ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

ARCHIVIO ISTITUZIONALE
DELLA RICERCA

Alma Mater Studiorum Università di Bologna
Archivio istituzionale della ricerca

Optimal solutions to a real-world integrated airline scheduling problem

This is the final peer-reviewed author's accepted manuscript (postprint) of the following publication:

Published Version:
Cacchiani, V., Salazar-González, J. (2017). Optimal solutions to a real-world integrated airline scheduling problem. TRANSPORTATION SCIENCE, 51(1), 250-268 [10.1287/trsc.2015.0655].

Availability:
This version is available at: https://hdl.handle.net/11585/590475 since: 2021-11-11

Published:
DOI: http://doi.org/10.1287/trsc.2015.0655

Terms of use:

Some rights reserved. The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

This item was downloaded from IRIS Università di Bologna (https://cris.unibo.it/).
When citing, please refer to the published version.

(Article begins on next page)

12 October 2024

This is the final peer-reviewed accepted manuscript of:

Cacchiani, Valentina, and Juan-José Salazar-González. "Optimal solutions to a real-world integrated airline scheduling problem." *Transportation Science* 51.1 (2017): 250-268.

The final published version is available online at: https://doi.org/10.1287/trsc.2015.0655

Rights / License:

The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

This item was downloaded from IRIS Università di Bologna (https://cris.unibo.it/)

When citing, please refer to the published version.

TRANSPORTATION SCIENCE
Vol. 00, No. 0, Xxxxx 0000, pp. 000–000
ISSN 0041-1655 | EISSN 1526-5447 | 00 | 0000 | 0001

INFORMS
DOI 10.1287/xxxx.0000.0000
© 0000 INFORMS

Authors are encouraged to submit new papers to INFORMS journals by means of a style file template, which includes the journal title. However, use of a template does not certify that the paper has been accepted for publication in the named journal. INFORMS journal templates are for the exclusive purpose of submitting to an INFORMS journal and should not be used to distribute the papers in print or online or to submit the papers to another publication.

Optimal solutions to a real-world integrated airline scheduling problem

Valentina Cacchiani
DEI, University of Bologna, Italy, valentina.cacchiani@unibo.it

Juan-José Salazar-González
DMEIO, Universidad de La Laguna, Spain, jjsalaza@ull.es

We study an integrated airline scheduling problem for a regional carrier. It integrates three stages of the planning process (i.e. fleet assignment, aircraft routing and crew pairing) that are typically solved in sequence. Aircraft maintenance is also taken into account. The objective function aims at minimizing a weighted sum of the number of aircraft routes, the number of crew pairings, and the waiting times of crews between consecutive flights. In addition, it aims at maximizing the *robustness* of the solution by also minimizing the number of times that crews need to change aircraft. We present two Mixed Integer Linear Programming models for the integrated problem. The first formulation, called *path-path model*, can be considered as the "natural model" in which both the crew pairings and the aircraft routes are represented by path-based variables. The other formulation, called *arc-path model*, is a novel model in which the aircraft routes are represented by arc-based variables and the crew pairings by path-based variables. We propose two exact methods (called path-path method and arc-path method) for solving the integrated problem, each one based on one of the proposed models. Both methods consist of three phases. In the first phase, the Linear Programming relaxation of the corresponding model is solved to optimality by column generation on the path-based variables, thus providing a lower bound. The second phase computes a heuristic solution (upper bound) by using only the variables generated in the first phase. The third phase makes use of the lower and upper bounds (obtained in the previous phases) to compute an optimal solution. We propose a *bounding cut* based on computing a lower bound on the number of aircraft changes that are needed in a feasible solution, and empirically show that this cut significantly speeds up the exact methods. The proposed methods are tested on real-world instances of a regional carrier with up to 172 flights and three fleet operators. The results show that the arc-path method outperforms the path-path method as well as a heuristic approach from the literature, and derives the optimal solutions for all the considered instances in at most two hours of computing time.

Key words: Integrated airline scheduling; exact algorithm; column generation

1. Introduction

Airline companies need solution approaches for solving complex logistic problems. Aircraft and crews are expensive resources that need efficient utilization. They must be allocated to flights in a schedule at minimum cost subject to many rules so that each flight is covered exactly once. The rules are related to technical reasons or requirements forced by the company or by the crew unions, and insert a high degree of complexity in the logistic problems. Therefore, the airline scheduling problem is usually decomposed into stages that are solved in sequence. Initially, the *schedule of the flights* has to be determined by specifying origin and destination airports, as well as departure and arrival times for each flight. Then, the *fleet assignment* must be solved: it consists of deciding which fleet (aircraft type) must be assigned to each flight. This stage is especially important when the airline company is composed of different fleet operators (as it occurs in the application motivating this article) or different aircraft types. Subsequently, the *aircraft routing* problem looks for determining minimum-cost aircraft routes so that each flight is operated by an aircraft and maintenance requirements are satisfied. Afterwards, the *crew pairing* problem calls for finding minimum-cost crew routes so that each flight is operated by one crew and the work rules are respected. The *tail assignment* problem is the task of assigning aircraft routes to specific aircraft (i.e. the roster for each aircraft) while maintaining operational constraints. Finally, the *crew rostering* problem is solved to derive the roster for each crew. We refer the reader to Barnhart et al. (1998a), Belobaba et al. (2009), Klabjan (2005) and Yu (1998) for a detailed description of the various stages.

There are many publications using Operations Research in the Airline Industry. Most of them concern models and algorithms for solving one stage of the logistic problem. However, as observed, for example, in Mercier et al. (2005) and in Papadakos (2009), because of the interdependence of the stages, even if applying a sequential procedure reduces the computational complexity of the whole problem, it produces suboptimal solutions. For this reason, several works integrate two or more stages into a single problem (see Section 1.2). Our work is also focused on integration of multiple stages. More precisely, we propose models and algorithms for solving a single problem that integrates fleet assignment, aircraft routing and crew pairing stages for a regional carrier. In the following, we provide a general description of these stages, while the specificities of dealing with a regional carrier are detailed in Section 2. The most relevant articles integrating various stages are then summarized in Section 1.2. The section ends with an outline of the contributions of our work (see Section 1.3).

1.1. Fleet assignment, Aircraft routing and Crew Pairing

A set of scheduled *flights* is given in a time horizon (e.g. a day or a week). They connect a given set of airports. Each flight is defined by its departure and arrival times, and departure and arrival airports.

Fleet assignment problem. In this stage, the number of aircraft of each fleet is given, as well as the flight schedule. *Fleet assignment* calls for determining which fleet should operate each scheduled flight, while using its available resources and minimizing the assignment costs. This stage is often performed in combination with flight scheduling or aircraft routing (see Section 1.2).

Aircraft routing problem. This stage is devoted to determining the best route for the aircraft, given the flight schedule and the fleet assignment. Maintenance rules for aircraft are taken into account. In particular, two types of *maintenance* are needed for the aircraft. A long-term maintenance is scheduled during a long-term period (e.g. a year), and requires several days (e.g. 8 days). Aircraft that are undergoing this type of maintenance are not available during these days, and this observation is considered when solving the fleet assignment (unavailable aircraft reduces the fleet size during those days). A short-term maintenance takes place every given number of days as defined by the carrier (e.g. 2 or 3 days) and must be done at airports with maintenance facilities, called *depots* or *maintenance stations*. Short-term maintenance must be considered in this stage, as it directly affects the aircraft routes. Indeed, aircraft need to be at a depot in order to perform maintenance. Other rules, which depend on the airline company (see Section 2), can be imposed on short-term maintenance.

Rules on flight sequencing for aircraft are considered in this stage. In particular, consider a pair of flights such that the arrival airport of the first flight coincides with the departure airport of the second flight, and the arrival time of the first flight is before the departure time of the second flight. The difference between the departure time of the second flight and the arrival time of the first flight is called *connection time* or *sit-time*. The two flights can be performed in sequence by the same aircraft if the connection time is at least equal to the so-called *plane-turn time* (e.g. 30 minutes). In addition, an aircraft is not allowed to go from one airport to another one unless it is serving a scheduled flight (we are not aware of articles allowing exceptions to this rule in the airline literature). A set of flights that can be performed in sequence by an aircraft is called *aircraft route*. The *aircraft routing* aims at determining the minimum-cost aircraft routes so that each flight is operated by exactly one aircraft and maintenance requirements are satisfied. This stage is often performed in combination with crew pairing (see Section 1.2).

Crew pairing problem. This stage is devoted to determining the best route for each crew, given the flight schedule, the fleet assignment and the aircraft routing. Each crew is associated with a fleet or operator, with specific and complex working rules (e.g. the maximum number of flights in a workday). The working rules also force rest on the crews on some itineraries, and this could impose significant costs to the company when, e.g., overnight rest requires crews to stay in a hotel. Rules on flight sequencing for crews are taken into account in this stage. Two flights can

be performed in sequence by the same crew if the connection time is within a given range (e.g. at least 30 minutes and at most 3 hours). In addition, a crew is not allowed to go from one airport to another one unless it is serving a scheduled flight (although there are exceptions to this rule in the literature: for example, Cordeau et al. (2001), Mercier et al. (2005) and Mercier and Soumis (2007) consider deadhead flights to reposition crews). Similar to the concept of aircraft route, a sequence of flights that can be performed by a crew is called *crew route* or *crew pairing* or *duty*. The *crew pairing* aims at determining the minimum-cost crew routes so that each flight is operated by exactly one crew and working rules are satisfied. This stage is often performed in combination with aircraft routing (see Section 1.2).

Robust schedule. A robust schedule is a schedule which allows to avoid delay propagation as much as possible. Robustness can be achieved in several stages of airline scheduling, such as fleet assignment, aircraft routing and crew pairing. In fleet assignment, tight plane-turn times can lead to delay propagation. In aircraft routing and crew pairing, delay propagation can increase when a crew needs to change aircraft during its duty, and it is more evident when flights are close in time. Robustness can be achieved by inserting buffer times to the plane-turn times or by limiting the number of times in which crews need to change aircraft. In particular, given a crew route and two consecutive flights in the route, we say that there is an *aircraft change* when the aircraft assigned to these flights are different (i.e. the crew needs to change aircraft). Analogously, given an aircraft route and two consecutive flights in the route, we say that there is a *crew change* when the crews assigned to these flights are different. Two flights are said to have a *short connection* when the connection time is smaller than a given amount of time (e.g. 60 minutes). Usually, short connections impose that the same crew and aircraft perform both flights, i.e. no aircraft change and/or crew change are allowed on short connections. Two flights are said to have a *restricted connection* when the connection time is larger than that of short connections, but smaller than a given amount of time (e.g. 90 minutes in Mercier et al. (2005)). On restricted connection it is not imposed to have the same crew and aircraft, although this is highly desired. Therefore, restricted connections can be seen as soft constraints leading to penalties in the objective function when the flights are operated by different crews or aircraft. Aircraft and crew changes need to be minimized in order to build a robust schedule.

1.2. Literature Review

There are several recent successful attempts in the literature to optimize two or more stages in an integrated airline scheduling problem. Due to the hardness of the problem, most of the articles propose heuristic approaches to solve it. In this section, we summarize the most relevant works, and classify them according to the integrated stages they consider. An important benefit that is

obtained, as shown in many existing works, by integrating two or more stages is that significant savings can be obtained with respect to the sequential solution.

