Rerouting Aircraft for Airline Recovery

Jay M. Rosenberger, Ellis L. Johnson, George L. Nemhauser,

Please scroll down for article—it is on subsequent pages

INFORMS is the largest professional society in the world for professionals in the fields of operations research, management
science, and analytics.
For more information on INFORMS, its publications, membership, or meetings visit http://www.informs.org

Rerouting Aircraft for Airline Recovery

Jay M. Rosenberger • Ellis L. Johnson • George L. Nemhauser

American Airlines Operations Research and Decision Support Department, MD 5358, P.O. Box 619616,
Dallas/Fort Worth Airport, Texas 75261-9616
School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332-0205
School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332-0205
jay.rosenberger@aa.com • ejohnson@isye.gatech.edu • gnemhaus@isye.gatech.edu

Disruptions in airline transportation systems can prevent airlines from executing their schedules as planned. Adverse weather conditions, congestion at airports, and mechanical failures often hinder a flight schedule. During such events, decision makers must reschedule flight legs, and reroute aircraft, pilots, and passengers. We present an optimization model that reschedules legs and reroutes aircraft by minimizing an objective function involving rerouting and cancellation costs. We develop a heuristic for selecting which aircraft are rerouted, and we provide proof of concept by evaluating our model using a simulation of airline operations. Finally, we revise the model to minimize crew and passenger disruptions.

Airlines frequently encounter disruptions that prevent them from operating as planned. Severe weather conditions, such as icing on a runway, can close an airport. Unscheduled maintenance problems with an aircraft can require days to repair. When disruptions occur, the airlines must reschedule flight operations, and an airline's *recovery policy* determines which flight legs to delay and cancel and how to reroute the aircraft, pilots, and passengers. Airlines typically recover from disruptions in stages. The first stage, *aircraft recovery*, reroutes the aircraft and delays and cancels flight legs. During the second recovery stage, *crew recovery*, the airline assigns pilots to the uncanceled flight legs by rerouting the regularly scheduled pilots and calling on reserve pilots. Finally, *passenger recovery* reroutes the passengers. Recent literature describes integrated or hybrid models that solve aircraft, crew, and passenger recovery problems. Lettovský (1997) presents a multifleet integrated recovery model, and Clarke (1997a, b) solves a multifleet aircraft recovery model that minimizes lost passenger revenue on single flight leg itineraries and enforces crew availability constraints.

In this paper, we present an optimization model for aircraft recovery (ARO). In §1, we describe the problem and how the airlines can use it in practice. In §2, we discuss relevant literature and our contributions. Section 3 describes an integer program for aircraft recovery. In §4, we give a heuristic for selecting a subset of aircraft that are considered in the recovery model. In §5, we present computational results that demonstrate that the results from solving our model are significantly better than those of traditional models in terms of cancellations, on-time performance, and disrupted passengers. Section 6 describes a revised aircraft recovery model that minimizes disrupted crews and passengers.

1. Problem Description

A *station* is an airport that an airline serves, and a *leg* has an origin station, a destination station, a departure time, and an arrival time. A *route* is a sequence of legs, and prior to each disruption, the aircraft are scheduled to fly a set of *initial routes*. Upon the realization of a disruption during which the initial aircraft routes become infeasible, ARO provides a new route for each aircraft. The new routes must comply with the airline's and the FAA's rules that require each aircraft to receive periodic maintenance service. A route that satisfies all of these rules is *maintenance feasible*.

In this paper, we show how ARO solves two classes of disruptions. *Aircraft disruptions* arise when a plane is not available to fly at least one of its assigned flight legs. Unscheduled maintenance problems and prolonged in-flight delays induce aircraft disruptions. Severe weather and airport congestion provoke *station disruptions* that reduce the number of landings and takeoffs allowed. In fact, weather accounts for approximately 75% of airline disruptions (Dobbyn 2000). During inclement weather and congestion at an airport, the Federal Aviation Administration (FAA) increases the time between consecutive landings. The FAA has experimented with several programs in an effort to reduce the rate of arrivals at an airport.

In the early 1980s, the FAA began ground delay programs (GDPs) that allocate arrival time periods, or *slots*, for each flight leg landing at a specific airport. Subsequently, the airlines could propose reassigning legs to the arrival slots. However, there are restrictions in reassignment, and the central flow control division of the FAA must approve the airline's revised schedule (Vasquez-Marquez 1991). The primary advantage of a GDP is that those legs arriving at a disrupted airport are delayed prior to takeoff. Consequently, air traffic control (ATC) does not force the aircraft to circle the airspace of the disrupted airport, which endangers passengers and increases the cost of fuel and pilots. Since the introduction of GDPs, the FAA has removed most of the reassignment restrictions, and airlines will likely assign legs to slots freely within a few years (Federal Aviation Administration 1999, Andreatta et al. 2000, Carlson 2000, Chang et al. 2001, Metron, Inc. 2001). In this paper, we make:

ASSUMPTION 1. *The airlines assign legs to a given set of arrival slots*.

Even though some disruptions are severe enough to impose slots on an airline flight schedule, other disruptions, such as mishandled luggage and overbooked flight legs, do not warrant such action. During these disruptions, a heuristic might provide a quick solution. *Push-back* is a recovery heuristic that delays legs until their assigned aircraft and crews are ready to fly. Figure 1 depicts a *disruption decision process* in which a decision maker must choose whether a disruption is severe enough to reroute the aircraft.
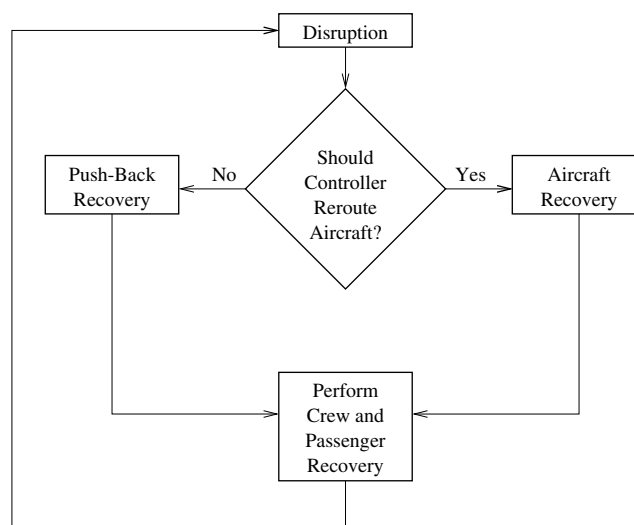

Figure 1   Disruption Decision Process

Consider the following *delay threshold policy*. Upon the realization of a disruption, estimate the delays incurred by using push-back. If the maximum estimated delay is over a threshold value, use ARO to construct new routes for the aircraft. Determining the optimal delay threshold is nontrivial, and we will consider the threshold as a parameter.

