Runways sequences and ground traffic optimisation

Raphael Deau Jean-Baptiste Gotteland Nicolas Durand
Direction des Techniques et de l'Innovation
7, Av. Edouard Belin - BP4005
F31055 Toulouse Cedex 4
Email: gotteland@tls.cena.fr
Telephone: (+33) 5 62 17 41 68
Fax: (+33) 5 62 25 95 99

***Abstract*—At the airport level, the new systems involved in the A-SMGCS (Advanced Surface Movement Guidance and Control System) give the possibility to take advantage of some innovative decision support tools bound to the optimisation of the ground traffic management.**

In this article, two different tasks assumed by airport controllers are analysed and modeled: the runway sequencing process and the application of runways sequences at the ground level. An existing ground traffic simulator is adapted to measure the potential improvements that could be expected by the use of some optimisation methods applied on these two modeled problems.

I. Introduction

Airport congestion is still a key point to be studied for the next years: the evolutions that are expected concerning the future management of the ground traffic situations (A-SMGCS: Advanced Surface Movement Guidance and Control System) at the airport control level have obviously several environmental and economical issues.

On major airports, these evolutions can be technically provided by taking advantage of some new developed systems such as the surface radars, the D-GPS (Differential Global Positioning System) associated with the ADS-B (Automatic Dependant Surveillance mode B) and numerous other coordination tools like AMAN (Arrival MANager) and DMAN (Departure MANager).

In this context, this article focuses on the possible optimisation of two major aspects of the airport controllers tasks: the aircraft sequencing at the runway level and the conflict resolution between taxiing aircraft. An airport simulation tool (ATOS: Airport Traffic Optimisation Simulator) is adapted and used to measure the delay reduction that could be expected.

II. Related work

A. Projects

Two main approaches called AMAN (Arrival MANager) and DMAN (Departure MANager) deal with the aircraft sequencing problem:

- AMAN are decision support tools that provide the controllers with information on arrival flows, including calculated arrival runways sequences. These informations are regularly updated with the actual positions of aircraft in the approach sectors.
- DMAN are planning tools developed to improve the departure flows at airports by optimizing the departure runways throughput.

Some projects as PHARE (Program for Harmonised ATM Research in Eurocontrol) [1] have defined some DMAN and/or AMAN systems but do not coordinate both of them at the airport ground traffic level. PHARE was a project instaured by Eurocontrol. It was a collaborative research program that investigates an air traffic management concept.

Gate to Gate [2] is a European project which takes into account aircraft from their departure gates to their arrival gates. It mostly improves an AMAN project by managing the air traffic problem. A DMAN is used for the mixed runways and shares out the main informations about the departures. The AMAN has to set the arrivals with the information provided by the DMAN.

The NLR develops a concept which schedules the aircraft on the airport by using constraint relaxation [3]. A lot of parameters are considered, as the runway separations, SID routes, exit points, . . . DMAN and AMAN are not explicitly described but implicitly defined.

OPS [4] defines a new DMAN to schedule departures. This project is based on more human interactions (from pilots and controllers for example) but is also more flexible (the users can setup a lot of values). They look forward to define a real-time decision support tool.

Total Airport Management (TAM) [5] tries to merge together as many concepts as possible defined by Eurocontrol, concerning AMAN, DMAN in the ATC in order to optimise the airport capacity and improve the predictability of airports traffic.

CADM (Coordinated Arrival Departure Management) [6] is a concept that mixes a DMAN and an AMAN but does not consider precisely the taxiing times. It uses fuzzy inference mechanism to determine rules to use to set the aircraft sequence.

All these projects focus on the definition and/or the prediction of airports runways sequences for arrivals and/or departures, trying to share as efficiently as possible all the available informations given by the approach sectors and the airport systems. However, the taxiing phases of the flights are still the ones that are the most difficult to predict with a good accuracy: the tested DMAN systems are still not so satisfying

