The Influence of Uncertainties on Traffic Control using Speed Adjustments
Géraud Granger, Cyril Allignol, Nicolas Durand

To cite this version:
Géraud Granger, Cyril Allignol, Nicolas Durand. The Influence of Uncertainties on Traffic Control using Speed Adjustments. ATM Seminar 2011, 9th USA/Europe Air Traffic Management Research and Development Seminar, Eurocontrol, Jun 2011, Berlin, Germany. hal-01293549

HAL Id: hal-01293549
https://hal-enac.archives-ouvertes.fr/hal-01293549
Submitted on 24 Mar 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
The Influence of Uncertainties on Traffic Control using Speed Adjustments

Géraud Granger STERIA
7, av Edouard Belin
31055 Toulouse cedex France
granger@recherche.enac.fr
Cyril Allignol, Nicolas Durand
DSNA/DTI/R&D
7, av Edouard Belin
31055 Toulouse cedex France
name@tls.cena.fr

Abstract—The RTA (Required Time Arrival) capabilities of aircraft FMS (Flight Management Systems) offer new opportunities to solve mid-term horizon conflicts (20 minutes in advance) with small speed adjustments. The ERASMUS project has shown promising results of up to 80% conflict resolution using small speed adjustments in the $[-6\%; +3\%]$ range with 20 minutes advance notice. The hypotheses were based on very accurate trajectory predictions (TPs). In this article we show how the quality of these results decreases as the uncertainties on the trajectory prediction increase. Therefore we used the CATS (Complete Air Traffic Simulator) developed in the late 90s at CENA (Centre d’Etudes de la Navigation Aérienne) and constrained the solver to use only speed maneuvers for leveled or descending aircraft with different hypotheses on speed ranges and speed uncertainties. Results show that Traffic Control using Speed Adjustments (TCSA) can solve most of the conflicts even when we consider uncertainty on the TP. However, the number of maneuvers that need to be given to aircraft is highly influenced by the uncertainties used in the TP.

Keywords: speed adjustment, traffic complexity, conflict resolution

INTRODUCTION

Traffic Control using Speed Adjustments (TCSA) has been recently studied through the ERASMUS project. Several experiments were carried out in the Aix-en-Provence control center to test the ability of controllers to detect speed changes, check the compatibility of an automatic ground based speed adjustment tool with the controllers and pilots’ working methods, and measure the potential benefit of the concept [BDG09], [GGG07], [CL07], [Ehr04].

From a theoretical point of view, conflict resolution using speed adjustments has been studied by Pallotino, Feron and Bicci [PFB02] and later by Vela, Solak, Singhose and Clarke [VSSC09] using mixed integer linear programming in a simplified context where aircraft are leveled and can all change their speeds at the same time. It as also been studied by Durand and Alliot in the mid 90s with a stochastic optimization approach [DAN95]. This approach was used in the ERCOS (En Route Conflict Optimized Solver) [GDA01] that is integrated in the CATS (Complete Air Traffic Simulator) fast time simulator.

The current Air Traffic Management system can be conceptually divided into four layers or filters, each with a decreasing time horizon relative to the moment of departure:

1) Strategic (several months), ASM (Air Space Management): design of routes, sectors and procedures
2) (Pre-)Tactical (a few days to a few hours), ATFM (Air Traffic Flow Management): control centers opening schedules define hourly capacities of each open sectors (or groups of sectors). To respect these capacity constraints, the CFMU (Central Flow Management Unit) computes and updates flow regulations and reroutings according to the posted flight plans and resulting workload excess.
3) Real time (5/10 minutes), tactical control: surveillance, coordination with adjacent centers, conflict resolution by various simple maneuvers (heading, flight level, speed) transmitted to the pilots.
4) Emergency (less than 5 min), safety nets: groundbased (Short Term Conflict Alert, Minimum Safety Altitude Warning) and airborne (Traffic Alert and Collision Avoidance System, Ground Proximity Warning System).

