13th USA/Europe Air Traffic Management Research and Development Seminar (ATM 2019)

Departure Management with Robust Gate Allocation

Ruixin Wang*†, Cyril Allignol†, Nicolas Barnier†, Jean-Baptiste Gotteland†
* China-France Joint Research Center of Applied Mathematics for ATM
Sino-European Institute of Aviation Engineering, CAUC, Tianjin, China
ruixin.wang@recherche.enac.fr
† ENAC, Université de Toulouse, France
{cyril.allignol,nicolas.barnier,jean-baptiste.gotteland}@enac.fr

***Abstract*—The Airport Collaborative Decision Making (A-CDM) concept yields concrete and promising solutions for airports, in terms of traffic punctuality and predictability, with possible delay, noise and pollution reduction. A key feature of A-CDM is Departure Management (DMAN): runway take-off sequences can be anticipated such that a significant part of the delay can be shifted at the gate, engines off, without penalizing the remaining traffic. During this process, an increase in the gate occupancy for delayed departures is unavoidable, therefore the airport layout must provide enough gates and their allocation must be robust enough w.r.t. departures delay.**

In this paper, we introduce a method to estimate the gate delays due to the DMAN pre-departure scheduling, then we propose a robust gate allocation algorithm and assess its performance with current and increased traffic at Paris-Charles-de-Gaulle international airport. Results show a significant reduction in the number of gate conflicts, when comparing such a robust gate allocation to current practice.

***Index Terms*—departure management, gate allocation, robustness**

I. Introduction

All airports have to handle many challenges regarding traffic increase, delay reduction, traffic predictability and environmental impact. In this scope, the Airport Collaborative Decision Making (A-CDM) concept offers concrete solutions for airports to improve the management of these issues.

A-CDM was first developed to help the airport stakeholders in adverse conditions (e.g. bad weather, capacity reduction, security, systems failure, etc.), but the European A-CDM program also includes the definition of some new organization and systems, in order to bring significant improvements in normal operating conditions:

- Optimizing the use of existing resources at any time can decrease delay and improve punctuality.
- Avoiding surface congestion enables to reduce noise and pollution.
- Estimating taxiing times more accurately, by including anticipated delay, increases the traffic predictability, not only at the considered airport but also for the other components of the air traffic network like other airports or the approach and en-route control sectors.

Many European airports have already obtained the A-CDM certification and many others are currently in the process of its implementation [1]. Even if this program is often summarized as information sharing between different stakeholders, it also includes a Departure Management (DMAN) process, which is an essential feature in normal operating conditions: its output is a pre-departure sequence that aims at keeping delayed departures at gate, engines off and apart from the rest of the traffic rather than on the taxiways or near the runways. The benefits of DMAN are threefold:

- runways takeoff sequences are anticipated;
- unavoidable delays due to the runways capacity are estimated;
- realistic taxiing times are taken into account in order to shift a significant part of these delays at the gates, without penalizing the departure flow.

The DMAN process of the European A-CDM concept has proven to be very efficient, once successfully calibrated to each airport: in particular, the taxiing times used to shift the delay at the gates is known to be a very sensitive parameter during the implementation. Another unavoidable effect of this process is to increase the gate occupancy before departure, notably during traffic peaks where the amount of delay due to runways capacities increases.

However, two measures can be taken to mitigate these effects:

- The airport layout must offers enough gates in order to accommodate the additional gate occupancy.
- The gate allocation must be compatible with the departures delays, which are not evenly distributed among all flights in a given terminal.

While the former is very expansive, the latter is not an easy task: the initial gate allocation must be achieved well before the tactical DMAN process, and deviations from the plan can result in very complex situations in which arrivals have to wait a long time before a suitable gate becomes available.

For these reasons, A-CDM-enabled airports need a gate allocation process that produces more robust schedules w.r.t. departures delays. To overcome this issue, we propose in this study to:

- Estimate the increase in gate occupancy due to the DMAN process, on a major international airport like Paris-Charles-de-Gaulle (Paris-CDG).
- Design an initial gate allocation method to optimize the robustness w.r.t. departure delays and assess its performance.

- Evaluate the scalability of this scheme w.r.t. the expected traffic growth.

After an analysis of existing works related to departure management and gate allocation, this article presents the strategy that was chosen to estimate the additional gate occupancy for departures, by simulating the DMAN process at Paris-CDG airport, and the general formulation and resolution method used to obtain a robust gate allocation for each terminal of the airport. Eventually, we describe the results obtained when combining these two optimization processes, first in terms of additional gate occupancy, then by measuring the impact of a robust gate allocation over the initial one on the amount of gate conflicts.

II. Related Works

In this section, we first review the state of the art on the optimization of the departure process at busy airport in an A-CDM context. Then we mention the main models of the well-studied Gate Allocation Problem, their diverse objectives and the interest to optimize its robustness prior to the DMAN sequencing.

A. Airport Collaborative Decision Making and Departure Management

The A-CDM program [1] is the result of a large body of research and development regarding the sharing of information and the management of departures at the airport level. In Europe, this program is already implemented at many airports and its positive effects on local and network operations have already been measured [2].

The main idea of the departure management process is to avoid surface congestion, by sequencing the departures from their gate positions rather than on the taxiways or in front of the runways. This task can involve tools and displays to support humans decisions: in [3], the authors conduct a functional analysis to define the needs and the role of a system called Departure Reservoir Coordinator that would support human controllers in the task of departure metering. In [4], the author provides a comprehensive analysis of the departure operations including the characterization of the airport capacity, and proposes a model of the departure management process, in order to define and assess the efficiency of algorithms designed to control the departure flow. The author measures the gate-hold times and the corresponding fuel burn and emissions that can be saved, but does not investigate the consequent gate conflicts that can appear between departures and arrivals.