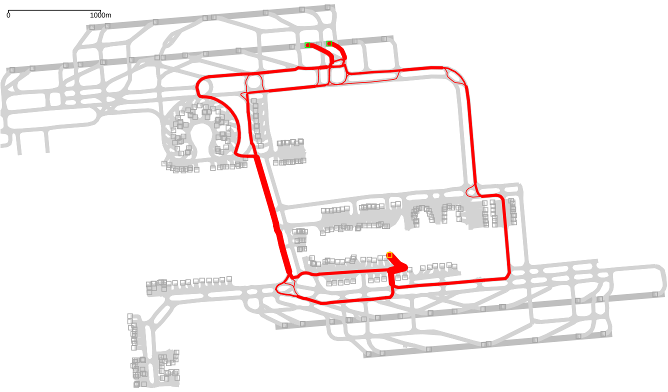

Fig. 1. Roissy map

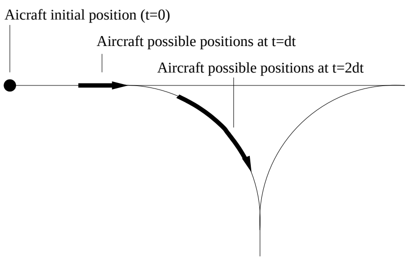

Fig. 2. Speed uncertainties

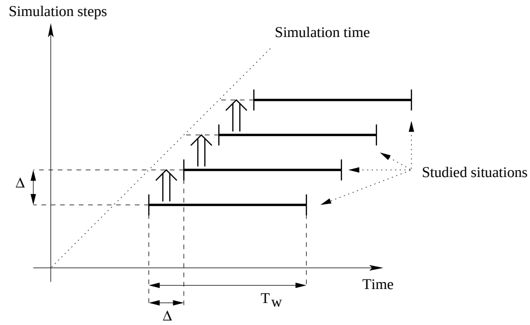

Fig. 3. Shifted windows

for ground controllers, as the predicted informations remain uncertain.

B. Background

Previous publications [7], [8], [9], [10] studied the ground traffic optimisation on Roissy Charles De Gaulle and Orly airports. The ATOS (Airport Traffic Optimisation Simulator) simulator was developed and used to compare the efficiencies of several optimisation methods applied to different traffic situations.

The simulator uses a detailed description of the airport taxiways, gates, push-backs, runways and the existing constraints (one-way taxiways for example) to calculate a set of possible paths for each aircraft (see figure 1). The whole traffic is simulated using the real airport flight plans demand of a day of traffic. The flight plans contain information such as the aircraft type, the gate position, the landing or take-off time, the runway used …

Using these informations, and for each traffic situation, the possible paths for each aircraft are calculated on a defined time window T_w, taking into account uncertainties on taxiing speeds and a trajectory prediction done: in such a prediction, the future aircraft position is not a point but a set of possible points (a line segment) on the taxiways used (see figure 2).

The problem to solve consists in assigning a path to each aircraft, with holding points if necessary, in order to solve every *conflict* with other aircraft within the time window T_w. Two aircraft are in *conflict* each time the separation standards defined by the operating rules of the airport are violated.

These rules are modelled as follows:

- parked aircraft are supposed to be conflict free;
- a minimum separation distance is required between each taxiing aircraft pair;
- time separations are required between aircraft on runways, depending on the aircraft types and their wake turbulence categories.

Among all the possible solutions, the best conflict free trajectory is search according to a global criterion taking into account the delay due to holding points and longer paths chosen.

The problem is very combinatorial because of the number of possible paths and holding points for each aircraft. When the number of aircraft involved increases, the problem can become very difficult to solve using exact methods. Different optimisation methods have been studied by the DSNA/DTI R&D POM team to solve it:

- A sequential deterministic approach consists in first ordering aircraft, give the shortest path to the aircraft with the highest priority, and optimise the $n+1^{th}$ trajectory solving conflicts with the n previous already optimised trajectories. Optimising the trajectory of one aircraft, solving conflicts with n other known trajectories is quite simple and can be done with an A^* or branch & bound method [HT95], [11].
- A global approach using stochastic optimisation based on genetic algorithms [12], [13] can be used to find the best paths combination. To increase the efficiency of the algorithm, the partial separability of the problem can be used to define *crossover* and *mutation* operators able to optimally recombine current solutions during the convergence process [14].

When a traffic situation is solved at time t, the paths obtained are applied to the moving aircraft during a time Δ ($\Delta < T_w$) called time shift window, to create the updated situation at time $t+\Delta$ (figure 3). A whole day of traffic can thus be simulated using this shifting window modeling, taking into account the uncertainties on aircraft taxiing speeds.

C. Obtained Results

The first simulations results obtained with this modeling have shown some characteristics on the traffic and can help quantifying some parameters of the problem:

- When ”realistic” uncertainties on taxiing speeds (from 20% to 50%) are used, the time window must be reduced to 10 minutes to be able to solve the problem. This emphasis the difficulties encountered on the DMAN concept because with a 10 minutes advance notice, the final take-off time for an aircraft on the runway is to lately known.
- The method using a genetic algorithm consider globally each traffic situation (without classifying aircraft with priority orders) reduces the delays from 1 to 2 minutes per aircraft during peak hours at Roissy Charles De Gaulle Airport, which shows that the possible time saved by optimising the taxiing phase of a flight is quite significant on such airports.
- During peak periods, knowing precisely the paths followed by each aircraft is necessary to manage correctly the CFMU[1] slots: an optimised simulation can help knowing the delay due to congestion and allows to anticipate departures from gates if necessary.

D. Conclusions

These last results show that on large airports such as Roissy Charles De Gaulle, the performance of systems such as AMAN and DMAN depends on an optimised taxiing management. On the one hand, the calculation of runway sequences must take into account the taxiway paths and holding points given to aircraft. On the other hand, it might be better to take into account the full runway sequence to optimise the gate time departures and the aircraft holding points and paths.