Aircraft of the same *fleet* type have approximately the same passenger capacities, ranges, and velocities. Airline planners assign fleet types to legs considering passenger demand and distance for each leg. Moreover, crews are limited to flying only a few fleets, and the crew-scheduling problem is separated by fleet type. Airlines will reassign fleet types as late as the morning of a leg if demand changes, and they find a feasible crew reassignment (Berge and Hopperstad 1993, Talluri 1996). The airlines typically avoid reassigning fleets during operational disruptions because reassignment could complicate crew and passenger recovery. On occasions in which an airline reassigns fleet types, it solves aircraft recovery within the same fleet as a subproblem, and reroutes crews and passengers (Lettovský 1997). ARO's structure allows for multiple fleet types; however, because it does not reroute crews and passengers, we make:

ASSUMPTION 2. *ARO reroutes aircraft only from the same fleet*.

Assumption 2 requires an amendment to Assumption 1 because arrival slots are infrequently specific to fleet types. We assume that a decision maker assigns slots to fleet types prior to solving ARO; that is, Assumption 1 is valid within each fleet.

In practice, recovery is implemented in the airline operations control center (AOCC), and most decisions are made intuitively by the controllers (Clarke 1997c). Controllers have several options when they encounter disruptions, which include delaying and cancelling legs, and rerouting the aircraft. During severe disruptions, operations controllers will occasionally have an aircraft fly without passengers (*ferrying*), to an alternate airport (*diverting*), or to another scheduled destination (*over-flying*). Such operations are expensive and disrupt passenger itineraries. Consequently, the airlines try to avoid such recovery policies in practice, and our implementation ignores them even though our model could include them.

EXAMPLE 1. Figure 2 displays two planes and their initial routes. The solid lines represent flights with their numbers below them. The horizontal dotted lines indicate time between legs, and the location is given by the three letter airport code above the lines. The vertical dotted lines depict the time shown below them.

Upon the arrival of Flight 14 into Madison (MSN), consider an unscheduled maintenance disruption that prevents Plane A from flying until Monday at 19:00. A solution to the disruption is to cancel Flights 15 and 16 and have Plane A continue to fly Flight 17. Another solution is to cancel Flights 11 and 12, assign Plane A

to Flights 13 and 24, assign Plane B to Flights 15, 16, 17, and 28, and return the planes to their original routes before the departures of Flights 21 and 25 on Tuesday morning at MSN.

2. Literature Review and Contributions

As disruptions become more frequent in airline operations, the demand for good automated recovery tools to assist decision makers becomes more significant. Although very few airlines use automated recovery policies, there are many papers on this topic. Related literature on aircraft disruptions includes Teodorović and Guberinić (1984), Teodorović and Stojković (1990, 1995), Jarrah et al. (1993), Mathaisel (1996), Rakshit et al. (1996), Yan and Yang (1996), and Thengvall et al. (2000). As GDPs continue to evolve and allow more freedom to the airlines, new models on station disruptions appear. In the earlier work, not only do the airlines recover from a GDP, but the FAA uses decision support to assign legs to slots. Literature on airline recovery in the event of a GDP includes Vasquez-Marquez (1991), Luo and Yu (1994a, b; 1997), and Cao and Kanafani (2000). Richetta and Odoni (1993), Vran et al. (1994), and Hoffman (1997) assign slots to each airline for the FAA. Although these articles are specific to older versions of GDPs, the airlines and the FAA can use them during modern GDPs with simple modifications. More recent literature, such as Andreatta et al. (2000) and Carlson (2000), assume that airlines are unrestricted when assigning legs to
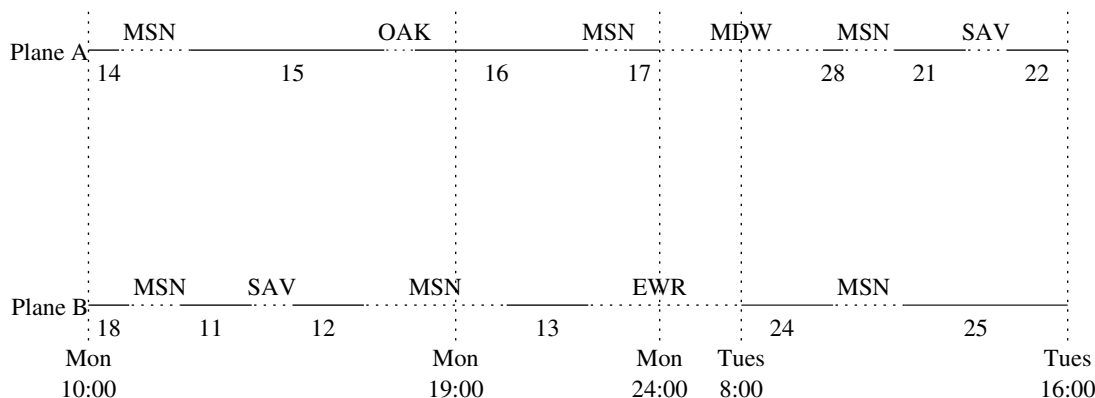

Figure 2   Example Initial Routes

slots. Yan and Lin (1997) describe a model for airport closures, and models in Berge et al. (1993), Talluri (1996), and Yan and Tu (1997) reassign fleets when flight demand changes.

Many recovery models consider only the next leg, or a small subset of legs at a single station, after a disruption. These models represent the cost of a cancelled or delayed leg as a single value. However, cancelling a leg could require an airline to cancel several additional legs or to ferry an aircraft. Similarly, rerouting aircraft for future legs could reduce the downstream impact of a delay. Therefore, we believe that a solution to a model should address these issues. Other recovery models solve a network flow problem that ignores maintenance constraints. In earlier literature, recovery models successively determine a route for each aircraft. Clarke (1997a, b) constructs new routes for each aircraft in a subproblem that considers single leg passenger itineraries, and uses constraints for crew availability in the master problem.

Integrated recovery models reassign fleets to legs, and solve aircraft, crew, and passenger recovery simultaneously (Lettovský 1997). However, exact algorithms for the integrated recovery are intractable. Lettovský (1997) proposes a model for aircraft recovery for each fleet as a subproblem but provides no computational results.

ARO generates possible new routes for each aircraft a priori, and then determines the optimal set of routes. This is similar to the model in Clarke (1997a, b) and the aircraft recovery component of the integrated model presented by Lettovský (1997). The primary advantage of these models is that an AOCC controller can exclude undesirable routes, such as high risk or maintenance infeasible ones, prior to optimization. However, such models are computationally intensive. In Clarke (1997a, b), most of the computational instances use a relatively small fleet of 49 aircraft and 201 legs, and CPU times are not given. Our contribution includes an aircraft selection heuristic (ASH) for ARO, which selects a subset of aircraft for optimization prior to generating new routes. Consequently, we can solve many large recovery instances quickly.

Traditional aircraft recovery models are tested on a small set of event disruptions. United Airlines implemented the model from Jarrah et al. (1993) and Rakshit et al. (1996). Clarke (1997a) provides a proof of concept by simulating a few scenarios of irregular operations. Our contribution includes an evaluation of ARO that is more rigorous than that of traditional models. We simulated 500 days of airline operations using a stochastic model from Rosenberger et al. (2002) and ARO to reroute the aircraft.

Integrated and hybrid recovery models include crew and passenger recovery when rerouting aircraft. We provide a revised aircraft recovery model that minimizes disruptions to crew pairings and passenger itineraries.