The central point of the departure management process is based on accurate estimates of runway delays and of the part of these delays that can be shifted at the gate without losing the takeoff rate at the runway level. This process requires the computation of feasible runway takeoff sequences, which also is a combinatorial optimization problem. Indeed, takeoff sequences must be optimized, can be constrained by Calculated TakeOff Times (CTOT), and are linked to the departure routing problem, as mentioned in [5] and [6] where DMAN is applied on real traffic at the London Heathrow and Paris-CDG airports respectively.

The goal of this article is to estimate the impact of such departures metering strategies on gate occupancy and the possible gate conflicts that can appear, which has been much less studied in the literature. One of the most complete work on this topic that we are aware of is [7]: the author provides a rigorous analysis of ramp operations and congestion, proposes various methods to optimize the gate assignment w.r.t. passengers, aircraft or operations, and a queueing model to meter the departure process and the number of gate conflicts w.r.t. the gate assignment strategy.

B. Gate Allocation Problem

The GAP[1] consists in finding an allocation of airport gates to arriving traffic with fixed occupancy periods, such that one aircraft at most is assigned to a given gate at any time. This problem has been studied since several decades with many variants as mentioned in [8].

If there were no compatibility restriction between gates and aircraft, the corresponding decision problem could be modeled as the *coloring* of an *interval graph*, which is polynomial [9] for the minimization of the number of colors (i.e. gates). Airport gates are generally not equivalent resources though, as they are dimensioned to accommodate specific types of aircraft. Therefore, the set of compatible gates for an aircraft usually is a strict subset of all the available gates and the decision version of the allocation problem is rather a *list-coloring* problem, which is NP-Complete even for interval graphs [10]. Other operational side constraints like the simultaneous occupancy of adjacent gates for large aircraft appear in some of the most refined models.

Moreover, gates may also be endowed with other secondary features (e.g. compatible airlines, domestic or international, terminal or apron) which should match the characteristics of the flight and the preferences of airlines as much as possible. These preferences can be modeled as costs associated with each possible assignment, and standard GAP objectives often include the minimization of their sum, which is NP-Hard [11]. Other classic objectives can be the passengers walking distance [7] (or other connection means like buses), which is similar to the *Quadratic Assignment Problem*, or the number of towing movements [12].

In order to absorb possible deviations from the original schedule, because of the DMAN sequencing (or other factors like traffic delays, severe weather conditions, equipment failures...), and avoid costly disruptions, our study rather focuses on optimizing the robustness of the allocation as proposed by [13], which minimizes the *variance of idle times* to balance and spread them over time and resources. Despite its practical importance, research on the robustness of solutions to the GAP is quite limited.

To solve these very diverse variants of the GAP, many classic combinatorial optimization methods were experimented,

[1]This problem is sometimes called Airport Gate Allocation Problem (AGAP) or Stand Allocation Problem (SAP) in the literature.

depending on the linearity of the model, the size of the instances and the requirements on the execution time of the solver. One of the most used tools is Mixed Integer Programming (MIP) or Integer Linear Programming (ILP) solvers like CPLEX or Gurobi to obtain proved optimal solutions like [12], [14]. Constraint Programming was also experimented by [15], [16] to solve the GAP as a *scheduling* problem similar to Fixed Job Scheduling (FJS). As previously mentioned, all considered variants of the GAP are NP-Complete or NP-Hard, so approximation algorithms like metaheuristics (e.g. Genetic Algorithm, Tabu Search) have also been used by [7], [14] to solve large instances in reasonable time or non-linear models.

III. Departure Management at Paris-CDG

In this section, we focus on the sequence of departures on each runway at Paris-CDG airport, in order to estimate the delay that can be shifted at the gate by the Departure Management (DMAN) process. We first describe the context and remind the overall principles of DMAN, and then detail the method used to estimate feasible optimized take-off sequences on each runway.

A. Simulation of the Departures Scheduling Process

We consider that a fixed terminal and a fixed runway are assigned to each departure, as it is often the case at Paris-CDG airport, where the terminal of each flight is normally defined long before it happens, and where runway assignment does not depend on the terminal, but on the departure route of the aircraft or its exit point from the approach sectors. In standard configuration (East or West), Paris-CDG airport offers two doublets of specialized runways (see Figure 1): 27R-09L, 27L-09R to the north and 26L-08R, 26R-08L to the south, with an arrival and a departure runway in each doublet, so that takeoff sequencing does not directly interfere with landing sequencing.

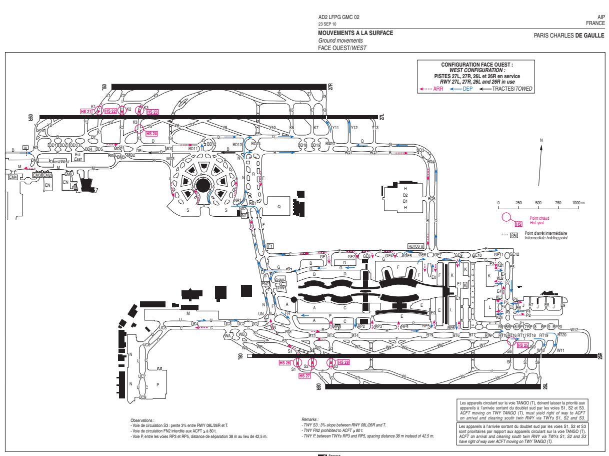

Fig. 1. Paris-CDG international airport map (West configuration).

Each departure has an EOBT (Estimated Off-Block Time) given by the airline that can be added to an estimate of the taxi-out time (depending on the gate and the runway used) to obtain the estimate of the minimal takeoff time of the flight.

