Integrated Airline Scheduling: Decomposition and Acceleration Techniques

Nikolaos Papadakos
Ph.D. Candidate

IC-PARC (centre for Planning And Resource Control), Imperial College London, SW7 2AZ, UK

nikolaos.papadakos@imperial.ac.uk

May 2006

Abstract

Airline scheduling is composed of fleet, maintenance, and crew decision subproblems. It is believed that the full problem is computationally intractable, and hence the constituent subproblems are solved sequentially so that the output of one is the input of the next. The sequential approach, however, provides suboptimal solutions and can also fail to satisfy the maintenance constraints of an otherwise feasible full problem. In this paper an integrated airline scheduling model is introduced, and its size is reduced by applying Benders decomposition combined with column generation. Several techniques are introduced to accelerate these decomposition algorithms. These techniques are generic enough to be applied to other airline problems. Solutions of several realistic data sets are computed using the integrated approaches, which are compared with solutions of the best known approaches from the literature. As a result, the integrated approach significantly reduces airlines costs, and the chosen formulation proves to be better than alternative integrations attempted.

1 Introduction

Airlines commence their tactical planning with *schedule generation* where the timetable of profitable legs is devised. A *leg* is a flight from a specific origin to a destination at given a departure time. Based on the timetable, airlines proceed onto solving the *airline scheduling* problem months before the day of operations. Since this problem is considered computationally intractable it is typically decomposed into its constituent stages, and the full problem is solved in a sequential manner where the output of one stage is the input of the next. In this manner, airlines initially solve the *fleet assignment* (`FA`) stage, deciding which fleet should fly each scheduled leg, while using the available aircraft and minimizing the operating costs. After this, the *maintenance routing* (`MR`) stage is solved, ensuring that aircraft will periodically be scheduled for maintenance, by devising individual aircraft itineraries. The obtained routings are used in the *crew pairing* (CP) stage, which devises the series of legs crew have to fly, while respecting labor rules and minimizing crew costs.

The knowledge of a MR solution is important for CP in order to determine whether crew can remain on the same aircraft for their following leg, rather than using precious time commuting within the airport to board a different aircraft; such a crew-connection is known as a *short-connection*. When considering short-connections in CP, one can solve instead of MR the *aircraft routing* problem, devising aircraft itineraries without maintenance constraints. In this case *plane-count constraints* are usually utilized (Klabjan et al., 2002, Sandhu and Klabjan, 2004), where short-connections, implying the repeated use of a specific aircraft, give the opportunity to evaluate the number of used aircraft.

Although the sequential procedure reduces computational complexity, because of the interdependence of each stage, the resulting solution is suboptimal. Even worse, in some cases feasible problems might not be solvable. As an example, Barnhart et al. (1998a) after solving the FA problem with the approximate MR considerations of Clarke et al. (1996) did not find MR solutions, to an otherwise feasible scheduling problem.

In order to circumvent the obstacles above, over the past years models considering several stages simultaneously have been proposed. The aim of these integrated models was to achieve better quality results, with the ultimate goal being to integrate all the stages. A model and a solution methodology that achieves complete integration of all stages is presented in this paper. The solution methodology is generic and can be applied to other airline operations research problems. Before going into detail about this paper's contributions, a summary of related work on integrated airline models and relevant solution methods is first presented.

1.1 Related work

Desaulniers et al. (1997b) integrated the FA problem with aircraft routing while giving legs' departure-time the flexibility to be within a time-window. Time-windows were also integrated with FA by Rexing et al. (2000), and with CP and plane-count constraints by Klabjan et al. (2002). Furthermore Barnhart et al. (1998a) exactly integrated FA with MR, and Lohatepanont and Barnhart (2004) combined schedule generation with itinerary based FA. Cohn and Barnhart (2003) integrated MR with CP, an integration also accomplished by Cordeau et al. (2001b) who reported evidence of cost savings.

Cordeau et al. (2001b) used Benders decomposition to handle the large number of constraints in their model. In this decomposition, the original problem is split into a master problem and a subproblem, with variables linking them (Benders, 1962). The algorithm iteratively solves the master problem and the subproblem. The solution of the master problem is passed to the subproblem through the linking variables. Then the solution of the subproblem introduces a cut on the master problem. In several cases, the master problem ends up having less constraints than the full problem, thus speeding-up the solution process.

Magnanti and Wong (1981) provided a theorem that accelerates Benders decomposition by finding the strongest possible cut that the Benders subproblem can provide, in the sense of Pareto-optimality. To use this theorem two preconditions have to be met. One first needs to solve the Benders subproblem to optimality, and then find a “core point”. In airline operations research, however, for tractability reasons, one is often content in obtaining a near-optimal solution. Even worse, a method for finding “core points” is not always available. Although these two preconditions might be prohibitive, Mercier et al. (2005) extended the methodology of Cordeau et al. (2001b) by using Pareto-optimal cuts with approximate “core points”. These approximations, however, were not that efficient when Mercier and Soumis (2006) extended the model to include time windows, and simple Benders cuts had to be used. Finally, since no method was available to find “core points”, Sandhu and Klabjan (2004) also had to use simple Benders cuts for their integrated `FA` and `CP` with plane-count constraints model.

1.2 Contributions

There are two areas of research this paper is contributing to: airline operations research in general and airline scheduling in specific.

Airline operations research

Regarding airline operations research, the preconditions for generating Pareto-optimal cuts are too stringent, as one cannot find a quick near-optimal solution for the Benders subproblem, and “core points” have to be allocated although a generic method does not exist. For this reason, in this paper the following are introduced concerning the Pareto-optimal cut generation problem:

- A proof that Magnanti and Wong’s version can be numerically unbounded when provided with a suboptimal Benders subproblem solution.
- A revised version independent of the Benders subproblem solution, but still requiring a “core point”.
- An extension of the previous version additionally independent of “core points”, for problems respecting a structure similar to `FA`.

Moreover, column generation is a basic ingredient for the solution of many airline operations research problems, as of those listed in the previous subsection. To the authors knowledge, the following methods accelerating column generation are implemented here for the first time:

- A heuristic deepest-cut pricing instead of Dantzig pricing.
- A dominance-relaxed constrained shortest path algorithm.

Airline scheduling

Concerning airline scheduling:

- An integrated model is presented, and a generic algorithm is devised to solve it.
- Based on realistic data of European and North American airlines, computational results are reported. These results compare the integrated method with the best known method from the literature, providing evidence that the former has significantly less costs than the latter.
- This integrated formulation proves more efficient than alternative integrated formulations attempted.
- Some of the alternative formulations include plane-count constraints combined with `MR`, and the results demonstrate that plane-count constraints alone cannot solve the `MR` problem.

1.3 Paper outline

Regarding the structure of this paper, in Section 2 sequential airline scheduling models, solution methods and techniques accelerating these methods are discussed. These models are amalgamated in Section 3 into an integrated model, on which a combination of Benders decomposition and column generation is applied. The Benders algorithm is accelerated with the introduction of special theorems concerning Pareto-optimal cuts in Section 4. In Section 5, the best known methods from the literature are presented and several alternative integrated formulations are summarized. Results of the performed computational experiments are presented in Section 6, comparing the integrated method with the best known method from the literature and with the alternative formulations. Finally, in Section 7 conclusions are drawn and future research directions are suggested.

2 Sequential models and methods

In order to lay the foundation for the integration developed in this paper, models and solution methods of sequential airline scheduling are discussed in this section. Additionally, several heuristics are introduced to accelerate these methods.

2.1 Fleet assignment

After devising the legs to be scheduled, airlines solve the fleet assignment problem, where the fleet flying each leg is decided. Given the departure-time the earliest time the aircraft will be ready to fly again has to be computed. This is known as *ready-time*, and depends both on the cruising speed of the aircraft, and the time needed for the crew to complete different tasks before take-off at a specific station. The FA cost is composed of: fuel and oil costs, landing fees, and loss of revenue for spilling passengers. The costs could potentially capture multi-leg passenger itineraries in the revenue component as Barnhart et al. (2002) discussed; this is not accounted here however.

In this paper the general structure of the FA model of Hane et al. (1995) is adopted:

$$\min \sum_{f \in F} \sum_{l \in L} c_{fl} x_{fl}, \tag{1a}$$

subject to

$$\sum_{f \in F} x_{fl} = 1, \quad \forall l \in L, \tag{1b}$$

$$\{Other\ Constraints\}, \tag{1c}$$

$$x_{fl} \in \{0, 1\}, \quad \forall l \in L, f \in F, \tag{1d}$$

where the notation is clarified in Table 1. In this generic formulation the goal (1a)

Table 1: Fleet assignment notation.

F	Set of fleets
L	Set of legs
c_{fl}	Cost of assigning fleet $f \in F$ to leg $l \in L$
x_{fl}	Binary variable representing the assignment of fleet $f \in F$ to leg $l \in L$ (= 1 if assigned)

is to minimize the total cost, and constraint (1b) is assigning one fleet per leg. The rest of the constraints (1c) are typically related to the routing of available aircraft for each fleet (Hane et al., 1995). The literature of FA is extensive, and the interested reader is referred to Sherali et al. (2006) for a review.

2.2 Maintenance routing

The solution of the FA problem partitions the legs into the set of those flown by each fleet, decomposing the maintenance routing problem. In MR an aircraft is assigned an itinerary that guarantees the periodic visit of *maintenance stations*, usually every 3 or 4 days, for a required time period, typically around 8 hours.

Sometimes additional constraints are placed on the maximum number of hours an aircraft is allowed to fly before going for maintenance. All these regulations are usually set by aircraft manufacturers and international organizations. With the assistance of maintenance routings it is possible to capture through flights' revenue as well as maintenance costs. A *through flight* is one where an aircraft is scheduled to go to a final destination after an intermediate short stop, which is cheaper as passengers do not have to change aircraft.

The `MR` model used here is in one-to-one correspondence with that of Cohn and Barnhart (2003). In the present model, a routing of an aircraft starts by performing an initial maintenance, flies a series of legs, and returns for its next maintenance, which will be the terminal one for that routing. The next routing flown by the same aircraft should have its initial maintenance at the same station and not earlier than the terminal maintenance of the previous routing. If these two maintenance opportunities do not coincide the aircraft can be maintained in either of them and remain on the ground for the rest of the time.

Maintenance routings are modeled as paths in a time-space network, known as *aircraft-connection network* (Mercier et al., 2005). The nodes of this network are departure-times of legs, as well as maintenance-start times. If the problem is solved in a weekly horizon, the nodes are duplicated once to account for the routings crossing over the end of the week. If the horizon is daily they are duplicated by the maximum number of days an aircraft can fly without maintenance, plus two days where the initial and terminal maintenance of the routing will be performed. In the aircraft-connection network there are directed arcs connecting leg nodes together, maintenance nodes with leg nodes, and leg nodes with maintenance nodes. In each case the connection is defined respectively: if the ready-time of the first leg is not later than the departure of the second, if the maintenance-completion time is not later than the departure-time of the leg, and if the ready-time of the leg is not later than the maintenance-start. In the daily scheduling horizon legs are repeated daily, hence the time difference between nodes should be a day at most.

