



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

ARCHIVIO ISTITUZIONALE
DELLA RICERCA

Alma Mater Studiorum Università di Bologna
Archivio istituzionale della ricerca

Heuristic approaches for flight retiming in an integrated airline scheduling problem of a regional carrier

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

Published Version:

Cacchiani, V., Salazar-González, J. (2020). Heuristic approaches for flight retiming in an integrated airline scheduling problem of a regional carrier. OMEGA, 91, 1-16 [10.1016/j.omega.2019.01.006].

Availability:

This version is available at: <https://hdl.handle.net/11585/717094> since: 2024-02-24

Published:

DOI: <http://doi.org/10.1016/j.omega.2019.01.006>

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)

Heuristic Approaches for Flight Retiming in an Integrated Airline Scheduling Problem of a Regional Carrier

Valentina Cacchiani,
Juan-José Salazar-González

the date of receipt and acceptance should be inserted later

Abstract Flight retiming in airline scheduling consists in slightly modifying the scheduled departure time of some flights with the goal of providing a better service with a cheaper cost. In this research, the departure times must be selected from a small discrete set of options. The whole problem embeds flight retiming, fleet assignment, aircraft routing and crew pairing. Thus, the aim is to determine the departure times of the flights, the fleet assignment and the minimum cost aircraft and crew routes. The objective function takes into account a large cost associated with each crew member, a penalization for short or long connection times, a cost for crew members changing aircraft along their routes, and a minor penalty associated with the use of each aircraft. The constraints enforce aircraft maintenance and crew working rules. In this setting, flight retiming is allowed to potentially reduce the total costs and increase the robustness of the solution against delays by decreasing the number of aircraft changes.

We propose and compare four heuristic algorithms based on a Mixed Integer Linear Programming model for the whole problem. The model contains path variables representing the crew pairings, and arc variables representing the aircraft routes. In the heuristic algorithms, column generation is applied on the path variables, and different flight retiming options are considered. The algorithms are tested on real-world instances of a regional carrier flying in the Canary Islands to evaluate their advantages and drawbacks. In particular, one of the algorithms, that uses the solution of the Linear Programming relaxation of the model to select promising options for the departure of the flights, turns out to be the most effective one. The obtained results show that costs can be significantly reduced through flight retiming while still keeping the computing times reasonably short. In addition, we perform a sensitivity analysis by including more retiming options and by using different aircraft and crew costs. Finally, we report the results on larger size instances obtained by combining real-world ones.

This work has been partially supported by the research projects ProID2017010132 (Gobierno de Canarias) and MTM2015-63680-R (MINECO/FEDER).

Valentina Cacchiani
DEI, University of Bologna, Viale Risorgimento 2, I-40136 Bologna, Italy,
E-mail: valentina.cacchiani@unibo.it

Juan-José Salazar-González
DMEIO, Universidad de La Laguna, 38200 Tenerife, Spain,
E-mail: jjsalaza@ull.es

Keywords Integrated airline scheduling, aircraft routing, crew scheduling, flight retiming, heuristic algorithm, column generation

1 Introduction

Airline scheduling is a very complex problem that is usually solved by decomposing it into several planning stages ([6], [7]): flight scheduling, in which origin and destination airports and departure and arrival times have to be determined for each flight based on the passenger demand; fleet assignment that consists of deciding which fleet (aircraft type) must be assigned to each flight; aircraft routing, in which the routes for the aircraft are designed while satisfying maintenance constraints; crew pairing, in which the routes for the crew members are defined while respecting the labor work rules; crew rostering that consists of finding a roster for each crew member. Each of these stages is usually characterized by an objective function that minimizes the costs appearing in that stage. In addition, the output of a stage becomes the input of the subsequent stage (see e.g. [19]). Therefore, suboptimal solutions might be obtained due to the problem decomposition. To overcome this issue, recent works integrate different stages and solve the integrated problem at once (see e.g. [3], [5], [9], [10], [12], [13], [18], [20], [21]).

Flight retiming consists of selecting one departure time (and consequently the arrival time) for each flight among a set of options. In some airlines, like the one motivating our research, the commercial department initially suggests a departure time for each flight, but a small modification (say ± 10 minutes) may be allowed if convenient. Each flight has fixed duration, departure airport and arrival airport. When planning in advance, the suggested departure time can be anticipated or postponed within a discrete set of options, typically a small time interval. Flight retiming can be used both to reduce the airline scheduling costs and to increase the robustness of the planned solution against possible delays, and therefore, it can be viewed also as a simple variant of flight scheduling.

In this work, we include the feature of flight retiming in an integrated airline scheduling problem combining fleet assignment, aircraft routing and crew pairing. The whole problem requires to determine the departure times (chosen inside a small set of options) for the flights, the fleet assignment and the minimum cost aircraft and crew routes. The objective function of the problem takes into account a large cost associated with each crew member, including penalties for short and long connection times between consecutive flights, a cost for the aircraft changes (since they increase the probability of delay propagation), and a cost associated with the use of each aircraft. In this setting, we apply flight retiming to possibly reduce the total costs, while satisfying constraints on aircraft maintenance and crew working rules. Note that costs not only aim at the efficiency of the schedule, but also include a penalty for short connections and a term for minimizing the aircraft changes, thus favoring robustness of the planned schedule. Flight retiming allows sequencing flights that otherwise would be too close or too far in time, i.e., it increases the flight connection possibilities. Thus, by using retiming, the number of crew members and/or aircraft needed to execute all flights can be reduced, and the number of aircraft changes and the cost for short or long connection can be decreased too. The studied problem arises at a regional carrier flying in the Canary Islands. In [9], we studied the problem without flight retiming for the same carrier, and provided several exact approaches to solve it. The best approach is based on a Mixed Integer Linear Programming (MILP) model with path (crew-route) variables representing the crew pairings and arc variables representing the aircraft routing. The approach was composed of three phases. In the first phase, the Linear Programming (LP) relaxation of the MILP model was solved to obtain

a lower bound by applying column generation on the crew-route variables, where the pricing problem was an Elementary Shortest Path with Resource Constraints (ESPRC) solved by a dynamic programming procedure on an acyclic graph. The second phase was used to find an integer solution (upper bound), by solving, within a given short time limit, the reduced MILP model containing only the crew-route variables generated in the first phase and all the arc aircraft variables. The third phase was used to derive the optimal solution by solving a MILP model including all variables with reduced cost smaller than the gap between the upper and lower bounds computed in the previous phases. The three-phase approach was enhanced with a bounding cut derived by computing a lower bound on the number of aircraft changes that are needed in a feasible solution. Computational experiments showed that the approach was able to solve to optimality all the real-world instances. The success was partially due to the size of the instances, each one involving about 150 flights, with no interaction between consecutive days because the carrier does not operate during night time. Those excellent results motivated the extension of the problem to also address the retiming characteristic. Indeed, the commercial department of the carrier initially proposes the flights with given departure times, but the regional carrier is willing to slightly move (anticipate or postpone) those suggested times if a better solution with respect to the one without retiming is generated.

Our first attempt to solve the problem with flight retiming was to adapt the exact approach in [9] by considering a flight for each alternative departure time. However, the number of available flight connections significantly increases when flight retiming is allowed, thus making the problem much harder to be solved. To reduce the computing times, we propose and evaluate four heuristic algorithms, all based on the MILP model with crew-route variables and aircraft-arc variables.

The main contributions of this paper are as follows:

- The paper tackles a real-world integrated airline scheduling problem that includes the feature of flight retiming. The problem requires to determine the departure times (chosen inside a small set of options) for the flights, the fleet assignment and the minimum cost aircraft and crew routes. While most of the articles on flight retiming in the literature consider one phase (aircraft routing or fleet assignment) or two phases (aircraft routing and crew pairing), our approaches solve the three phases and flight retiming all together.
- The problem is of interest to regional carriers, characterized by many short flights between a small set of airports and by a large number of flight connections. These features make the flight retiming opportunity very challenging. In addition, the problem includes contrasting goals: the objective function contains costs aiming at the efficiency of the schedule (e.g. aircraft and crew costs), and costs for minimizing the number of aircraft changes, thus maximizing also the robustness of the planned schedule.
- The paper describes four two-phase algorithms for solving heuristically the integrated airline scheduling problem. The algorithms are based on a MILP model with route variables representing the crew pairings and arc variables representing the aircraft routing. Column generation is applied on the crew-route variables, and different procedures for selecting the retiming options are considered. These algorithms rely on the MILP model and column generation developed in [9] for the variant without retiming, but incorporate procedures for appropriately selecting flight copies. As shown in the computational results, finding a good set of flight copies is not trivial, but important for limiting the computing times while still obtaining improved solutions.
- Computer implementations of the algorithms are tested on real-world instances of the regional carrier. The results show the advantages and drawbacks of the different algorithms,

with a better trade-off between solution quality and computing time of one algorithm with respect to the other three.

- We perform a sensitivity analysis on the impact that a larger number of retiming options has on the quality of the obtained solutions and on the computing times. In addition, we consider different aircraft and crew costs in the objective function, and evaluate their effects on the obtained airline schedules. Finally, we test the performance of the most effective algorithm on larger size instances, showing that they can effectively be solved in acceptable computing times.

The paper is organized as follows. Section 2 gives an overview of the works from the literature that apply flight retiming. Section 3 describes formally the studied problem, emphasizing the main features that characterize regional carriers. Section 4 presents the MILP model by [9] adapted to deal with flight retiming, and Section 5 describes four heuristic approaches. Section 6 reports the computational results obtained on real-world instances of the regional carrier, showing the improvement that can be achieved by flight retiming. It also reports our sensitivity analysis and the computational results on larger size instances. Finally, we conclude with some remarks and ideas for future research in Section 7.

2 Literature Overview

Due to the wide research in the field of airline scheduling, we focus on related works that consider flight retiming in the planning phase to increase either the robustness against delays or the profits/reduce costs, even in non-integrated settings. Retiming is also used for disruption management (see, e.g., the literature review in [24]), but our problem occurs in the planning phase afforded about six months before the operational phase. The planning phase includes our problem in a first step and rostering problems (for crew and aircraft) in a second step. The operational phase deals with disruption issues.

2.1 Robustness Increase

To our knowledge, all the works that use retiming in order to increase the robustness of the schedule against delays, except for [11], focus on the aircraft-routing stage, rather than on an integrated problem. The article [4] describes a Mixed Integer Programming (MIP) model that aims at determining flight departure times and building robust aircraft routes. The goal is to insert buffer times between connecting flights so as to increase the protection against delays. A Monte Carlo simulation study is used to show the robustness of the derived plans. Reliability and flexibility of the schedules are improved in [8], by a memetic algorithm that applies flight retiming and aircraft rerouting. An extensive simulation study is performed to evaluate the effect of these robustness objectives on the operational performance of the schedules. The article [16] proposes two alternative approaches to minimize passenger disruptions through robust plans: the first one considers aircraft routing, while the second involves retiming flight departure times within a small time window to reduce the number of passengers that miss their connection. A two-level method is proposed in [2] to obtain robust schedules by using buffer times. In the first level, two Mixed Integer Quadratic Programming (MIQP) models are iteratively solved: the first one is used to generate aircraft routes and lower bounds on aircraft connection buffer times, with the goal of maximizing the robustness of aircraft connections; the second one is used to compute the departure time of each flight, with the goal of maximizing the passenger connection robustness. To evaluate the robustness of the solutions derived by

the MIQP models, the Monte Carlo simulation procedure of [4] is applied. The second level consists of an evolutionary algorithm to improve the solution derived in the first level. The problem of determining robust weekly schedules for a fleet of aircraft while taking into account maintenance constraints is studied in [1]. Flight retiming is applied in order to derive schedules that are less sensitive to delays. A compact Mixed Integer Non-Linear Programming (MINLP) model is proposed and reformulated as a MILP model: in this first stage, the departure time of each flight is set at the earliest departure time. Then, a second stage is applied in which flights can be retimed with the goal of improving the robustness of the solution: this is obtained by a heuristic algorithm that iteratively applies a Monte Carlo simulation procedure for estimating the delays, and a retiming algorithm for increasing the buffer times where it is more useful.