For a given period of the day and a given runway, feasible takeoff times are deduced from the minimal takeoff times by taking into account the runway constraints:

- the runway separation rules, which can be modeled as a minimal separation time between each pair of aircraft, depending on their wake turbulence category;
- the Calculated TakeOff Time (CTOT) of the concerned departures, which specifies a time interval of $15\,\mathrm{min}$ in which each aircraft is constrained to take off.

The resulting runway sequence is given by the TTOT (Target TakeOff Time) of each aircraft, which is a delayed takeoff time compared to the initial (minimal) one. In case of a tactical slot allocation, the TTOT already takes into account the CTOT assigned by the network management center.

In the DMAN process, the estimate of the taxi-out time of each aircraft (sometimes re-calibrated for operational reasons, specific to the airport) is then subtracted from the TTOT to obtain the TSAT (Target Start-up Approval Time), and the TOBT (Target Off-Block Time), equal to or later than the initial EOBT. Thus, the difference between the TOBT and EOBT is the additional gate occupancy for the departure.

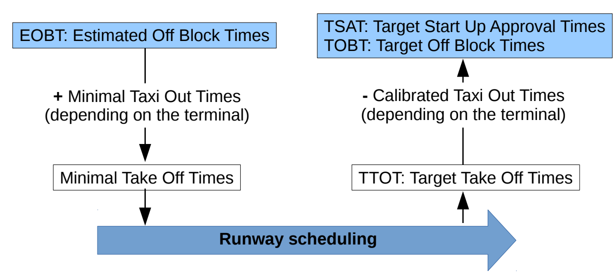

Fig. 2. Departure management.

As illustrated on Figure 2, the crux of the departure scheduling is the computation of the takeoff sequence on each runway, given the minimal takeoff times of each departure. In this article, we propose to compute an optimal runway sequence for each $30\,\mathrm{min}$ period of the day, in order to minimize the additional gate occupancy that will result from it. The minimization criterion is therefore the total delay due to the takeoff sequence.

B. Runway Scheduling

For a set of n departures scheduled on a given period on a given runway, we formulate the departure runway scheduling problem as follows:

- The decision variables are the target takeoff times of each departure:

$$\mathcal{T} = \{t_i, \forall i \in [1, n]\} \tag{1}$$

- The domain of each variable t_i takes into account the minimal takeoff time $T_i^{\min}$ and the maximal takeoff time $T_i^{\max}$ of the regulated departures of set $\mathcal{C}$ that are constrained with a CTOT:

$$T_i^{\min} \leqslant t_i, \forall i \in [1, n] \tag{2}$$

$$t_i \leqslant T_i^{\max}, \forall i \in \mathcal{C} \tag{3}$$

- The constraints of the problem are the separation rules on the runway, modeled as a minimal separation time $\mathrm{Sep}_{i,j}$ between each successive takeoffs i and j, depending on their wake turbulence categories:

$$t_i + \mathrm{Sep}_{i,j} \leqslant t_j \ \vee \ t_j + \mathrm{Sep}_{j,i} \leqslant t_i, \forall i \neq j \in [1, n] \tag{4}$$

- The minimization criterion is the total delay generated by the takeoff sequence:

$$f(\mathcal{T}) = \sum_{i=1}^{n} t_i - T_i^{\min} \tag{5}$$

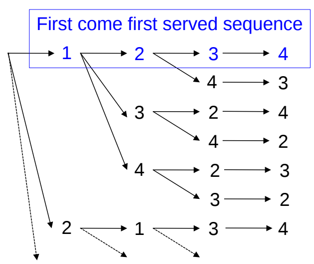

Fig. 3. Permutation tree.

This formulation corresponds to a standard scheduling problem, known to be highly combinatorial with the number of variables (aircraft). However, as explained in [6], by taking advantage of the numerous symmetries of the problem and by pruning as many sub-optimal parts of the search space as possible, a *Branch and Bound* algorithm can find an optimal solution in a very short time (a few seconds), for small instances corresponding to a period from 30 to $60\,\mathrm{min}$ of planned traffic on the runway (less than 100 aircraft). The algorithm begins with the first-come first-served sequence (i.e. aircraft are initially sorted by their minimal takeoff time) and explores the permutation tree (see Figure 3) while pruning the subtrees that cannot improve the best solution so far:

- During the exploration of the permutation tree, when the delay resulting from the already instantiated variables becomes higher than the one of the best known solution so far, the current node is pruned (and the algorithm backtracks).
- As the exploration starts with the first-come first-served sequence, there is no advantage to swap *equivalent* aircraft. Two aircraft are equivalent when they have the same wake vortex category and the same class of CTOT constraints: both without CTOT or both with CTOTs in the same order as their minimal takeoff times.
- Many other sub-optimal sequences can be pruned, because it cannot be optimal to schedule an aircraft j before an aircraft i that could be scheduled earlier without penalizing it, as this would result in an idle period on the runway during which the aircraft i could be scheduled. According to this rule, once a variable t_i is instantiated, all the permutations of i and j verifying the following inequation are pruned:

$$t_i + \mathrm{Sep}_{i,j} \leqslant T_j^{\min} \tag{6}$$

In order to obtain a valid runway sequence for the whole day of traffic, the problem is solved for each successive $30\,\mathrm{min}$ period and for each departure runway, adding to each period the departures that could not be scheduled before the end of the last period, if any. As explained in the previous section, the results of this scheduling process are the TTOT and the TOBT for each departure, which directly give the estimate of the additional gate occupancy due to the DMAN process.

IV. Gate Allocation Problem

In the following sections, we first present a simple integer *scheduling* model of the GAP which can directly be solved by Constraint Programming as in [16] for small instances. Then we describe a (much larger) boolean ILP version (similar to one of the models of [14]), which state-of-the-art MIP solvers can optimize using the *Branch and Cut* algorithm mentioned in Section IV-D, even for realistic instances at major international airports as shown in Section V for Paris-CDG.