Integration between flight scheduling and fleet assignment. An integration of flight scheduling and fleet assignment is studied in Lohatepanont and Barnhart (2004). The authors present integrated models that simultaneously optimize the selection of flights and the assignment of aircraft types to the selected flight legs. The goal is to maximize the profits while minimizing the operating costs. The schedule design is performed by modifying a given schedule (from the current or previous season). The solution method consists of a heuristic algorithm based on branch-and-bound, combined with column and row generation. An approximated integrated model is also proposed to solve larger instances. Computational results on data from a major U.S. airline suggest that significant benefits can be achieved through the proposed integration with respect to the planners solutions. Sherali et al. (2010) propose a MILP model for integrating flight scheduling with fleet assignment, that, as opposed to Lohatepanont and Barnhart (2004), directly incorporates multiple fare classes and the number of passengers to accept on each active path. The authors perform a polyhedral analysis to tighten the model with several classes of valid inequalities. Heuristic solution approaches are developed by applying Benders decomposition. Computational results, showing the benefits of the integration, are presented using real data obtained from United Airlines. Sherali et al. (2013a) extend the Benders decomposition heuristic methods presented in Sherali et al. (2010) by allowing flexibility on the departure times of the scheduled flights, by balancing flight schedules throughout the day, and by "recapturing" considerations, i.e. re-accommodating passengers to other flights operated by the same airline. The authors show the benefits (such as facilitating the generation of new connecting itineraries for serving certain markets through flight retiming or improving market shares by schedule balancing) of having these additional features, by using real data obtained from United Airlines. Pita et al. (2013) study the integrated flight scheduling and fleet assignment problem under airport congestion, which is increasing and is a major cause of delays. They propose an exact MILP model that takes aircraft and passenger delay costs into account, and aims at maximizing the expected profits. The model is solved for a case study involving the main network of TAP Portugal. The main benefit of this model is that aircraft and passenger delay costs are explicitly taken into account.

Integration between flight scheduling and aircraft routing. In Sherali et al. (2013b), flight scheduling, fleet assignment, and aircraft routing, including maintenance constraints, are integrated. In addition, other features, such as flight retiming and demand recapture issues, are incorporated. A MILP model is proposed for the integrated problem and heuristic Benders decomposition-based methods are designed. Symmetry breaking constraints and valid inequalities are proposed

to enhance the model. The methods are tested on instances based on real data provided by United Airlines. The results show the potential impacts on profits obtained by jointly considering several stages, as well as the additional features, and this is a major advantage of the proposed method.

Lan et al. (2006) propose two approaches to derive robust plans, in order to reduce the occurrence and impact of flight delays and disruptions. The first one aims at reducing delay propagation by appropriately routing aircraft. The second approach involves retiming flight departure times, in order to minimize the number of passenger missed connections.

Integration between fleet assignment and aircraft routing. Liang and Chaovalitwongse (2013) propose an integrated model to solve simultaneously the weekly aircraft maintenance routing problem and the weekly fleet assignment problem. The former is modeled by a network-based MILP formulation for the aircraft routing problem, in which weekly maintenance requirements are also taken into account. While the weekly schedule is more realistic, it also highly increases the complexity of the problem. The advantages of their proposed formulation are that its size increases only linearly with the size of the weekly schedule, and it provides a tight Linear Programming (LP) relaxation. A diving heuristic is used for solving the problem. In Haouari et al. (2011), the problem of integrated aircraft fleeting and routing at TunisAir is studied. The authors propose two methodologies in order to derive optimal solutions: the first one is based on Benders decomposition, while the second one is a branch-and-price algorithm. They observe that, for the considered real-world instances, the branch-and-price approach outperforms the Benders decomposition approach. However, the latter is able to obtain good solutions in very short computing times. The authors show that an important advantage of the proposed approach, which integrates aircraft fleeting and routing, is in obtaining significant cost savings with respect to a sequential approach. Zeghal et al. (2011) investigate the impact of flexibility on integrated aircraft fleeting and routing.

Integration between fleet assignment and crew pairing. Barnhart et al. (1998b) propose an approximate model for fleet assignment and crew pairing. It does not completely integrate the two problems, but rather includes a relaxation of the crew scheduling problem in the fleet assignment model. It requires that all the flight legs are assigned to an appropriate crew, but does not impose constraints on the crew pairings (such as maximum number of duties, or maximum time away from the crew base). Using data provided by a large airline, the authors show that their approach is both tractable and capable of producing improved solutions to the airline planning problem. Gao et al. (2009) propose an integrated fleet and crew robust planning method to provide fleet assignment solutions that are also robust to real-time operations, while taking into account crew planning constraints. The developed approach integrates crew connections, instead of explicit crew pairings, within the fleet assignment model. The numbers of fleet types and crew bases allowed

to serve each airport are limited. This assumption is useful to increase the opportunity of finding a *move-up* crew for crew recovery, i.e. a crew that can replace another crew, since it is able to operate a specific aircraft type and is from the same crew base. The integrated model is solved by a MILP solver.

Integration between aircraft routing and crew pairing. Cordeau et al. (2001) propose a heuristic approach for the simultaneous aircraft routing and crew scheduling problem. In particular, they present a mathematical formulation with exponentially many variables both for the aircraft routes and for the crew itineraries. The LP relaxation of the model is solved by column generation, and a diving heuristic approach is used to derive feasible solutions. In addition, the authors propose to apply Benders decomposition: the master problem takes into account the aircraft routing problem, while the subproblem contains the crew pairing problem. They propose a three phase heuristic approach. In the first phase the LP relaxation of the proposed decomposed model is solved by Benders decomposition and column generation. In the second phase, aircraft variables are imposed to be integer, and a diving approach is applied to derive a heuristic solution. Finally, in the third phase, integrality is imposed also on the crew variables and the subproblem is solved once while keeping the aircraft routes as fixed. The authors present computational results on data by a Canadian airline and show that the method based on Benders decomposition is able to obtain good heuristic solutions in shorter computing times than the diving heuristic approach. Mercier et al. (2005) extend the model to deal with *restricted connections*, i.e. connections between two flights that are not served by the same aircraft and are penalized because they can lead to delay propagation. This is an important achievement of their model, as it allows to obtain robust solutions. In addition, the authors improve the heuristic Benders decomposition approach by, on one hand, having the crew pairing problem in the master and the aircraft routing problem in the subproblem and, on the other hand, generating Pareto-optimal cuts. Mercier (2008) performs a theoretical comparison of different types of feasibility cuts and proposes a procedure to strengthen them. Mercier and Soumis (2007) further extend Cordeau et al. (2001) and Mercier (2008) by introducing a model that takes into account flight retiming, i.e. the possibility of changing the schedule of the flights in a time window of plus or minus five minutes. The three phase heuristic approach is modified by using a dynamic constraint generation to deal with the increased number of constraints and variables. Even if the problem with flexibility in the flight departure times is more complex, computational experiments on instances provided by two major airlines show that allowing this flexibility yields significant cost savings. Klabjan et al. (2002) show that in order to partially integrate aircraft routing and crew pairing, in absence of aircraft maintenance constraints, it is enough to add the so-called *plane-count constraints* to a set-partitioning model for

the crew pairing problem. They solve the two problems sequentially, but reverse the usual order in which they are solved in practice, i.e. they first solve the crew pairing problem, by including the plane-count constraints. The advantage, in this case, is that impact of an aircraft routing on the crew cost can be significant. Cohn and Barnhart (2003) integrate the crew pairing stage with the aircraft maintenance routing problem. In the aircraft maintenance routing problem, aircraft are assigned to a route of flights while ensuring that every aircraft goes through maintenance. In this stage, short connections are also considered. The authors propose an *extended crew pairing model* which contains exponentially many crew pairing variables and a collection of variables representing the complete solutions to the maintenance routing problem. An iterative heuristic approach is proposed in Weide et al. (2010) for solving aircraft routing and crew pairing problems, with the aim of deriving solutions that are robust to delays in airline operations. A minimal-cost crew pairing solution is determined without taking into account the aircraft routings. Then, at each iteration, the aircraft routing problem is solved, taking into account the current crew pairing solution. The obtained aircraft routing solution is used to solve the crew pairing problem again and the process is iterated until the level of robustness cannot be further improved. The objective function of each problem is used to pass information from one problem to the other one. In addition, the approach is extended to consider multiple crew types, such as captains, first officers and flight attendants, at the same time. The approach is tested on different schedules of a domestic airline. The results show that, although optimality cannot be guaranteed, more robust solutions with smaller costs are obtained with respect to those used in practice by the airline. Dunbar et al. (2012) present a heuristic approach to minimize the cost of propagated delay in a framework that integrates aircraft routing and crew pairing. Following the approach proposed in Weide et al. (2010), the interactions between the aircraft and the crew are modeled in an iterative way. The main elements of the approach consist of the accurate calculation of the combined effects of the delay propagation along the aircraft routes and the crew pairings, and the use of this information for both the calculation of the column costs and the dynamic selection of the optimal columns. Robustness is also taken into account in Froyland et al. (2014), who propose a recoverable robust approach to reduce the effect of disruptions and the cost of recovery for the tail assignment problem.

Integration between fleet assignment, aircraft routing and crew pairing. Papadakos (2009) propose a heuristic algorithm for the integrated fleet assignment, aircraft routing and crew pairing problem. Aircraft maintenance is also taken into account. The author presents a Benders decomposition approach where the master problem consists of the aircraft routing problem and the fleet assignment problem, and the subproblem corresponds to the crew pairing problem. Column generation is used for solving the master problem and is accelerated by using heuristic

methods. Pareto-optimal cuts are generated through the method proposed in Magnanti and Wong (1981). A three phase heuristic algorithm is proposed for solving instances of a major European airline and a major North American airline. The results show that operating costs are significantly reduced thanks to the proposed method. Sandhu and Klabjan (2007) study the integration of fleet assignment, aircraft routing and crew pairing problems, but aircraft maintenance requirements are neglected. The model uses the plane-count constraints presented in Klabjan et al. (2002) for ensuring aircraft routing feasibility. In order to solve the integrated model, the authors present two iterative heuristic approaches: one is based on a combination of Lagrangian relaxation and column generation, while the other one is based on Benders decomposition. In both approaches, the final solution is obtained by using the fleet assignment produced by the last iteration and by applying the traditional sequential approach to obtain aircraft routes and crew pairings. Based on computational experiments on real-world data from a major U.S. carrier, the authors conclude that the Lagrangian approach outperforms, on average, the Benders decomposition approach.

1.3. Contributions of the paper

The above-cited articles deal with problems arising in large airline companies. In this paper, we focus on a real-world application for a regional carrier. Regional carriers have specific requirements, which make the problem different from the ones addressed in the airline literature. These differences are mainly related to the constraints on the aircraft maintenance and on the crew overnight rest, but also to the importance of determining robust solutions. Although these issues are also important to large carriers, they have a significant impact, and thus cannot be neglected, for regional carriers (see Section 2). Section 2 details the characteristics of the regional carrier motivating this paper. It is also interesting to observe that existing approaches integrating fleet assignment, aircraft routing and crew pairing problems are of heuristic nature (while we propose exact methods), and neglect one or more aspects (e.g. aircraft changes, aircraft maintenance) considered in this work.