The power of TCSA discribed by [BDG09] is that it has the potential to be added between layer 2 and layer 3 without affecting them. Consequently, it could increase the safety of the whole system by adding a layer and also increase the capacity if the number of conflicts solved is high enough. The purpose of this paper is to evaluate the potential number of conflicts that could be solved using fast time simulations on real data, and measure the effect of uncertainties on the results obtained.

Airlines and aircraft manufacturers will agree to pay for new onboard systems if it can be proven that the money invested provides some benefit in terms of efficiency (for example reducing delays) and/or safety. Slightly modifying the speeds of aircraft to increase their separation without interfering with the existing control system could add a safety layer, but the question is to measure the efficiency of the concept for different scenarios of uncertainties.

In this paper we propose to analyse the influence of speed uncertainty on the efficiency of mid-term speed adjustments.
Therefore we test different scenarios of speed adjustments using a [−5%,+5%] speed range, an anticipation time of 20
minutes corresponding to a mid-term time horizon, and two
different update time steps (3 and 5) minutes. These scenarios
are tested on real 2010 traffic data in the French airspace with
different ground speed uncertainties (2%, 5% or 10%) and
different vertical speed uncertainties (10% or 30%).
In the first part, we describe CATS, the simulator used.
The second part details the maneuver modeling and the solver
algorithm used. The third part describes the solver algorithm.
In the last part, numerical results are presented.

I. The Complete Air Traffic Simulator (CATS)

A. General framework

Results presented in this paper rely on real data traffic simul-
ation. The CAT Simulator takes as input flight plans given by
airlines or pilots with or without regulation. In the application,
the traffic used is the French real traffic of July 17th 2010.
The simulator uses the BADA tabulated model for modeling
aircraft performances. For a given aircraft type, it gives a
vertical speed and a ground speed which depends on the
aircraft attitude (whether it is climbing, leveled or descending).
For example, a B747 leveled at FL-300 has a GS of 490 kts. If
it is climbing, its GS will be 480 kts and its VS 1000 ft/min.
At FL-150, values would be respectively 430, 420 and 1800.
Aircraft performances are in tabulated form describing ground
speed, vertical speed, and fuel burn as a function of altitude,
aircraft type and flight segment (cruise, climb or descent). The
main dataset for aircraft flight performance used is the base
of aircraft data (BADA) performance summary tables derived
from the total energy model of EUROCONTROL.
Aircraft speeds can be modified by a random value to take
into account different factors of uncertainty (aircraft load,
winds, etc…). These values can be either computed once at
aircraft activation and remain the same for all the flight, or can
be modified anytime during the flight. Uncertainty modeling
for conflict detection and resolution is introduced in the next
section.

Aircraft follow classical routes (from way-point to way-
point). The flight model is simple. An aircraft first climbs
up to its RFL (Requested Flight Level), then remains leveled
until its top of descent, then descends to its destination.
Aircraft flight is calculated per time step. The time step is
always chosen in order to guarantee that two aircraft face to
face flying at 500 kts could not cross without being closer
than one standard separation at at least one time step. For all
our simulations, we use a 15s time step.