3. Model

We model the aircraft recovery problem as a set-packing problem in which each leg is either in exactly one route or cancelled. Consider a set of aircraft P, a set of disrupted aircraft $P^* \subseteq P$, and a time horizon (t_0, T). For each $p \in P$, let $r(p)$ be the initial route of aircraft p, and let $F = \bigcup_{p \in P} r(p)$ be the set of all legs in the initial routes. For each $f \in F$, let b_f be the cost of cancelling leg f, and let

$$K_f = \begin{cases} 1 & \text{if leg } f \text{ is cancelled,} \\ 0 & \text{otherwise.} \end{cases}$$

Typically, b_f would be related to revenue lost from cancelling leg f. For each aircraft $p \in P$, let $R_{(p, F)}$ be the set of maintenance feasible routes of aircraft p that can be constructed from legs in F. For each route $r \in R_{(p, F)}$, let c_r be the cost of assigning Route r to aircraft p, and let

$$X_r = \begin{cases} 1 & \text{if Route } r \text{ is assigned to aircraft } p, \\ 0 & \text{otherwise.} \end{cases}$$

For example, c_r could include penalties for scheduling delays imposed by Route r, plane swaps, and any other factors that yielded differences from the original schedule. Determining costs b_f and c_r can be challenging. However, anecdotal evidence suggests that operations controllers minimize cancellations, delay minutes, and total delays sequentially until they find a unique optimal solution. Controllers could adjust b_f and c_r according to similar priorities.

Let U be the set of allocated arrival slots. For each slot $u \in U$, we use the FAA practice of allowing only one landing at a station within a time period, and we let R_u be the set of routes that include a leg that lands in arrival slot u. In addition to slots restrictions assigned by the FAA, an airline will sometimes reduce the number of legs in its flight schedule by imposing a capacity constraint over some time horizon. Let A be the set of such capacity constraints. For each capacity constraint $a \in A$, we restrict the number of landings at a station within a time period to capacity α_a. Let R_a be the set of routes that include legs that land during the time period of capacity constraint a, and for each route $r \in R_a$, let $H(r, a)$ be the set of legs in r that impact constraint a.

The aircraft recovery integer program for ARO(P) is:

$$\text{Min} \sum_{p \in P} \sum_{r \in R_{(p, F)}} c_r X_r + \sum_{f \in F} b_f K_f \qquad (1)$$

$$\sum_{r \in R_{(p, F)}} X_r = 1 \qquad \forall p \in P \qquad (2)$$

$$\sum_{r \ni f} X_r + K_f = 1 \qquad \forall f \in F \qquad (3)$$

$$\sum_{r \in R_u} X_r \leq 1 \qquad \forall u \in U \qquad (4)$$

$$\sum_{r \in R_a} |H(r, a)| X_r \leq \alpha_a \qquad \forall a \in A \qquad (5)$$

$$X_r \in \{0, 1\} \qquad \forall r \in R_{(p, F)}, p \in P \qquad (6)$$

$$K_f \in \{0, 1\} \qquad \forall f \in F. \qquad (7)$$

Objective (1) is the cost of assigning routes to aircraft and the cost of cancelling the unassigned legs. The *assignment constraints* (2) assign one route to each aircraft, and the *packing constraints* (3) ensure that each leg is either in a route or is cancelled. Constraints (4) and (5) are the *slot constraints* and the capacity constraints, respectively. Constraints (6) and (7) require integral solutions. Lettovský (1997) describes the aircraft recovery problem that minimizes cancellations, $c = 0$ and $b = 1$.

Prior to solving the integer program, ARO(P) generates the set of routes $R_{(p, F)}$ for each $p \in P$. In order to ensure that an aircraft p can fly an assigned route in $R_{(p, F)}$, and the planes can fly legs that arrive after

time T, we impose conditions upon the new routes. For each aircraft $p \in P$, let $n(p) + 1$ be the number of legs in the initial route $r(p)$ of aircraft p, let $(f_1(p), \ldots, f_{n(p)-1}(p))$ be the set of legs that depart after time t_0 and arrive before time T, let *commencing flight* $f_0(p)$ be the leg previous to $f_1(p)$, and let *terminating flight* $f_{n(p)}(p)$ be the next leg after $f_{n(p)-1}(p)$ assigned to aircraft p. Observe that at time t_0 each commencing flight is either in flight or has already landed. In Figure 2, Flights 14 and 18 are the commencing flights for Planes A and B, respectively. Flights 22 and 25 are the terminating flights. Because the disruption occurs after the arrival of commencing Flight 14, every feasible route created for Plane A must begin with Flight 14. Similarly, every route assigned to Plane B must start with commencing Flight 18. For each route $r \in R_{(p, F)}$, r begins with commencing flight $f_0(p)$, and r excludes all other commencing flights. We guarantee that the legs arriving after Tuesday at 16:00 are assigned a plane by requiring that terminating Flights 22 and 25 be assigned to different aircraft. For each route $r \in R_{(p, F)}$, r ends with exactly one terminating flight, and r excludes all other terminating flights. Observe that with Constraints (2) and (3), every solution to ARO will assign each terminating flight to a unique aircraft.

EXAMPLE 1 (CONTINUED). Table 1 displays the possible routes that can be constructed without delaying any legs. Routes 1–6 are possible new routes for Plane A, while Routes 7–22 can be assigned to Plane B. Observe that Routes 1–6 begin with commencing Flight 14, Routes 7–22 begin with commencing Flight 18, and no route includes both commencing flights. Each route ends with either terminating Flight 22 or 25, and no route includes both flights. Assuming that every route in Table 1 satisfies maintenance requirements, any pair of nonoverlapping routes is a feasible solution.

Airlines often prefer recovery solutions that return aircraft to their initial routes after a disruption. Observe that ARO satisfies this preference by imposing a penalty on each route $r \in R_{(p, F)}$ that does not end with terminating flight $f_{n(p)}$.

3.1. Aircraft Disruptions

When a disruption prevents an aircraft from flying at least one of its assigned flight legs, other planes may

412

Table 1   Routes That Can Be Constructed for the Example in Figure 2 Without Using Delays

Plane	Routes			

Plane A

	Route 1					Route 2					Route 3			
Flt	Departure Station	Arrival Station	Initial Plane		Flt	Departure Station	Arrival Station	Initial Plane		Flt	Departure Station	Arrival Station	Initial Plane
14	EWR	MSN	A		14	EWR	MSN	A		14	EWR	MSN	A
17	MSN	MDW	A		17	MSN	MDW	A		13	MSN	EWR	B
28	MDW	MSN	A		28	MDW	MSN	A		24	EWR	MSN	B
21	MSN	SAV	A		25	MSN	OAK	B		21	MSN	SAV	A
22	SAV	MSN	A							22	SAV	MSN	A

	Route 4					Route 5					Route 6			
Flt	Departure Station	Arrival Station	Initial Plane		Flt	Departure Station	Arrival Station	Initial Plane		Flt	Departure Station	Arrival Station	Initial Plane
14	EWR	MSN	A		14	EWR	MSN	A		14	EWR	MSN	A
13	MSN	EWR	B		21	MSN	SAV	A		25	MSN	OAK	B
24	EWR	MSN	B		22	SAV	MSN	A					
25	MSN	OAK	B										