A. Instance

An instance of the GAP is defined by:

- $\mathcal{F} = \{f_1, \ldots, f_n\}$ a set of n flights (or *tasks*), with $\forall f_i \in \mathcal{F}$:
  - f_i^s and f_i^e the start and end times of the gate occupancy by f_i;
  - $\mathcal{G}_i \subseteq \mathcal{G}$ a set of compatible gates which can be assigned to the aircraft.
- $\mathcal{G} = \{g_1, \ldots, g_m\}$ the set of m gates (or *resources*), with $\forall g_j \in \mathcal{G}$:
  - g_j^s and g_j^e the opening and closing times of gate g_j. However, except when mentioned otherwise, all gates are considered available during the same period[2] in the following, therefore $\forall j, g_j^s = g^s$ and $g_j^e = g^e$.
  - $\mathcal{F}_j = \{f_i \in \mathcal{F} \text{ s.t. } g_j \in \mathcal{G}_i\}$ the set[3] of compatible aircraft that can be executed on gate g_j.

B. Scheduling Model

The scheduling of tasks with fixed start and end times on non-identical resources is a versatile NP-complete problem [17] which occurs in many applications beside the GAP, like processors scheduling or staff rostering. Though various objectives can be associated with this problem, our approach is dedicated to the optimization of resource costs based on the idle times to ensure the robustness of solutions w.r.t. delays.

[2]Note that this is not a limitation as unavailability periods over a given gate can easily be modeled by additional aircraft with a single compatible resource.

[3]Redundantly defined from $\mathcal{G}_i$ to simplify notations in the next section.

1) Decision Variables: A solution to the GAP consists in assigning a gate to each aircraft while satisfying the constraints described in the next section. We define the set of decision variables associated to the aircraft of $\mathcal{F}$ as:

$$\mathcal{X} = \{x_i \in \{j \text{ s.t. } g_j \in \mathcal{G}_i\}, \forall f_i \in \mathcal{F}\}$$

2) Constraints: The only constraints of this essential version of the problem are the non-overlapping of the aircraft occupancies scheduled on the same gate. As aircraft occupancy start and end times are fixed, we require that overlapping aircraft are assigned to different resources:

$$\forall i \neq i',\ [f_i^s, f_i^e[\cap [f_{i'}^s, f_{i'}^e[\neq \varnothing \Rightarrow x_i \neq x_{i'} \tag{7}$$

3) Cost: Many different kind of costs can be taken into account to optimize the allocation of fixed tasks on non-identical resources. For the GAP, one of the most crucial objectives is the robustness of the schedule to prevent the DMAN sequencing (and other delay-inducing uncertainties that burden air traffic) from disrupting airport operations. To be able to absorb those possible delays, [13] proposes to minimize the variance of *idle times*, which tends to balance them over resources and time while allowing necessary short or large pauses required by some instances.

Since all tasks must be scheduled, the sum of the duration of the $n+m$ idle times (one before each task and an extra one before the closing of each resource), and therefore their mean, are constant. Hence, minimizing the variance of idle times amounts to minimizing the sum of their squares:

$$\text{cost} = \sum_{g_j \in \mathcal{G}} c_j \tag{8}$$

where c_j is the cost of a single resource g_j:

$$c_j = (f_{\text{first}(g_j)}^s - g_j^s)^2 + \sum_{f_i \in \mathcal{F}_j \text{ s.t. } x_i = j} (\text{next}(f_i) - f_i^e)^2$$

with:

- $\text{next}(f_i) = g_j^e$ if f_i is the last task assigned on resource g_j or the start time of the task immediately following f_i on g_j otherwise;
- $\text{first}(g_j)$ is the index of the first task scheduled on resource g_j.

Note that our approach could be generalized to any convex objective function of the idle times to evenly distribute them between tasks and among resources.

C. Integer Linear Programming Model

In this section, we describe an ILP model of the GAP similar to the model called "P5" in [14] and designed to obtain a linear expression for the cost specified in Section IV-B3, i.e. the sum of the square of all idle times for all the gates of a given terminal.

1) Decision Variables: First, we define the set $\mathcal{X}_{\text{ILP}}$ of binary decision variables $x_{i,j,k}$, which are equal to 1 iff flight f_j immediately succeeds to flight f_i on gate g_k, as the union of the four following sets defined by Equations 9 to 12, provided that the flights f_i are ordered by increasing start times f_i^s (so that only pairs of ordered indices $i < j$ need to be considered):

$\forall i < j \in [1, n],$ s.t. $f_i^e \leqslant f_j^s$, $\forall g_k \in \mathcal{G}_i \cap \mathcal{G}_j$,

$$x_{i,j,k} = \begin{cases} 1 & \text{if } f_j \text{ directly succeeds to } f_i \text{ on } g_k \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

$\forall j \in [1, n]$, $\forall g_k \in \mathcal{G}_j$,

$$x_{0,j,k} = \begin{cases} 1 & \text{if } f_j \text{ is the first flight of } g_k \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

$\forall i \in [1, n]$, $\forall g_k \in \mathcal{G}_i$,

$$x_{i,n+1,k} = \begin{cases} 1 & \text{if } f_i \text{ is the last flight of } g_k \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

$\forall k \in [1, m]$,

$$x_{0,n+1,k} = \begin{cases} 1 & \text{if there is no flight assigned to } g_k \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

Note that flight indices in interval $[1, n]$ correspond to regular flights that belong to $\mathcal{F}$, whereas indices 0 and $n+1$ correspond to fictive tasks representing the opening and closing of the gates. This ILP model requires $\mathcal{O}(n^2 m)$ decision variables (instead of $\mathcal{O}(n)$ for the scheduling one).

