Benders Decomposition for Simultaneous Aircraft Routing and Crew Scheduling

Jean-François Cordeau • Goran Stojković • François Soumis • Jacques Desrosiers
École des Hautes Études Commerciales and GERAD, 3000, chemin de la Côte-Sainte-Catherine, Montréal, Canada H3T 2A7
École Polytechnique and GERAD, C.P. 6079, Succ. Centre-Ville, Montréal, Canada H3C 3A7
École Polytechnique and GERAD, C.P. 6079, Succ. Centre-Ville, Montréal, Canada H3C 3A7
École des Hautes Études Commerciales and GERAD, 3000, chemin de la Côte-Sainte-Catherine, Montréal, Canada H3T 2A7

Given a set of flight legs to be flown by a single type of aircraft, the simultaneous aircraft routing and crew scheduling problem consists of determining a minimum-cost set of aircraft routes and crew pairings such that each flight leg is covered by one aircraft and one crew, and side constraints are satisfied. While some side constraints such as maximum flight time and maintenance requirements involve only crews or aircraft, linking constraints impose minimum connection times for crews that depend on aircraft connections. To handle these linking constraints, a solution approach based on Benders decomposition is proposed. The solution process iterates between a master problem that solves the aircraft routing problem, and a subproblem that solves the crew pairing problem. Because of their particular structure, both of these problems are solved by column generation. A heuristic branch-and-bound method is used to compute integer solutions. On a set of test instances based on data provided by an airline, the integrated approach produced significant cost savings in comparison with the sequential planning process commonly used in practice. The largest instance solved contains more than 500 flight legs over a 3-day period.

Introduction

After creating a schedule that defines origin and destination cities as well as departure and arrival times for each flight leg to be flown during a given period, most airlines use a sequential procedure to plan their operations. The first step of this procedure is the *fleet assignment problem*, which consists of assigning an aircraft type to each flight leg so as to maximize anticipated profits. For each aircraft type, an *aircraft routing problem* is then solved to determine the sequence of flight legs to be flown by each individual aircraft so as to cover each leg exactly once while ensuring appropriate aircraft maintenance. To comply with safety regulations stipulated by transport authorities, airlines must ensure that every aircraft regularly undergoes different types of maintenance checks that vary in scope, frequency, and duration. For example, routine checks are usually performed at night every 3 to 4 days, whereas more complete checks performed annually may require the aircraft to be grounded for several days at once. When solving the fleet assignment problem and the aircraft routing problem, airlines sometimes take into consideration *through values* that represent the extra revenues obtained by assigning the same aircraft to a pair of consecutive flight legs (i.e., a *through*) so that passengers flying from the origin of the first leg to the destination of the second leg do not have to change aircraft. When through values are not considered, the aircraft routing problem usually reduces to a feasibility problem.

0041-1655/01/3504/0375$05.00
1526-5447 electronic ISSN

Given fixed aircraft routes and a set of work rules defined by the collective agreement, the airline then builds crew rotations or *pairings* by solving a *crew scheduling problem*. In general terms, a crew pairing is a sequence of duty and rest periods that typically lasts between 2 and 5 days. A *duty period* corresponds to a single workday for a crew and can be seen as a sequence of flight legs separated by short rest periods, called *sits*. A pairing is thus a sequence of duty periods separated by overnight rests. Every pairing begins and ends at a specific location, called the *crewbase*, and must satisfy a set of applicable work rules related to a large number of factors, such as flight time, rest time, connection time, etc. The objective of the crew scheduling problem is to determine a minimum-cost set of pairings so that every flight leg is assigned a qualified crew. The cost of a pairing depends not only on the total flight time but also on the waiting time during connections as well as on related accommodation expenses such as transport, meals, and hotel rooms when overnight connections take place outside the crewbase. In the last step of the planning process, pairings are finally combined to form personalized monthly schedules that are assigned to employees by solving a *crew bidding problem* or a *crew rostering problem*.

Using a sequential procedure considerably reduces the complexity of the process but may also yield solutions that are far from optimal. Furthermore, finding feasible solutions may become difficult when flexibility is reduced by previously made decisions. This paper introduces a model and a solution methodology for the simultaneous routing of aircraft and scheduling of crews. It thus represents an attempt at integrating two interacting parts of the planning process. The main source of interaction between aircraft routing and crew scheduling resides in variable connection times: because the minimum connection time (or *sit time*) required between two successive flight legs covered by the same crew depends on whether the same aircraft is used on both legs, the set of feasible pairings depends on the aircraft routing decisions made in the previous step. Hence, a suboptimal solution is likely to be obtained if a sequential planning procedure is used. The contribution of this paper is to present a model and a solution approach for handling both types of decisions simultaneously so as to obtain an improved global solution. Because crew pairings are normally updated monthly, the model would typically be solved before preparing personalized crew schedules for the next month. However, as explained by Stojković (1999), the approach can also be used at the operational level to update aircraft routes and crew pairings following schedule perturbations.

The operations research literature contains few references regarding the integration of aircraft routing and crew scheduling decisions. An effort in this direction is the work of Barnhart et al. (1998b), who proposed an integrated but approximate model for combined fleet assignment and crew scheduling. The approach does not truly solve the two problems simultaneously but rather incorporates a relaxation of the crew scheduling problem in the fleet assignment model. This relaxation, which is based on a duty network (Barnhart and Shenoi 1998), ensures that all flight legs are covered by eligible crews but does not impose constraints on the maximum number of duties within a pairing or the maximum time away from the crewbase. Very recently, Klabjan (1999) described an approach that solves the crew scheduling and aircraft routing problems sequentially but adds plane count constraints to the crew scheduling model so as to ensure the feasibility of the resulting aircraft routing problem. In addition, this model allows the departure time of each flight leg to be moved within a given time window so as to further reduce crew costs. On test instances involving up to 450 flight legs, the method produced very significant savings. Issues related to the introduction of maintenance and crew considerations in the fleet assignment problem were also discussed by Clarke et al. (1996). Finally, other interesting contributions with respect to the integration of the planning process are the two approaches presented by Barnhart et al. (1998a) and Desaulniers et al. (1997b) for the combined fleet assignment and aircraft routing problem.

While few integrated planning models exist, several modeling and solution approaches have been proposed to address the individual components of the sequential process described above (Yu 1998). Models for the fleet assignment problem were proposed, among others, by Abara (1989) and Hane et al. (1995), while the aircraft routing problem was addressed by Feo and Bard (1989), Clarke et al. (1997), Talluri (1998), and Gopalan and Talluri (1998). Numerous contributions regarding the different variants of the crew scheduling problem can also be found in the operations research literature. In particular, column generation was first applied to the crew pairing problem by Lavoie et al. (1988), while further developments and alternative solution methods can be found in the recent works of Hoffman and Padberg (1993), Graves et al. (1993), Barnhart et al. (1995), Vance et al. (1997), and Desaulniers et al. (1997a).