Plane B

	Route 7					Route 8					Route 9			
Flt	Departure Station	Arrival Station	Initial Plane		Flt	Departure Station	Arrival Station	Initial Plane		Flt	Departure Station	Arrival Station	Initial Plane
18	MDW	MSN	B		18	MDW	MSN	B		18	MDW	MSN	B
15	MSN	OAK	A		15	MSN	OAK	A		11	MSN	SAV	B
16	OAK	MSN	A		16	OAK	MSN	A		12	SAV	MSN	B
17	MSN	MDW	A		17	MSN	MDW	A		17	MSN	MDW	A
28	MDW	MSN	A		28	MDW	MSN	A		28	MDW	MSN	A
21	MSN	SAV	A		25	MSN	OAK	B		21	MSN	SAV	A
22	SAV	MSN	A							22	SAV	MSN	A

	Route 10					Route 11					Route 12			
Flt	Departure Station	Arrival Station	Initial Plane		Flt	Departure Station	Arrival Station	Initial Plane		Flt	Departure Station	Arrival Station	Initial Plane
18	MDW	MSN	B		18	MDW	MSN	B		18	MDW	MSN	B
11	MSN	SAV	B		11	MSN	SAV	B		11	MSN	SAV	B
12	SAV	MSN	B		12	SAV	MSN	B		12	SAV	MSN	B
17	MSN	MDW	A		13	MSN	EWR	B		13	MSN	EWR	B
28	MDW	MSN	A		24	EWR	MSN	B		24	EWR	MSN	B
25	MSN	OAK	B		21	MSN	SAV	A		25	MSN	OAK	B
					22	SAV	MSN	A					

	Route 13					Route 14					Route 15			
Flt	Departure Station	Arrival Station	Initial Plane		Flt	Departure Station	Arrival Station	Initial Plane		Flt	Departure Station	Arrival Station	Initial Plane
18	MDW	MSN	B		18	MDW	MSN	B		18	MDW	MSN	B
11	MSN	SAV	B		11	MSN	SAV	B		13	MSN	EWR	B
12	SAV	MSN	B		12	SAV	MSN	B		24	EWR	MSN	B
21	MSN	SAV	A		25	MSN	OAK	B		21	MSN	SAV	A
22	SAV	MSN	A							22	SAV	MSN	A

Table 1 **(cont'd.)**

Plane												

Routes

	Route 16				Route 17				Route 18		
Flt	Departure Station	Arrival Station	Initial Plane	Flt	Departure Station	Arrival Station	Initial Plane	Flt	Departure Station	Arrival Station	Initial Plane
18	MDW	MSN	B	18	MDW	MSN	B	18	MDW	MSN	B
13	MSN	EWR	B	17	MSN	MDW	A	17	MSN	MDW	A
24	EWR	MSN	B	28	MDW	MSN	A	28	MDW	MSN	A
25	MSN	OAK	B	21	MSN	SAV	A	25	MSN	OAK	B
				22	SAV	MSN	A				

	Route 19				Route 20				Route 21		
Flt	Departure Station	Arrival Station	Initial Plane	Flt	Departure Station	Arrival Station	Initial Plane	Flt	Departure Station	Arrival Station	Initial Plane
18	MDW	MSN	B	18	MDW	MSN	B	18	MDW	MSN	B
15	MSN	OAK	A	15	MSN	OAK	A	21	MSN	SAV	A
16	OAK	MSN	A	16	OAK	MSN	A	22	SAV	MSN	A
21	MSN	SAV	A	25	MSN	OAK	B				
22	SAV	MSN	A								

	Route 22		
Flt	Departure Station	Arrival Station	Initial Plane
18	MDW	MSN	B
25	MSN	OAK	B

fly legs on the route of the disrupted aircraft. Suppose that, at time t_0, a controller realizes an aircraft p^* is unavailable to fly until time t_{p^*}. Consequently, the set of feasible routes $R_{(p^*, F)}$ excludes legs departing during time interval (t_0, t_{p^*}). Let $s(p^*)$ be the station where p^* will be located at t_{p^*}. Because we would like p^* to continue flying out of station $s(p^*)$, we select a time horizon (t_0, T), where $T > t_{p^*}$, such that the set of all legs F includes at least one *continuation flight*, which departs station $s(p^*)$ after time t_{p^*} and before time T. For example, Flights 13, 17, 21, and 25 are continuation flights, departing from Madison after 19:00 on Monday. If more than one aircraft is disrupted, ARO restricts the set of routes for each disrupted aircraft to exclude those routes that have legs during their disruptions, and we select a sufficiently large T to include continuation flights for each aircraft.

3.2. Station Disruptions
When weather and congestion reduce the *capacity* at an airport, the FAA provides a set of slots U for each

airline at the disrupted station, and an airline assign legs to the set of slots U. The set of disrupted aircraft includes those whose initial routes have a leg that must be assigned a slot; that is,

$$P^* = \{p \in P \mid \exists u \in U, R_u \ni r(p)\}.$$

The time horizon (t_0, T) includes every leg that needs slot assignment, and so

$$t_0 \leq \min_{u \in U}\{t_u^1\},$$

and

$$T \geq \max_{u \in U}\{t_u^2\},$$

where (t_u^1, t_u^2) is the time period of slot $u \in U$. The time horizon is determined similarly when an airline applies a capacity constraint to its flight schedule. In rare instances, such as repaving or icing on a runway, the FAA will temporarily close an airport. For this purpose, we include a *zero-capacity constraint*, $a \in A$ such that $\alpha_a = 0$.

4. Aircraft Selection Heuristic

Typically, the number of new routes in $\bigcup_{p \in P} R_{(p, F)}$ is extremely large when the time horizon (t_0, T) is significantly long. Table 1 displays 22 routes for an example that consists of only 2 planes and a 30-hour time horizon. Because most real aircraft recovery instances have between 30 and 150 planes and time horizons can be longer, the number of routes can be very large. However, decision makers must solve ARO in real time. In order to reduce the complexity of ARO, we select a subset of aircraft P' such that $P^* \subseteq P' \subseteq P$. We would like to select $P' \subset P$ such that the optimal value of solutions to ARO(P') is near the optimal value of a solution to ARO(P). In some instances $P^* = P$, so $P' = P$. However, in practice such instances are extremely rare and are often limited to multiple hubs shutting down for a prolonged period of time or closing the National Airspace System.

We define some theoretical properties of aircraft recovery, and then we present an aircraft selection heuristic (ASH) for selecting P'.

4.1. Theoretical Properties

Consider a route of an aircraft, $r = (f_0, \ldots, f_n)$. A route must maintain *flow balance*; that is, for any leg $f_i \in r$, the next leg f_{i+1} must depart from the same station where f_i arrives. A subsequence of legs, *subroute* $(f_i, \ldots f_j)$, is a *cancellation cycle* if f_i departs from the same station where f_j arrives. Observe that cancellation cycles include at least two flight legs because a single leg never has the same origin and destination.

The following proposition is obvious.