The adopted `MR` model is the following problem:

$$\min \sum_{r \in R^f} c_r v_r, \tag{2a}$$

subject to

$$\sum_{r \in R^f} e_{lr} v_r = x_{fl}, \quad \forall l \in L, \tag{2b}$$

$$q_m - q_{m^-} + \sum_{r \in R^f} (e^+_{mr} - e^-_{mr}) v_r = 0, \quad \forall m \in M^f, \tag{2c}$$

$$\sum_{m \in M'^f} q_m + \sum_{r \in R^f} e'_r v_r \leq n_f, \tag{2d}$$

$$v_r \in \{0,1\}, \quad \forall r \in R^f. \tag{2e}$$

$$q_m \geq 0, \quad \forall m \in M^f, \tag{2f}$$

where the notation is clarified in Table 2. The objective function (2a) minimizes

Table 2: Maintenance routing notation.

r_f	A routing of fleet f
R^f	Set of routings of fleet f (distinct for each fleet)
$r \in R^f$	A simplified notation for a routing r_f (as R^f are distinct)
c_r	Cost of routing r
v_r	Binary variable representing the usage of routing r (=1 if used)
e_{lr}	Equal to 1 if leg l is in routing r
e'_r	Equal to the number of times routing r (excluding the terminal maintenance) crosses the scheduling horizon
m	The start time of a maintenance opportunity at a specific station
M^f	Set of maintenance start times of fleet f
M'^f	Set of maintenances crossing the scheduling horizon
m^-	Maintenance on the same station as m, predating m
m^+	Maintenance on the same station as m, postdating m
q_m	Variable counting the aircraft on the ground between m and m^+
e^+_{mr}	Equal to 1 if m is the initial maintenance of routing r
e^-_{mr}	Equal to 1 if m is the terminal maintenance of routing r
n_f	Number of available aircraft for fleet f

the total cost, and constraint (2b) restricts each leg assigned to fleet f to be flown by one routing exactly. Constraint (2c) conserves flow of routings consecutively flown by the same aircraft, and constraint (2d) restricts the used aircraft to be at most the number of available ones.

2.3 Crew pairing with short-connections

After determining aircraft itineraries, airlines have to schedule crew flying these aircraft. Usually crew are qualified to fly only one fleet, and for this reason the crew pairing problem, like `MR`, decomposes per fleet. A crew schedule is constructed on a monthly basis and it is made out of crew pairings, training sessions, and vacations. A *crew pairing* is a multi-day schedule for a crew group starting and terminating at the same crew base. Daily schedules are named *duties* and are made of legs that crew either operate or fly as passengers; the latter are named *deadheads*.

Pairings must respect labor and contractual regulations which usually guarantee maximum: flying time per duty, duration of a duty, and *time away from* *base*. Furthermore, there are regulations on the flying hours per day (*8-in-24 rule*), and on the maximum number of duties per pairing. Additionally there are minimum and maximum rest periods between duties as well as between flights of the same duty. The rest between flights is known as *sit-time*, and it is assumed here to be 30 minutes. The minimum sit-time is the time required for crew to travel within the airport to board to their next flight, and it is computed on top of the ready-time (Cordeau et al., 2001b, Mercier et al., 2005). This is because several tasks that have to be performed by the crew, i.e. aircraft inspections, are accounted for in the ready-time. These constraints are not hard, as they can be bent in two cases. Firstly, the 8-in-24 rule can be relaxed with additional rest periods. Secondly the minimum sit-time can be violated when the crew remain on the same aircraft. This can be known through a `MR` solution giving the sequence of flights flown by each aircraft. The crew-connection in violation of the minimum sit-time rule is known as a *short-connection*.

Concerning crew costs, for each specific pairing there is a minimum guaranteed pay per duty, a guaranteed percentage of duty time counting as flying time, and a per diem away from base payment. Finally, expenses of deadheading as well as crew overnight away from base must also be considered.

In the present case the well known *crew-connection network* is employed to model `CP`; for a thorough review of `CP` models see Barnhart et al. (2003). The crew-connection network is similar to the aircraft-connection network of the previous subsection. It is once more a time-space network, but in this case consists of leg departure-time nodes. Crew in this case can either operate these legs or fly as deadheads. When the problem is solved in a weekly horizon the nodes are duplicated once to account for the routings crossing over the end of the week. If the horizon is daily, they are duplicated by the maximum number of days crew are allowed to be away from base. The directed arcs are connecting leg with leg nodes whenever the arrival of the first leg is not later than the departure of the second leg, and not later than the maximum rest period between duties. A pairing is a path in the crew-connection network, starting with a leg leaving from the crew base and terminating with a leg arriving at the same base.

The `CP` with short-connections problem for fleet f is:

$$\min \sum_{p \in P^f} c_p w_p, \tag{3a}$$

subject to

$$\sum_{p \in P^f} a_{lp} w_p = x_{fl}, \quad \forall l \in L, \tag{3b}$$

$$\sum_{p \in P^f} s_p^{ij} w_p \leq s_f^{ij}, \quad \forall ij \in S^f, \tag{3c}$$

$$w_p \in \{0,1\}, \quad \forall p \in P^f, \tag{3d}$$

where s_f^{ij} is evaluated given the solution $v_r, r \in R^f$ for fleet f of MR model (2) by:

$$s_f^{ij} = \sum_{r \in R^f} s_r^{ij} v_r, \quad \forall ij \in S^f, \tag{4}$$

and notation in all cases is clarified in Table 3. The goal (3a) of CP is to minimize

Table 3: Crew pairing with short-connections notation.

p_f	A pairing of fleet f
P^f	Set of pairings of fleet f (distinct for each fleet)
$p \in P^f$	A simplified notation for a pairing p_f (as P^f are distinct)
c_p	Cost of pairing p
w_p	Binary variable representing the usage of pairing p (=1 if used)
a_{lp}	Equal to 1 if leg l is in pairing p
s_p^{ij}	Equal to 1 if short-connection of legs i and j is in pairing p
s_r^{ij}	Equal to 1 if routing r can short-connect legs i and j
s_f^{ij}	Equal to 1 if MR solution can short-connect legs i and j for fleet f
S^f	Set of pairs of legs that can be short-connected for fleet f

the total amount of crew costs, and constraint (3b) restricts each leg assigned to fleet f to be flown by one pairing only. Finally constraint (3c) allows a crew short-connection between two legs under the condition that there is an aircraft flying those legs consecutively.

2.4 Column generation

The *linear programming* (LP) relaxation of both the MR model (2) and CP model (3) are typically solved with the column generation method (Dantzig and Wolfe, 1960) and then the full *integer programming* (IP) with branch-and-price (Barnhart et al., 1998b). Paradigms of this methodology for CP and MR models can be found in Desaulniers et al. (1997a), Vance et al. (1997) and Mercier et al. (2005).

As an example with column generation for CP, one relaxes the problem by setting $w_p \geq 0$ and substitutes P^f with a feasible subset of pairings P_0^f. If such a subset is not available, one may consider artificial pairings with very high costs. This restricted problem is known as the *column generation master problem*. After solving the master problem one can consider the columns in its basis as the columns in the basis of the full problem (Chvátal, 1983).

Following the structure of the simplex method, in order to decide whether optimality is reached or new columns have to be introduced, one needs to solve a pricing problem, also known as *column generation subproblem*. Using the classic

Dantzig pricing, one has to find the column with the minimum reduced cost. The reduced cost of a column of problem (2) corresponding to a pairing p is given by:

$$\bar{c}_p = c_p - \sum_{l \in L} \alpha_l a_{lp} - \sum_{ij \in S^f} \varsigma^{ij} s_p^{ij}, \quad p \in P^f, \tag{5}$$

where α_l and ς^{ij} are the dual values of constraints (3b) and (3c) respectively. Since the subproblem is the pricing step of the simplex method, if there are negative reduced cost pairings they have to be added in the master problem, and continue iterating between the master problem and the subproblem; otherwise the iterations terminate with the optimum solution of the full problem in the basis of the master problem.

The pricing problem can be seen as a resource-constrained shortest path problem (Desrochers, 1986, Desrosiers et al., 1995). This is achieved by introducing lengths for the arcs connecting leg l with any other leg, and legs i with j, of the crew-connection network defined in Subsection 2.2, as instructed by equation (5). Additionally, the CP constraints have to be implemented for the paths in this network. With these modifications the shortest path will correspond to the pairing with the minimum reduced cost.

To solve the full IP problem a branch-and-bound tree is constructed and column generation is solved on each node of the tree. In such a framework it is useless branching a w_p variable on 0, since this non-null variable is in the basis and is therefore beneficial. Setting it to null removes it initially from the master problem, however, being beneficial, nothing prevents the subproblem from regenerating it. The theorem of Ryan and Foster (1981) devises a strategy where on one branch a pair of legs flown consecutively is imposed and on the other is forbidden; this pair of legs is named *follow-on*. The branching rule successfully adapts the branching in the subproblem by removing the relevant arcs in the crew-connection network (Desaulniers et al., 1997b, Barnhart et al., 1998b).

2.5 Accelerating column generation

Column generation is known for its poor convergence, especially when employing the simplex method on the master problem. In this case, although a near-optimal solution is approached quickly enough, the algorithm seems to be stalling close to optimum; this phenomenon is known as the *tailing-off effect*. Lasdon (1970) proposed a method for estimating a lower bound of a column generation problem, which can be used to compute the maximum optimality gap, that is the difference between the objective and lower bound. Column generation iterations can be terminated when this gap is less than a user-defined tolerance, giving a near-optimal solution of defined quality.

Heuristic deepest-cut pricing

There are other methods that can be used in conjunction with the Lasdon bound to accelerate the column generation, e.g. du Merle et al. (1999). One such good candidate could be steepest-edge pricing (Goldfrab and Reid, 1977) which is typically used in the simplex method to reduce the number of iterations. Its implementation, however, requires complex matrix computations that increase the solution-time. Thus instead, its dual pendent, known as deepest-cut, is chosen in this paper's implementation, and for problem (3) it is:

$$\bar{c}_p^{\mathrm{DC}} = \frac{\bar{c}_p}{\sqrt{1 + \sum_{l \in L} \left(a_{lp}\right)^2 + \sum_{ij \in S^f} \left(s_p^{ij}\right)^2}} = \frac{\bar{c}_p}{\sqrt{1 + \sum_{l \in p} 1 + \sum_{ij \in S^f \cap p} 1}}, \quad p \in P^f, \quad (6)$$

where the first equality is the definition of deepest-cut (Lübbecke, 2001) and the second is derived from the definitions of Table 3. Notice that $\sum_{l \in p} 1$ is equal to the number of legs in pairing p and $\sum_{ij \in S^f \cap p} 1$ is equal to the number of short-connections. Hence once given a pairing p it is trivial to compute $\bar{c}_p^{\mathrm{DC}}$.

To discuss how this is implemented, one has to look into the implementation of the resource-constrained shortest path algorithm. In this algorithm for each path reaching a node, labels are used to model the resource usage of `MR` or `CP` and apply constraints on them (Desaulniers et al., 1997a, 1998a,b). Fortunately, not all possible paths need to be stored, but by considering a *dominance relation* between the paths' labels, one eliminates those paths whose extension cannot lead to shorter paths than others will (Desrochers, 1986, Desrosiers et al., 1995). While processing each non-dominated path p the heuristic deepest-cut algorithm evaluates the $\bar{c}_p^{\mathrm{DC}}$. This way, upon termination of the algorithm, one firstly knows the pairing with most negative reduced cost and secondly has a pairing with a good deepest-cut value, although not optimal. The possibility of using the deepest-cut criterion in column generation was also discussed by Vanderbeck (1994) and Lübbecke (2001), however, it does not seem to have been implemented previously in the literature.

