CIRRELT

Centre interuniversitaire de recherche
sur les réseaux d'entreprise, la logistique et le transport

Interuniversity Research Centre
on Enterprise Networks, Logistics and Transportation

Extended Aircraft Arrival Management under Uncertainty: A Chance-Constrained Two-Stage Stochastic Mixed-Integer Programming Model

Ahmed Khassiba
Fabian Bastin
Sonia Cafieri
Bernard Gendron
Marcel Mongeau

December 2018

CIRRELT-2018-55

Extended Aircraft Arrival Management under Uncertainty: A Chance-Constrained Two-Stage Stochastic Mixed-Integer Programming Model

Ahmed Khassiba[1,2,3,*], Fabian Bastin[1,2], Sonia Cafieri[3], Bernard Gendron[1,2], Marcel Mongeau[3]

[1] Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT)

[2] Department of Computer Science and Operations Research, Université de Montréal, P.O. Box 6128, Station Centre-Ville, Montréal, Canada H3C 3J7

[3] École Nationale de l'Aviation Civile (ENAC), 7, avenue Edouard Belin, CS 54005, 31055 Toulouse Cedex 4, France

Abstract. The extended aircraft arrival management problem, as an extension of the classic Aircraft Landing Problem, seeks to pre-schedule aircraft on a destination airport a few hours before their planned landing times. A two-stage stochastic mixed-integer programming model enriched by probability constraints is proposed. The first-stage optimization problem determines an aircraft sequence and target times over the entry point to the terminal area, called initial approach fix (IAF), so as to minimize the landing sequence length. Actual times over the IAF are assumed to deviate randomly from target times following known probability distributions. In the second stage, actual times over the IAF are assumed to be revealed, and landing times are to be determined in view of minimizing a time-deviation impact cost function such as air traffic control workload in the terminal area. Three Benders reformulations are proposed: a simple-cut, a partially aggregated-cut, and a multi-cut versions. This study considers a single IAF and a single landing runway. Results on realistic instances involving 10 aircraft show an improvement of objective-function values by 10% when compared to a wait-and-see policy. The multi-cut version of Branch-and-Benders-Cut achieves shorter computing times, up to a factor of 6 compared to CPLEX, in several difficult test cases.

Keywords. Aircraft arrival management, two-stage mixed-integer stochastic programming, Benders decomposition.

Acknowledgements The authors would like to thank Serge Roux (ENAC) for providing the arrivals data on CDG airport. F. Bastin and B. Gendron are funded by the Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grants.

---

* Corresponding author: Ahmed.Khassiba@cirrelt.ca

Dépôt légal – Bibliothèque et Archives nationales du Québec
Bibliothèque et Archives Canada, 2018

1. Introduction

Predicted growth in air traffic, capacity limitations of the overall air transportation system, environmental and human-factor challenges have been the main motivations for air transportation experts to formulate and tackle problems arising in Air Traffic Management (ATM). At the airport level, landings are considered to be among the most critical, bottleneck operations, where safety and efficiency are of great importance. Accordingly, the Aircraft Landing Problem (ALP) was introduced more than four decades ago (see Dear 1976 and, later, Bennell, Mesgarpour, and Potts 2011). The ALP deals with *sequencing* and *scheduling* aircraft landings optimally on the available runways at a given airport. Sequencing consists in finding an order among the considered aircraft, while scheduling is related to the timing of aircraft landings. Optimality criteria usually include maximizing airport throughput or minimizing aircraft delay, while satisfying operational and safety constraints, mainly separation constraints between operating aircraft near the runway threshold, called *final-approach separations*. Final-approach separations are based on aircraft wake-turbulence categories, presented in Table 1, and are expressed as inter-aircraft distances in nautical miles (NM; 1 NM = 1.852 m) as in Table 2. The difficulty of the ALP is due to the non-symmetry of the final approach separations, unlike in other flight phases. For example, in the near-to-airport airspace, called the *terminal area*, before the final approach phase, aircraft are horizontally separated by 5 NM.

Table 1 Wake-turbulence categories according to the International Civil Aviation Organization (ICAO).

Wake-turbulence category	Max certificated take-off mass (kg)	Aircraft-type examples
Heavy (H)	above 136,000	A350, A340, B747, B777
Medium (M)	between 7,000 and 136,000	A320, B737
Light (L)	below 7,000	General aviation and executive jets

source: `https://www.skybrary.aero/index.php/ICAO_Wake_Turbulence_Category`

Table 2 Minimal final-approach separations (NM) according to ICAO's wake-turbulence categories.

		Following aircraft		
		H	M	L
Leading aircraft	H	4	5	6
	M	2.5	2.5	4
	L	2.5	2.5	2.5

source: de Neufville et al. 2013

On the operational side, since the early 90's in the USA and Europe, air traffic controllers (ATCs), responsible for air traffic flows' safety and efficiency around major airports, have been using decision-support tools that attempt at sequencing and scheduling landings optimally at available runways according to ATCs' input criteria (Garcia 1990, Neuman and Erzberger 1990, Völckers 1990, Hasevoets and Conroy 2010). Nowadays, the main such tool in the USA is known as the Traffic Management Advisor, while in Europe it is called Arrival Manager (AMAN). Without loss of generality, we will retain the European naming in the sequel. AMAN typically captures inbound aircraft at distances under 200 NM from their destination airport *i.e.*, around 40 minutes before landing (Division-NAAS 2016, Tielrooij et al. 2015). Then, using predicted landing times and aircraft characteristics such as wake-turbulence categories, AMAN determines an "optimal" landing sequence and target landing times according to the ATCs' input criteria. Afterwards, the controllers have to communicate control actions to pilots in order to enforce this optimal sequence, and to satisfy at best the target landing times. Apart from recoursing to *holding stacks*, where aircraft keep flying in a circle at low altitudes close to the *terminal area* (formally called *Terminal Control Area* (TCA) in the USA, and *Terminal Maneuvering Area* (TMA) in Europe), controllers are allowed to change the aircraft speeds and trajectories in order to avoid terminal area congestion. The latter two control actions are more likely to achieve the so-called *linear holding*, which is preferred to holding stacks in terms of safety, ATC workload and eco-efficiency. However, recoursing to linear holding is most effective when flights are still relatively far from the destination airport, *e.g.* while still in their cruise phase.

These facts motivate the extension of AMAN's horizon in order to diminish the need for holding stacks and to rely more on linear holding techniques. Accordingly, important ATM research and development programs, NextGen in the USA and SESAR in Europe, foresee their decision support tools' operational horizons to be extended up to 500 NM, *i.e.,* about 2 hours before landing (Tielrooij et al. 2015). The new European decision-support tool is called *Extended-AMAN* (E-AMAN). However, with extended horizons come greater uncertainties on the predicted times, such as those used by AMAN, when optimizing the landing sequence (Meyn and Erzberger 2005, Tielrooij et al. 2015). A recent attempt to quantify the uncertainty on predicted landing times at a horizon of three hours, using actual flight data, is presented in Tielrooij et al. 2015. To deal with predicted-time errors, current AMANs rely on regularly re-optimizing the schedule (for example every time aircraft data are updated). Although it may appear satisfying in practice, re-optimizing addresses uncertainty by brute force instead of embedding it within the optimization problem. One of the obvious drawbacks of frequent re-optimization is the instability of the optimal sequence it produces. With an extended operational horizon, addressing uncertainty through frequent re-optimizations will, very likely, result in highly-instable sequences, increasing the workload of ATCs who cannot easily build and maintain the continuously-changing sequence of aircraft.

To the best of our knowledge, the ALP has been most-commonly studied when considering the *deterministic* case (Dear 1976, Beasley et al. 2000, Balakrishnan and Chandran 2010, Bennell, Mesgarpour, and Potts 2011, Furini et al. 2015), while uncertainty has less often been taken into account. Pioneer studies of the ALP considering uncertainty were conducted by Meyn and Erzberger (2005), Chandran and Balakrishnan (2007), and Lee (2008) who basically added probabilistic considerations to the deterministic ALP. Stochastic optimization models, including two-stage and multi-stage models, were applied by Sölveling et al. (2011), Sölveling (2012), Sölveling and Clarke (2014), and Bosson and Sun (2016) to address a variant of the ALP under uncertainty that considers departures and surface operations on the airport. Recently, Heidt et al. (2016), Kapolke et al. (2016), and Heidt (2017) proposed various robust optimization models to address the *runway scheduling problem* under uncertainty. The aforementioned studies focus on the ALP under uncertainty with an operational horizon under one hour, whereas Kapolke et al. (2016) investigates the *pre-tactical ALP* which starts several hours before the planned landing times. In Kapolke et al. 2016, a simplified one-stage stochastic optimization model is compared with several robust optimization models. Their study tends to show that robust optimization is more promising than stochastic optimization for solving the pre-tactical ALP. Remark that the proposed one-stage stochastic optimization model only addresses the "expected-scenario" problem, *i.e.,* the variant in which uncertain data are replaced by their expected values. However, as stated by Birge and Louveaux (2011): "planning for the expected case is in fact "forgetting" uncertainty". We believe there is room for considering more practical aspects and algorithmic enhancements in order to get the most from stochastic optimization applied to the ALP under uncertainty.