Proposition 1. *For any cancellation cycle $\gamma \in r$, $r - \gamma$ maintains flow balance.*

For example, by removing the cancellation cycle that includes Flights 17 and 28 from Route 1 in Table 1, we have Route 5, which maintains flow balance. Consequently, removing any set of disjoint cancellation cycles from a feasible route also maintains flow balance. Unfortunately, the removal of a cancellation cycle can prevent a route from being maintenance feasible if a station in the cancellation cycle is a maintenance base.

During aircraft recovery, operations controllers might delay legs. Delays allow for many additional

Table 2   **Flights 11, 12, and 25 Are Sufficiently Delayed to Construct a New Route for Plane B, Which Does Not Maintain Original Ordering**

Flight	Departure Station	Arrival Station
18	MDW	MSN
13	MSN	EWR
24	EWR	MSN
11	MSN	SAV
12	SAV	MSN
25	MSN	OAK

feasible routes, even though some may be impractical. For example, by sufficiently delaying Flights 11, 12, and 25 in Route 12 from Table 1, we could construct the route shown in Table 2 for Plane B. A route $r = (f_0, \ldots, f_n)$ maintains *original ordering* if for any pair of flights $f_i, f_{i+k} \in r$ from the same initial route $r(\cdot)$, f_i precedes f_{i+k} in $r(\cdot)$. The route depicted in Table 2 does not maintain original ordering because Flights 11 and 12 precede Flights 13 and 24 in the initial route of Plane B.

For each aircraft $p \in P$, the set of routes $R_{(p, r(p))}$ includes legs from the initial route $r(p)$. From Table 1, Routes 1 and 5 are in $R_{A, r(A)}$, and Routes 12, 14, 16, and 22 are in $R_{B, r(B)}$. If $r(p)$ is feasible, then $r(p) \in R_{(p, r(p))}$, and Proposition 1 implies that for any set of disjoint cancellation cycles Γ, $r(p) - \Gamma$ maintains flow balance.

Proposition 2. *For any $r \in R_{(p, r(p))}$ where r is originally ordered and does not include any ferries, diversions, or over-flies, there exists a set of disjoint cancellation cycles Γ such that $r = r(p) - \Gamma$.*

Proof. $r \subseteq r(p)$ because r does not include ferries, diversions, over-flies, or legs from other initial routes. It remains to be proven that $r(p) - r$ is a set of disjoint cancellation cycles. We assume $r \subset r(p)$ because if $r = r(p)$, the proof is trivial. Let $f_i(p)$ be the first leg in $r(p)$ but not in r. Route r must begin with commencing flight $f_0(p)$, $i \geq 1$, and so $f_{i-1}(p)$ must be in $r(p)$ and r. Similarly, Route r must end with a terminating flight, so r includes at least two legs. Let f_j be the next leg after $f_{i-1}(p)$ in r. $f_j \in r(p)$ because $r \subseteq r(p)$, and f_j flies after $f_{i-1}(p)$ in $r(p)$ because r is originally ordered. Let the sequence of legs from $f_{i-1}(p)$ to f_j in $r(p)$ be

$$(f_{i-1}(p), f_i(p), \ldots, f_{j-1}, f_j).$$

The arrival station of $f_{i-1}(p)$ is the same as the departure station of f_j because r maintains flow balance. Because $r(p)$ maintains flow balance, the departure station of $f_i(p)$ is the same as the arrival station of $f_{i-1}(p)$, and the arrival station of f_{j-1} is the same as the departure station of f_j. Consequently, the departure station of $f_i(p)$ is the same as the arrival station of f_{j-1}. Thus, $(f_i(p), \ldots, f_{j-1})$ is a cancellation cycle. Without loss of generality, this proof applies to the first leg after f_j that is in $r(p)$ but not in r. We can construct r by iteratively removing disjoint cancellation cycles from $r(p) - r$. $\square$

By ignoring ferries, diversions, and over-flies, and relaxing maintenance constraints, Propositions 1 and 2 imply that if $r(p)$ is feasible, then $|R_{(p, r(p))}|$, the number of originally ordered routes for aircraft p that can be constructed with legs from $r(p)$, equals the number of sets of disjoint cancellations cycles in $r(p)$ plus one to account for the original route. The three sets of nonempty disjoint cancellation cycles in $r(B)$ are shown in Table 3, and so, with the empty set, $R_{B, r(B)}$ consists of Routes 12, 14, 16, and 22 in Table 1.

For any two aircraft $p_i, p_j \in P$, let $F(p_i, p_j)$ be the set of legs in $r(p_i) \cup r(p_j)$, and let

$$w_{p_i p_j} = |R_{(p_i, F(p_i, p_j))}| - |R_{(p_i, r(p_i))}|$$

be the *interaction* of aircraft p_i with aircraft p_j; that is, $w_{p_i p_j}$ is the number of routes that can be constructed for p_i with legs from $F(p_i, p_j)$ and include at least one leg from $r(p_j)$. Observe that from Table 1,

Table 3   **Flights 13 and 24, Flights 11 and 12, and Flights 11, 12, 13, and 24 Are Nonempty Sets of Disjoint Cancellation Cycles in the Initial Route of Plane B**

Flight	Departure Station	Station Arrival
13	MSN	EWR
24	EWR	MSN
11	MSN	SAV
12	SAV	MSN
11	MSN	SAV
12	SAV	MSN
13	MSN	EWR
24	EWR	MSN

$w_{AB} = 6 - 2 = 4$ and $w_{BA} = 16 - 4 = 12$. Let the *interaction graph* be a directed graph $G = (P, E = \{(p_i, p_j) \mid w_{p_i p_j} > 0\})$.

We refer to a route of the form

$$\hat{r} = (f_0(p_i), \ldots, f_l(p_i), f_k(p_j), \ldots, f_{n(p_j)}(p_j))$$

as a *single swap route* from p_j to p_i. Routes 2, 4, and 6 in Table 1 are single swaps from Plane B to Plane A, and Routes 7, 9, 13, 15, 17, 19, and 21 are single swaps from Plane A to Plane B.

PROPOSITION 3. *If there exists an arc $(p_i, p_j) \in E$, then there exists a single swap Route $\hat{r}$ from p_j to p_i that maintains flow balance.*

PROOF. Because $(p_i, p_j) \in E$, there exists a route $r \subset F(p_i, p_j)$ for plane p_i that includes at least one leg from $r(p_j)$. Let $f_k(p_j)$ be the first leg in r from $r(p_j)$, let $f_l(p_i)$ be the leg in Route r previous to leg $f_k(p_j)$, let $(f_{k+1}(p_j), \ldots, f_{n(p_j)}(p_j))$ be the set of legs after $f_k(p_j)$ in $r(p_j)$, and let

$$\hat{r} = (f_0(p_i), \ldots, f_l(p_i), f_k(p_j), f_{k+1}(p_j), \ldots, f_{n(p_j)}(p_j)).$$

Because, $r(p_i)$, $r(p_j)$, and r maintain flow balance, $\hat{r}$ maintains flow balance. $\square$

Observe that if there exists an arc $(p_i, p_j) \in E$, $\hat{r}$ is maintenance feasible and $(f_k(p_j), \ldots, f_{n(p_j)}(p_j))$ excludes zero-capacity constraints, then $\hat{r} \in R_{(p_i, F(p_i, p_j))}$.

