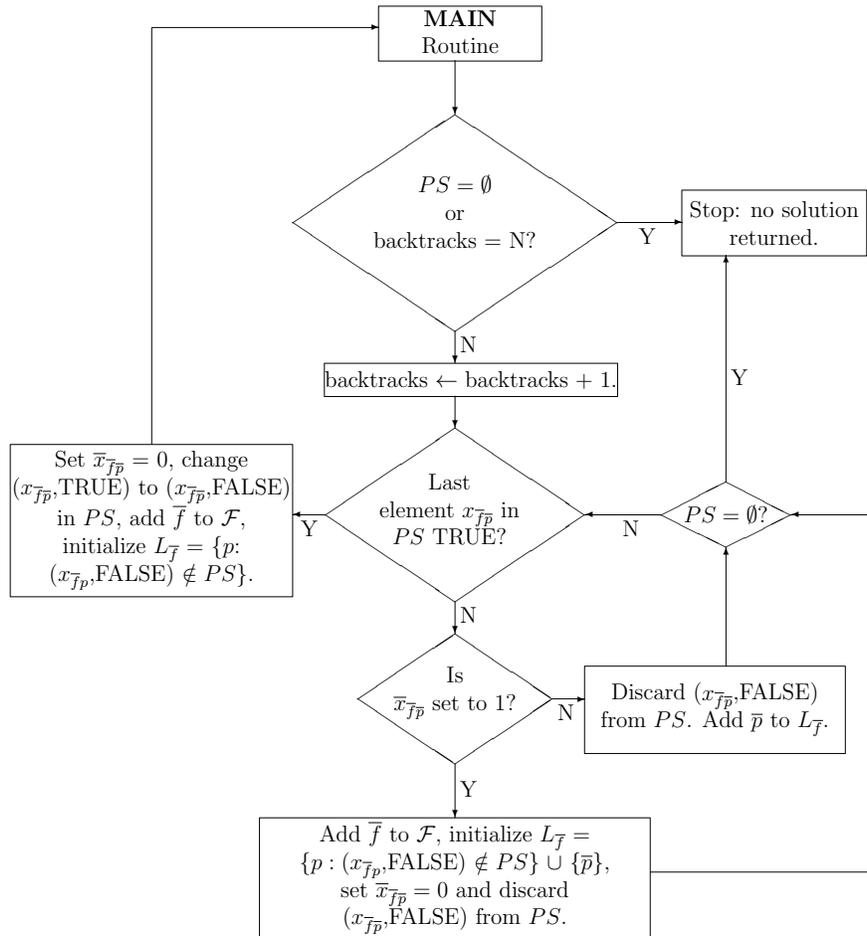
Figure 5   Subroutine SUB for the Heuristic Procedure

this LP solution, $\forall\,(f, p)$. We use these reduced costs to guide the selection of plans for the various flights. To record this selection process, we maintain a partial solution vector $\bar{x}$, where $\bar{x}_{fp}$ is initialized at 1 if its corresponding value in the linear programming relaxation is equal to 1, and 0 otherwise. For each such $\bar{x}_{fp}$ that is fixed at 1, we place x_{fp} in PS with a Boolean value of TRUE. In our implementation, we arrange these items that are introduced concurrently within PS in nondecreasing order of their corresponding $\bar{c}_{fp}$ values to induce the procedure to retain the best mix of flight-plans in its final solution.

Through the remainder of this heuristic, we keep track of the feasibility of the current partial solution $\bar{x}$ with respect to the assignment constraints (21b), the fatal conflict constraints (21g), the nonfatal con-

flict constraints (21h), and the maximum inequity constraint in (21i). The heuristic maintains a running list of nonfatal conflicts that will occur within each conflict subgraph based on the currently selected flight-plans. Each time an additional flight-plan is selected, this list is updated for all conflict subgraphs that contain the selected flight-plan. Similarly, for the equity restrictions, a list of aggregate inequities for the different airlines is maintained and updated with each chosen flight-plan. Using these lists, along with the list of fatal conflicts, logical tests are performed after each solution augmentation step to reduce the list of permissible surrogates for each flight.