In this paper, we assume that the fleet assignment problem has been solved, so that the type of aircraft assigned to each flight leg is known. For ease of exposition, we also assume that crews are qualified to fly a single type of aircraft, although this assumption is easily relaxed. In this context, the simultaneous aircraft routing and crew scheduling problem decomposes into one problem for each aircraft type. Given a set of flight legs to be flown by the aircraft of a specific type, the problem is to determine a minimum-cost set of aircraft routes and crew pairings such that each flight leg is covered by one aircraft and one crew, and side constraints are satisfied. Our formulation assumes a dated planning horizon in which the set of flight legs may vary from day to day, although it can easily be adapted to handle the daily problem that is common in the crew scheduling literature. The maintenance requirements imposed correspond to the routine checks performed every 3 or 4 days by most airlines. Other types of maintenance checks need not be considered explicitly within the model since they are normally scheduled separately according to the availability of maintenance facilities. Finally, approximate crew costs are used, and no routing costs are considered.

The remainder of the article is organized as follows. The next section introduces some notation and a mathematical formulation of the problem, while §2 presents a solution methodology that combines column generation and Benders decomposition. Computational experiments that show the benefits of integration are reported in §3. Conclusions and directions for future work are discussed in the final section.

1. Mathematical Formulation

Consider a set L of flight legs to be flown by a single aircraft type during the planning horizon. Each flight leg $l \in L$ is defined by origin and destination stations and by fixed departure and arrival dates and times. Let $G = (N, A)$ be a time-space network, where N is the node set and A is the arc set. Each node $i \in N$ corresponds to a flight leg $l_i \in L$ that must be covered exactly once by one aircraft and one crew. Each arc $(i, j) \in A$ represents a feasible connection between two successive flight legs: an arc is defined between nodes i and j if the destination station of leg l_i is the departure station of leg l_j and if the connection time between the two legs is larger than a given station-specific threshold that represents the minimum connection time when both legs are covered by the same aircraft. This minimum connection time is often called the *minimum plan turn time*.

Let F and K denote the sets of available aircraft and crews, respectively. For each aircraft $f \in F$, let o^f and d^f be nodes that represent, respectively, the origin of this aircraft at the beginning of the planning horizon and its destination at the end. Define $O^f \subseteq \{(o^f, j) | j \in N\}$ as the set of arcs linking the origin node of aircraft f to nodes representing legs that can be covered first by this aircraft at the beginning of the horizon. The set $D^f \subseteq \{(i, d^f) | i \in N\}$ is defined similarly for legs that can be covered last at the end of the planning horizon. One then defines a network $G^f = (N^f, A^f)$ where $N^f = N \cup \{o^f, d^f\}$ and $A^f = A \cup O^f \cup D^f$. Sets O^f and D^f can be used to impose initial and final conditions on each aircraft. If no particular condition is to be imposed on aircraft f, then arcs (o^f, i) and (i, d^f) can be defined for every node $i \in N$. For every crew $k \in K$, nodes o^k and d^k, sets O^k and D^k, and a network $G^k = (N^k, A^k)$ are defined in a similar fashion.

Given two flight legs $l_i, l_j \in L$ such that $(i, j) \in A$, the connection between these legs is said to be *short* if the difference between the departure time of leg l_j and the arrival time of leg l_i is smaller than a given threshold that corresponds to the minimum sit time for crews. In this case, legs l_i and l_j can be covered by the same crew only if both legs are covered by the same aircraft. Otherwise, insufficient time is available for the crew to make the connection. Let $C \subseteq A$ be the set of arcs representing short connections in the network G.

For every aircraft $f \in F$, let Ω^f be the set of feasible paths between nodes o^f and d^f in G^f. More specifically, a feasible path is a sequence of flight legs that provides a maintenance opportunity at least once every ℓ days. A maintenance opportunity is provided when a connection taking place in a station where maintenance can be performed is longer than the minimum maintenance time (e.g., 6 or 8 hours). Similarly, for every crew $k \in K$, let Ω^k denote the set of feasible paths in the network G^k. Because a dated planning horizon is considered, a path may represent either a complete pairing or the initial or final portion of a pairing. Each duty period in a pairing must respect daily limits on total work time, total flight time, and total number of landings. In addition, the number of duty periods in a pairing must not exceed a certain limit, which is typically 4 or 5.

For every aircraft path $\omega \in \Omega^f$, define binary constants a_ω^i and b_ω^{ij} that take value 1 if node $i \in N^f$ and arc $(i, j) \in A^f$ belong to this path, respectively. Let also c_ω be the cost of sending one unit of flow between o^f and d^f along path ω, and let θ_ω be a binary variable that represents the flow on this path. Finally, corresponding parameters a_ω^i, b_ω^{ij}, and c_ω, as well as binary variables ζ_ω, are defined for every crew path $\omega \in \Omega^k$.

The model for simultaneous aircraft routing and crew scheduling can be stated as follows:

$$\text{Minimize} \sum_{f \in F} \sum_{\omega \in \Omega^f} c_\omega \theta_\omega + \sum_{k \in K} \sum_{\omega \in \Omega^k} c_\omega \zeta_\omega \tag{1}$$

subject to

$$\sum_{f \in F} \sum_{\omega \in \Omega^f} a_\omega^i \theta_\omega = 1 \quad (i \in N), \tag{2}$$

$$\sum_{k \in K} \sum_{\omega \in \Omega^k} a_\omega^i \zeta_\omega = 1 \quad (i \in N), \tag{3}$$

$$\sum_{k \in K} \sum_{\omega \in \Omega^k} b_\omega^{ij} \zeta_\omega - \sum_{f \in F} \sum_{\omega \in \Omega^f} b_\omega^{ij} \theta_\omega \leq 0 \quad ((i, j) \in C), \tag{4}$$

$$\sum_{\omega \in \Omega^f} \theta_\omega = 1 \quad (f \in F) \tag{5}$$

$$\sum_{\omega \in \Omega^k} \zeta_\omega = 1 \quad (k \in K) \tag{6}$$

$$\theta_\omega \in \{0, 1\} \quad (f \in F; \omega \in \Omega^f), \tag{7}$$

$$\zeta_\omega \in \{0, 1\} \quad (k \in K; \omega \in \Omega^k). \tag{8}$$

The objective function (1) minimizes the sum of all aircraft routing and crew scheduling costs. Constraints (2) and (3) ensure that each leg is covered by exactly one aircraft and one crew while constraint (4) guarantees that a crew does not change aircraft when the connection time is too short. Constraints (5) and (6) state that a path must be assigned to each aircraft and each crew.

