Les Cahiers du GERAD ISSN: 0711–2440

Airline fleet assignment with stochastic demand and re-fleeting recourse

D. Lasalle Ialongo, G. Desaulniers,
F. Soumis

G–2015–135

December 2015

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

Airline fleet assignment with stochastic demand and re-fleeting recourse

David Lasalle Ialongo

Guy Desaulniers

François Soumis

GERAD & Mathematics and Industrial Engineering Department, Polytechnique Montréal, Montréal (Québec) Canada, H3C 3A7

david.ialongo@gmail.com
guy.desaulniers@gerad.ca
francois.soumis@gerad.ca

December 2015

Les Cahiers du GERAD

G–2015–135

Abstract: Given a flight schedule and a set of aircraft of different types, the airline fleet assignment problem (FAP) consists of assigning an aircraft type to each flight with the objective of maximizing the expected profits. Typically, the expected revenues are computed using the average demand for each potential passenger itinerary (this case is referred to as the FAP with deterministic demand). In this paper, we assume that the demand is stochastic and we address the FAP with stochastic demand. We propose a two-stage stochastic optimization model with recourse where an initial fleet assignment is implemented in the first stage and re-fleeting of pre-defined flight leg sequences can be performed in the second stage to face deviations from the average demand. Demand stochasticity is modeled using a limited set of demand scenarios. Computational results obtained on instances derived from a real-world flight network involving up to 5,180 flight legs show that the resulting model can be solved by a commercial mixed integer programming solver in reasonable computational times (up to an average of 12 hours) and that the computed solutions can yield significant additional expected profits compared to those derived from the solutions of the FAP with deterministic demand.

Key Words: Airline fleet assignment, stochastic passenger demand, two-stage stochastic optimization, re-fleeting recourse.

Acknowledgments: The authors are very grateful to the Natural Sciences and Engineering Research Council of Canada that partially funded this work under the grant 157935-2012. They would also like to thank Mathilde Peyrega who initiated this projet during an internship at the GERAD.

1 Introduction

Airline fleet assignment consists of assigning an aircraft type to each flight leg of a given schedule such that the flow of aircraft per type is balanced in each station at each time, fleet availability is satisfied, and expected profits are maximized. Given that the flight schedule of an airline is typically regular from one week to another in a season, the fleet assignment problem (FAP) is often solved only once per season, considering a representative one-week schedule and an average passenger demand (for airlines operating the same schedule every day, a one-day schedule is considered). The FAP is solved three to four months before the beginning of the season. Even if the same flight schedule is offered each week, passenger demand may vary from one week to another. In this case, the fleet assignment computed for the whole season might not be optimal for a given week. Re-fleeting (i.e., changing the aircraft type assigned to some flight legs) is sometimes performed by the airline during the booking period when the forecasted demand becomes more accurate. However, in the last few weeks prior to the operations, re-fleeting is rather limited because it may be in conflict with other planned activities such as aircraft maintenance and crew scheduling. Nevertheless, re-fleeting within the same aircraft type family is often acceptable because it yields little drawbacks (crew members are usually qualified for all aircraft types in a family). Re-fleeting involving types of different families may also be performed on certain sequences of flights when it is possible to retrieve a feasible aircraft flow and a feasible crew schedule.

This paper addresses the FAP with stochastic demand taking into account the re-fleeting possibilities that can occur during the booking period. For this model, we propose a two-stage stochastic optimization model that considers limited re-fleeting recourse actions. The re-fleeting possibilities, within the same aircraft type family or not, are identified a priori from a pre-computed solution to the FAP. The solutions to the stochastic model are robust against demand variations in the sense that they offer feasible re-fleeting opportunities where they might be profitable.

1.1 Literature review

The FAP is a classical optimization problem in air transportation that has been studied for more than forty years. The fleet assignment model introduced by Hane et al. (1995) is at the basis of several subsequent works, including the itinerary-based fleet assignment model of Barnhart et al. (2002, 2009) that computes the expected revenues through decision variables determining which itineraries will be chosen by the passengers (corresponding to a system-optimization model where the passenger choices are guided by the airline) and the bilevel optimization approach of Dumas et al. (2009) that evaluates the expected revenues using an external passenger flow model (corresponding to a user-optimization model where the passengers make their own decisions). The FAP tackled by Dumas et al. (2009) can be seen as a FAP with stochastic demand because the passenger flow model of Dumas and cois Soumis (2008) that they use considers demand stochasticity. Nevertheless, their approach does not handle re-fleetings. As surveyed by Klabjan (2005) and Sherali et al. (2006), several other extensions to the basic FAP have also been studied. We do not review them here because they are not of direct interest for this paper.

The literature related to the FAP with possible re-fleeting can be divided in two categories. The first one concerns only the re-fleeting subproblem that aims at modifying a known fleet assignment and the second the whole problem that consists of finding a complete fleet assignment. Let us start by reviewing the works in the first category.

Berge and Hopperstad (1993) introduce the Demand Driven Dispatch system to propose dynamically fleet assignment changes within the same aircraft type family before the day of operations. To identify these changes, this system takes into account the most recent demand forecasts and the current bookings for all flight legs. They report a profit increase of up to 5% when applying their system. Talluri (1996) improves this approach by developing an algorithm which guarantees finding a profitable aircraft type swap if one exists during a day. However, this algorithm considers a maximum of two aircraft types.

Jarrah et al. (2000) develop a re-fleeting model to identify a sequence of re-fleeting solutions, where one solution contains a strict subset of the changes proposed in the preceding solution. In this way, the airline can choose the desired level of modifications to bring to the current fleet assignment. Bish et al. (2004) study the benefits of an approach, called Demand Driven Swapping, that swaps aircraft types on compatible loops, i.e., back-and-forth pairs of flight legs starting from the same station and with similar departure and arrival times. Sherali et al. (2005) propose a Demand Driven re-fleeting model for a single family of aircraft. They consider a deterministic demand per passenger itinerary and incorporates into their model the passenger flow model of Barnhart et al. (2002) without passenger recapture.

Jiang (2006) and Warburg et al. (2008) present a mixed integer programming (MIP) model to modify flight departure times and fleet assignment during the booking period. They also integrate the passenger flow model of Barnhart et al. (2002) without passenger recapture. The results of Warburg et al. (2008) show that re-fleeting is mostly responsible for the observed profit increase. Jiang and Barnhart (2009) improve the model of Jiang (2006) by assuming a complete recapture of the passengers between the itineraries of the same market and no recapture between the itineraries of different markets. They report a profit increase of 2.28% when applying their methodology.

The following papers fall in the second category as they deal with the whole problem. Sherali and Zhu (2008) further develop the work of Sherali et al. (2005) by presenting a two-stage stochastic optimization model. In the first stage, an aircraft family is assigned to each flight leg while, in the second stage (representing the re-fleeting process), a specific aircraft type within the selected family is assigned to each leg. Several demand scenarios are considered to model the stochastic nature of the passenger demand and an L-shaped method is applied to solve the proposed model. The authors report profit increases varying between 1.1 and 1.7% compared to solutions computed using a deterministic model based on average demand. Pilla et al. (2008) introduce a similar two-stage model but propose a statistical approach to estimate the expected profits. Pilla et al. (2012) complete this research by developing a cutting plane algorithm where the cuts approximate the expected profit function. Their computational results show faster computational times than an L-shaped method and similar profit increases. These computational times remain, however, very long: more than 5 days for an instance involving 2,538 flight legs and 60 demand scenarios.

Starting from an initial schedule, Jiang and Barnhart (2013) propose an integer programming model to design a robust de-banked flight schedule for a hub-and-spoke network together with an initial fleet assignment that will favor, if required, modifications to flight departure times and fleet assignment using the method of Jiang and Barnhart (2009). This model aims at maximizing potential revenues by increasing the number of potential itineraries that would become feasible if some flexibility was added to certain flight departure times. It does not take into account aircraft costs and assumes that all aircraft types operate at the same speed and require the same connection times. To incorporate flight departure time flexibility, they consider in their model flight leg copies with different departure times up to 30 minutes before or after the planned departure time. They assume that the revenues from each itinerary are the same independently of the flight copies selected for this itinerary and also of the other offered itineraries. Their computational results show that using a robust schedule computed by their model yields additional profits compared to a standard flight schedule.

1.2 Contributions

As discussed above, very few papers have addressed the FAP with stochastic demand taking into account the possibility of re-fleeting during the booking period. The two-stage optimization approaches of Sherali and Zhu (2008) and Pilla et al. (2008, 2012) limit the aircraft changes within the same aircraft family and requires impractical computational times. The robust-optimization approach of Jiang and Barnhart (2013) does not consider directly the stochastic demand, but aims at maximizing re-fleeting and re-timing recourse opportunities without evaluating the possible profits from these recourses.

In this paper, we propose a two-stage stochastic optimization approach that is not restricted to re-fleetings within the same aircraft family. It is rather based on a limited set of possible re-fleetings that are identified a priori from an initial solution to the FAP. Each possibility in this set is defined by a sequence of flight legs (possibly a single leg) and a pair of aircraft types that can both be assigned to this sequence and be feasibly swapped during the booking period. A swap is considered feasible if it is possible to retrieve after the swap a feasible aircraft schedule and a feasible crew schedule for each aircraft type. The stochasticity of the demand is represented by a set of demand scenarios. The resulting model corresponds to a mixed integer program that can be solved directly by a commercial solver such as CPLEX when the number of scenarios considered is not too large. Our computational results on instances derived from a real-life one-week instance involving up to 5,180 flight legs, 15 aircraft types, and 71,000 passenger itineraries show that the post-processor can yield expected profits increases of about 3% while requiring less than 12 hours of computational time.

The rest of this paper is structured as follows. The FAP with stochastic demand is stated in Section 2. The proposed two-stage stochastic optimization model is introduced in Section 3 before describing the solution algorithm in Section 4. Next, computational results on instances derived from a real-world dataset are reported in Section 5. Finally, conclusions are drawn in Section 6.

2 Problem statement

The FAP with deterministic demand can be stated as follows. Consider an airline flight network servicing a set S of stations (airports). Let L be the set of flight legs operated in this network in a cyclic schedule that spans a time period (e.g., one week). This schedule is repeated over several consecutive time periods (the operations horizon) and the computed solution must, therefore, be repeatable period after period. A leg $l \in L$ can also be denoted by the triplet (o, d, t), where o and d represent the origin and destination stations of l, and t its departure time.

To operate this schedule, the airline possesses aircraft of different types. We denote by F the set of aircraft types and by n_f the number of aircraft available in fleet $f \in F$. Furthermore, let $F_l \subseteq F$ be the subset of aircraft types that can be assigned to leg $l \in L$ (i.e., those having sufficient autonomy, capacity, etc.). Assigning an aircraft of type $f \in F_l$ to a leg $l \in L$ incurs a cost C_{fl} and yields a flight duration h_{fl}. The seats available in an aircraft are partitioned into fare classes. Let U be the set of fare classes and cap^u_{fl}, $u \in U$, $l \in L$, $f \in F_l$, the number of seats in class u on leg l if fleet f is assigned to it.