B. General architecture of the system

We sketch here the architecture of the simulator. Each
part will be detailed in the following sections. The system
architecture is presented in figure 1 and 2. The system relies
on three main processes P1, P2, and P3:
• P1 is the traffic simulator.
• P2 is in charge of conflict pair detection, clustering of
pairs, and verification of new trajectories built by the
solver.
• P3 is the problem solver.
P1 sends current aircraft positions and flight plans to process
P2. Process P2 builds trajectory forecasts for \( T_w \) minutes,
does conflict detection by pairs and transforms 1-to-1 conflicts
in n-aircraft conflicts. Then, process P3 (the problem solver)
solves in parallel each cluster, as aircraft in each cluster are
independent from aircraft in the other clusters. The problem
solver sends to P2 new orders and P2 builds new trajectory
forecasts based on these orders. Then P2 once again runs
a conflict detection process to check that modified aircraft
trajectories do not interfere with aircraft in another cluster,
or with new aircraft. If no interference is found, new flight
orders are sent to P1. If there are interferences, interfering
clusters are joined and the solver is used again on that
(cluster) cluster(s). The process is iterated until no interference
between clusters remains, or no new aircraft is concerned by
modified trajectories. The new orders are sent back to the
traffic simulator.
The above process is iterated and all trajectories are opti-
mized each \( \delta \) minutes (3 or 5 minutes in the experiments).
However, during the computation time, aircraft are flying and
need to know if they must change their route or not. \( \delta \) should
be large enough to compute a solution, send it to the pilot and
leave him enough time to begin the maneuver. Consequently,
for each aircraft, at the beginning of the current optimization,
trajectories are determined by the previous run of the problem
solver and cannot be changed for the next \( \delta \) minutes.
has only traveled 00 kts. This means that 166 and minimal climb rate are speed prediction uncertainties error about the aircraft's future location because of ground simulator. But, as stated above, we assume that there is an a simulation on a slightly modified version of the Air Traffic uncertainty is 5% speed. Possible speed, and the last point at the minimum possible in the direction of uncertainty (the speed direction here, see and many other uncertainties, an aircraft is represented by a point at the initial time. The point becomes a line segment in the direction of uncertainty (the speed direction here, see figure 3). The first point of the line "flies" at the maximum possible speed, and the last point at the minimum possible speed.

When changing direction on a beacon, the heading of the line segment’s "fastest point" changes as described on figure 3.

To check the standard separation at time t, we compute the distance between the two line segments modeling the aircraft positions and compare it to the standard separation at each time step of the simulation.

In the vertical plane, we use a cylindrical modeling (figure 3). Each aircraft has a mean altitude, a maximal altitude and a minimal altitude. To check if two aircraft are in conflict, the minimal altitude of the higher aircraft is compared to the maximal altitude of the lower aircraft.

Let us take an example. A B747 is leaving its departing airport (altitude 0) at t = 0. Its climb rate is 1800 ft/min and its ground speed is 175 kts. If we suppose that ground speed uncertainty is 5% and vertical speed uncertainty 20%, maximal and minimal climb rate are 1800 × 1.2 = 2160ft and 1800 × 0.8 = 1440 ft/min and ground speeds are respectively 184 and 166 kts. This means that 15s later, the fastest and highest point has traveled 0.76 Nm and 540 ft while the slowest and lowest has only traveled 0.69 Nm and 360 ft. This time however,

when computing maximal and minimal speeds, the difference of altitude of both points must be taken into account. At 540 ft, the tabulated model gives a standard ground speed of 197 kts, so max ground speed is 197 × 1.2 = 237 kts. At 360 ft, standard ground speed is 189 kts, with a minimal ground speed of 151 kts. Thus, the height of the segment grows much faster than the 20% factor for some aircraft.

Duration Tw can be changed, but must be at least equal to 2 × δ. A good evaluation of Tw is difficult. With a perfect trajectory prediction, the largest Tw should be chosen. However, this is not true as soon as uncertainties are included in the model. A large value of Tw induces a large number of 1-to-1 conflicts, as the size of segments (modeling aircraft positions) grows quickly with time. Therefore, the conflict solver can become saturated.

2) Clustering: After pair detection, P2 clusters conflicting aircraft. Each equivalence class for the relation “is in conflict with” becomes a cluster.

For example, if aircraft A and B are in conflict in the Tw window, and if B is also in conflict with C in the same time window, then A, B, C is the same cluster and will be solved globally by the conflict solver.

The conflict solver sends back to P2 maneuver orders for solving conflicts. Then P2 computes new trajectories for all aircraft and checks if new interferences appear. For example, if the new trajectory given to aircraft B to solve the conflict with A and C interferes with cluster D, E and with aircraft F, then A, B, C, D, E, F will be sent back to the problem solver as one conflict to solve.