The heuristic implicitly tracks (21b) in its main routine via a list L_f that maintains the permissible surrogates for flight $f \in \mathcal{F}$, where $\mathcal{F} = \{f \in \{1, \ldots, F\} :$

Flight f has not yet been assigned a flight-plan}. When $\mathscr{F} = \emptyset$, all flights have been assigned corresponding flight-plans, and the heuristic successfully terminates. Otherwise, we examine all the sets L_f, for $f \in \mathscr{F}$, and proceed based on one of the following three cases as applicable.

Case 1. $L_f = \emptyset$ for some $f \in \mathscr{F}$, implying that the current partial solution $\bar{x}$ is infeasible. The procedure must *backtrack* using the subroutine of Figure 5 and remove the previous assignment(s) that led to this infeasibility (this process is described below).

Case 2. $|L_f| = 1$ for some $f \in \mathscr{F}$, implying that only one feasible choice of a flight-plan remains for flight f. For each f such that $|L_f| = 1$, the lone remaining feasible flight-plan for flight f is chosen and is placed in PS with a FALSE Boolean value. This augmentation of $\bar{x}$ is then examined for feasibility.

Case 3. $|L_f| > 1$ for all $f \in \mathscr{F}$. In this case, we select a feasible flight-plan corresponding to an unassigned flight having the least $\bar{c}$-value and place it in PS with a TRUE Boolean value.

After each flight-plan assignment augments $\bar{x}$ in Cases 2 and 3, we must update the current conflict and inequity lists, eliminate potential fatal conflicts, and revise $L_f \ \forall f \in \mathscr{F}$. If infeasibility with respect to the previously selected flight-plans is detected in this process, the procedure must backtrack. This main routine continues until a solution is found, or the maximum permissible number of backtracking steps is executed.

Note that in our implementation, we have used the reduced costs $\bar{c}_{fp} \ \forall (f, p)$ obtained from the LP relaxation to guide the selection of flight-plans. Alternatively, we could have used the actual costs c_{fp} of the flight-plans themselves, or any other cost index. This would allow the user to skip the LP-solution phase altogether if necessary (the partial solution $\bar{x}$ would be initialized as a null vector in this case). However, using the LP-based information often provides a good advanced start, and the reduced costs tend to provide a more accurate measure of the influence of selecting flight-plans on the overall objective value.

The *backtracking subroutine* (SUB) examines the list PS to determine the most recent TRUE assignment that is responsible for the infeasibility. The items in PS that have a FALSE Boolean value correspond to variables that have either been necessarily fixed at 0 by logical tests, or have been set equal to 1 because no other flight-plan choices for that flight were available. The procedure removes each FALSE item of this type from PS in a last-in-first-out fashion (given that the last item was FALSE), by releasing the variables that were fixed at 0 to be considered again by the procedure, and by unassigning the variables fixed at 1, until a TRUE item is found. Once a TRUE item is found, the flight corresponding to this item is inserted into $\mathscr{F}$, and the corresponding flight-plan variable is fixed at 0 (because setting it to 1 resulted in an infeasibility). This item therefore remains in PS, but now with a FALSE Boolean value. The overall logic of this process is displayed in Figure 5.

We may extend this heuristic to additionally perform a limited number of branch-and-bound steps to improve the quality of the solution obtained. Let N' be the number of backtracking steps executed by the heuristic. The extended version of this procedure provides APM1 (or APM2) with the derived heuristic solution as an incumbent upper-bounding solution, and then solves APM1 via CPLEX-MIP with a limit of $N - N'$ on the number of branch-and-bound nodes enumerated. In our computational experiments, we will refer to the basic construction heuristic procedure as HP, and to the foregoing enhanced heuristic procedure as EHP.