For each decision variable $x_{i,j,k}$, we also define a corresponding idle period $I_{i,j,k}$ to take into account the contribution of each pair of successive flights (as well as the opening and closing idle periods) in the cost of a solution (cf. Equation 19):

$$I_{i,j,k} = \begin{cases} f_j^s - f_i^e & \text{if } 1 \leqslant i < j \leqslant n \\ f_j^s - g_k^s & \text{if } i = 0,\ j \leqslant n \\ g_k^e - f_i^e & \text{if } i \geqslant 1,\ j = n+1 \\ g_k^e - g_k^s & \text{if } i = 0,\ j = n+1 \end{cases} \tag{13}$$

2) Constraints: As mentioned in Section IV-B3, there are exactly $n+m$ idle periods (possibly of length 0) in a feasible solution, so:

$$\sum_{\forall x_{i,j,k} \in \mathcal{X}_{\text{ILP}}} x_{i,j,k} = n + m \tag{14}$$

Moreover, either a gate k is empty and $x_{0,n+1,k} = 1$, or its first flight of index $j_0 \leqslant n$ is unique and $x_{0,j_0,k} = 1$, all other $x_{0,j,k}$ being equal to 0, therefore:

$$\forall k \in [1, m], \sum_{\forall x_{0,j,k} \in \mathcal{X}_{\text{ILP}}} x_{0,j,k} = 1 \tag{15}$$

Furthermore, in order to guarantee that only one flight can be assigned immediately before a given flight f_j (Equation 16) or after flight f_i (Equation 17), we add the two following sets of constraints:

$$\forall j \in [1, n], \sum_{\forall x_{i,j,k} \in \mathcal{X}_{\text{ILP}}} x_{i,j,k} = 1 \quad (16)$$

$$\forall i \in [1, n], \sum_{\forall x_{i,j,k} \in \mathcal{X}_{\text{ILP}}} x_{i,j,k} = 1 \quad (17)$$

Eventually, we have to ensure that if a pair of flights f_i and f_j are successively assigned on gate g_k, no successor $f_{j'}$ of f_j can be assigned another gate $g_{k'}$ than g_k:

$$\forall i < j \in [0, n], \forall k \in [1, m],$$
$$x_{i,j,k} + \sum_{\substack{\forall x_{j,j',k'} \in \mathcal{X}_{\text{ILP}} \\ k' \neq k}} x_{j,j',k'} \leqslant 1 \quad (18)$$

These last $\mathcal{O}(n^2m)$ constraints dominate the number of constraints of this model.

3) Cost: Finally, the linear objective function to minimize, which coincides with the cost of the scheduling model defined by Equation 8, can be expressed as the sum of the products of the square of each idle period $I_{i,j,k}$ by the corresponding binary decision variable $x_{i,j,k}$:

$$\text{cost} = \sum_{\forall x_{i,j,k} \in \mathcal{X}_{\text{ILP}}} I^2_{i,j,k} x_{i,j,k} \quad (19)$$

D. Resolution with a MIP Solver

Even if the ILP model of the previous section is much larger than the scheduling one in terms of numbers of variables and constraints, MIP solvers have become such powerful tools for modeling and solving real-world combinatorial optimization problems [18] that optimal solutions could be proved for all instances of our data set, including for terminal F, the busiest one at Paris-CDG airport, with almost 200 flights per day to allocate among 30 gates.

The main resolution algorithm to solve ILP and MIP models, *Branch and Cut* (BC) [19], is based on a *Branch and Bound* algorithm where a *Simplex* algorithm is run at each node of the search tree to solve continuous relaxations of the problem. Cutting planes that exclude non-integral solutions are then added to the model to reduce the search space.

State of the art MIP solvers like Gurobi [20], which we used to obtain the results presented in this paper, significantly improve the efficiency of the previously mentioned basic BC algorithm with preliminary transformation techniques to reduce the size of combinatorial problems, as well as heuristics to obtain better objective bounds during the search. The description of these sophisticated refinements falls beyond the scope of this paper and inquisitive readers may refer to [21] to obtain more information.

Figure 4 shows the distribution of idle time durations in minutes with the initial gate allocation (blue) and the robust gate allocation (orange) for one day of traffic at Paris-CDG, terminal F. The robust distribution is significantly shifted towards larger values, leaving more room for the absorption of potential delays during operations. Table I provides the minimum value, average and standard deviation for those distributions. On this particular instance, the smallest idle time has also been increased from 1 min to 2 min.

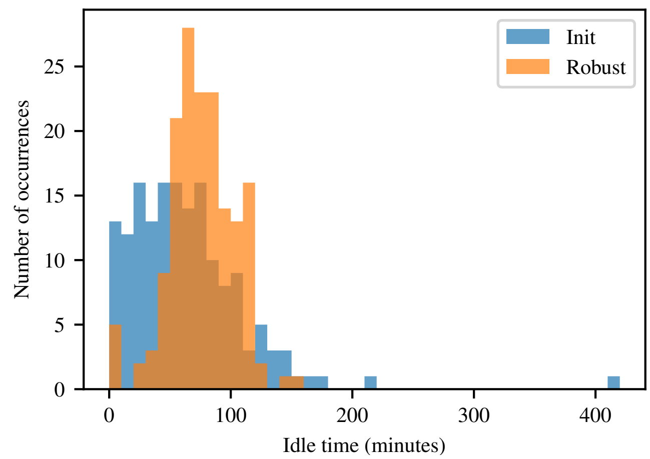

Fig. 4. Distribution of idle time durations for the initial and robust gate allocations.

TABLE I
COMPARISON OF INITIAL AND ROBUST GATE ALLOCATION FOR ONE DAY OF TRAFFIC AT PARIS-CDG, TERMINAL F.