For this schedule, the airline earns revenues by selling tickets to passengers. Let I be the set of itineraries that can be requested by the passengers. An itinerary is defined by a sequence of flight legs in L and a fare class in U for each leg. For each itinerary $i \in I$, its average demand d_i (number of passengers requesting it) over the operations horizon is assumed to be known as well as the average price p_i paid by a passenger. When passengers cannot obtain a ticket on their preferred itinerary because of a lack of available seats of the appropriate fare class on a leg of this itinerary, they most likely try to buy a ticket on a different itinerary. In this case, we say that the passengers are spilled and recaptured. We model the spill and recapture process by defining for each pair of itineraries $i, j \in I$, $i \neq j$, a parameter r_{ij} that indicates the proportion of the passengers that are spilled from i onto j when they are rejected on i. Note that $\sum_{j \in I} r_{ij}$ can be less than one to model passengers opting for another airline or a different mode of transportation. Given the aircraft type assigned to each flight leg $l \in L$, a passenger flow model (such as the one proposed in Dumas and cois Soumis (2008)) can be used to determine the average number of tickets sold for each itinerary $i \in I$ and, therefore, the total expected revenues from this flight schedule.

The FAP with deterministic demand consists of assigning an aircraft type $f \in F_l$ to each flight leg $l \in L$ such that the expected profits (expected revenues minus costs) are maximized, aircraft flow conservation by aircraft type is satisfied in every station at every time, a minimum connection time is imposed between any two consecutive legs to be assigned to the same aircraft, and the maximum number of aircraft of each type is not exceeded.

As mentioned above, the considered flight schedule is assumed to be operated over several consecutive time periods. Typically, the demand varies from one time period to another. In consequence, in the FAP with stochastic demand, we consider the demand as a set of stochastic variables D_i, one for each passenger itinerary $i \in I$. We assume that the demand variables D_i follow known probability distributions (e.g., truncated normal distributions) and that these distributions are not necessarily independent. The computation of the expected revenues must be done using these probability distributions or a suitable approximation of them (e.g., a set of demand scenarios).

To temper the effect of demand variability, the airlines often proceed to re-fleeting during the booking period. Each re-fleeting decision targets a specific period of the operations horizon. When crews have been scheduled for this period, re-fleeting possibilities are limited as they must be compatible with the planned crew schedules. As mentioned in Section 1.1, previous works have focused on exchanging aircraft types within the same family because crew members are, in general, qualified for all aircraft types in the same family. In this case, the crew schedules can remain the same: the crew members just fly a different aircraft type. Other re-fleeting opportunities can be considered. Indeed, for two (short) sequences of flight legs starting and ending at the same stations at approximately the same departure and arrival times, it might be feasible to switch both the aircraft and the crew on these sequences even if the aircraft belong to different aircraft families. In this case, the crew schedules change: the crew members work on different flight legs but on the same aircraft. Such a switching is feasible for the crews if each sequence is assigned to a single crew and both sequences exhibit the same characteristics with regards to the crew scheduling regulations (e.g., they have similar departure times, arrival times, and total flying times). Because the crew schedules are unknown when solving the FAP, it is not possible to determine if a given flight sequence will be assigned to a single crew. Nevertheless, many sequences can be identified as such a priori: for example, it can be a two-leg sequence starting from station s_1, visiting station s_2 and returning to s_1 in the same day, where s_1 is a hub corresponding to a crew base and s_2 is a station with a single flight per day. Furthermore, it might be possible to impose in the crew pairing (or scheduling) problem that the legs of a sequence be flown by the same crew. In general, such constraints should not deteriorate much the crew costs if the leg sequences are well selected.

Figure 1 illustrates an example of an aircraft swap between two flight leg sequences. It involves three stations denoted HUB, A and B. The first sequence is composed of two flight legs: one from HUB to A departing at 7h00 and one from A to HUB arriving at 11h20. Initially, the demands on these two legs (derived from the itinerary passenger demands) were estimated at 120 and 122, respectively. In consequence, an aircraft with a 130-seat capacity was assigned to this sequence. The second sequence is similar to the first sequence but visits station B instead of station A. It starts at 7h05 and ends at 11h15. At the time of planning, its demands were estimated at 142 and 136 passengers for its two legs and a 150-seat aircraft was assigned to it. During the booking period, the anticipated demands are revised to an increased 135 and 138 passengers for the legs of the first sequence and to a decreased 125 and 122 passengers for the legs of the second sequence. Clearly, swapping the aircraft initially assigned to these two sequences would certainly benefit the airline.

Note that aircraft swaps can involve more than two sequences. For instance, consider three sequences q_1, q_2 and q_3 to which three different fleet f_1, f_2 and f_3 have been assigned, respectively. It might be possible to

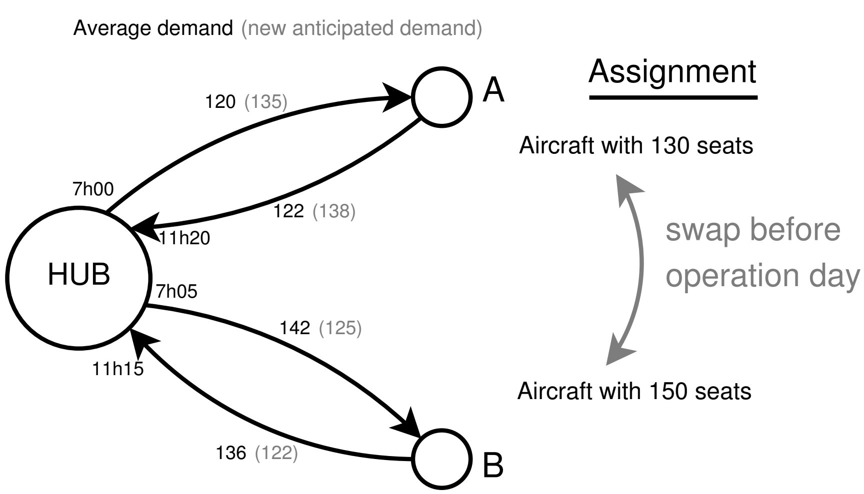

Figure 1: Aircraft swap between two compatible pairs of flight sequence and aircraft type

reassign the aircraft of q_1 to q_2, that of q_2 to q_3 and, finally, that of q_3 to q_1. In Section 3, we define formally the feasibility of an aircraft swap.

The FAP with stochastic demand considered in this paper is the same as the FAP with deterministic demand stated above except that the passenger demand for each itinerary is stochastic and the expected revenues must be computed taking into account that leg sequence re-fleeting can occur during the booking period.

It should be noted that, in the example of Figure 1, the aircraft swapping is possible because the initial fleet assignment proposed aircraft of different types on the two flight leg sequences. This is to be expected from a solution to the FAP with stochastic demand for sequences with relatively high demand variability and between which it is possible to exchange the assigned aircraft.

3 A two-stage stochastic optimization model

In this section, we introduce a two-stage stochastic optimization model that considers a stochastic demand and re-fleeting recourse actions. As in the solution method proposed by Dumas et al. (2009), this model is integrated into a loop with a passenger flow model to compute the expected passenger revenues.

The proposed model can be classified as a multi-commodity network flow model that involves one network per aircraft type. In most fleet assignment models, the nodes of the underlying networks correspond to flight leg departures and arrivals, and the arcs to flight legs and ground waitings. We use more compact networks where nodes correspond to leg banks and arcs to leg sequences and ground waitings. Let us start by defining the concepts of leg banks and leg sequences.

First, we define a bank of flight legs that is commonly known as a sequence of consecutive legs arriving at a station followed by a sequence of consecutive legs departing from the same station. Here, we use a sligthly different definition because we associate the banks with a fleet and the fleet are not yet assigned to the legs. Consider a fleet $f \in F$ and let $L_f \subseteq L$ be the set of legs that can be operated by an aircraft in this fleet. For each leg $l \in L_f$, we denote by t_l^D its departure time and by t_{lf}^A its arrival time, which is fleet-dependent (due to possible different flying speeds), extended by a minimum connection time. Let $s \in S$ be a station and let $L_{sf}^D \subset L_f$ and $L_{sf}^A \subset L_f$ be the subsets of legs departing from s and arriving at s, respectively. The departure times t_l^D of the legs $l \in L_{sf}^D$ and the arrival times t_{lf}^A of the legs $l \in L_{sf}^A$ are sorted together in chronological order. These times wrap around the time period and their corresponding legs form a cyclic list that can be divided into banks. A *bank* is composed of a maximal sequence of consecutive arrivals in this list followed by a maximal sequence of consecutive departures. In consequence, a bank starts by an arrival that is preceded by a departure in the cyclic list, and ends with a departure that is succeeded by an arrival.

Figure 2 illustrates the banks at a station for an aircraft type. In this figure, the diagonal arrows represent flight legs arriving or departing from this station (the arrival times include a minimum connection time). These legs are divided into two banks. The first bank contains three arrivals and two departures, whereas the second contains two arrivals followed by three departures. Notice that a bank favors multiple connection opportunities between its arrivals and departures as the connection between any pair of arrival and departure in the bank is feasible.

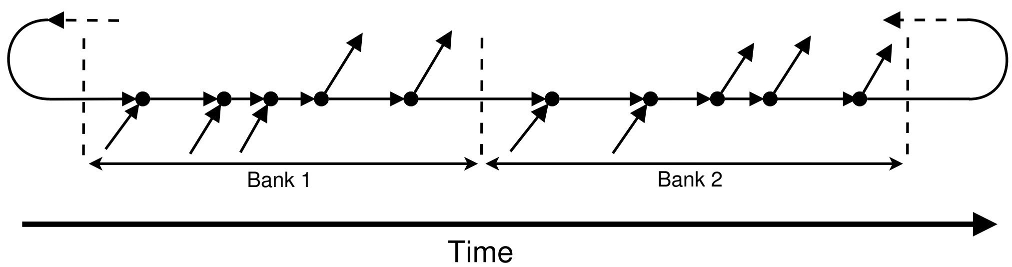

Figure 2: Example of banks at a station for an aircraft type

Let K_{sf} be the ordered (cyclic) set of banks at station $s \in S$ associated with aircraft type $f \in F$. For a given bank $k \in K_{sf}$, we denote by k^- and k^+, its predecessor and successor banks in K_{sf}, respectively. When a station contains a single bank k, then $k^+ = k^- = k$. Furthermore, for bank k, we denote by t_k^D and t_k^A, its earliest departure time and its latest arrival time, respectively. We say that bank $k \in K_{sf}$ begins and ends at times t_k^D and t_k^A, respectively. For ease of exposition only, we assume that $t_k^A \geq t_k^D$ for all banks $k \in K_{sf}$, $s \in S$, $f \in F$, i.e., there is no bank that spans the start/end of the time period.