The literature on regional carriers is quite limited. Salazar-González (2014) studies the same real-world application motivating the research leading to this paper. It concerns a regional carrier flying in Canary Islands with 18 aircraft owned by three operators to serve up to 172 flights per day. The article formulates the problem using arc-based variables both for the aircraft routes and for the crew pairings, and describes an iterative heuristic procedure that sequentially applies two steps at each iteration. The first step generates, in a greedy way, crew pairings covering all the flights. The second step determines aircraft routes minimizing the aircraft changes. This heuristic algorithm allowed the company to automate a difficult task that was previously done manually by a team of technicians.

The first contribution of this paper is to propose two alternative MILP formulations for the integrated problem. The first model, called *path-path model*, uses path-based variables to describe both the aircraft routes and the crew pairings. The second model, called *arc-path model*, is built on arc-based variables to describe the aircraft routes and path-based variables to represent the crew pairings. While the first model can be considered as a "natural model" for the integrated problem (see e.g. Cordeau et al. (2001)), the second one is a novel model.

The second contribution is the development of two exact algorithms (called path-path method and arc-path method, respectively) to solve the integrated problem, each one based on one of the two MILP models. Both methods consist of three phases. In the first phase, the LP relaxation of the corresponding model is solved to optimality by column generation on the path-based variables, and a lower bound is derived. The second phase derives a heuristic solution (upper bound) to the problem. The third phase computes the optimal solution: in the path-path method a branch-and-price algorithm is adopted for finding the optimal solution, while the arc-path method solves a reduced MILP containing all the variables with reduced cost in the gap between the obtained upper and lower bounds. Both exact algorithms, besides minimizing a weighted sum of the aircraft route costs and crew pairing costs, aim at the maximization of the robustness of the computed solution, i.e. they search for a solution which minimizes the aircraft changes. More precisely, the objective function considered in the two algorithms minimizes the number of aircraft routes, the number of crew routes, the connection times between consecutive flights in crew routes, and the number of aircraft changes.

The third contribution is to determine a new *bounding cut* that significantly speeds up the solution process. It is derived by computing a lower bound on the number of aircraft changes that are needed in a feasible solution. The two exact algorithms, enhanced with this cut, are tested on real-world instances provided by a regional carrier flying in Canary Islands, and optimal solutions are found for instances with up to 172 flights. According to our computational results, the arc-path method outperforms the path-path method. In addition, the arc-path method outperforms the heuristic algorithm proposed in Salazar-González (2014), currently used by the airline company. Providing optimal solutions for the real-world integrated problem is the fourth contribution of this paper.

The structure of the paper is the following. Section 2 describes the integrated problem. Section 3 presents the two proposed MILP formulations. The two exact algorithms, as well as the bounding cuts, are described in Section 4. Computational results are discussed in Section 5 and conclusions are drawn in Section 6.

2. Problem description

This section details the specific characteristics of the integrated problem arising in the regional airline company motivating our research. The section underlines the differences with respect to the problems studied in the literature, concerning large airline companies.

The considered company owns small aircraft (ATR 72), which are very sensitive to weather conditions (e.g. fog). Therefore, delays are common in practice, and determining a robust schedule is a crucial goal that cannot be neglected. In addition, the airline has a single depot which can perform maintenance to at most half of the aircraft simultaneously. Since no flight is scheduled between 11pm and 7am, maintenance is always performed during the night. Aircraft must undergo a short-term maintenance every 3 days at most. The company implements this requirement by imposing (with a few exceptions described below) that each aircraft stays one night at the airport where the depot is, and the next night at a different airport. A *base* is an airport where an aircraft can stay during the night. Crews are also subject to rules on the origin and destination airports of their duties: due to budget limitation, the company requires (with a few exceptions described below) that every crew ends its duty at the airport closest to its home, thus avoiding overnight rest outside home. A *home base* is an airport close to the home of some crews, i.e. an airport where a crew route can start and end without creating an overnight cost to the company.

These specific requirements (robustness, maintenance, overnight rest) are crucial for regional carriers. In the following, we give details on the problem at study.

The regional carrier is composed of three *operators*. An operator is a company owning some aircraft and managing some crews. Even though all aircraft are identical, the crews of each operator have specific rules and different salaries. A crew of an operator cannot operate an aircraft of another operator. In addition, a crew route has constraints, which depend on the operator, on the maximum number of flights that it can contain and on its duration. In particular, given the departure time of the first flight of the route, the maximum number of flights in the route and the duration of the route are specified. The aircraft and crews of each operator available at each airport at the beginning of the time horizon is given.

The carrier operates flights between islands, connecting small airports which are located near residential areas and are closed during the night. Therefore we consider a time horizon of one day. At the time of writing this paper, the number of aircraft is 18, the number of flights per day is 150 on average, and the number of involved airports is 11 (they are LPA, TFN, TFS, SPC, VDE, ACE, FUE, GMZ, FNC, EUN, RAK). Although in principle each one of the 11 airports could be a base, in our real-world instances only four airports (TFN, LPA, SPC, ACE) are bases, and two are home bases (TFN and LPA). The company has a single depot which is LPA, and requires that each aircraft stays one night in LPA and the next night in a non-depot base. However, a few exceptions

are possible: it is allowed that up to D_{NB} (equal to 2 in our real-world application) aircraft stay two consecutive nights at TFN, and it is also allowed that up to D_B (equal to 2 in our real-world application) aircraft stay two consecutive nights in LPA. These special cases are called *short-term maintenance exceptions*. In addition, aircraft starting from a non-home base must end its route at the depot, and aircraft can end its route at a non-home base only if it departed from the depot. Similarly, each crew must return to its home base by the end of the duty. However, due to the flight schedule, an *overnight rest* outside the home base is sometimes unavoidable (in SPC or in ACE). The overnight rest must be done by a crew whose home base is the closest to the base in which it will stay during the night, and is only allowed by a subset of the operators. These special cases are called *overnight rest exceptions*. Some of the scheduled flights (*international flights*) go to/from airports (CMN, RAK, EUN, FNC and AGA) located outside the islands. Only a subset of the operators is allowed to perform these flights.

Flight sequencing rules are also given. In the considered application, the plane turn time is 20 minutes, and a connection is short when the connection time is smaller than 30 minutes. Two flights can be performed in sequence by the same crew if the connection time is at least 20 minutes and at most 3 hours. On short connections aircraft changes are not allowed. On all other connections, aircraft changes are penalized. This is different from what happens in the literature, where aircraft changes are only considered on a subset of the non-short connections (called restricted connections in Mercier et al. (2005)). This also increases the complexity of the considered problem.

A cost is assigned to each crew route according to the operator and the connection times in the route: very long and very short connection times are considerably penalized. Indeed, short connection times are not desired because they can lead to delay propagation, while long connection times should be avoided because, for this company, the crew cost increases as a monotone piecewise linear function of the time spent without flying. Instead of assigning a cost to each aircraft route, the company is interested in minimizing the number of aircraft used in the solution: indeed, aircraft that are not used during the day do not need short-term maintenance, and this leads to a cost reduction. In addition, a (non-negligible) cost is assigned to each aircraft change.

Summarizing, the problem studied in this paper calls for simultaneously finding a feasible assignment of operators to flights, and a routing for aircraft and crews, such that all the flights are performed, with the goal of minimizing the total cost and maximizing the robustness of the schedule.

3. Mathematical Formulations

Salazar-González (2014) presents a mathematical model for the integrated problem using arc-based variables to represent both aircraft and crew routes. The model has the disadvantage of requiring

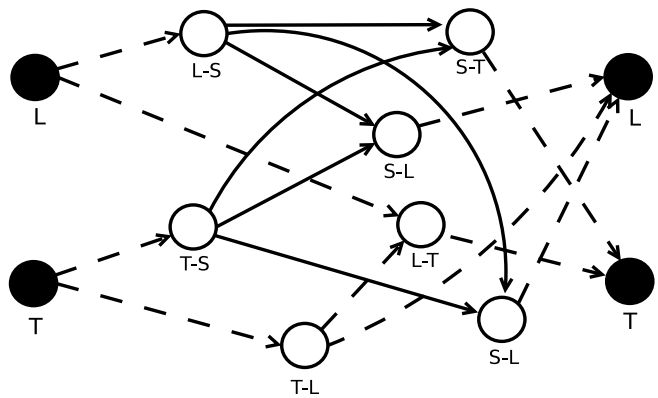

Figure 1 **Example of an aircraft graph.**

weak inequalities to avoid infeasible crew routes. A branch-and-cut algorithm based on this model is not able to determine optimal solutions on real-world instances in reasonable computing time. In this section we present two alternative MILP formulations. The first formulation (called *path-path model*) uses path-based variables to represent both the aircraft routes and the crew routes. The second formulation (called *arc-path model*) uses arc-based variables to represent the aircraft routes and path-based variables to represent the crew routes.

We construct two directed acyclic graphs used in both formulations: the first graph, called *aircraft graph*, is used to represent feasible sequences of flights in aircraft routes, while the second one, called *crew graph*, is used to represent feasible sequences of flights in crew routes. A sequence of flights is feasible for an aircraft if the plane turn time is respected, while it is feasible for a crew if the maximum connection time of three hours is also respected. Both graphs have the same set of nodes N, given by the union of the sets of nodes N^f, representing the flights, N^{bd}, representing the departure bases, and N^{ba}, representing the arrival bases. We duplicate the nodes corresponding to the bases in departure and arrival base nodes, respectively, in order to work with acyclic graphs. The aircraft graph $G^a = (N, A^a)$ (the crew graph $G^c = (N, A^c)$ resp.) contains an arc $(i, j) \in A^a$ ($(i, j) \in A^c$ resp.) if either

- $i, j \in N^f$ represent two flights that can be in sequence in a route, or
- $i \in N^{bd}$ corresponds to a base and $j \in N^f$ departs from that base, or
- $j \in N^{ba}$ corresponds to a base and $i \in N^f$ arrives at that base.

Note that A^c is a subset of A^a since connection times larger than three hours are allowed for aircraft but not for crews. We denote by A^c_s the subset of arcs corresponding to short connections.

Figure 1 shows an example of an aircraft graph G^a. It corresponds to an instance with seven flights (indicated by white nodes), connecting three airports (LPA denoted by L, TFN by T, and SPC by S), two of which (LPA and TFN) are bases (indicated by black nodes). The letter under a

base indicates the corresponding airport, and the pair of letters under each flight reports the origin-destination airports. We do not report the departure and arrival times of the flights, but consider that the time increases along the horizontal axis from left to right. In this graph, N^{bd} contains the two black nodes on the left, N^{ba} contains the two black nodes on the right, and N^f contains the seven white nodes. From every departure base, there is an arc to every flight that departs from that base. Similarly, there is an arc to every arrival base from every flight that arrives at that base. These arcs are indicated as dotted arrows. In addition, there is an arc between every two flights that can be done in sequence by an aircraft (the arrival airport of the first flight coincides with the departure airport of the second flight, and the plane turn time is respected). These arcs are represented as normal arrows. In the example, we can see that flight L-T cannot be done after flight S-L because they are partially overlapping in time (i.e. the plane turn time is not respected). A crew graph is very similar to an aircraft graph, with the only difference that arcs connecting flights with connection time over three hours are not allowed.