	Idle times (minutes)		
	Minimum	Average	Std Deviation
Init	1	63	48
Robust	2	76	26

V. RESULTS

In this section, we first describe how we extracted the flights and gates data from recorded traffic at Paris-CDG airport, then the simulation of the DMAN on the departures and the resulting additional gate occupancy. Eventually, we present the benefits of a robust gate allocation when departures are sequenced by a DMAN over the initially recorded gates, in terms of number and duration of potential gate conflicts.

A. Data Processing and DMAN Simulation

For this study, we have selected the ten heaviest days of actual traffic recorded at Paris-CDG airport during the whole

TABLE II
NUMBER OF FLIGHTS AND GATES BY TERMINAL (PER DAY ON AVERAGE).

Terminal	Flights	Gates	Flights per Gate
A	49	17	2.9
B	15	10	1.5
C	21	14	1.5
D	83	18	4.6
E	111	23	4.8
F	185	27	6.8
J	76	20	3.8
K	53	19	2.8
L	18	7	2.6
X	27	6	4.5

month of July, 2017, and the ten busiest terminals in terms of average number of flights per gate and per day (see Table II).

For each day of traffic and each turnaround movement on the airport, the following information was gathered:

- aircraft type;
- used gate and corresponding terminal;
- time of arrival at the gate;
- time of departure from the gate.

In some rare cases (probably due to some gate-to-gate movements that were not recorded), one of the two times is missing. The data were completed as follows:

- If the arrival time is missing:
  - If there is not any other flight occupying the same gate before its departure time, the flight is considered at the gate from its opening.
  - Otherwise, the flight is considered at the gate $30\,\mathrm{min}$ before its departure time (which does not create any conflict with our data set).
- In the same way, if the departure time is missing:
  - If there is no flight occupying the same gate after its arrival time, the flight is considered at the gate until its closing.
  - Otherwise, the flight is considered at the gate $30\,\mathrm{min}$ after its arrival time (without generating any conflict).

These data provide the initial gate allocation (called *Init* in the figures) and will be compared to the robust gate allocation (called *Robust* in the figures), obtained as explained in IV, using the Gurobi Commercial Optimizer 8.1.0 [20], without changing the terminal nor the gate occupation period, but only the assigned gate. This robust gate allocation takes into account the compatibility between gates and aircraft types: only the gates that were actually used (at least once) by a given aircraft type are considered compatible with it. Note that the actual gates capabilities could include more aircraft types than the ones we infered from our data set.

By simulating the DMAN process as described in III, we obtain the Target Off-Block Times (TOBT), which are new, potentially delayed, departure times from the gate. The difference between these new off-block times and the initial ones gives the additional gate occupancy due to the DMAN process, and permits to detect the number and duration of gate conflicts that would appear in the considered allocation scheme (initial or robust): we consider that there is a gate conflict when there is less than $2\,\mathrm{min}$ between the arrival time of a flight and the departure of the previous one at the same gate, because there would not be enough time for the push-back of the departing flight.

In order to estimate the effects of a traffic increase on the additional gate occupancy and conflicts induced by the DMAN and the gate allocation scheme, we reproduced the same computations for samples with respectively $3\,\%$ and $5\,\%$ more traffic. To increase the traffic, we have copied some randomly selected turnaround movements, among the ones that can be assigned the same gates with the same duration at a different time without conflict.

TABLE III
ADDITIONAL GATE OCCUPANCY BY TERMINAL (min PER DAY ON AVERAGE).

Terminal	Actual	$3\,\%$	$5\,\%$
A	20.6	23.2	23.6
B	3.5	4.2	4.9
C	8.7	9.4	10.3
D	24.2	25.9	27.5
E	45.6	50.9	53.1
F	52.6	59.8	62.3
J	30.1	30.9	31.8
K	33.6	35.1	37.5
L	10.2	13.8	12.5
X	8.2	8.3	10.2
others	70.4	77.25	77.75

B. Additional Gate Occupancy

Figure 5 shows the total amount of additional gate occupancy due to the DMAN process, in average per day, for actual and increased traffic. The additional gate occupancy vary from $5.1\,\mathrm{h}$ to $5.9\,\mathrm{h}$, according to the rate of the traffic increase, which represents around $0.3\,\%$ of the total gate occupancy per day. These values provide a global measure of the non-negligible effects of the DMAN process.

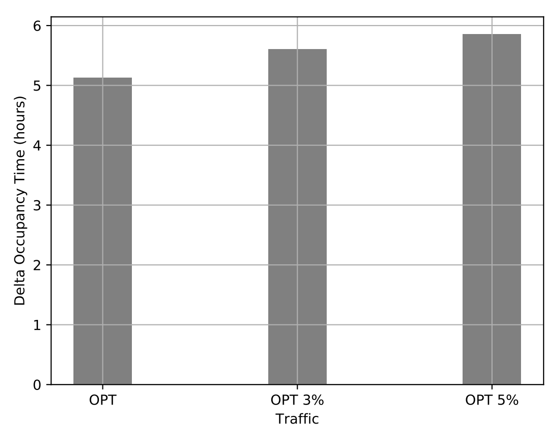

Fig. 5. Additional gate occupancy in hours (average per day) due to DMAN for actual traffic, a $3\,\%$ and a $5\,\%$ increase.

Results can be detailed by terminal (see table III), in order to point out the ones that are the most affected. As an example, terminal F appears to be the busiest one, with an amount of additional gate occupancy around $1\,\mathrm{h}$ per day. The analysis of the data explains this result: more than 180 flights are scheduled on this terminal each day, while it offers only 27 gates. During traffic peaks, this terminal is close to saturation and can cause significant operations disruptions in case of gate conflicts between arrivals and departures.