The aircraft routing and crew pairing are combined in [11], in an iterative framework, to determine the dependencies that affect the propagation of delays: both problems are modelled by using route variables, and are solved separately by column generation and label setting pricing algorithms. Then, the propagated delay is computed along each aircraft route (crew route, resp.), taking into account propagated delays from crew (aircraft, resp.). Based on these delays, crew and aircraft route costs are updated, and the aircraft routing and crew pairing problems are solved again. Flight retiming is applied, together with the iterative aircraft routing and crew pairing algorithm, to provide more slack over critical connections, but flights are retimed without altering the aircraft and crew assignments of the incumbent solution. In addition, to further improve the solution method, an exact approach and a local-search procedure are developed to incorporate delay scenarios in the aircraft routing and crew pairing subproblems. Both methods incorporate stochastic delay information by considering a set of primary delay values for each connection in the network. In particular, the exact method consists of enumerating all feasible aircraft and crew paths in the aircraft and crew subproblems, and then calculating, for each path, the average delay propagation along the path over all delay scenarios. The heuristic method computes the average delay propagation at each node in the label setting algorithm, and then, based on this delay, chooses which labels to propagate. The two methods are used within the iterative framework, i.e., the exact or heuristic methods are used to solve the aircraft and crew pricing subproblems. Finally, flight retiming is integrated in the iterative framework with the delay scenarios, by considering, for the aircraft routing problem, a network that contains multiple copies for each flight, while it is assumed that the departure times chosen for the flights are also followed by the crew.

2.2 Profit Increase / Cost Reduction

Flight retiming is often used in combination with fleet assignment to improve the airline company profits by increasing the connection possibilities for the passengers and by adapting flights to the passenger demand that changes over time. A dynamic scheduling approach is proposed in [14]. It reoptimizes the departure times of the flights and the fleet assignment, during the passenger booking process, at regular intervals, based on booking data and improved forecasts, in order to avoid empty seats or lack of seats in the flights. The goal is to improve the total profits. The solution process consists of solving two MILP models: the first one is a passenger mix model to find the revenue maximizing assignment of passengers to itineraries based on passenger demand and with capacity constraints on the number of available seats; the second model is a reoptimization model that considers flight copies for each flight and decides the fleet assignment and flight retiming that maximizes the revenue minus the operating costs, while taking into account passenger demand, fleet capacity, maximum number of aircraft arrivals and departures for each airport, and service to previously booked passengers. A MILP

model is proposed in [23] and [22] to integrate flight scheduling and fleet assignment, and it includes additional features such as path/itinerary-based demands, flexible flight times, schedule balance, recapture issues, and multiple fare-classes. Valid inequalities are generated through a polyhedral analysis, and a Benders decomposition solution approach is applied. The goal is to maximize the net profit as given by the revenue minus the cost, while considering multiple fare-classes in each path/itinerary. Flight retiming helps to determine new connecting itineraries for serving certain markets, and thereby improve profits by appropriately assigning fleet to flights.

Aircraft routing and crew pairing problems have been integrated in a single stage and combined with the possibility of flight retiming in [15] and [17] with the goal of reducing the airline costs for performing all flights. The former work exploits a partial integration of the two problems by including plane-count constraints (that count the number of available aircraft on the ground at any time) to the crew scheduling problem, and additionally allowing, for each flight, the departure time to be chosen within a given time window. A subset of pairings based on the original schedule are generated, and the crew scheduling model with plane-count constraints and only the generated pairings is solved: in this case, the plane-count constraints are approximated because flight retiming makes it hard to model them exactly. In [17], aircraft routing and crew pairing problems are integrated in a single ILP model and flight retiming is allowed to reduce the costs. Both aircraft routing and crew pairing are modelled using path variables on a time-space networks, and a discrete set of departure times is given for each flight. Constraints on aircraft maintenance, and total work time, flight time and number of landings for the crew are imposed in the definition of the paths. Constraints on the flight covering, maximum number of available aircraft and crew are imposed in the model. In addition, deadhead flights, i.e. flights where crew members travel as passengers, are allowed, and it is imposed that the same flight schedule is chosen for the working crew and for the travelling crew. Similarly, it is guaranteed that for each flight the same schedule is chosen for the aircraft and the crew. Additional constraints require that on short connections aircraft changes are not allowed. The ILP model is then reformulated by imposing the latter constraints in an aggregated form, i.e. they are imposed on every short connections without specifying the departure time chosen by the pairs of connecting flights. A three-phase heuristic algorithm is proposed, based on Benders decomposition of the ILP model where the crew pairing is kept in the master problem. The first phase consists of solving the LP-relaxation of the ILP reformulated model by Benders decomposition and column generation. During this phase, constraints on deadhead flight schedule and on detailed short connection are dynamically generated. In the second phase, crew variables are imposed to be integer and the model is solved by generating Benders cuts. Finally, in the third phase, integrality constraints are imposed also on the aircraft variables and the resulting model is solved.

In our work, we consider both goals of robustness increase and cost reduction in a weighted linear objective function, while these contrasting goals have usually been studied separately in the literature. The weights given to the different terms are the same used in [9] and give a high importance to the minimization of the crew costs, and next to the minimization of the number of aircraft changes (when the crew member needs to change aircraft), thus including, in a simplified way, the goal of robustness. In addition, we contribute to the existing literature by adding fleet assignment (i.e. we consider the three operators in which the airline company is divided) in the integrated problem together with aircraft routing and crew pairing. Moreover, the studied application arises at a regional carrier, and has some specific features that make the problem quite different from the existing literature. The studied problem and its features are described in Section 3.

3 Problem Description

We address a daily fleet assignment, aircraft routing and crew pairing problem. A set F of flights is given. Each flight $f \in F$ is characterized by origin and destination airports, and departure and arrival times. No flight is scheduled during the night. Flights in F are called *original flights*. In addition, for each flight $f \in F$, we are given a discrete set of alternative departure times, corresponding to the retiming possibilities. For example, in the context motivating our research, the set contains the departure time of the original flight and two other options corresponding to the original departure time plus and minus 10 minutes. We call C^f the set of copies of flight $f \in F$, and $C = \cup_{f \in F} C^f$ the set of all *flight copies*.

The regional carrier motivating our research consists of three operators (Binter, Naysa, Canair), each one owning some aircraft and managing some crew members. The set of operators is denoted by K . Even though all aircraft are identical, crew members can only operate aircraft of their operator. Crew members of different operators have different salaries and different work rules. In particular, constraints are imposed on the maximum number of flights in a crew route and on its duration. These constraints depend on the departure time of the first flight of the route and on the operator. Moreover, some flights can be executed only by some of the operators (e.g., flights outside Canary Islands can be operated by low-cost operations only). An aircraft cannot fly unless operating a flight, and a crew member cannot be planned to travel as a passenger.

The regional carrier operates flights between eleven airports, denoted by M , among which

- $M^B \subseteq M$ are bases, i.e. airports where aircraft and crew members can stay during the night,
- $H \subseteq M^B$ are home bases, i.e. hub airports where a crew route can start and end without any overnight cost to the company, and
- $B \subseteq H$ are the airports with maintenance facilities. In our real-world application, it is a singleton set containing the depot (Las Palmas de Gran Canaria); with abuse of notation, we will identify the depot as B .

The numbers n_a^{kl} of available aircraft and n_c^{kl} of available crew members of each operator $k \in K$ at the beginning of the day at each base $l \in M^B$ are given.

Each aircraft needs a long-term maintenance which takes it out of service during two consecutive weeks each year. The two weeks are decided in another phase by the carrier, and it is known when solving our problem. Instead our problem must also determine the time for the short-term maintenance of each aircraft, which are checking operations to undergo every 2 days where the aircraft was used. Since no flight departure is scheduled during the night (from 22:00 to 07:00), aircraft short-term maintenance is always performed at that time. Since the depot B can check half of the fleet each night, the carrier imposes that each aircraft stays one night at B , and the next night at a different airport $M^B \setminus B$. However, it is allowed that up to D_{NB} (equal to 2 in our real-world application) aircraft can stay two consecutive nights at $H \setminus B$, and it is also allowed that up to D_B (equal to 2 in our real-world application) aircraft can stay two consecutive nights at B . These special cases are called *short-term maintenance exceptions*.

Due to budget limitation, the company requires that every crew member ends its duty at its home base in H , thus avoiding overnight rest outside his/her house. However, a few exceptions are possible when the overnight rest outside the home base is unavoidable, and occurs in $M^B \setminus H$ (in our application, it occurs in two airports: La Palma and Arrecife). These special cases are called *overnight rest exceptions*, and are only possible for some (low-cost) operators.

The flight sequencing rules are as follows. The plane-turn time is 20 minutes, and a connection is *short* when smaller than 30 minutes. Two flights can be performed in sequence by the same crew member if the connection time is at least 20 minutes and at most a maximum allowed connection time (3 hours in our application). On short connections, aircraft changes are not allowed. On all other connections, aircraft changes are penalized.

The objective function takes into account:

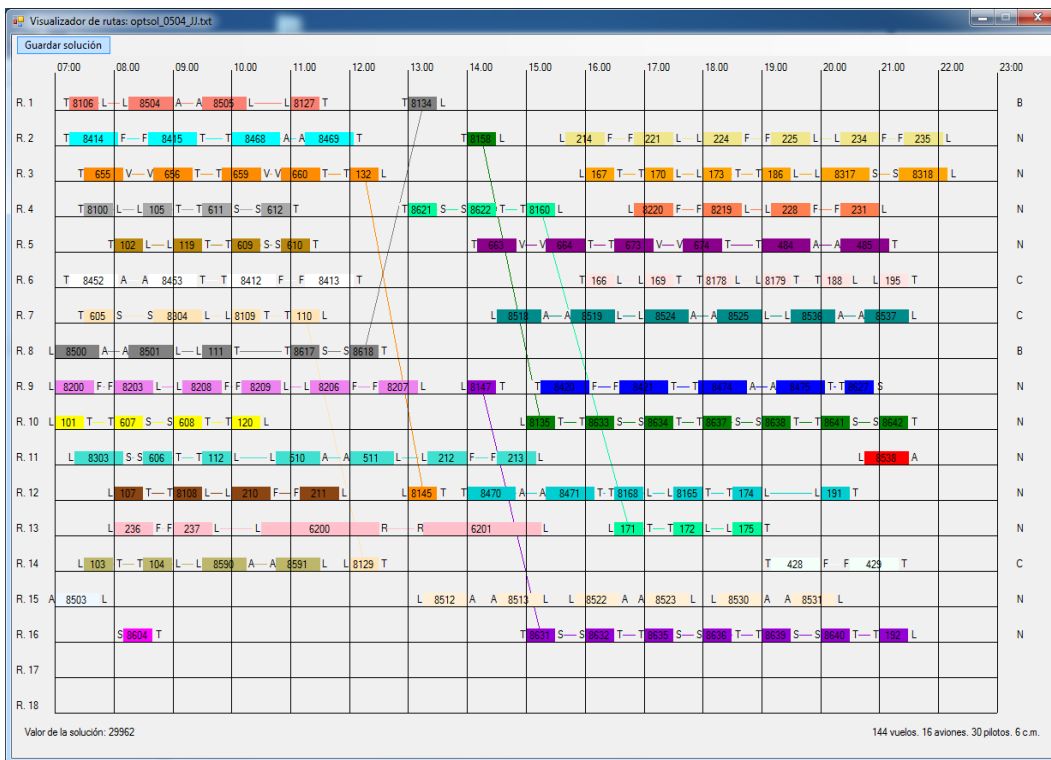
1. A cost associated with each crew route: it is based on the connection times between consecutive flights in the route. In particular, long and short connection times are penalized. Long connection times should be avoided because (for the regional carrier motivating our work) the crew cost increases as a monotone piecewise linear function of the time spent without flying. On the other hand, short connection times often cause delay propagation.
2. A cost associated with the number of crew routes, as the crew salary (which depends on the operator) is augmented each day the crew is operating.
3. A cost associated with the number of aircraft routes: even if all aircraft belongs to the carrier, it is desired to minimize the number of aircraft needed in the solution in order to spare some aircraft that could substitute another aircraft in case of disruptions.
4. A cost associated with each aircraft change in order to find solutions that can be more robust against delays.