The process will always converge. In the worst case, P3 will have to solve a very large cluster including all aircraft present in the next Tw minutes. However, this technique is usually efficient because a very large number of clusters can be solved very quickly in parallel.

II. SOLVER MODELING

A. Maneuver modeling

In the CATS simulator, the conflict solver offers different maneuver options including heading changes (10, 20 or 30 degrees right or left), vertical maneuvers (flight level change, climbing interruption, descent anticipation, depending on if the aircraft is leveled, climbing or descending), or speed maneuvers. A maneuver starts at time t1 and ends at time t2. In this paper, we will limit the maneuvers to speed modifications when the aircraft is leveled or descending. We do not allow speed modifications during the climbing phase because many aircraft cannot change their speeds during this phase without changing the climbing rates. Different ranges ([-15%, +15%], [-10%, +10%], [-5%, +5%]) of speed changes have been tested in the experiments in order to measure the influence of the this parameter.

A maneuver is determined by:
- t0: the beginning of the speed change.
- α: the speed change
- t1: the end of the speed change.

For a conflict involving n aircraft, the dimension of the search space is 3n.
B. Maneuver decision time

Because of uncertainties, some conflicts that are detected too early would not actually occur in the end. Consequently, deciding to move aircraft in such cases would be useless and could even generate other conflicts that would not occur if no maneuver had been decided. This explains why controllers do not solve conflicts too early. When there is no uncertainty, the earlier the maneuver is started, the lower the delay. However, if speed is not strictly maintained, the earlier the conflict is detected, the lower the probability it will actually happen. Thus, a compromise must be reached between the delay generated and the risk of conflict.

Because of uncertainties, maneuvers should be started as late as possible with respect to the aircraft constraints. First, it prevents the system from deciding unnecessary maneuvers. Second, an aircraft that is already maneuvering (i.e. changing speed) cannot do another maneuver before resuming its initial speed. The solver was modeled this way to keep the maneuvers simple to understand and execute. Starting maneuvers as late as possible increases the number of maneuverable aircraft.

C. A sliding forecast time window model

In order to limit the size of the problem and to be reactive to uncertainties, only the next \( T_w \) minutes of the flights are considered. \( T_w \) represents the lookahead time also called forecast time window. The situation is reconsidered every \( \delta \) minutes with \( \delta << T_w \). \( \delta \) is the time step used in the model to make the \( T_w \) time window slide. This approach ensures that the problem can be updated every \( \delta \) minutes: current aircraft positions are updated which reduces uncertainties.

In figure 4 at \( t = 0 \), the aircraft trajectory cannot be modified before \( t = \delta \) because any maneuver requires advance notice. Any maneuver that would occur between \( t = \delta \) and \( t = 2\delta \) would be kept as a constraint for the next optimization run (in the example, no maneuver is decided). In figure 2 the maneuver described on the first line resulting from an optimization at \( t = 0 \) is reconsidered at time \( t = \delta \) and then kept at time \( t = 2\delta \). The end of the maneuver can be recalculated until the optimization starting at \( t = 3\delta \).

Pilots should only be given maneuver orders that will not be modified; if no conflict occurs, no order will be given. In the example, the pilot will be notified of the beginning of the speed change at time \( \delta \) and the end at time \( 3\delta \).

The size of the forecast time window is an important parameter. If it is too big, the size of the problem will include a very large number of variables and the resolution might be more difficult. If it is too small, the solutions found might be worse and the total delay induced over the day much higher. In order to solve conflicts with small speed adjustments, \( T_w \) needs to be large enough. Based on the ERASMUS experiments, we decided to set \( T_w = 20\text{min} \) for this analysis.

III. Solver algorithm

Classical Evolutionary Computation (CEC) principles such as described in the literature \([\text{Gol89}, \text{Mic92}]\) is used in the solver.

A. Fitness function

The cost function used in this part is simply the sum of the delays over the aircraft population.