The airline company is divided in operators, represented by the set K. For each operator $k \in K$ and each base $l \in N^{bd}$, let n_c^{kl} and n_a^{kl} be the number of crews and aircraft, respectively, of operator k available at base l at the beginning of the workday.

Crew and aircraft routes are identified by an operator $k \in K$ and a departure base $l \in N^{bd}$. Indeed, it is necessary to know the departure base of a crew because it restricts the possible airports at which the crew can arrive at the end of the workday. In particular, a crew must go back to its home base, unless an overnight rest outside the home base is needed. In this case, a crew is allowed to arrive at the base that is closest to its home base. Similarly, it is necessary to know the departure base of an aircraft because it restricts the possible airports at which the aircraft can arrive at the end of the workday, according to the short-term maintenance constraints.

In both models, we use path-based variables to represent the crew routes. Since we duplicate the nodes corresponding to the bases, a crew route corresponds to a path in G^c from a departure base to an arrival base. We define $\mathcal{R}_c^{kl}$ as the set of feasible crew routes in the graph $G^c = (N, A^c)$ for a crew of operator $k \in K$ departing from base $l \in N^{bd}$. A crew route is feasible if it satisfies constraints on the time duration of the route, on the maximum number of flights it executes and on the flight sequencing. In addition, it must respect the constraints on the departure and arrival bases and on the international flights, as described before.

When convenient for the notation, a route $R \in \mathcal{R}_c^{kl}$ is a sequence of arcs or a sequence of nodes. Each crew route R has an associated cost c_R which depends on the operator and on the weighted sum of the connection times between consecutive flights. The weights on the connection times are used to penalize very long and very short connection times.

We define the following variables, used in both formulations, to represent crew routes and aircraft changes, respectively. For each crew route $R \in \mathcal{R}_c^{kl}$ let x_R be a binary variable such that $x_R = 1$ if and only if route R is assigned to a crew of operator $k \in K$ departing from base $l \in N^{bd}$. For each arc $(i,j) \in A^c \setminus A_s^c$ let z_{ij} be a binary variable such that $z_{ij} = 1$ if and only if an aircraft change occurs between flights i and j $(i,j \in N^f)$.

Recall that D_B is the maximum number of aircraft that can stay two consecutive nights in the depot, and D_{NB} is the maximum number of aircraft that can stay two consecutive nights in a non-depot base. The set of depots is represented by B and the set of home bases by H. In our real-world application, $N^{bd} = \{LPA, TFN, SPC, ACE\}$, $H = \{\text{LPA,TFN}\}$ and $B = \{\text{LPA}\}$.

To model the multi-criteria nature of the problem through a single objective function we consider four weights:

- α: weight on the sum of the connection times in the crew routes;
- β_k: weight on the number of crew routes of operator $k \in K$;
- γ_k: weight on the number of aircraft routes of operator $k \in K$ (it takes into account the cost for short-term maintenance exceptions);
- δ: weight on the number of aircraft changes in the crew routes.

These weights are input parameters defined by the airline company after performing a sensitivity analysis.

Sections 3.1 and 3.2 present the two proposed models.

3.1. Path-path model

Let $\mathcal{R}_a^{kl}$ be the set of feasible aircraft routes in graph $G^a = (N, A^a)$ for an aircraft of an operator $k \in K$ starting from base $l \in N^{bd}$. An aircraft route is feasible if it satisfies constraints on the flight sequencing and on the departure and arrival bases according to the short-term maintenance rules. For each aircraft route $R \in \mathcal{R}_a^{kl}$, we introduce a binary variable y_R such that $y_R = 1$ if and only if route R is assigned to an aircraft. Let $\mathcal{R}_B^{kl} \subseteq \mathcal{R}_a^{kl}$ be the subset of aircraft routes such that they start and end at depot bases. Let $\mathcal{R}_H^{kl} \subseteq \mathcal{R}_a^{kl}$ be the subset of aircraft routes such that they start and end at non-depot bases.

A mathematical formulation for the problem minimizes

$$\sum_{k \in K, l \in N^{bd}, R \in \mathcal{R}_c^{kl}} (\alpha \cdot c_R + \beta_k) \cdot x_R + \sum_{k \in K, l \in N^{bd}, R \in \mathcal{R}_a^{kl}} \gamma_k \cdot y_R + \delta \cdot \sum_{(i,j) \in A^c \setminus A_s^c} z_{ij} \tag{1}$$

subject to constraints on the crew variables:

$$\sum_{k \in K, l \in N^{bd}, R \in \mathcal{R}_c^{kl} : i \in R} x_R = 1 \qquad \forall i \in N^f \tag{2}$$

$$\sum_{R \in \mathcal{R}_c^{kl}} x_R \le n_c^{kl} \qquad \forall k \in K, l \in N^{bd} \tag{3}$$

$$x_R \in \{0,1\} \qquad \forall k \in K, l \in N^{bd}, R \in \mathcal{R}_c^{kl}, \tag{4}$$

constraints on the aircraft variables:

$$\sum_{R \in \mathcal{R}_a^{kl}} y_R \leq n_a^{kl} \qquad \forall k \in K, l \in N^{bd} \tag{5}$$

$$\sum_{k \in K, l \in N^{bd}, R \in \mathcal{R}_H^{kl}} y_R \leq D_{NB} \tag{6}$$

$$\sum_{k \in K, l \in N^{bd}, R \in \mathcal{R}_B^{kl}} y_R \leq D_B \tag{7}$$

$$y_R \in \{0,1\} \qquad \forall k \in K, l \in N^{bd}, R \in \mathcal{R}_a^{kl}, \tag{8}$$

and constraints on linking crew routes and aircraft routes, on the short connections and on the aircraft changes:

$$\sum_{l \in N^{bd}, R \in \mathcal{R}_c^{kl}: i \in R} x_R = \sum_{l \in N^{bd}, R \in \mathcal{R}_a^{kl}: i \in R} y_R \qquad \forall k \in K, i \in N^f \tag{9}$$

$$\sum_{l \in N^{bd}, R \in \mathcal{R}_c^{kl}: (i,j) \in R} x_R = \sum_{l \in N^{bd}, R \in \mathcal{R}_a^{kl}: (i,j) \in R} y_R \qquad \forall k \in K,\ i,j \in N^f : (i,j) \in A_s^c \tag{10}$$

$$\sum_{k \in K, l \in N^{bd}, R \in \mathcal{R}_c^{kl}: (i,j) \in R} x_R \leq \sum_{k \in K, l \in N^{bd}, R \in \mathcal{R}_a^{kl}: (i,j) \in R} y_R + z_{ij} \qquad \forall i,j \in N^f : (i,j) \in A^c \setminus A_s^c \tag{11}$$

$$z_{ij} \in \{0,1\} \qquad \forall i,j \in N^f : (i,j) \in A^c \setminus A_s^c. \tag{12}$$

The objective function (1) takes into account the four criteria through a single weighted function. Constraints (2) impose that each flight must be assigned to a crew. Constraints (3) require to respect the maximum number of crews available for each operator at each base. Constraints (5) impose not to use more than the maximum number of aircraft available for each operator at each base. Constraints (6) and (7) take into account the short-term maintenance exceptions. At most D_{NB} aircraft are allowed to depart from and return to a home base that is not the depot. Similarly, at most D_B aircraft are allowed to depart from and return to the depot. Constraints (9) are linking constraints between crew routes and aircraft routes. They impose that each flight is operated by a crew and an aircraft of the same operator. Constraints (10) concern the short connections: if an aircraft route (crew route resp.) executes in sequence flights i and j such that their connection time is shorter than 30 minutes, then also a crew route (aircraft route resp.) must execute these flights. Essentially aircraft changes are forbidden for short connections. Constraints (11) are used to count the aircraft changes: given two flights i and j, if they are operated in sequence by a crew route, then either an aircraft route executes them in sequence or there is an aircraft change. Finally, constraints (4), (8) and (12) impose the variables to be binary.

3.2. Arc-path model

In order to define the aircraft routes, we introduce, for each arc $(i,j) \in A^a$, each operator $k \in K$ and each base $l \in N^{bd}$, a binary arc-flow variable y_{ij}^{kl} such that $y_{ij}^{kl} = 1$ if and only if arc (i,j) is operated by an aircraft of operator k departing from base l.

A mathematical formulation for the problem minimizes

$$\sum_{k\in K, l\in N^{bd}, R\in\mathcal{R}_c^{kl}} (\alpha \cdot c_R + \beta_k) \cdot x_R + \sum_{k\in K, l\in N^{bd}, i\in N^{bd}, j\in N^f:(i,j)\in A^a} \gamma_k \cdot y_{ij}^{kl} + \delta \cdot \sum_{(i,j)\in A^c \setminus A_s^c} z_{ij} \tag{13}$$

subject to constraints on the crews:

$$\sum_{k\in K, l\in N^{bd}, R\in\mathcal{R}_c^{kl}: i\in R} x_R = 1 \qquad \forall i \in N^f \tag{14}$$

$$\sum_{R\in\mathcal{R}_c^{kl}} x_R \le n_c^{kl} \qquad \forall k \in K, l \in N^{bd} \tag{15}$$

$$x_R \in \{0,1\} \qquad \forall k \in K, l \in N^{bd}, R \in \mathcal{R}_c^{kl}, \tag{16}$$

constraints on the aircraft:

$$\sum_{(i,j)\in A^a} y_{ij}^{kl} = \sum_{(j,i)\in A^a} y_{ji}^{kl} \qquad \forall k \in K, l \in N^{bd}, i \in N^f \tag{17}$$

$$\sum_{i\in N^{bd}, j\in N^f:(i,j)\in A^a} y_{ij}^{kl} \le n_a^{kl} \qquad \forall k \in K, l \in N^{bd} \tag{18}$$

$$y_{ij}^{kl} = 0 \qquad \forall k \in K, l \in N^{bd} \setminus H, i \in N^f, j \in N^{ba} \setminus B \tag{19}$$

$$y_{ij}^{kl} = 0 \qquad \forall k \in K, l \in H \setminus B, i \in N^f, j \in N^{ba} \setminus H \tag{20}$$

$$\sum_{k\in K, l\in H\setminus B, i\in N^f, j\in B:(i,j)\in A^a} y_{ij}^{kl} \le D_{NB} \tag{21}$$

$$\sum_{k\in K, l\in B, i\in N^f, j\in B:(i,j)\in A^a} y_{ij}^{kl} \le D_B \tag{22}$$

$$y_{ij}^{kl} \in \{0,1\} \qquad \forall k \in K, l \in N^{bd}, (i,j) \in A^a, \tag{23}$$

and constraints on linking crew routes and aircraft routes, on the short connections and on the aircraft changes:

$$\sum_{l\in N^{bd}, R\in\mathcal{R}_c^{kl}: i\in R} x_R = \sum_{l\in N^{bd}, (i,j)\in A^a} y_{ij}^{kl} \qquad \forall k \in K, i \in N^f \tag{24}$$

$$\sum_{l\in N^{bd}, R\in\mathcal{R}_c^{kl}:(i,j)\in R} x_R = \sum_{l\in N^{bd}} y_{ij}^{kl} \qquad \forall k \in K,\ i,j \in N^f : (i,j) \in A_s^c \tag{25}$$

$$\sum_{k\in K, l\in N^{bd}, R\in\mathcal{R}_c^{kl}:(i,j)\in R} x_R \le \sum_{k\in K, l\in N^{bd}} y_{ij}^{kl} + z_{ij} \qquad \forall i,j \in N^f : (i,j) \in A^c \setminus A_s^c \tag{26}$$

$$z_{ij} \in \{0,1\} \qquad \forall i,j \in N^f : (i,j) \in A^c \setminus A_s^c. \tag{27}$$