C. Gate Conflicts Due to Delayed Departures

As explained in Section V-A, we consider that there is a gate conflict between two successive aircraft i and j at the same gate when there is less than $d_{\min} = 2\,\mathrm{min}$ between the departure time of flight i and the arrival time of flight j:

$$f_j^s - f_i^e < d_{\min}$$

The duration of such a gate conflict is then defined as:

$$d_{ij} = d_{\min} - (f_j^s - f_i^e)$$

Figures 6 and 7 give the total number and duration of the gate conflicts that appear when the delayed off-block times are applied in each gate allocation (initial or robust), and for each traffic density (actual, $3\,\%$ and $5\,\%$ increase), during the ten considered days.

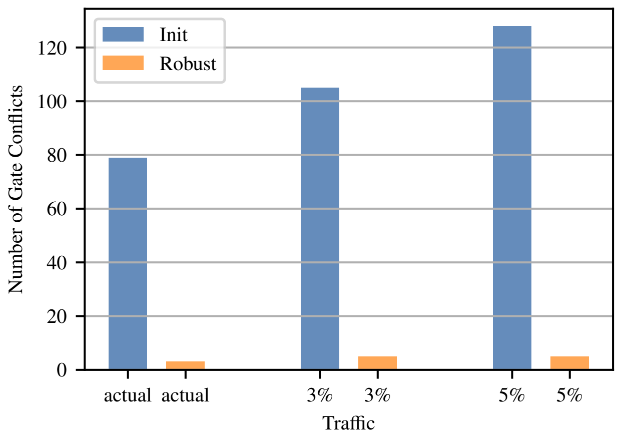

Fig. 6. Number of gate conflicts due to DMAN with the initial and robust gate allocation for actual and increased traffic.

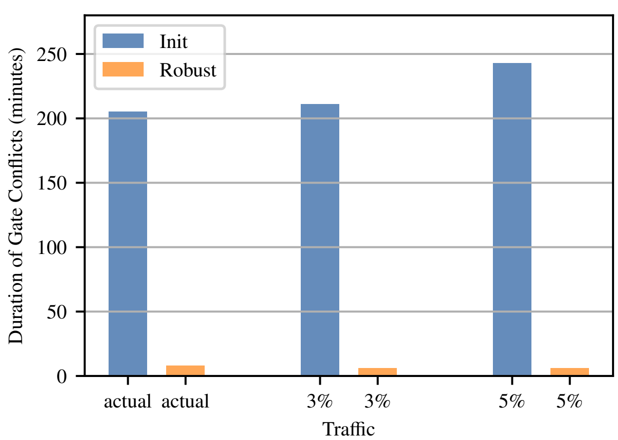

Fig. 7. Total duration of gate conflicts due to DMAN with the initial and robust gate allocation for actual and increased traffic.

The robustness of the second gate allocation method is obvious, with no more than 5 gate conflicts for the $5\,\%$ increase during the whole period, while the initial gate allocation causes from 79 to 128 gate conflicts, depending on the traffic density.

TABLE IV
NUMBER OF GATE CONFLICTS DUE TO THE DMAN.

Terminal	Actual		3 %		5 %	
	Init	Robust	Init	Robust	Init	Robust
A	3	0	10	0	7	0
B	2	0	2	0	3	0
C	1	0	1	0	1	0
D	11	0	19	0	23	0
E	14	0	15	0	17	0
F	22	0	27	0	40	0
J	2	0	3	0	6	0
K	9	0	11	0	11	0
L	6	3	10	5	8	5
X	6	0	7	0	7	0
others	3	0	4	0	5	0

TABLE V
DURATION OF GATE CONFLICTS DUE TO THE DMAN (IN min).

Terminal	Actual		3 %		5 %	
	Init	Robust	Init	Robust	Init	Robust
A	12	0	21	0	9	0
B	8	0	2	0	3	0
C	3	0	1	0	1	0
D	29	0	26	0	34	0
E	33	0	42	0	44	0
F	41	0	39	0	64	0
J	4	0	4	0	11	0
K	24	0	32	0	33	0
L	30	8	32	6	30	6
X	16	0	8	0	9	0
others	5	0	4	0	5	0

The effects on the duration of gate conflicts is even more blatant as there is at least two orders of magnitude between both allocation strategies: only a few minutes for the robust gate allocation, against more than $200\,\mathrm{min}$ for the initial one with actual traffic, growing to $243\,\mathrm{min}$ with the $5\,\%$ increase.

It is quite important to notice that the duration of a gate conflict is a poor indicator of all the operational problems that it can cause in real time: it only represents the time that the arrival flight would have to wait for its gate if no other solution is found. However, such situations involving a holding aircraft can rapidly cause far more trouble to the rest of the traffic in the stand area.

Tables IV and V present the details of the number and the duration of conflicts by terminal. These values shows that terminal F appears almost saturated with the initial gate allocation strategy, as it is the less resilient to departure delays, while no more gate conflict occurs when we use the robust gate allocation, even with the $5\,\%$ increase. This confirms the efficiency of such a gate allocation method, which remains robust even in the case of a noticeable traffic increase and allows to exploit the terminal to their full capacity.

We can also notice that the few conflicts that appear with the robust gate allocation are all located in terminal L. Actually, the reason for these conflicts is quite simple and can be directly found in the data: many flights have a very long gate occupancy in this terminal (almost half a day for some of them), so that there is no possible asiignment that could allow any departure delay or traffic increase with this kind of use scheme.

Conclusion and Further Works

In this article, we first propose a method to estimate the additional gate occupancy expected when an airport implements a Departure Management (DMAN) process with a pre-departure sequencing. These additional gate occupancies are deduced from realistic and optimized runway take-off sequences, involving all the departures on all the runways of the airport at a given period. These results can be analyzed for each terminal, in order to point out the ones that are the most affected by the departure scheduling.