Solutions respecting the separation constraints cannot be built easily. Consequently, we need to include the separation constraint in the fitness function.

The fitness function chosen is:

\[
F = \frac{n - \sum_{i=1}^{n} \left( \frac{\delta_i}{\delta_{\text{max}}} \right)}{1 + n_{rc}}
\]

where \( n \) is the number of aircraft and \( n_{rc} \) is the number of remaining conflicts.

The fitness function increases when the number of remaining conflicts and the delays decrease. It takes its values in \([0, n]\).

B. Crossover operator

The conflict resolution problem is partially separable as defined in \([\text{DA98}, \text{DAN96}]\). In order to increase the probability of producing children with a better fitness than their parents, principles applied in \([\text{DA98}]\) were used. For each aircraft \( i \) of a population element, a local fitness \( F_i \) value is defined as follows:

\[
F_i = 1 - \frac{\delta_i}{\delta_{\text{max}}} \frac{1}{1 + n_{rc_i}}
\]

where \( n_{rc_i} \) is the number of remaining conflicts involving aircraft \( i \).

Figure 5 presents the crossover operator. First two population elements are randomly chosen. For each parent \( A \) and \( B \), fitness \( A_i \) and \( B_i \) of aircraft \( i \) are compared. If \( A_i < B_i \), the children will take aircraft \( i \) of parent \( A \). If \( B_i < A_i \), the children will take aircraft \( i \) of parent \( B \). If \( A_i = B_i \), children randomly choose aircraft \( A_i \) or \( B_i \) or even a combination of \( A_i \) and \( B_i \).

C. Mutation operator

For each candidate to mutation, the delay of an aircraft as described in the literature \([\text{Gol89}, \text{Mic92}]\) is used in the solver.

The size of the forecast time window is an important parameter. If it is too big, the size of the problem will include a very large number of variables and the resolution might be more difficult. If it is too small, the solutions found might be worse and the total delay induced over the day much higher. In order to solve conflicts with small speed adjustments, \( T_w \) needs to be large enough. Based on the ERASMUS experiments, we decided to set \( T_w = 20\text{min} \) for this analysis.

III. Solver algorithm

Classical Evolutionary Computation (CEC) principles such as described in the literature \([\text{Gol89}, \text{Mic92}]\) is used in the solver.

A. Fitness function

The cost function used in this part is simply the sum of the delays over the aircraft population.

Solutions respecting the separation constraints cannot be built easily. Consequently, we need to include the separation constraint in the fitness function.

The fitness function chosen is:

\[
F = \frac{n - \sum_{i=1}^{n} \left( \frac{\delta_i}{\delta_{\text{max}}} \right)}{1 + n_{rc}}
\]

where \( n \) is the number of aircraft and \( n_{rc} \) is the number of remaining conflicts.

The fitness function increases when the number of remaining conflicts and the delays decrease. It takes its values in \([0, n]\).

B. Crossover operator

The conflict resolution problem is partially separable as defined in \([\text{DA98}, \text{DAN96}]\). In order to increase the probability of producing children with a better fitness than their parents, principles applied in \([\text{DA98}]\) were used. For each aircraft \( i \) of a population element, a local fitness \( F_i \) value is defined as follows:

\[
F_i = 1 - \frac{\delta_i}{\delta_{\text{max}}} \frac{1}{1 + n_{rc_i}}
\]

where \( n_{rc_i} \) is the number of remaining conflicts involving aircraft \( i \).

Figure 5 presents the crossover operator. First two population elements are randomly chosen. For each parent \( A \) and \( B \), fitness \( A_i \) and \( B_i \) of aircraft \( i \) are compared. If \( A_i < B_i \), the children will take aircraft \( i \) of parent \( A \). If \( B_i < A_i \), the children will take aircraft \( i \) of parent \( B \). If \( A_i = B_i \), children randomly choose aircraft \( A_i \) or \( B_i \) or even a combination of \( A_i \) and \( B_i \).