A dominance-relaxed constrained shortest path algorithm

One more technique is introduced here to further reduce column generation execution-time. One can accelerate the resource-constrained shortest path algorithm of the subproblem by not always considering all labels in the dominance relation. This is beneficial because for the complex constraints of `CP` not that many paths end up being dominated, hence a vast number of them are stored, rendering the solution of the full problem slow. Such an algorithm is named hereafter *dominance-relaxed constrained shortest path*, and in the present implementation for `CP` it runs in three different operating modes. In each mode the following labels are considered in the dominance relation:

I. Only the reduced cost and the time away from base.

II. All except for two: the sum of previous duties time flown, and the flown time during the last 24 hours (corresponding to the 8-in-24 rule).

III. All.

The algorithm always starts from Mode I, and in any mode solves the resource-constrained shortest path problem with the relevant labels in the dominance relation. If negative reduced cost paths are found in any mode, the algorithm terminates, returning them as a solution; otherwise the algorithm enters the next mode. If in the final mode no negative reduced cost paths are found the algorithm terminates with no solution.

3 An integrated model and its decomposition

The previously mentioned sequential airline scheduling models are merged in this section to form an integrated model. As seen earlier in the `FA` model (1) constraints concerning the routing and the availability of aircraft as those of Hane et al. (1995) were not explicitly given. This is because one can amalgamate this generic `FA` model with the `MR` model (2), obtaining a complete integration of `FA` with `MR`. Furthermore one can incorporate the combined `CP` with short-connections model (3), and with the assistance of equation (4), create an integrated airline scheduling model:

$$\min \sum_{f\in F}\sum_{r\in R^f} c'_r v_r + \sum_{f\in F}\sum_{p\in P^f} c_p w_p, \tag{7a}$$

subject to

$$\sum_{f\in F}\sum_{r\in R^f} e_{lr} v_r = 1, \quad \forall l \in L, \tag{7b}$$

$$\sum_{p\in P^f} a_{lp} w_p - \sum_{r\in R^f} e_{lr} v_r = 0, \quad \forall l \in L, \forall f \in F, \tag{7c}$$

$$\sum_{p\in P^f} s_p^{ij} w_p - \sum_{r\in R^f} s_r^{ij} v_r \le 0, \quad \forall ij \in S^f, \forall f \in F, \tag{7d}$$

$$q_m - q_{m^-} + \sum_{r\in R^f} (e^+_{mr} - e^-_{mr})\, v_r = 0, \quad \forall m \in M^f, \forall f \in F, \tag{7e}$$

$$\sum_{m\in M'^f} q_m + \sum_{r\in R^f} e'_r v_r \le n_f, \quad \forall f \in F, \tag{7f}$$

$$v_r \in \{0,1\}, \quad \forall r \in R, \tag{7g}$$

$$q_m \ge 0, \quad \forall m \in M, \tag{7h}$$

$$w_p \in \{0,1\}, \quad \forall p \in P, \tag{7i}$$

where equation (2b) was used to substitute all occurrences of x_{fl} variables, obtaining significant model size reduction in comparison to similar ones from the literature (Papadakos, 2006). Additionally R, M, and P are respectively the set of all routings, maintenance opportunities, and pairings. c'_r is the combined routing cost due to FA and MR. All these are defined by:

$$R = \cup_{f\in F} R^f, \quad (8a)$$

$$M = \cup_{f\in F} M^f, \quad (8b)$$

$$P = \cup_{f\in F} P^f, \quad (8c)$$

$$c'_r = c_r + \sum_{l\in L} c_{fl} e_{lr}. \quad (8d)$$

It should be stressed that all sets R^f, M^f, and P^f are distinct for each fleet f as the pairings, maintenance opportunities, and routings they are respectively made of are all distinct being related to legs with fleet-dependent ready-times. The goal (7a) is to minimize the total cost, and constraint (7b) assigns one routing flown by an aircraft of a specific fleet to one leg. This way a FA decision is accomplished and that decision is corresponded to a single crew pairing by constraint (7c) with the assistance of constraint (7b). Constraint (7d) allows a crew short-connection between two legs under the condition that there is an aircraft flying those legs consecutively. Finally, constraint (7e) preserves the flow of maintenance routings and constraint (7f) restricts the number of aircraft used to be not more than n^f.

3.1 Benders decomposition

The integrated model (7) has a large number of constraints and Benders decomposition is employed to reduce them. This method has recently been successfully used for the integrated MR with CP problem by Cordeau et al. (2001b) and Mercier et al. (2005), and in the extension of this model where time-windows were included (Mercier and Soumis, 2006).

The decomposition chosen here for model (7) consists of considering the CP problems of each fleet f as the Benders subproblem:

$$\min \sum_{p\in P^f} c_p w_p, \quad (9a)$$

subject to

$$\sum_{p\in P^f} a_{lp} w_p = \bar{x}_{fl}, \quad \forall l \in L, \quad (9b)$$

$$\sum_{p\in P^f} s_p^{ij} w_p \le \bar{s}_f^{ij}, \quad \forall ij \in S^f, \quad (9c)$$

$$w_p \ge 0, \quad \forall p \in P^f, \quad (9d)$$

where $(\bar{\mathbf{x}}, \bar{\mathbf{s}})^*$ are computed with the assistance of equations (2b) and (4), given $\bar{v}_r$ that satisfy the constraints of Benders master problem (11) to be shortly presented. Problem (9) is the same as (3) and the methods discussed in Subsections 2.4 and 2.5 will be used to solve it. Notice additionally that the above problem is an LP relaxation, as this is considered in classic Benders algorithm. The extension of the Benders algorithm by McDaniel and Devine (1977) to find the IP solution is discussed later in Subsection 4.3.

The dual of the Benders subproblem is utilized to obtain the Benders cuts. The dual of problem (9) is:

$$\max \sum_{l \in L} \bar{x}_{fl} \alpha_l + \sum_{ij \in S^f} \bar{s}_f^{ij} \varsigma^{ij} \tag{10a}$$

subject to

$$(\boldsymbol{\alpha}, \boldsymbol{\varsigma}) \in \Pi^f, \tag{10b}$$

where $\boldsymbol{\alpha} \equiv (\alpha_l | \, l \in L)$ and $\boldsymbol{\varsigma} \equiv \left(\varsigma^{ij} \middle| \, ij \in S^f \right)$, and Π^f is the polyhedron defined by the constraints of the problem.

The Benders master problem is formulated by replacing in the `AS` model (7) the `CP` constraints (7c), (7d) and (7i), with Benders cuts. One should also replace the `CP` cost for each fleet f in the objective (7a) with a variable z_f. Hence the Benders master problem is:

$$\min \sum_{f \in F} \sum_{r \in R^f} c'_r v_r + \sum_{f \in F} z_f, \tag{11a}$$

subject to

$$\sum_{f \in F} \sum_{r \in R^f} e_{lr} v_r = 1, \quad \forall l \in L, \tag{11b}$$

$$q_m - q_{m^-} + \sum_{r \in R^f} (e^+_{mr} - e^-_{mr}) \, v_r = 0, \quad \forall m \in M^f, \forall f \in F, \tag{11c}$$

$$\sum_{m \in M'^f} q_m + \sum_{r \in R^f} e'_r v_r \leq n_f, \quad \forall f \in F, \tag{11d}$$

$$-z_f + \sum_{r \in R^f} \left(\sum_{l \in L} \alpha_l e_{lr} + \sum_{ij \in S^f} \varsigma^{ij} s_r^{ij} \right) v_r \leq 0, \quad \forall \, (\boldsymbol{\alpha}, \boldsymbol{\varsigma}) \in \Pi^f_{\text{points}}, \forall f \in F, \tag{11e}$$

$$\sum_{r \in R^f} \left(\sum_{l \in L} \alpha_l e_{lr} + \sum_{ij \in S^f} \varsigma^{ij} s_r^{ij} \right) v_r \leq 0, \quad \forall \, (\boldsymbol{\alpha}, \boldsymbol{\varsigma}) \in \Pi^f_{\text{rays}}, \forall f \in F, \tag{11f}$$

$$v_r \in \{0, 1\}, \quad \forall r \in R, \tag{11g}$$

$$q_m \geq 0, \quad \forall m \in M^f, \tag{11h}$$

*From now on the simplified notation $\mathbf{x} \equiv \left(x_{fl} \middle| \, l \in L, f \in F \right)$ and $\mathbf{s} \equiv \left(s_f^{ij} \middle| \, f \in F, ij \in S^f \right)$ will be employed interchangeably.

where $\Pi^f_{\text{points}} \subseteq \Pi^f$ and $\Pi^f_{\text{rays}} \subseteq \Pi^f$ are respectively the sets of extreme points and extreme rays. The Benders cuts (11e) and (11f) are imposed by weak duality theorem applied on objectives (9a) and (10a), and by using equations (2b) and (4).

From the structure of this model and the decomposition employed it is not difficult to see that it is possible to further decompose, if needed, the CP Benders subproblems into cockpit and cabin crew. This can be achieved by formulating the relevant problems and substituting z_f by $z_f^{\text{cockpit}} + z_f^{\text{cabin}}$. One can finally have crew flying different types of aircraft in a similar manner to Sandhu and Klabjan (2004).

3.2 Benders master problem solution

The Benders master problem (11) is solved with the method of Barnhart et al. (1998a), and column generation is accelerated here with the techniques discussed in Subsections 2.4 and 2.5. There is a column generation subproblem for each fleet solved in the corresponding aircraft-connection network. These networks are distinct, as aircraft have fleet dependent ready-times. On the i-the iteration, the beneficial routings found for each fleet f will be added on the set R^f_i, whose routings are consequently distinct from any other fleet's routings, once more because of the different ready-times. For each routing $r \in R^f \subseteq R$ there is a fleet index implied, not denoted here explicitly. In the actual implementation, however, this information is of course kept with each routing.

The IP solution of the Benders master problem is accomplished through a branch-and-price algorithm, whose search is composed of two stages. The first stage assigns fleets to legs and the second determines follow-on legs in the routings.

In detail, on the first stage with the assistance of equation (2b) the fractional $\bar{x}_{fl}$ whose value is closest to 1 is allocated. On one branch fleet f is assigned to leg l, by setting to 0 all routing variables $v_r, r \in R^{\tilde{f}}, \tilde{f} \neq f$ containing that leg, and removing the leg from the aircraft-connection network of the rest fleets. On the other branch the assignment of fleet f to leg l is forbidden by setting to 0 all routing variables $v_r, r \in R^f$ containing it and removing leg l from the aircraft-connection network of fleet f.