In this paper, we consider an *extended Aircraft Landing Problem*, the ALP variant in which the operational horizon is extended, as for E-AMAN. Moreover, we aim at embedding the uncertainty *within* the optimization model. In view of simplifying the presentation, the scope of this preliminary study is limited to the case involving a unique entry point to the terminal area, called the *initial approach fix* (IAF), and a unique landing runway. We propose a two-stage stochastic optimization model with recourse, that is enhanced by probability constraints in the first stage, to mitigate the risk of separation violations over the IAF. Our two-stage stochastic model seeks to find a schedule that minimizes both the runway sequence length and the expected time-deviation impact costs (one may think of the expected ATC workload in the terminal area, as a possible instance). In a first stage, aircraft are sequenced and scheduled at the terminal area entry point so as to minimize the runway sequence length. In this stage, while *IAF target times* at the entry point of the terminal area are decision variables, *IAF actual times* are considered to be random variables. In a hypothetical second stage, uncertainty is assumed to be revealed and aircraft are scheduled to the runway threshold so as to minimize the delay impact costs. The first-stage problem will be shown to boil down to the classical Asymmetric Traveling Salesman Problem with Time Windows (ATSP-TW), an NP-hard problem which can be modeled as a mixed integer linear program (MILP). Assuming a piecewise linear time-deviation impact cost function, the second-stage problem reduces to a simple linear program (LP). Three Benders reformulations are proposed: a simple-cut, a partially-aggregated-cut, and a multi-cut versions.

The paper is organized as follows. The problem statement along with the operational context are introduced in Section 2. In Section 3, we propose a two-stage stochastic model with recourse. Solution methods are proposed in Section 4. Results of numerical experiments are discussed in Section 5. Section 6 presents some conclusions and future research tracks.

2. Problem statement

We consider a set of aircraft planning to land at a given destination airport in two to three hour look-ahead time. For the sake of simplification, in this preliminary study we make the following two operational assumptions. Firstly, all considered aircraft pass over the same IAF to enter the airport's terminal area. Secondly, all aircraft land on the same runway of the considered airport. Hence, this study concentrates on the case where a *single IAF* and a *single runway* are considered. Paris Charles-de-Gaulle airport (CDG) is an illustration of this simplified setting when arrival flows from North and South are disaggregated, the subset of aircraft coming from a same corner (north-west, for example) passes over a same IAF and lands on a same runway.

Our problem involves two types of separations: final-approach separations and separation over the IAF. For modeling and optimization purposes, separations expressed in terms of nautical miles are converted to seconds. Remark that this practice corresponds to common practice; Table 3 shows final-approach separations converted to seconds, as used in CDG. Detail of such a conversion may be found in de Neufville et al. 2013. For the sake of exposition simplification, we assume that all aircraft ground speeds over the IAF are equal to 250 knots (1 knots = 1 NM per hour), which is typically the maximal allowed on-board indicated air speed over the IAF. Hence, the usual 5 NM minimal separation over the IAF may be converted into 72 seconds.

Table 3 Final-approach separations (seconds) at CDG according to ICAO's wake-turbulence categories

		Following aircraft		
		H	M	L
	H	96	157	207
Leading aircraft	M	60	69	123
	L	60	69	82

Given a set of aircraft, we seek to find a *target aircraft sequence over the IAF*, and a *target time over the IAF* for each aircraft. We assume that the target sequence over the IAF is the same as the target landing sequence. In the sequel, the *target sequence* will equivalently designate any of these two target sequences. We aim at finding a target sequence so as to maximize the runway throughput. Target times over the IAF have to satisfy the separation requirements over the IAF. *Actual times over the IAF* correspond to the times at which aircraft effectively pass over the IAF. The order in which aircraft effectively pass over the IAF is called the *actual sequence over the IAF*. We assume that actual times over the IAF randomly deviate from the target times following known distributions. These deviations are unknown when the target sequence and the target times over the IAF are decided. Because of these deviations, actual times may violate the separation constraints over the IAF, even though aircraft were safely separated in terms of target times. Also, the actual sequence over the IAF may differ from the target sequence. In practice, ATCs have to make control decisions to prevent such violations over the IAF, and to build the target sequence for landing. To limit subsequent delay impact costs (such as ATC workload), we consider probability constraints to express the acceptable rate of separation violations (in terms of actual times) over the IAF (for instance, one may expect these probability constraints to prevent excessive subsequent re-sequencing). Furthermore, target times over the IAF have to respect predefined time-window constraints. These constraints, when correctly defined, will prevent aircraft from being either excessively delayed or excessively expedited, with respect to their planned times. As suggested in Bennell, Mesgarpour, and Potts (2011), this may also help fulfill fairness requirements among aircraft, similarly to the more classical *constraint position shifting (CPS)* approach (see Balakrishnan and Chandran 2010 for a comprehensive study in the deterministic case), where each aircraft position cannot be shifted by more than a predefined number of positions in the first-come first-serve sequence. Because of the uncertainty on actual times, decisions over the IAF (target sequence and target times) may result in different air traffic situations over the IAF (actual sequence and actual times) that are likely to deteriorate runway throughput and to incur further delay costs.

We define the hypothetical *second stage* in view of anticipating (ideally all) the different outcomes of the decisions over the IAF, subsequently called the *first-stage* decisions. In this second stage, deviations from target times over the IAF are assumed to be revealed. The second-stage problem consists in finding a target landing time for each aircraft in order to minimize delay costs, while satisfying realistic flight times through the terminal area and, more importantly, without violating final-approach separations. These re-scheduling decisions represent the ATCs recourse to handle the air traffic situation from the IAF to the runway threshold once uncertainties are revealed. Therefore, we seek to minimize the expected cost of this recourse through considering different (eventually all) *scenarios*, *i.e.*, realizations of the uncertainties.

3. A two-stage stochastic optimization model with recourse

Let $\mathcal{A} = \{1, 2, \ldots, n\}$ be the set of aircraft indices to be sequenced and scheduled over the IAF. The minimal time separation required over the IAF is given and noted $\underline{S}^I$. The minimal time separation during the final approach between a leading aircraft $i \in \mathcal{A}$ and a following aircraft $j \in \mathcal{A}$ is also given; it is noted S_{ij}. In the first stage, the n aircraft need to be sequenced and scheduled over the IAF. Let δ_{ij} be the binary decision variable that takes the value 1 if and only if aircraft $i \in \mathcal{A}$ **directly precedes** aircraft $j \in \mathcal{A}$ in the sequence, and 0 otherwise. These variables are called the *sequencing variables*. We seek to find the aircraft sequence that maximizes the runway throughput taking into account final-approach separations. Such a sequence can be obtained by solving an (open) Asymmetric Traveling Salesman Problem (ATSP) instance where the city set corresponds to the aircraft set $\mathcal{A}$. Equivalently, we can consider a classical ATSP instance involving the set $\mathcal{A}^+ = \{1, 2, \ldots, n+1\}$, where index $n+1$ corresponds to a fictitious extra aircraft to close the Hamiltonian circuit. Then, $2n$ more sequencing binary decision variables, $\delta_{i,n+1}, \delta_{n+1,i}, i \in \mathcal{A}$, are introduced to take into account the $(n+1)^{st}$ aircraft. This spurious aircraft has null minimal time separation with the n original aircraft.

To summarize, the (first-stage) sequencing variables are:

$$\delta_{ij} = \begin{cases} 1 & \text{if aircraft } i \textbf{ directly precedes } \text{aircraft } j \\ 0 & \text{otherwise} \end{cases} \qquad (i,j) \in \mathcal{A}^+ \times \mathcal{A}^+, \quad i \neq j.$$

At a look-ahead time of two to three hours before landing, every aircraft $i \in \mathcal{A}$ has a fixed (given) time window $[E_i^I, L_i^I]$ to pass over the IAF, where E_i^I and L_i^I are respectively the given earliest and latest times. Let x_i be a first-stage decision variable representing the target time over the IAF of aircraft $i \in \mathcal{A}$; it must satisfy the bound constraints:

$$x_i \in \left[E_i^I, L_i^I\right], \qquad i \in \mathcal{A}.$$

Let $\boldsymbol{\omega_i}$ be the random variable representing the deviation of the actual time over the IAF of aircraft $i \in \mathcal{A}$ with respect to its target time x_i. Let ω_i be a realization of the random variable $\boldsymbol{\omega_i}$. Then, the actual time over the IAF of aircraft $i \in \mathcal{A}$ is simply $x_i + \omega_i$. Let $\alpha \in [0,\, 1]$ be the lowest acceptable probability that separation over the IAF is satisfied between the pair of aircraft $(i,j) \in \mathcal{A} \times \mathcal{A}$, $i \neq j$, once uncertainties are revealed.