C. Mutation operator

For each candidate to mutation, the delay of an aircraft having one of the worst local fitnesses is modified. If every conflict is solved, an aircraft is randomly chosen and its parameters changed. In practice, a number \( m \) is randomly chosen in the interval \([1, \frac{n}{2}]\) and we pick up \( m \) times an
aircraft to find the most constrained aircraft among these $m$ trials. The delay of this aircraft is then either locally optimized or randomly modified with a probability of 50%. We may be tempted to always locally optimize the delay of the worst aircraft, but this would make the algorithm become very deterministic and lead to a premature convergence of the algorithm.

The crossover and mutation operators are more deterministic during the first generations because there are many conflicts to solve. They focus on making feasible solutions. When the solutions without conflicts appear in the population, they become less deterministic.

Sharing: The problem is highly combinatorial and may have many local optima. In order to prevent the algorithm from premature convergence, the sharing process introduced by Yin and German [YG93] is used. The complexity of this sharing process has the great advantage to be in $n \log(n)$ (instead of $n^2$ for classical sharing) if $n$ is the size of the population. The distance used to compare two population elements $p$ and $q$ is:

$$D = \sum_{i=1}^{n} \left| \delta^p_i - \delta^q_i \right|$$

D. Parameters

In the experiments, the following parameters were empirically chosen: the size of the population was set to 100, 20% of the population is crossed, 60% is muted, the selection uses the stochastic remainder without replacement. A sharing process is used. As time to solve a problem is limited, the number of generations is limited to 500.

IV. Experimental Results

CATS was used on a busy day of traffic (July 17th 2010) in the French airspace. We were only interested in the upper airspace (above FL195). The number of flights in this airspace is 8870 for that specific day. Without any control maneuver, 2305 conflicts are detected (26% of the aircraft population). The mean time of flights is 57 minutes and the mean travelled distance in the French airspace is 394 nautical miles. Different experiments were done using CATS with different hypotheses. Tables I and II give the results of simulations for $\delta = 3\text{min}$ and $\delta = 5\text{min}$ for different values of horizontal uncertainty (0, 2, 5 or 10) and vertical uncertainty (10 or 30%). Columns 4 and 5 give the number and percentage of remaining conflicts. Column 4 also gives the number of remaining conflicts that involve two leveled aircraft. Columns 6 and 7 give the number of maneuvered aircraft and the percentage of aircraft maneuvered. Columns 7 and 8 give the number of maneuvers and mean number of maneuvers per aircraft.

A scenario with no uncertainty and a $[-15\%, +15\%]$ speed range was also tested (first line of tables I and II) in order to show what “theoretical” gain can be achieved on this specific day. In this case 97% of the conflicts could be solved by changing the speed of less than 2000 aircraft (20% of the aircraft population) which is very low. When the speed range is reduced to $[-5\%, +5\%]$ the number of remaining conflicts doubles with a number of aircraft maneuvered slightly above 2000. Those results show that with a perfect TP most of the conflicts could be solved 20 minutes in advance with a small speed adjustment. The lower lines of the tables show how the results worsen when the uncertainty on the TP increases. For example, with a 5% horizontal speed uncertainty and 10% vertical speed uncertainty the number of aircraft maneuvered to solve the conflicts is tripled and the number of remaining conflicts grows to 15% which is still good. Results show that vertical uncertainty increases the number of conflicts and maneuvers because many conflicts involve climbing or descending aircraft. In the worse case tested (10% of uncertainty on both horizontal and vertical speeds and a $\delta = 5\text{min}$) the number of remaining conflicts reaches 50% and the number of necessary maneuvers increases dramatically.

5% of horizontal speed uncertainty corresponds to 20 knots. When comparing the results using $\delta = 3\text{min}$ to those using $\delta = 5\text{min}$ it appears that the number of aircraft maneuvered increases by 10% and the number of remaining conflicts by 50%. This means that the solver is sensitive to the frequency of the detection and resolution cycle updates.