As in [9], we reduce the multi-criteria nature of the problem to a single linear objective by introducing weights: α weights the sum of the connection times in the crew routes, β_k the number of crew routes of operator $k \in K$, γ_k the number of aircraft routes of operator $k \in K$, and δ the number of aircraft changes in the crew routes.

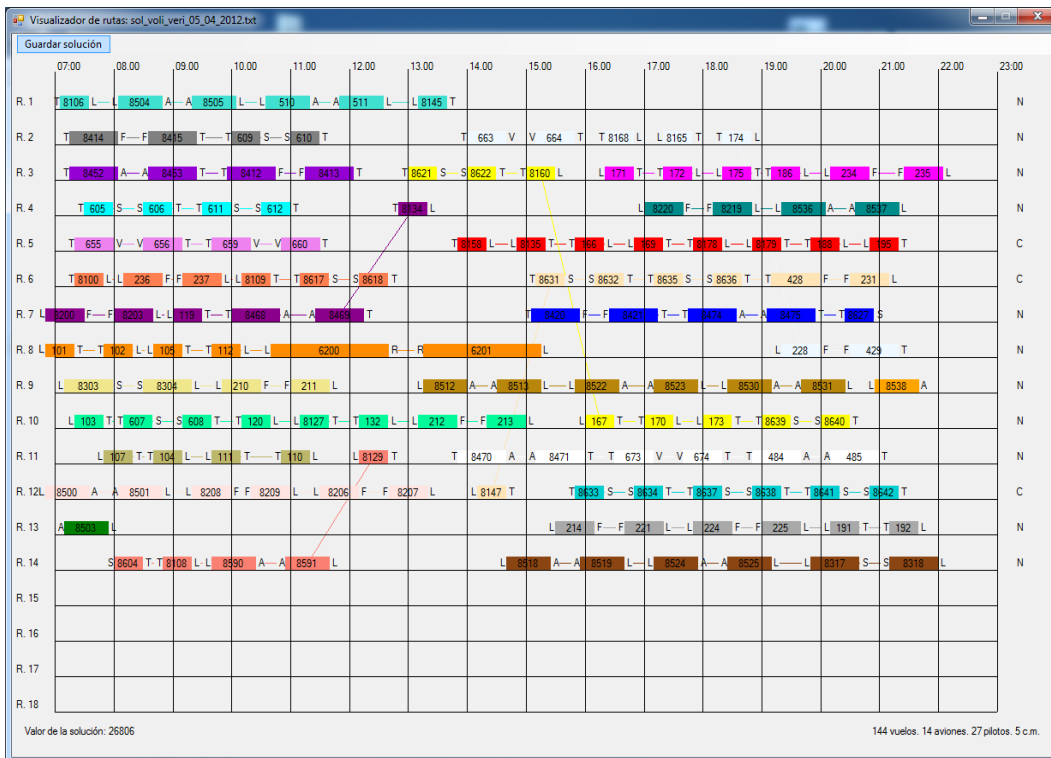
As mentioned above, regional carriers present some special features with respect to major airline companies:

1. Even though no flight is performed during the night, during the day many short flights connect the airports (located on islands in our application). Thus there exists a very large number of possible connections and alternative options for crew and aircraft routes, which makes the problem even more challenging when it also includes flight retiming.
2. Most crew and aircraft routes contain several (often 6 to 8) flights, since the minimum connection time is short (20 minutes) and the flight time is also short compared to major airlines considering intercontinental flights.
3. Only one airport (the depot B) can host maintenance, and no overnight rest outside the home base of the crew is allowed for the crew members (with very few exceptions, as mentioned above). Hence, some aircraft changes are needed, making the solution less robust against delays (a late arrival of the flight before the aircraft change may cause late departures of the two flights operated by the same aircraft and crew, respectively, after the aircraft change, and so on).
4. Since small aircraft (ATR 72 in our application) are used by the regional carrier, weather conditions (e.g. fog) can deeply affect the planned schedule, making robustness a critical issue. For that reason, one of the main goals of the regional carrier is to minimize the number of aircraft changes. In our case, this means that δ is a large number.

We conclude the description of the problem with two images in Figure 1 to illustrate how the solutions look like. Images (a) and (b) are solutions of the problem without and with retiming, respectively. In both cases, the data correspond to 144 flights on Thursday 5 April 2012. The boxes represent flights, the horizontal lines represent aircraft, and colors are used for crew routes. Non-horizontal lines in color depicts aircraft changes, which is one of the major objective to be minimized in the problem. The letters on the left and right sides of each flight



(a) Optimal solution of the problem without retiming



(b) Feasible solution of the problem with retiming

Fig. 1 Instance example (5 April 2012)

represent the departure and arrival airports, respectively (T for TFN, L for LPA, V for VDE, S for SPC, A for ACE, F for FUE, R for RAK) and the letters in the right margin represent the operator owning each aircraft needed (B for Binter, N for Naysa, C for Canair). Image (a) is the optimal solution computed by the exact algorithm in [9], and consists of 16 aircraft routes and 30 crew routes, with 6 aircraft changes. By allowing ± 10 minutes for retiming with respect to the original flight departures, the solution shown in (b) consists of 14 aircraft routes and 27 crew routes, with 5 aircraft changes. Also relevant for the cost reduction, it is worth noting that (a) uses 2 aircraft and 2 crew members from the most-expensive operator (Binter), while (b) succeeds in performing all the flights with the two low-cost operators (Naysa and Canair). The goal of the current paper is to propose and compare several algorithms to find solutions like the one in (b). As shown in the figure, the cost reduction can be significant, thus confirming the high interest of the regional carrier in pursuing this goal.

4 Mathematical Model

This section presents the MILP model proposed in [9] and extended in this paper to deal with flight retiming. This model will be used by all the proposed algorithms. It contains route variables representing the crew pairings and arc variables representing the aircraft routing. The model is based on two directed acyclic graphs, used to represent the feasible flight sequences as (aircraft and crew) routes.

Let $G^a = (N, A^a)$ and $G^c = (N, A^c)$ be, respectively, the aircraft and the crew graphs, where $N = N^{bd} \cup N^r \cup N^{ba}$ is the set of nodes: N^{bd} is the set of departure base nodes, N^{ba} the set of arrival base nodes and N^r the set of flight nodes. The set N^r includes one node for each original flight f and for each retimed copy of the flight in C^f , being $f \in F$. Departure base nodes and arrival base nodes correspond to bases in M^B . The sets of arcs A^a and A^c represent, respectively, the feasible flight sequences for aircraft and crew. In particular, arc $(i, j) \in A^a$ represents a feasible departure flight when $i \in N^{bd}$ is a departure base and $j \in N^r$ is a flight departing from that base, a feasible arrival flight when $j \in N^{ba}$ is an arrival base and $i \in N^r$ is a flight arriving at that base, and a feasible connection of two flights when $i, j \in N^r$. The same holds for arcs $(i, j) \in A^c$ but, in this case, crew sequencing rules are considered. Let A_s^c be the subset of crew arcs corresponding to short connections.

To deal with flight retiming, we further define the set $NFC^f \subset N^r$ for each original flight $f \in F$. The set NFC^f includes the node corresponding to the original flight f and the nodes corresponding to its copies in C^f . Let \mathcal{R}_c^{kl} be the set of feasible crew routes in $G^c = (N, A^c)$ for a crew member of operator $k \in K$ departing from base $l \in N^{bd}$. Each crew route R has an associated cost c_R which depends on the connection times between consecutive flights, and a cost β_k which represents the crew salary and depends on the operator. The parameter α is used to normalize the two terms. The cost γ_k associated with the number of aircraft routes also depends on the operator, while the cost δ of the aircraft changes is the same for all operators. Appropriate values for these parameters were selected by the regional carrier in the initial step, before setting up the model for the problem.

We introduce the following binary variables:

- for each crew route $R \in \mathcal{R}_c^{kl}$, a binary variable x_R assuming value 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 $(i, j) \in A^a$, each operator $k \in K$ and each base $l \in N^{bd}$, a binary arc-flow variable y_{ij}^{kl} assuming value 1 if and only if arc (i, j) is operated by an aircraft of operator k departing from base l ;

- for each arc $(i, j) \in A^c \setminus A_s^c$ a binary variable z_{ij} assuming value 1 if and only if an aircraft change occurs between flights i and j .

The MILP model then reads as follows:

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

$$\sum_{k \in K, l \in N^{bd}, i \in N^{FC^f}, R \in \mathcal{R}_c^{kl}: i \in R} x_R = 1 \quad \forall f \in F \quad (2)$$

$$\sum_{R \in \mathcal{R}_c^{kl}} x_R \leq n_c^{kl} \quad \forall k \in K, l \in N^{bd} \quad (3)$$

$$\sum_{(i, j) \in A^a} y_{ij}^{kl} = \sum_{(j, i) \in A^a} y_{ji}^{kl} \quad \forall k \in K, l \in N^{bd}, i \in N^r \quad (4)$$

$$\sum_{i \in N^{bd}, j \in N^r: (i, j) \in A^a} y_{ij}^{kl} \leq n_a^{kl} \quad \forall k \in K, l \in N^{bd} \quad (5)$$

$$y_{ij}^{kl} = 0 \quad \forall k \in K, l \in N^{bd} \setminus H, i \in N^r, j \in N^{ba} \setminus B \quad (6)$$

$$y_{ij}^{kl} = 0 \quad \forall k \in K, l \in H \setminus B, i \in N^r, j \in N^{ba} \setminus H \quad (7)$$

$$\sum_{k \in K, l \in H \setminus B, i \in N^r, j \in H \setminus B: (i, j) \in A^a} y_{ij}^{kl} \leq D_{NB} \quad (8)$$

$$\sum_{k \in K, l \in B, i \in N^r, j \in B: (i, j) \in A^a} y_{ij}^{kl} \leq D_B \quad (9)$$

$$\sum_{l \in N^{bd}, R \in \mathcal{R}_c^{kl}: i \in R} x_R = \sum_{l \in N^{bd}, (i, j) \in A^a} y_{ij}^{kl} \quad \forall k \in K, i \in N^r \quad (10)$$

$$\sum_{l \in N^{bd}, R \in \mathcal{R}_c^{kl}: (i, j) \in R} x_R = \sum_{l \in N^{bd}} y_{ij}^{kl} \quad \forall k \in K, i, j \in N^r: (i, j) \in A_s^c \quad (11)$$

$$\sum_{k \in K, l \in N^{bd}, R \in \mathcal{R}_c^{kl}: (i, j) \in R} x_R \leq \sum_{k \in K, l \in N^{bd}} y_{ij}^{kl} + z_{ij} \quad \forall i, j \in N^r: (i, j) \in A^c \setminus A_s^c \quad (12)$$

$$x_R \in \{0, 1\} \quad \forall k \in K, l \in N^{bd}, R \in \mathcal{R}_c^{kl}, \quad (13)$$

$$y_{ij}^{kl} \in \{0, 1\} \quad \forall k \in K, l \in N^{bd}, (i, j) \in A^a, \quad (14)$$

$$z_{ij} \in \{0, 1\} \quad \forall i, j \in N^r: (i, j) \in A^c \setminus A_s^c. \quad (15)$$