In the hypothetical second stage, actual times over the IAF are assumed to be known with certainty. Recall that the actual sequence over the IAF might not correspond to the target sequence. As mentioned in Section 2, we choose to enforce the target landing sequence to be the same as the target sequence over the IAF, since the latter was computed so as to minimize the runway sequence length. Hence, no (re-)sequencing variables are needed in the second stage. However, the n aircraft need to be scheduled at the runway threshold. Let y_i be the decision variable representing the target landing time of aircraft $i \in \mathcal{A}$. These variables are called *second-stage scheduling variables* and have to satisfy the time separation constraints during the final approach. In order to keep these target times realistic, we introduce a landing time window $[E_i,\, L_i]$ for every aircraft $i \in \mathcal{A}$ so that second-stage variables must satisfy the bound constraints:

$$y_i \in [E_i, L_i]\,, \qquad i \in \mathcal{A}.$$

For an aircraft i, recalling that the actual IAF time is $x_i + \omega_i$, the earliest and the latest landing times can be expressed using (given) minimal and maximal flight times from the IAF to the runway threshold, $\underline{V_i}$ and $\overline{V_i}$ respectively, where $0 < \underline{V_i} \le \overline{V_i}$, as follows: $E_i = (x_i + \omega_i) + \underline{V_i}$ and $L_i = (x_i + \omega_i) + \overline{V_i}$.

Following Bennell, Mesgarpour, and Potts (2017), we define the *unconstrained* (or *uncongested*) *landing time* of aircraft $i \in \mathcal{A}$, noted U_i, to be the landing time of aircraft i as if it were alone in the terminal area, and the unconstrained flight time $\hat{V}_i$ such that $\underline{V_i} \le \hat{V}_i \le \overline{V_i}$ and $U_i = (x_i + \omega_i) + \hat{V}_i$. Let f be a function that estimates time-deviation costs incurred (*e.g.*, ATC workload in the terminal area). In this study, we propose to model such costs with a convex piecewise linear function.

We introduce the following vector notation: $x = (x_1, x_2, \ldots, x_n)^T$. Vectors $\boldsymbol{\omega}$, ω, and y are defined likewise. We also introduce the matrix notation: $\delta = (\delta_{ij})_{\substack{(i,j) \in \mathcal{A}^+ \times \mathcal{A}^+ \\ i \neq j}}$. Given the expectation operator $\mathbb{E}_{\boldsymbol{\omega}}[.]$ over the random vector $\boldsymbol{\omega}$, we propose the following two-stage stochastic optimization model with recourse, also called the *true model*:

$$\begin{aligned}
\min_{\delta,\, x} \quad & \sum_{\substack{(i,j) \in \mathcal{A}^+ \times \mathcal{A}^+ \\ i \neq j}} \delta_{ij} S_{ij} + \mathbb{E}_{\boldsymbol{\omega}}[Q(\delta, x, \omega)] && && (1)\\
\text{s.t.} \quad & \sum_{\substack{j \in \mathcal{A}^+ \\ j \neq i}} \delta_{ji} = 1 && i \in \mathcal{A}^+ && (2)\\
& \sum_{\substack{j \in \mathcal{A}^+ \\ j \neq i}} \delta_{ij} = 1 && i \in \mathcal{A}^+ && (3)\\
& x_j \ge x_i + \underline{S}^I - M^I_{ij}(1 - \delta_{ij}) && (i,j) \in \mathcal{A} \times \mathcal{A}, \quad i \neq j && (4)\\
& \mathbb{P}(x_j + \omega_j \ge x_i + \omega_i + \underline{S}^I - M^{I\alpha}_{ij}(1 - \delta_{ij})) \ge \alpha && (i,j) \in \mathcal{A} \times \mathcal{A}, \quad i \neq j && (5)\\
& E^I_i \le x_i \le L^I_i && i \in \mathcal{A} && (6)\\
& \delta_{ij} \in \{0, 1\} && (i,j) \in \mathcal{A}^+ \times \mathcal{A}^+, \quad i \neq j && (7)
\end{aligned}$$

where:

$$\begin{aligned}
Q(\delta, x, \omega) = \quad \min_{y} \quad & f\,(x\,, \omega\,, y) && && (8)\\
\text{s.t.} \quad & y_j \ge y_i + S_{ij} - M_{ij}(1 - \delta_{ij}) && (i,j) \in \mathcal{A} \times \mathcal{A}, \quad i \neq j && (9)\\
& \underline{V_i} \le y_i - (x_i + \omega_i) \le \overline{V_i} && i \in \mathcal{A} && (10)
\end{aligned}$$

The objective function (1) is the sum of: the first-stage objective function, $\sum_{\substack{(i,j)\in\mathcal{A}^+\times\mathcal{A}^+\\ i\neq j}} \delta_{ij}S_{ij}$, and the expected cost of the second stage, $\mathbb{E}_{\boldsymbol{\omega}}[Q(\delta,x,\omega)]$. The first-stage problem minimizes the length of the sequence in terms of final-approach separations, subject to constraints (2) to (7). Given a scenario ω, the cost of the second-stage (so-called *recourse*) problem, $Q(\delta,x,\omega)$, is defined by (8) to (10). Big-M constants appearing in constraints (4), (5) and (9) will be further commented below.

First-stage model: Constraints (2), (3) and (4) are directly inspired from the classical ATSP formulation. Constraints (2) and (3) ensure that all aircraft in $\mathcal{A}^+$ are sequenced, which corresponds to visiting all cities in an ATSP. Constraints (4) express the minimal time separation requirement over the IAF between any two successive aircraft, where the big-M type constants M^I_{ij} are large enough so that the corresponding constraint is necessarily satisfied as soon as $\delta_{ij}=0$. Constraints (5) are probability constraints that ensure separation based on actual times over the IAF between any two different aircraft with a probability higher than some given threshold value α. Under the assumption of independent and identically distributed (i.i.d.) random variables $\boldsymbol{\omega_i}$ for all $i\in\mathcal{A}$, probability constraints (5) can be expressed in a manner analogous to the big-M separation constraints (4). This will be detailed in Subsection 3.2. Constraints (6) are time-window constraints on target times over the IAF. Constraints (7) stipulate the binary nature of the δ_{ij} variables.

Without the probability constraints (5), the first-stage problem defined by the first-stage objective function and constraints (2) to (4), (6) and (7), reduces to an instance of the ATSP with time windows (ATSP-TW). The reduction goes as follows: cities correspond to aircraft, the traveling salesperson corresponds to the IAF, costs of travel between cities is represented by final-approach separations (S_{ij}), and times of travel between cities correspond to the IAF separation ($\underline{S}^I$). Remark, however, that the scheduling part of the problem is a special simple cas, since the IAF separation is not aircraft-dependent, unlike typical ATSP-TW travel time between cities. Finally, subtour elimination constraints are not required since the IAF separation constraints (4) play the role of MTZ constraints (Miller, Tucker, and Zemlin 1960). As mentioned above, big-M constants must be large enough for the formulation to be correct. However, very large big-M values are known to lead to numerical instabilities during resolution. The best expression (smallest while sufficiently large) for the big-M constants M^I_{ij} in (4) can easily be shown to be: $M^I_{ij}=L^I_i-E^I_j+\underline{S}^I$.

Second-stage model: With regard to the second-stage model, the objective function (8) minimizes a cost function f that represents the impact of time-deviation with respect to unconstrained landing times. This time-deviation impact can be interpreted as the additional approach-controller's workload to handle the inbound traffic. A candidate expression of f is proposed in Subsection 3.1. Constraints (9) ensure final approach minimal time separation. Minimal and maximal flight times are enforced by constraints (10). Hence, the second-stage problem consists in finding a landing schedule for n aircraft that minimizes the cost function f, given a target sequence and landing time windows. Big-M constants M_{ij} in (9) can be computed as the lowest upper bound to $(y_i - y_j + S_{ij})$. Using constraints (9) and (10) and bound constraints (6) on x_i, the best expression for M_{ij} can be shown to be: $M_{ij} = \left(L_i^I + \omega_i + \overline{V_i}\right) - \left(E_j^I + \omega_j + \underline{V_j}\right) + S_{ij}$.

3.1. Second-stage objective function: minimizing total time-deviation impact cost

A problem-specific second-stage objective is to minimize the total impact cost of time-deviations with respect to unconstrained landing times. To give an example, this impact cost can be interpreted as a simplified estimation of the approach-controller's additional workload to handle inbound traffic. Briefly explained, a one-minute advance of an aircraft landing time can be costless for an approach-controller, since he only has to give "a bit earlier" one instruction to the pilot, that is to follow the standard approach procedure. However, for a delay of one to four minutes, the approach controller has to communicate a sequence of instructions to modify the trajectory and/or the speed of the given aircraft. For delays larger than four minutes, the approach controller has to keep the aircraft in a holding stack, a predefined circular circuit in a confined space, often seen as an "airborne waiting room". Holding patterns are known to generate much more workload for controllers and for pilots than trajectory and speed changes.