To assess the complexity of the heuristic HP, note that the initial step involves the solution of the LP relaxation of the problem. Depending on whether we use APM1 or APM2 and assuming that we employ a polynomial-time LP solver, let $O(g)$ represent the complexity of this step, where g denotes an associated appropriate polynomial in the size of the problem. Furthermore, respectively denoting by m_v and m_r the number of variables and restrictions in the model formulation, the complexity of the node analysis embodied in the process of Figure 4 (including the feasibility and the logical tests), is of order $O(m_v m_r)$. Noting that the maximum depth of a path in the enumeration tree is given by the number of x-variables, and that we examine at most N (overlapping) paths, the number of nodes analyzed is bounded above by $O(m_v N)$. Hence, for a fixed N, the heuristic HP is of (polynomial) complexity $O(g + m_v^2 m_r N)$. (In the case of EHP,

where LP relaxations might be solved at each node in the subsequent enumeration process, this complexity is bounded above by $O(m_v N[g + m_v m_r])$.) While this is a worst-case performance analysis, the proposed heuristic consumes a moderately reasonable effort in practice, while producing solutions within 0.01% of optimality as demonstrated in the next section.

4. Computational Results

In this section, we study the performance of applying CPLEX-MIP Version 6.5 to the models developed in §1 as well as evaluate the heuristic procedure developed in §3. Toward this end, we generated a set of 12 test problems based on information derived from the FAA-ETMS (Enhanced Traffic Management System) database over the Miami (ZMA) and Jacksonville (ZJX) ARTCCs. The number of flights for these problems ranged from 50 to 200 in increments of 50, and instances were generated for each of these flight-mix scenarios using 3, 4, and 5 surrogate flight-plans for each flight. The flights chosen span the time interval from 1:15 a.m. to 7:00 p.m. A distribution of the flight times is given in Figure 6, where each histogram bar corresponds to a 30-minute time interval and the height of the bar represents the number of flights active at some point during that time

segment. For each instance, the cost of each flight-plan was given by its direct fuel cost that was computed using the Base of Data (BADA) Eurocontrol Database. This database provides information on fuel costs as a function of the flight altitude and aircraft speed, where both of these parameters are specified in the ETMS database. The sector workload costs (see Figure 2) were assumed to increase according to a defined quadratic function. The measure of inequity for each flight was given by its normalized excess delay duration, see (9), and the objective coefficients in (7) were taken as appropriate cost-conversion factors. For these scenarios, we defined our SUA as a cylindrical region that approximates the volume of enroute airspace around the Cape Canaveral spaceport. The midpoint of each SUA enforcement duration interval was taken to be 10:00 a.m. The surrogates for each flight were generated by equitably varying the departure time of the flight, and then correspondingly determining the shortest-detour-based rerouting of the flight around the SUA, as necessary. Given this generated mix of flight-plans, the models AOM and AEM of Sherali et al. (2000) were run to respectively ascertain sector occupancy durations, and collision risk intervals along with their associated severity levels. This information was then used to generate the corresponding workload and collision resolution management constraints. On an IBM Pen-
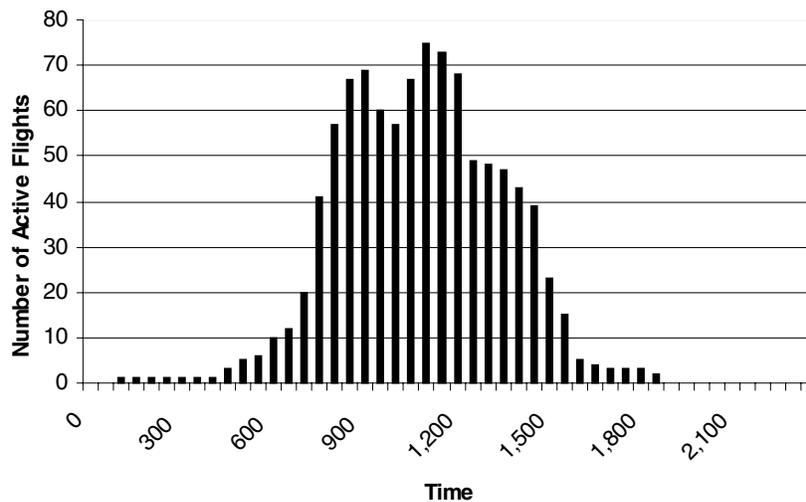

Figure 6 **Relative Time Distribution of the Selected 200 Flights**

tium III 333 computer using MATLAB 5.3, the effort required to generate this information via AOM and AEM ranged from roughly 20 minutes, for problems having 150 total flight-plans, to 2 hours, for problems having 1,000 total flight-plans. (This effort can be significantly reduced by using a compiled language (such as C/C++) as opposed to MATLAB.)