Results also show that uncertainties have a big influence on the number of maneuvers needed to solve the conflicts. Controllers are used to saying that they often have to solve conflicts that might never happen because of uncertainties. It appears that an automatic TCSA tool would encounter the same issue.

The results obtained during the ERASMUS project were slightly different because the speed range used was $[-6\%, +3\%]$ and the separation standard was 8 nautical miles.
### TABLE I

<table>
<thead>
<tr>
<th>horiz uncert</th>
<th>vert uncert</th>
<th>speed range</th>
<th>remaining conflicts</th>
<th>% of rem conflicts</th>
<th>maneuvered aircraft</th>
<th>% of acft maneuvered</th>
<th>nb of maneuvers</th>
<th>nb of man per aircraft</th>
</tr>
</thead>
<tbody>
<tr>
<td>0%</td>
<td>0%</td>
<td>[-15 15]</td>
<td>78 (69)</td>
<td>3%</td>
<td>1889</td>
<td>21%</td>
<td>2397</td>
<td>0.27</td>
</tr>
<tr>
<td>0%</td>
<td>0%</td>
<td>[-10 10]</td>
<td>82 (72)</td>
<td>4%</td>
<td>1940</td>
<td>22%</td>
<td>2428</td>
<td>0.27</td>
</tr>
<tr>
<td>2%</td>
<td>10%</td>
<td>[-5 5]</td>
<td>144 (131)</td>
<td>6%</td>
<td>2102</td>
<td>24%</td>
<td>2605</td>
<td>0.29</td>
</tr>
<tr>
<td>2%</td>
<td>30%</td>
<td>[-5 5]</td>
<td>173 (142)</td>
<td>7%</td>
<td>3609</td>
<td>41%</td>
<td>5023</td>
<td>0.57</td>
</tr>
<tr>
<td>5%</td>
<td>10%</td>
<td>[-5 5]</td>
<td>347 (244)</td>
<td>8%</td>
<td>4663</td>
<td>53%</td>
<td>7093</td>
<td>0.80</td>
</tr>
<tr>
<td>5%</td>
<td>30%</td>
<td>[-5 5]</td>
<td>392 (279)</td>
<td>11%</td>
<td>6194</td>
<td>70%</td>
<td>11447</td>
<td>1.29</td>
</tr>
<tr>
<td>10%</td>
<td>10%</td>
<td>[-5 5]</td>
<td>872 (612)</td>
<td>14%</td>
<td>6971</td>
<td>79%</td>
<td>14808</td>
<td>1.67</td>
</tr>
<tr>
<td>10%</td>
<td>30%</td>
<td>[-5 5]</td>
<td>1019 (663)</td>
<td>17%</td>
<td>7445</td>
<td>84%</td>
<td>16952</td>
<td>1.91</td>
</tr>
</tbody>
</table>

$\delta = 3\min$: NUMBER OF REMAINING CONFLICTS, NUMBER OF MANEUVERED AIRCRAFT AND NUMBER OF MANEUVERS FOR DIFFERENT UNCERTAINTIES AND DIFFERENT SPEED MODIFICATION RANGES.