The objective function (13), constraints (14) and constraints (15) are the same as in the path-path model. Constraints (17) impose the flow conservation at every node corresponding to a flight (and together with constraints (14) impose that each flight must be operated by exactly one aircraft). Constraints (18) require to respect the maximum number of aircraft available for each operator at each base. Constraints (19)–(22) are used to express the maintenance rules, including the exceptions. Constraints (19) and (20) avoid infeasible aircraft routes due to maintenance and overnight rest constraints. In particular, constraints (19) impose that if an aircraft route departs from a non-home base, then it must arrive at the depot where maintenance is performed; constraints (20) impose that if an aircraft route departs from a non-depot home base then it must arrive at a home base. Constraints (21) and (22) take into account the short-term maintenance exceptions. At most D_{NB} aircraft routes are allowed to start and end in non-depot bases. Similarly, at most D_B aircraft routes are allowed to start and end at depot bases. Constraints (24) are linking constraints between crew and aircraft routes. They impose that each flight is operated by exactly one crew and one aircraft of the same operator. In particular, we consider all the outgoing arc-flow variables from node $i \in N^f$ (to an arrival base or to another flight). If there is an outgoing arc from node i of an operator k, then there must be a crew route of the same operator visiting node i. Constraints (25) forbid aircraft changes on short connections: given a pair of nodes corresponding to two flights such that their connection time is shorter than 30 minutes, if this pair belongs to a crew route of an operator k, then there must be an aircraft of operator k that executes the arc connecting the two nodes. Finally, constraints (26) are used to count the aircraft changes through z variables: given two flights i and j, if they are operated in sequence by a crew, then either an aircraft executes them in sequence or there is an aircraft change.

We conclude this section by observing that the LP relaxations of both models have the same optimal objective value. Recall that aircraft do not have any constraints neither on the number of flights nor on the time duration of a route. The maintenance constraints are managed by imposing conditions on the origins and destinations of the aircraft routes in the time horizon. Thus, for fixed values of x and z variables, the y variables in the arc-path model represent a flow (of aircraft) in a capacitated network. Since a flow can be decomposed into paths (see e.g. Cook et al. (1998)), each solution of the LP relaxation of the arc-path model corresponds to a solution of the LP relaxation of the path-path model of the same value, and vice versa. Indeed, also the y variables in the path-path model define a flow, and are related to the y variables in the arc-path model through the identities:

$$y_{ij}^{kl} = \sum_{R \in \mathcal{R}_a^{kl}:(i,j) \in R} y_R \qquad \forall k \in K, l \in N^{bd}, (i,j) \in A^a.$$

For this reason, the integrality constraints are not required on the y variables once they are forced on the x and z variables, and the path-path model can be seen as a Dantzig-Wolfe decomposition of the arc-path model.

4. Solution Approaches

This section describes two exact algorithms, based on the above models, to obtain optimal solutions to the integrated problem. Section 4.1 describes the column generation process used in both approaches for generating the crew routes. This process is also used to manage the aircraft routes when considering the path-path model. Section 4.2 details the three phases (lower bound computation, upper bound computation and optimal solution computation) in the two algorithms. Finally, Section 4.3 describes the basic and improved versions of a bounding cut that speeds up the two approaches.

4.1. Column generation

The models proposed in Section 3 have exponentially many variables x_R ($R \in \mathcal{R}_c^{kl}$). In addition, the path-path model has exponentially many variables y_R ($R \in \mathcal{R}_a^{kl}$). To solve the LP relaxation of these models, we apply a column-generation approach that dynamically introduces variables. The approach iteratively generates crew routes associated with variables having negative reduced costs, and also aircraft routes when applied to the path-path model. It is based on solving a pricing problem that calls for finding the route with the smallest reduced cost, and corresponds to an Elementary Shortest Path with Resource Constraints (ESPRC) to be computed on the acyclic graphs defined above. Indeed a path from a departure base $l \in N^{bd}$ to an arrival base $j \in N^{ba}$ in graph G^c corresponds to a feasible crew route if the constraints for the crew pairings are respected (see Section 2). More precisely, the overnight rest outside the home base and the international flights can be done only by a subset of the operators, two flights can be in sequence only if the minimum connection time is respected, the maximum number of flights in a route and the route duration must be respected, and the arrival airport of the route must coincide with the departure airport unless an overnight rest is needed.

The pricing problem is solved by a dynamic programming approach that is described in this section for the generation of crew routes in the case of the path-path model. The procedure is the same in the case of the arc-path model, and can be easily adapted to generate aircraft routes for the path-path model.

Let φ_i ($i \in N^f$) be the dual variables of constraints (2), ψ^{kl} ($k \in K$, $l \in N^{bd}$) the dual variables of constraints (3), ζ_i^k ($k \in K$, $i \in N^f$) the dual variables of constraints (9), ξ_{ij}^k ($k \in K$, $(i,j) \in A_s^c$) the dual variables of constraints (10), and η_{ij} ($(i,j) \in A^c \setminus A_s^c$) the dual variables of constraints (11). The set of arcs corresponding to short connections in route $R \in \mathcal{R}_c^{kl}$ is indicated with $R \cap A_s^c$. The

pricing problem consists of determining, for each operator $k \in K$ and for each base $l \in N^{bd}$, a crew route $R \in \mathcal{R}_c^{kl}$ such that:

$$\alpha \cdot c_R + \beta_k - \sum_{i \in R} (\varphi_i + \zeta_i^k) - \sum_{(i,j) \in R} \eta_{ij} - \sum_{(i,j) \in R \cap A_s^c} \xi_{ij}^k - \psi^{kl} < 0$$

and such that all the constraints described in Section 2 for the crew routes are satisfied.

The dynamic programming approach is executed for each operator $k \in K$ and for each departure base $l \in N^{bd}$. Given k and l, each flight $i \in N^f$ is assigned a set of labels (instead of a single label). Indeed, recall that the route duration and the maximum number of flights in the route depend on the departure time of the first flight of the route. Therefore, for each value of resource consumption (route duration and number of flights in the route), we need to store the information on the best path from l to i. If flight i has a departure airport that is different from the departure base l, then it is not considered.

Given k and l, let λ_i be one of the labels of flight $i \in N^f$. The label stores the following information:

1. the profit of a path R from l to i, defined as

$$\pi(\lambda_i) := -\alpha \cdot c_R + \sum_{f \in R} (\varphi_f + \zeta_f^k) + \sum_{(f,j) \in R} \eta_{fj} + \sum_{(f,j) \in R \cap A_s^c} \xi_{fj}^k + \psi^{kl},$$

2. the predecessor flight $\nu(\lambda_i)$ of i in the path,
3. the number of flights $F(\lambda_i)$, including flight i, operated in the path,
4. the time instant $\theta(\lambda_i)$ of the departure of the first flight in the path
5. the duration $D(\lambda_i)$ of the path until the end of flight i.

The flights are sorted according to increasing departure times. The dynamic programming approach consists of labeling, from each label of a flight i, each successor flight j (i.e. each flight j such that $(i,j) \in A^c$) when the following conditions simultaneously hold:

1. the maximum number of flights (including j) is respected;
2. the maximum duration of the route (including j) is respected;
3. if j is an international flight and the operator k allows to execute it;
4. if the maximum number of flights is reached by inserting j in the route, then the arrival airport of j must coincide with the departure base of the first flight in the route (this also includes the check on the possible overnight rest outside the home base).

The constraints on the sequencing of the flights in a route are imposed directly through the definition of the arcs in graph G^c.

Before labeling, from each label of a flight i, each successor flight j, we check whether we can remove some of the labels of flight i with the following dominance rule. Let λ_i and n_i be two labels

of flight i. Then label λ_i is dominated by label n_i (thus λ_i can be removed from the set of labels of flight i) if $\pi(n_i) > \pi(\lambda_i)$, $F(n_i) \leq F(\lambda_i)$ and $D(n_i) \leq D(\lambda_i)$.

After all the flights have been labeled, we select the one with the best profit (i.e. the one with minimum reduced cost), choosing among those that have an arrival airport that coincides with the base, and the operator of the crew for which we are generating the route. The route is reconstructed from the selected flight going backward to its predecessor and so on until we find the departure base $l \in N^{bd}$.

At each iteration of the column generation process, we generate one route with the smallest negative reduced cost (if it exists) for each operator $k \in K$ and each departure base $l \in N^{bd}$. It could be possible to add, at each iteration, more than one route for each k and l but we did not find advantage in speeding up the solution process. The process is iterated until no crew route with negative reduced cost exists.

4.2. Exact algorithms

We now describe two approaches for solving the integrated problem to optimality. The first approach is based on model (1)–(12) and is called *path-path method*. The second approach is based on model (13)–(27) and is called *arc-path method*. Both approaches consist of three steps: in the first step a lower bound on the optimal solution cost is obtained; in the second step an upper bound is computed by building a feasible solution to the problem; the third step is devoted to determine an optimal solution by using the generated lower and upper bounds. The latter step is different in the two proposed approaches. In the path-path method a branch-and-price algorithm is developed to obtain an optimal solution. In the arc-path method a reduced MILP model is derived and solved by a general-purpose MILP solver.

First phase: Lower Bound computation. The lower bound is computed by solving the LP relaxation of each model by using column generation on the path-based variables. The column generation for the aircraft routes is similar to the one described in Section 4.1 for the crew routes. Dynamic programming is used also in this case. The procedure is less complex for aircraft routes because we do not have to take into account the duration of the routes nor the number of flights in the routes, i.e. an aircraft has no resources limitation in a workday. Therefore, it suffices to use one label for each flight and no dominance rule needs to be applied. The constraints that need to be taken into account are related to short-term maintenance, i.e. we need to check that the arrival airport of the last flight of a route respects the maintenance rules concerning the departure airport of the first flight of the route, taking into account the exceptions (see Section 2). Constraints on the flight sequencing are directly imposed by the graph definition. As already mentioned, the lower bounds obtained from the two models are the same. However, in Section 5, we have developed the

computation for each model in order to compare the corresponding computational times. We call LB the obtained lower bound on the optimal objective value.

Second phase: Upper Bound computation. An upper bound on the optimal objective value is computed by solving a MILP model which may not contain all the path-based variables but only those generated in the first phase. This reduced MILP model is solved by using a general purpose MILP solver with a given time limit T_{short}. Let UB be the objective value of the best heuristic solution found, if any.

Even if the model with all the variables is feasible, using only the variables generated in the first phase may lead to an infeasible model. Therefore, there is no guarantee that a feasible solution is found in the second phase. This holds both for the path-path and the arc-path models. However, in our computational experiments, the arc-path model was able to find, within T_{short}, feasible solutions with small percentage gaps from LB for all the tested real-world instances. On the contrary, using the path-path model and even time limits larger than T_{short}, no feasible solution was obtained for several real-world instances in our computational experiments. See Section 5.