As it was shown that the acceptable time window adapted to realistic uncertainties is less than 10 minutes, it is proposed in this article to split the resolution process of each traffic situation in two steps:

- First, the best runways sequences compatible with the current aircraft positions and the known arrival flows should be computed, with a large anticipation time (about 30 minutes if possible). This point is described in part III.
- Then (part IV), the aircraft paths and holding points should be optimised in order to fit as close as possible to these targeted runways sequences, with an adapted anticipation time (less than 10 minutes)

III. Runways sequences optimisation

A. Goals

This part, focuses on finding some optimal runway sequences, respecting a given traffic situation and considering the arrival flows.

A system merging the AMAN and the DMAN informations at the airport level is defined: on runways shared by arrivals and departures, this is the only way to optimise correctly the sequence, and as far as arrivals and departures have to share the same airport infrastructure, they have to be managed together at the ground level.

[1]Central Flow Management Unit

B. Problem modeling

To meet these goals, the problem is defined as follows:

- The variables of the problem are the slots that must be assigned to each aircraft;
- The main constraints are the landing times, the minimal remaining taxiing time of each departure, the runway separation rules and the CFMU[2] slots allocated to some departing aircraft;
- The criterion to minimise measures the departures delays and the deviations from the CFMU slots;
- The prediction time should be close to 30 minutes.

1) Constraints: At the airport level, the arrival sequence cannot be substantially modified: the ordering of arrivals is fixed by approach sectors and it is reasonable to consider that each arriving aircraft cannot be delayed more than a reduced time λ ($\lambda < 1$ minute) if these kind of decisions can be taken in advance enough.

Concerning the departures sequences, the minimum possible taxiing time to the runway becomes a constraint for aircraft leaving parking positions: to obtain a feasible sequence, each aircraft's shortest time to runway is considered as a constraint in the sequence search.

The most important constraints used in the sequence optimisation is the separation due to the wake turbulence. The minimum time between two aircraft depends on their weight category. For example: a ”low weighted” aircraft cannot take off less than 180 seconds after a ”heavy weighted” aircraft has taken off. Three categories of aircraft (and associated wake turbulence) are defined: ”low”, ”medium” and ”heavy”. The following table shows each separation time (in seconds):

1st acft $\rightarrow$	A. L	A. M	A. H	D. L	D. M	D. H
A. L	60	120	180	60	120	180
A. M	60	60	90	60	60	120
A. H	60	60	90	60	60	90
D. L	60	120	180	60	120	180
D. M	60	60	60	60	60	120
D. H	60	60	90	60	60	90

L = Low, M = Medium, H = Heavy
A = Arrival, D = Departure

Other constraints concern aircraft which are assigned some CFMU slots: these aircraft have to be inserted in the sequence between arrivals and classical departures, respecting their fixed CFMU slots. According to the CFMU official acceptance, a CFMU slot is respected if the aircraft takes off from 5 minutes before to 10 minutes after the schedule. However, the actual off-gate times does not always allow to respect all the CFMU slots (as for example when a departure leaves the gate later than its assigned slot). In consequence, the only constraint that is defined as such is the interdiction to take off more than 5 minutes before the slot and the other CFMU requirements will be integrated in the criterion.

[2]CFMU: Central Flow Management Unit

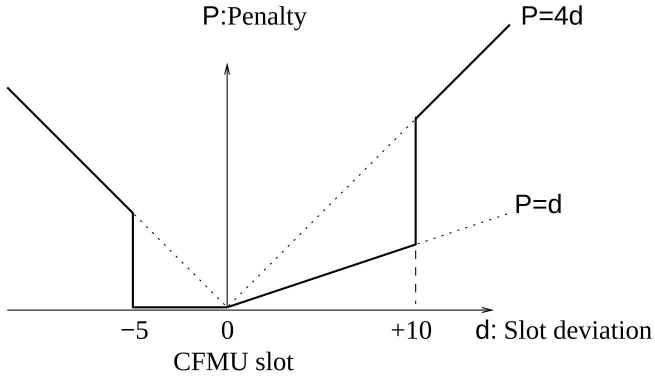

Fig. 4. Criterion for CFMU slots

2) Criterion: In the estimated sequence, the penalties relative to the deviation from each CFMU slot (see fig. 4) and the delay regarding the minimal runway access time for each other departures are computed. The aim of the optimisation is to minimise the sum of those values: the more the CFMU slots are respected and the shorter is the time spent by aircraft on taxiways, the better the sequence is evaluated.

On the other hand, arrivals sequencing delays are not taken into account in the criterion as these little delays (less then λ seconds for each arrival) are not penalising. Of course, sequences which would cause one landing to be delayed more than λ seconds are considered not valid regarding the constraints that are defined and cannot be accepted.

3) Prediction time: The sequence optimisation takes into account all the aircraft that may appear in the sequence during the next T_s minutes. The choice of this prediction time is influenced by several factors: as the sequence optimisation problem is very combinatorial (for n aircraft, the complexity of this scheduling problem is proportional to $n!$), the prediction time should be short enough to keep the problem size small enough. On the other hand, it seems logical that the larger the prediction is, the better the sequence will be optimised on the whole day.