The objective function (1) calls for the minimization of the weighted sum of crew, aircraft and aircraft changes costs. Constraints (2) require to select, for each original flight $f \in F$, a crew route visiting it or one of its copies, i.e. one node in N^{FC^f} : these constraints are used to determine the retiming for each flight. Constraints (3) impose to respect the maximum number of crew members available, for each operator, at each base at the beginning of the workday. Since we use arc-flow variables to represent aircraft routing, flow conservation at every flight node is imposed through constraints (4). With constraints (5) we ensure that the maximum number of aircraft available is respected, for each operator, at each base at the beginning of

the workday. Constraints (6) and (7) impose, respectively, that 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, as imposed by the short-term maintenance requirements. Constraints (8) and (9) take into account the short-term maintenance exceptions: at most D_{NB} aircraft routes are allowed to start and end in non-depot bases, and at most D_B aircraft routes are allowed to start and end at depot bases. With constraints (10) we impose that, if a flight node is visited, then the corresponding flight is operated by exactly one crew member and one aircraft of the same operator. Aircraft changes must be avoided on short connections: this is imposed by constraints (11), which requires to have a crew route and an aircraft of the same operator on pairs of consecutive flights whose connection is short. Constraints (12) are used to count the aircraft changes penalized in the objective function: given a pair of flights (i, j) such that the connection between these flights is not short, if a crew member operates them in sequence, then either an aircraft executes them in sequence or an aircraft change occurs. Finally, constraints (13)–(15) define the variable domains. Notice that, as explained in [9], once the integrality constraints are forced on the x and z variables, they are unnecessary on the y variables.

With respect to the model proposed in [9], constraints (2) are imposed for each original flight rather than for each node in N^r : in this way, we guarantee that either the original flight or one of its copies is visited by a crew route: this is equivalent to choosing the departure time for each original flight. Constraints (10), (11) and (12) are imposed for each flight node in N^r but they are active either for the node corresponding to the original flight or for one of its copies: indeed, not all the flight nodes $i \in N^r$ will be visited, as we select, for each $f \in F$, only one node in the set NFC^f . Another difference is that we deal with larger size graphs, since they include one node for each original flight and for each flight copy.

5 Heuristic Algorithms

This section presents four heuristic algorithms that we propose for the problem with retiming. We first describe the common features that all the algorithms share, and then, in Sections 5.1, 5.2, 5.3 and 5.4, we highlight the main differences and specific characteristics of each approach. All the algorithms are based on model (1)–(15), and consist of two phases. The first phase is used to compute a lower bound on the optimal solution value and to define the set of crew routes to be considered in the second phase. The second phase is used to find a feasible solution for the whole problem. The main difference between the four algorithms is the way to choose flight retiming options. In particular, the first two methods consider all flight retiming options (i.e. all original flights and all flight copies), while the other two methods select a subset of the flight copies.

To compute a lower bound, in the first phase of the algorithms, we consider the LP-relaxation of model (1)–(15), and enhance it with a bounding cut on the minimum number of aircraft changes needed, as proposed in [9]. This cut takes into account that a minimum number of aircraft changes is needed in any optimal solution. Indeed, aircraft changes are mandatory because the short-term maintenance requires every aircraft to stay overnight at the depot every two days, while every crew member is required to go back to its home base to avoid overnight rest outside the home base. The computation of the bounding cut consists of three steps: first, we count, for every time instant t , the maximum number of aircraft departing from (arriving at, resp.) every home base in H before time t . Then, we decrease the counters by the number of aircraft routes that can start and end at the same base (D_{NB} and D_B). Finally, the maximum of all counters gives the minimum number of aircraft changes needed

in an optimal solution. The short-term maintenance exceptions are taken into account. The computation of the bounding cut has been extended to consider flight copies. In particular, we consider only the latest copy of a flight when counting the aircraft departures, and the earliest copy of a flight when counting the aircraft arrivals.

Due to the very large number of crew-route variables in model (1)–(15), solving its LP-relaxation with *all* such variables could be extremely time-consuming. Therefore, a column-generation procedure is applied. The pricing problem calls for determining, for each operator $k \in K$ and each base $l \in N^{bd}$, the feasible crew route with the smallest reduced cost. A crew route is feasible if it satisfies all constraints on flight sequencing and the work rules for the crew members:

- (i) plane turn time and maximum connection time must be respected between every pair of consecutive flights in the route;
- (ii) limits on the maximum number of flights and the route duration must be respected;
- (iii) consecutive arrival and departure airports must coincide and belong to the set of home bases H , unless an unavoidable overnight rest outside the home base is needed (and only allowed to a subset of the operators).

The pricing problem corresponds to an Elementary Shortest Path with Resource Constraints (ESPRC) to be computed on the acyclic graph $G^c = (N, A^c)$ defined above. Dynamic programming is used to solve ESPRC: each node in N^r is associated with a set of labels that store, for different levels of resource consumption (route duration and number of flights in the route), the best path from base l to the node. Labels are then propagated and simple dominance rules applied to discard unpromising labels (i.e. we take into account the consumption of each resource and the label profit that derives from the dual variables associated with the constraints of the LP-relaxation of model (1)–(15)). Finally, for each operator $k \in K$ and each base $l \in N^{bd}$, the route with the smallest reduced cost is reconstructed.

The column-generation procedure is iteratively applied and, at each iteration, up to CR crew routes (500 in our experiments) having the smallest negative reduced cost are generated and added to the LP-relaxed model. Different terminating conditions are used to stop the column-generation procedure in the four algorithms, and are described in the next sections. When the column-generation procedure is terminated, the second phase starts: it consists in solving, for a given time limit and by using a general-purpose solver, a *reduced* MILP model created from (1)–(15) but containing variables only for the crew routes generated during the first phase. Note that, since the four algorithms consider different retiming options, the set of crew routes is different in each algorithm. In addition, the set of aircraft variables y and aircraft change variables z contained in the reduced MILP model varies according to the algorithm. More details will be provided in the next sections.

5.1 Method Adapted from [9]

The first method consists of applying the first and second phases of the exact algorithm proposed in [9], now adapted to handle flight retiming. In particular, we consider all original flights and flight copies in graphs $G^a = (N, A^a)$ and $G^c = (N, A^c)$, i.e., all retiming options are available. The first phase applies column generation on the crew routes and, to limit its computing time, we stop the column-generation procedure when the improvement of the lower-bound value obtained in the current iteration is smaller than a given threshold θ compared to the one obtained NM iterations before. Based on preliminary experiments, we fixed $\theta = 0.5\%$ and $NM = 5$. In order to obtain a valid lower-bound, when the column-generation procedure

stops, we compute the most negative reduced cost crew route for each operator $k \in K$ and departure base $l \in N^{bd}$. Then, we decrease the current lower-bound value by the sum of the computed reduced costs. Therefore, this is a valid lower bound on the MILP model (1)-(15) with all flight retiming possibilities, and can be used to measure the quality of the obtained heuristic solutions. Afterwards, we apply the second phase on the reduced MILP model that contains all the y arc-flow variables, all the z variables for counting the aircraft changes, and only the x variables corresponding to crew routes generated during the first phase. The reduced MILP model is solved by a general purpose solver for a given time limit.

The main advantage of this method is that all retiming options are considered. Hence, more flexibility is allowed and larger improvement can be expected. A drawback is certainly that the computing time can be rather large, as it will be shown in Section 6.

5.2 Fixed Aircraft Routes Retiming

To limit the computing time, but nevertheless keeping all retiming options, we developed a second method in which aircraft routes are fixed as in the solution without retiming. More precisely, we fix the assignment of flights to aircraft as in the optimal solution without retiming, which represents the existing plan, and allow using all original flights and all flight copies (i.e., all retiming options are available). This algorithm should resemble the behavior of a manual planner, who applies slight modifications to the flight schedule, without destroying the existing plan. By fixing the assignment of flights to aircraft, only a subset of the arc-flow variables y is present in the LP-relaxed model, namely those corresponding to the assignment of flights to aircraft in the optimal solution without retiming. However, by allowing retiming, different crew routes can be obtained, having a different assignment of the flights to the crew members, and flight copies can be selected to replace the original flights. The reason of fixing the assignment of flights to aircraft is that we want to limit the problem size but, at the same time, possibly allow the reduction of the number of crew routes, which is the most-expensive component of the costs.

Once the assignment of flights to aircraft is fixed, we apply the first phase to generate crew routes by column generation. In this case, the column-generation procedure is executed until no new columns with negative reduced costs are found. Clearly, the lower bound obtained at the end of the column-generation process is only valid for the fixed flight assignment and is not a valid lower bound on the MILP model with all y variables. The second phase solves the reduced MILP model that includes the arc-flow variables selected according to the assignment of flights to aircraft (as described above), all z variables, and the crew routes generated during the first phase.

This method has the advantage of dealing with a much smaller number of arc-flow variables, and thus requires much shorter computing times (see Section 6). However, it has far-less flexibility, that results in limited improvement of the solution costs.

5.3 Method with Flight Copy Pre-selection

This method follows a different approach to reduce the size of the problem: it selects a subset of the flight copies, i.e. limits the set of retiming options. In particular, flight copies are pre-selected before applying the first phase of the algorithm, so that graphs $G^a = (N, A^a)$ and $G^c = (N, A^c)$ include a smaller number of nodes. Several alternative selection criteria have been tested: (i) selection of all flight copies in a peak period, i.e. a time interval in which

many flights are scheduled; (ii) selection of copies of flights that (at least partially) overlap in time; (iii) selection of copies as in (ii) and also copies of flights whose connection time is larger than a given threshold; (iv) selection of copies in two steps, where the first one, as in (ii), selects copies of flights that (at least partially) overlap in time, and the second one performs the selection of additional copies by considering the (at least partial) overlapping between the original flights and the copies generated in the first step.

All these selection criteria aim at selecting flight copies so as to allow new connection possibilities between flights, while limiting the set of flight retiming possibilities. By allowing new flight connections it might be possible to reduce the number of crew and/or aircraft routes, the connection costs, as well as the number of aircraft changes.

Once the flight copies have been selected, we execute the two phases. In the first phase, column generation on the crew routes is applied, and the LP solution of model (1)–(15), having only the selected subset of nodes in N , is computed. The column-generation procedure is executed until no new column with negative reduced-cost is found. Notice that the lower bound is only valid with respect to the current flight copies selection. In the second phase, we solve the reduced MILP model that contains only the y and z variables corresponding to the original flights and the selected flight copies, and the x variables corresponding to crew routes generated during the first phase. We report, in Section 6, only the results obtained by applying the best selection criterion, namely (ii).