First, we examined the efficacy of model APM1 versus model APM2. For each of the 12 scenarios, we recorded the number of branch-and-bound nodes enumerated (Nodes), the CPU time in seconds (CPU Time), and the objective function value (Obj. Value). Because this problem set was based on actual recorded flight data, few conflicts occurred between flight-plans. As in Table 1, we would expect APM1 to outperform APM2 on such a data set, and Table 2 confirms this anticipation. Although APM1 exhibits only a slight computational advantage over APM2 in most cases, as demonstrated by the 200 flight scenario instances having three or five surrogate flight-plans per flight, APM1 occasionally greatly outperforms APM2. Note also that the optimal objective function value decreases somewhat as more surrogates become available for each set of three instances having 50, 100, 150, and 200 flights, respectively.

Next, we considered the heuristic developed in §3. For each of the data sets described above, we ran the variants HP and EHP and obtained their solutions, recording their execution times and the number

of backtracking steps performed. Model APM1 was selected for deriving the LP-based starting solution and the reduced costs for HP, and for implementing the limited branch-and-bound exploration for EHP. As seen in the tables, the solution provided by HP is within 1% of optimality for 9 out of the 12 test cases. (Here the % *optimality* is measured as the percentage gap between the objective value of the best solution found and the optimal solution value recorded in Table 2.) In the first test case, the linear programming solution turned out to be integer-valued, and thus the heuristic terminated with this optimal solution in its initialization phase. Note that the number N' of backtracking steps actually performed prior to termination is usually very small and, hence, the possibility of an unsuccessful termination due to the maximum allowable backtracking limit was never imminent. The enhancement EHP provided an optimal solution in 8 out of the 12 test cases, and provided a solution within 0.01% of optimality in the other 4 test cases. These results show that EHP is a promising alternative to solving the mixed-integer programming model to optimality, and might be especially useful for large instances of the airspace planning problem. We comment here that although HP can also be used to find an incumbent solution for the exact branch-and-bound method, our experimental experience revealed that the time consumed in executing this heuristic offsets the gains achieved by utilizing this initial upper bound.

**Table 2   Comparison of APM1 Versus APM2**

Scenario:		APM1:		APM2:		
Flights	Surrogates	Nodes	CPU Time	Nodes	CPU Time	Obj. Value
50	3	0	0.71	0	0.99	175852.64
50	4	4	0.9	4	1.33	173262.40
50	5	3	1.14	3	1.63	172070.34
100	3	3	1.24	3	1.71	314482.50
100	4	6	1.75	6	2.23	309787.65
100	5	218	18.19	218	24.68	307920.21
150	3	35	4.32	34	7.87	480031.62
150	4	20	6.28	21	10.41	472970.44
150	5	577	252.1	750	367.09	470370.34
200	3	49	17.89	4648	585.32	623677.20
200	4	3755	1176.71	3846	1523.7	611747.83
200	5	485	339.87	2208	2046.9	607498.12

Table 3   **Effectiveness of the Proposed Heuristic Procedures**

Flights	Surrogates	HP: % Optimal	HP: CPU Time	N'	EHP: % Optimal	EHP: CPU Time
50	3	0	0.01	0	0	1.01
50	4	0.01	0.05	0	0	1.33
50	5	0.24	0.07	0	0	1.54
100	3	2.64	0.39	1	0	2.04
100	4	0.60	0.95	1	0	2.64
100	5	0.42	1.57	30	3.87×10^{-3}	11.49
150	3	2.48	3.24	1	0	7.41
150	4	0.26	9	1	0	17.02
150	5	7.58	7.9	9	5.78×10^{-3}	166.41
200	3	0.63	20.14	1	0	42.36
200	4	0.39	40.02	2	9.15×10^{-3}	209.64
200	5	0.37	76.64	1	7.06×10^{-4}	287.48