Given an initial solution to the FAP, one can derive a set of aircraft routes that satisfy aircraft availability per fleet. Each route can be divided into disjoint (relatively short) sequences of consecutive flight legs such that, for each sequence, all its legs will be assigned to the same crew. Note that a sequence can contain a single leg and that each leg in L belongs to a single sequence. Let Q be the set of all sequences. In Section 4, we propose an algorithm to construct the sequences in Q, which are used to determine the possible aircraft swaps as follows. For each sequence $q \in Q$, we denote by s_q^D and s_q^A its origin and destination stations, respectively, and by F_q the set of aircraft types that can be assigned to it. Furthermore, for each fleet $f \in F_q$, we denote by t_q^D and t_{qf}^A the departure time of its first leg and the arrival time of its last leg, respectively, and by k_{qf}^D and k_{qf}^A the origin and destination banks of q, respectively, when q is assigned to an aircraft of type f. Let $q_1, q_2 \in Q$ be two leg sequences. We say that q_1 and q_2 are *compatible with respect to an aircraft type* $f \in F$ if

1. f can be assigned to both sequences q_1 and q_2, i.e., $f \in F_{q_1} \cap F_{q_2}$;
2. both sequences begin at the same station, i.e., $s_{q_1}^D = s_{q_2}^D$;
3. both sequences end at the same station, i.e., $s_{q_1}^A = s_{q_2}^A$;
4. for type f, both sequences belong to the same origin bank, i.e., $k_{q_1 f}^D = k_{q_2 f}^D$;
5. for type f, both sequences belong to the same destination bank, i.e., $k_{q_1 f}^A = k_{q_2 f}^A$;
6. both sequences have similar flying times and durations (as well as other similar characteristics if relevant for crew schedule feasibility) when assigned to fleet f.

Given that the banks are specific to each aircraft type, two sequences can be compatible for an aircraft type but not for another.

An aircraft swap between two sequences can be feasible if these sequences are compatible with respect to at least two aircraft types. Consider two aircraft: the first is of type f_1 and flies the three sequences $q_1^1 - q_2^1 - q_3^1$ consecutively in this order and the second is of type f_2 and operates the sequences $q_1^2 - q_2^2 - q_3^2$ consecutively. If the sequences q_2^1 and q_2^2 are compatible with respect to both aircraft types f_1 and f_2, then the aircraft can be swapped on these two sequences, that is, the two aircraft would then be assigned to $q_1^1 - q_2^2 - q_3^1$ and $q_1^2 - q_2^1 - q_3^2$, respectively. These aircraft routes remain feasible because the compatible sequences q_2^1 and q_2^2 have the same departure and arrival banks for both fleet f_1 and f_2. From a crew scheduling point of view, the crews assigned to q_2^1 and q_2^2 can remain on the same sequence if f_1 and f_2 belong to the same aircraft family. On the other hand, the crews also need to be swapped with the aircraft if the aircraft are not in the same family. Given that the sequences have similar characteristics, the new schedules should be feasible.

Aircraft swaps involving more than two sequences are also possible. In this case, it is easy to see that not all pairs of sequences need to be compatible with respect to more than one aircraft type.

Let $Q_f \subseteq Q$ be the subset of sequences that can be assigned to aircraft type $f \in F$. For each type $f \in F$, the sequences in Q_f are partitioned into a set E_f of clusters of pairwise compatible sequences of maximal size or containing a single sequence. Thus, each sequence $q \in Q_f$ belongs to a single cluster in E_f. The common departure and arrival banks of the sequences in cluster $e \in E_f$ are denoted k_e^D and k_e^A, respectively. A cluster in E_f whose cardinality is greater than or equal to 2 contains sequences which can be re-fleeted during the booking period. We denote by $Q^* \subseteq Q$ the subset of the sequences that can be re-fleeted, i.e., those that belong to a non-singleton cluster in E_f for at least one type $f \in F$.

The proposed model imposes aircraft balance per aircraft type at every station and every time in the time period considered. Consequently, to ensure that aircraft availability per type is met, one just need to count the number of aircraft on the ground (excluding the minimum connection time after an arrival) and in the air (including the minimum connection time after an arrival) at a given time $\bar{t}$. For each fleet $f \in F$, let $O_f \subseteq Q_f$ be the subset of sequences q in the air at time $\bar{t}$ when assigned to fleet f, i.e., such that $t_q^D < \bar{t} \leq t_{qf}^A$. If $\bar{t}$ corresponds to a time when there are no aircraft in the air (e.g., $\bar{t}$ corresponds to the middle of a night when there are no night flights), then $O_f = \emptyset$ and one can only count the number of aircraft on ground waiting arcs overlapping $\bar{t}$. Otherwise, because nodes in the networks correspond to banks that are associated with time intervals rather than specific times, one must be careful when counting the number of aircraft on the ground at time $\bar{t}$. Consider a station $s \in S$ and an aircraft type $f \in F$. If $\bar{t}$ falls in the time interval associated with a bank $k \in K_{sf}$, i.e., $t_k^D < \bar{t} \leq t_k^A$, then the number of aircraft of type f on the ground in station s at time $\bar{t}$ is given by the sum of the number of aircraft on the ground before the start of bank k and the number of aircraft that arrived in bank k prior to $\bar{t}$ but that did not depart from s before $\bar{t}$. In this case, let $\bar{k}_{sf} = k^-$ and denote by G_{sf}^D and G_{sf}^A the subsets of legs in $\bar{k}_{sf}$ departing before $\bar{t}$ and arriving before $\bar{t}$, respectively. Otherwise, if $\bar{t}$ falls between the intervals associated with two consecutive banks $k_1, k_2 \in K_{sf}$, i.e., $t_{k_1}^A < \bar{t} \leq t_{k_2}^D$ (where $k_1 = k_2$ if K_{sf} contains a single bank), then the number of aircraft of type f on the ground in station s at time $\bar{t}$ is given by the number of aircraft on the ground at the end of bank k_1. In this case, we set $\bar{k}_{sf} = k_1$.

The objective function consists of maximizing the expected profits that are computed as the difference between the expected revenues and the operation costs. Let C_{qf} be the operation costs of assigning an aircraft of type $f \in F_q$ to leg sequence $q \in Q$. The expected revenues depend on the stochastic demand variables D_i for each passenger itinerary $i \in I$. As it is often used in stochastic optimization, we propose to approximate the probability distributions of the variables D_i using randomly generated demand scenarios. Let W be the set of scenarios, where a scenario indicates a possible demand realization for each itinerary $i \in I$. The probability that scenario $w \in W$ occurs is denoted p^w.

For a demand scenario $w \in W$, the revenues yielded by a fleet assignment can be computed using the iterative algorithm introduced by Dumas et al. (2009). In fact, instead of considering expected revenues, Dumas et al. (2009) proposed to use expected revenue losses per leg with respect to the maximum revenues that could be achieved by assigning a fictitious aircraft of infinite capacity. The computation of these revenue losses is explained in Section 4. With these revenue losses, the objective function of the fleet assignment model consists of minimizing the sum of the operational costs and the expected revenue losses, which is equivalent to maximizing the total expected profits. Once a new fleet assignment is computed using the estimated revenue losses, these losses are revised and the process is repeated for a given number of iterations. In our model, we rather use expected revenue losses per leg sequence RL_{qf}^w, $w \in W$, $q \in Q$, $f \in F_q$, that are computed like the expected revenue losses per leg in the Dumas et al. (2009) approach.

The proposed model relies on the following three types of decision variables:

X_{qf}: Binary variable that takes value 1 if aircraft type $f \in F_q$ is assigned to leg sequence $q \in Q$ in the planned fleet assignment, and 0 otherwise;

Y_{fkk^+}: Nonnegative variable indicating the number of aircraft of type $f \in F$ on the ground between the consecutive banks k and k^+ in K_{sf} at station $s \in S$ in the planned fleet assignment solution;

Z_{qf}^w: Binary variable that takes value 1 if aircraft type $f \in F_q$ is assigned to leg sequence $q \in Q$ in scenario $w \in W$, and 0 otherwise.

The variables $\mathbf{X} = \left(X_{qf}\right)_{q\in Q, f\in F_q}$ and $\mathbf{Y} = \left(Y_{fkk^+}\right)_{f\in F, s\in S, k\in K_{sf}}$ are the first-stage variables, i.e., they define the planned fleet assignment to use at the start of the booking period. For each scenario $w \in W$, the second-stage variables $\mathbf{Z}^w = \left(Z_{qf}^w\right)_{q\in Q, f\in F_q}$ indicate the final fleet assignment resulting from the re-fleetings performed during the booking period if scenario w occurs.

The FAP with stochastic demand can be modeled using the following two-stage optimization model with recourse:

$$\min \qquad \sum_{w\in W} p^w \left[\sum_{q\in Q} \sum_{f\in F_q} (C_{qf} + RL_{qf}^w) Z_{qf}^w + \eta(\mathbf{X}, \mathbf{Z}^w) \right] \tag{1}$$

`subject to:`
$$\sum_{f \in F_q} X_{qf} = 1, \quad \forall q \in Q, \tag{2}$$
$$\sum_{\substack{e \in E_f:\\ k_e^A = k}} \sum_{q \in e} X_{qf} + Y_{fk^-k} - \sum_{\substack{e \in E_f:\\ k_e^D = k}} \sum_{q \in e} X_{qf} - Y_{fkk^+} = 0, \quad \forall s \in S, f \in F, k \in K_{sf}, \tag{3}$$
$$\sum_{q \in O_f} X_{qf} + \sum_{s \in S} \Big(\sum_{q \in G_{sf}^A} X_{qf} - \sum_{q \in G_{sf}^D} X_{qf} + Y_{f\bar{k}_{sf}\bar{k}_{sf}^+} \Big) \leq n_f, \quad \forall f \in F, \tag{4}$$
$$\sum_{f \in F_q} Z_{qf}^w = 1, \quad \forall q \in Q, w \in W, \tag{5}$$
$$\sum_{q \in e} Z_{qf}^w - \sum_{q \in e} X_{qf} = 0, \quad \forall f \in F, e \in E_f, w \in W, \tag{6}$$
$$X_{qf} \in \{0,1\}, \quad \forall q \in Q, f \in F_q, \tag{7}$$
$$Z_{qf}^w \in \{0,1\}, \quad \forall q \in Q, f \in F_q, w \in W, \tag{8}$$
$$Y_{fkk^+} \geq 0, \quad \forall s \in S, f \in F, k \in K_{sf}. \tag{9}$$

The objective function (1) minimizes the sum of the expected operation costs, revenue losses and re-fleeting penalties, where the expectation is computed over the set W of considered scenarios. The re-fleeting penalties, expressed by the term $\eta(\mathbf{X}, \mathbf{Z}^w)$, are introduced to avoid re-fleeting when a small gain can be realized. They are discussed below. The first three sets of constraints concern the first-stage decisions. Constraints (2) ensure that a valid aircraft type is assigned to each leg sequence. Constraints (3) impose flow conservation per aircraft type at each bank of the network. Aircraft availability per fleet is enforced through constraints (4). The next two sets of constraints ensure the feasibility of the possible re-fleetings in the second stage. Constraints (5) stipulate that one aircraft type must be assigned to each leg sequence for each scenario. Constraints (6) guarantee that any type assigned to a sequence in the second stage is either the type assigned in the first stage or a type assigned to a compatible sequence. Finally, binary and nonnegativity requirements on the decision variables are imposed through (7)–(9) (the integrality of the $\mathbf{Y}$ variables is implied by (7)). Note that the feasibility of the final fleet assignment $\mathbf{Z}^w$ for each scenario $w \in W$ ensues from the feasibility of the planned assignment $\mathbf{X}$ and the pairwise compatibility of the sequences in each sequence cluster $e \in E_f$, $f \in F$.