4.2.   Heuristic Algorithm

Our ASH uses the graph G to create a set of aircraft P' from a set of disrupted aircraft P^*. For each disrupted aircraft $p^* \in P^*$, ASH finds directed cycles in G that include p^*. Let $C = \{p_1, \ldots, p_n\}$ be a directed cycle in G where aircraft $p_1 \in P^*$, and let aircraft $p_{n+1} = p_1$. By Proposition 3, for each $p_i \in C$, there exists a single swap route $\hat{r}_{p_i}$ from p_{i+1} to p_i that maintains flow balance, and so $\widehat{R} = \{\hat{r}_{p_1}, \ldots, \hat{r}_{p_n}\}$ is a set of single swap routes. Moreover, for each $\hat{r}_{p_i} \in \widehat{R}$, if $\hat{r}_{p_i}$ is maintenance feasible and excludes zero-capacity constraints, then route $\hat{r}_{p_i}$ is a feasible route for aircraft p_i. Observe that $\widehat{R}$ might not provide a set of feasible routes if a leg is covered by two routes. ASH attempts to overcome this difficulty by finding many directed cycles for each disrupted aircraft.

ALGORITHM 1: AIRCRAFT SELECTION HEURISTIC (ASH).

$P' \leftarrow P^*$
for all $p^* \in P^*$ **do**
  $P'(p^*) \leftarrow \{p^*\}$
  $l(P - p^*) \leftarrow \infty$
  $l(p^*) \leftarrow 0$
  $\rho(P) \leftarrow NULL$
  $\widehat{P} \leftarrow [p^*]$
  $i = 1$
  while $(|P'(p^*)| < MINPLANES) \wedge (i \leq |\widehat{P}|)$ **do**
    for all $p' \in \{P - \{\widehat{P}_i\} \mid w_{\widehat{P}_i p'} > 0\}$ **do**
      if $l(p') > l(\widehat{P}_i) + 1$ **then**
        $l(p') \leftarrow l(\widehat{P}_i) + 1$
        $\rho(p') \leftarrow \widehat{P}_i$
        if $p' \notin \widehat{P}$ **then**
          $\widehat{P} \leftarrow [\widehat{P}, p']$
        end if
      else if $p' = p^*$ **then**
        $\tilde{p} \leftarrow \widehat{P}_i$
        while $\tilde{p} \neq p^*$ **do**
          $P'(p^*) \leftarrow P'(p) \cup \{\tilde{p}\}$
          $P' \leftarrow P' \cup \{\tilde{p}\}$
          $\tilde{p} \leftarrow \rho(\tilde{p})$
        end while
      end if
    end for
    $i \leftarrow i + 1$
  end while
end for

For each disrupted aircraft $p^* \in P^*$, ASH, shown by Algorithm 1, searches for directed cycles in G with a minimum number of aircraft. The *distance* of a path in G is the number of aircraft on the path. Given a disrupted aircraft p^*, ASH maintains a set of aircraft $P'(p^*)$ that are in directed cycles that include aircraft p^*, and it uses an ordered list of aircraft $\widehat{P}$ whose shortest path distances to p^* have already been determined. Let $\widehat{P}_i$ be the ith aircraft in the ordered list of aircraft $\widehat{P}$, and we use the notation $[\cdot]$ to represent an ordered set. For each $p \in P$, let $l(p)$ be the distance of a shortest path from p^* to p, and let $\rho(p)$ be an aircraft that precedes aircraft p on a shortest path. ASH sequentially updates l and ρ for each aircraft adjacent to the aircraft in $\widehat{P}$, and if an adjacent

aircraft p' is not already in $\widehat{P}$, ASH appends it to $\widehat{P}$; that is, $\widehat{P} \leftarrow [\widehat{P}, p']$. If ASH discovers that p^* is adjacent to an aircraft in $\widehat{P}$, it identifies a directed cycle, and it adds each aircraft on the cycle to $P'(p^*)$ and P'. Let parameter *MINPLANES* be the minimum number of aircraft in cycles that include aircraft p^*, and so when ASH collects at least *MINPLANES* aircraft in $P'(p^*)$, it constructs a set of shortest paths and cycles for another disrupted aircraft. Finally, ASH returns P' to ARO when the heuristic has found directed cycles for each disrupted aircraft.

In §5, we show that by using ASH with *MINPLANES* = 5, ARO solves 3,264 instances of a fleet of 96 aircraft and 469 legs in 14 hours and 21 minutes, which is less than 16 seconds per instance. However, without ASH, ARO cannot solve instances for the same fleet with a time horizon longer than two days. Constructing the graph G is the computational bottleneck of ASH. Determining $w_{p_i p_j}$ requires $O(c^{|F(p_i, p_j)|})$ computations, where c is a constant. Although the complexity grows exponentially in $|F(p_i, p_j)|$, it is rarely very large in practice. An aircraft typically flies three or four legs per day. Station disruptions usually last a few hours, and unscheduled maintenance problems rarely require more than one day to repair. In an unscheduled maintenance distribution provided by a major domestic carrier, over 95% of the service times were less than twelve hours, and 99.7% were less than two days. Consequently, a decision maker would probably choose the time horizon (t_0, T) to be less than two or three days, and so dynamically constructing a subgraph of G is not computationally intensive.

5. Computational Results

We validated ARO using a simulation of airline operations, SimAir, described in Rosenberger et al. (2002) on the daily flight schedule of fleets from a major domestic airline carrier shown in Table 4. We compared ARO with a shortest cycle cancellation policy (SCC) from Rosenberger et al. (2002). The simulation time was 500 days of flight operations. On average, hubs were closed for a few hours once per week, and after 8% of arrivals, an unscheduled maintenance

Table 4   Three Fleets from a Major Domestic Carrier

Fleet	Number of Planes	Number of Flights
1	96	469
2	70	302
3	32	139

delay of up to two days occurred. The delay threshold that determined whether to use ARO was 30 minutes. ARO allowed delays of up to three hours in the construction of the new routes, and the objective was to minimize cancellations. The unscheduled maintenance distribution is from a major domestic carrier. We found our results using a 500 MHz Intel Pentium III, and CPLEX 6.0 Callable Library solved ARO's integer program. In Table 5, CPU time is in seconds; OT+15 is the on-time percentage; OT+60 is the percentage of legs arriving within 60 minutes of their originally scheduled arrival time; lateness is the average number of minutes a leg arrives after its originally scheduled arrival time; can % is the percentage of legs that are cancelled; que % is the percentage of legs that were delayed on the runway of the departure station or in the airspace of the arrival station for longer than 25 minutes; miss pass % is the percentage of passengers that have a disrupted itinerary and must be rerouted; incon pass % is the percentage of passengers that either have a disrupted itinerary or arrived more than 30 minutes after the originally scheduled arrival time of their itinerary; ARO calls is the number of times SimAir uses ARO during the 500 days; and swaps is the number of legs that are flown by an aircraft different from their originally assigned aircraft. For each fleet, ARO signif-

icantly reduced cancellations and passenger disruptions, while improving on-time performance. Because ARO preferred delaying legs for up to one hour over cancelling legs, the average lateness and the number of legs delayed more than 25 minutes either at the runway or in the airspace was marginally greater using ARO than SCC for Fleet 3. In addition to solving recovery instances for Fleet 1 at an average rate less than 16 seconds per instance, ARO solved 2,206 and 1,252 instances at a rate less than 11 and 6 seconds per instance for Fleets 2 and 3, respectively.