Finally, we demonstrate the usefulness of the proposed airspace planning model as a strategic planning tool for the FAA. In particular, we explore the sensitivity of the solution provided by the model to changes in the radius of the SUA formulated around the spaceport, and to the duration of the launch-window during which the SUA is activated. To illustrate the effect of these parameters, we chose the set of data containing 100 flights. First, we generated three surrogate flight-plans for each flight based only on varying the departure times (assuming no SUA restrictions), and we ran APM1 to determine an optimal set of flight-plans for this problem instance. Using the resulting optimal 100 flight-plan data set, we determined the number of these flight-plans that would intersect SUAs having different radii and launch-window durations. Specifically, we examined radii of R (the original radius of the SUA), $R/2$, and $R/4$, and we varied the launch-window durations between 0 and 12 hours in intervals of 2 hours. The results of this experiment are displayed in Figure 7. By cutting the SUA radius in half, the number of flights impacted by the SUA is reduced by about 25%, while reducing the SUA radius to one-fourth its original size cuts the number of flights impacted by about 33%. Also, note that increasing the duration of the launch-window for any given SUA radius tends to affect the number of impacted flights almost linearly until the duration
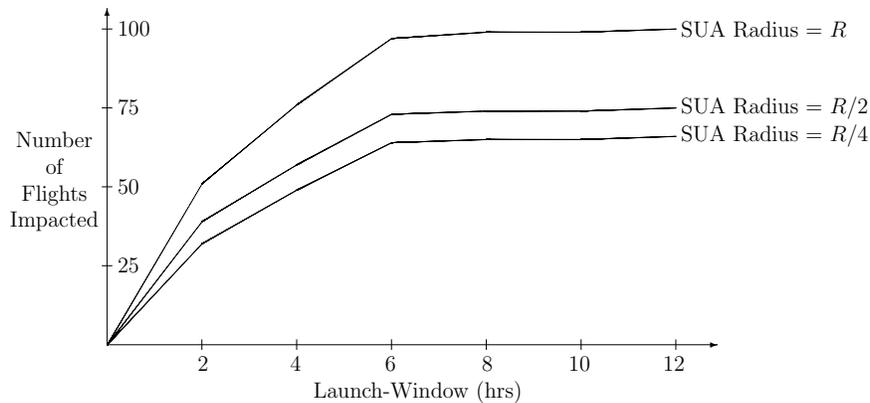
Figure 7   **Number of Flights Impacted by the SUA for Different Radii and Launch-Window Durations**

reaches 6 hours. After this point, the number of flights impacted by the SUA is relatively insensitive to an increase in the launch-window duration. This is due to the nature of the particular data set used, whereby most of the 100 flights considered have crossed the SUA region within the six hour time-window duration. In general, the number of flights impacted would depend on the space-time distribution of flights, as well as on the SUA volume, timing, and activation duration.

We then evaluated the effect of these parameters on the total extra system cost as given by APM1. For this experiment, we examined the cases of launch-windows having durations of 4, 5, and 6 hours and generated problem instances having SUA radii equal to R, $3R/4$, $R/2$, and $R/4$. The flight data used was composed of the 100 flight-plan scenario described above, modified by generating three surrogate flight-plans based on adjusting the departure times and by rerouting the aircraft via efficient detour paths around the SUA. The total extra system cost for any instance based on a specified SUA enforcement is thus given by the difference between the optimal objective function value of this instance minus the cost of the system with no SUA enforcement. The results of this experiment are depicted in Figure 8. Note that the approximate quadratic increase in extra system cost due to an increase in SUA radius is indicative of a direct proportionality between the extra system cost and the SUA area. A best quadratic fit for this output

was determined via the equation $C = \beta_t r^2$, where C is the total extra cost of the scenario, r is the fraction of the radius R of the SUA, and β_t is a suitable constant of proportionality for a launch-window of duration t, for $t = 4, 5, 6$. The respective values of β_t were obtained as $\beta_4 = \$49,278.3$, $\beta_5 = \$55,643.7$, and $\beta_6 = \$65,444.4$. The goodness-of-fit R^2-statistic values for the cases of 4, 5, and 6 hour launch-window durations were computed as 0.991, 0.988, and 0.990, respectively, indicating that these results are very well described by the stated quadratic equations.