C. Resolution

The sequence optimisation problem is a classical scheduling problem that can be solved with deterministic Constraint Satisfaction Problem algorithms.

1) Problem modeling: The problem is to find the best sequencing for a given list of aircraft considering the almost fixed arrivals and the separation time between two aircraft. To find the best solution, each permutation of the aircraft list has to be explored.

2) Branch & bound algorithm: To solve this problem, a classical branch & bound algorithm is used: each branch of the sequences tree (see fig. 5) is potentially explored and at each node of the tree, the delay generated by the already assigned slots is calculated. Once a solution has been found (i.e.: when one of the leafs of the tree is joined), the cumulated delay obtained updates the "best current solution found" for the next steps: when a node generates a higher delay than this "best current solution found", the branch is cut.

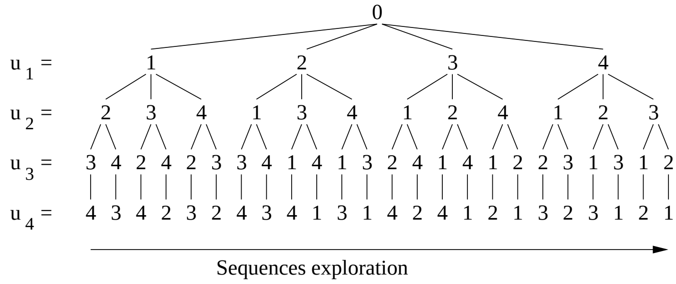

Fig. 5. Sequences tree

This algorithm can be summarised as follows:

Mainloop: For each non yet inserted aircraft 'a':

- Insert 'a' in the current (partial) sequence
- Calculate the new resulting penalty
- If this penalty is acceptable then:
  - If some aircraft still remain to insert, explore the next node of this branch (call back the **Mainloop**)
  - Otherwise, mark the current sequence as the best solution found
- remove 'a' from the sequence

To obtain the best cuts in the tree exploration, the initial order plays a major role. It was observed that defining the initial order in accordance to the "ideal" time on the runway (i.e: the minimal runway access time for non-CFMU departures and the CFMU slots for the others) was a good strategy.

Moreover, some other cuts must be implemented to minimise the number of explorated nodes. These cuts are relative to some specific characteristics of the problem:

- There is obviously no need to explore a branch in which some arrivals are not in the right order.
- There is no need to explore a branch which swaps two equivalent aircraft in the sequence. The equivalence between aircraft is defined according to their wake turbulence category, their type (arrival or departure) and their CFMU profile(with or without a CFMU slot).

D. Results

In order to measure the efficiency of the proposed runway sequence optimisation method, ground traffic simulations were carried out with a traffic sample relative to a heavy day at Roissy Charles De Gaulle, when the fourth runway was not yet in operation (this period presents the advantage to provide a mixed runway 09-27 shared by both departures and arrivals).

At each simulation step (every $\Delta = 2$ minutes), an optimal sequence is computed for each runway, with different anticipation times (from 20 minutes to 50 minutes), and the resulting theoretical delay for departures in these optimal sequences are recorded (this theoretical delay is measured by the difference

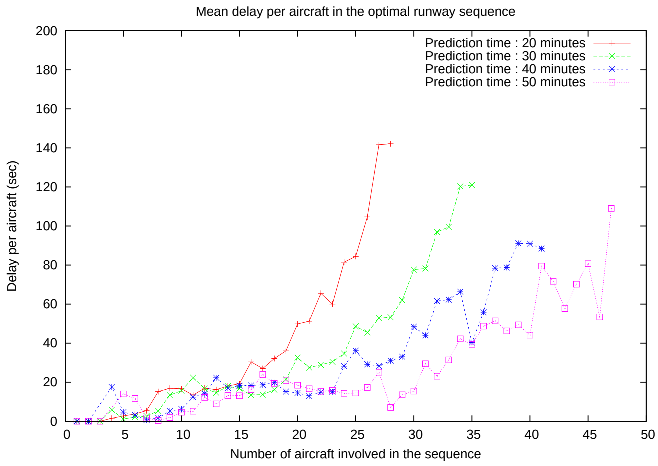

Fig. 6. Mixed runway 27 at Roissy

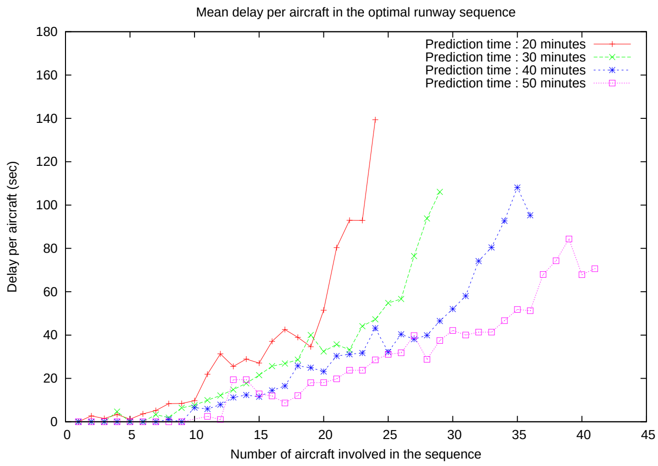

Fig. 7. Runway 26R at Roissy

between the proposed slots in the sequence and the "ideal" slots for aircraft.)