In practice, re-fleeting will not be implemented if it does not yield a sufficiently large expected gain. In consequence, we introduce re-fleeting penalties $\eta(\mathbf{X}, \mathbf{Z}^w)$ to ensure that re-fleeting is proposed only when it is really worth it. Two types of re-fleeting penalties can be considered: constant and fleet-and-sequence-dependent penalties. In the first type of penalties, a constant penalty α is incurred for each re-fleeting and, in this case,
$$\eta(\mathbf{X}, \mathbf{Z}^w) = \alpha \sum_{q \in Q^*} \sum_{f \in F_q} \max\{0, X_{qf} - Z_{qf}^w\}. \tag{10}$$

This expression can be linearized by introducing additional variables and constraints. Let V_q^w be a binary variable equal to 1 if re-fleeting occurs on leg sequence $q \in Q$ in scenario $w \in W$. Then $\eta(\mathbf{X}, \mathbf{Z}^w)$ is redefined in terms of these new variables $\mathbf{V}^w = \big(V_q^w\big)_{q \in Q}$ as
$$\eta(\mathbf{X}, \mathbf{Z}^w) \equiv \eta(\mathbf{V}^w) = \alpha \sum_{q \in Q^*} V_q^w, \tag{11}$$

and the following constraints are added to model (1)–(9):
$$V_q^w \geq X_{qf} - Z_{qf}^w, \quad \forall q \in Q^*, f \in F_q, w \in W, \tag{12}$$
$$V_q^w \in \{0,1\}, \quad \forall q \in Q^*, w \in W. \tag{13}$$

Constraints (12) ensure that, for each sequence $q \in Q^*$ and scenario $w \in W$, the variable V_q^w is set to 1 if re-fleeting occurs on sequence q in scenario w, i.e., if $X_{qf} = 1$ and $Z_{qf}^w = 0$ for a fleet $f \in F_q$, whereas the minimization of the re-fleeting penalties in the objective function forces V_q^w to be equal to 0 in any other situation.

In the second type of penalties, the penalty $\alpha_{qf_1f_2}$ for changing the aircraft type assigned to a sequence $q \in Q^*$ from $f_1 \in F_q$ in the first-stage assignment to f_2 in the second stage is fleet- and sequence-dependent. Such dependency can be useful when the impact of re-fleeting a sequence q from f_1 to f_2 is different from f_1 to f_3 ($f_2 \neq f_3$). For instance, if f_1 and f_2 are in the same aircraft family, then the schedule of the crew assigned to q does not need to change, while it has to change if f_1 and f_3 are not in the same family. Also, passengers that frequently take the same flight leg often prefer the same aircraft type as it ensures stability (same gate, same seat configuration, etc.). Consequently, re-fleeting from f_1 to f_2 might be preferable to re-fleeting from f_1 to f_3 if the former re-fleeting incurs less perturbations for the passengers than the latter. Finally, the penalty can depend on the characteristics of the sequence, e.g., on its duration or on its number of legs, which can reflect the level of perturbation experienced by the passengers. In this case,

$$\eta(\mathbf{X}, \mathbf{Z}^w) = \sum_{q \in Q^*} \sum_{f_1 \in F_q} \sum_{f_2 \in F_q \setminus \{f_1\}} \alpha_{qf_1f_2} \max\{0, X_{qf_1} + Z^w_{qf_2} - 1\}. \tag{14}$$

This expression can also be linearized by introducing new variables and constraints. Let $V^w_{qf_1f_2}$ be a binary variable equal to 1 if sequence $q \in Q^*$ is re-fleeted from type $f_1 \in F_q$ to type $f_2 \in F_q \setminus \{f_1\}$ in scenario $w \in W$ and 0 otherwise. The re-fleeting penalty term then rewrites as

$$\eta(\mathbf{X}, \mathbf{Z}^w) \equiv \eta(\mathbf{V}^w) = \sum_{q \in Q^*} \sum_{f_1 \in F_q} \sum_{f_2 \in F_q \setminus \{f_1\}} \alpha_{qf_1f_2} V^w_{qf_1f_2}, \tag{15}$$

where $\mathbf{V}^w = \left(V^w_{qf_1f_2}\right)_{q \in Q, f_1 \in F_q, f_2 \in F_q \setminus \{f_1\}}$ this time. The constraints to add are:

$$V^w_{qf_1f_2} \geq X_{qf_1} + Z^w_{qf_2} - 1, \quad \forall q \in Q^*, f_1 \in F_q, f_2 \in F_q \setminus \{f_1\}, w \in W, \tag{16}$$

$$V^w_{qf_1f_2} \in \{0, 1\}, \quad \forall q \in Q^*, f_1 \in F_q, f_2 \in F_q \setminus \{f_1\}, w \in W. \tag{17}$$

Constraints (16) ensure that a variable $V^w_{qf_1f_2}$ is set to 1 when re-fleeting from f_1 to f_2 occurs on sequence q in scenario w, i.e., if $X_{qf_1} = Z^w_{qf_2} = 1$. Otherwise, $V^w_{qf_1f_2}$ takes value 0 because the objective function seeks at minimizing its value.

4 Solution algorithm

The proposed solution algorithm requires an initial solution that is used to identify the leg sequences in Q. This initial solution can be obtained in several ways. It can be obtained by solving (exactly or heuristically) the FAP with deterministic demand or by adapting the solution used the previous year if the flight schedule is sufficiently similar. To solve the FAP with stochastic demand, we execute the algorithm described in Figure 3. This algorithm starts by creating the set of leg sequences Q. Then, it enters into a loop by computing first the expected revenue losses RL^w_{fq} for each scenario $w \in W$, each sequence $q \in Q$, and each aircraft type $f \in F_q$ that can be assigned to q. Next, it solves model (1)–(9) augmented by the appropriate re-fleeting penalties (defined by (11)–(13) or (15)–(17)) using a commercial MIP solver such as CPLEX. The computed solution proposes a fleet assignment $\mathbf{X}$ to implement, and, for each scenario $w \in W$, a final fleet assignment $\mathbf{Z}^w$ if scenario w is realized. For each scenario w, the expected revenues from the assignment $\mathbf{Z}^w$ are evaluated by solving the passenger flow model of Dumas and cois Soumis (2008) using the fixed-point algorithm they proposed. These revenues allow to estimate the expected profits of the fleet assignment $\mathbf{X}$, taking into account the possible re-fleetings. Once a stopping criterion is reached (a maximum number of iterations in our case), the loop is exited. If needed, a more extensive evaluation of the expected profits is performed using a larger set of demand scenarios that excludes those used for computing the solution (i.e., those in W). Note that the passenger flow model of Dumas and cois Soumis (2008) takes into account the demand stochasticity that still exists when re-fleeting possibilities are evaluated. Thus, as opposed to most stochastic optimization approaches that model stochasticity using a set of deterministic scenarios, our approach relies on stochastic scenarios. In the rest of this section, we provide further details on the selection of the leg sequences, the computation of the revenue losses, and the final evaluation of the expected profits.

In general, the approach used to determine the leg sequences should depend on the network topology, the flight schedule and the airline itself. It should try to maximize aircraft swapping opportunities. For

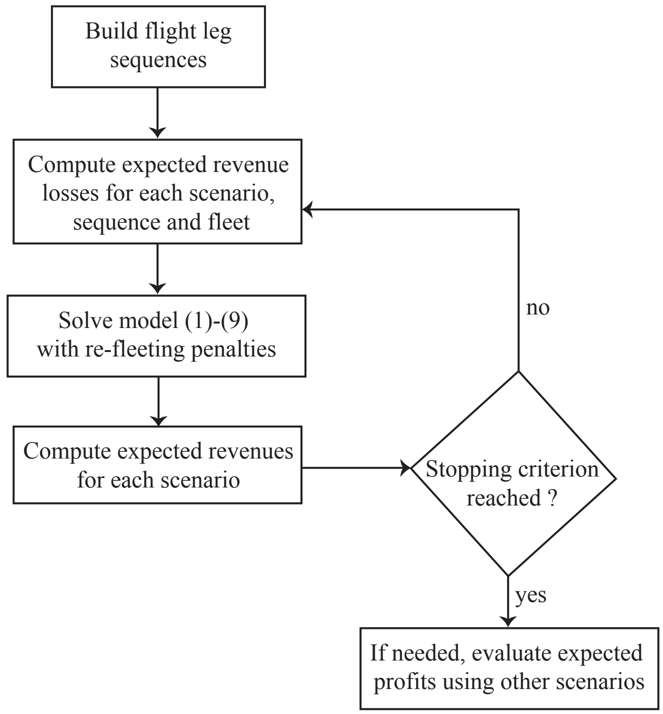

Figure 3: Overview of the algorithm

our tests, we used data provided by Air Canada whose flight schedules contain many two-leg loops that start and end at one of the major stations serviced by Air Canada: for example, loops Montreal-Boston-Montreal and Montreal-New York-Montreal. These loops have similar characteristics and are often scheduled at approximately the same time (to accommodate business travelers), giving the opportunity to swap their aircraft if needed. Furthermore, as Montreal is a crew base, there are high chances that the same crew will fly both legs of these loops. In consequence, the proposed algorithm exploits this observation and creates leg sequences that form two-leg loops starting and ending at the same station among the major stations. Legs that are not part of these loops form single-leg sequences.