For most airlines, the cost of a duty period is proportional to the maximum of three quantities: (a) the total flying time, (b) a fraction of the total elapsed time, and (c) the minimum guaranteed number of hours per duty period. The cost of a pairing is also roughly equal to the maximum of three quantities: (a) the sum of the costs of the individual duties that constitute the pairing; (b) a fraction of the total duration of the pairing (or *time away from base*); and (c) a minimum guaranteed time per duty multiplied by the number of duties in the pairing. In addition, when an overnight rest period takes place in a city that is not the crewbase, additional expenses are incurred for hotel, meals, and transportation. Finally, *deadhead* costs are incurred when crews travel as passengers on certain flights.

Deadheads are useful to reposition crews in a different city where they are needed to cover a flight leg. They can also be used to ensure that the crew can return to their base at the end of a pairing. Deadheads are introduced in the model by replacing each flight node $i \in N$ by a pair of distinct departure and arrival nodes. These two nodes are then linked by a regular flight arc and a deadhead arc, and constraints (2) and (3) are modified accordingly.

2. Solution Methodology

Given the potentially large size of the sets Ω^f and Ω^k, model (1)–(8) can be solved by a branch-and-bound algorithm, in which linear relaxation lower bounds are computed by a column generation approach (see Dantzig and Wolfe 1960). We first explain the functioning of this approach. We then describe how model (1)–(8) can be decomposed through Benders decomposition so as to take advantage of its particular structure. In the latter decomposition, both the master problem and the subproblem can be solved by column generation.

2.1. Column Generation

At each node of the branch-and-bound tree, the linear relaxation of model (1)–(8), together with the branching decisions applicable to the current node, is solved by an iterative column generation process that relies on a restricted master problem and a set of subproblems. The restricted master problem is obtained by replacing the sets Ω^f and Ω^k by the subsets $\Omega_t^f \subseteq \Omega^f$ and $\Omega_t^k \subseteq \Omega^k$ of paths available at iteration $t = 0, 1, \ldots$. This problem, which is solved by the primal simplex algorithm, finds the best solution for the restricted subset of path variables. It also provides dual variable values that are needed in the subproblems to identify new path variables with negative reduced cost.

The column generation process starts with an empty subset of path variables (artificial variables are added to the restricted master problem to ensure its feasibility during initial iterations). New variables for the master problem are then generated at each iteration by solving a resource-constrained shortest-path problem for each network G^f $(f \in F)$ and G^k $(k \in K)$. In these networks, arc costs are modified to reflect the current values of the dual variables associated with the constraints of the restricted master problem. New path variables are then added to the current restricted master problem, which is reoptimized to yield a new primal solution as well as new dual-variable values. The column generation process stops when no negative-cost path can be identified in any of the networks.

To obtain an integer solution of model (1)–(8), different branching schemes can be devised. An exact branching method is obtained by making branching decisions on sequences of flight legs to be covered by the same aircraft or the same crew (see, e.g., Desaulniers et al. 1998). For large instances, this scheme is likely to require excessive computing time because of the large number of possible connections. Another scheme consists of branching directly on the flow variables θ_ω and ζ_ω. However, because imposing the constraint $\theta_\omega = 0$ or $\zeta_\omega = 0$ would result in the same path being generated again by the subproblems, unless a special, and very time-consuming, shortest-path algorithm is used (see, e.g., Villeneuve, 2000), one must often be content with finding heuristic solutions by a depth-only search. In this case, a fractional variable θ_ω (or ζ_ω) is selected at each node of the tree and the constraint $\theta_\omega = 1$ or $(\zeta_\omega = 1)$ is added to the restricted master problem. If several variables take fractional values that exceed a certain threshold, branching decisions can also be made simultaneously on all variables to accelerate the search. This process is repeated until all variables satisfy integrality requirements or the restricted master problem becomes infeasible. If infeasibility occurs, the search should backtrack to the previous node, and an alternative fractional variable should be chosen for branching. If branching on any fractional variable results in an infeasible problem, the search should backtrack farther. In our experiments, backtracking was never necessary to avoid infeasibility.

2.2. Benders Decomposition

When the number of flight legs and the number of short connections each exceed a few hundred, solving model (1)–(8) by column generation becomes difficult because of the large number of constraints in the restricted master problem. In computational experiments, we have observed that computation time seems to grow quadratically with the number of constraints in the restricted master problem. Model (1)–(8) can, however, be decomposed so as to obtain a pair of problems that can be solved more easily. We first show how the LP relaxation of (1)–(8) can be solved by Benders decomposition and then explain how this solution method can be embedded in a branch-and-bound approach to obtain integer solutions.

For given nonnegative values $\bar{\theta}_\omega$ $(f \in F;\ \omega \in \Omega^f)$ satisfying aircraft constraints (2) and (5), the LP relaxation of model (1)–(8) reduces to the following *primal subproblem* involving only crew variables:

$$\text{Minimize} \sum_{k \in K} \sum_{\omega \in \Omega^k} c_\omega \zeta_\omega \tag{9}$$

subject to

$$\sum_{k \in K} \sum_{\omega \in \Omega^k} a_\omega^i \zeta_\omega = 1 \quad (i \in N), \tag{10}$$

$$\sum_{k \in K} \sum_{\omega \in \Omega^k} b_\omega^{ij} \zeta_\omega \le \sum_{f \in F} \sum_{\omega \in \Omega^f} b_\omega^{ij} \bar{\theta}_\omega \quad ((i,j) \in C), \tag{11}$$

$$\sum_{\omega \in \Omega^k} \zeta_\omega = 1 \quad (k \in K), \tag{12}$$

$$\zeta_\omega \ge 0 \quad (k \in K;\ \omega \in \Omega^k). \tag{13}$$

Let $\boldsymbol{\beta} = (\beta_i | i \in N)$, $\boldsymbol{\gamma} = (\gamma_{ij} \leq 0 | (i,j) \in C)$, and $\boldsymbol{\delta} = (\delta_k | k \in K)$ be the dual variables associated with constraints (10), (11), and (12), respectively. The dual of (9)–(13) is the following *dual subproblem*:

$$\text{Minimize} \sum_{i \in N} \beta_i + \sum_{(i,j) \in C} \sum_{f \in F} \sum_{\omega \in \Omega^f} b_\omega^{ij} \bar{\theta}_\omega \gamma_{ij} + \sum_{k \in K} \delta_k \quad (14)$$

subject to

$$\sum_{i \in N} a_\omega^i \beta_i + \sum_{(i,j) \in C} b_\omega^{ij} \gamma_{ij} + \delta_k \leq c_\omega \quad (k \in K; \omega \in \Omega^k), \quad (15)$$

$$\gamma_{ij} \leq 0 \quad ((i,j) \in C). \quad (16)$$

Assuming that $c_\omega \geq 0$ for all $k \in K$ and $\omega \in \Omega^k$, the dual subproblem is always feasible since the null vector $\mathbf{0}$ satisfies constraints (15) and (16). Hence, the primal subproblem is either infeasible or it is feasible and bounded. Let $\boldsymbol{\Delta}$ denote the polyhedron defined by constraints (15) and (16), and let $P_{\boldsymbol{\Delta}}$ and $R_{\boldsymbol{\Delta}}$ be the sets of extreme points and extreme rays of $\boldsymbol{\Delta}$ respectively.