We also show how to compute optimally robust solution for the Gate Allocation Problem (GAP), even for the Paris-Charles-de-Gaulle international airport and its busiest terminals. The robustness of the solution stems from the minimization of the variance of idle time periods which tends to balance them over time and resources. The resulting allocation appears very efficient at Paris-CDG airport, as it remains robust w.r.t. the departure delays that can be expected from the DMAN process, even in case of a $5\,\%$ traffic increase: departures in other terminals can be delayed as needed by the DMAN sequencing, without creating any additional gate conflict and allowing to operate the terminals at their full capacity.

Further works could confirm and enrich these promising results at Paris-CDG airport, using fast time simulations that reproduce the whole taxiing phase of the flights. Under different surface management hypotheses, the robustness of the gate allocation could be tested in more realistic and difficult conditions, including uncertainties on departure times, landing times, taxiing speeds and the resolution of conflicts between taxiing aircraft. The same approach could also be applied to other airports that are planning to implement a Departure Management process and wish to estimate their needs in terms of terminals and gates development, in the scope of an A-CDM certification.

References

[1] EUROCONTROL, *Airport CDM Implementation Manual*. EUROCONTROL Airport CDM Team, 2017.

[2] D. Huet and S. Pickup, "Local and network impact assessment of airport collaborative decision making (a-cdm)," in *12th USA/Europe Air Traffic Management Research and Developpment Seminar*, 2017.

[3] A. B. Fernandes, P. J. Smith, K. Weaver, K. Durham, M. Evans, and D. Johnson, "Identifying support requirements for airport departure management," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 56, no. 1, pp. 65–69, 2012. [Online]. Available: https://doi.org/10.1177/1071181312561034

[4] I. Simaiakis, "Analysis, modeling and control of the airport departure process," Ph.D. dissertation, Massachusetts Institute of Technology, 2013.

[5] S. R. J. A. D. Atkin, E. K. Burke, "A more realistic approach for airport ground movement optimisation with stand holding," in *Multidisciplinary International Scheduling Conference (MISTA)*, 2011.

[6] R. Deau, J.-B. Gotteland, and N. Durand, "Airport surface management and runways scheduling," in *8th USA/Europe Air Traffic Management Research and Developpment Seminar*, 2009.

[7] S. H. Kim, "Airport control through intelligent gate assignment," Ph.D. dissertation, Georgia Institute of Technology, 2013.

[8] A. Bouras, M. A. Ghaleb, U. S. Suryahatmaja, and A. M. Salem, "The airport gate assignment problem: A survey," *The Scientific World Journal*, 2014.

[9] U. I. Gupta, D. T. Lee, and J. Y. Leung, "Efficient algorithms for interval graphs and circular-arc graphs," *Networks*, vol. 12, no. 4, pp. 459–467, 1982.

[10] M. Biró, M. Hujter, and Z. Tuza, "Precoloring extension. I. Interval graphs," *Discrete Mathematics*, vol. 100, no. 1, pp. 267 – 279, 1992.

[11] L. G. Kroon, A. Sen, H. Deng, and A. Roy, "The optimal cost chromatic partition problem for trees and interval graphs," in *Graph-Theoretic Concepts in Computer Science*. Springer, 1997, pp. 279–292.

[12] J. Guépet, R. Acuna-Agost, O. Briant, and J. Gayon, "Exact and heuristic approaches to the airport stand allocation problem," *European Journal of Operational Research*, vol. 246, no. 2, pp. 597 – 608, 2015.

[13] A. Bolat, "Procedures for providing robust gate assignments for arriving aircrafts," *European Journal of Operational Research*, vol. 120, no. 1, pp. 63–80, 2000.

[14] ——, "Models and a genetic algorithm for static aircraft-gate assignment problem," *Journal of the Operational Research Society*, vol. 52, no. 10, pp. 1107–1120, Oct 2001.

[15] H. Simonis, "Models for global constraint applications," *Constraints*, vol. 12, no. 1, pp. 63–92, March 2007.

[16] R. Wang and N. Barnier, "Propagation of idle times costs for fixed job scheduling," in *2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI)*, Nov 2018, pp. 718–725.

[17] E. M. Arkin and E. B. Silverberg, "Scheduling jobs with fixed start and end times," *Discrete Applied Mathematics*, vol. 18, no. 1, pp. 1 – 8, 1987.

[18] M. Jünger, T. M. Liebling, D. Naddef, G. L. Nemhauser, W. R. Pulleyblank, G. Reinelt, G. Rinaldi, and L. A. Wolsey, *50 Years of integer programming 1958-2008: From the early years to the state-of-the-art*. Springer Science & Business Media, 2009.

[19] J. E. Mitchell, "Branch-and-cut algorithms for combinatorial optimization problems," *Handbook of applied optimization*, pp. 65–77, 2002.

[20] L. Gurobi Optimization, "Gurobi optimizer reference manual," 2018. [Online]. Available: http://www.gurobi.com

[21] T. Achterberg, R. E. Bixby, Z. Gu, E. Rothberg, and D. Weninger, "Multi-row presolve reductions in mixed integer programming," in *Proceedings of the Twenty-Sixth RAMP Symposium*, Tokyo, Oct. 2014.

Authors

Ruixin Wang is a PhD student at École Nationale de l'Aviation Civile (ENAC) working on the cooperation between combinatorial optimization solvers. He graduated from ENAC and Civil Aviation University of China in 2016.

Cyril Allignol is an assistant professor at ENAC. He graduated from ENAC as an engineer in 2006, and received a PhD (2011) in Computer Science from the University of Toulouse.

Nicolas Barnier is an assistant professor at ENAC. He graduated from ENAC as an engineer in 1997, and received a PhD in Computer Science in 2002 from the University of Toulouse. He is one of the authors of FaCiLe, an open source Constraint Programming library for the functional language OCaml.

Jean-Baptiste Gotteland is an assistant professor at ENAC. He graduated from ENAC as an engineer in 1995, and received a PhD (2004) in Computer Science from the University of Toulouse.