This algorithm has the advantage that by selecting a subset of the flight copies it is possible to control the computing time. However, it is quite hard to determine a “good” subset of flight copies: preliminary computational experiments showed that better results can be obtained by enlarging the set of retiming options, but the computing time rapidly increases too.

5.4 LP-based Retiming Selection Heuristic Algorithm

The last method that we present is also based on selecting a subset of the flight copies. However, instead of using a pre-selection criterion, it uses the solution of the LP-relaxed model to effectively limit the flight retiming choices, as follows. The column-generation process is executed for IT iterations ($IT = 10$ in our experiments), and the LP solution obtained after IT iterations is used to decide which arc-flow variables y should be kept and which ones could be discarded. Recall that each arc-flow variable corresponds to an arc of the aircraft graph $G^a = (N, A^a)$, and represents either the sequencing of two flights or the departure or arrival of a flight from/to a base. These arcs can connect original flights with other original flights, original flights with copies of other flights, or can also connect flight copies. Therefore, the choice of the arc-flow variables influences the choice of the flight copies that can be used in the aircraft and crew routes, i.e., by selecting a subset of the arc-flow variables we select the allowed retiming options. This choice is based on the LP solution obtained after IT iterations: in particular, we discard from the LP-model all y variables that are set to 0 in the LP solution, unless the corresponding arcs connect original flights. Indeed, we always want to keep the original flights while limiting the retiming choices. We decided to examine the LP solution after IT iterations, since the first iterations require rather short computing times, as it is usual in column generation, while later the process becomes more heavy.

After IT iterations have been executed, the column generation process is iterated on the obtained reduced LP-relaxed model until it is solved to optimality. Note that, also in this case, the obtained lower bound is not a valid lower bound for the problem including all flight copies (retiming options), but is a valid lower bound for the reduced MILP model including the reduced set of arc-flow variables. As it will be shown in Section 6, it is much more effective to

delegate the choice of the flight retiming options to the LP solution than making a pre-selection of the flight copies.

The second phase consists of deriving a heuristic solution by solving the reduced MILP model that contains the arc-flow variables that have not been discarded during the first phase, the crew-route variables generated in the first phase, and all the z variables.

The main advantage of this algorithm is that there is no need to appropriately select a subset of the flight copies a priori, as this choice is based on the LP solution, which can take into account the crew routes generated during the column-generation process and, consequently, the most-useful retiming options. We also tried other LP-based selection methods, such as discarding the non-used y variables as soon as the lower bound value became smaller than the lower bound value without retiming, or discarding, after IT iterations of the column-generation process, the y variables that were used a small number of times during the IT iterations. However, these variants obtained results of worse quality.

6 Computational Experiments

This section is divided into three parts. Section 6.1 compares the four heuristic algorithms in terms of solution quality and computing time, and determines which algorithm achieves the best performance. Section 6.2 reports a sensitivity analysis of how the most effective algorithm behaves under different numbers of alternative departure times and different costs for aircraft and for crew connections. Finally, Section 6.3 shows the performance of the most effective algorithm on instances of double size.

6.1 Comparison of the Algorithms

To compare the four algorithms we decided to use the same benchmark instances tested in [9], where results for the problem without retiming are presented and discussed. These instances correspond to the real-world flights of the regional carrier in Canary Islands during the first week of September 2012 and the first week of April 2012. Table 1 shows the day to identify each instance, the number $\#f$ of flights, the number $\#conn$ of all connections, the number $\#SC$ of short connections (less than 30 minutes), the number $\#RC$ of arcs where an aircraft change may occur (connection times between 30 minutes and 3 hours), and the number $\#OC$ of other connections (which are feasible for aircraft but not for crew). Table 1 also reports, for each instance, the number $\#a$ of aircraft, the number $\#c$ of crew members, the number $\#ch$ of aircraft changes, and the optimal solution value when retiming is not allowed. We use these features for evaluating the solutions obtained by the retiming algorithms introduced in this paper.

On these instances, when computing c_R (for each $R \in \mathcal{R}_c^{kl}$, $k \in K$, $l \in N^{bd}$), the weights for the time duration of the connections in crew routes are as follows: 2 for connections between 20 and almost 30 minutes, 1 for the connections between 30 minutes and one hour, 3 for the connections larger than one hour but not larger than two hours, 5 for larger connections. The weights in the objective function are as follows: $\alpha = 1$, $\beta_1 = 1000$, $\beta_2 = 950$, $\beta_3 = 910$, $\gamma_k = 10$ (for each $k \in K$) for the aircraft routes that respect short-term maintenance rules and $\gamma_k = 20$ (for each $k \in K$) for the short-term maintenance exceptions (with $D_B = 2$ and $D_{NB} = 2$), and $\delta = 100$. All these values are in [9], and we decided to use them in our experiments to compare the obtained solutions. According to the considered weights, the main goal is to minimize the numbers of crew members used to serve all the flights, but also to minimize the number of

Inst.	#f	#conn	#SC	#RC	#OC	#a	#c	#ch	opt value
1/9	102	938	19	253	666	12	22	4	22028
2/9	140	1661	35	515	1111	15	28	5	27828
3/9	130	1540	44	450	1046	14	25	5	25172
4/9	124	1409	41	408	960	13	24	4	24017
5/9	124	1407	42	407	958	13	24	4	24121
6/9	128	1474	45	416	1013	13	25	4	24796
7/9	150	2041	40	580	1421	15	28	5	27970
1/4	138	1786	41	560	1185	17	27	6	27287
2/4	132	1804	34	493	1277	16	29	6	28985
3/4	138	2000	31	578	1391	16	29	6	29266
4/4	136	1966	30	564	1372	16	29	6	29101
5/4	144	2176	33	640	1503	16	30	6	29962
6/4	172	2915	46	830	2039	17	33	6	33103
7/4	100	1001	9	282	710	12	21	4	21300

Table 1 Details of the instances and optimal solutions without retiming.

aircraft changes, the cost related to the waiting times at the connections for the crew members, and the number of aircraft used to serve all the flights. To adapt the instances for the problem with retiming, we considered a retiming of ± 10 minutes for each flight, hence each flight has three possible departure times in our experiments.

The four algorithms described in this paper were implemented in C. All the tests were performed on a personal computer with a i5-2400 at 3.10 Ghz and 16 Gb of RAM. IBM Ilog Cplex 12.4 was used to solve the LP-relaxed models and the reduced MILP models. A single thread and default Cplex parameter settings were used. We fixed a time limit of 1800 seconds for the second phase of each algorithm.

The remainder of this section is devoted to analyze the performances of the four algorithms, described in Sections 5.1–5.4, in terms of solution quality and computing times. The results of each algorithm are reported in Sections 6.1.1–6.1.4. Each table displays the following information for each instance: the number of aircraft ($\#a$), crew members ($\#c$) and aircraft changes ($\#ch$) in the best solution found by the method, the lower bound (LB) obtained at the end of the first phase, the upper bound (UB) obtained at the end of the second phase, the percentage gap ($\text{gap}\%$) between upper and lower bounds, the computing times (t_{LB} and t_{UB}) of the first and second phases, respectively, the whole computing time (time) of the two-phase heuristic algorithm (i.e., the sum of t_{LB} and t_{UB}), and the percentage improvement ($\text{impr}\%$) with respect to the optimal solution of the non-retiming problem (reported in Table 1). All times are expressed in seconds.

Recall that, except from those reported in Table 2, lower bounds LB are not valid lower bounds for the whole problem (with all the retiming options) because fixed aircraft routes (see Section 5.2) or subsets of flight copies (see Sections 5.3 and 5.4) are considered. However, the lower bound values are *locally* valid, and hence they are used to evaluate the gap between upper and lower bounds of each algorithm.

The last line of each table reports the average values for the percentage gap, computing time and percentage improvement. We show in boldface the number of aircraft, crew or aircraft changes, when they are smaller than the corresponding values in the optimal solution without retiming. Note that any solution of the problem without retiming is a valid heuristic solution that could be given as an upper bound to any of the four algorithms. To better measure the performance of each approach, we show the results obtained without using a starting solution.

6.1.1 Results of the Method Adapted from [9]

Table 2 reports the results obtained by the method of [9] adapted to handle retiming. No feasible solution has been obtained for instance 6/4, and, for that reason, the last line of the table reports the average values computed over all other instances. As we can see, the improvement on the optimal solution without retiming is remarkable (6.58% on average). However, the computing time is also rather large (about 50 minutes) even without considering the unsolved instance. If we include also the computing time of this instance, the average becomes almost one hour (more precisely, 3522 seconds). This method shows its limits, as it is not able to derive a feasible solution for the instance 6/4, which is the largest one in our collection. On this instance, the first phase took 8418 seconds to compute LB and then the second phase was interrupted after 1800 seconds without having found any feasible solution. This is mainly because we allow the use of all flights and flight copies, and no fixing is applied. On the other instances, the improvements of the obtained solutions respect to the non-retiming ones are quite relevant.

Inst.	#a	#c	#ch	LB	UB	gap%	t_{LB}	t_{UB}	time	impr%
1/9	12	20	4	19809	19829	0.10	523	1800	2323	9.98
2/9	17	26	6	26184	26328	0.55	1904	1800	3704	5.39
3/9	14	24	5	24076	24100	0.10	1668	850	2518	4.26
4/9	13	23	4	23229	23396	0.71	1396	1800	3196	2.59
5/9	13	23	4	23450	23476	0.11	1328	1800	3128	2.67
6/9	14	24	4	23981	24052	0.30	1615	1800	3415	3.00
7/9	15	26	8	26474	26853	1.41	2597	1800	4397	3.99
1/4	16	26	6	26025	26065	0.15	2364	1800	4164	4.48
2/4	14	26	5	25788	25812	0.09	2295	311	2606	10.95
3/4	14	26	5	26195	26204	0.03	2617	200	2817	10.46
4/4	14	26	5	25842	25844	0.01	2537	918	3455	11.19
5/4	14	27	5	26736	26806	0.26	2367	285	2652	10.53
6/4	-	-	-	30965	-	-	8418	1800	10218	-
7/4	11	20	4	20012	20022	0.05	654	73	727	6.00
Avg.						0.30			3007.8	6.58

Table 2 Results of the Method Adapted from [9].

6.1.2 Results of the Fixed Aircraft Routes Retiming

Table 3 shows the results computed by the Fixed Aircraft Routes algorithm. It is evident that the computing time is much shorter than that of the previous method. This is certainly due to the limited flexibility for the flight connections as the assignment of flights to aircraft is fixed. This is also shown by the lower-bound values, that are significantly larger than those of Table 2. For the same reason, the improvement is rather small (only 1.01%): on two instances (1/9 and 5/4) the number of crew members can be reduced, while all other cost reductions are only due to the decrease of connection costs. Therefore this method can become useful when a very strict limit on the computing time is imposed, or if an existing aircraft routing plan should not be changed.

Inst.	#a	#c	#ch	LB	UB	gap%	t_{LB}	t_{UB}	time	impr%
1/9	12	21	8	21502	21522	0.09	49	1	50	2.30
2/9	15	28	5	27605	27675	0.25	261	6	267	0.55
3/9	14	25	5	24957	24957	0.00	141	2	143	0.85
4/9	13	24	4	23800	23800	0.00	103	1	104	0.90
5/9	13	24	4	23887	23887	0.00	100	1	101	0.97
6/9	13	25	4	24573	24573	0.00	97	1	98	0.90
7/9	15	28	5	27785	27785	0.00	228	2	230	0.66
1/4	17	27	6	27137	27137	0.00	720	3	723	0.55
2/4	16	29	6	28893	28893	0.00	324	2	326	0.32
3/4	16	29	6	29032	29032	0.00	397	3	400	0.80
4/4	16	29	6	28922	28922	0.00	341	2	343	0.62
5/4	16	29	7	29051	29051	0.00	448	4	452	3.04
6/4	17	33	6	32993	32993	0.00	1569	8	1577	0.33
7/4	12	21	4	21019	21019	0.00	85	1	86	1.32
Avg.						0.02			350.0	1.01

Table 3 Results of the Fixed Aircraft Routes retiming algorithm.