After the first stage, the problem is partitioned into the legs flown by each fleet, and hence a reduced MR problem can be reconstructed for each fleet by adding the corresponding Benders cuts. In the second stage for these reduced problems the Ryan-Foster rule is used, and pairs of follow-on legs with fractional variables are imposed on one branch and disallowed on the other branch. Thus, on the first branch variables v_r containing any of these legs without them following each other are set to 0, and the arcs connecting these legs with any other leg are removed from the aircraft-connection network. On the second branch any variables containing those legs as follow-on are set to 0 and any arcs connecting them are eliminated. If the second stage is infeasible one has to backtrack on the first and attempt a different fleet assignment.

To reduce computational effort one can diminish the size of both column generation problems. One can fix to 1: x_{fl} on the first stage of the IP algorithm, and v_r on the second, whenever these are higher than 0.99. The former is achieved by setting to 0 all $v_r, l \in r, r \in R^{\tilde{f}}, \tilde{f} \neq f$, and removing leg l from the aircraft-connection network of all other fleets. The latter is achieved by setting to 0 all $v_{r'}$ variables whose routings r' contains any of the legs of r, and removing all the legs of r from the aircraft-connection network.

In order to further speed-up the algorithm columns will be generated on each node only while the objective exceeds a lower bound by more than 0.05%. The objective value obtained at the root node is initially used as that lower bound. If at a node column generation is solved to optimality, this optimal value will be considered as the lower bound for its child nodes.

As experimentation reveals, the basis of a column generation iteration is a good starting point for the next one, and primal simplex should be used. This is not very efficient on the first column generation iteration executed on each node of the search tree because the branching step introduces many changes, and the previous basis is no longer relevant. In this case barrier method with dual crossover proves the most efficient.

In the implemented algorithm, the search is depth-first and terminates upon the first encountered solution. Such IP heuristics are not uncommon in the airline scheduling literature, and have already been successfully applied by Barnhart et al. (1998a), Cordeau et al. (2001b), Mercier et al. (2005) and Mercier and Soumis (2006). Their success lies on the fact that the experiments have shown short IP solution-times with surprisingly low optimality gaps.

3.3 The Benders algorithm

The Benders algorithm is dual to column generation in the sense that constraints are added to a restricted master problem. One starts with empty $\Pi^f_{0,\text{points}}$ and $\Pi^f_{0,\text{rays}}$ sets of cuts, and on the i-th iteration the restricted Benders master problem is solved. The acquired solution $\bar{v}_r$, with the assistance of equations (2b) and (4), gives $\bar{x}_{fl}$ and $\bar{s}^{ij}_f$. These are usually passed to the dual Benders subproblems. In the present case, however, it is not sensible to solve this dual subproblem, and instead one solves the actual Benders subproblem (9) (Cordeau et al., 2001b, Mercier et al., 2005). From the solution of this subproblem, the dual values (α, ς) are acquired, and if the problem is feasible they are added in $\Pi^f_{i,\text{points}}$ generating an *optimality cut*; otherwise they are added in $\Pi^f_{i,\text{rays}}$ generating a *feasibility cut*. The Benders subproblem of each fleet is here the CP problem (9) and as already mentioned this is solved with column generation where artificial columns with high costs are used. If these artificial columns are in the optimal basis of the column generation master problem, then the original problem is infeasible. In this case the acquired dual values are not extreme rays but rather extreme points approximating them. For the rest of this paper the cut derived from an originally infeasible subproblem will still be named feasibility cut.

The Benders algorithm terminates when no more cuts need to be generated or if the Benders master problem is infeasible; the latter case implies that the full problem is infeasible. Sometimes one is content with near-optimal solutions and the algorithm can terminate when the gap between the upper and lower bound is less than a user defined constant (McDaniel and Devine, 1977, Mercier et al., 2005). In the present implementation the algorithm terminates when no more cuts have to be generated and enough savings have been achieved.

An optimality cut for a specific fleet f is generated when the ratio of the difference between the upper and lower bound of the corresponding CP problem to the lower bound of the Benders master problem is more than h^f, a user defined constant:

$$\frac{z\left(\bar{w}^i_{p_f}\right) - z_f}{\max_i z\left(\bar{v}^i_r\right)} > h^f, \tag{12}$$

where $z\left(\bar{w}^i_{p_f}\right)$ and $z\left(\bar{v}^i_r\right)$ are respectively the objective values of CP problem (9) and of the full problem (11) on the i-th iteration.

Furthermore, one can have a rough estimate of savings generated due to the integration, and terminate the Benders algorithm when there are relatively small potential savings through any future iterations. The estimate is made by observing that the solution of the Benders master problem on the first iteration corresponds to that of a sequential method, due to the absence of any cuts. In that sequential method the integrated FA with MR problem is followed by the solution of each CP with short-connections problem. The potential savings that can be generated through the rest of the iterations are given by the sum of the differences between the upper and lower bounds of the CP problems of each fleet. The condition to continue the Benders iterations in the present implementation is:

$$\frac{\sum_{f\in F}\left[z\left(\bar{w}^i_{p_f}\right) - z_f\right]}{z^{\text{Seq}} - z\left(\bar{v}^1_r\right)} > h, \tag{13}$$

where $z^{\text{Seq}} = \sum_{f\in F}\sum_{r\in R^f} c'_r\bar{v}^1_r + \sum_{f\in F}\sum_{p\in P^f} c_p\bar{w}^1_p$ is the integrated cost of the first iteration obtained by objective (7a), and h is a user defined constant.

4 Accelerating the Benders algorithm

Benders (1962) showed that after a finite number of steps his algorithm finds an optimal solution or proves that none exists. Finding a solution in a finite number of steps is in practice not good enough, and hence performance issues of the Benders algorithm are addressed in this section. On this topic Magnanti and Wong (1981) comment that there is a major computational bottleneck in the Benders algorithm:

<1> The Benders master problem has to be repeatedly solved, and if it is an IP problem its solution is usually slow.

As Magnanti and Wong (1981) comment, an increased number of iterations can usually be attributed to the following:

<2> The algorithm starts with no initial information.

<3> Towards the end of the algorithm the values of $(\bar{\mathbf{x}}, \bar{\mathbf{s}})$ tend to be similar to those of the previous iteration, resulting in similar cuts generated.

<4> A degenerate Benders subproblem has a non-unique dual solution, and the chosen one can correspond to a weak cut.

<5> The problem formulation might be "improper".

Earlier, another issue was encountered:

<6> Only the LP relaxation of the Benders subproblem is solved.

There are several solutions to the above problems. For instance concerning issues <1> and <6>, one can use the three-phase algorithm proposed by McDaniel and Devine (1977) presented here in Subsection 4.3. All these issues are going to be discussed in this section, with the exception of Issue <5> which is addressed in Subsection 5.2, where alternative formulations are attempted.

4.1 Pareto-optimal cuts

The Benders subproblem (9) has a set partitioning structure, making it degenerate and giving rise to Issue <4>. Magnanti and Wong (1981) solved this problem by choosing the strongest cut in the sense of the Pareto-optimality. To define this, observe that in equations (11e) and (11f), a cut is defined by the dual values $(\boldsymbol{\alpha}, \boldsymbol{\varsigma})$, thus given two of them $(\boldsymbol{\alpha}^1, \boldsymbol{\varsigma}^1)$ and $(\boldsymbol{\alpha}^2, \boldsymbol{\varsigma}^2)$, it is said that the first *dominates* the second if:

$$\sum_{l \in L} \alpha_l^1 x_{fl} + \sum_{ij \in S^f} \varsigma^{1,ij} s_f^{ij} \geq \sum_{l \in L} \alpha_l^2 x_{fl} + \sum_{ij \in S^f} \varsigma^{2,ij} s_f^{ij}, \quad \forall (\mathbf{x}, \mathbf{s}) \in \left(\hat{X}, \hat{S}\right), \tag{14}$$

with strict inequality for at least one point $(\mathbf{x}, \mathbf{s}) \in \left(\hat{X}, \hat{S}\right)$, where sets $\hat{X}$ and $\hat{S}$ are made of points satisfying equations (11b)–(11h). In definition (14) a simplified notation was used instead of this provided in equations (11e) and (11f), obtained with the assistance of equations (2b) and (4). A cut is said likewise to dominate another if the same dominance relation holds for their corresponding dual values. A dual value and its corresponding cut are said to be *Pareto-optimal* if no other dominates them.

Magnanti and Wong (1981) present a theorem which assists in computing a Pareto-optimal cut by employing the notion of core points. A point $x \in X$ is a *core point* if $x \in ri\,(X^c)$, where $ri\,(X)$ and X^c are respectively the relative interior and the convex hull of set X. The Pareto-optimal cut generation problem is similar to the dual of the Benders subproblem. However being in a column generation environment it makes more sense to solve the dual of Magnanti and Wong's problem, which is what Cordeau et al. (2001a) and Mercier et al. (2005) did for the integrated `MR` with `CP` problem. In that case and for the present Benders subproblem, the Pareto-optimal cut generation problem for fleet f is:

$$\min \sum_{p \in P^f} c_p w_p + z\left(\bar{w}_p\right)\zeta, \tag{15a}$$

subject to

$$\sum_{p \in P^f} a_{lp} w_p + \bar{x}_{fl}\zeta = x^0_{fl}, \quad \forall l \in L, \tag{15b}$$

$$\sum_{p \in P^f} s^{ij}_p w_p + \bar{s}^{ij}_f \zeta \leq s^{0,ij}_f, \quad \forall ij \in S^f, \tag{15c}$$

$$w_p \geq 0, \quad \forall p \in P^f, \tag{15d}$$

where $(\mathbf{x}^0, \mathbf{s}^0)$ is a core point and $(\bar{\mathbf{x}}, \bar{\mathbf{s}})$ corresponds to the solution $\bar{v}_r$ of the Benders master problem. $\bar{w}_p$ and $z\left(\bar{w}_p\right)$ are the optimal solution and optimal value of the Benders subproblem (9) for fleet f using the same $(\bar{\mathbf{x}}, \bar{\mathbf{s}})$.

Limitations for airline problems

Benders subproblem (9) is solved with column generation, and as discussed in Subsection 2.5, for tractability reasons the algorithm usually terminates with a near-optimal solution. This suboptimal solution can potentially make the numerical solution of problem (15) unbounded. The following example, introduced in this paper, illustrates where this numerical unboundedness stems from.

Example 4.1 Assume that a suboptimal solution $\bar{w}_p$ is found on the Benders subproblem (9), thus there must be another solution w'_p such that $z\left(w'_p\right) < z\left(\bar{w}_p\right)$. Define a third solution $w_p = -\zeta w'_p$, and divide equations (15b) and (15c) by a negative ζ. For very large negative ζ, the right hand side of equality (15b) and the inverted inequality (15c) will be numerically null. Since w'_p is a solution of Benders subproblem (9) the left side of (15b) and (15c) will be numerically null too, hence the chosen w_p is a numerically satisfactory solution. In this case, however, the objective (15a) becomes $\left[-\sum_{p\in P^f} c_p w'_p + z(\bar{w}_p)\right]\zeta = \left[-z(w'_p) + z(\bar{w}_p)\right]\zeta$ and because $z(w'_p) < z(\bar{w}_p)$ the objective's value and sign depend on ζ. But ζ can be an arbitrary large negative number, and since this is a minimization problem it is numerically unbounded.

This shortcoming was encountered in the performed experiments and could well plague the solution of other airline models when Benders decomposition with Pareto-optimal cuts is applied.