In several additional experiments, we calculated the variance of the number of cancellations per day and found 90% and 95% confidence intervals around the percentage of cancelled flights. The magnitudes of the 90% and 95% confidence intervals were less than 0.65% and 0.8%, respectively.

During the construction of new routes, ARO will not delay a leg beyond a *maximum delay tolerance*. In Table 6, we give simulated results for Fleet 3 with different maximum delay tolerances, Max Delay, to study the tradeoff between delays and cancellations. ARO minimized total delay and cancellations by equating a cancellation to a three-hour delay. Observe that on-time performance is insensitive to the delay tolerance, whereas average lateness and cancellation percentage are affected.

6.   An Alternate Objective Function

In §5, we used on-time performance, lateness, cancellations, legs delayed at the runway or in the air, and passenger disruptions and inconvenience as measurements to evaluate ARO and SCC in airline operations.

Table 5   The Results of Simulating 500 Days Airline Operations Comparing Recovery Policies ARO and SCC

Fleet	Recovery	CPU Time	OT+15	OT+60	Lateness	Can %	Que %	Miss Pass %	Incon Pass %	ARO Calls	Swaps
1	SCC	543	68.44	85.64	16.753	8.41	1.11	21.89	33.44	0	0
1	ARO	51,643	69.25	87.43	18.558	4.77	1.19	17.24	31.55	3,264	120,244
2	SCC	405	67.65	85.93	16.534	8.28	1.18	19.14	30.73	0	0
2	ARO	23,627	68.88	88.06	17.117	4.93	1.28	14.52	27.88	2,206	80,130
3	SCC	103	67.30	85.51	15.817	8.96	0.83	15.91	27.72	0	0
3	ARO	6,463	69.14	88.74	17.412	4.47	1.00	10.32	24.47	1,252	38,254

Table 6   The Results of Simulating 500 Days Airline Operations for Fleet 3 Varying the Maximum Delay of a Flight

Max Delay	OT+15	OT+60	Lateness	Can %	Que %	Miss Pass %	Incon Pass %	ARO Calls	Swaps
30	68.45	85.92	11.312	10.80	0.90	19.43	27.21	2,324	43,231
60	69.34	88.36	12.030	7.85	0.93	14.92	24.80	2,620	43,821
120	69.87	89.32	13.849	5.17	1.02	11.24	23.53	2,828	39,448
180	70.22	89.67	14.493	4.50	0.98	10.18	22.88	2,762	42,637
240	70.25	89.67	14.864	4.24	1.01	9.75	22.72	2,775	43,649
300	70.19	89.61	15.140	4.34	0.98	10.00	22.86	2,727	42,001
600	70.65	90.03	19.241	3.18	0.99	8.08	22.07	2,685	30,580
1,200	70.47	89.73	23.923	2.56	0.97	7.26	22.41	2,735	6,211

However, crew and passenger misconnections require additional crew and passenger recovery solutions that are often expensive to implement (Lettovský et al. 2000). Consequently, operations controllers will try to find aircraft recovery solutions that avoid rerouting crews and passengers. Here we provide a revised aircraft recovery model that attempts to maintain crew and passenger connections.

One difficulty in solving passenger and crew recovery is that they can fly on other airline carriers, so realistic crew and passenger recovery models include every leg in all airline systems. Our revised approach is to assign a cost of disruption to a *trip*, a crew pairing or a passenger itinerary, and then minimize the cost of the disrupted trips. For example, the cost of a disrupted passenger itinerary might be the lost revenue from rerouting the passengers, and the cost of a disrupted crew pairing might be the expected cost of rerouting the regular crew and reassigning another regular or reserve crew to fly the pairing.

In the revised aircraft recovery model, we add constraints to the aircraft recovery integer program described by (2)–(7), and we change the objective function (1). In order to minimize disrupted trips, we include a constraint to calculate the delay of each uncanceled leg. Let d_{rf} be the delay of leg f in Route r. We include constraints

$$\sum_{r \ni f} d_{rf} X_r = D_f \qquad \forall f \in F, \qquad (8)$$

where D_f is the delay of leg f. Let Q be a set of flight connections. For each $q \in Q$, let f_q^1 be the first leg, let

f_q^2 be the second leg, let σ_q be the scheduled *slack*, time between the arrival of leg f_q^1 and the departure of leg f_q^2, and let

$$O_q = \begin{cases} 1 & \text{if connection } q \text{ is disrupted,} \\ 0 & \text{otherwise.} \end{cases}$$

Let τ be the minimum amount of time for a passenger or crew to make a connection, and let δ_f be the maximum delay of a leg f. We add the following constraints:

$$D_{f_q^2} - D_{f_q^1} + \left(\delta_{f_q^1} + \tau - \sigma_q\right) O_q \geq \tau - \sigma_q \qquad \forall q \in Q. \quad (9)$$

Let V be the set of trips. For each trip $v \in V$, let g_v be the cost of disrupting v, let $Q(v)$ be the set of connections, let $F(v)$ be the set of legs in v, and let

$$Z_v = \begin{cases} 1 & \text{if trip } v \text{ is disrupted,} \\ 0 & \text{otherwise.} \end{cases}$$

We add the following constraints:

$$Z_v \geq O_q \qquad \forall q \in Q(v), \forall v \in V, \qquad (10)$$

$$Z_v \geq K_f \qquad \forall f \in F(v), \forall v \in V. \qquad (11)$$

Objective function (1) is replaced by one that minimizes the cost of disrupting the trips, which is given by

$$\min \sum_{v \in V} g_v Z_v. \qquad (12)$$

The advantage of our revised model is that it provides a framework to minimize the effect of disruptions on crews and passengers. Unlike the model in

Table 7   The Results of Simulating 500 Days Airline Operations

Fleet	Obj	OT+15	OT+60	Lateness	Can %	Que %	Miss Pass %	Incon Pass %	ARO Calls	Swaps
1	Can	69.07	84.62	12.287	11.89	1.05	26.73	33.82	7,538	15,4138
1	Pax	68.14	84.15	12.788	12.23	1.10	25.91	33.53	7,719	15,1792
2	Can	68.34	85.79	12.547	10.44	1.14	22.24	30.21	4,563	99,354
2	Pax	67.38	85.36	13.126	10.72	1.19	20.86	29.71	4,792	10,2130
3	Can	68.10	85.29	12.535	11.16	0.86	19.74	27.76	2,338	40,895
3	Pax	67.50	85.22	12.528	11.22	0.90	18.55	27.25	2,370	44,762

Clarke (1997a, b), our model considers multiflight passenger itineraries and regular crew pairings. A controller must use it in addition to passenger and crew recovery models because our revised model does not reroute passengers and crews. For example, our revised model could solve an aircraft recovery subproblem in an integrated model such as the one in Lettovský (1997).