Introducing the additional free variable z_0, the LP relaxation of model (1)–(8) can thus be reformulated as the following *Benders master problem*:

$$\text{Minimize} \sum_{f \in f} \sum_{\omega \in \Omega^f} c_\omega \theta_\omega + z_0 \quad (17)$$

subject to

$$z_0 - \sum_{f \in F} \sum_{\omega \in \Omega^f} \sum_{(i,j) \in C} b_\omega^{ij} \gamma_{ij} \theta_\omega \geq \sum_{i \in N} \beta_i + \sum_{k \in K} \delta_k \quad ((\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}) \in P_{\boldsymbol{\Delta}}), \quad (18)$$

$$- \sum_{f \in F} \sum_{\omega \in \Omega^f} \sum_{(i,j) \in C} b_\omega^{ij} \gamma_{ij} \theta_\omega \geq \sum_{i \in N} \beta_i + \sum_{k \in K} \delta_k \quad ((\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}) \in R_{\boldsymbol{\Delta}}), \quad (19)$$

$$\sum_{f \in F} \sum_{\omega \in \Omega^f} a_\omega^i \theta_\omega = 1 \quad (i \in N), \quad (20)$$

$$\sum_{\omega \in \Omega^f} \theta_\omega = 1 \quad (f \in F), \quad (21)$$

$$\theta_\omega \geq 0 \quad (f \in F; \omega \in \Omega^f). \quad (22)$$

Feasibility constraints (19) ensure that the values given to the θ_ω variables lead to a bounded dual subproblem. The value of z_0 is then restricted to be larger than or equal to the optimal value of this subproblem by optimality constraints (18). In general, model (17)–(22) contains more constraints than the LP relaxation of (1)–(8) but most optimality and feasibility constraints are inactive in an optimal solution. Hence, these constraints need not be generated exhaustively. Instead, an iterative algorithm is used to generate only a subset of cuts that are sufficient to identify an optimal solution.

Each iteration of the algorithm solves a relaxed Benders master problem obtained by replacing the sets $P_{\boldsymbol{\Delta}}$ and $R_{\boldsymbol{\Delta}}$ with the subsets $P_{\boldsymbol{\Delta}}^t \subseteq P_{\boldsymbol{\Delta}}$ and $R_{\boldsymbol{\Delta}}^t \subseteq R_{\boldsymbol{\Delta}}$ of extreme points and extreme rays available at iteration $t = 0, 1, \ldots$. The optimal solution of the relaxed Benders master problem is used to set up the primal subproblem (9)–(13). If the primal subproblem is feasible, the values of the dual variables associated with constraints (10)–(12) determine an extreme point of $P_{\boldsymbol{\Delta}}$. Otherwise, an extreme ray of $R_{\boldsymbol{\Delta}}$ violating one of the constraints (19) is identified. Hence, exactly one constraint is added to the relaxed Benders master problem at each iteration, and the process continues until its optimal solution yields a feasible primal subproblem whose cost is equal to the value of z_0.

At each iteration of the Benders decomposition algorithm, the relaxed Benders master problem and the primal subproblem can both be solved by column generation. When the primal subproblem is feasible, the values of the dual variables associated with the constraints of the column generation master problem identify an extreme point of $\boldsymbol{\Delta}$, even though only a subset of all columns has been generated. In fact, columns that have not been generated correspond to constraints of the dual subproblem that are already satisfied by the current solution. To avoid generating feasibility constraints from extreme rays in the case when the primal subproblem is infeasible, this subproblem can be made feasible for any solution of the relaxed Benders master problem. This is accomplished by introducing artificial variables, with large positive costs, in constraints (10) and (12). Constraints (19) are then no longer necessary, because the dual subproblem polyhedron becomes bounded and contains no ray. Indeed, introducing these artificial variables in the primal subproblem imposes upper bounds on the dual subproblem variables β_i and δ_k. Hence, all dual-subproblem variables become bounded, and $\boldsymbol{\Delta}$ contains no ray.

Optimal integer solutions of model (1)–(8) can be computed by using the branch-and-bound approach of §2.1 with exact branching rules and solving the linear relaxations by Benders decomposition and column generation. When integrality gaps are not too large, heuristic solutions can also be computed with the following three-phase approach.

In Phase I, all integrality requirements are relaxed, and the LP relaxation of model (1)–(8) is solved to optimality by Benders decomposition and column generation. Retaining all cuts generated in this first phase, Phase II reintroduces integrality constraints on the master problem aircraft path variables and solves the resulting mixed-integer problem by generating additional cuts. In this phase, an integer master problem must be solved at each iteration of the Benders decomposition algorithm. This problem is not necessarily solved to optimality at each iteration, because heuristic branching is performed on path variables, as explained at the end of §2.1. As a result, the Benders decomposition algorithm may stop with a suboptimal solution, even though this solution satisfies the stopping criterion. In Phase III, integrality constraints are finally added on subproblem crew path variables, and the integer subproblem is solved once, while the values of the master problem variables are held fixed. In this last phase, branching is also performed directly on the path variables. It is worth mentioning that because the subproblem does not have the integrality property, the integer subproblem may be infeasible for the given solution of the master problem, even if the original problem is feasible. In practice, however, solutions of very good quality were obtained with this three-phase approach.

Applications of Benders decomposition to transportation problems can be found, among others, in previous work by Florian et al. (1976), Vander Wiel and Sahinidis (1996), and Cordeau et al. (2000, 2001). The three-phase approach described above was used successfully by Cordeau et al. (2000, 2001) for solving simultaneous locomotive and car assignment problems. In that application, enormous reductions in computation times were achieved by first relaxing integrality constraints on master problem variables. Furthermore, even though the Benders decomposition subproblem did not possess the integrality property, optimal solutions were found with this approach, as there was no integrality gap in the subproblem. Because crew scheduling problems tend to exhibit small integrality gaps, this approach is likely to produce good quality solutions in the current application also.

3. Computational Experiments

In this section, we present computational experiments that were carried out on instances based on data provided by a large Canadian airline. We first provide a description of these instances, followed by a summary of our computational experiments.

3.1. Description of Data Sets

To generate test instances, we started from a weekly schedule, provided by the airline, that contains a total of 3,205 short-haul flight legs. Using a fleet assignment algorithm similar to that of Hane et al. (1995), we first assigned an aircraft type to each scheduled flight so as to maximize total anticipated profits. The airline has six aircraft types available to cover these flights, but in the fleet assignment solution a total of 2,950 flights were covered by just three of these six types. We thus constructed three instances from these flights, discarding the remaining 255 flights that were covered by the other three types.