Overcoming the limitations

The previously mentioned problem is resolved using a theorem introduced by the author (Papadakos, 2006), which specializes in the subsequent form for the present problem.

Theorem 4.1 *Let* $\left(\mathbf{x}^0, \mathbf{s}^0\right) \in \left(\hat{X}, \hat{S}\right)$ *be a core point, then the optimal solution* w_p^0 *of the following problem:*

$$\min \sum_{p\in P^f} c_p w_p, \tag{16a}$$

subject to

$$\sum_{p\in P^f} a_{lp} w_p = x^0_{fl}, \quad \forall l \in L, \tag{16b}$$

$$\sum_{p\in P^f} s_p^{ij} w_p \leq s_f^{0,ij}, \quad \forall ij \in S^f, \tag{16c}$$

$$w_p \geq 0, \quad \forall p \in P^f, \tag{16d}$$

corresponds to a dual solution $(\boldsymbol{\alpha}^0, \boldsymbol{\varsigma}^0)$ *which is is Pareto-optimal.*

The proof of the above theorem concerns the dual of problem (16) and is rather trivial as it follows the steps of Magnanti and Wong (1981) by using any Benders subproblem solution and not necessarily the optimal (Papadakos, 2006).

It is obvious that the above model will not have the numerical unboundedness that (15) had. Hence, the Lasdon bound, discussed in Subsection 2.5, can be used when solving the Benders subproblems and terminate with a near-optimal solution avoiding the tailing-off effect and without facing any numerical unboundedness. In fact the Benders subproblem does not even have to be solved to generate a cut.

Notice that (16) is identical to Benders subproblem (9), with $(\bar{\mathbf{x}}, \bar{\mathbf{s}})$ substituted by $(\mathbf{x}^0, \mathbf{s}^0)$. Additionally problem (16) deals with Issue <3> as the generated cut depends on the core point $(\mathbf{x}^0, \mathbf{s}^0)$ and not on the solution $(\bar{\mathbf{x}}, \bar{\mathbf{s}})$ of the master problem. In practice however, there are no generic methods available for computing core points, and instead one has to rely on approximations (Mercier et al., 2005). The approximation chosen in this paper is computed by:

$$x^0_{fl} \leftarrow g \cdot x^0_{fl} + (1-g) \cdot \bar{x}_{fl}, \tag{17a}$$

$$s^{0,ij}_f \leftarrow \min\left\{ g \cdot s^{0,ij}_f + (1-g) \cdot \bar{s}^{ij}_f, \min\{x_{fi}, x_{fj}\} \right\}, \tag{17b}$$

as it gives a point between two solutions: $(\mathbf{x}^0, \mathbf{s}^0)$ and $(\bar{\mathbf{x}}, \bar{\mathbf{s}})$, where an an interior point is most likely to be found. Experiments show that $g = 1/2$ is the best choice. Approximations (17) are only partially spoiling the resolution of Issue <3> as the the Pareto-optimal cut problem is not entirely formulated by the latest solution but also depends on previous ones.

4.2 Pareto-optimal cuts independent of core points

As Sandhu and Klabjan (2004) comment, there is often no available method to estimate core points, and whenever approximation schemes are employed the strength of the corresponding cuts is compromised. For this reason one can take advantage of the structure of the problem, and instead of a core point employ a given point to generate Pareto-optimal cuts.

This structure concerns the `FA` part of the integrated model. Thus the result is initially only related to the instance of the present problem where all short-connection constraints and variables are eliminated. For the full problem with short-connections one has still to rely on approximations for variables $\mathbf{s}$ in the generated cuts, as discussed at the end of the present subsection.

The Benders master problem (11) without short-connections has the following structure:

$$\min \sum_{f\in F} \sum_{r\in R^f} c'_r v_r + \sum_{f\in F} z_f, \tag{18a}$$

subject to

$$\sum_{f\in F} x_{fl} = b_l, \quad \forall l \in L, \tag{18b}$$

$$x_{fl} - \sum_{r\in R^f} e_{lr} v_r = 0, \quad \forall l \in L, f \in F, \tag{18c}$$

$$q_m - q_{m^-} + \sum_{r\in R^f} (e^+_{mr} - e^-_{mr}) v_r = 0, \quad \forall m \in M^f, \forall f \in F, \tag{18d}$$

$$\sum_{m\in M'^f} q_m + \sum_{r\in R^f} e'_r v_r \le n_f, \quad \forall f \in F, \tag{18e}$$

$$-z_f + \sum_{l\in L} \alpha_l x_{fl} \le 0, \quad \forall \boldsymbol{\alpha} \in \Pi^f_{\text{points}}, \forall f \in F, \tag{18f}$$

$$\sum_{l\in L} \alpha_l x_{fl} \le 0, \quad \forall \boldsymbol{\alpha} \in \Pi^f_{\text{rays}}, \forall f \in F, \tag{18g}$$

$$v_r \in \{0,1\}, \quad \forall r \in R, \tag{18h}$$
$$q_m \geq 0, \quad \forall m \in M, \tag{18i}$$

where the x_{fl} variables were derived with the assistance of equation (2b), and the right-hand side of equation (18b) is generalizing that of equation (11b) by substituting 1 with $b_l = 1, \forall l \in L$. For this generic model a Pareto-optimal cut generation problem dependent only on the given b_l can be devised as proven by the next theorem.

Theorem 4.2 *The optimal solution w_p^0 of the following problem:*

$$\min \sum_{p \in P^f} c_p w_p, \tag{19a}$$

subject to

$$\sum_{p \in P^f} a_{lp} w_p = b_l, \quad \forall l \in L, \tag{19b}$$
$$w_p \geq 0, \quad \forall p \in P^f, \tag{19c}$$

corresponds to a dual solution $\boldsymbol{\alpha}^0$ which is is Pareto-optimal.

Proof. The dual of the problem (19) is:

$$\max \sum_{l \in L} b_l \alpha_l, \tag{20a}$$

subject to

$$\boldsymbol{\alpha} \in \Pi^f, \tag{20b}$$

Assume that $\boldsymbol{\alpha}^0$ is not Pareto-optimal, then there must be a cut (18f) or (18g) given by the dual value $\boldsymbol{\alpha}^d$ that dominates $\boldsymbol{\alpha}^0$:

$$\sum_{l \in L} \alpha_l^d x_{fl} \geq \sum_{l \in L} \alpha_l^0 x_{fl}, \quad \forall \mathbf{x} \in \hat{X}, \tag{21}$$

and there is at least one point $\bar{\mathbf{x}} \in \hat{X}$ such that:

$$\sum_{l \in L} \alpha_l^d \bar{x}_{fl} > \sum_{l \in L} \alpha_l^0 \bar{x}_{fl}. \tag{22}$$

Additionally because $\boldsymbol{\alpha}^d$ gives a cut (18f) or (18g), $\boldsymbol{\alpha}^d \in \Pi^f$ holds and hence it is a solution of (20), and since $\boldsymbol{\alpha}^0$ is the optimal one for (20):

$$\sum_{l \in L} b_l \alpha_l^0 \geq \sum_{l \in L} b_l \alpha_l^d. \tag{23}$$

Furthermore $\bar{\mathbf{x}} \in \hat{X}$, and it therefore satisfies equation (18b), which for $\mathbf{x} = \bar{\mathbf{x}}$ becomes:

$$\bar{x}_{fl} = b_l - \sum_{\tilde{f} \in F, \tilde{f} \neq f} \bar{x}_{\tilde{f}l}, \quad \forall l \in L. \tag{24}$$

From equation (22) $\bar{\mathbf{x}}$ obviously satisfies equation (21). Thus with $\mathbf{x} = \bar{\mathbf{x}}$ and by summing all equations (21) except for $\tilde{f} = f$, multiplying by -1, which reverses the inequality, and adding (23), one obtains:

$$\sum_{l \in L} b_l \alpha_l^d - \sum_{\tilde{f} \in F, \tilde{f} \neq f} \sum_{l \in L} \alpha_l^d \bar{x}_{\tilde{f}l} \leq \sum_{l \in L} b_l \alpha_l^0 - \sum_{\tilde{f} \in F, \tilde{f} \neq f} \sum_{l \in L} \alpha_l^0 \bar{x}_{\tilde{f}l}, \tag{25}$$

which with the assistance of equation (24) is obviously written as:

$$\sum_{l \in L} \alpha_l^d \bar{x}_{fl} \leq \sum_{l \in L} \alpha_l^0 \bar{x}_{fl}, \tag{26}$$

contradicting equation (22) and demonstrating that the assumption that $\boldsymbol{\alpha}^0$ is not Pareto-optimal is untenable. ■

Problem (20) tackles Issue <2> as one can generate some initial Pareto-optimal cuts before starting the solution of the Benders algorithm. A similar theorem is proved for a somewhat more general model than (18) by Papadakos (2006). Any models of this generic class, involve general assignments to subproblems. This could be particularly useful for airline operations research problems involving `FA` or any similar structure. In such a case Benders decomposition can become more attractive even when no method is available to compute core points. The Pareto-optimal cut is then computed by employing the point b_l given in constraint (18b) of the problem.

Notice that in the case of the integrated model without short-connections $x^0_{fl} = b_l = 1, \forall l \in L$. This $\mathbf{x}^0$ is not a core point, as even if $\mathbf{x}^0 \in \hat{X}$ it will be at the edge of $\left(\{0,1\}^{|L|}\right)^c$ and not in the relative interior. This choice is not even a solution for realistic `FA` problems as the available aircraft of each fleet are typically not enough to fly all legs.

The cut described by Theorem 4.2 is not Pareto-optimal for problem (11), since the short-connection variables $\mathbf{s}$ are not involved. For these variables one has still to rely on approximations while, as instructed by Theorem 4.2, keeping $x^0_{fl} = 1, \forall l \in L$ for each fleet's f subproblem. Experiments show that the approximation $\mathbf{s}^0 = \mathbf{1}$ outperforms that with $\mathbf{s}^0 = \mathbf{0}$. This result is very similar to the one experienced by Mercier et al. (2005) for the combined `MR` and `CP` problem.

4.3 The accelerated algorithm

The methods discussed so far can be used to formulate the extension of the three-phase Benders algorithm of McDaniel and Devine (1977), which includes

Pareto-optimal cuts (Cordeau et al., 2001a, Mercier et al., 2005). Willing to accommodate the *initial cuts* derived with the help of Theorem 4.2 one has to modify the execution order of the Benders master problem, subproblem, and Pareto-optimal cut problem. In this sense the algorithm is:

Initialization: set $\mathbf{x}^0 = \mathbf{1}$ and $\mathbf{s}^0 = \mathbf{1}$; LP relax both Benders problems

Phase 1: generate a Pareto-optimal cut (16) for the given $(\mathbf{x}^0, \mathbf{s}^0)$; solve the Benders master problem and then the subproblem

- **if (**all subproblems are feasible **and** criterion (13), $h = 1/8$, is not met**)** **or** no new cuts can be generated due to violation of criterion (12), $h^f = 0.01/100$
- **then** introduce IP constraints on Benders master problem; **goto Phase 2**
- **else** use (17) to compute $(\mathbf{x}^0, \mathbf{s}^0)$; **goto Phase 1**

Phase 2: generate a Pareto-optimal cut (16) for the given $(\mathbf{x}^0, \mathbf{s}^0)$; solve the Benders master problem and then the subproblem

- **if (**all subproblems are feasible **and** criterion (13), $h = 1/2$, is not met**)** **or** no new cuts can be generated due to violation of criterion (12), $h^f = 0.1/100$
- **then** introduce IP constraints on Benders subproblem; **goto Phase 3**
- **else** use (17) to compute $(\mathbf{x}^0, \mathbf{s}^0)$; **goto Phase 2**

Phase 3: solve the IP Benders subproblem

- **if** all subproblems are feasible
- **then** terminate the algorithm with IP solutions for both problems
- **else** add a Pareto-optimal cut (11f),† remove IP constraints from Benders subproblem; **goto Phase 2**

The success of the three-phase algorithm lies in the fact that, hopefully, good enough cuts will be generated mostly on Phase 1, where the easy LP relaxation is solved on the Benders master problem. Then, once Phases 2 and 3 are solved with IP constraints reintroduced, not that many iterations will be needed.