More generally, we assume that a deviation, within predefined bounds, of any aircraft target time (y_i) with respect to its unconstrained landing time (U_i) has an impact cost proportional to the deviation amplitude within these bounds. A possible form of total time-deviation impact cost function f is an additive form $f(\cdot) = \sum_{i \in \mathcal{A}} f_i(\cdot)$, where each f_i is a convex piecewise linear function of y_i that estimates the time-deviation impact cost of aircraft $i \in \mathcal{A}$. For instance, given the slopes $c_1, c_2, c_3 \in \mathbb{R}_+$ such that $c_2 \leq c_3$ and some intermediate landing times L_i^{med}, such that $U_i \leq L_i^{\text{med}} \leq L_i$:

$$f_i(x_i, \omega_i, y_i) = \begin{cases} c_1.\,(U_i - y_i) & \text{if } \ E_i \leq y_i \leq U_i \\ 0 & \text{if } \ y_i = U_i \\ c_2.\,(y_i - U_i) & \text{if } \ U_i \leq y_i \leq L_i^{\text{med}} \\ c_2.\,(L_i^{\text{med}} - U_i) + c_3.\,(y_i - L_i^{\text{med}}) & \text{if } \ L_i^{\text{med}} \leq y_i \leq L_i \end{cases}$$

is an example of time-deviation impact cost function f_i for an aircraft $i \in \mathcal{A}$ (Figure 1). For an aircraft $i \in \mathcal{A}$, inline with the definitions of E_i, U_i and L_i given above, the intermediate landing time L_i^{med} can be defined using an intermediate flight time V_i^{med} such that $0 < \underline{V_i} \leq \hat{V}_i \leq V_i^{\text{med}} \leq \overline{V}_i$ and $L_i^{\text{med}} = (x_i + \omega_i) + V_i^{\text{med}}$.

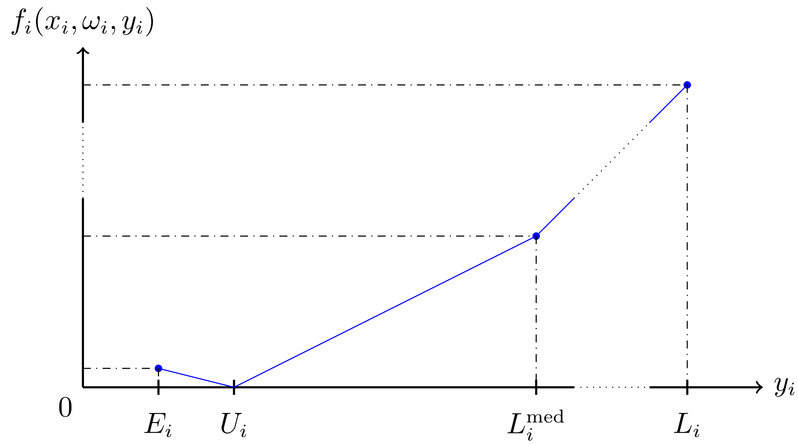

Figure 1 **Time-deviation impact cost function f_i of aircraft $i \in \mathcal{A}$.**

Given such a separable convex piecewise-linear form, the objective function (8) can easily be linearized using, for the example above, three auxiliary variables z_i^-, z_i^+ and z_i^{++} per aircraft $i \in \mathcal{A}$ as follows:

$$\min_{\substack{y, z^-, z^+ \\ z^{++}}} \sum_{i \in \mathcal{A}} \left(c_1 z_i^- + c_2 z_i^+ + c_3 z_i^{++} \right) \tag{11}$$

$$y_i - U_i = z_i^+ + z_i^{++} - z_i^- \qquad i \in \mathcal{A} \tag{12}$$

$$z_i^+ \leq L_i^{\text{med}} \qquad i \in \mathcal{A} \tag{13}$$

$$z_i^-, z_i^+, z_i^{++} \geq 0 \qquad i \in \mathcal{A} \tag{14}$$

3.2. Re-writing probability constraints in the i.i.d. case

Assuming that for all aircraft $i \in \mathcal{A}$, the random variables $\boldsymbol{\omega_i}$ are i.i.d., the chance constraints (5) can be re-written as linear constraints. Indeed, consider a couple $(i,j) \in \mathcal{A} \times \mathcal{A}$ such that $i \neq j$. Then, (5) can be re-written as:

$$\mathbb{P}\left(\omega_i - \omega_j \leq x_j - x_i - \underline{S}^I + M_{ij}^{I\alpha}(1 - \delta_{ij}) \right) \geq \alpha$$

Let us define $\boldsymbol{\gamma_{ij}} \stackrel{def}{=} \boldsymbol{\omega_i} - \boldsymbol{\omega_j}$, and let $F_{\boldsymbol{\gamma_{ij}}}$ be its distribution function. As the $\boldsymbol{\omega_i}$'s are i.i.d., the $\boldsymbol{\gamma_{ij}}$'s are also i.i.d. and we can drop the subscript ij to use simply γ. Let us define $F_{\boldsymbol{\gamma}}^{-1}(\alpha)$ the quantile value at probability α of the random variable $\boldsymbol{\gamma}$, and the buffered separation over the IAF $S^I(\alpha) \stackrel{def}{=} \underline{S}^I + F_{\boldsymbol{\gamma}}^{-1}(\alpha)$. Therefore, the chance constraints (5) can be re-written under the form:

$$x_j \geq x_i + S^I(\alpha) - M_{ij}^{I\alpha}(1 - \delta_{ij}) \qquad (i,j) \in \mathcal{A} \times \mathcal{A}, \quad i \neq j \tag{15}$$

The best expression (smallest while sufficiently large) for the big-M constants $M_{ij}^{I\alpha}$ in (15) can easily be shown to be: $M_{ij}^{I\alpha} = L_i^I - E_j^I + \underline{S}^I(\alpha)$.

Definition 1. Let a, $a' \in \mathbb{R}^n$ and b, $b' \in \mathbb{R}$ be given. Let $x \in \mathcal{X} \subset \mathbb{R}^n$ be a vector of decision variables, where $\mathcal{X}$ is some given subset of $\mathbb{R}^n$. Then we say that $a'^T x \geq b'$ **dominates** $a^T x \geq b$ if:

- $a'^T x \geq b' \Rightarrow a^T x \geq b$, $\forall x \in \mathcal{X}$
- and $\exists\, x' \in \mathcal{X}$ such that $a'^T x' \geq b'$ and $a^T x' > b$.

One can easily show:

Proposition 1. *Consider a couple $(i,j) \in \mathcal{A} \times \mathcal{A}$ such that $i \neq j$ and the constraints:*

$$x_j \geq x_i + \underline{S}^I - M^I_{ij}(1-\delta_{ij}) \tag{4_{ij}}$$
$$x_j \geq x_i + S^I(\alpha) - M^{I\alpha}_{ij}(1-\delta_{ij}) \tag{15_{ij}}$$

where $M^I_{ij} = L^I_i - E^I_j + \underline{S}^I$ and $M^{I\alpha}_{ij} = L^I_i - E^I_j + \underline{S}^I(\alpha)$. We have the following relations between constraint (4_{ij}) and constraint (15_{ij}) when $\delta_{ij} = 1$ –otherwise, (4_{ij}) and (15_{ij}) are redundant constraints:

- *constraint (15_{ij}) dominates constraint (4_{ij}) if and only if $\alpha > \mathbb{P}(\gamma \leq 0)$*
- *constraint (4_{ij}) dominates constraint (15_{ij}) if and only if : $\alpha < \mathbb{P}(\gamma \leq 0)$*
- *constraint (4_{ij}) is equivalent to constraint (15_{ij}) if and only if : $\alpha = \mathbb{P}(\gamma \leq 0)$*

Proposition 2. *Assume that $\boldsymbol{\omega}$ is a vector of n i.i.d. normal random variables. For any value of $\alpha \geq 0.5$, constraints* (15) *can substitute for constraints* (4) *and* (5) *in the true model.*

Proof of Proposition 2 Consider $\boldsymbol{\omega}$ a vector of n i.i.d. random variables following the normal distribution with mean μ and standard deviation σ, noted $\mathcal{N}(\mu,\sigma^2)$. Then, $\boldsymbol{\gamma}$ follows the normal distribution $\mathcal{N}(0,2\sigma^2)$ and $\mathbb{P}(\gamma \leq 0) = 0.5$. Then, the proof follows directly from Proposition 1 and the fact that constraints (5) can be written as constraints (15) in the i.i.d. case. □

Remark 1. Proposition 2 may be extended to any probability distribution on an i.i.d. random vector $\boldsymbol{\omega}$ implying a symmetric distribution (with respect to zero) on the random variable $\boldsymbol{\gamma}$.

In the remainder of this article, we make the following two assumptions:

Assumption 1. *$\boldsymbol{\omega}$ is a vector of i.i.d. normal random variables.*

Assumption 2. *The protection level α from IAF separation violations is always set to values greater than (or equal to)* 0.5.