To obtain valuable test instances, a certain number of steps were needed to complete the partial data provided by the airline. First, we solved a weekly aircraft routing problem to estimate the required fleet size and determine initial positions for the aircraft. This was accomplished by minimizing the objective (1) under constraints (2) and (5)–(7). Because fleet size and initial and final positions were at first unknown, the cardinality of F was set to a large number and for each aircraft $f \in F$, we defined $O^f = \{(o^f, i) : i \in N\}$ and $D^f = \{(i, d^f) : i \in N\}$. To each aircraft f, we also assigned an integer m_f, randomly generated from a uniform distribution over the set $\{0, 1, 2, 3\}$, representing the number of days elapsed since the last maintenance. Every aircraft was then required to spend a minimum of 8 hours, at least once every 4 days, in one of the four cities where maintenance can be performed. An additional arc (o^f, d^f) was also introduced in the set A^f to allow extra aircraft to remain idle during the complete planning horizon. Assigning a large cost to arcs in O^f allowed us to determine a minimum fleet size, as well as initial conditions that were used in all further experiments. Because this process makes aircraft with small values of m_f more likely to be used in the solution, each aircraft that covered at least one flight in the solution was assigned a new integer m_f, randomly generated as explained above, to obtain unbiased initial conditions. The aircraft routing problem was then solved with the reduced fleet size and new m_f values to ensure that all instances remained feasible after performing this adjustment.

The last step in constructing test instances consisted of solving a crew scheduling problem to determine the number of crews necessary, as well as their initial positions, at the beginning of the planning horizon. To do this, we minimized (1) under constraints (3) and (6)–(8), thus neglecting linking constraints (4). As in the case of aircraft, the cardinality of K was first set to a large number, and no specific initial or final conditions were imposed. Each crew was, however, randomly assigned to one of the four crewbases used by the airline. In addition, each crew k was assigned an integer w_k, chosen randomly in the set $\{0, 1, 2, 3\}$, representing the cumulated workdays at the beginning of the planning horizon. When solving the crew scheduling problem, crews are required to return to their base after a maximum of 4 workdays and can begin a new pairing after spending at least 1 full day at the base. In addition, the duty period for each workday must respect limits on total flight time, total duty time, and total number of landings. Again, assigning a positive cost to arcs in O^k allowed us to determine the required number of crews, as well as initial conditions, that were used in the subsequent experiments. After solving this problem once, each crew covering at least one flight leg in the solution was assigned a new random integer w_k, and the problem was solved again by retaining only these active crews. Here, additional crews were introduced in each instance to ensure the feasibility of the problem with the linking constraints (4).

Because the set of flight legs to be flown varies significantly from one day to the next in these test instances, solving a daily problem would not be appropriate. Instead, one should either solve a weekly problem or a sequence of shorter subproblems on overlapping time horizons. For test purposes, we thus generated instances with horizons varying between one and three days. However, when solving a p-day problem, an extended horizon of $p+3$ days is in fact considered: aircraft and crew path constraints (5)–(8) are imposed over the complete horizon, whereas covering and linking constraints (2)–(4) are imposed only for the planning horizon, i.e., the first p days. This strategy reduces the effects of the heuristic time horizon decomposition by ensuring the existence of aircraft routes and crew pairings that satisfy maintenance constraints and maximum pairing length constraints, respectively.

The characteristics of the different instances generated are summarized in Table 1. In this table, $|L|$ and $|C|$ correspond to the number of legs and the number of short connections in the first p days of the planning horizon, i.e., those for which constraints (2)–(4) are imposed. Here, a connection between two flight legs l_i and l_j is feasible if the difference between the departure time of leg l_j and the arrival time of leg l_i is greater than or equal to a station-specific minimum plan turn time t_s, that normally varies between 20 and 45 minutes. However, the connection is said to be short if the connection time exceeds t_s by less than 30 minutes. Hence, if the minimum plan turn time t_s is equal to 30 minutes, all connections whose duration lies between 30 and 60 minutes are said to be short. The numbers in Table 1 also provide the size of model (1)–(8). For example, one can verify that the column generation master problem for the largest instance, D9S-3, has a total of $2 \times 525 + 470 + 35 + 67 = 1,622$ constraints. For this instance, if model (1)–(8) is solved by Benders decomposition, the relaxed Benders master problem of the first iteration has $525 + 35 = 560$ constraints while the subproblem has $525 + 470 + 67 = 1,062$ constraints.

As explained in the introduction, airlines sometimes consider through values when solving the aircraft routing problem. In our experiments, pairs of flight legs to be covered by the same aircraft were specified directly by the airline according to exogenous marketing considerations. These pairs involved only a small fraction of all flight legs (about 5%),

Table 1 Characteristics of Test Instances

Instance	Planning Horizon (p) (days)	No. of Legs ($\|L\|$)	No. of Short Connections ($\|C\|$)	No. of Aircraft ($\|F\|$)	No. of Crews ($\|K\|$)
A320-1	1	140	93	34	68
A320-2	2	273	211	34	68
A320-3	3	406	324	34	68
CRJ-1	1	128	85	21	51
CRJ-2	2	260	183	21	51
CRJ-3	3	394	305	21	51
D9S-1	1	166	128	35	67
D9S-2	2	350	309	35	67
D9S-3	3	525	470	35	67

and their feasibility was validated when solving the fleet assignment problem. Hence, through values do not need to be considered explicitly in the aircraft routing problem. For every feasible through, all arcs that are incompatible with the implied connection are removed from the aircraft networks G^f $(f \in F)$. Of course, all legs that are not part of a through can be covered before and after any other leg with which an aircraft connection is possible. Finally, crew networks are not affected by this process.

Since through values are not considered in the aircraft routing problem, the only costs that remain in the problem, once fleet size is determined and fixed, are the operational costs that are associated with the flight legs. However, because all legs must be covered exactly once and all aircraft of a given type are assumed to have equal operating costs, this is in fact a fixed cost. The aircraft routing problem thus becomes a feasibility problem to ensure that each aircraft is maintained appropriately.