In summary issues <1> and <6> are tackled by the three-phase algorithm, and Issue <4> is resolved by the Pareto-optimal cut problem (16). Problem (16) because of approximations (17) only partially handles Issue <3>. Finally, Issue <2> is only partially dealt with by problem (20) since approximations are used for the short-connection variables.

†The interested reader is referred to Chu and Xia (2005) for a more elaborate method to generate strong cuts in Phase 3. Their method is out of this paper's scope, as in the present experiments, Phase 3 is only solved once without generating any cuts.

5 Antagonistic methods and formulations

This section discusses airline scheduling methods and formulations to be compared with the integrated approach provided in Sections 3 and 4.

5.1 Best known methods from the literature

The best method available in the literature to solve the airline scheduling problem is a combination of integrated models executed in a sequential manner. This method consists of initially solving the integrated `FA` with `MR` (Barnhart et al., 1998a) and feeding the acquired solution of each fleet into the integrated `MR` with `CP` problem (Mercier et al., 2005); this method will be referred to hereafter as *semi-integrated*. Both of the integrated models included in this method are special cases of the integrated airline scheduling model (7). For this reason instead of the Barnhart et al. model, the Benders master problem (11) with empty Π^f_{points} and Π^f_{rays} sets is solved using branch-and-price. This does not have to be implemented anew since it is the solution of the Benders master problem on the first iteration if one starts directly from Phase 2. Furthermore, in order to assist the solution of this model towards low `CP` costs, the crew costs per flying-time for each fleet f and each leg l are temporarily added to the c_{fl} costs of `FA`. Of course these costs are removed before feeding the solution to the combined `MR` with `CP` problem. Finally, the integrated `MR` with `CP` problem, including Pareto-optimal cuts is nothing more than model (7) for a single fleet.

To further compare the integrated method savings with the savings coming from the semi-integrated method, the next-best method has to be used too. In this instance the results of the integrated `FA` with `MR` problem for each fleet are fed to the `CP` with short-connections problem (3); this method is referred hereafter as *sequential*. This `CP` model is solved with branch-and-price, and it is once more a special case of model (7) where Phase 3 is solved only.

5.2 Alternative integrated formulations

A summary is presented here for alternative integrated airline scheduling formulations attempted by the author (Papadakos, 2006). The study of these alternatives is related with Issue <5> discussed in Section 4, as different formulations could well result in more efficient solutions. The most direct reformulation stems from the results reported by Mercier et al. (2005), for their integrated `MR` with `CP` model. In the aforementioned paper two different decompositions were presented. In the first one the `MR` problem was on the Benders master problem and the `CP` on the subproblem, and in the second one these were inverted. The experimental results they provided gave evidence that the second formulation is more efficient.

Tempted by the successful results of Mercier et al. (2005), a similar reformulation is devised for the integrated model (7), by replacing constraint (7b) with:

$$\sum_{f\in F}\sum_{p\in P^f} a_{lp}w_p = 1, \quad \forall l \in L, \tag{7b'}$$

stemming from the substitution of equation (3b) in (1b). Constraint (7b′) is used in the Benders master problem to assign one pairing per leg. One can then formulate the Benders subproblem by constraints (7c)–(7h), and the master problem with the rest. This way, on the Benders master problem one decides which pairings are beneficial for each fleet, and on the subproblem the feasibility of MR routing is verified. Since no through revenue is considered in the experiments of the next section, c_r is null and no optimality cuts have to be generated. This decomposition, however, proves rather inefficient as on the Benders master problem the CP solution is simultaneously assigning legs to each fleet without having enough information concerning the available aircraft (Papadakos, 2006).

For this reason the previously mentioned model, was extended to include FA constraints on the Benders master problem. To be more specific, these constraints were those of the Hane et al. (1995) model, concerning the flow conservation in the time-line and the usage of the available aircraft. Both of these constraints rely on FA variables x_{fl} discussed in Subsection 2.1 and are easily amalgamated in the Benders master problem through equation (2b) (Papadakos, 2006). Another attempt concerned further extending the previous Benders mater problem by including plane-count constraints (Papadakos, 2006) resulting in a model similar to that of Sandhu and Klabjan (2004).

6 Computational experiments

The algorithm presented in Subsection 4.3 was implemented in order to evaluate the benefits of the integrated methodology. Thus, the dominance-relaxed constrained shortest path algorithms were implemented in C++ and compiled using gcc 3.4.3. The rest of the algorithms were implemented in ECL^{i}PS^e version 5.8 (Cheadle et al., 2003). ECL^{i}PS^e is a constraint logic programming language of Cisco Technology Inc., and includes a branch-and-price-and-cut library that can use different linear solvers (Eremin, 2003); in the present paper ILOG CPLEX 9.030 (ILOG S.A., 2004) was chosen. The experiments were performed on a single computer having a single-core 64-bit AMD Athlon™ processor at 2.4 GHz, with 2 GB of RAM memory, and running the 64-bit Linux kernel version 2.6.11. Although there was a 64-bit processor involved, both ECL^{i}PS^e and the C++ code were compiled in the 32-bit mode.

6.1 The data sets

The data sets used for the experiments of this paper were derived from those provided by a major European and a major North American airline. The schedule of the former is medium-haul with a single central hub, while the schedule of the latter is long-haul with a hub-and-spoke structure. *Hub-and-spoke* networks are characterized by high activity in hub stations, which are connected with each other as well as with spoke stations with lower activity. In the European instance, almost all the legs depart from or arrive at that central hub. The European airline is scheduling in total 372 legs per day, and the American over 2,100. In both cases 6 fleets are considered, and although this is accurate in the case of the European airline it is not for the American. In the latter case different fleets were merged into fleet groups having similar cost characteristics and ready-times.

To enable examination of the introduced algorithms' scalability, reduced data sets had to be generated by eliminating some of the present stations. Moreover, the number of aircraft for each fleet was kept in proportion to the number of legs. The various instances considered in this paper are shown in Table 4, where for each *Instance*: *All, Maintenance,* and *Crew* represent respectively the number of: all stations, stations with maintenance facilities, and crew bases. *Aircraft* and *Short-connections* give respectively the aggregate number of available aircraft and all possible short-connections for all fleets.

Table 4: Characteristics of the data sets.

Instance	Stations				Short-
(network \ legs)	All	Maintenance	Crew	Aircraft	connections
c\214	28	1	1	48	1,709
c\280	40	1	1	62	2,768
c\372	55	1	1	77	4,650
hs\196	25	2	2	49	1,583
hs\346	38	3	4	86	3,825
hs\506	54	5	4	120	7,832
hs\705	62	6	5	167	13,486

Note: In all cases 6 fleets were accounted, *legs* is the total number of scheduled scheduled ones, and the *network* structure is either with one central hub (*c*) or hub-and-spoke (*hs*).

Concerning the FA, ready-times were provided for all legs and all possible fleets they could be assigned to. Additionally, there was information supplied on the passenger demand and spilling costs depending on the aircraft used, as well as the rest of the operating and maintenance costs involved. No through flight's revenue was available and for this reason routing costs c_r were considered null, while maintenance costs were accounted on average in c_{fl} constants. Aircraft were allowed to fly for 4 days maximum before returning to a maintenance station to spend 8 hours. The scheduling horizon was daily, which is accurate for the European airline, however, the American's weekly schedule had to be adapted.

Regarding the crew data, no exact information was provided, thus labor regulations were obtained from Ho et al. (1999) and adapted for the medium- and long-haul case. The most important deviation from Ho et al. concerns the time away from base, which was set for the European airline to 4 days, and the for American to 5 days. Moreover, the minimum sit-time was considered 30 minutes. Wages were inferred from the average wages per person and per year of the airline that provided the data. Overnight costs were inferred from typical hotel prices of the cities the airline was flying to. Deadheading was not considered since such information was not available.

Finally concerning the crew cost function, one should first remember that the CP constraints are quite complex. Therefore, in order to reduce labels used in the dominance relation, the cost was sometimes considered as a function of the time away from base alone (Barnhart and Shenoi, 1998, Cordeau et al., 2001b, Mercier et al., 2005, Mercier and Soumis, 2006). This is equivalent to operating only in Mode I of the dominance-relaxed constrained shortest path algorithm introduced in Subsection 2.5. However, desiring to have an accurate account of crew costs they were considered to be a function of all labels, and in the current implementation Mode III was only excluded.

6.2 Experimental results

Integrated method

In Table 5 the experimental results of the integrated model, for the instances discussed in the previous subsection are demonstrated. This table provides statistics of the *Overall* algorithm execution as well as for each *Phase* individually. *CPU MP*, *CPU SP*, and *CPU* are respectively: the CPU run-times of the Benders master problem, the Benders subproblem and the sum of run-times of both problems. Additionally *Iterations*, *Optimality cuts*, and *Feasibility cuts* are respectively the number of Benders iterations, and the number of optimality and feasibility cuts generated. There were no feasibility cuts on Phase 2, however the feasibility cuts generated in Phase 1 imply that there is always a possibility of infeasible CP problems that any sequential method cannot handle. Furthermore, Phase 3 was executed once and no cuts had to be generated. Finally ${}_{max}gap$ % is the maximum optimality gap due to early termination criteria (12) and (13) as well as due to the termination of the IP search on the first encountered solution. The maximum optimality gap is computed by $\left(c^{\mathrm{IP}} - LB^{\mathrm{LP}}\right)\Big/ LB^{\mathrm{LP}} \times 100$, where $LB^{\mathrm{LP}} = \max_i \left(\bar{v}_r^i\right)$ and c^{IP} are respectively the lower bound of the LP relaxation at the end of Phase 1, and the cost upon termination of the algorithm.

Regarding the results of Table 5 it should be commented that the termination parameters of criteria (12) and (13) were kept the same across all instances, and the IP algorithm was always terminating when the first solution was encountered. These particular choices were the most efficient for the largest data set.