The figures 6 and 7 give the mean delay per aircraft as a function of the number of aircraft involved in the runway sequence, for the two runways used for departures: runway 27 (shared with arrivals) and runway 26R (only used for departures.)

On these figures, the influence of the prediction time window on the quality of the runway sequence can be observed and confirms the expected conclusion: the highest the prediction time is, the best the runways sequences are. Moreover, this relation can be quantified: when the prediction time grows from 20 minutes to 50 minutes, the mean departures delay decreases from 140 seconds to less than 100 seconds in heavy traffic situations.

Of course these delays measures are theoretical and could only be realised if the calculated runway sequences were exactly applied. This is the subject of the next part, in which the retained prediction time window that is considered for the runway sequences is 30 minutes, as it seems to be the best compromise between what can be expected from AMAN and DMAN systems and what can be treated by the optimisation process.

IV. Application to the Ground Traffic Situations

A. Goals

In this part, the runway sequences previously optimised are considered as a target for the ground traffic solver: the objective is to fit as close as possible to the predefined sequences while solving aircraft conflicts on taxiways and gates.

Many options can be studied in this framework: it can be interesting, for example, to favor the earliest departures of each runway sequence and to delay in priority aircraft which might arrive on the runway in advance according to their allocated slot. It can also be interesting to strategically sequence departures before they leave their gate position, by assigning them an initial delay.

These different concepts will be studied and compared to the results obtained without the runway sequence optimisation in the last part of this article.

Therefore, the different optimisation methods must be adapted to take into account the new objective (i.e. fit to the optimised runways sequences) and not only minimise the aircraft delay: this part details the modification that were done for each resolution method.

B. Sequential resolution method

The sequential resolution method deals with a simplified problem, in which aircraft are initially sorted and then considered one after an other (first considered aircraft have priority on last considered ones). As a consequence, this resolution method can be easily adapted to fit some given runways sequences, as these sequences will directly provide the aircraft classification to be considered: each aircraft a can be associated with its slot t_a in the concerned runway sequence, and the sequential method will be applied in the order given by (t_a).

In most of the cases, this process ensures that on each runway, departures always take off in the order defined by the runway sequence (except in some very particular cases relative to the limited aspect of the prediction time window.) However, on runways shared both by departures and arrivals, the order between arrivals and departures is not ensured: if a departure planned just before an arrival reaches the runway area too late, it will be forced to take off after the arrival (in this case, the resolution method has to modify the aircraft classification to find a conflict free solution).

Another point has to be considered, concerning the departures constrained by CFMU slots: generally, the CFMU slots correspond to some delayed take-off times. When such aircraft are involved in a runway sequence, the classification of departures is not the only factor to assume: the exact take-off time of concerned aircraft must also precisely correspond to their given CFMU slots, in agreement with the official CFMU acceptance (no more than five minutes before the slot, nor ten minutes after). During low traffic periods, these departures could take off much earlier while respecting the runway sequence order. For these reasons, a minimal departure time (from the gate) has to be assigned to aircraft [9] and the resulting delay must be propagated over the following departures, in order to keep a consistent sequencing of departures.

Therefore, an *initial wait* w_d is calculated for each departure d, as a function of:

- its minimal runway access time $t_{\min d}$,
- its optional CFMU slot t_{cd},
- the official acceptance for CFMU slots ($\delta_c = 5$ minutes),
- and the initial required wait w_p of the prior aircraft p in the sequence (all aircraft are sorted in the order given by the sequence).

For the first aircraft d_0:

- If d_0 has a CFMU slot t_{cd_0}:

$$w_{d_0} = \max\{0,\ t_{cd_0} - \delta_c - t_{\min d_0}\}$$

- Otherwise,

$$w_{d_0} = 0$$

For the following departures d_i ($i > 0$):

- If d_i has a CFMU slot t_{cd_i}:

$$w_{d_i} = \max\{0, t_{cd_i} - \delta_c - t_{\min d_i}, w_{d_{i-1}} + t_{\min d_{i-1}} - t_{\min d_i}\}$$

- Otherwise,

$$w_{d_i} = \max\{0,\ w_{d_{i-1}} + t_{\min d_{i-1}} - t_{\min d_i}\}$$

The best trajectory for a departure d_i (looked forward by the sequential branch & bound algorithm) is the one corresponding to the shortest path allowing a delay as near as possible to the requested wait d_i.

C. Genetic algorithm solver

The conflict resolution method based on a genetic algorithm consider globally each traffic situation, without assuming any classification between aircraft: thus, the logical way to adapt this method to the new problem (which is the application of some predefined runways sequences) consist in modifying the global criterion to minimise: in this way, the predefined sequences will be considered as a goal but not as a constraint, which is necessary to ensure that some acceptable solutions still exist, even when the traffic situation does not allow to carry out the targeted sequences.