First, given the initial FAP solution, a first-in, first-out heuristic is applied to compute aircraft routes for each aircraft type. Note that these routes can be determined using a more sophisticated procedure (e.g., that might consider maintenance constraints) if available. Then, each route is divided into leg sequences containing at most two legs. To favor creating two-leg sequences that start at the major stations, a priority β_s is given to each station $s \in S$. A two-leg sequence containing legs l_1 and l_2 is admissible if and only if $s^A_{l_2} = s^D_{l_1}$ and $\beta_{s^D_{l_1}} \geq \beta_{s^A_{l_1}}$. Algorithm 1 provides the pseudo-code of the algorithm used to create the leg sequences. Each route r is represented by its ordered list of legs $(l_1, l_2, \ldots, l_{n_r})$, where n_r indicates the number of legs in r. The first leg l_1 corresponds arbitrarily to the first leg flown at the beginning of the time period. Starting from the first leg of the route as the current leg l_i, the algorithm tries to build a two-leg sequence with the next leg on the route. If this is not possible (i.e., one of the conditions in Step 4 is met), a one-leg sequence is added to Q. Otherwise, a sequence with legs l_i and l_{i+1} is created. The algorithm then moves on to the next leg in the route to create a new sequence. The process repeats until all legs in all routes have been included in a sequence. Once the set of sequences is built, the subsets E_f, $f \in F$, of compatible sequences with respect to f are computed in Step 14 and the set Q^* of sequences that are compatible with at least another sequence is determined. In practice, Q^* contains almost only two-leg sequences. However, there might be two-leg sequences that are not compatible with any other sequence. In Steps 17 and 18, such a sequence $q \in Q \setminus Q^*$ is replaced in Q by two one-leg sequences to allow more flexibility during optimization.

Algorithm 1 Creation of the leg sequences in Q

for each aircraft route $r = (l_1, l_2, \ldots, l_{n_r})$ **do**
  $i := 1$
  while $i \leq n_r$ **do**
    if $i + 1 > n_r$ or $\beta_{s^A_{l_i}} < \beta_{s^D_{l_i}}$ or $s^A_{l_{i+1}} \neq s^D_{l_i}$ **then**
      $j := i$
    else
      $j := i + 1$
    if $i = j$ **then**
      Add one-leg sequence (l_i) to Q
    else
      Add two-leg sequence (l_i, l_j) to Q
    $i := j + 1$
for each fleet $f \in F$ **do**
  Construct the set E_f of compatible sequences with respect to fleet f
Compute subset Q^* of the sequences compatible with at least another sequence
for each two-leg sequence $q = (l_1, l_2) \in Q \setminus Q^*$ **do**
  $Q := Q \setminus \{q\}$
  $Q := Q \cup \{(l_1), (l_2)\}$

The revenue losses for a demand scenario $w \in W$ are computed as follows. Given a fleet assignment (either the initial one or the assignment $\mathbf{Z}^w$ computed at a given iteration of the algorithm), the passenger flow model of Dumas and cois Soumis (2008) is solved using their fixed-point algorithm for the corresponding demand. The solution to this model indicates the expected number of tickets sold on each itinerary $i \in I$ and therefore the total expected revenues. Then, for each leg $q \in Q$, the algorithm of Dumas and cois Soumis (2008) is run again $1+|F_q|$ times on a restricted version of the network: once with an infinite capacity assigned to sequence q to yield total expected revenues $\rho^w_{q,\infty}$ and, for each fleet $f \in F_q$, once with the capacity of an aircraft of type f assigned to q to yield expected revenues ρ^w_{qf}. The revenue loss expected when assigning an aircraft of type $f \in F_q$ to sequence $q \in Q$ is given by $RL^w_{qf} = \rho^w_{q,\infty} - \rho^w_{qf}$. For a sequence q, the restricted network is defined as follows. Let I_q be the subset of itineraries that contain a leg in q and all itineraries that can capture spill from these itineraries. Let L_q be the set of legs in the itineraries of I_q. The network is restricted to the legs in L_q and the itineraries in I_q. However, it considers as constant the flow (from the solution to the complete network) of the other itineraries on the legs in L_q. Such a restricted network is used to speed up the computational time. Compared to using a complete network, Dumas and cois Soumis (2008) obtained a speed up factor of 400 and measured an acceptable average error of 1.3% on the revenue losses.

In the next section, we will compare different solutions to the FAP with stochastic demand that were obtained by considering a different number of demand scenarios. To perform a fair comparison, we evaluate each solution using a set of scenarios $\hat{W}$ that differ from the ones used during the optimization process. The evaluation of a fleet assignment $\bar{\mathbf{X}}$ for a given scenario $w \in \hat{W}$ is performed in three steps as follows. First, the revenue losses RL^w_{qf} for each sequence $q \in Q$ and each aircraft type $f \in F_q$ are computed as described above. Second, a mixed integer program is solved using CPLEX to identify the re-fleetings to perform if scenario w occurs. This program corresponds to model (1)–(9), augmented by (11)–(13) or (15)–(17), but restricted to a single scenario, namely, w and to the evaluated fleet assignment $\bar{\mathbf{X}}$ (i.e., $\mathbf{X} = \bar{\mathbf{X}}$). The solution to this modified model provides a final fleet assignment $\mathbf{Z}^w$ if scenario w is realized. Finally, the passenger flow model is solved again to evaluate the expected profits resulting from fleet assignment $\mathbf{Z}^w$. The expected profits of the fleet assignment $\bar{\mathbf{X}}$ is then obtained as a weighted average of the expected profits yielded by the scenarios in $\hat{W}$.

5 Computational experiments

In this section, we report the results obtained by the solution algorithm that we propose for solving the FAP with stochastic demand. Before presenting these results, we describe the instances used for our tests, including the generation of the demand scenarios.

5.1 Instances

For our experiments, we use the flight network used by Dumas et al. (2009) that corresponds to a part of the Air Canada network in 2002. This network involves 5,180 flight legs over a one-week time period and 205 aircraft of 15 different types. The passengers are distributed over 71,844 itineraries (23,948 triplets of itineraries where the itineraries in a triplet only differ by the fare class). To obtain a larger set of test instances of various sizes, we built smaller flight networks by extracting subsets of this complete network. To create a network, we select first a subset of consecutive days in the week, then the flight legs, the passenger itineraries, and the aircraft. Initially, all legs with a departure time in the selected days are considered. However, there might not exist a feasible solution for this set of legs because of the aircraft flow conservation constraints and the need to produce a solution that can be repeated week after week. After solving a FAP with deterministic demand where a leg might remain uncovered at a high penalty cost, the uncovered legs are removed from the flight schedule. From the set of itineraries, we retain only those whose flight legs have been selected. However, to compensate the demand of the itineraries that have been removed but that contain one leg in the selected set, the demand on each selected itinerary is increased in such a way that the competition for a ticket on each flight leg remains similar to that occurring in the complete network. Furthermore, the spill and recapture rates from one itinerary to another are also adjusted. Finally, the number of aircraft available in each fleet and their fixed costs are modified. Because most aircraft are used each day, the number of available aircraft does not change much for networks spanning just one or a few days. The fleet have been reduced manually and gradually until reaching a number of aircraft that provides a minimal leeway. The aircraft fixed costs have been set according to the number of days in the time period.

With this approach, we generated 8 new flight networks: 3 spanning one day (Monday, Thursday and Sunday), 3 spanning two days (Monday-Tuesday, Thursday-Friday, and Saturday-Sunday), and 2 spanning three days (Monday to Wednesday and Friday to Sunday). In all these networks, a total of 194 aircraft were made available. The numbers of flight legs ($|L|$) and itineraries ($|I|$) in each network are given in Table 1, together with the number of aircraft types ($|F|$) and the total number of aircraft ($\sum_{f \in F} n_f$).

For each of the nine flight schedules, we produced an initial fleet assignment by solving a FAP with deterministic demand (just one evaluation of the leg-based revenues was performed). Then, we built leg sequences using Algorithm 1. For this algorithm, the Toronto station, which is the most frequented by Air Canada, received priority 1, Montreal and Vancouver had priority 2, and Calgary and Ottawa had priority 3. All other stations were assigned priority 4. The total number of sequences in Q^* (i.e., that can allow aircraft swapping) for each flight schedule is also reported in Table 1. Recall that almost all sequences in Q^* contain two legs. Therefore, the proportion of legs that can be subject to re-fleeting varies between 14% and 20% approximately.

In its last four columns, Table 1 provides for each network the numbers of variables and constraints in model (1)–(9), augmented by (11)–(13), when 20 demand scenarios are considered. The first pair of columns (Var., Const.) indicates the model size before the CPLEX preprocessing, whereas the second pair gives it after preprocessing.

For each network 1 to 9, we created two new demand structures denoted structures II and III (the original demand corresponds to structure I). To do so, we perturbed the expected demand d_i of each itinerary $i \in I$ by multiplying it by a random number drawn from a uniform distribution in the interval $[0.55, 1.55]$. These new demands are then scaled to ensure that the total average demand over all itineraries remains the same. In 2002, most airlines were not profitable in the aftermath of the events of September 11, 2001. In particular, the seat occupancy rate of Air Canada was much below its normal rate. To perform our tests with normal occupancy rates, we multiplied the demands of each pair of network and demand structure by three different

Table 1: Flight network characteristics

Network	Days	$\|L\|$	$\|I\|$	$\|F\|$	$\sum_{f\in F} n_f$	$\|Q^*\|$	Before preproc.		After preproc.	
							Var.	Const.	Var.	Const.
1	1	678	7,590	15	194	68	27,359	26,514	14,477	7,023
2	1	756	8,871	15	194	56	28,729	26,656	14,179	6,775
3	1	759	8,733	15	194	67	30,199	28,945	15,699	7,662
4	2	1,391	15,474	15	194	106	53,117	48,720	26,136	12,651
5	2	1,508	17,625	15	194	132	58,215	53,954	29,462	14,346
6	2	1,518	17,943	15	194	119	57,673	52,706	28,556	13,778
7	3	2,151	24,459	15	194	186	82,120	75,613	40,904	20,011
8	3	2,257	26,505	15	194	175	85,077	77,331	41,859	20,324
9	7	5,180	71,844	15	205	394	194,588	176,162	95,026	46,361

values, namely, 1.2, 1.325, and 1.45 to yield average occupancy rates of 77.1 %, 81.2 % and 84.3 %, respectively (Air Transport Action Group (2014) reports that the average occupancy rate across the industry was 79% in 2014). We say that each multiplier generates a demand level and we denote these levels by L, M, and H for low, medium, and high, respectively. In summary, we used a total of 81 instances (9 networks $\times$ 3 demand structures $\times$ 3 demand levels) for our computational experiments.

Note that, even if the demand can vary significantly from one instance to another, we used the same leg sequences for all nine instances derived from a flight network. Given that most leg sequences are two-leg loops originating from one of the major stations serviced by Air Canada, we believe that, if the sequences were computed for each instance, they would not differ much, the reported results would be similar, and the conclusions would remain valid.

The demand scenarios used for our experiments should represent the variability of the demand for the airline, the market and the season considered. We assume that the total demand over the whole network does not vary significantly but the demand for an itinerary can be subject to a large variation. More specifically, we assume that, for a given scenario $w \in W$ or $w \in \hat{W}$, the total demand D^w follows a uniform distribution in the interval $[0.85D, 1.15D]$ where $D = \sum_{i\in I} d_i$ is the average total demand, and the demand d_i^w for an itinerary $i \in I$ is generated as a uniform random number $\mathcal{U}(0.5, 1.5)d_i D^w/D$. Each scenario w has the same occurrence probability $p^w = 1/|W|$ (or $1/|\hat{W}|$).