Table 5: Results of the integrated method.

	c214	c280	c372	hs196	hs346	hs506	hs705
Phase 1							
CPU MP	0.18	0.34	0.48	0.20	0.66	2.41	9.8
CPU SP	0.09	0.21	0.50	0.22	0.98	3.89	10.7
CPU	0.27	0.55	0.98	0.42	1.64	6.30	20.5
Iterations	8	9	8	15	16	18	10
Optimality cuts	29	35	34	73	75	91	43
Feasibility cuts	0	0	0	0	0	0	2
Phase 2							
CPU MP	0.07	0.34	1.20	0.17	1.91	3.22	6.04
CPU SP	0.00	0.00	0.05	0.01	0.14	0.22	0.02
CPU	0.07	0.34	1.25	0.18	2.05	3.44	6.06
Iterations	1	1	5	1	3	2	1
Optimality cuts	0	0	5	0	9	4	0
Phase 3							
CPU	0.00	0.00	0.01	0.00	0.02	0.13	1.21
Overall							
CPU	0.35	0.89	2.25	0.60	3.71	9.87	27.8
$_{max}$gap %	1.10	0.79	0.21	1.41	1.06	0.83	0.73

Note: All *CPU* times are given in hours.

Since, however, schedules with different structure and number of legs concern different airlines, it generally makes more sense to adapt these parameters or to have a more thorough IP search, leading to smaller $_{max}gap$ %.

Performance summary of introduced acceleration methods

Concerning the heuristic steepest-edge pricing introduced in Subsection 2.5, in the experiments performed it was at least 1.5 times faster than Dantzig pricing (Papadakos, 2006). This was even more dramatic for the biggest instances, where, for example, in the case of *hs705* steepest-edge pricing was 4.7 times faster. This was achieved by reducing the number of column generation iterations and improving the quality of generated columns.

Experiments were performed without the initial cuts of Theorem 4.2, providing evidence that even with approximations (17), the initial cuts are especially beneficial, reducing the number of Benders iterations and cuts by at least 66%, while being at least 2 times faster (Papadakos, 2006). Finally, experiments were conducted with simple Benders cuts, proving that Pareto-optimal cuts speed-up the solution. (Papadakos, 2006).

Best known methods from the literature

In Table 6 the experimental results of the semi-integrated and sequential methods discussed in Subsection 5.1 are demonstrated. *Overall CPU* is the total

Table 6: Results of the semi-integrated and sequential methods.

	c214	c280	c372	hs196	hs346	hs506	hs705
FA + MR							
CPU LP	0.07	0.10	0.20	0.05	0.18	0.56	1.33
CPU IP	0.10	0.09	0.57	0.06	0.60	1.76	4.18
CPU	0.17	0.19	0.77	0.11	0.42	1.20	2.85
Gap %	2.07	1.05	0.22	0.09	0.54	0.22	0.11
MR + CP							
CPU	0.01	0.06	0.13	0.04	0.32	0.66	3.62
Gap %	-0.25	-0.31	-0.20	-0.35	-0.26	-0.48	-0.30
(combined with FA + MR is semi-integrated)							
Overall CPU	0.18	0.25	0.90	0.15	0.74	1.86	6.47
CP							
CPU	0.01	0.04	0.10	0.02	0.07	0.38	2.17
Gap %	-0.11	-0.06	-0.21	0.12	0.13	0.16	0.03
(combined with FA + MR is sequential)							
Overall CPU	0.18	0.23	0.87	0.13	0.49	1.58	5.02

Note: All *CPU* times are given in hours.

execution-time of the previously described sequential methods. Finally, *Gap %* is the optimality gap computed by $\left(c^{\mathrm{IP}} - c^{\mathrm{LP}}\right) / c^{\mathrm{LP}} \times 100$, where c^{LP} and c^{IP} are respectively: the cost of the LP relaxation and the cost of the full IP problem. The negative *Gap %* found when solving the IP CP problem should be attributed to the fact that Mode III was finally excluded in the dominance-relaxed constrained shortest path algorithm, as explained at the end of Subsection 6.1. Thus, two labels were excluded from the dominance relation, leading to suboptimal LP solutions. Thus, after the LP relaxation is solved in the root node, the branching choices perturb the problem and drive it to better solutions.

Willing to estimate the quality sacrificed due to label exclusion in the dominance relation, the IP CP problem was solved including Mode III, for all instances except for *hs705* which was extremely slow. In these cases the gap between the sequential solutions including Mode III and those excluding Mode III did not exceed 0.16% which is relatively small. As a conclusion concerning these approximations in cases where Mode III can be efficiently solved one could only include it in crucial stages of the algorithm like for instance in Phase 3 where the final IP CP is solved.

Comparison with best known methods

Provided with the results of all these methods, one can compare them by the cost due to one and due to the other and estimate the cost savings in each case. The first such comparison is between the integrated method and the best one available from the literature, the semi-integrated. The second comparison is between the semi-integrated and the next-best, the sequential. Finally, it is tempting to contrast the cost savings due to the first comparison and the savings due to the second. This contrast is illustrated by the *Savings ratio* which is computed by $Savings^{\text{Integrated}} / Savings^{\text{Semi-integrated}}$, where $Savings^{\text{Integrated}}$ and $Savings^{\text{Semi-integrated}}$ are the savings due to the first and the second comparison respectively. All these are presented in Table 7, where *Savings %* and *CPU ratio* are respectively computed by $\left(c^2 - c^1\right) / c^2 \times 100$ and $\text{CPU}^1/\text{CPU}^2$, when comparing method 1 versus method 2. In this case c^i and CPU^i are the costs and CPU time of method i. Since a daily schedule was solved, *Savings/Year* is the projection of the savings in a year's time. *Savings/$_{max}$gap* is given by $\left(c^{IP}_{\text{Integrated}} - c^{IP}_{\text{Semi-integrated}}\right) / \left(c^{\text{IP}}_{\text{Integrated}} - LB^{\text{LP}}_{\text{Integrated}}\right)$.

Table 7: Comparison of the integrated, semi-integrated and sequential methods.

	c214	c280	c372	hs196	hs346	hs506	hs705
Integrated versus Semi-integrated							
Savings %	1.82	1.14	0.50	2.57	1.68	1.51	1.91
Savings/Year	7.08	6.02	3.53	9.10	10.8	14.6	24.2
CPU ratio	1.94	3.56	2.50	4.00	5.01	5.31	4.30
Savings/$_{max}$gap	1.66	1.44	2.45	1.82	1.59	1.80	2.63
Semi-integrated versus Sequential							
Savings %	0.03	0.10	0.08	0.31	0.26	0.47	0.22
Savings/Year	0.13	0.50	0.59	1.10	1.66	4.58	2.80
CPU ratio	1.00	1.09	1.03	1.15	1.51	1.18	1.29
(Integrated versus Semi-integrated) / (Semi-integrated versus Sequential)							
Savings ratio	56.8	12.0	5.95	8.30	6.53	3.19	8.66

Note: All *Savings/Year* are given in millions of US dollars.

From the results of Table 7 it is clear that significant savings can be achieved if one utilizes the integrated airline scheduling model. These savings are up to

24 million US dollars per year for the largest instance of 700 legs and 6 fleets. Although optimality was not reached, the ratio of savings to the maximum optimality gap is high enough to justify the heuristics and the early termination strategies used. This could be further improved for the smaller instances, by changing the parameters of criteria (12) and (13) and employing a more thorough IP search.

The significance of these savings can also be seen in contrast to those savings due to the semi-integrated. This contrast is illustrated by the *Savings ratio* of the former to the latter. Obviously, the integrated model is by far more successful in cutting-down airlines' costs. It is fair to say though that this is because the savings due to the present integration concern the overall costs of the airline schedule for different possible fleet assignments. In contrast, the other methods are working within those fleets without having the flexibility to consider different assignments. Additionally, the other methods concern CP alone, and crew costs are typically only a fraction of the overall costs; for the data used here this fraction was approximately a third, which is in line with the industry standards.

The advantage of the integrated methodology is significant cost reductions coming with the disadvantage of slower run-times. Although further experiments are required, the first impression that *CPU ratios* give, is that the present method follows the scalability of the semi-integrated model, which in its turn follows that of the sequential.

Comparison with alternative integrated formulations

Concerning the alternative integrated formulations, experiments show that Phase 1 of the three-phase algorithm needs too many iterations to generate MR feasibility cuts slowing-down the overall process. For this reason the algorithm for the alternative formulations begins instead directly from Phase 2. Pareto-optimal cuts, however, prove invaluable as the algorithm performs rather poorly in their absence (Papadakos, 2006). Additionally since Theorem 4.2 is used with approximation (17), in the experiments that were performed the option to generate initial cuts was either turned on, or off. The most efficient initial value for $\mathbf{s}^0$ proved to be $\mathbf{0}$, for similar reasons that this held for integrated CP with MR model of Mercier et al. (2005). Apart from these modifications the algorithm was identical to that of Subsection 4.3. Finally, the Benders master problem was solved using an algorithm similar to that presented in Subsection 3.2.

The computational results indicated that no alternative outperforms any other for all instances (Papadakos, 2006). For this reason in Table 8 the results of the best performing alternative, for each instance, are presented, and compared with the results of the original integrated model (7). *PCC* refers to the usage of plane-count constraints and *CPU ratio* is obtained by $\text{CPU}^{\text{alternative}}/\text{CPU}^{\text{integrated}}$. As before, Phase 3 was executed only once without generating any cuts, in all

Table 8: Results of the most efficient alternative integrated model and decomposition for each data set, and comparison with the original integrated model.

	c214	c280	c372	hs196	hs346	hs506	hs705
Best alternative							
PCC	no	no	yes	yes	no	n/a	n/a
Initial cuts	on	on	off	off	on	n/a	n/a
Phase 2							
CPU MP	0.37	0.81	15.9	2.44	3.65	> 21	> 28
CPU SP	0.03	0.02	0.37	0.13	0.59	> 1	> 4
CPU	0.40	0.83	16.3	2.57	4.24	> 24	> 36
Iterations	5	2	9	22	7	> 9	> 4
Cuts	5	1	11	56	9	> 19	> 11
Phase 3							
CPU	0.01	0.01	0.05	0.02	0.07	n/a	n/a
Overall							
CPU	0.41	0.84	16.3	2.72	4.31	> 24	> 36
$_{max}$gap %	1.05	0.95	1.33	2.72	2.41	n/a	n/a
Savings %	1.86	0.99	-0.61	1.31	0.37	n/a	n/a
Savings/$_{max}$gap	1.82	1.07	n/a	0.50	0.16	n/a	n/a
Savings/Year	7.25	5.22	-4.28	4.64	2.36	n/a	n/a
versus Original							
CPU ratio	1.17	0.94	7.25	4.53	1.16	> 2.4	> 1.3

Note: All *CPU* times are given in hours and all *Savings/Year* are given in millions of US dollars.

cases.

The results presented in Table 8 demonstrate that even the best alternative model is rather inefficient in comparison to the original one. Moreover the maximum optimality gap of the alternative models is large affecting the savings. This is illustrated by the low *Savings/*$_{max}$*gap* ratios and by the loss instead of savings for instance *c372*. These could be improved by a more thorough IP search, but since all iterations are in Phase 2, these could make the already poor solution-times even worse. Finally, no method was able to find a solution for instances *hs506* and *hs705* within a timeout period set to 24 and 36 hours respectively; however, more than 19 and 11 cuts were accordingly generated.