6.1.3 Results of the Method with Flight Copy Pre-selection

Table 4 reports the results obtained by the algorithm in which we apply a pre-selection of the flight copies. The percentage gap is very small (0.04% on average), and the computing time is reduced with respect to the method adapted from [9] (see Table 2), but the improvement is also smaller (4.82% versus 6.58%). However, the method was able to find a feasible solution for every instance. By comparing the results to those in Table 3, we can observe that much larger improvements can be obtained by selecting copies rather than by allowing all retiming options and keeping the aircraft routes fixed.

Inst.	#a	#c	#ch	LB	UB	gap%	t_{LB}	t_{UB}	time	impr%
1/9	12	20	4	20008	20027	0.09	99	1800	1899	9.08
2/9	15	27	5	26960	26980	0.07	409	1800	2209	3.05
3/9	14	25	5	25003	25004	0.00	253	34	287	0.67
4/9	13	24	4	23864	23884	0.08	218	1800	2018	0.55
5/9	13	24	4	24018	24038	0.08	262	1800	2062	0.34
6/9	13	25	4	24676	24696	0.08	294	1800	2094	0.40
7/9	15	27	5	26897	26917	0.07	841	1800	2641	3.76
1/4	16	26	6	26136	26156	0.08	629	1800	2429	4.14
2/4	14	27	5	26830	26830	0.00	331	26	357	7.43
3/4	14	27	5	26845	26845	0.00	1036	23	1059	8.27
4/4	14	27	5	26825	26825	0.00	1141	62	1203	7.82
5/4	14	27	5	26858	26859	0.00	1165	79	1244	10.36
6/4	16	31	6	31053	31055	0.01	3776	1533	5309	6.19
7/4	11	20	4	20142	20149	0.03	141	340	481	5.40
Avg.						0.04			1806.6	4.82

Table 4 Results of the method with flight copy pre-selection.

6.1.4 Results of the LP based Retiming Selection Algorithm

We present in Table 5 the results obtained with the LP-based retiming selection algorithm. As we can see, the gaps are small also in this case (0.16% on average, and at most 1.04%). The computing times of about 22 minutes on average are shorter than those of Table 4, and

the improvement is larger (6.32% versus 4.82%). Compared to Table 2, we can see that the LP-based retiming selection algorithm is able to derive a feasible solution for all instances, and the improvement is similar (6.32% versus 6.58%). In addition, the computing time is (more than twice) shorter, and the lower bound values are very close to those of Table 2.

A significant improvement with respect to the optimal solution without retiming is obtained for most of the instances, with reduction of the number of crew members in all but one cases (instance 4/9). The number of aircraft used is increased for two instances (5/9 and 6/9), but this still leads to an overall improvement of the solution (with the reduction of one crew member in both instances). In addition, the number of aircraft changes is reduced for four instances and increased only in one case (instance 2/09). We can also observe that the lower-bound values are smaller than in the case of the method with flight copy pre-selection, suggesting that it is better to select the flight copies by using the LP solution rather than by applying the a-priori criterion.

Inst.	#a	#c	#ch	LB	UB	gap%	t_{LB}	t_{UB}	time	impr%
1/9	12	20	4	19815	19836	0.11	210	53	263	9.95
2/9	15	26	6	26245	26368	0.47	583	112	695	5.25
3/9	14	24	5	24126	24132	0.02	510	478	988	4.13
4/9	13	24	4	23741	23764	0.10	442	1800	2242	1.05
5/9	14	23	4	23557	23581	0.10	405	17	422	2.24
6/9	14	24	4	24093	24135	0.17	519	162	681	2.67
7/9	15	27	5	26616	26896	1.04	1000	1800	2800	3.84
1/4	16	26	6	26070	26094	0.09	1026	1800	2826	4.37
2/4	14	26	5	25833	25836	0.01	775	101	876	10.86
3/4	14	26	5	26267	26280	0.05	863	108	971	10.20
4/4	14	26	5	25863	25868	0.02	865	40	905	11.11
5/4	14	27	5	26823	26825	0.01	852	13	865	10.47
6/4	16	31	6	30990	30999	0.03	2501	1800	4301	6.36
7/4	11	20	4	20032	20032	0.00	280	2	282	5.95
Avg.						0.16			1365.5	6.32

Table 5 Results of the LP-based Retiming Selection Algorithm.

6.1.5 Comparison Summary

In order to provide a compact comparison of the four algorithms, Table 6 reports the following values for each algorithm: total computing time (expressed in seconds) used for executing both phases of the algorithm, solution value (UB), cost for the crew (CCost), cost for the aircraft (ACost) and number of aircraft changes (#ch). In the last row, average values are displayed. The four algorithms are identified in Table 6 as follows: CS17 is the method adapted from [9] (Section 5.1), FIX is the algorithm that uses fixed aircraft routes retiming (Section 5.2), SEL is the method with flight copy pre-selection (Section 5.3), and LPH is the LP-based retiming selection algorithm (Section 5.4).

As it can be seen, for all instances except for 6/4, the lowest crew cost is obtained by CS17 which, however, shows the largest computing times. In addition, algorithm CS17 finds the smallest aircraft cost for all instances except for 2/9, 6/9, and 6/4 (for which no solution is found by CS17). The second best crew cost is found by LPH for all instances. The same algorithm also finds the smallest aircraft cost for all instances except for 5/9, 6/9 and 7/9. For what concerns the number of aircraft changes, LPH finds the smallest one for all instances except for 2/9; this exception requires 6 changes while the lowest number of changes is 5.

The FIX and SEL algorithms provide higher crew costs than CS17 and LPH: by comparing only FIX and SEL, we can see that FIX finds a crew cost lower than that of SEL only for four instances, while the opposite occurs in the remaining ten cases. However, FIX requires much shorter computing times than the other three algorithms. We can also observe that SEL obtains the best aircraft cost and number of aircraft changes for all instances except for 7/9. However, given that the crew cost is much higher than the other costs in the objective function, SEL is outperformed by LPH.

We can conclude that all four algorithms obtained improvements with respect to the optimal solutions without retiming, as shown in Figure 1 (instance 5/4). In addition, they all show small gaps with respect to the *local* lower bounds. While the method adapted from [9] leads to the largest improvements, it also requires the longest computing times, and shows its drawbacks, since it is not able to find a feasible solution for the largest instance within the time limit. By fixing the assignment of flights to aircraft, the computing time is significantly reduced, but the flexibility is also decreased, so that the improvement is marginal. The algorithms that select flight copies show better performances, being able to obtain a feasible solution for all instances and good improvements in reasonable computing times. Especially the LP-based Retiming Selection Algorithm shows a good trade-off between solution quality and computing time, as it is also evident from the results reported in Table 6. We therefore select LPH in the next sections to perform a sensitivity analysis and to test larger size instances.

6.2 Sensitivity Analysis

Given that LPH turns out to achieve the best trade-off between the improvement over the solution without retiming and the corresponding computing time, we now analyze the impact of different numbers of retiming options on its performance. In addition, we compare the results obtained by considering three departure time options spaced by 10 minutes with the results found by including five departure time options spaced by 5 minutes. Finally, we evaluate the effect of different costs for the aircraft and for the crew connections.

6.2.1 Departure Time Options

In the previous section, we considered three departure time options, i.e., the original flight departure time (suggested by the regional carrier and used in [9] for the non-retiming problem) and two retiming options of ± 10 minutes. We also executed other experiments on the same real-world instances with three departure time options and retiming of ± 5 minutes on all flights, instead of ± 10 minutes. As expected, the improvement obtained on the ± 5 minutes instances was much smaller than on the ± 10 minutes instances. More precisely, it was only possible to reduce the costs for the short and long connection times, while the reduction of the number of crews, aircraft, or aircraft changes was not achievable. Since the computing times of the algorithms in both families of instances are very similar (as the graphs and models have nearly the same size in both cases), retiming of ± 10 minutes is definitely recommended in our application when using three departure time options.

We now consider five departure time options either spaced by 5 minutes or by 10 minutes, and seven departure time options spaced by 5 minutes. The obtained results are reported in Table 7, where each configuration is identified by the number of departure time alternatives and by the spacing time interval: e.g., $3D\pm 10'$ corresponds to three departure time options spaced by 10 minutes; in particular this configuration corresponds to the one used for the

Inst.	CS17					FIX					SEL					LPH				
	time	UB	CCost	ACost	#ch	time	UB	CCost	ACost	#ch	time	UB	CCost	ACost	#ch	time	UB	CCost	ACost	#ch
1/9	2323	19829	19279	150	4	50	21522	20572	150	8	1899	20027	19477	150	4	263	19836	19286	150	4
2/9	3704	26328	25518	210	6	267	27675	26985	190	5	2209	26980	26290	190	5	695	26368	25578	190	6
3/9	2518	24100	23440	160	5	143	24957	24297	160	5	287	25004	24344	160	5	988	24132	23472	160	5
4/9	3196	23396	22836	160	4	104	23800	23240	160	4	2018	23884	23324	160	4	2242	23764	23204	160	4
5/9	3128	23476	22916	160	4	101	23887	23327	160	4	2062	24038	23478	160	4	422	23581	23001	180	4
6/9	3415	24052	23472	180	4	98	24573	24013	160	4	2094	24696	24136	160	4	681	24135	23555	180	4
7/9	4397	26853	25893	160	8	230	27785	27105	180	5	2641	26917	26237	180	5	2800	26896	26216	180	5
1/4	4164	26065	25275	190	6	723	27137	26327	210	6	2429	26156	25366	190	6	2826	26094	25304	190	6
2/4	2606	25812	25152	160	5	326	28893	28113	180	6	357	26830	26170	160	5	876	25836	25176	160	5
3/4	2817	26204	25544	160	5	400	29032	28252	180	6	1059	26845	26185	160	5	971	26280	25620	160	5
4/4	3455	25844	25184	160	5	343	28922	28142	180	6	1203	26825	26165	160	5	905	25868	25208	160	5
5/4	2652	26806	26146	160	5	452	29051	28171	180	7	1244	26859	26199	160	5	865	26825	26165	160	5
6/4	10218	-	-	-	-	1577	32993	32193	200	6	5309	31055	30275	180	6	4301	30999	30219	180	6
7/4	727	20022	19492	130	4	86	21019	20469	150	4	481	20149	19619	130	4	282	20032	19502	130	4
Avg:	3007.8	24522.1	23857.5	164.6	5.0	350.0	26517.6	25800.4	174.3	5.4	1806.6	25447.5	24804.6	164.3	4.8	1365.5	25046.1	24393.3	167.1	4.9

Table 6 Comparison of all heuristic algorithms.

LPH approach in the previous section (Table 5), and reported again here (Table 7) for ease of comparison. For each configuration, we show the number $\#a$ of aircraft, the number $\#c$ of crew members and the number $\#ch$ of aircraft changes needed in the solution found, the solution value UB, the total computing time (expressed in seconds), and the percentage improvement with respect to the optimal solution without retiming. In the last row, we report average values, over all instances, for the computing time and for the percentage improvement.