Concerning acceptable (conflict free) solutions, the global criterion is defined as the sum of each specific criterion relative to each aircraft: for a departure, this specific criterion must be refined, in order to estimate the difference between the take-off time that would result from the proposed solution and the take-off time targeted in the optimal runway sequence.

Different definitions of such a criterion can be considered. The main difficulty is obviously relative to the difference between the anticipation time used for the ground conflicts resolution and the one used to compute the runways sequences: on large airports such as Roissy Charles De Gaulle, departures taxiing times can easily exceed the prediction time window, so that the take-off times that will result from the proposed paths and holding positions are uncertain on the long range.

Moreover, looking forward to hold the departures when their positions seem to be in advance compared to their targeted take-off slots is not appropriate, as such a ground traffic management would clearly risk to propagate every form of ground delay to the whole airport.

As a consequence, the proposed criterion is still proportional to aircraft delay, but a balance is applied, in order to penalise more the delay of an aircraft when its minimal runway access time becomes closer to its targeted slot.

With this kind of criterion (based on delay), the same treatment as before has to be considered concerning the management of the CFMU slots (see IV-B): an initial wait w_d has to be computed for each departure d, in order to ensure the correct insertion of these particular departures in the rest of the traffic. Obviously, this initial wait also affects the definition of the criterion.

Finally, the penalty $P(a)$ to be minimised for each aircraft a is estimated as a function of the delay dl_a of the aircraft (including assigned wait and/or path lengthening), the minimal runway access time $t_{\min a}$ and the targeted slot t_a of the aircraft, as follows:

- For an arrival a:

$$P(a) = dl_a \ \ (\texttt{unchanged})$$

- For a departure d with a CFMU slot t_{cd} which is late ($t_{\min d} > t_{cd} + \delta_c$):

$$P(d) = 20 * (dl_d + t_{min_d} - t_{cd})$$

- For a departure d with a CFMU slot t_{cd} which is in advance ($t_{\min d} < t_{cd} - \delta_c$) and which required wait is w_d:

$$P(d) = 10 * (|dl_d - w_d| + t_{cd} - t_{\min d})$$

- For each other departure d which required wait is w_d:

$$P(d) = 5 * (|dl_d - w_d| + \max(0,\ t_{\min d} - t_d))$$

In these definitions, the balance that is applied to departures delay is defined in order to favor as often as possible departures against arrivals.

V. Results

To measure the efficiency of the proposed optimisation methods, four simulations are carried out (with the same traffic sample as in III-D): the two different ground traffic solvers (i.e. the sequential method and the genetic algorithm solver) are tested on two scenarios: in the first one, there is no runway sequence to target, while in the second one, the optimal runways sequences are computed and targeted as explained before.

These four simulations are compared by the generated delay, the slots deviations for concerned departures, and the differences between the targeted slots and the final ones.

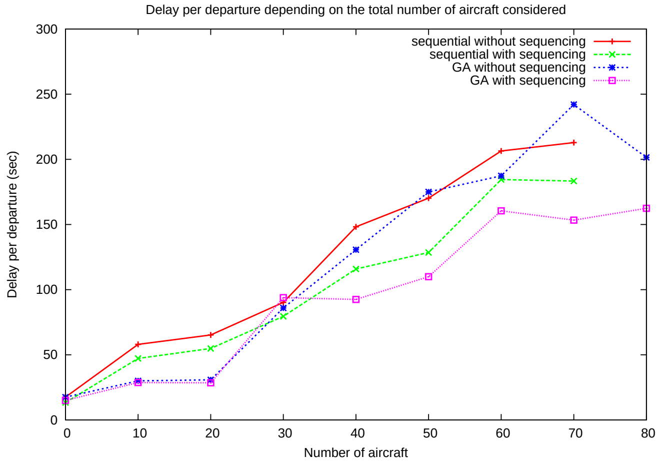

Fig. 8. Mean delay per departure (seconds)