Notice additionally, that for all instances more than one Benders iteration and numerous cuts had to be generated before obtaining a feasible `MR` solution. This implies that the integrated models of the Benders master problem cannot generally give `MR` feasible solutions. These models were the simple `FA` time-line and its enhancement with plane-count constraints. Thus one way or another, it seems that the MR problem has to be solved, and the alternative models attempted offer at least such an opportunity.

Regarding the application of Theorem 4.2 the results of the alternative formulations indicate that the approximation used for $\mathbf{s}^0$ did not prove that efficient for the feasibility cuts utilized there (Papadakos, 2006).

6.3 Speed-up estimate of a straightforward parallelization

Since the main objective of airlines is to solve their problems as fast as possible, one could use parallelization to further speed-up the overall solution-time. Although elaborate techniques could possibly be devised, a straightforward technique is evaluated here. This parallelization is based on the observation that some problems are decomposed per fleet, and instead of solving them sequentially one could use a number of single-core CPU computers equal to the number of fleets and solve these subproblems in parallel. The problems that are fleet decomposed are: the column generation subproblem of the Benders master problem, the Benders subproblem, the Pareto-optimal cut problem, and the second stage of the IP algorithm presented in Subsection 3.2.

To have an estimate of these parallel run-times, each time a problem needs to be solved for more than one fleet, the fleet problem that took most time has to be accounted, as this would only affect the parallel's algorithm run-time. Such an estimate is presented in Table 9, where the *Speed-up %* is computed

Table 9: Estimate of solution-times for the straightforward parallelization of the integrated method.

	c214	c280	c372	hs196	hs346	hs506	hs705
CPU MP	0.19	0.51	1.23	0.29	1.97	4.23	12.5
CPU SP	0.03	0.07	0.24	0.06	0.39	1.23	3.9
CPU	0.22	0.58	1.47	0.35	2.36	5.46	16.5
Speed-up %	59	53	53	71	57	81	68

Note: *CPU* times are given in hours.

by $(\text{CPU}^{\text{sequential}} - \text{CPU}^{\text{parallel}})/\text{CPU}^{\text{parallel}} \times 100$. The above estimates are reasonably accurate as the information that needs to be exchanged between the computers is minimal, and hence the network speed will not significantly influence the run-times.

7 Conclusions and future work

In this paper, the previously thought intractable integrated airline scheduling problem was solved for realistic instances of European and North American airlines. These airlines had different network structures, allowing a wide evaluation. The largest solved instance scheduled 700 legs for 6 fleets, and succeeded in reducing overall operating costs by 24 million US dollars in comparison to the best known method from the literature. The solution-time of this instance can be 16.5 hours when using a straightforward parallelization method with 6 single-core-CPU computers. Alternative integrated models and decompositions were also attempted but did not prove to be as efficient as the original.

The success of the algorithm is due to introduced theorems for generating Pareto-optimal cuts in Benders decomposition, that could prove useful in other areas of airline operations research. Moreover special techniques to speed-up column generation were implemented, and could be applied in other airline problems too.

One could possibly try to extend the present work by introducing time-windows to departure-time of the legs. Such a flexibility within the airline scheduling model may well generate even greater cost savings.

Acknowledgments

The author is indebted to Dr. Jonathan Lever, Prof. Mark Wallace, Dr. Andy Eremin, Dr. Wilhelm Cronholm, Dr. Olli Kamarainen and Imogen Rivers for their support in different stages of the project.

References

Barnhart, C., Boland, N., Clarke, L., Johnson, E., Nemhauser, G., and Shenoi, R. (1998a). Flight string models for aircraft fleeting and routing. *Transportation Science*, 32(3):208–220.

Barnhart, C., Cohn, A., Johnson, E., Klabjan, D., Nemhauser, G., and P.Vance (2003). Airline crew scheduling. In Hall, R., editor, *Handbook of Transportation Science*, chapter 3, pages 57–93. Kluwer Scientific Publishers, Boston.

Barnhart, C., Johnson, E., Nemhauser, G., Savelsbergh, M., and Vance, P. (1998b). Branch-and-price: Column generation for solving huge integer programs. *Operations Research*, 46(3):316–329.

Barnhart, C., Kniker, C., and Lohatepanout, M. (2002). Itinerary-based airline fleet assignment. *Transportation Science*, 36(2):199–217.

Barnhart, C. and Shenoi, R. (1998). An approximate model and solution approach for the long-haul crew pairing problem. *Transportation Science*, 32(3):221–231.

Benders, J. (1962). Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(3):238–252.

Cheadle, A., Harvey, W., Sadler, A., Schimpf, J., Shen, K., and Wallace, M. (2003). ECL^{i}PS^e: An introduction. Technical Report IC-Parc-03-1, IC-Parc, Imperial College London.

Chu, Y. and Xia, Q. (2005). A hybrid algorithm for a class of resource constrained scheduling problems. In Barták, R. and Milano, M., editors, *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems: Second International Conference, CPAIOR 2005, Prague Czech Republic*, pages 110–124, Berlin Heidelberg. Springer-Verlag GmbH.

Chvátal, V. (1983). *Linear Programming*. W.H. Freeman and Co., New York.

Clarke, L., Hane, C., Johnson, E., and Nemhauser, G. (1996). Maintenance and crew considerations in fleet assignment. *Transportation Science*, 30(3):249–260.

Cohn, A. and Barnhart, C. (2003). Improving crew scheduling by incorporating key maintenance routing decisions. *Operations Research*, 51(3):387–396.

Cordeau, J.-F., Soumis, F., and Desrosiers, J. (2001a). Simultaneous assignment of locomotives and cars to passenger trains. *Operations Research*, 49(4):531–548.

Cordeau, J.-F., Stojković, G., Soumis, F., and Desrosiers, J. (2001b). Benders decomposition for simultaneous aircraft routing and crew scheduling. *Transportation Science*, 35(4):375–388.

Dantzig, G. and Wolfe, P. (1960). Decomposition principle for linear programs. *Operations Research*, 8(1):101–111.

Desaulniers, G., Desrosiers, J., Dumas, Y., Marc, S., Rioux, B., Solomon, M., and Soumis, F. (1997a). Crew pairing at Air France. *European Journal of Operational Research*, 97:245–259.

Desaulniers, G., Desrosiers, J., Dumas, Y., Solomon, M., and Soumis, F. (1997b). Daily aircraft routing and scheduling. *Management Science*, 43(6):841–855.

Desaulniers, G., Desrosiers, J., Gamache, M., and Soumis, F. (1998a). Crew scheduling in air transportation. In Crainic, T. G. and Laporte, G., editors, *Fleet Management and Logistics*, chapter 8, pages 169–185. Kluwer, Boston.

Desaulniers, G., Desrosiers, J., Ioachim, I., Solomon, M., and Soumis, F. (1998b). A unified framework for deterministic time constrained vehicle routing and crew scheduling problems. In Crainic, T. G. and Laporte, G., editors, *Fleet Management and Logistics*, chapter 3, pages 57–93. Kluwer, Boston.

Desrochers, M. (1986). *La Fabrication d'horaires de Travail Pour Les Conducteurs d'autobus Par Une Méthode de Génération de Colonnes*. PhD thesis, Université de Montréal, Centre de recherche sur les Transports, CP 6128, Succursale "Centreville", Montréal, Québec, H3C 3J7. Publication # 470.

Desrosiers, J., Dumas, Y., Solomon, M., and Soumis, F. (1995). Time constrained routing and scheduling. In Ball, M., Magnanti, T. L., Monma, C., and Nemhauser, G., editors, *Handbooks in Operations Research and Management Science*, volume 8: Network Routing, chapter 2, pages 35–139. North-Holland, Amsterdam.

du Merle, O., Villeneuve, D., and Desrosiers, J. (1999). Stabilized column generation. *Discrete Mathematics*, 194:229–237.

Eremin, A. (2003). *Using Dual Values to Integrate Row and Column Generation into Constraint Logic Programming*. PhD thesis, Imperial College London.

Goldfrab, D. and Reid, J. (1977). A practicable steepest-edge simplex algorithm. *Mathematical Programming*, 12:361–371.

Hane, C., Barnhart, C., Johnson, E., Marsten, R., Nemhauser, G. L., and Sigismondi, G. (1995). The fleet assignment problem: solving a large-scale integer program. *Mathematical Programming*, 70:211–232.

Ho, R., Johnson, E., and Shaw, T. (1999). Modeling tools for airline crew scheduling and fleet assignment problems. In Ciriani, T., Gliozzi, S., Johnson, E., and Tadei, R., editors, *Operational Research in Industry*, pages 1–24. MacMillan Press Ltd, London.

ILOG S.A. (2004). *ILOG Cplex 9: User's Manual and Reference Manual*.

Klabjan, D., Johnson, E., Nemhauser, G., Gelman, E., and Ramaswamy, S. (2002). Airline crew scheduling with time windows and plane-count constraints. *Transportation Science*, 36(3):337–348.

Lasdon, L. (1970). *Optimization Theory for Large Systems*. Macmillan, London.

Lohatepanont, M. and Barnhart, C. (2004). Airline schedule planning: Integrated models and algorithms for schedule design and fleet assignment. *Transportation Science*, 38(1):19–32.

Lübbecke, M. (2001). *Engine Scheduling by Column Generation*. PhD thesis, Technischen Universität Braunschweig.

Magnanti, T. and Wong, R. (1981). Accelerating Benders decomposition: Algorithmic enhancement and model selection criteria. *OR*, 29(3):464–484.

McDaniel, D. and Devine, M. (1977). A modified Benders' partitioning algorithm for mixed integer programming. *Management Science*, 24(2):312–319.

Mercier, A., Cordeau, J.-F., and Soumis, F. (2005). A computational study of Benders decomposition for the integrated aircraft routing and crew scheduling problem. *Computers and Operations Research*, 32(6):1451–1476.

Mercier, A. and Soumis, F. (2006). An integrated aircraft routing, crew scheduling and flight retiming model. *Computers and Operations Research*, In Press.

Papadakos, N. (2006). *Integrated Airline Scheduling: Models and Decomposition Techniques*. PhD thesis, Imperial College London.

Rexing, B., Barnhart, C., Kniker, T., Jarrah, A., and Krishnamurthy, N. (2000). Airline fleet assignment with time windows. *Transportation Science*, 34(1):1–20.

Ryan, D. and Foster, B. (1981). An integer programming approach to scheduling. In Wren, A., editor, *Computer Scheduling of Public Transport Urban Passenger Vehicle and Crew Scheduling*, pages 269–280. North-Holland, Amsterdam.

Sandhu, R. and Klabjan, D. (2004). Integrated airline planning. In *Proceedings 44th AGIFORS Annual Symposium*, Singapore, Singapore.

Sherali, H., Bish, E., and Zhu, X. (2006). Airline fleet assignment concepts, models, and algorithms. *European Journal of Operational Research*, 172(1):1–30.

Vance, P., Atamtürk, A., Barnhart, C., Gelman, E., Johnson, E., Krishna, A., Mahidhara, D., Nemhauser, G., and Rebello, R. (1997). A heuristic branch-and-price approach for the airline crew pairing problem. Technical Report LEC 97-06, School of Industrial and Systems Engineering, Georgia Institute of Technology, Georgia.

Vanderbeck, F. (1994). *Decomposition and Column Generation for Integer Programs*. PhD thesis, Université catholique de Louvain.