As it can be observed, when five departure time options are considered, the computing time increases of about three times. This is not surprising, as LPH is based on model (1)–(15) that uses aircraft and crew graphs whose sizes rapidly increase with the number of flight copies.

We can also notice that, by allowing five departure time options with time spacing of 5 minutes (i.e. $5D\pm 5'$) instead of three departure time options with time spacing of 10 minutes (i.e. $3D\pm 10'$), we obtain a slight decrease of the improvement (from 6.32% to 5.89% on average). Since all retiming options allowed by $3D\pm 10'$ are also allowed by $5D\pm 5'$, the worsening is mainly due to the choice of the subset of arcs deleted after IT iterations of the LP-relaxed solution process: as a larger number of retiming options is available, it is indeed harder to select the best arc-flow variables based on the LP-solution. Therefore, a configuration with a smaller set of retiming options and a larger time spacing (between the original flight and its copies) is to be preferred over a combination that includes more retiming options spaced by shorter time intervals, when the two configurations allow the same maximum shift for each flight departure. This could be expected, since solution cost reduction is obtained by enabling new flight connections, that are more likely to be possible when larger time spacings are considered. The availability of more retiming options covering the same time interval does not appear to be advantageous since it increases the problem size and requires longer computing times.

From Table 7, we also observe that, although configuration $5D\pm 10'$ requires longer computing times, it also provides a larger improvement (11.87% on average) over the solution without retiming. Indeed, each flight can be shifted up to 20 minutes earlier or later than in the original schedule, and new flight connections become available. Clearly, a change of 20 minutes for the flight departures has to be carefully analyzed, based on the passenger demand, especially in the setting of the considered regional carrier (where many flights are used by commuters for reaching their work place and going back home, even on the same day).

Finally, we also notice that, when seven departure time options are considered, the computing time further increases. While instances with five departure times can be solved in about one hour and a half (which is still a reasonable time), instances with larger sets of alternatives tend to need longer computing times (about two hours and fifteen minutes on average). The computing time is especially long for the larger size instances (7/9, 5/4 and 6/4). These instances contain a higher number of flights and a very large number of connections, which has a strong impact on the performance in the retiming setting. This observation is typical in a regional carrier, operating many short flights between a small set of airports. In these cases, as observed in the comparison between $3D\pm 10'$ and $5D\pm 5'$, the choice of restricting the number of retiming options to a small set and possibly enlarging the time spacing between the provided departure times appears to be more effective. It is worth mentioning that, in the considered real-world application, the time range for the retiming options should not be too wide (e.g. at most 10 minutes), as also noted by other researchers (see e.g. [8], [11], [17]).

6.2.2 Aircraft Costs

In this section, we compare the results obtained by using LPH to solve instances with different aircraft costs in the objective function, namely $\gamma_k \in \{1, 10, 50\}$ for each operator $k \in K$; we

Table 7 Results with different numbers of departure time options.

Inst.	3D±10'					5D±5'					5D±10'					7D±5'								
	#a	#c	#ch	UB	time	impr%	#a	#c	#ch	UB	time	impr%	#a	#c	#ch	UB	time	impr%	#a	#c	#ch	UB	time	impr%
1/9	12	20	4	19836	263	9.95	12	20	4	19915	808	9.59	11	18	4	17841	1055	19.01	12	20	4	19786	2451	10.18
2/9	15	26	6	26368	695	5.25	15	27	5	26428	3718	5.03	16	24	4	25283	4178	9.15	16	25	8	25548	7240	8.19
3/9	14	24	5	24132	988	4.13	15	24	5	24332	2820	3.34	14	22	5	22243	2385	11.64	14	24	5	24050	6205	4.46
4/9	13	24	4	23764	2242	1.05	13	24	4	23791	3139	0.94	14	22	5	22434	3446	6.59	13	23	4	22971	4732	4.36
5/9	14	23	4	23581	422	2.24	13	24	4	23919	3241	0.84	15	22	6	22390	3525	7.18	13	23	4	23263	5033	3.56
6/9	14	24	4	24135	681	2.67	16	24	5	24586	2694	0.85	15	22	6	22345	3750	9.88	13	24	4	23727	5503	4.31
7/9	15	27	5	26896	2800	3.84	15	27	5	26932	5451	3.71	17	26	9	26864	6804	3.95	16	25	6	25653	11326	8.28
1/4	16	26	6	26094	2826	4.37	17	26	6	26211	4708	3.94	16	24	6	24588	5310	9.89	17	24	8	25058	9413	14.08
2/4	14	26	5	25836	876	10.86	14	26	5	25824	3114	10.91	14	24	5	23946	3386	17.38	14	25	5	24903	7229	14.08
3/4	14	26	5	26280	971	10.20	14	26	5	26335	3182	10.02	15	24	5	24399	3779	16.63	14	25	7	25447	8822	13.05
4/4	14	26	5	25868	905	11.11	14	26	5	25901	3386	11.00	14	24	5	24009	4104	17.50	14	25	5	24959	9654	14.23
5/4	14	27	5	26825	865	10.47	14	27	5	26823	3877	10.48	15	25	5	25376	6562	15.31	14	26	5	25896	11265	13.57
6/4	16	31	6	30999	4301	6.36	16	31	7	31138	10952	5.94	17	28	11	29192	11516	11.81	17	28	11	29190	29192	11.81
7/4	11	20	4	20032	282	5.95	11	20	4	20060	1186	5.82	13	19	5	19118	3327	10.24	11	20	4	19906	2716	6.54
Avg.				1365.5		6.32				3734.0		5.89				4509.1		11.87				8144.0		8.91

recall that $\gamma_k = 10$ for the results in Table 5 and that for the short-term maintenance exceptions γ_k is doubled, hence it becomes 2, 20 or 100 in those exceptional cases. We identify the three settings by C1, C10 and C50. For each setting, Table 8 reports the computing time (expressed in seconds), the number $\#a$ of aircraft, the number $\#c$ of crew members, the number $\#ch$ of aircraft changes, and the number $\#exc$ of short-term maintenance exceptions. We do not report the solution value, since it is computed, in every setting, with different aircraft costs, and thus the solution values could not be directly compared. In the last row, we report average values over all instances.

Inst.	C1					C10					C50				
	time	$\#a$	$\#c$	$\#ch$	$\#exc$	time	$\#a$	$\#c$	$\#ch$	$\#exc$	time	$\#a$	$\#c$	$\#ch$	$\#exc$
1/9	216	12	20	4	3	263	12	20	4	3	345	12	20	5	1
2/9	2445	17	26	6	4	695	15	26	6	4	2070	15	26	6	4
3/9	503	15	24	5	3	988	14	24	5	2	509	14	24	5	2
4/9	1107	13	24	4	3	2242	13	24	4	3	2287	13	24	5	1
5/9	452	13	24	4	3	422	14	23	4	4	2180	13	24	5	1
6/9	495	13	25	4	3	681	14	24	4	4	2301	13	25	4	3
7/9	1111	15	27	5	3	2800	15	27	5	3	2821	15	27	7	1
1/4	1512	17	26	6	4	2826	16	26	6	3	2743	16	26	6	3
2/4	717	14	26	5	2	876	14	26	5	2	721	14	26	5	2
3/4	899	14	26	5	2	971	14	26	5	2	1087	14	26	5	2
4/4	887	14	26	5	2	905	14	26	5	2	825	14	26	5	2
5/4	1689	14	27	5	2	865	14	27	5	2	1128	14	27	5	2
6/4	4152	16	31	6	2	4301	16	31	6	2	4778	17	31	8	3
7/4	707	12	20	4	3	282	11	20	4	2	260	11	20	4	2
Avg.	1206.6	14.2	25.1	4.9	2.8	1365.5	14.0	25.0	4.9	2.7	1718.2	13.9	25.1	5.4	2.1

Table 8 Results with different aircraft costs.

A first observation from the table is that the computing time, on average, is not strongly affected by the different γ_k values, although instances 4/9-1/4 show an increase of the computing time in setting C50 with respect to C1. Moreover, we can see that, as expected, the numbers of aircraft and short-term maintenance exceptions decrease as the aircraft cost increases. In particular, in setting C50, the smallest numbers of aircraft and short-term exceptions occur for all instances except for 6/4. The number of crew members employed is almost the same in every setting. On the contrary, to reduce the number of aircraft used when $\gamma_k = 50$, the number of aircraft changes increases, since aircraft changes have lower cost than crew members. Reducing the number of aircraft used and the short-term maintenance exceptions can become very useful if the airline company needs to rent some aircraft, or if more complex maintenance rules are required on the aircraft. In these cases, the C50 setting is recommended.

6.2.3 Crew Connection Costs

The second cost variation that we consider is to give more or less importance to the crew connection costs in the objective function. We recall that the crew cost is made of two components: a salary cost β_k that depends on the airline operator (with $k \in K$), and a cost c_R that depends on the connection times between consecutive flights in the crew pairing R (with $R \in \mathcal{R}_c^{kl}$, $k \in K$, $l \in N^{bd}$). We keep the same salary cost and weights used in Section 6.1 for computing c_R . Remember that we used weight 2 for the connections between 20 and almost 30 minutes, 1 for the connections between 30 minutes and one hour, 3 for the connections larger than one hour but not larger than two hours, and 5 for larger connection times. In this section, for the cost variation analysis, we multiply the crew connection cost c_R by 1/2 or by 2 in the objective function, thus giving half or double importance to this cost component

with respect to the one used for obtaining the results in Table 5. Table 9 reports the results obtained by LPH when the crew connection cost is weighted a half (CW.5), one (CW1) or double (CW2). For each setting, we display the computing time (expressed in seconds), the numbers w_2 , w_1 , w_3 and w_5 of connections with weights 2, 1, 3 and 5, respectively. In addition, we show the number $\#a$ of aircraft, the number $\#c$ of crew members and the number $\#ch$ of aircraft changes. Finally, in the last row, we report average values over all instances.

By the weight definition, connections w_1 are the preferred ones; indeed, the other types of connections have either short connection times and can cause delay propagation, or long connection times and lead to inefficient crew pairings. Table 9 shows that, as the connection cost increases, the number of connections of type w_1 increases too. In addition, the number of connections of type w_3 and w_5 decreases notably. As a drawback, we also observe that, for instances 5/9, 6/9 and 3/4, the number of crew members in CW2 is larger than that in CW1. On the contrary, in CW.5, a further reduction of $\#c$ with respect to CW1 occurs for instances 4/9 and 7/9. Both the number of aircraft used and the number of aircraft changes are not very affected by the different crew connection costs.