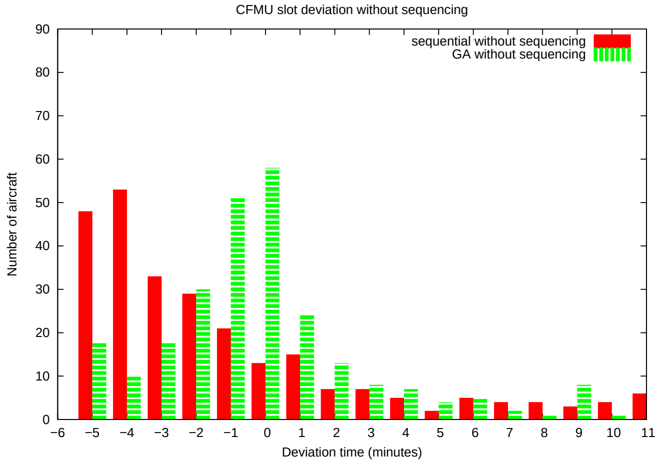

Fig. 10. CFMU deviations without sequencing

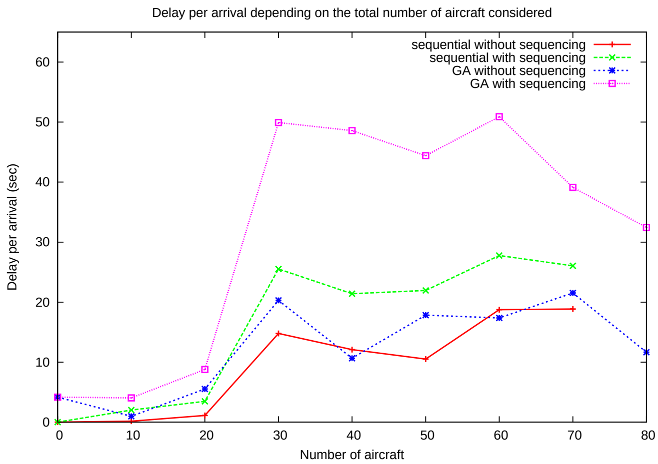

Fig. 9. Mean delay per arrival (seconds)

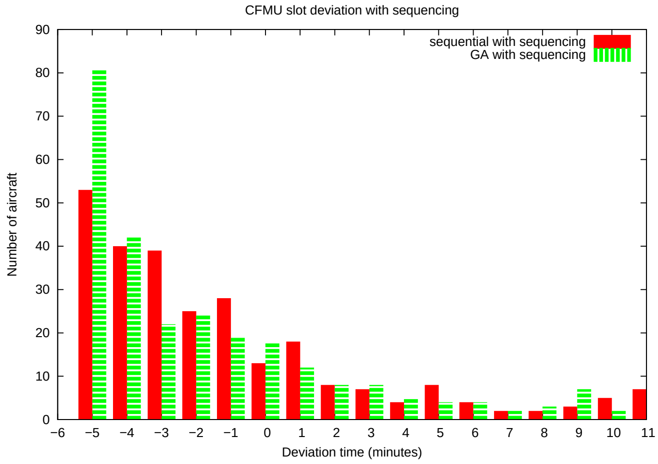

Fig. 11. CFMU deviations with sequencing

A. Delay comparisons

The following table gives the global results of each simulation concerning the delay of "classical" departures (i.e. the departures that are not constrained by a CFMU slot) and arrivals:

		Aircraft delay	
		Without	With sequencing
Sequential method	Dep.	16h42 2min.10'/acft	13h43 1min.45'/acft
	Arr.	1h56 10'/acft	3h12 16'/acft
GA	Dep.	14h46 1min.55'/acft	11h26 1min.25'/acft
	Arr.	2h12 11'/acft	5h33 28'/acft

Aircraft total and mean delay

These delays can also be measured as a function of the number of taxiing aircraft on the airport, as shown on figure 8: the mean delay is calculated for each period of 10 minutes of the day and is put in relation with the corresponding number of taxiing aircraft in this period.

As one can see, the runway sequencing optimisation process enhances the departures results of both solvers, and the sequential solver becomes almost as efficient as the genetic algorithm one. Globally, the difference between a "basic" management of taxiing aircraft (i.e. the sequential method without runway sequencing) and the final genetic algorithm solver is really significant: 45 seconds per aircraft are saved on the whole day, and more than 1 minute par aircraft can be saved during traffic peaks.

Of course, the arrivals delay shown on figure 9 follow an opposite progression, especially with the final genetic algorithm solver (for which the defined criterion voluntarily give priority to departures). In an operational point of view, the 20 seconds of delay added per arrival in compensation to departures management enhancement should be profitable.

B. Deviations to CFMU slots

The figures 10 and 11 shows the distribution of the CFMU slots deviations observed for the concerned departures.

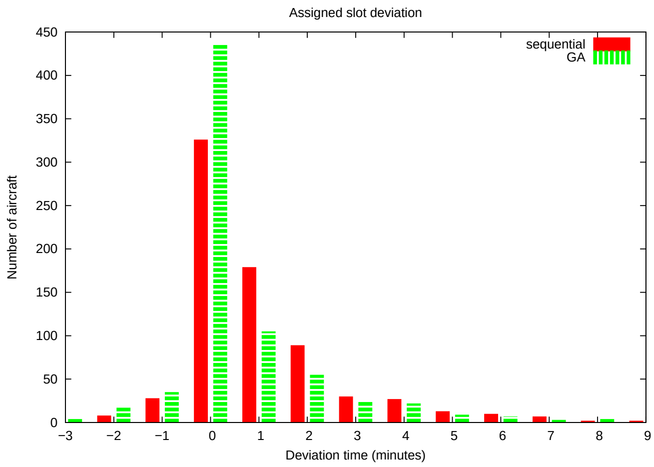

Fig. 12. Deviations to targeted slots (minutes)

These results show that the CFMU slots are quite respected (in fact, the only exceptions concern departures that are ready to leave their gate too late to catch their slot, in accordance with the CFMU requirements).