For the set W of demand scenarios used in our model, we ensure that the total demands of its scenarios are uniformly distributed in $[0.85D, 1.15D]$. Indeed, when using a relatively small number of scenarios (say, less than 30), the distribution may be far from being uniform. Thus, we enforce the distribution. For example, for a set of 5 scenarios, we generate scenarios with total demands $0.88D$, $0.94D$, $1.00D$, $1.06D$ and $1.12D$. For the larger-sized set $\hat{W}$ used to evaluate a solution, we generate randomly the total demand of each of its scenarios.

For a demand scenario set W (also for $\hat{W}$), we also ensure that, for each itinerary $i \in I$, the average of the generated demands d_i^w, $w \in W$, is equal to the average demand d_i, i.e., $\sum_{w\in W} d_i^w/|W| = d_i$. To achieve this while preserving as much as possible the total demand D^w for each scenario w, an iterative procedure that alternates between re-scaling the itinerary demands in each scenario to achieve the average demand for each itinerary over the scenarios and re-scaling for each scenario the itinerary demands to reach the targeted total demand is applied.

Detailed information about the generation of the instances and the demand scenarios can be found in Lasalle Ialongo (2014).

5.2 Test plan and environment

With our computational experiments, we pursue three goals. First, we want to assess the impact of the number of scenarios used in our model on the quality of the computed solutions. Second, we want to highlight the computational effort required to achieve these solutions. Third, we wish to compare the quality of the solutions obtained with our two-stage stochastic optimization model compared to that derived from the FAP with deterministic demand.

To perform these analyses, we solved each of the 81 instances 6 times: once as a FAP with deterministic demand, and five times with a different number of scenarios, namely, 1, 5, 10, 15, and 20 scenarios. The computed solutions throughout the solution process are then evaluated using 100 scenarios (not used in our model). Note that the FAP with stochastic demand and 1 scenario is very similar to the FAP with deterministic demand. The difference resides in the fact that two-leg sequences are considered in the former while only one-leg sequences are considered in the latter with very few compatible sequences. In consequence, because re-fleeting can only take place between compatible leg sequences when the final solution is evaluated at the end, only the solution of the former model can really benefit from re-fleeting. On the other hand, even if the single demand scenario corresponds to the average demand (and thus to the demand used for the FAP with deterministic demand), the fleet assignments may differ between the two models because the same aircraft type must be assigned to the legs of each leg sequence. This restriction is not present in the FAP with deterministic demand.

In our model, we used the constant re-fleeting penalty α, which was set to 200. Following preliminary tests, we decided to stop the overall algorithm after 5 iterations because no significant improvements were observed beyond this point. To avoid too long computational times, we allow in each iteration of the algorithm a maximum of 2 hours (resp. 6 hours) of computational time to solve our model and we apply a tolerance of 0.0002% (resp. 0.0005%) on the optimality gap for the instances derived from networks 1 to 8 (resp. network 9).

For our experiments, we used the IBM CPLEX MIP solver, version 12.4. All tests were performed using a single thread on an Intel Xeon X5670 processor clocked at 2.93GHz and with 24Gb of RAM.

5.3 Results

In this section, we report average results computed over all instances or over the instances with the same demand structure and level or derived from the same network. Detailed results (expected profits and computational times) can be found in the Appendix.

For the six models (FAP with deterministic demand, and FAP with 1, 5, 10, 15, and 20 demand scenarios), Figure 4 presents the evolution of the average expected profits of the best solutions found by the algorithm over the iterations. The averages are computed over all 81 instances. We omit to report the average expected profits of the solutions computed at the first iteration because they are much lower than the others and including them would compress the graphic. Note that the curves for the FAP with deterministic demand and the FAP with 1 demand scenario are almost superimposed.

From these average results, we observe that the average expected profits yielded by the FAP with deterministic demand and the FAP with a single demand scenario are much less than those yielded by the other models, independently of the iteration. Compared to the average expected profits for the FAP with deterministic demand, the average gain achieved after 5 iterations increases with the number of scenarios and varies between 3.32 and 3.44%. These gains slightly increase with the number of scenarios considered and clearly show that assigning the aircraft types to the flight legs while taking into account stochastic demand can yield a solution that facilitates profitable re-fleeting during the booking process. Note that the magnitude of these gains might depend on the distributions of the demand variables, the flight network topology and the flight schedule.

The average expected profits by demand structure and level obtained after 5 iterations of the algorithm are reported in Table 2. The last column of this table specifies the average gain in percentage obtained on the expected profits when using the FAP with 20 scenarios compared to the FAP with deterministic demand. Here, we observe that, for each instance, the expected profits does not increase monotonically with the number of scenarios considered. This is due to the heuristic nature of the solution process and the stochasticity of the scenario generation process. However, the induced noise is small compared to the general trend (see the Average row). As anticipated, as the demand level increases, the expected profits are larger.

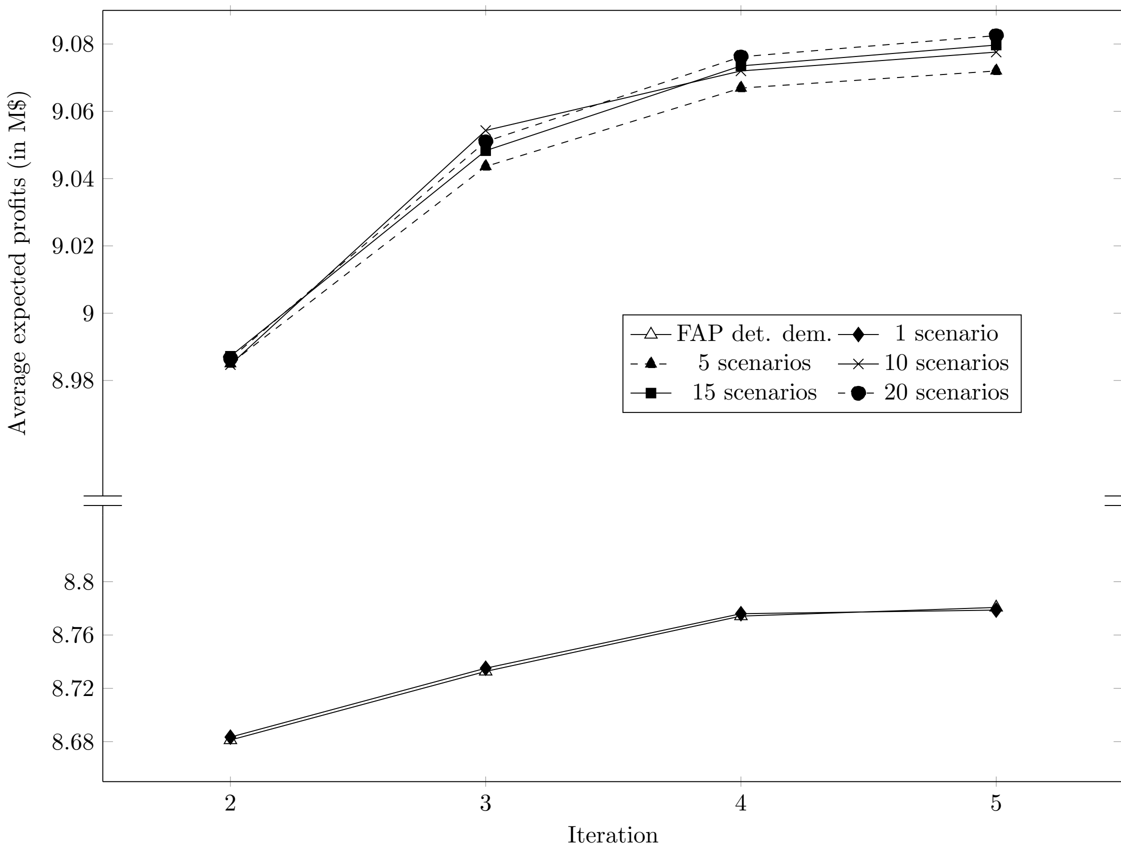

Figure 4: Progress of the average expected profits over all instances

Table 2: Average final expected profits (in M$) by demand structure and level

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.	Gain (%)
I	L	5.848	5.852	6.126	6.134	6.130	6.139	4.98
	M	8.506	8.511	8.858	8.864	8.858	8.861	4.17
	H	18.822	10.835	11.143	11.137	11.145	11.165	3.17
II	L	6.538	6.546	6.797	6.808	6.818	6.818	4.28
	M	9.171	9.159	9.439	9.442	9.449	9.443	2.96
	H	11.334	11.341	11.632	11.629	11.631	11.642	2.71
III	L	6.476	6.452	6.733	6.745	6.736	6.750	4.23
	M	9.075	9.064	9.364	9.381	9.372	9.368	3.23
	H	11.255	11.249	11.556	11.558	11.577	11.557	2.69
Average		**8.781**	**8.779**	**9.072**	**9.078**	**9.080**	**9.082**	**3.60**

However, note that, in this case, the gain in the expected profits decreases. This can be explained by the fact that the expected profits are much larger when demand increases but the swapping possibilities remain the same. An interesting average gain of 2.86% is nevertheless observed for the instances with the largest demands (demand level H).

Table 3 reports the average final expected profits, but this time by network. Recall that networks 1 to 3 span one day, networks 4 to 6 span two days, networks 7 and 8 span three days, and network 9 spans one week. Again, we observe that the average expected profits increase, in general, with the number of scenarios. Obviously, these profits are larger for larger networks. The last column indicates that the average gains

Table 3: Average final expected profits (in M$) by network

Instances from network	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.	Gain (%)
1	3.064	3.057	3.177	3.173	3.178	3.178	4.03
2	3.454	3.453	3.572	3.569	3.580	3.581	3.83
3	3.779	3.771	3.912	3.905	3.901	3.907	3.43
4	6.598	6.603	6.860	6.874	6.880	6.879	4.46
5	7.444	7.439	7.692	7.700	7.698	7.702	3.71
6	7.052	7.075	7.290	7.305	7.292	7.304	3.82
7	10.224	10.212	10.543	10.543	10.546	10.547	3.40
8	11.146	11.159	11.514	11.516	11.525	11.534	3.66
9	26.267	26.239	27.088	27.114	27.117	27.110	3.31

yielded by considering a FAP model with 20 demand scenarios instead of a FAP with deterministic demand tend to decrease with the network size. This can be explained by the proportion of flight legs that can be re-fleeted which decreases with the network size (from around 20% for the one-day instances to around 14% for the one-week instance).