Third phase: Optimal solution computation. The last phase is devoted to find an optimal solution. Since the percentage gap between the upper and lower bounds is large when using the path-path model and small when using the arc-path model, we have implemented two different approaches. When using the path-path model, a column generation is fundamental, thus motivating the branch-and-price approach that is described in the following. Instead, when using the arc-path model all the variables with reduced cost smaller than the gap between the upper and lower bounds can be generated, thus motivating an alternative approach based on a reduced MILP model that is also now described.

Branch-and-Price approach. Once the lower and the upper bounds have been determined, we apply a branch-and-price scheme to solve the path-path model to optimality.

At each node of the decisional tree, we construct the values of the flows along the arcs in the graph $G^c = (N, A^c)$ for the crews, based on the values of the variables $x_R \in \mathcal{R}$. Then, we select to branch on the node $i \in N^f$ of graph G^c such that it has the highest fractional flow on its outgoing arcs (i.e. the highest number of outgoing arcs with positive flow). We consider a binary branching. In particular, in order to keep the decisional tree balanced, we apply the following branching rule. We order the outgoing arcs (i, j) according to the time instant associated to the departure time of the flight j. Let f be the total value of the flow from node i along its outgoing arcs. In the first generated child, we forbid the use of the first set of outgoing arcs (in the order) such that the sum of their flows is at least $f/2$. In addition, we forbid to use node i as a terminal node for the

route. In the second generated child we forbid the use of the remaining set of outgoing arcs. At each generated child, we forbid to use and to generate routes that contain forbidden arcs.

The upper bound (if any) computed in the second stage is used to fathom unattractive branches of the decisional tree. At each node of the tree, column generation is applied again for the crews when using the arc-path model, and for both the crews and the aircraft when using the path-path model. A depth-first strategy is used in order to dive quickly towards an integer solution. The process is iterated until an optimal solution is obtained.

Reduced MILP model. Once the lower and the upper bounds have been determined, we derive a MILP model from (13)–(27) with the arc-based variables y, the aircraft change variables z, and the crew route variables x such that their reduced cost is smaller than the gap between the upper and lower bounds. In particular, the dynamic programming procedure described above, in which we do not apply the dominance rules, is used in this phase to generate all the x variables in the gap. The y and z variables with reduced cost larger or equal to the gap between the upper and lower bounds are deleted from the model. Note that variables with reduced cost larger or equal to the gap are involved only in solutions with objective value larger than or equal to UB. The obtained model is then solved to optimality by the general purpose MILP solver.

4.3. Bounding cut

This section describes a valid inequality that is fundamental to reduce the computation time taken by the two exact approaches described in the previous section. The inequality is related to the short-maintenance constraints and imposes a minimum number RHS of aircraft changes in a feasible solution:

$$\sum_{(i,j)\in A^c \setminus A^c_s} z_{ij} \geq RHS. \tag{28}$$

As described in Section 2, if we do not consider exceptions on maintenance nor on overnight rest, short-term maintenance requires that each aircraft stays at the depot one night and at a non-depot home base the night after, and each crew returns every night at its home base. Therefore "some" aircraft changes are mandatory. We now propose two procedures to compute RHS.

Basic RHS. Consider the example presented in Figure 2. Let A and B be home bases for the crews and let B be the depot. Suppose that every aircraft must perform a route either from A to B or from B to A, and that every crew must perform a route either from A to A or from B to B. Two aircraft routes are shown as arrows: one departs from A and arrives at B and the other one departs from B and arrives at A. In dotted lines we show three crew routes: two of them go from B to B and one from A to A. We can see that one aircraft change is needed for the crew going from A to A. Indeed the crews go back to their home bases, while the aircraft change base. Note

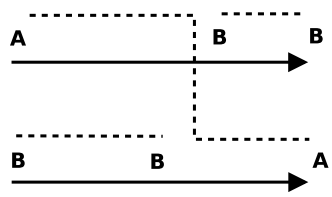

Figure 2 **Example of an aircraft change at B.**

that we do not need two aircraft changes since two crews can finish their duty at an intermediate airport, as it corresponds to their home base.

In general, if we have K aircraft departing from (arriving at, resp.) A and arriving at (departing from, resp.) B, we need (at least) K aircraft changes. Therefore, to compute a lower bound on the number of aircraft changes, we compute a lower bound on the number of aircraft that depart from (arrive at, resp.) a base.

We first count the number of aircraft departing from A or B and the number of aircraft arriving at A or B, i.e. we count the flights from A or B that do not have any possible predecessor (usually they are at the beginning of the day) and the flights to A or B that do not have any possible successor (usually they are at the end of the day). We are sure that if a flight does not have any predecessor (successor resp.), then an aircraft must depart from (arrive at resp.) its departing (arrival resp.) airport. Then, we take the maximum among all the counters (from A, from B, to A, to B) and this gives a lower bound on the number of aircraft changes.

The described procedure directly extends to the case in which there is a set of bases, as long as every aircraft route must go from a base in the set to B or from B to a base in the set. Let us now extend the procedure to deal with the short-term maintenance exceptions. We need to consider the cases in which the aircraft goes from B to B and the cases in which the aircraft goes from a home base different from B to the same home base. Obviously, along these aircraft routes no aircraft changes are needed, since the crew can follow the aircraft along its route and reach its home base. The overnight rest exceptions are dealt with in a similar way.

In conclusion, the value RHS is computed as follows. In a first step, we count the number CD_h of aircraft departing from every base $h \in H$ and the number of CA_h of aircraft arriving at every base $h \in H$. In a second step, we decrease these counters by the number of aircraft routes that can start and end in the same base (i.e., we decrease CD_h and CA_h by $D_{NB} + D_B$). Finally, in a third step, we take the maximum among all the counters and this gives the lower bound on the number of aircraft changes needed in a feasible solution.

Improved RHS. This section describes a better lower bound on the number of aircraft changes. Consider the example of five flights presented in Figure 3. If we apply the counting described in the previous section, we obtain that there are 2 flights (flight 1 and 2) without predecessors departing

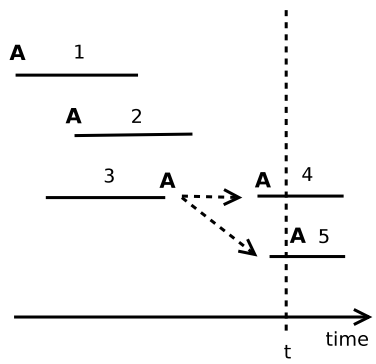

Figure 3 **Example of five flights:** $CD_A = 3$.

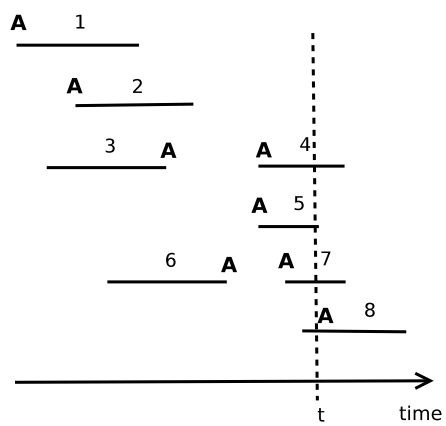

Figure 4 **Example with** $CD_A = 4$ **at time** t.

from A, i.e. $CD_A = 2$. However, there must be three aircraft that depart from A, i.e. $CD_A = 3$. Indeed, at time t, four flights departed from A and just one arrived at A (the aircraft that arrived at A with flight 3 can depart either with flight 4 or with flight 5 but not with both of them). Clearly this increases the lower bound on the number of aircraft changes.

Therefore in general, we want to count the maximum number of aircraft that depart from every base $h \in H$ before a time instant t of the day, for every $t \in \{0, \ldots, 1439\}$. For example, in Figure 4, the maximum number of aircraft departing from A before time t is four, by considering all departures from A and all arrivals at A before time t.

To count the departures from h (with $h \in H$) before a time instant t of the day, for every $t \in \{0, \ldots, 1439\}$, we start from $t = 0$ and we initialize $CD_h = 0$. We consider the time instants of the day in increasing order (from 0 to 1439). When we find a flight departure from h we increase CD_h by one unit. When we find a flight arrival at h we decrease CD_h by one unit. In this way, we know the number of aircraft that need to depart from h before every time instant t, ($t \in \{0, \ldots, 1439\}$). Then we consider the maximum value of CD_h assumed during the computation. The same holds for the arrivals, but in this case one needs to start from the end of the day ($t = 1439$) and to go

Table 1 Details of the instances.

Inst.	#f	#SC	#RC	#OC
1/9	102	19	253	666
2/9	140	35	515	1111
3/9	130	44	450	1046
4/9	124	41	408	960
5/9	124	42	407	958
6/9	128	45	416	1013
7/9	150	40	580	1421
1/4	138	41	560	1185
2/4	132	34	493	1277
3/4	138	31	578	1391
4/4	136	30	564	1372
5/4	144	33	640	1503
6/4	172	46	830	2039
7/4	100	9	282	710

back to $t = 0$. In addition, plane turn time is taken into account, for flight sequencing, both for the departures and for the arrivals.

5. Computational Results

To measure the performance of the proposed approaches we have considered a set of real-world instances given by a major regional airline company flying in Canary Islands. The set of instances corresponds to the flights of the first week of September 2012 and of the first week of April 2012. Table 1 shows the day to identify each instance, its number #f of flights, the number #SC of short connections, and the number #RC of arcs where an aircraft change may occur, and the number #OC of other connections. The RC arcs correspond to potential aircraft changes, and play the role of "restricted connections" introduced in other works (see e.g. Mercier et al. (2005)). These arcs link the aircraft-routing and crew-pairing decisions together, and are critical to the solution process. Comparing the numbers of these arcs in our instances and in e.g. Mercier et al. (2005), we can observe that we have a higher percentage of restricted connections, making the aircraft-change aspect harder in our case.

We have considered the following values to weight the time duration of the connections in crew routes: 2 for the connections between 20 and 30 minutes, 1 for the connections between 30 minutes and one hour, 3 for the connections between one and two hours, 5 for the connections between two and three hours. We have considered the following weights in the objective function: $\alpha = 1$, $\beta_1 = 1000$, $\beta_2 = 950$, $\beta_3 = 910$, $\gamma_k = 10$ $(k \in K)$ for the aircraft routes that respect short-term maintenance rules and $\gamma_k = 20$ $(k \in K)$ for the short-term maintenance exceptions (with $D_B = 2$ and $D_{NB} = 2$), and $\delta = 100$. These parameters were given us by the airline company after performing a sensitivity analysis using our computer code to measure their impact on the generated solutions. This means that the main goal is to minimize the numbers of crews used to serve all the flights, but

also to minimize the number of aircraft changes, the waiting times at the connections for the crews, and the number of aircraft used to serve all the flights. The same weights were used in Salazar-González (2014). The presented algorithms were implemented in C. All the tests were performed on a personal computer with CORE i5-2400 3.10GHZ, 16GB RAM. Cplex 12.4 was used to solve the LP relaxation models and the reduced MILP models. A single thread and default parameter settings were used. We set the time limit T_{short} to two minutes for computing the upper bound during the second phase described in Section 4.2, and a time limit of five hours for computing an optimal solution in the third phase.