As no penalty was defined from 5 to 0 minutes before the CFMU slot, the major slots deviations observed are globally spread in this time period. As a consequence of the criterion defined for the genetic algorithm solver, the runway sequencing process allows to concentrate the departures take-off times at the beginning of the allowed period.

C. Deviations to targeted slots

The difference between the calculated runway sequences and the final ones is measured: the figure 12 shows the number of aircraft concerned by each value of slot deviation formulated in minutes.

This figure shows that the runway sequences generated with the genetic algorithm solver are closer to the targeted ones, but there are still a lot of aircraft that don't succeed in catching correctly their initial assigned slot. This result shows that the management of the taxiing aircraft is a very critical step and should really take advantage of some appropriate optimisation methods, rather than only be defined with some basic circulation rules.

VI. Conclusions and Further Work

In the first part of this article, a classical (and exact) method has been implemented and tested to compute some optimal runways sequences at Roissy Charles De Gaulle airport, taking into account each traffic situation with precision, adding the expected arrival flow fixed by approach sectors to the actual ground positions of all taxiing aircraft. The simulation results show that the runways sequences can be largely enhanced if they are well organised with a sufficient anticipation time (the largest possible anticipation time greater than 30 minutes). As a consequence, the delay (and then the taxiing times) assigned to departures at the airport level could be significantly reduced if the surface management of the airport allows to perform these targeted optimal runways sequences.

The second part of this article explores and proposes some concepts that should be developed at the airport control level to manage correctly the taxiing aircraft, while targeting the computed runways sequences. This concepts can decrease significantly the departures delay, but the simulations carried out also confirm that this task is very complex during traffic peaks, as far as the speed of the taxiing aircraft is not precisely known and as far as the conflicts that have to be solved between these aircraft affect the feasibility of the runways sequences. The ground management of aircraft becomes again more complex for runways shared by both departures and arrivals, as departure delays can sometimes totally change the optimal sequence to target.

Further work will consists in refining the way to target a runway sequence at the airport level, by considering for example some new methods to enhance the estimation of the appropriate decisions that must be taken concerning the taxiing aircraft, in order to keep the ground traffic situations consistent with the targeted runways sequence.

Another development will concern the generalisation of the constraints defined to perform the runways sequences optimisation, taking into account more operational issues, like the sequencing rules used by the airport controllers relative to aircraft SID (Standard Instrument Departure) and STAR (Standard Terminal Arrival Route).

References

[1] Eurocontrol *PHARE presentation* http://www.eurocontrol.int/
[2] Eurocontrol *Gate to Gate project* http://www.eurocontrol.int/ 2004
[3] P. van Leeuwen and N. van Hanxleden Houwert *Scheduling aircraft using constraint relaxation* NLR: Nationaal Lucht & Ruimtevaartlaboratorium (National Aerospace Laboratory), 2003
[4] H.W.G. de Jonge and E.E. Tuinstra and R.R Seljée *Outbound Punctuality Sequencing by Collaborative Planning* NLR: Nationaal Lucht-en Ruimtevaartlaboratorium, (National Aerospace Laboratory), 2005
[5] Günther, Y., A. Inard, B. Werther, M. Bonnier, G. Spies, A. Marsden, M.-M. Temme, D. Bohme, R. Lane and H. Niederstrasser *Total Airport Management (Operational Concept and Logical Architecture)* 2006
[6] D. Bohme, R. Brucherseifer and L. Christoffels, *Coordinated Arrival Departure Management*, German Aerospace Center DLR, Institute of Flight Guidance, Braunschweig, Germany, 2007.
[7] Gotteland, J-B. *Optimisation du trafic au roulage sur des grands aéroports* PHD Thesis, EDIT, 2004.
[8] Gotteland, J-B. and Durand, N. and Alliot, J-M. *Genetic Algorithms Applied to Airport Ground Traffic Optimization* Proceedings of the Congress on Evolutionnary Computation, Canberra, 2003
[9] Gotteland, J-B. and Durand, N. and Alliot, J-M. *Handling CFMU slots in busy airports* Proceedings of the Air Traffic Management R&D Seminar, Budapest, 2003
[10] B. Pesic, N. Durand and J-M. Alliot. *Aircraft Ground Traffic Optimization using a Genetic algorithm* Proceedings of the Genetic and Evolutionary Computation Conference, San Francisco, 2001.
[HT95] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches* Springer-Verlag, 1996.
[11] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows, Theory, Algorithms and Applications* Prentice Hall, 1993.
[12] D-E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning* Addison-Wesley, 1989.
[13] Durand, N. and Alliot, J-M. and Noailles, J. *Automatic aircraft conflict resolution using genetic algorithms* Proceedings of the Symposium on Applied Computing, Philadelphia, 1996.
[14] N. Durand and J. M. Alliot. Genetic crossover operator for partially separable functions. *Genetic Programming Conference.* 1998.