Because each flight leg must be covered by exactly one crew, a large portion of total crew costs is also fixed in the sense that each flight must be covered by one crew. Hence, the only relevant costs considered in these experiments are those that can be reduced by a better planning of crew pairings. Variable expenses are incurred for connections whose duration exceeds a given threshold because crews must then be credited work time even though they are not actually working. Additional accommodation expenses are also incurred when the rest period between successive duties does not take place at the crewbase. Finally, deadhead expenses are incurred when crew members travel as passengers on flights to which they are not assigned to work. Reasonable approximations of variable expenses can thus be obtained as follows: (a) if a rest period between two consecutive duty periods does not take place at the crewbase, a fixed cost is imposed to account for accommodation; (b) if the duration of a connection exceeds a given threshold (e.g., 45 minutes), a cost proportional to the connection time is imposed to account for paid waiting time; (c) if crew members must travel as passengers on a flight leg, a deadhead cost is imposed. Because we do not consider the fixed portion of crew costs, our approximation is in fact similar to that used by Barnhart and Shenoi (1998), who use the total duration of the pairing (time away from base) to estimate the cost. The objective of our experiments was to verify whether significant savings result from integrating aircraft routing and crew scheduling decisions. Hence, although capturing exact crew costs in a path-based formulation solved by column generation can be done as explained by Vance et al. (1997), we believe that the additional implementation effort was not justified in this context.

Finally, because we were provided only partial information regarding costs, the exact coefficients used in the experiments were parameterized so that variable crew costs represent approximately 5% of total crew costs, a percentage which is close to the industry average for short-haul flights.

3.2. Summary of Computational Experiments

To evaluate the benefits of using the combined Benders decomposition and column generation approach of §2.2 as opposed to solving model (1)–(8) directly with column generation only, we first tried to solve each of the nine instances with both methods. Our algorithms were coded in C and use GENCOL[1] for column generation. All experiments were performed on a Sun Enterprise 10000 (400 MHz) computer, using a single processor.

When solving the problem with column generation, the LP relaxation of the problem is solved to optimality but a heuristic branching scheme is used to find an integer solution, as explained in §2.1. For the combined Benders decomposition and column generation approach, the same branching scheme is used but the LP relaxation of Phase I is not always solved to optimality. Instead, cuts are generated until the relative difference between the lower bound provided by the Benders relaxed master problem and the upper bound provided by the subproblem is less than or equal to 1%. In both approaches, the column generation restricted master problem is, however, solved to optimality at every iteration of the column generation process.

In Table 2, we compare the CPU time and computational effort needed to solve each of the nine instances with the two approaches. For the basic column generation approach, we indicate the total number of branch-and-bound nodes explored during the solution process (*BB nodes*), the total number of column generation iterations performed (*CG iter.*), and the total number of columns generated (*Columns*). We also report the time spent solving the LP relaxation of the problem and the total CPU time (in minutes). The total CPU time can be divided into three parts: the time spent solving the column generation restricted master problem (*CPU MP*), the time spent solving the column generation shortest-path problems (*CPU CG*), and the overhead associated with problem parsing and algorithmic control. We finally report the cost of the LP relaxation of the problem (*Cost LP*), the cost of the heuristic integer solution that was obtained (*Cost IP*), and the percentage gap between these two costs.

For the combined Benders and column generation approach, we report similar statistics for the total effort of solving the relaxed Benders master problem and the subproblem by column generation. We also indicate the time spent in each of the first two phases, as well as the number of cuts generated in each phase. In this case, *Cost* $\widehat{LP}$ indicates the cost of the solution at the end of Phase I. This constitutes an upper bound on the true LP relaxation value of the problem because the LP relaxation is not necessarily solved to optimality. Finally, the percentage gap reported on the last line is computed with respect to the true LP relaxation value obtained when solving the LP relaxation exactly with the basic column generation approach.

Because the algorithm is stopped before reaching an optimal solution in the first phase of the Benders decomposition algorithm, comparing CPU times for solving the LP relaxation is difficult. However, when considering total CPU times for computing an integer solution, one observes that the combined Benders decomposition and column generation approach produces integer solutions faster than the basic column generation approach and that the difference in performance increases with problem size. For the largest instance (D9S-3), the former approach is more than four times faster than the latter. The difference in performance cannot be attributed only to the fact that the LP relaxation is not solved to optimality with the Benders decomposition approach. Indeed, one can see from Table 2 that when using the basic column generation approach, solving the LP relaxation of the problem often represents less than one-third of the total CPU time. The total CPU time is rather large because of the time spent in the branch-and-bound search. For example, on instance D9S-3, solving the LP relaxation took approximately 439 minutes, but an additional 815 minutes were required to find an integer solution. This figure is clearly larger than the 225 minutes spent in Phases II and III by using the combined Benders and column generation approach.

As briefly mentioned before, the difference in performance between the two methods can be attributed mostly to the fact that as problem size increases, a growing portion of total CPU time is spent solving the column generation restricted master problem. This is apparent by observing that the total CPU time spent in solving this problem goes from 45 minutes on instance D9S-2 which contains 1,111

[1] GENCOL is an optimization software that was developed at GERAD in Montreal.

Table 2 Computational Results for Basic CG and Combined Benders Decomposition and CG

	A320-1	A320-2	A320-3	CRJ-1	CRJ-2	CRJ-3	D9S-1	D9S-2	D9S-3
Basic CG									
BB nodes	8	61	80	39	44	63	51	80	93
CG iter.	44	459	1,167	164	910	2,777	259	930	2,920
Columns	1,405	5,682	12,444	1,903	7,899	25,475	3,671	13,882	42,654
CPU LP	0.51	3.70	19.79	1.03	8.72	112.44	1.84	25.93	438.76
CPU Total	0.60	14.54	62.09	3.34	28.29	377.81	7.82	93.77	1,253.98
CPU MP	0.04	5.76	30.87	0.22	11.29	302.08	0.44	44.99	1,004.94
CPU CG	0.50	8.59	30.84	3.03	16.84	75.25	7.23	48.38	247.96
Cost LP	28,528.3	51,674.5	75,946.4	21,314.7	35,162.1	46,358.4	20,041.8	49,072.9	68,872.5
Cost IP	28,528.3	52,250.8	80,338.3	22,541.7	36,806.7	50,349.2	20,230.0	49,398.3	70,759.2
Gap (%)	0.00	1.12	5.78	5.76	4.68	8.61	0.94	0.66	2.74
Benders with CG									
BB nodes	12	103	83	91	51	113	82	124	175
CG iter.	72	667	1,301	538	1,090	4,658	456	2,043	4,534
Columns	1,408	6,043	13,284	3,023	8,057	37,022	4,161	25,140	64,464
CPU Ph. I	0.44	2.82	9.05	1.00	4.50	23.87	2.93	23.45	64.66
CPU Ph. II	0.07	1.74	3.77	1.18	2.84	55.99	2.30	16.98	161.75
CPU Total	0.56	7.41	23.71	3.24	13.76	106.79	5.86	63.95	289.36
CPU MP	0.04	1.44	7.40	0.28	2.45	57.62	0.27	14.12	163.43
CPU CG	0.36	5.58	15.94	2.60	11.04	48.61	5.01	48.97	124.75
Cuts Ph. I	1	4	3	3	2	2	5	7	3
Cuts Ph. II	0	1	0	3	0	2	1	1	3
Cost $\widehat{\text{LP}}$	28,528.3	51,745.7	76,251.8	21,319.5	35,190.7	46,426.5	20,061.5	49,261.6	68,998.5
Cost IP	28,528.3	52,680.8	77,301.7	21,814.2	35,755.0	47,945.8	20,236.7	51,386.7	69,642.5
Gap (%)	0.00	1.95	1.78	2.34	1.69	3.42	0.97	1.80	1.12