Table 7 displays the comparison between minimizing cancellations and disrupted passenger itineraries using the revised model; obj indicates whether ARO minimized cancellations (can) or missed passenger connections (pax). Observe that the initial model had slightly better on-time performance, less average lateness, and fewer cancellations, whereas the revised model had less passenger inconvenience and significantly fewer passenger disruptions.

7. Conclusions and Future Research

Automated recovery policies allow operations controllers to make better recovery decisions. As disruptions have become more frequent and severe, airlines have begun to explore automated recovery, and operations models are becoming more prevalent in the research literature. Network flow models yield a solution in polynomial time, but they do not include maintenance constraints and refueling. Set-packing models optimize over a set of new routes, and so column generation can check maintenance feasibility. Although set-packing problems are *NP*-hard, we overcome this difficultly by providing an aircraft selection heuristic (ASH) that efficiently determines a subset of aircraft to reroute. However, airlines are considering integrated recovery models (Lettovský 1997). In an integrated recovery model, the master problem determines which flight legs to delay and cancel, and the subproblems reroute aircraft, crews, and passengers. To implement ASH within an integrated recovery model, the master problem, instead of the aircraft subproblem, would use ASH. We could evaluate an integrated recovery model using a stochastic model of airline operations, as we examined our aircraft recovery optimization model (ARO) in this paper.

The airline environment is constantly changing, and an aircraft recovery, such as ARO, assists operations controllers in overcoming adverse conditions. After a controller recovers from a disruption, conditions, such as weather, continue to change. A robust recovery policy that considers future conditions might provide a better solution than one that assumes the new flight schedule will operate as planned. For example, we could model aircraft recovery as a two-stage stochastic programming problem with a distribution for the change in weather. We could solve the problem using a Benders' decomposition formulation in which the master problem determines the new aircraft routes for the first stage, and the subproblems reroute the aircraft for different second stage scenarios.

References

Andreatta, G., L. Brunetta, G. Guastalla. 2000. From ground holding to free flight: An exact approach. *Transportation Sci.* **34** 394–401.

Berge, M. E., C. A. Hopperstad. 1993. Demand driven dispatch: A method for dynamic aircraft capacity assignment, models and algorithms. *Oper. Res.* **41** 153–168.

Cao, J.-M., A. Kanafani. 2000. The value of runway time slots for airlines. *Eur. J. Oper. Res.* **126** 491–500.

Carlson, P. M. 2000. Exploiting the opportunities of collaborative decision making: A model and efficient solution algorithm for airline use. *Transportation Sci.* **34** 381–393.

Chang, K., K. Howard, R. Oiesen, L. Shisler, M. Tanino, M. C. Wambganss. 2001. Enhancements to the FAA ground-delay program under collaborative decision making. *Interfaces* **31** 57–76.

Clarke, M. D. D. 1997a. The airline schedule recovery problem. Technical report, Massachusetts Institute of Technology, Operations Research Center, Cambridge, MA.

——. 1997b. Development of heuristic procedures for the flight rescheduling in the aftermath of irregular airline operations. Technical report, Massachusetts Institute of Technology, Operations Research Center, Cambridge, MA

——. 1997c. Irregular airline operations: A review of the state-of-the-practice in airline operations control centers. Technical report, Massachusetts Institute of Technology, Operations Research Center, Cambridge, MA.

Dobbyn, T. 2000. U.S. prepares plan to cut summer flight delays. *Reuters Wire* (March 3).

Federal Aviation Administration. 1999. Federal aviation regulations. http://www.faa.gov/avr/AFS/FARS/far-121.txt.

Hoffman, R. L. 1997. Integer programming models for ground-holding in air traffic flow management. Ph.D. thesis, The National Center of Excellence for Aviation Operations Research, University of Maryland, College Park, MD.

Jarrah, A. I. Z., G. Yu, N. Krishnamurthy, A. Rakshit. 1993. A decision support framework for airline flight cancellations and delays. *Transportation Sci.* **27** 266–280.

Lettovský, L. 1997. Airline operations recovery: An optimization approach. Ph.D. thesis, Georgia Institute of Technology, Atlanta, GA.

——, E. L. Johnson, G. L. Nemhauser. 2000. Airline crew recovery. *Transportation Sci.* **34** 337–348.

Luo, S., G. Yu. 1994a. Airline schedule perturbation problem: Ground delay program with nonsplitable resources. Technical Report 93/94-3-13, Department of Management Science and Information Systems, University of Texas at Austin, Austin, TX.

——, ——. 1994b. Airline schedule perturbation problem: Ground delay program with splitable resources. Technical Report 93/94-3-14, Department of Management Science and Information Systems, University of Texas at Austin, Austin, TX.

——, ——. 1997. On the airline schedule perturbation problem caused by the ground delay program. *Transportation Sci.* **31** 293–311.

Mathaisel, D. F. X. 1996. Decision support airline system operations control and irregular operations. *Comput. Oper. Res.* **23** 1083–1098.

Metron, Inc. 2001. Collaborative decision making. http://www.metsci.com/cdm.

Rakshit, A., N. Krishnamurthy, G. Yu. 1996. System operations advisor: A real-time decision support system for managing airline operations at United Airlines. *Interfaces* **26** 50–58.

Richetta, O., A. M. Odoni. 1993. Solving optimally the static ground holding policy problem in air traffic control. *Transportation Sci.* **27** 228–238.

Rosenberger, J. M. 2001. Topics in airline operations. Ph.D. thesis, The Logistics Institute, Georgia Institute of Technology, Atlanta, GA.

——, A. J. Schaefer, D. Goldsman, E. L. Johnson, A. J. Kleywegt, G. L. Nemhauser. 2002. A stochastic model of airline operations. *Transportation Sci.* **36**(4) 357–377.

Talluri, K. T. 1996. Swapping applications in a daily airline fleet assignment. *Transportation Sci.* **30** 237–284.

Teodorovic, D., S. Guberinio. 1984. Optimal dispatching strategy on an airline network after a schedule perturbation. *Eur. J. Oper. Res.* **15** 178–182.

——, G. Stojkovio. 1990. Model for operational daily airline scheduling. *Transportation Planning Tech.* **13** 273–285.

——, ——. 1995. Model to reduce airline schedule distrubances. *J. Transportation Engrg.* **121** 324–331.

Thengvall, B. G., J. F. Bard, G. Yu. 2000. Balancing user preferences for aircraft schedule recovery. *IIE Trans.* **32** 181–193.

Vasquez-Marquez, A. 1991. American Airlines arrival slot allocation system (ASAS). *Interface* **21** 42–61.

Vranas, P. B., D. J. Bertsimas, A. R. Odoni. 1994. The multi-airport ground-holding problem in air traffic control. *Oper. Res.* **42** 249–261.

Yan, S., C.-G. Lin. 1997. Airline scheduling for the temporary closure of airports. *Transportation Sci.* **31** 72–82.

——, Y. P. Tu. 1997. Multifleet routing and multistop flight scheduling for schedule perturbation. *Eur. J. Oper. Res.* **103** 155–169.

——, D. Yang. 1996. A decision support framework for handling schedule perturbation. *Transportation Res.* **30** 405–419.