Average total computational times (excluding the computation of the initial fleet assignment and the evaluation of the solution with 100 different scenarios) for each model are reported in Tables 4 and 5. In the first table, averages are given by demand structure and level. We observe that larger computational times are often required for the FAP with deterministic demand compared to the other models. This is somewhat surprising because there are more variables and constraints in the models with scenarios. However, the two-leg sequences considered in these models impose additional restrictions that reduce the combinatorial aspect of the problem and might help CPLEX to rapidly find good heuristic solutions. Note that this behavior is not observed for all instances (see the Appendix). On the other hand, the results clearly show that the total computational time increases, in general, with the number of scenarios considered. This is due to a corresponding increase in the number of variables and constraints. For the instances derived from the complete network, we also computed the percentage of time devoted to solving the passenger flow model (not shown in the tables). On average, this percentage is 1.1% for the FAP with deterministic demand and 24.7% for the FAP model with 20 demand scenarios. Therefore, as could be expected, this percentage seems proportional to the number of scenarios. Furthermore, from Table 4, we notice that the computational time is impacted by the level of the demand. In general, instances with larger demands require larger computational times because it becomes more difficult to assign the proper aircraft type to each flight leg (or leg sequence) when aircraft capacity is almost reached on many legs.

Table 4: Average computational times (in seconds) by demand structure and level

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.
I	L	4,564	3,138	2,226	2,236	3,994	5,506
	M	7,027	5,527	3,628	3,489	4,472	8,261
	H	9,350	9,856	8,788	8,066	9,667	9,644
II	L	8,332	5,392	3,709	3,911	5,856	7,426
	M	8,233	9,892	5,767	8,757	11,127	10,647
	H	10,250	11,077	9,088	12,325	11,590	14,423
III	L	7,990	8,805	5,334	5,460	6,941	8,299
	M	14,848	13,831	6,686	6,650	8,395	13,351
	H	16,470	14,501	11,248	10,456	14,542	15,599
Average		**9,674**	**9,113**	**6,275**	**6,816**	**8,509**	**10,351**

Table 5: Average computational times (in seconds) by network

Instances from network	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.
1	197	146	300	513	705	893
2	398	311	444	679	899	1,096
3	250	266	486	695	896	1,214
4	2,245	1,365	1,629	1,737	2,510	3,621
5	4,779	4,721	3,081	3,593	4,667	4,902
6	3,816	2,880	1,833	1,776	3,796	4,682
7	8,998	7,675	6,957	7,264	12,434	15,730
8	21,399	13,417	13,342	16,102	17,959	18,196
9	44,983	51,237	28,402	28,990	32,717	42,824

Table 5 presents average computational times by network. As expected, the average computational time increases with the size of the instances for each model. We highlight that an average computational time of almost 12 hours is necessary to solve the FAP model with 20 scenarios for the instances derived from the complete network 9 with more than 5,000 legs. These times are much less than those reported by Sherali and Zhu (2008) and Pilla et al. (2012) who considered smaller networks.

To complete our study, we computed the average number of flight legs re-fleeted in the solutions produced by the models with scenarios for the network-9 instances. The averages are computed over the 100 scenarios in $\hat{W}$. For models involving 1, 5, 10, 15, and 20 demand scenarios, there is an average of 3.44, 3.48, 4.04, 4.60, and 4.61% of the legs that are re-fleeted, respectively. This shows that considering more scenarios allows more profitable re-fleeting possibilities and, in consequence, additional expected profits.

6 Conclusion

In this paper, we addressed the FAP with stochastic demand using a two-stage stochastic optimization model with recourse. Once a planned fleet assignment is disclosed to the public, the possible recourse considered consists of changing the aircraft type on certain leg sequences that are identified a priori. This is possible only if there exist compatible leg sequences that can be involved in an aircraft swap that preserve aircraft flow balance as well as crew schedule feasibility when the crew schedule is already known. Our computational results show that the proposed solution approach can solve instances with more than 5,000 legs in acceptable computation times (less than 12 hours on average). Furthermore, it has the potential to yield a significant increase of the expected profits when our solutions are compared to those obtained by solving a FAP with deterministic demand.

For future work, we envision to define a similar stochastic model that would not depend on a priori identified leg sequences, but that would determine them as needed. The development of a solution approach capable of handling a much larger set of scenarios would also be an interesting research avenue.

A Detailed results

In this appendix, we provide the final expected profits and the total computational times for all instances grouped by network. There are two sets of tables. The first set (tables with an even number) presents the final expected profits whereas the second set (tables with an odd number) provides the total computational times.

Table 6: Final expected profits (in M$) for the nine network-1 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.	Gain (%)
I	L	1.944	1.952	2.052	2.032	2.042	2.043	5.08
	M	2.951	2.965	3.089	3.055	3.073	3.085	4.54
	H	3.792	3.796	3.943	3.926	3.953	3.936	3.79
II	L	2.110	2.134	2.253	2.265	2.260	2.267	7.45
	M	3.144	3.115	3.232	3.231	3.235	3.228	2.67
	H	3.995	3.979	4.088	4.097	4.103	4.097	2.57
III	L	2.258	2.239	2.363	2.367	2.352	2.366	4.79
	M	3.284	3.258	3.370	3.371	3.378	3.369	2.60
	H	4.096	4.079	4.199	4.216	4.205	4.209	2.76
Average		**3.064**	**3.057**	**3.177**	**3.173**	**3.178**	**3.178**	**4.03**

Table 7: Computational times (in seconds) for the nine network-1 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.
I	L	56	36	216	202	568	598
	M	137	359	249	426	657	606
	H	166	99	260	576	728	856
II	L	219	87	122	376	714	697
	M	154	321	288	563	618	865
	H	66	47	222	757	830	957
III	L	94	46	240	265	334	802
	M	218	62	372	650	575	1,088
	H	666	256	730	798	1,324	1,568
Average		**197**	**146**	**300**	**513**	**705**	**893**

Table 8: Final expected profits (in M$) for the nine network-2 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.	Gain (%)
I	L	2.217	2.194	2.308	2.328	2.342	2.339	5.51
	M	3.302	3.337	3.457	3.473	3.470	3.462	4.85
	H	4.289	4.310	4.429	4.420	4.410	4.440	3.54
II	L	2.584	2.599	2.690	2.679	2.691	2.684	3.87
	M	3.671	3.659	3.784	3.776	3.785	3.769	2.66
	H	4.560	4.551	4.660	4.685	4.679	4.695	2.96
III	L	2.461	2.436	2.553	2.554	2.578	2.566	4.26
	M	3.548	3.540	3.665	3.644	3.676	3.664	3.27
	H	4.455	4.451	4.606	4.566	4.591	4.613	3.55
Average		**3.454**	**3.453**	**3.572**	**3.569**	**3.580**	**3.581**	**3.83**

Table 9: Computational times (in seconds) for the nine network-2 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.
I	L	55	42	254	317	978	831
	M	200	298	328	754	827	1,676
	H	1,172	330	256	871	948	904
II	L	183	213	1,071	1,327	1,046	1,588
	M	161	63	330	616	805	1,286
	H	228	112	418	683	668	748
III	L	584	424	192	263	804	758
	M	578	144	388	636	914	1,104
	H	417	1,176	762	645	1,103	967
Average		**398**	**311**	**444**	**679**	**899**	**1,096**

Table 10: Final expected profits (in M$) for the nine network-3 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.	Gain (%)
I	L	2.509	2.500	2.592	2.580	2.578	2.574	2.60
	M	3.560	3.560	3.727	3.717	3.720	3.726	4.68
	H	4.531	4.529	4.677	4.680	4.645	4.674	3.15
II	L	3.057	3.058	3.190	3.162	3.164	3.164	3.50
	M	4.114	4.123	4.238	4.233	4.238	4.234	2.91
	H	4.944	4.951	5.101	5.095	5.085	5.103	3.21
III	L	2.757	2.728	2.870	2.869	2.865	2.877	4.33
	M	3.812	3.778	3.958	3.956	3.957	3.957	3.79
	H	4.726	4.709	4.854	4.852	4.855	4.854	2.71
Average		**3.779**	**3.771**	**3.912**	**3.905**	**3.901**	**3.907**	**3.43**

Table 11: Computational times (in seconds) for the nine network-3 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.
I	L	52	64	270	326	697	480
	M	264	99	507	390	944	1,024
	H	659	407	756	935	1,276	1,514
II	L	43	46	370	951	773	1,246
	M	59	58	201	406	519	1,189
	H	509	151	747	1,044	814	1,151
III	L	50	77	657	781	1,117	1,512
	M	409	1,010	396	661	907	1,164
	H	209	480	472	761	1,021	1,642
Average		**250**	**266**	**486**	**695**	**896**	**1,214**

Table 12: Final expected profits (in M$) for the nine network-4 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.	Gain (%)
I	L	4.341	4.382	4.555	4.595	4.564	4.607	6.14
	M	6.449	6.514	6.760	6.759	6.784	6.767	4.92
	H	8.297	8.348	8.573	8.627	8.635	8.653	4.29
II	L	4.654	4.625	4.919	4.929	4.947	4.918	5.68
	M	6.824	6.807	7.052	7.083	7.091	7.072	3.64
	H	8.598	8.608	8.869	8.874	8.858	8.897	3.47
III	L	4.793	4.769	5.007	4.993	5.015	5.016	4.65
	M	6.828	6.795	7.126	7.142	7.130	7.132	4.45
	H	8.601	8.581	8.876	8.864	8.899	8.850	2.89
Average		**6.598**	**6.603**	**6.860**	**6.874**	**6.880**	**6.879**	**4.46**

Table 13: Computational times (in seconds) for the nine network-4 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.
I	L	794	254	426	770	1,055	1,891
	M	2,017	302	500	887	1,097	1,964
	H	2,009	1,578	1,543	1,351	1,572	3,021
II	L	252	459	537	1,305	1,521	2,500
	M	382	800	1,168	3,713	2,218	5,142
	H	638	2,695	2,630	1,628	5,806	8,064
III	L	4,024	1,110	642	1,285	1,681	2,886
	M	5,950	2,471	1,404	3,462	2,003	2,163
	H	4,140	2,618	5,811	1,231	5,635	4,960
Average		**2,2245**	**1,365**	**1,629**	**1,737**	**2,510**	**3,621**

Table 14: Final expected profits (in M$) for the nine network-5 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.	Gain (%)
I	L	4.866	4.920	5.149	5.152	5.152	5.159	6.03
	M	7.120	7.089	7.446	7.434	7.419	7.449	4.62
	H	9.068	9.085	9.323	9.283	9.291	9.281	2.35
II	L	5.887	5.868	6.090	6.098	6.082	6.100	3.62
	M	7.988	7.991	8.232	8.247	8.244	8.251	3.29
	H	9.720	9.696	9.987	9.980	9.986	9.976	2.63
III	L	5.396	5.390	5.627	5.673	5.643	5.675	5.16
	M	7.577	7.563	7.787	7.808	7.821	7.798	2.92
	H	9.372	9.347	9.586	9.623	9.643	9.632	2.78
Average		**7.444**	**7.439**	**7.692**	**7.700**	**7.698**	**7.702**	**3.71**