Note: All CPU times are in minutes.

constraints in the master problem to more than 1,000 minutes on instance D9S-3, which contains 1,622 constraints. When using Benders decomposition on the latter instance, the column generation master problems for the aircraft routing problem and the crew scheduling problem have, respectively, 560 and 1,062 constraints. It has often been observed that computation times tend to grow quadratically with the number of constraints in the master problem when one is using column generation. This increase is mainly a result of the larger number of simplex pivots that must be executed when solving the restricted master problem. On most instances, one can also notice that, even though more column generation iterations are performed and more columns are generated when using Benders decomposition, the total time spent solving the column generation shortest-path problems is larger in the basic column generation approach. This is explained by the fact that in some iterations of the latter approach, all shortest-path problems are solved, but only a few of them generate negative reduced cost columns.

One can also observe that the good performance of the combined Benders decomposition and column generation approach is in part explained by the fact that very few cuts are generated in the second phase when the relaxed Benders master problem must be solved at each iteration with integrality constraints. Because the aircraft routing problem is not affected much by the introduction of integrality constraints, few additional cuts have to be generated before finding a good solution to the mixed-integer problem.

Although a heuristic stopping criterion is used for the combined Benders decomposition and column generation approach, the performance improvement does not come at the price of lower solution quality. In Table 2, one can check that the average percentage gap between the cost of the integer solution found and that of the LP relaxation solution is less than 1.7% when using the combined approach. This average gap is smaller than that obtained when solving the problem with the basic column generation approach. This result may seem surprising at first sight but is explained as follows. When using the Benders decomposition approach, the aircraft routing solution changes from one iteration to the next in Phase II of the algorithm when the aircraft routing problem is solved with integrality constraints but the crew scheduling problem is solved as a continuous problem. Hence, even if a depth-only search is used, a poor branching decision made when solving the aircraft routing problem in one of these iterations is not likely to have dramatic consequences because the resulting aircraft routing solution will be discarded after generating a new Benders cut, and a new solution that yields a crew scheduling problem with a lower cost will be computed. When using the basic column generation approach, however, poor branching decisions made on the aircraft routing variables cannot be reversed. This explains the large gaps that are sometimes observed for the basic algorithm. Of course, if an exact branching scheme were used, the conclusion might be different. In the course of our experiments, we did try branching on sequences of flight legs as explained at the end of §2.1. However, this often resulted in CPU times exceeding 24 hours for the larger instances.

Finally, it is worth mentioning that in these experiments we solve the LP relaxation heuristically when using Benders decomposition because of slow convergence when nearing optimality in Phase I. When trying to solve the LP relaxation to optimality, CPU times increase dramatically even for small instances, often making the approach comparatively slower than the basic column generation approach. Run times are always an important aspect of any optimization tool to be used in practice. Obviously, if the model is solved once a month before preparing personalized schedules for the crews, run times may not be so critical. Nevertheless, run times exceeding 20 hours (see, e.g., instance D9S-3) are usually not acceptable for airline planners who want to compare different scenarios before making a decision. Furthermore, several successive overlapping problems must be solved to obtain monthly solutions. Finally, as mentioned in the introduction, the model could also be used in short-term planning to modify crew and aircraft schedules following operational perturbations. In the latter context, short response times are extremely important.

In the second part of our experiments, we wanted to compare the cost of the solutions obtained with the integrated model (1)–(8) to those obtained with a sequential planning approach in which the aircraft routing problem is solved first and the crew scheduling problem (9)–(13) is solved next, taking the aircraft routing solution as an input. Because the aircraft routing problem is in fact a feasibility problem in which all variables have a cost of 0, it usually has several alternative (optimal) solutions. Hence, two different scenarios were used to compute aircraft routing solutions. In the first scenario (A), penalties were imposed on short connection arcs so as to avoid using them in the solution, thus maximizing the number of linking constraints (11) whose right-hand side is equal to 0 when solving the crew scheduling problem. This scenario does not necessarily generate the worst aircraft routing but is likely to make most of the linking constraints active. In the second scenario (B), negative penalties were assigned to all short connection arcs so as to make them more attractive when computing the aircraft routing solution and thus minimize the number of linking constraints that are active. In practice, solutions that are close to those of scenario A are often used because they are more robust and less vulnerable to delays. For both sequential approaches, the same column generation methodology was used and heuristic branching was performed as explained in §2.1.

Table 3 compares the cost of the solutions obtained with the different solution strategies. Columns *Scenario A* and *Scenario B* indicate the cost of the integer solutions obtained with the corresponding sequential planning approach, whereas the last column reports the cost of the integer solution produced by the combined Benders decomposition and column generation approach. These results show that for all instances, the simultaneous planning produced an integer solution of lower cost than those produced by both sequential solution scenarios. The total cost of scenario A and scenario B solutions exceeds that

Table 3 Crew Cost Comparisons Between Simultaneous and Sequential Planning

	Sequential Planning		
Instance	Scenario A	Scenario B	Simultaneous Planning
A320-1	29,902.5	29,005.0	28,528.3
A320-2	56,652.5	56,393.3	52,680.3
A320-3	85,069.2	82,105.8	77,301.7
CRJ-1	23,325.8	22,830.8	21,814.2
CRJ-2	40,794.2	36,314.2	35,755.0
CRJ-3	51,277.5	49,517.5	47,945.8
D9S-1	23,207.5	22,639.2	20,236.7
D9S-2	53,117.5	50,456.7	49,955.0
D9S-3	78,575.0	76,661.7	69,642.5
Total	441,921.7	425,924.2	403,859.5
Daily avg.	73,653.6	70,987.4	67,309.9

of the simultaneous planning solutions by 9.4% and 5.5%, respectively. The largest relative differences are obtained for instance D9S-1 and are larger than 14.6% for scenario A and 11.8% for scenario B. The last line of Table 3 reports the sum, over all three types of aircraft, of the average daily cost. The average daily cost for an aircraft type was computed by summing the costs of the three solutions for this aircraft type and dividing by 6 (the total number of days in the three instances). From these numbers, one can see that the average daily savings for the complete fleet of 90 aircraft are approximately $6,350 with respect to scenario A and $3,700 with respect to scenario B. These figures translate into annual savings of more than 2.3 and 1.3 million dollars, respectively. For larger airlines with hundreds of aircraft, the annual savings would thus represent several million dollars.