4. Solution methods

The two-stage stochastic program introduced in Section 3 presents two main challenges. The first challenge is to deal with the probability constraints in the first stage. In Subsection 3.2, we have shown that under the assumptions of i.i.d. normal random variables (Assumption 1) and large protection levels α (Assumption 2), the probability constraints can be equivalently written as linear constraints. The second challenge comes from the expectation term in the objective function of the first stage. Since we assume continuous random variables, the exact expression of the expectation term is a multivariate integral, often impracticable to compute. One widely-used method to approximate the expectation term in stochastic programs is the Sample Average Approximation (SAA) (Fu et al. 2015), that is obtained by replacing the expectation by a sample average over a finite number of scenarios. A brief description of the SAA method as well as the SAA model describing our problem are provided in Subsection 4.1. The subsequent SAA problem can be seen as a one-stage mixed integer linear problem (MILP), called the *deterministic equivalent problem*. A straightforward solution method is then to use a state-of-the-art MILP solver on the deterministic equivalent problem. Nevertheless, it is well known in the literature (Birge and Louveaux 2011) that an efficient solution method to two-stage stochastic linear programs is the L-Shaped method that derives from Benders decomposition. In Subsection 4.2, we propose a generic Benders reformulations of the SAA model, based on a partial aggregation of the second-stage problems.

4.1. Model with Sample Average Approximation

Let $\mathcal{S}$ denote the set of $n_{\mathcal{S}}$ equally-probable scenarios. We introduce the following scenario-specific notations for an aircraft $i \in \mathcal{A}$ and a scenario $s \in \mathcal{S}$: ω_i^s, y_i^s, z_i^{s-}, z_i^{s+} and z_i^{s++}. For a given scenario $s \in \mathcal{S}$, the corresponding vector notations are naturally deduced: $\omega^s = (\omega_1^s, \omega_2^s \dots \omega_n^s)^T$, $y^s = (y_1^s, y_2^s \dots y_n^s)^T$, $z^{s-} = \left(z_1^{s-}, z_2^{s-} \dots z_n^{s-}\right)^T$, $z^{s+} = \left(z_1^{s+}, z_2^{s+} \dots z_n^{s+}\right)^T$ and $z^{s++} = \left(z_1^{s++}, z_2^{s++} \dots z_n^{s++}\right)^T$. According to the SAA method, for a sufficiently large number of scenarios, $n_{\mathcal{S}}$, the objective function (1) can be replaced by:

$$\min_{\delta,\, x} \sum_{\substack{(i,j) \in \mathcal{A}^+ \times \mathcal{A}^+ \\ i \neq j}} \delta_{ij} S_{ij} + \sum_{s \in \mathcal{S}} \frac{1}{n_{\mathcal{S}}} Q\left(\delta, x, \omega^s\right) \tag{16}$$

Replacing (1) by (16) in the true model leads to the so-called *SAA model*. The SAA method relies on the uniform law of large numbers to prove that, as $n_{\mathcal{S}} \to \infty$, the SAA-model optimal objective value converges to the true-model optimal objective value (Shapiro and Homem-de Mello 2000). Using the linearized second-stage objective function proposed in Subsection 3.1, we can express the optimal value of the second-stage problem corresponding to a given scenario $s \in \mathcal{S}$, $Q\left(\delta, x, \omega^s\right)$, (as appearing in (16)) as follows:

$$Q\left(\delta, x, \omega^s\right) = \min_{\substack{y^s,\, z^{s-} \\ z^{s+},\, z^{s++}}} \sum_{i \in \mathcal{A}} \left(c_1 z_i^{s-} + c_2 z_i^{s+} + c_3 z_i^{s++}\right) \tag{17}$$

$$-y_i^s + z_i^{s++} + z_i^{s+} - z_i^{s-} = -x_i - \omega_i^s - \hat{V}_i \qquad i \in \mathcal{A} \quad (\beta_i^s) \tag{18}$$

$$-y_i^s \geq -x_i - \omega_i^s - \overline{V_i} \qquad i \in \mathcal{A} \quad (\sigma_i^s) \tag{19}$$

$$y_i^s \geq x_i + \omega_i^s + \underline{V_i} \qquad i \in \mathcal{A} \quad (\rho_i^s) \tag{20}$$

$$
\begin{aligned}
y_j^s - y_i^s &\geq S_{ij} - M_{ij}^s(1-\delta_{ij}) & (i,j)\in\mathcal{A}\times\mathcal{A}, i\neq j \quad (\pi_{ij}^s) && (21)\\
-z_i^{s+} &\geq -x_i - \omega_i^s - V_i^{\text{med}} & i\in\mathcal{A} \quad (\mu_i^s) && (22)\\
z_i^{s++},\ z_i^{s+},\ z_i^{s-} &\geq 0 & i\in\mathcal{A} && (23)
\end{aligned}
$$

where dual variables corresponding to constraints (18) to (22) are shown between parenthesis. Recall that the big-M constant M_{ij}^s can be set to $\left(L_i^I + \omega_i^s + \overline{V_i}\right) - \left(E_j^I + \omega_j^s + \underline{V_j}\right) + S_{ij}$.

The SAA model basically describes a deterministic (possibly large-scale) MILP: the deterministic equivalent problem. The extensive form of the deterministic equivalent problem is:

$$
\begin{aligned}
\min_{\substack{\delta,\, x\\ y^s,\, z^{s-}\\ z^{s+},\, z^{s++}}} \quad & \sum_{\substack{(i,j)\in\mathcal{A}^+\times\mathcal{A}^+\\ i\neq j}} \delta_{ij} S_{ij} + \sum_{s\in\mathcal{S}} \frac{1}{n_{\mathcal{S}}} \sum_{i\in\mathcal{A}} \left(c_1 z_i^{s-} + c_2 z_i^{s+} + c_3 z_i^{s++}\right)\\
\text{s.t.} \quad & (2), (3), (15), (6), (7)\\
& (18), (19), (20), (21), (22), (23)
\end{aligned} \qquad \text{(Determ. Eq.)}
$$

The deterministic equivalent problem can be directly solved using a state-of-the-art MILP solver. One weakness of the extensive form is that the problem size can become very large as the number of scenarios increases. For example, for $n = 10$ aircraft and $n_{\mathcal{S}} = 500$ scenarios, there are $20,000$ second-stage variables.

We remark that if the first-stage variables, x and δ, are fixed, then the second stage turns to be $n_{\mathcal{S}}$ separate linear programs that are straightforward to solve. This property allows us to reformulate our SAA problem using Benders decomposition, as presented in Subsection 4.2.

4.2. Benders reformulations

Using Benders decomposition (Birge and Louveaux 2011, Rahmaniani et al. 2017), we can decompose our two-stage stochastic integer problem described by the SAA model into a master problem, called *Benders master problem*, and one or many separate subproblem(s), called *Benders subproblem(s)*, corresponding to the second-stage problems. According to the level of aggregation chosen for the Benders subproblem(s), we can propose different Benders reformulations of our SAA model.

When the second-stage problems are completely aggregated, we are left with one Benders subproblem. Then, only one cut can be generated at each iteration. We call this reformulation: *simple-cut* Benders reformulation. When the second-stage problems are completely disaggregated (not aggregated at all), we have one Benders subproblem for each scenario. Accordingly, at most one cut per scenario can be generated by iteration. Hence, one can add up to $n_{\mathcal{S}}$ cuts at each iteration. We call this reformulation: *multi-cut* Benders reformulation. When the second-stage problems are aggregated into different subsets, where each subset corresponds to multiple scenarios, we say that the second-stage problems are *partially aggregated*. This yields one Benders subproblem for each subset of scenarios, called a cluster of scenarios. In this case, at most one cut per cluster can be generated at each iteration. We call this reformulation: *partially-aggregated-cut* Benders reformulation. In the following, we only present the partially-aggregated-cut Benders reformulation, since it encompasses the other two versions above, that represent the two extreme cases.

Let $\mathcal{C} = \{c_1, c_2, \dots c_K\}$ be a partition of $\mathcal{S}$, where each c_i $(i = 1, 2, \dots, K)$ is a (non-empty) subset of $\mathcal{S}$, referred to as a *cluster of scenarios*. Remark that the simple-cut version corresponds to $K = 1$, while the multi-cut version corresponds to $K = n_{\mathcal{S}}$. The second-stage problems corresponding to scenarios belonging to a same cluster are aggregated to form a single Benders subproblem. Hence, there are K Benders subproblems and, consequently, K additional optimization variables ν^c $(c \in \mathcal{C})$, are introduced to approximate the expected second-stage cost. Following the standard Benders' decomposition methodology (*e.g.*, Rahmaniani et al. (2017)), the initial Benders master problem, in the partially-aggregated-cut version, is therefore:

$$
\begin{aligned}
\min_{\delta,\, x,\, \nu} \quad & \sum_{\substack{(i,j)\in\mathcal{A}^+\times\mathcal{A}^+ \\ i\neq j}} \delta_{ij} S_{ij} + \sum_{c\in\mathcal{C}} \nu^c && (24)\\
\text{s.t.} \quad & \text{first-stage constraints:} \quad (2), (3), (15), (6), (7) && \\
& \nu^c \geq 0 \qquad\qquad c \in \mathcal{C} && (25)
\end{aligned}
$$

where $\nu = (\nu^1, \nu^2, \dots, \nu^K)^T$. Constraints (25) are obvious bound constraints on variables ν^c that strengthen the standard Benders reformulation and can be included directly in the initial Benders master problem.