5. Conclusions

In this paper, we have proposed 0–1 mixed-integer airspace planning models (APM1 and APM2) to provide guidelines in a collaborative decision-making process between the FAA and commercial airlines. These models seek to select flight-plans that satisfy various equity, workload, and safety considerations under different airspace scenarios. Computational results indicate that model APM1 is preferable in most instances of the airspace planning problem, although the alternative model APM2 might become more effective when several nonfatal conflicts occur between flight-plans. We have also devised an effective (polynomial-time) heuristic technique that was shown to identify optimal solutions for 8 out of 12 instances of a particular set of real test problems,
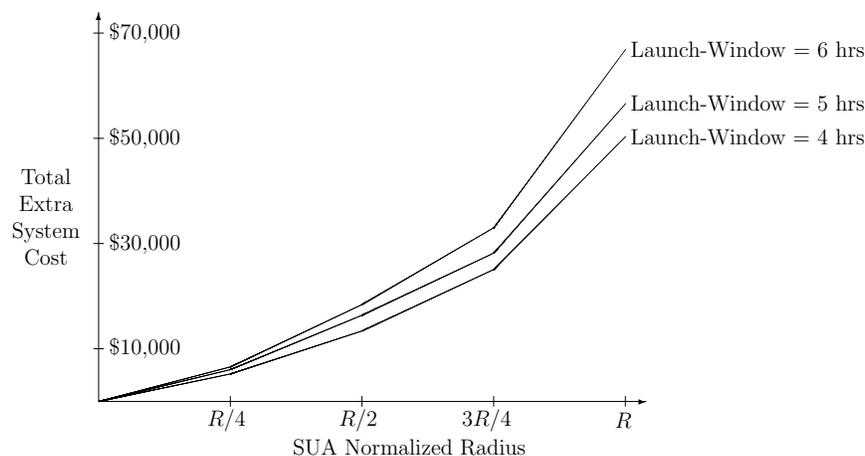

Figure 8   Total Extra System Cost due to SUAs Having Different Radii and Launch-Window Durations

while finding feasible solutions having objective function values within 0.01% of the corresponding optimal value for the remaining 4 instances. Further modeling refinements of the workload, equity, and collision risk issues, as well as more extensive computational tests, are recommended for future research.

The proposed model can be utilized in one of two ways.

(a) *Air-Traffic Management: Generator of a suitable mix of flight-plans for a set of flights.* In this role, the model can be coordinated with NASPAC, a large-scale simulation model for analyzing airport operations related to a given set of flight-plans, by using the latter simulation package to evaluate in more detail the airport operations pertaining to the prescribed solution suggested by the model.

(b) *Policy Evaluator:* Various what-if scenarios can be evaluated by policy/decision makers in formulating operational guidelines. In particular, the following types of investigations can be considered.

(i) The effect of alternative flight-plans, or protocols/directives for generating such flight-plans can also be evaluated using this model. In fact, this model can itself serve to evaluate the efficacy of various flight-plan-generation programs.

(ii) Similar to (b), different aircraft separation regulations imposed by the FAA might yield different interpretations on what poses a "conflict." These policies could be evaluated by translating them into appropriate constraints of the type (15a–c) and examining their effect on the model solution.

(iii) Alternative restrictions with respect to volume and duration on the cordoning of airspace around the RLV spaceport during launches could be evaluated with respect to this model. Different airspace restrictions would yield different values of cost coefficients in the model based on fuel and delay computations. In addition, one might develop certain measures of safety and incorporate appropriate penalties in the objective cost coefficients to reflect the relative safety of trajectories with respect to RLV operations. The effect of launch timings and frequencies could also be studied in this context.