It is worth mentioning that because all relevant costs are in fact crew costs, it may seem reasonable first to determine crew pairings before trying to identify feasible aircraft routes. In the course of our experimentation, we thus implemented another sequential approach in which the crew scheduling problem was solved first, followed by the aircraft routing problem in which linking constraints were imposed: if a crew made a short connection between legs l_i and l_j, then the same aircraft had to cover both legs. However, this always resulted in an infeasible aircraft routing problem, unless fleet size was increased by a few units. This alternative approach could nevertheless be useful when using Benders decomposition.

Finally, another potential solution approach would be to use Langrangean relaxation by dualizing constraints (4). Although we have not experimented with this technique, it is likely that several iterations would have to be performed before finding appropriate values for the Lagrangean multipliers. At each iteration, one would have to solve the aircraft routing and the crew scheduling subproblems by column generation. In contrast, Table 2 shows that very few iterations have to be performed before finding feasible solutions of good quality. Hence, we believe that Benders decomposition is more appropriate because complete information from all previous iterations is conserved in the relaxed Benders master problem through the Benders cuts.

4. Conclusion

This paper has introduced a model and a solution methodology for the simultaneous routing of aircraft and scheduling of crews. The model incorporates aircraft maintenance constraints as well as the most important crew scheduling constraints. These constraints are in fact the basic ones that must be considered by most airlines and may completely represent the work rules of a small regional carrier or those of an airline with a simple collective agreement. It is thus sufficiently realistic to properly evaluate the benefits provided by a better integration of aircraft routing and crew scheduling decisions. On instances based on real data, the simultaneous approach produced important savings with respect to a traditional sequential planning process. It is worth mentioning that the data used in these experiments do not exhibit the popular hub-and-spoke structure. We believe that the potential for savings would be much higher for such networks because of the high number of arrivals and departures that take place within narrow time windows. In this context, there are several crew connections that are feasible only if the same aircraft is used on both legs. Future developments of our approach will address such network structures and will also consider the possibility of having some flexibility on the departure times of the flight legs so as to further reduce crew costs by slightly modifying the flight schedule in a long-term planning context.

Acknowledgments

This work was partially supported by the Québec Government (Fonds pour la Formation de Chercheurs et l'Aide à la Recherche), the Natural Sciences and Engineering Research Council of Canada, and Ad Opt Technologies Inc. This support is gratefully acknowledged. We also thank the referees for their valuable comments.

References

Abara, J. 1989. Applying integer linear programming to the fleet assignment problem. *Interfaces* **19** 211–232.

Barnhart, C., N. Boland, L. Clarke, E. Johnson, G. Nemhauser, R. Shenoi. 1998a. Flight string models for aircraft fleeting and routing. *Transportation Sci.* **32** 208–220.

——, L. Hatay, E. Johnson. 1995. Deadhead selection for the long-haul crew pairing problem. *Oper. Res.* **43** 491–499.

——, F. Lu, R. Shenoi. 1998b. Integrated airline schedule planning. *Operations Research in the Airline Industry.* Kluwer, Boston, 384–403.

——, R. Shenoi. 1998. An approximate model and solution approach for the long-haul crew pairing problem. *Transportation Sci.* **32** 221–231.

Clarke, L., C. Hane, E. Johnson, G. Nemhauser. 1996. Maintenance and crew considerations in fleet assignment. *Transportation Sci.* **30** 249–260.

——, E. Johnson, G. Nemhauser, Z. Zhu. 1997. The aircraft rotation problem. *Ann. Oper. Res.* **69** 33–46.

Cordeau, J-F., F. Soumis, J. Desrosiers. 2000. A Benders decomposition approach for the locomotive and car assignment problem. *Transportation Sci.* **34** 133–149.

——, ——, ——. 2001. Simultaneous assignment of locomotives and cars to passenger trains. *Oper. Res.* **49** 531–548.

Dantzig, G. B., P. Wolfe. 1960. Decomposition principle for linear programming. *Oper. Res.* **8** 101–111.

Desaulniers, G., J. Desrosiers, Y. Dumas, S. Marc, B. Rioux, M. Solomon, F. Soumis. 1997a. Crew pairing at Air France. *Eur. J. Oper. Res.* **97** 245–259.

——, ——, ——, M. Solomon, F. Soumis. 1997b. Daily aircraft routing and scheduling. *Management Sci.* **43** 841–855.

——, ——, I. Ioachim, M. Solomon, F. Soumis, D. Villeneuve. 1998. A unified framework for deterministic time constrained vehicle routing and crew scheduling problems. T. Crainic, G. Laporte, eds. *Fleet Management and Logistics*. Kluwer, Norwell, MA, 57–93.

Feo, T., J. Bard. 1989. Flight scheduling and maintenance base planning. *Management Sci.* **35** 1415–1432.

Florian, M., G. Bushell, J. Ferland, G. Guérin, L. Nastansky. 1976. The engine scheduling problem in a railway network. *INFOR* **14** 121–138.

Gopalan, R., K. Talluri. 1998. The aircraft maintenance routing problem. *Oper. Res.* **46** 260–271.

Graves, G., R. McBride, I. Gershkoff, D. Anderson, D. Mahidhara. 1993. Flight crew scheduling. *Management Sci.* **39** 736–745.

Hane, C., C. Barnhart, E. Johnson, R. Marsten, G. Nemhauser, G. Sigismondi. 1995. The fleet assignment problem: Solving a large-scale integer program. *Math. Programming* **70** 211–232.

Hoffman, K., M. Padberg. 1993. Solving airline crew scheduling problems by branch-and-cut. *Management Sci.* **39** 657–682.

Klabjan, D. 1999. Topics in airline crew scheduling and large scale optimization. Ph.D. dissertation. Georgia Institute of Technology, Atlanta, GA.

Lavoie, S., M. Minoux, E. Odier. 1988. A new approach for crew pairing problems by column generation with an application to air transportation. *Eur. J. Oper. Res.* **35** 45–58.

Stojković, G. 1999. Gestion des avions et des équipages durant le jour d'opération. Ph.D. dissertation. École Polytechnique de Montréal, Canada.

Talluri, K. 1998. The four-day aircraft maintenance routing problem. *Transportation Sci.* **32** 43–53.

Vance, P., C. Barnhart, E. Johnson, G. Nemhauser. 1997. Airline crew scheduling: A new formulation and decomposition algorithm. *Oper. Res.* **45** 188–200.

Vander Wiel, R. J., N. V. Sahinidis. 1996. An exact solution approach for the time-dependent traveling-salesman problem. *Naval Res. Logist.* **43** 797–820.

Villeneuve, B. 2000. Répartition et routage d'un système automatique de chariots à l'aide d'une méthode de génération de colonnes. Master's thesis. École Polytechnique, Montréal, Canada.

Yu, G. ed. 1998. *Operations Research in the Airline Industry.* Kluwer, Boston.

Received: August 2000; revisions received: January 2001; accepted: January 2001.