Consider a (non-empty) cluster of scenarios $c \in \mathcal{C}$. The Benders subproblem corresponding to cluster c consists of $|c|$ separate scenario subproblems that can be solved separately. The results of these $|c|$ scenario subproblems are aggregated to compute the results of the Benders subproblem associated to cluster c (objective-function value, dual-variables values, etc). Let $\mathcal{R}^c$ and $\mathcal{T}^c$ be respectively the set of extreme rays and the set of extreme points of the Benders-dual-subproblem polyhedron corresponding to cluster c. Benders feasibility and optimality cuts for the partially-aggregated-cut version of our SAA model are given by constraints (26) and (27) respectively:

$$
\begin{aligned}
0 \geq \sum_{s\in c} \Bigg[\frac{1}{n_{\mathcal{S}}} \sum_{i\in\mathcal{A}} \Big[\Big(-x_i - \omega_i^s - \hat{V}_i\Big) \beta_i^{sr} + \left(-x_i - \omega_i^s - \overline{V_i}\right) \sigma_i^{sr} + \left(x_i + \omega_i^s + \underline{V_i}\right) \rho_i^{sr} \\
+ \left(-x_i - \omega_i^s - V_i^{\text{med}}\right) \mu_i^{sr} \Big] + \frac{1}{n_{\mathcal{S}}} \sum_{\substack{(i,j)\in\mathcal{A}\times\mathcal{A} \\ i\neq j}} \left(S_{ij} - M_{ij}^s (1 - \delta_{ij})\right) \pi_{ij}^{sr} \Bigg] \qquad r \in \mathcal{R}^c,\ c \in \mathcal{C}
\end{aligned}
\tag{26}
$$

$$\nu^c \geq \sum_{s \in c} \left[\frac{1}{n_{\mathcal{S}}} \sum_{i \in \mathcal{A}} \left[\left(-x_i - \omega_i^s - \hat{V}_i \right) \beta_i^{st} + \left(-x_i - \omega_i^s - \overline{V_i} \right) \sigma_i^{st} + \left(x_i + \omega_i^s + \underline{V_i} \right) \rho_i^{st} \right.\right.$$
$$\left.\left. + \left(-x_i - \omega_i^s - V_i^{\text{med}} \right) \mu_i^{st} \right] + \frac{1}{n_{\mathcal{S}}} \sum_{\substack{(i,j) \in \mathcal{A} \times \mathcal{A} \\ i \neq j}} \left(S_{ij} - M_{ij}^s (1 - \delta_{ij}) \right) \pi_{ij}^{st} \right] \qquad t \in \mathcal{T}^c,\ c \in \mathcal{C} \tag{27}$$

where we use β_i^s, σ_i^s, ρ_i^s, π_{ij}^s, and μ_i^s the dual variables associated to constraints (18) to (22) respectively, to which we add the index r or t depending upon whether we refer to an extreme ray $r \in \mathcal{R}^c$, or to an extreme point $t \in \mathcal{T}^c$.

Notes on the size of the models

The first-stage problem involves n continuous variables, $n\,(n+1)$ binary variables, and $n\,(n+1)+2$ constraints (apart from the $2n$ bound constraints on x). Regarding the second stage, one scenario subproblem involves $4n$ continuous variables, and $n\,(n+3)$ constraints (apart from the $3n$ bound constraints on z^-,z^+, and z^{++}). For $n_{\mathcal{S}}$ scenarios, the model of the deterministic equivalent problem, called the *extensive form*, requires $n\,(4n_{\mathcal{S}}+1)$ continuous variables, $n\,(n+1)$ binary variables, and $n\,(n+3)\,n_{\mathcal{S}} + n\,(n+1) + 2$ constraints.

Regardless of the degree of aggregation, Benders reformulations comprise the same number of binary variables ($n\,(n+1)$) as the extensive form, since these variables only appear in the first stage. In terms of continuous variables, the three Benders reformulations differ. The general partially-aggregated cut version has $n+K$ continuous variables, where $1 \leq K \leq n_{\mathcal{S}}$. The simple-cut version has $n+1$ continuous variables. The multi-cut version involves $n+n_{\mathcal{S}}$ continuous variables.

5. Computational study

This section aims firstly at showing the viability of our proposed model. The benefit of taking into account uncertainty is highlighted through the *value of stochastic solution* metric (Birge and Louveaux 2011). Secondly, we compare the different solution methods presented in Section 4. The remaining part of this section is organized as follows. Instances as well as parameter values are presented in Subsection 5.1. The methodology used to determine a satisfying number of scenarios, as well as our main computational results are presented in Subsection 5.2. Subsection 5.3 discusses the viability of our approach through analysis of the values of stochastic solution under different test settings. A comparison of the performance of four solution methods is presented in Subsection 5.4. Throughout this section, we shall refer to a preliminary computational study by Khassiba et al. (2018). Results are obtained on a Linux platform with 8 x 2.66 GHz Xeon processors and 32 GB of RAM. CPLEX version used is 12.6.3.

Table 4 Instance features

Total number of aircraft (n)	10
Wake-turbulence category mix	H : 70% M: 30%
IAF time windows*	narrow: [−5 min ; +5 min] wide : [−5 min ; +15 min]
Landing time window	[−1 min ; +4 min ; +19 min]

* Narrow/wide yields two different types of instance

5.1. Instances and parameter values

We consider n =10 aircraft that were planned to enter the terminal area around Paris CDG airport between 6:00 AM and 6:20 AM on May 15$^{\text{th}}$, 2015 and that landed on a same CDG runway. A more operational description of this instance is provided in Khassiba et al. 2018; some features are summarized in Table 4.

IAF time windows: We consider two possibilities for the IAF time-window width: *narrow* and *wide*, yielding two different types of instances. In the narrow instances, the aircraft IAF time window is given by $E_i^I = P_i^I - 5$ minutes and $L_i^I = P_i^I + 5$ minutes, where P_i^I is the planned IAF time for aircraft $i \in \mathcal{A}$. In the wide instances, the IAF time window is given by $E_i^I = P_i^I - 5$ minutes and $L_i^I = P_i^I + 15$ minutes.

Landing time windows: Each landing time window is piecewise defined over three time intervals related to the unconstrained landing time of each aircraft according to the form of the second-stage objective function introduced in Subsection 3.1. In our tests, deviation costs incurred within the first time segment: [−1 minutes ; 0 minutes] are proportional to the weight $-c_1$. Delays within [0 minutes ; +4 minutes] yield costs proportional to the weight c_2. Finally, delays within [+4 minutes ; +19 minutes] are proportional to the weight c_3.

Second-stage time-deviation weights: In this study, c_1, c_2 and c_3 are set to values of 0.5, 1.0 and 4.0 respectively. The main motivation is to have $0 \simeq c_1 < c_2 < c_3$.

Uncertainty: The random variables $\boldsymbol{\omega_i}$'s are i.i.d. following the normal distribution $\mathcal{N}(0, \sigma^2)$, with mean zero and standard deviation $\sigma = 30$ seconds. According to this value of standard deviation, most of the time (with probability 0.99), an aircraft will not actually arrive at the IAF later (or earlier) by more than 3 minutes with respect to its target IAF times.

Protection level against IAF separation violation: Three values for the protection level α are considered: 50%, 90% and 95%. The lowest value of α corresponds to the situation where airborne conflicts near the IAF are likely to happen at most 50% of the time (and consequently ATC must intervene to solve such conflicts). The largest value of α corresponds to rare IAF separation violations (at most 5% of the time).

Table 5 **Rounded buffered separation $S^I(\alpha)$ (in seconds) for uncertainty $\sigma = 30$ sec**

α	50%	90%	95%
$S^I(\alpha)$	72	126	142

Separations: Final-approach time separations (S_{ij}), and minimum separation over the IAF $\left(\underline{S}^I\right)$ are as indicated in Section 2. The buffered IAF separation $S^I(\alpha)$ depends on the value of the protection level α, as shown in Table 5.

5.2. Determining a satisfying number of scenarios

In order to verify whether a given number of scenarios is sufficiently large, we use the *out-of-sample validation* technique that consists in computing a *validation score* of an SAA problem's solution using a large sample of scenarios called a *validation set*. By definition, the validation set should not contain any of the scenarios used during optimization. Then, a *validation gap* is computed as the relative difference between the SAA-problem objective-function value and the validation score. In order to find a satisfying number of scenarios, $n_{\mathcal{S}}$, we propose to solve the deterministic equivalent problem using CPLEX for increasing values of $n_{\mathcal{S}}$ ranging from 50 to 500. For each number of scenarios, 10 replications of the SAA problem are constructed and solved. For each replication, a validation score is computed using a validation set of $10,000$ scenarios, and a validation gap is deduced. For a given number of scenarios, an *average validation gap* is computed (over the validation gaps of the 10 replications). As computation time increases rapidly with the number of scenarios, we are content to a satisfying number of scenarios to be the smallest number of scenarios that yields an average validation gap not smaller than -0.1%.