(iv) The effect of various ATC-maximum-sector workload policies can be evaluated with respect to their influence on system performance, along with associated costs. Alternative measures of, and controls on, workload can also be investigated.

Hence, the model can be used both in a tactical decision-making mode as an air-traffic management tool, as well as for formulating strategic guidelines and policies.

Acknowledgments

References
Associate Administrator for Commercial Space Transportation (AST). 1998. 1998 LEO commercial market projections. *Federal Aviation Admin.* (May).

Bertsimas, D., S. S. Patterson. 1998. The air traffic flow management problem with enroute capacities. *Oper. Res.* **46** 406–422.

——, ——. 2000. The traffic flow management rerouting problem in air traffic control: A dynamic network flow approach. *Transportation Sci.* **34** 239–255.

CPLEX Optimization, Inc. 1999. *Using the CPLEX Callable Library, Version 6.5.*

Erzberger, H., B. D. McNally, M. Foster. 1999. Direct-to tool for enroute controllers. *ATM '99: IEEE Workshop on Adv. Techn. Impact on Air Traffic Management in the 21st Century*, Capri, Italy, September 26–30.

Falker, J. M. 1999a. Conflict analysis of air and space modes of transportation. Phase III Interim Report, NEXTOR Research Report RR-99-05, Massachusetts Institute of Technology, Cambridge, MA.

——. 1999b. Integration of reusable launch vehicles into air traffic management. NEXTOR Research Report RR-99-05, Massachusetts Institute of Technology, Cambridge, MA.

Federal Aviation Administration/Eurocontrol. 1998. A concept paper for separation safety modeling. Technical report (March).

Geoffrion, A. M. 1967. Integer programming by implicit enumeration and Balas' method. *SIAM Rev.* **9** 178–198.

——. 1969. An improved implicit enumeration approach for integer programming. *Oper. Res.* **17** 437–454.

Jeppesen Sanderson, Inc. 1996. *FliteStar User Manual* (*Mentor Plus*). (October).

Lindsay, K. S., E. A. Boyd, R. Burlingame. 1993. Traffic flow management modeling with the time assignment model. *Air Traffic Control Quart.* **1** 255–276.

McNally, B. D., R. E. Back, W. Chan. 1998. Field test evaluation of the CTAS conflict prediction and trial planning capability. *AIAA Guidance, Navigation, and Control Conf.*, Boston, MA, August 10–12.

NAS Concept Development (ASD). 1998. RAMS-OPGEN integration activities: Fiscal year 1998. Internal FAA/NASA Document, Federal Aviation Administration, Washington, D.C.

Nemhauser, G. L., L. Wolsey. 1988. *Integer and Combinatorial Optimization*. John Wiley & Sons, Inc., New York.

Odoni, A. R., J. Bowman, D. Delahaye, J. J. Deyst, E. Feron, R. J. Hansman, K. Khan, J. K. Kuchar, N. Pujet, R. W. Simpson. 1997. Existing and required modeling capabilities for evaluating ATM systems and concepts. Final report: Modeling Research

Under NASA/AATT. International Center for Air Transportation, Massachusetts Institute of Technology, Cambridge, MA, March.

Sherali, H. D., E. L. Brown. 1994. A quadratic partial assignment and packing model and algorithm for the airline gate assignment problem. *DIMACS Ser. Discrete Math. Theoretical Comput. Sci., Quadratic Assignment and Related Problems* **16** 343–364.

——, W. P. Adams, P. J. Driscoll. 1998. Exploiting special structures in constructing a hierarchy of relaxations for 0-1 mixed integer problems. *Oper. Res.* **46** 396–405.

——, J. C. Smith, A. A. Trani, S. Sale. 2000. National airspace sector occupancy and conflict analysis models for evaluating scenarios under the free-flight paradigm. *Transportation Sci.* **34** 321–336.