Model size. For each instance and for each method, Table 2 reports information on the number of crew and aircraft variables during the solution process when using the bounding cut with the improved RHS value. In particular, for the path-path method the table reports:

- the number (x_R LP) of crew variables generated in the first phase,
- the number (y_R LP) of aircraft variables generated in the first phase,
- the number (x_R B&P) of crew variables generated in the third phase,
- the number (y_R B&P) of aircraft variables generated in the third phase.

For the arc-path method the table reports:

- the number (x_R LP) of crew variables generated in the first phase,
- the number (y_a LP) of aircraft variables (which are all the aircraft variables) included in the first phase,
- the number (x_R Gap) of crew columns generated in the third phase, i.e. the number of columns having reduced cost smaller than the gap between the upper and lower bounds (we indicate with '-' the cases for which the gap was zero),
- the number (y_a Gap) of aircraft arc variables included in the third phase, i.e. the number of columns having reduced cost smaller than the gap between the upper and lower bounds (we indicate with '-' the cases for which the gap was zero).

On the last three columns of Table 2, we report:

- the total number (T x_R) of path-based variables corresponding to feasible crew routes,
- the total number (T y_R) of path-based variables corresponding to feasible aircraft routes,
- the total number (T y_a) of arc-based variables representing aircraft routes.

For the instance of 6/4 our computer was not able to compute the total number of y_R variables due to memory limitation.

We can see that both methods generate a small number of crew-route variables to solve the LP relaxations. The number of crew-routes is almost double when solving the path-path LP relaxation than when solving the arc-path LP relaxation. The number of path-based variables representing

Table 2 Number of crew and aircraft variables in the two models.

		Path-path model				Arc-path model						
Inst.	#f	x_R LP	y_R LP	x_R B&P	y_R B&P	x_R LP	y_a LP	x_R Gap	y_a Gap	T x_R	T y_R	T y_a
1/9	102	1259	1576	2337	7069	654	10128	537	3779	10838	1065025	10128
2/9	140	2195	2773	7592	19383	997	17609	1061	6130	34643	6603705	17609
3/9	130	1731	2229	3575	9203	839	16346	516	4206	30398	5989657	16346
4/9	124	1493	1995	5300	13592	754	15005	827	6281	23048	3843521	15005
5/9	124	1528	2142	5061	14712	811	14985	658	5361	22896	3848204	14985
6/9	128	1674	2274	4243	17123	785	15676	843	5679	25566	4570737	15676
7/9	150	2271	2848	6335	15515	1119	21511	1030	8434	60910	16501692	21511
1/4	138	2578	2873	24519	11349	1092	18892	1647	6882	73120	14196210	18892
2/4	132	1988	2329	2220	3115	908	19071	-	-	63804	18144693	19071
3/4	138	2006	2308	2611	4285	861	21095	5192	18606	84627	23753712	21095
4/4	136	2024	2396	2193	3079	957	20744	-	-	83098	23425413	20744
5/4	144	2190	2631	2317	3330	999	22903	-	-	102195	31424952	22903
6/4	172	3219	3695	3517	4824	1321	30473	-	-	207776		30473
7/4	100	1179	1370	4918	10504	709	10786	2563	9873	15411	1979296	10786

aircraft routes in the path-path LP relaxation is one order of magnitude smaller than the number of arc-based variables in the arc-path LP relaxation. However, the number of path-based variables is still very small with respect to the total number of feasible aircraft-routes, reported in Column T y_R. This may explain why the arc-path method was always able to find feasible solutions and provide good UB values in the second phase.

Clearly the number of aircraft-route variables is huge, and much larger than the number of crew-route variables, since crews have more constraints on the route feasibility.

Comparison between the two exact methods. Table 3 compares the path-path method and the arc-path method on our instances when the bounding cut (28) is not used. Table 4 shows the same comparison when using the bounding cut with the improved RHS values. The columns of these tables show the day to identify each instance, its number #f of flights and, for each method,

- the lower bound LB obtained in the first phase (see Section 4.2),
- the upper bound UB obtained in the second phase (see Section 4.2),
- the percentage gap %Gap between UB and LB,
- the computing times T_{LB} and T_{UB} to compute LB and UB, respectively,
- the best solution value BEST obtained in the third phase (see Section 4.2) within five hours of time limit: we indicate with '$*$' the cases in which the solution was proven to be optimal,
- the percentage gap %Gap between BEST and LB,
- the total computing time T_{Tot} (including T_{LB} and T_{UB}).

TL indicates that the time limit of five hours was reached. In this case, we report the best solution value found. We indicate with '-' the cases in which no solution was obtained within the given time limit. Note that, for some instances, an optimal solution can be found, but it cannot be proven to be the optimal one (in this case, we do not mark the instance with the '$*$').

Table 3 Comparison between path-path and arc-path methods, without bounding cut.

		Path-path method								Arc-path method							
Inst.	#f	LB	UB	%Gap	T_{LB}	T_{UB}	BEST	%Gap	T_{Tot}	LB	UB	%Gap	T_{LB}	T_{UB}	BEST	%Gap	T_{Tot}
1/9	102	21616	23046	6.62	40	120	22228	2.83	TL	21616	22030	1.92	32	120	22028	1.91	TL
2/9	140	27344	-	100.0	231	120	29591	8.22	TL	27344	28096	2.75	173	120	27828	1.77	TL
3/9	130	24673	-	100.0	149	120	28063	13.74	TL	24673	25233	2.27	99	120	25181	2.06	TL
4/9	124	23599	-	100.0	106	120	25749	9.11	TL	23599	24058	1.94	86	120	24018	1.78	TL
5/9	124	23711	-	100.0	106	120	25608	8.00	TL	23711	24155	1.87	86	120	24122	1.73	TL
6/9	128	24383	-	100.0	125	120	27896	14.41	TL	24383	24817	1.78	82	120	24796	1.69	TL
7/9	150	27454	-	100.0	319	120	30360	10.58	TL	27454	28133	2.47	278	120	27970	1.88	TL
1/4	138	26673	-	100.0	292	120	28745	7.77	TL	26673	27470	2.99	238	120	27287	2.30	TL
2/4	132	28389	-	100.0	186	120	31527	11.05	TL	28389	29044	2.31	166	120	28992	2.12	TL
3/4	138	28666	-	100.0	216	120	30613	6.79	TL	28666	29346	2.37	218	120	29277	2.13	TL
4/4	136	28505	-	100.0	214	120	32867	15.30	TL	28505	29316	2.85	195	120	29142	2.23	TL
5/4	144	29377	-	100.0	272	120	34986	19.09	TL	29377	30109	2.49	273	120	29968	2.01	TL
6/4	172	32507	-	100.0	784	120	38079	17.14	TL	32507	33681	3.61	841	120	33112	1.86	TL
7/4	100	20862	-	100.0	37	120	21736	4.19	TL	20862	21390	2.53	39	120	21300	2.10	TL

Table 4 Comparison between path-path and arc-path methods, with the bounding cut and the improved RHS.

		Path-path method								Arc-path method							
Inst.	#f	LB	UB	%Gap	T_{LB}	T_{UB}	BEST	%Gap	T_{Tot}	LB	UB	%Gap	T_{LB}	T_{UB}	BEST	%Gap	T_{Tot}
1/9	102	22008	22107	0.45	42	120	22028*	0.09	2868	22008	22028	0.09	31	22	22028*	0.09	551
2/9	140	27808	-	100.00	218	120	28886	3.88	TL	27808	27828	0.07	148	13	27828*	0.07	315
3/9	130	25169	-	100.00	132	120	25172*	0.01	3228	25169	25174	0.02	108	5	25172*	0.01	136
4/9	124	23997	-	100.00	101	120	24018	0.09	TL	23997	24017	0.08	73	120	24017*	0.08	2088
5/9	124	24101	-	100.00	116	120	24122	0.09	TL	24101	24121	0.08	89	62	24121*	0.08	664
6/9	128	24776	24876	0.40	134	120	24796	0.08	TL	24776	24796	0.08	94	120	24796*	0.08	1777
7/9	150	27950	-	100.00	286	120	28049	0.35	TL	27950	27972	0.08	255	120	27970*	0.07	6589
1/4	138	27267	-	100.00	231	120	27288	0.08	TL	27267	27287	0.07	223	120	27287*	0.07	7068
2/4	132	28985	-	100.00	150	120	28985*	0.00	693	28985	28985	0.00	130	4	28985*	0.00	134
3/4	138	29266	-	100.00	172	120	29266*	0.00	1241	29266	29333	0.23	156	120	29266*	0.00	286
4/4	136	29101	-	100.00	162	120	29101*	0.00	724	29101	29101	0.00	188	6	29101*	0.00	194
5/4	144	29962	-	100.00	207	120	29962*	0.00	842	29962	29962	0.00	223	22	29962*	0.00	245
6/4	172	33103	-	100.00	640	120	33103*	0.00	1983	33103	33103	0.00	653	29	33103*	0.00	682
7/4	100	21256	-	100.00	32	120	21300	0.21	TL	21256	21350	0.44	31	34	21300*	0.21	498

When the bounding cut is not used, the path-path method performed poorly: it was able to find only one feasible solution (instance 1/9), within the given time limit of two minutes.On the contrary, the arc-path method obtained feasible solutions for all the instances with small percentage gaps (close to 2%) over the lower bound. We also performed experiments using a time limit T_{short} of 30 minutes, but still no feasible solution was obtained by the path-path method for most of the instances. We therefore decided to keep the time limit of two minutes. The lower bounds obtained by both methods coincide as expected, and the computing times are comparable. None of the methods was able to determine an optimal solution for any instance within five hours. Interestingly, the two approaches found feasible solutions after the third phase. We can observe that the solutions found by the arc-path method are of much better quality than those obtained by the path-path method.

It is evident that the bounding cut is fundamental for both methods. When using this cut, the path-path method was able to derive seven optimal solutions in the third phase, even if only two feasible solutions were obtained in the second phase. The arc-path method always found feasible

solutions in the second phase with gaps between 0.00% and 0.44% within two minutes of computing time. In addition, it was able to solve to optimality every instance in at most two hours of computing time. This behavior is quite positive from a practical point of view. Clearly, the arc-path method outperforms the path-path method for the considered real-world instances. The weakness of the path-path method is in the number of aircraft variables, even if these variables were dynamically managed by a column generation approach.

It is worth mentioning that the high weight ($\delta = 100$) of the aircraft changes in the objective function makes the solution process harder. In particular, we performed experiments with $\delta = 0$ and all the instances were solved to optimality (even without the bounding cut) in a few minutes, though using a very large number of aircraft changes (up to 35). This behavior may partially explain why the bounding cut is very effective. Indeed, we observed, in our experiments with $\delta = 100$, that the optimal solution of the LP relaxation had $z_{ij} = 0$ for all (i, j) when the bounding cut was not used. In other words, the LP relaxations of the models without the bounding cut have optimal (fractional) solutions with the left-hand side of (28) equal to zero in all the considered instances. While it is not guaranteed that an integer solution with the left-hand side of (28) equal to the RHS exists, this was the case in our experiments when RHS corresponds to the improved value.