Results with narrow and wide IAF time windows are displayed in Tables 6 and 7 respectively. In column "CPU", the average CPLEX solving time over 10 replications is expressed in seconds. Column "$\bar{v} \pm I_{95\%}$" gives the average objective-function value, $\bar{v}$, over the 10 replications as well as the mid-length Student-based 95% confidence interval $(I_{95\%})$. The last two columns, "Validation score" and "Validation gap", report average validation scores, and average validation gaps over 10 replications. All solutions were proved optimal by CPLEX in all test cases reported in Tables 6 and 7 .

Based on the results reported in Tables 6 and 7, satisfying numbers of scenarios with narrow and wide IAF time windows are $n_{\mathcal{S}} = 200$ and 100 respectively. As remarked in Khassiba et al. 2018, the optimization problem with narrow IAF time windows is easier to solve. In fact, the smaller the IAF time windows are, the more time windows are likely to be disjoint. This may, in turn, reduce the solution space for the (NP-hard) first-stage problem. Optimal objective-function values are slightly smaller with wide IAF time windows as the problem is then more combinatorial (and therefore harder), and featuring more feasible candidate solutions.

Table 6 Results with narrow IAF time windows.

n_S	α	CPU (sec)	$\bar{v} \pm I_{95\%}$	Validation score	Validation gap
50	50%	1.5	829.5 ± 0.7	831.8	-0.3%
	90%	1.1	829.7 ± 0.8	831.7	-0.2%
	95%	1.0	830.5 ± 0.9	832.5	-0.2%
100	50%	4.0	830.0 ± 0.5	831.7	-0.2%
	90%	3.0	830.2 ± 0.6	831.7	-0.2%
	95%	2.8	831.1 ± 0.6	832.4	-0.2%
200	50%	10.8	830.5 ± 0.4	831.6	-0.1%
	90%	7.5	830.6 ± 0.4	831.5	-0.1%
	95%	6.6	831.4 ± 0.5	832.3	-0.1%
500	50%	46.1	830.9 ± 0.1	831.3	-0.1%
	90%	35.1	830.9 ± 0.1	831.4	-0.1%
	95%	32.5	831.8 ± 0.2	832.2	-0.1%

Table 7 Results with wide IAF time windows.

n_S	α	CPU (sec)	$\bar{v} \pm I_{95\%}$	Validation score	Validation gap
50	50%	6.5	826.0 ± 0.0	827.3	-0.2%
	90%	2.8	826.0 ± 0.0	827.3	-0.2%
	95%	2.6	826.0 ± 0.0	827.5	-0.2%
100	50%	62.4	826.1 ± 0.1	827.0	-0.1%
	90%	27.1	826.1 ± 0.1	827.0	-0.1%
	95%	24.3	826.1 ± 0.1	827.0	-0.1%
200	50%	258.5	826.3 ± 0.1	826.9	-0.1%
	90%	136.0	826.3 ± 0.1	826.9	-0.1%
	95%	105.2	826.3 ± 0.1	826.9	-0.1%
500	50%	1466.0	826.4 ± 0.1	826.8	-0.0%
	90%	670.9	826.4 ± 0.1	826.7	-0.0%
	95%	493.2	826.4 ± 0.1	826.7	-0.0%

5.3. Value of the stochastic solution

The *value of the stochastic solution* (VSS) expresses the benefit of solving a two-stage stochastic problem over simply implementing a policy where the decision maker waits until the uncertainty is revealed to react (also known as *wait-and-see* policy). Let $v^\star_{SP}$ be the optimal objective-function value of the two-stage stochastic problem (SP), represented by the true problem in our context. Let $v^\star_{W\&S}$ be the optimal objective-function value obtained under the wait-and-see policy. Then, the VSS is defined as the absolute difference between $v^\star_{SP}$ and $v^\star_{W\&S}$. The *relative VSS* is defined as follows:

$$\text{VSS}(\%) \quad \stackrel{\text{def}}{=} \quad 100 \times \frac{v^\star_{SP} - v^\star_{W\&S}}{v^\star_{W\&S}}$$

In order to compute the objective-function value under the wait-and-see policy, $v^\star_{W\&S}$, we start by solving the expected-value problem (EP) defined as the "stochastic" problem when a single scenario,

Table 8 Relative VSS with both IAF time-window lengths.

		Narrow			Wide		
	α	50%	90%	95%	50%	90%	95%
n_S	50	-7.50%	-2.28%	-1.28%	-10.93%	-2.34%	-1.83%
	100	-7.43%	-2.22%	-1.22%	-10.92%	-2.32%	-1.81%
	200	-7.38%	-2.17%	-1.18%	-10.90%	-2.30%	-1.79%
	500	-7.34%	-2.13%	-1.14%	-10.89%	-2.29%	-1.78%

specifically the average scenario, is considered. Remark that, defined as such, the (EP) completely overlooks uncertainty. Let $(\delta^\star_{EP}, x^\star_{EP})$ be an optimal solution of the (EP). When the first-stage solution is $(\delta^\star_{EP}, x^\star_{EP})$, the objective-function value of (SP) represents the optimal objective-function value obtained under the wait-and-see policy, $v^\star_{W\&S}$.

The relative VSS for both IAF time-window lengths, for different values of protection level, α, and for increasing numbers of scenarios, $n_{\mathcal{S}}$, are reported in Table 8. With a low protection level against IAF separation violations ($\alpha = 50\%$), relative VSS's are less than -7% for narrow IAF time windows, and less than -10% for wide IAF time windows. Therefore, solving a two-stage stochastic problem is clearly more beneficial than waiting until the uncertainty is revealed to react. However, relative VSS's dramatically increase (up to almost -1%) when increasing the protection level, α, expressing thereby a smaller benefit of solving a two-stage stochastic program. Recall that for high values of α, the IAF separation is enlarged as shown in Table 5. Such buffered separations contribute to hedge against uncertainty as follows. If target IAF times are spaced out more than the minimal requirement $\underline{S}^I$, the actual IAF times are expected to be less disrupted when the uncertainty is revealed. Therefore, the recourse cost to restore the target sequence while not deviating much from the uncontrained landing times, is expected to be smaller. Consequently, hedging against uncertainty using buffered separations may lead to a better performance of the wait-and-see policy. Moreover, we remark that the higher the number of scenarios the higher the relative VSS. This may be explained through the fact that objective-function values of SAA problems with limited numbers of scenarios are negatively biased with respect to the true-problem objective-function value. Therefore, computing VSS using a small number of scenarios for (SP) slightly overestimates the benefit of solving a two-stage stochastic program.

As different IAF time-window lengths are studied, we remark that with wide IAF time-windows, the benefit of solving a two-stage stochastic problem is greater. Recall that in our study, we limit the delay with respect to the IAF planned time to 15 minutes (for wide IAF time windows), while in the literature (Balakrishnan and Chandran 2010), potential delay can be extended up to one hour.

5.4. Solution-method performance comparison

Modern implementations of Benders decomposition often refer to integrating Benders cuts into a Branch-and-Cut process, giving rise to Branch-and-Benders-Cut. To improve on solving times of the deterministic equivalent directly by CPLEX, we propose three solution methods based on modern implementations of Benders decomposition: one with multiple cuts, one with partially-aggregated cuts, and one with simple cuts. In the partially-aggregated-cut version, the clustering policy simply corresponds to aggregating every two scenarios in a same cluster ($K = \frac{n_{\mathcal{S}}}{2}$ clusters). In Tables 9 and 10, we report average solving times (over 10 replications) of four solution methods: the deterministic equivalent solved directly by CPLEX, noted "Determ. Eq.", multi-cut Benders decomposition, noted "MC-Benders", partially-aggregated-cut Benders decomposition, noted "PAC-Benders", and simple-cut Benders decomposition, noted "SC-Benders". A time limit of one hour, noted "TiLim", was enforced. Based on results in Tables 9 and 10, the simple-cut Benders decomposition is clearly the least competitive solution method. Solving the deterministic equivalent directly with CPLEX appears to be efficient in most cases. However, CPLEX efficiency to solve the deterministic equivalent problem dramatically decreases as the number of scenarios increases, especially with wide IAF time windows. For example, with wide IAF time windows and $n_{\mathcal{S}} = 500$ scenarios, CPLEX average CPU times exceed 8 minutes for all test values of protection level, α. Therefore, the implementation of such a solution method in a real-time context may be questioned. On the other hand, both the multi-cut and the partially-aggregated-cut Benders decompositions outperform the deterministic equivalent for the cases with wide time windows, $\alpha \geq 90\%$ and $n_{\mathcal{S}} \geq 100$. In all these cases, the multi-cut version performs slightly better than the partially-aggregated-cut version. For example, with wide IAF time windows, $n_{\mathcal{S}} = 500$ scenarios and $\alpha = 90\%$, the multi-cut and the partially-aggregated-cut Benders decompositions improve CPLEX average CPU time by more than 84% and more than 80% respectively.