Inst.	CW.5								CW1								CW2							
	time	w_2	w_1	w_3	w_5	$\#a$	$\#c$	$\#ch$	time	w_2	w_1	w_3	w_5	$\#a$	$\#c$	$\#ch$	time	w_2	w_1	w_3	w_5	$\#a$	$\#c$	$\#ch$
1/9	2044	4	74	4	0	12	20	4	263	10	69	3	0	12	20	4	206	10	69	3	0	12	20	4
2/9	2193	20	88	4	2	17	26	6	695	24	84	4	2	15	26	6	941	16	92	4	2	16	26	6
3/9	474	28	74	3	1	14	24	5	988	25	77	3	1	14	24	5	542	23	79	3	1	14	24	5
4/9	704	24	68	6	3	14	23	4	2242	4	95	0	1	13	24	4	1235	6	93	0	1	13	24	4
5/9	602	22	69	6	4	14	23	4	422	26	67	4	4	14	23	4	375	9	88	2	1	13	24	4
6/9	594	23	76	3	2	13	24	4	681	23	77	2	2	14	24	4	996	3	100	0	0	13	25	4
7/9	3093	17	100	2	5	15	26	5	2800	17	103	3	0	15	27	5	1449	12	109	2	0	15	27	5
1/4	2758	25	87	0	0	16	26	6	2826	15	97	0	0	16	26	6	1228	18	94	0	0	16	26	6
2/4	845	19	87	0	0	14	26	5	876	21	85	0	0	14	26	5	637	21	85	0	0	14	26	5
3/4	975	30	78	2	2	14	26	5	971	33	75	2	2	14	26	5	2917	24	87	0	0	14	27	6
4/4	915	29	81	0	0	14	26	5	905	23	87	0	0	14	26	5	713	35	75	0	0	14	26	5
5/4	1015	19	98	0	0	14	27	5	865	20	97	0	0	14	27	5	2319	11	106	0	0	14	27	5
6/4	2519	15	126	0	0	16	31	6	4301	20	121	0	0	16	31	6	2904	8	133	0	0	16	31	6
7/4	360	4	71	5	0	11	20	4	282	8	67	5	0	11	20	4	257	9	66	5	0	11	20	4
Avg.	1363.6	19.9	84.1	2.5	1.4	14.1	24.9	4.9	1365.5	19.2	85.8	1.9	0.9	14.0	25.0	4.9	1194.2	14.6	91.1	1.4	0.4	13.9	25.2	4.9

Table 9 Results with different crew connection costs.

To conclude this section, we underline that many parameters are present in the problem formulation, and many combinations could be analyzed. However, we decided to focus on some configurations of interest for the practical application. Based on our analysis and the opinion of experts of the regional carrier, we recommend to keep the objective function (1) and the setting used in the previous section.

6.3 Results on Larger Instances

To further evaluate the performance of LPH, we artificially built larger size instances by combining pairs of the real-world instances used in the previous experiments. More precisely, we created a new instance $dd/94$ by merging the flights of the two instances $d/4$ and $d/9$, for each $d = 1, \dots, 7$. In our application, both overnight rest exceptions and short-term maintenance exceptions are taken into account. Thus, to keep these constraints on the larger size instances, and still ensure they are feasible, we selected and removed a few flights from some instances. The new instances contain between 240 and 296 flights. To have reasonable numbers of flight connections, each large instance includes all the existing connections between flights in a same

day, but no connection between flights in different days. In this way, the new instances keep the structure of the real-world ones but on a double number of airports.

We applied LPH to the new instances using the standard configuration, i.e., three departure time options and ± 10 minutes changes from the original departure time, exactly as done in Section 6.1 (Table 5). The new results are reported in Table 10. In addition to the results obtained by LPH, we report the results obtained without retiming so as to see the improvement that can be achieved. For the case without retiming, we executed the first two phases of the method developed in [9]. Table 10 shows, for the case without retiming and for the LPH algorithm, the instance name, the corresponding number $\#f$ of flights, and the number $\#conn$ of flight connections. Similarly to what was done in Section 6.1, the table reports the number $\#a$ of aircraft, the number $\#c$ of crew members and the number $\#ch$ of aircraft changes needed in the solution found, the lower bound (LB) obtained at the end of the first phase, the upper bound (UB) obtained at the end of the second phase, the percentage gap (gap%) between upper and lower bounds, the computing times (t_{LB} and t_{UB}) of the first and second phases, respectively, and the whole computing time (time). Moreover, for the LPH algorithm, we report the percentage improvement with respect to the solution without retiming. Finally, in the last row, we display average values for the percentage gap, total computing time and percentage improvement.

Inst.	#f	#conn	Without Retiming								With Retiming										
			#a	#c	#ch	LB	UB	gap%	t_{LB}	t_{UB}	time	#a	#c	#ch	LB	UB	gap%	t_{LB}	t_{UB}	time	impr%
11/94	240	2724	29	48	12	48181	48381	0.41	42	45	87	28	45	12	44818	45157	0.75	1556	1800	3356	6.66
22/94	270	3393	32	56	13	55955	56155	0.36	37	1800	1837	30	51	16	51366	52062	1.34	1801	1800	3601	7.29
33/94	264	3391	28	51	12	51572	51579	0.01	34	1800	1834	28	48	12	47962	48403	0.91	1458	1800	3258	6.16
44/94	256	3228	27	50	11	50120	50220	0.20	40	1800	1840	27	46	11	46691	46913	0.47	1580	1800	3380	6.59
55/94	264	3426	27	51	11	51085	51185	0.20	36	1800	1836	28	47	13	47631	48322	1.43	1681	1800	3481	5.59
66/94	296	4206	30	55	12	55021	55021	0.00	69	118	187	30	52	12	52436	52580	0.27	2864	1800	4664	4.44
77/94	248	2992	25	46	10	46453	46476	0.05	37	48	85	26	44	11	43925	44453	1.19	1760	1800	3560	4.35
Avg.								0.18			1100.9						0.91			3614.3	5.87

Table 10 Results on larger size instances without or with retiming.

We can conclude that, when including the flight retiming feature, the computing time for the approach LPH is about one hour, and the average percentage gap is below 1%. In addition, the obtained improvement with respect to the non-retiming solution is relevant (5.87% on average) and, for all instances, the number of crew members is reduced. Therefore, LPH can handle larger size instances in an effective way by considering a small set of alternative departure time options.

7 Conclusions and Future Research

This paper has addressed a new problem that embeds flight retiming in an airline scheduling problem that includes three phases (fleet assignment, aircraft routing and crew pairing). The goal is to use flight retiming to reduce the aircraft and crew costs, as well as the number of aircraft changes that can lead to delay propagation. The problem was motivated by a regional carrier operating about 150 flights every day within Canary Islands and other nearby regions. It is a problem to be solved during the planning stage, afforded about six months before the days of the operation. The integrated problem without retiming was solved to optimality in a previous work, and afterwards the regional carrier was also interested in investigating the extension in which flight departure times can slightly be moved. This extra freedom has been

investigated in other scheduling problems, including ground transportation among others, and there are also recent articles considering retiming in air transportation.

We proposed four heuristic approaches that were tested on real-world instances from the literature with three optional departure times for each flight. Although each algorithm has its advantages and drawbacks, the comparison showed that the best trade-off between solution improvement and computing time is achieved by an algorithm that selects flight copies based on the Linear Programming solution obtained within a column generation process. In addition, this research has confirmed that significant improvement can be obtained by considering retiming when solving the scheduling problem.

We conducted a sensitivity analysis of the impact of different sets of retiming options and different aircraft and crew costs on the performance of the most effective algorithm, and also performed experiments solving instances with up to 296 flights. The results show that the complexity of the problem increases significantly with the number of optional departure times per flight, especially due to the large number of flight connections available in our real-world instances. However, the algorithm was able to handle up to five departure time options in reasonable computing times. A challenging area of research for the future is to analyze the robustness of the obtained solutions under delay scenarios, and possibly improve them by inserting additional retiming options only for the most critical flights.

Acknowledgments

We thank Binter Canarias for motivating the research in this article.

References

1. M.B. Ahmed, W. Ghroubi, M. Haouari, and H.D. Sherali. A hybrid optimization-simulation approach for robust weekly aircraft routing and retiming. *Transportation Research Part C: Emerging Technologies*, 84:1–20, 2017.
2. M.B. Ahmed, F.Z. Mansour, and M. Haouari. A two-level optimization approach for robust aircraft routing and retiming. *Computers & Industrial Engineering*, 112:586–594, 2017.
3. M.B. Ahmed, F.Z. Mansour, and M. Haouari. Robust integrated maintenance aircraft routing and crew pairing. *Journal of Air Transport Management*, 73:15–31, 2018.
4. M.A. Aloulou, M. Haouari, and F.Z. Mansour. A model for enhancing robustness of aircraft and passenger connections. *Transportation Research Part C: Emerging Technologies*, 32:48–60, 2013.
5. B. Atasoy, M. Salani, and M. Bierlaire. An integrated airline scheduling, fleet and pricing model for a monopolized market. *Computer-Aided Civil and Infrastructure Engineering*, 29(2):76–90, 2014.
6. C. Barnhart and A. Cohn. Airline schedule planning: Accomplishments and opportunities. *Manufacturing & service operations management*, 6(1):3–22, 2004.
7. C. Barnhart and V. Vaze. Airline schedule optimization. In Barnhart C Belobaba P, Odoni A, editor, *The global airline industry, Second edition*, pages 189–222. Wiley, Chichester, UK, 2017.
8. E.K. Burke, P. De Causmaecker, G. De Maere, J. Mulder, M. Paelinck, and G.V. Berghe. A multi-objective approach for robust airline scheduling. *Computers & Operations Research*, 37(5):822–832, 2010.
9. V. Cacchiani and J.J. Salazar-González. Optimal solutions to a real-world integrated airline scheduling problem. *Transportation Science*, 51(1):250–268, 2017.
10. L. Cadarso, V. Vaze, C. Barnhart, and Á. Marín. Integrated airline scheduling: considering competition effects and the entry of the high speed rail. *Transportation Science*, 51(1):132–154, 2016.
11. M. Dunbar, G. Froyland, and C-L. Wu. An integrated scenario-based approach for robust aircraft routing, crew pairing and re-timing. *Computers & Operations Research*, 45:68–86, 2014.
12. O. Faust, J. Gönsch, and R. Klein. Demand-oriented integrated scheduling for point-to-point airlines. *Transportation Science*, 51(1):196–213, 2017.

13. H. Gürkan, S. Gürel, and M.S. Aktürk. An integrated approach for airline scheduling, aircraft fleet and routing with cruise speed control. *Transportation Research Part C: Emerging Technologies*, 68:38–57, 2016.
14. H. Jiang and C. Barnhart. Dynamic airline scheduling. *Transportation Science*, 43(3):336–354, 2009.
15. D. Klabjan, E.L. Johnson, G.L. Nemhauser, E. Gelman, and S. Ramaswamy. Airline crew scheduling with time windows and plane-count constraints. *Transportation science*, 36(3):337–348, 2002.
16. S. Lan, J-P. Clarke, and C. Barnhart. Planning for robust airline operations: Optimizing aircraft routings and flight departure times to minimize passenger disruptions. *Transportation science*, 40(1):15–28, 2006.
17. A. Mercier and F. Soumis. An integrated aircraft routing, crew scheduling and flight retiming model. *Computers & Operations Research*, 34(8):2251–2265, 2007.
18. O.O. Özener, M.Ö. Matoglu, G. Erdoğan, M. Haouari, and H. Sözer. Solving a large-scale integrated fleet assignment and crew pairing problem. *Annals of Operations Research*, 253(1):477–500, 2017.
19. N. Safaei and A.K.S. Jardine. Aircraft routing with generalized maintenance constraints. *Omega*, 80:111–122, 2018.
20. J.J. Salazar-González. Approaches to solve the fleet-assignment, aircraft-routing, crew-pairing and crew-rostering problems of a regional carrier. *Omega*, 43:71–82, 2014.
21. S. Shao, H.D. Sherali, and M. Haouari. A novel model and decomposition approach for the integrated airline fleet assignment, aircraft routing and crew pairing problem. *Transportation Science*, 51(1):233–249, 2015.
22. H.D. Sherali, K-H. Bae, and M. Haouari. 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, 2013.
23. H.D. Sherali, K-H. Bae, and M. Haouari. An integrated approach for airline flight selection and timing, fleet assignment and aircraft routing. *Transportation Science*, 47(4):455–476, 2013.
24. R. van Lieshout, J. Mulder, and D. Huisman. The vehicle rescheduling problem with retiming. *Computers & Operations Research*, 96:131–140, 2018.