Evaluating the impact of the RHS value. Table 5 reports a comparison of the LB values computed in the first phase by the arc-path method when using different RHS values in (28). If the cut is not used (i.e. $RHS = 0$), the LB value is, on average, 1.91% below the optimal solution value. Using the basic RHS reduces the gap to 0.63%, and using the improved RHS gives a gap of 0.05%. Indeed, we found that, with the improved RHS, for five instances, the solution of the LP relaxation was an optimal integer solution, while on the remaining instances the gap was at most 0.2% with respect to the optimal solution value.

We can see that the basic RHS is smaller than the improved RHS for 12 instances. In addition, the improved RHS coincides with the number of aircraft changes in the optimal solution for all the instances in the considered benchmark collection. This result may not occur when using other weights in the objective function or other instances.

Comparison with the heuristic currently in use by the airline company. Table 6 reports a comparison between the optimal solutions obtained by the proposed arc-path method and the solutions generated by the heuristic approach, currently in use by the airline company, described in Salazar-González (2014) (obtained on a similar computer). At the time of writing this paper, this heuristic algorithm is being used by the airline company. Therefore, this comparison highlights the improvement that can be obtained by the proposed method with respect to the current solutions. We used the same initial configuration at the beginning of each day (i.e. the number of aircraft

Table 5 Comparison between the lower bounds obtained with different RHS values.

		$RHS = 0$	Basic RHS		Improved RHS	
Inst.	#f	LB	RHS	LB	RHS	LB
1/9	102	21616	4	22008	4	22008
2/9	140	27344	3	27615	5	27808
3/9	130	24673	2	24871	5	25169
4/9	124	23599	2	23798	4	23997
5/9	124	23711	2	23902	4	24101
6/9	128	24383	2	24579	4	24776
7/9	150	27454	3	27750	5	27950
1/4	138	26673	3	26969	6	27267
2/4	132	28389	5	28885	6	28985
3/4	138	28666	5	29166	6	29266
4/4	136	28505	5	29001	6	29101
5/4	144	29377	5	29862	6	29962
6/4	172	32507	4	32903	6	33103
7/4	100	20862	4	21256	4	21256

available for each operator at each base at the beginning of each duty) and the same weights in the objective function used in Salazar-González (2014) . Table 6 shows the number #a of used aircraft, the number #c of used crews, the number #ch of aircraft changes and the value of the solutions with the corresponding computing times (expressed in seconds) generated by the arc-path algorithm and by the heuristic approach presented in Salazar-González (2014). In the last column we show the percentage gap %Gap between the solution values. The results show that the solutions obtained by the heuristic approach presented in Salazar-González (2014) are significantly improved by the proposed method. Indeed, the approach presented in Salazar-González (2014) is not able to find any optimal solution while the arc-path approach is able to generate optimal solutions for all the benchmark instances. Observing the objective function values, the exact method obtains a significant improvement (between 4.37% and 23.56%) on all instances. As for the computational time, the heuristic approach presented in Salazar-González (2014) finds solutions in around a couple of minutes, while the proposed algorithm requires longer computing times to provide the optimal solutions. However, at the end of the second phase of the arc-path method, feasible solutions of better quality (see Column UB in Table 4) are obtained within a similar computational time (see Columns T_{LB} and T_{UB} in Table 4). The third phase proves that the heuristic solutions found in the second phase are optimal for 10 instances, while, in the remaining instances, the gap between UB and BEST is smaller than 0.3%. Therefore, the proposed arc-path method can be used as a heuristic algorithm to obtain, in short computing times, better solutions than those used by the airline company.

Table 6 shows that the arc-path method always reduces either the number of crews or the number of aircraft changes or both, which are the two main goals in practice. In addition, the arc-path

Table 6 Comparison of the arc-path method with the heuristic proposed in Salazar-González (2014), currently in use by the airline company.

		Arc-path method					Salazar-González (2014)					
Inst.	#f	#a	#c	#ch	Opt Value	Time	#a	#c	#ch	Heu Value	Time	%Gap
1/9	102	12	22	4	22028	551	14	24	8	24600	20	11.68
2/9	140	15	28	5	27828	315	17	28	7	29045	101	4.37
3/9	130	14	25	5	25172	136	15	26	6	26818	65	6.54
4/9	124	13	24	4	24017	2088	13	25	5	25450	55	5.97
5/9	124	13	24	4	24121	664	14	25	4	25466	53	5.58
6/9	128	13	25	4	24796	1777	13	26	4	26368	63	6.34
7/9	150	15	28	5	27970	6589	15	30	5	30127	135	7.71
1/4	138	17	27	6	27287	7068	17	28	7	29163	121	6.88
2/4	132	16	29	6	28985	134	16	31	6	31467	72	8.56
3/4	138	16	29	6	29266	286	16	31	6	32273	131	10.27
4/4	136	16	29	6	29101	194	16	35	6	35956	96	23.56
5/4	144	16	30	6	29962	245	16	31	6	31798	145	6.13
6/4	172	17	33	6	33103	682	17	36	8	36341	242	9.78
7/4	100	12	21	4	21300	498	11	24	4	24622	24	15.60

method reduces, in four instances, the number of aircraft involved in a solution. Only for instance 7/4 the solution found by the arc-path method uses an additional aircraft (but spares three crews). The computational times of the exact approach are larger than the ones of the heuristic algorithm but, being 30 minutes on average and less than 2 hours in the worst case, they are, according to the company, acceptable for practical purposes.

6. Conclusions

This paper studies an integrated fleet assignment, aircraft routing and crew pairing problem of a regional airline company. One characteristic of this company is that no flights are scheduled during the night, so the problem can be solved on a daily basis. Another special feature of the problem is that there are two home bases for the crews, but only one is a depot for the aircraft. Due to short-term aircraft maintenances, each aircraft must be one night at the depot and the next night outside it (exceptions to these rules are imposed by the company and taken into account by the proposed approaches). On the contrary, the crews should return to their home bases to avoid extra costs to the company due to overnight rests. Therefore a crew may need to operate different aircraft during its duty, and this is undesired because it can propagate delays. The problem is to assign fleets, crews and aircraft to flights in order to minimize the aircraft and crew costs and the number of aircraft changes. Although these features are specific of the considered regional carrier, we think that other regional carriers may have the same (or very similar) requirements, making the proposed approaches suitable to solve their problems.

This paper presents two MILP formulations. In both formulations the crew pairings are modeled through path-based variables. The aircraft routes are modeled through path-based variables in

the first formulation (path-path model) and through arc-flow variables in the second formulation (arc-path model). The paper proposes two exact algorithms, each one based on one of these models.

To speed up the performance of the algorithms, a bounding cut is proposed, based on computing a lower bound on the minimum number of aircraft changes needed in a feasible solution. Computational results show that the arc-path method outperforms the path-path method, and is able to solve to optimality all the considered real-world instances, involving up to 172 flights, in reasonable computing times (at most 2 hours), leading to a significant improvement over the solutions currently used by the company.

Acknowledgments

We thank the airline company "Binter Canarias" for motivating this research and for providing us with the instances for our computational experiments. This work has been partially supported by the Spanish Government through the research project MTM2012-36163-C06-01.

References

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

Barnhart, C., L. Fang, R. Shenoi. 1998b. Integrated airline schedule planning. *Operations Research in the airline industry*. International Series in Operations Research & Management Science 9, 384–403.

Belobaba, P., A. Odoni, C. Barnhart. 2009. *The Global Airline Industry*. John Wiley & Sons, West Sussex, United Kingdom.

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

Cook, W.J., W.H. Cunningham, W.R. Pulleyblank, A. Schrijver. 1998. *Combinatorial optimization*. John Wiley and Sons, New York.

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

Dunbar, M., G. Froyland, C.L. Wu. 2012. Robust airline schedule planning: Minimizing propagated delay in an integrated routing and crewing framework. *Transportation Science* **46**(2) 204–216.

Froyland, G., S.J. Maher, C.L. Wu. 2014. The recoverable robust tail assignment problem. *Transportation Science* to appear.

Gao, C., E.L. Johnson, B.C. Smith. 2009. Integrated airline fleet and crew robust planning. *Transportation Science* **43**(1) 2–16.

Haouari, M., H.D. Sherali, F.Z. Mansour, N. Aissaoui. 2011. Exact approaches for integrated aircraft fleeting and routing at tunisair. *Computational Optimization and Applications* **49**(2) 213–239.

Klabjan, D. 2005. Large-scale models in the airline industry. G. Desaulniers, J. Desrosiers, M.M. Solomon, eds., *Column generation*. Kluwer, 163–196.

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

Lan, S., J.P. Clarke, C. Barnhart. 2006. Planning for robust airline operations: Optimizing aircraft routings and flight departure times to minimize passenger disruptions. *Transportation Science* **40**(1) 15–28.

Liang, Z., W.A. Chaovalitwongse. 2013. A network-based model for the integrated weekly aircraft maintenance routing and fleet assignment problem. *Transportation Science* **47**(4) 455–476.

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

Magnanti, T., R. Wong. 1981. Accelerating benders decomposition: algorithmic enhancement and model selection criteria. *Operations Research* **29**(3) 464–484.

Mercier, A. 2008. A theoretical comparison of feasibility cuts for the integrated aircraft-routing and crew-pairing problem. *Transportation Science* **42**(1) 87–104.

Mercier, A., J.F. Cordeau, F. Soumis. 2005. A computational study of benders decomposition for the integrated aircraft routing and crew scheduling problem. *Computers & Operations Research* **32**(6) 1452–1476.

Mercier, A., F. Soumis. 2007. An integrated aircraft routing, crew scheduling and flight retiming model. *Computers & Operations Research* **34**(8) 2251–2265.

Papadakos, N. 2009. Integrated airline scheduling. *Computers & Operations Research* **36**(1) 176–195.

Pita, J.P., C. Barnhart, A.P. Antunes. 2013. Integrated flight scheduling and fleet assignment under airport congestion. *Transportation Science* **47**(4) 477–492.

Salazar-González, J.J. 2014. Approaches to solve the fleet-assignment, aircraft-routing, crew-pairing and crew-rostering problems of a regional carrier. *Omega* **43** 71–82.

Sandhu, R., D. Klabjan. 2007. Integrated airline fleeting and crew-pairing decisions. *Operations Research* **55**(3) 439–456.

Sherali, H.D., K.H. Bae, M. Haouari. 2010. Integrated airline schedule design and fleet assignment: Polyhedral analysis and benders' decomposition approach. *INFORMS Journal on Computing* **22**(4) 500–513.

Sherali, H.D., K.H. Bae, M. Haouari. 2013a. A benders decomposition approach for an integrated airline schedule design and fleet assignment problem with flight retiming, schedule balance, and demand recapture. *Annals of Operations Research* **210**(1) 213–244.

Sherali, H.D., K.H. Bae, M. Haouari. 2013b. An integrated approach for airline flight selection and timing, fleet assignment, and aircraft routing. *Transportation Science* **47**(4) 455–476.

Weide, O., D. Ryan, M. Ehrgott. 2010. An iterative approach to robust and integrated aircraft routing and crew scheduling. *Computers & Operations Research* **37**(5) 833–844.

Yu, G., ed. 1998. *Operations Research in the airline industry*. Kluwer Academic Publishers, Boston, MA.

Zeghal, F.Z., M. Haouari, H.D. Sherali, N. Aissaoui. 2011. Flexible aircraft fleeting and routing at tunisair. *Journal of the Operational Research Society* **62**(2) 368–380.