6. Conclusion and perspectives

In this paper, we propose a chance-constrained two-stage stochastic mixed-integer programming model for the extended aircraft arrivals management problem under uncertainty. In the first stage, aircraft are captured 2 to 3 hours away from the IAF. The first-stage problem is to find a target sequence and target times of aircraft arrival over the IAF so as to minimize the landing sequence length. First-stage constraints are IAF time windows and IAF separation constraints. The first-stage problem is enriched by chance constraints to limit the risk of IAF separation violations to an acceptable level, that we call the protection level. We show that under mild conditions first-stage chance constraints can be transformed into linear separation constraints with buffered minimal IAF separation that depends on the protection level. The second-stage problem considers aircraft shortly

Table 9 **Average CPU time (seconds) with narrow IAF time windows.**

n_S	Method	α 50%	90%	95%
50	Determ. Eq.	**1.54**	**1.06**	**0.99**
	MC-Benders	7.01	4.14	3.30
	PAC-Benders	15.81	5.39	4.07
	SC-Benders	997.95	149.04	53.18
100	Determ. Eq.	**3.98**	**2.96**	**2.75**
	MC-Benders	17.63	7.96	6.23
	PAC-Benders	24.02	11.85	9.29
	SC-Benders	3005.49	445.23	167.91
200	Determ. Eq.	**10.79**	**7.48**	**6.58**
	MC-Benders	40.73	15.59	11.28
	PAC-Benders	71.29	25.61	18.96
	SC-Benders	TiLim	1521.98	629.64
500	Determ. Eq.	**46.08**	**35.10**	**32.48**
	MC-Benders	75.40	49.10	31.52
	PAC-Benders	221.15	63.95	58.77
	SC-Benders	TiLim	TiLim	TiLim

Table 10 **Average CPU time (seconds) with wide IAF time windows.**

n_S	Method	α 50%	90%	95%
50	Determ. Eq.	**6.47**	**2.81**	**2.56**
	MC-Benders	23.24	4.75	3.83
	PAC-Benders	46.95	8.30	6.12
	SC-Benders	1772.69	556.69	371.76
100	Determ. Eq.	**62.38**	27.09	24.33
	MC-Benders	133.78	**15.75**	**12.09**
	PAC-Benders	448.56	23.04	13.70
	SC-Benders	3489.72	3049.71	2499.45
200	Determ. Eq.	**258.49**	136.02	105.18
	MC-Benders	890.25	**35.10**	**31.93**
	PAC-Benders	507.48	44.49	38.52
	SC-Benders	TiLim	TiLim	TiLim
500	Determ. Eq.	**1466.01**	670.91	493.16
	MC-Benders	2444.29	**106.60**	**71.45**
	PAC-Benders	2789.83	133.07	71.93
	SC-Benders	TiLim	TiLim	TiLim

before arriving at the IAF up to landing. It aims at finding target landing times so as to minimize a time-deviation impact cost function. Second-stage constraints are landing time windows and final-approach separations. The two-stage stochastic program minimizes the sum of the landing sequence length, and the expected second-stage time-deviation impact cost function. Chance constraints are linearized, and the expectation term is approximated through Sample Average Approximation. We are then left with a large-scale deterministic mixed-integer linear problem. Apart from solving directly the deterministic equivalent problem by CPLEX, we propose two Benders-decomposition reformulations: multi-cut, partially-aggregated-cut, and simple-cut versions. Our computational study shows that it is more beneficial to solve a two-stage stochastic program than to wait until uncertainty is revealed to react (wait-and-see policy). Moreover, we observe a decrease of the values of the stochastic solution (VSS) with the presence of chance constraints. This indicates that chance constraints contributes significantly to the benefit of solving a stochastic program over the wait-and-see policy. With regard to solution-method performance, results on instances of 10 aircraft show that the deterministic equivalent is efficiently solved by CPLEX, while our multi-cut version of Benders decomposition provides shorter computing times for instances with wide IAF time windows and high protection levels.

Future work will focus on extending the proposed model to the case with multiple IAFs and multiple runways. Our perspectives also include solving the dynamic case where the arrival set evolves in time. In terms of solution methods based on partially-aggregated-cut Benders decomposition, more scenario clustering policies can also be explored.

Acknowledgments

The authors would like to thank Serge Roux (ENAC) for providing the arrivals data on CDG airport. F. Bastin and B. Gendron are funded by Canada NSERC Discovery Grants.

References

Balakrishnan H, Chandran BG, 2010 *Algorithms for scheduling runway operations under constrained position shifting. Operations Research* 58(6):1650–1665.

Beasley JE, Krishnamoorthy M, Sharaiha YM, Abramson D, 2000 *Scheduling aircraft landings: The static case. Transportation Science* 34(2):180–197.

Bennell JA, Mesgarpour M, Potts CN, 2011 *Airport runway scheduling. 4OR* 9(2):115–138.

Bennell JA, Mesgarpour M, Potts CN, 2017 *Dynamic scheduling of aircraft landings. European Journal of Operational Research* 258(1):315–327.

Birge JR, Louveaux F, 2011 *Introduction to stochastic programming* (Springer Science & Business Media).

Bosson CS, Sun D, 2016 *Optimization of airport surface operations under uncertainty. Journal of Air Transportation* 24(3):84–92.

Chandran BG, Balakrishnan H, 2007 *A dynamic programming algorithm for robust runway scheduling. Proceedings of the 2007 American Control Conference*, 1161–1166 (IEEE).

de Neufville R, Odoni A, Belobaba P, Reynolds T, 2013 *Airport Systems: Planning, Design and Management* (McGraw-Hill), 2nd edition.

Dear RG, 1976 *The dynamic scheduling of aircraft in the near terminal area.* Technical report, Massachusetts Institute of Technology.

Division-NAAS, 2016 *Traffic management advisor*. URL `https://www.aviationsystemsdivision.arc.nasa.gov/research/foundations/tma.shtml`.

Fu MC, et al., 2015 *Handbook of Simulation Optimization*, volume 216 (Springer).

Furini F, Kidd MP, Persiani CA, Toth P, 2015 *Improved rolling horizon approaches to the aircraft sequencing problem. Journal of Scheduling* 18(5):435–447.

Garcia JL, 1990 *MAESTRO-A metering and spacing tool. Proceedings of the 1990 American Control Conference*, 502–507 (IEEE).

Hasevoets N, Conroy P, 2010 *Arrival Manager - Implementation Guidelines and Lessons Learned*. Technical report, EUROCONTROL, URL `https://www.eurocontrol.int/sites/default/files/article/content/documents/nm/fasti-aman-guidelines-2010.pdf`.

Heidt A, 2017 *Uncertainty Models for Optimal and Robust ATM Schedules*. Ph.D. thesis, Friedrich Alexander University, Erlangen, Germany.

Heidt A, Helmke H, Kapolke M, Liers F, Martin A, 2016 *Robust runway scheduling under uncertain conditions. Journal of Air Transport Management Part A* 56:28–37.

Kapolke M, Fürstenau N, Heidt A, Liers F, Mittendorf M, Weiß C, 2016 *Pre-tactical optimization of runway utilization under uncertainty. Journal of Air Transport Management Part A* 56:48–56.

Khassiba A, Bastin F, Cafieri S, Mongeau M, Gendron B, 2018 *Aircraft arrival management under uncertainty: A computational study*. Technical Report 2018-52, CIRRELT, Université de Montréal, Canada.

Lee H, 2008 *Tradeoff evaluation of scheduling algorithms for terminal-area air traffic control*. Master's thesis, Massachusetts Institute of Technology.

Meyn LA, Erzberger H, 2005 *Airport arrival capacity benefits due to improved scheduling accuracy. Proceedings of the 5th Aviation, Technology Integration and Operations and the 16th Lighter-Than-Air Systems Technology and Balloon Systems Conferences*.

Miller CE, Tucker AW, Zemlin RA, 1960 *Integer programming formulation of traveling salesman problems. Journal of the ACM* 7(4):326–329.

Neuman F, Erzberger H, 1990 *Analysis of sequencing and scheduling methods for arrival traffic*. Technical report, National Aeronautics and Space Administration.

Rahmaniani R, Crainic TG, Gendreau M, Rei W, 2017 *The benders decomposition algorithm: A literature review. European Journal of Operational Research* 259(3):801–817.

Shapiro A, Homem-de Mello T, 2000 *On the rate of convergence of optimal solutions of Monte Carlo approximations of stochastic programs. SIAM Journal on Optimization* 11(1):70–86.

Sölveling G, 2012 *Stochastic programming methods for scheduling of airport runway operations under uncertainty*. Ph.D. thesis, Georgia Institute of Technology, Atlanta, GA.

Sölveling G, Clarke JP, 2014 *Scheduling of airport runway operations using stochastic branch and bound methods. Transportation Research Part C* 45:119–137.

Sölveling G, Solak S, Clarke JP, Johnson E, 2011 *Runway operations optimization in the presence of uncertainties. Journal of Guidance, Control, and Dynamics* 34(5):1373–1382.

Tielrooij M, Borst C, Van Paassen MM, Mulder M, 2015 *Predicting arrival time uncertainty from actual flight information. Proceedings of the 11th USA/Europe Air Traffic Management Research and Development Seminar.*

Völckers U, 1990 *Arrival planning and sequencing with COMPAS-OP at the Frankfurt ATC-Center. Proceedings of the 1990 American Control Conference*, 496–501 (IEEE).