Table 15: Computational times (in seconds) for the nine network-5 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.
I	L	3,257	1,050	601	1,223	1,689	2,228
	M	2,132	5,726	4,542	1,429	3,907	3,248
	H	5,458	2,640	5,474	7,056	9,451	4,019
II	L	1,310	400	2,478	2,714	6,640	4,380
	M	7,502	3,510	3,696	2,767	5,208	3,594
	H	3,194	8,012	963	3,157	2,700	5,025
III	L	1,293	3,220	2,197	5,323	4,618	6,530
	M	10,266	8,454	3,539	2,488	3,607	6,505
	H	8,595	9,476	4,241	6,182	4,183	8,585
Average		**4,779**	**4,721**	**3,081**	**3,593**	**4,667**	**4,902**

Table 16: Final expected profits (in M$) for the nine network-6 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.	Gain (%)
I	L	4.639	4.613	4.864	4.860	4.875	4.874	5.06
	M	6.946	7.006	7.164	7.217	7.201	7.188	3.49
	H	8.838	8.838	9.148	9.131	9.134	9.156	3.59
II	L	5.003	5.072	5.257	5.265	5.257	5.268	5.31
	M	7.256	7.327	7.529	7.526	7.529	7.544	3.98
	H	9.109	9.106	9.377	9.379	9.393	9.387	3.04
III	L	5.096	5.148	5.323	5.329	5.340	5.368	5.33
	M	7.367	7.377	7.536	7.606	7.499	7.564	2.67
	H	9.212	9.192	9.413	9.430	9.401	9.391	1.94
Average		**7.052**	**7.075**	**7.290**	**7.305**	**7.292**	**7.304**	**3.82**

Table 17: Computational times (in seconds) for the nine network-6 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.
I	L	2,437	2,613	991	1,882	5,379	4,395
	M	3,044	2,601	1,745	1,107	2,671	3,038
	H	4,896	2,991	1,906	3,957	7,493	5,341
II	L	3,922	2,846	648	993	1,332	2,395
	M	4,464	2,845	1,235	1,508	2,419	2,772
	H	2,849	3,852	1,871	1,364	1,810	4,672
III	L	1,236	899	3,472	1,562	4,641	3,187
	M	5,642	703	3,648	1,796	2,538	9,286
	H	5,850	6,571	985	1,814	5,882	7,054
Average		**3,816**	**2,880**	**1,833**	**1,776**	**3,796**	**4,682**

Table 18: Final expected profits (in M$) for the nine network-7 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.	Gain (%)
I	L	6.679	6.770	7.121	7.127	7.108	7.111	6.47
	M	10.053	10.055	10.472	10.481	10.445	10.444	3.89
	H	12.880	12.941	13.303	13.254	13.272	13.308	3.33
II	L	7.073	7.122	7.338	7.326	7.427	7.406	4.70
	M	10.409	10.400	10.658	10.664	10.677	10.668	2.49
	H	13.168	13.183	13.487	13.502	13.473	13.477	2.35
III	L	7.546	7.417	7.743	7.791	7.777	7.771	2.98
	M	10.792	10.650	11.024	11.032	11.032	11.063	2.51
	H	13.416	13.368	13.741	13.710	13.706	13.671	1.91
Average		**10.224**	**10.212**	**10.543**	**10.543**	**10.546**	**10.547**	**3.40**

Table 19: Computational times (in seconds) for the nine network-7 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.
I	L	8,047	14,779	8,146	3,144	7,089	12,800
	M	2,027	1,656	3,658	6,173	7,199	16,616
	H	9,077	8,420	10,955	4,087	6,227	10,212
II	L	14,267	18,648	11,698	9,078	14,335	23,111
	M	7,031	4,629	3,476	5,113	13,922	4,529
	H	10,791	4,479	5,751	8,747	7,929	11,991
III	L	8,324	3,652	9,384	11,477	29,806	29,171
	M	5,256	5,829	5,702	13,414	14,965	24,579
	H	16,161	6,979	3,839	4,144	10,438	8,561
Average		**8,998**	**7,675**	**6,957**	**7,264**	**12,434**	**15,730**

Table 20: Final expected profits (in M$) for the nine network-8 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand 1 sc.	5 sc.	10 sc.	15 sc.	20 sc.	Gain (%)
I	L	7.365	7.354	7.715	7.741	7.734	7.762	5.40
	M	10.689	10.723	11.144	11.142	11.190	11.186	4.65
	H	13.615	13.658	13.976	13.990	14.029	14.026	3.02
II	L	8.622	8.632	8.896	8.940	8.916	8.936	3.64
	M	11.852	11.809	12.168	12.144	12.152	12.153	2.54
	H	14.428	14.453	14.786	14.795	14.779	14.803	2.60
III	L	8.197	8.199	8.588	8.559	8.560	8.566	4.51
	M	11.384	11.429	11.797	11.795	11.784	11.817	3.81
	H	14.159	14.179	14.555	14.540	14.579	14.554	2.79
Average		**11.146**	**11.159**	**11.514**	**11.516**	**11.525**	**11.534**	**3.66**

Table 21: Computational times (in seconds) for the nine network-8 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand				
			1 sc.	5 sc.	10 sc.	15 sc.	20 sc.
I	L	13,341	3,544	6,270	7,978	12,153	15,264
	M	24,523	3,370	10,151	8,908	8,975	13,444
	H	24,016	21,559	14,237	15,652	15,814	9,504
II	L	17,564	9,666	8,727	11,030	15,165	15,470
	M	8,737	12,722	22,454	29,707	28,280	30,321
	H	26,870	13,330	19,516	24,084	25,300	25,862
III	L	20,954	17,923	2,731	7,930	3,350	5,232
	M	27,499	17,512	10,778	14,206	20,790	16,453
	H	29,090	21,128	25,210	25,421	31,808	32,217
Average		**21,399**	**13,417**	**13,342**	**16,102**	**17,959**	**18,196**

Table 22: Final expected profits (in M$) for the nine network-9 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand					Gain (%)
			1 sc.	5 sc.	10 sc.	15 sc.	20 sc.	
I	L	18.076	17.983	18.775	18.795	18.779	18.785	3.92
	M	25.486	25.354	26.467	26.494	26.424	26.445	3.76
	H	32.088	32.010	32.912	32.923	32.939	33.007	2.86
II	L	19.852	19.807	20.543	20.605	20.616	20.616	3.85
	M	27.285	27.199	28.056	28.076	28.092	28.064	2.86
	H	33.487	33.538	34.337	34.257	34.321	34.339	2.54
III	L	19.781	19.742	20.524	20.575	20.497	20.544	3.86
	M	27.086	27.188	28.010	28.074	28.067	27.951	3.19
	H	33.259	33.331	34.172	34.224	34.318	34.243	2.96
Average		**26.267**	**26.239**	**27.088**	**27.114**	**27.117**	**27.110**	**3.31**

Table 23: Computational times (in seconds) for the nine network-9 instances

Demand structure	Demand level	FAP with det. dem.	FAP with stochastic demand				
			1 sc.	5 sc.	10 sc.	15 sc.	20 sc.
I	L	13,039	5,861	2,861	4,283	6,339	11,065
	M	28,899	35,330	10,971	11,324	13,967	32,737
	H	36,697	50,680	43,706	38,105	43,498	51,425
II	L	37,231	16,159	7,730	7,424	11,175	15,449
	M	45,611	64,076	19,051	34,417	46,156	46,130
	H	47,109	67,016	49,678	69,461	58,450	71,334
III	L	35,350	51,894	28,495	20,255	16,117	24,616
	M	77,811	88,290	33,946	22,534	29,259	57,817
	H	83,102	81,829	59,179	53,105	69,488	74,840
Average		**44,983**	**51,237**	**28,402**	**28,990**	**32,717**	**42,824**

References

Air Transport Action Group. Aviation benefits beyond borders report 2014, April 2014.

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

C. Barnhart, A. Farahat, and M. Lohatepanont. Airline fleet assignment with enhanced revenue modeling. Operations Research, 57(1):231–244, 2009.

M. Berge and C. Hopperstad. Demand driven dispatch: A method for dynamic aircraft capacity assignment, models and algorithms. Operations Research, 41(1):153–168, 1993.

E.K. Bish, R. Suwandechochai, and D.R. Bish. Strategies for managing the flexible capacity in the airline industry. Naval Research Logistics, 51(5):654–685, 2004.

J. Dumas and F. Soumis. Passenger flow model for airline networks. Transportation Science, 42:197–207, 2008.

J. Dumas, F. Aithnard, and F. Soumis. Improving the objective function of the fleet assignment problem. Transportation Research Part B, 43(4):466–475, 2009.

C. Hane, C. Barnhart, E. Johnson, R. Marsten, G. Nemhauser, and G. Sigismondi. The fleet assignment problem: Solving a large-scale integer program. Mathematical Programming, 70(3):211–232, 1995.

A.I. Jarrah, J. Goodstein, and R. Narasimhan. An efficient airline re-fleeting model for the incremental modification of planned fleet assignments. Transportation Science, 34(4):349–363, 2000.

H. Jiang. Dynamic airline scheduling and robust airline schedule de-peaking. PhD thesis, Massachusetts Institute of Technology, Cambridge, USA, 2006.

H. Jiang and C. Barnhart. Dynamic airline scheduling. Transportation Science, 43(3):336–354, 2009.

H. Jiang and C. Barnhart. Robust airline schedule design in a dynamic scheduling environment. Computers & Operations Research, 40(3):831–840, 2013.

D. Klabjan. Large-scale models in the airline industry. In G. Desaulniers, J. Desrosiers, and M. Solomon, editors, Column Generation, pages 163–195. New York : Springer, 2005.

D. Lasalle Ialongo. Problème d'affectation des types d'avion aux vols: Optimisation robuste et intégration de la demande des passagers. PhD thesis, Polytechnique Montréal, Montréal, Canada, 2014.

V.L. Pilla, J.M. Rosenberger, V.C.P. Chen, and B. Smith. A statistical computer experiments approach to airline fleet assignment. IIE Transactions, 40(5):524–537, 2008.

V.L. Pilla, J.M. Rosenberger, V. Chen, N. Engsuwan, and S. Siddappa. A multivariate adaptive regression splines cutting plane approach for solving a two-stage stochastic programming fleet assignment model. European Journal of Operational Research, 216(1):162 – 171, 2012.

H.D. Sherali and X. Zhu. Two-stage fleet assignment model considering stochastic passenger demands. Operations Research, 56(2):383–399, 2008.

H.D. Sherali, E.K. Bish, and X. Zhu. Polyhedral analysis and algorithms for a demand-driven refleeting model for aircraft assignment. Transportation Science, 39(3):349–366, 2005.

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

K.T. Talluri. Swapping applications in a daily airline fleet assignment. Transportation Science, 30(3):237–248, 1996.

V. Warburg, T.G. Hansen, A. Larsen, H. Norman, and E. Andersson. Dynamic airline scheduling: An analysis of the potentials of refleeting and retiming. Journal of Air Transport Management, 14(4):163–167, 2008.