### TABLE II

<table>
<thead>
<tr>
<th>horiz uncert</th>
<th>vert uncert</th>
<th>speed range</th>
<th>remaining conflicts</th>
<th>% of rem conflicts</th>
<th>maneuvered aircraft</th>
<th>% of acft maneuvered</th>
<th>nb of maneuvers</th>
<th>nb of man per aircraft</th>
</tr>
</thead>
<tbody>
<tr>
<td>0%</td>
<td>0%</td>
<td>[-15 15]</td>
<td>100 (84)</td>
<td>4%</td>
<td>2026</td>
<td>21%</td>
<td>2634</td>
<td>0.3</td>
</tr>
<tr>
<td>0%</td>
<td>0%</td>
<td>[-10 10]</td>
<td>107 (89)</td>
<td>5%</td>
<td>2057</td>
<td>23%</td>
<td>2700</td>
<td>0.3</td>
</tr>
<tr>
<td>0%</td>
<td>0%</td>
<td>[-5 5]</td>
<td>205 (166)</td>
<td>9%</td>
<td>2349</td>
<td>26%</td>
<td>3060</td>
<td>0.34</td>
</tr>
<tr>
<td>2%</td>
<td>10%</td>
<td>[-5 5]</td>
<td>251 (204)</td>
<td>11%</td>
<td>4162</td>
<td>47%</td>
<td>6058</td>
<td>0.68</td>
</tr>
<tr>
<td>2%</td>
<td>30%</td>
<td>[-5 5]</td>
<td>334 (232)</td>
<td>14%</td>
<td>5284</td>
<td>60%</td>
<td>8258</td>
<td>0.93</td>
</tr>
<tr>
<td>5%</td>
<td>10%</td>
<td>[-5 5]</td>
<td>517 (359)</td>
<td>22%</td>
<td>5773</td>
<td>65%</td>
<td>9688</td>
<td>1.09</td>
</tr>
<tr>
<td>5%</td>
<td>30%</td>
<td>[-5 5]</td>
<td>591 (386)</td>
<td>26%</td>
<td>6544</td>
<td>74%</td>
<td>11995</td>
<td>1.35</td>
</tr>
<tr>
<td>10%</td>
<td>10%</td>
<td>[-5 5]</td>
<td>1150 (801)</td>
<td>50%</td>
<td>7133</td>
<td>80%</td>
<td>14428</td>
<td>1.62</td>
</tr>
<tr>
<td>10%</td>
<td>30%</td>
<td>[-5 5]</td>
<td>1410 (935)</td>
<td>61%</td>
<td>7593</td>
<td>86%</td>
<td>16675</td>
<td>1.88</td>
</tr>
</tbody>
</table>

$\delta = 5\min$: NUMBER OF REMAINING CONFLICTS, NUMBER OF MANEUVERED AIRCRAFT AND NUMBER OF MANEUVERS FOR DIFFERENT UNCERTAINTIES AND DIFFERENT SPEED MODIFICATION RANGES.
V. CONCLUSION

Controllers are aware of the fact that part of their task is to solve many conflicts that would never occur, because they have to deal with uncertainty. Improvements in Trajectory Prediction offer new opportunities to enhance the efficiency and the safety of Air Traffic Management. Traffic Control using Speed Adjustments is one of the means that could be used to reach this goal. However, the expected gain of efficiency depends on the quality of the TP that can be used by the new tools.

This paper shows that having an accurate Trajectory Prediction is necessary to define a TCSA tool. These results do not suggest that we need to constrain the flight path to meet some Required Time Arrivals with a long advance notice. Instead, we need to be able to predict the future positions 20 minutes in advance with accuracy. Pilot attitude uncertainties are difficult to predict and our research was built on the strong hypothesis that the pilot would execute the maneuvers proposed by the solver using the Aircraft Flight Management System. Adding maneuver execution time uncertainty would probably worsen the results presented in this paper. This extra uncertainty needs to be studied carefully. Having both controllers and pilots interacting with a medium term solver is challenging because the controller will probably solve conflicts that are handled by the solver. This is why the ERASMUS experiments used a bigger separation standard. Pilot controller interaction with the medium term solver also needs to be modeled carefully and studied in future simulations to be able to give a realistic measure the possible gain of using TCSA.

REFERENCES


Géraud Granger graduated from the Ecole Nationale de l’Aviation Civile (ENAC) in 1998. He holds a Ph.D. in Computer Science from the Ecole Polytechnique de Paris (2002) and is currently a design engineer at STERIA working on SESAR.

Cyril Allignol graduated from the Ecole Nationale de l’Aviation Civile (ENAC) in 2006. He is currently a Ph.D. student at the DSNA/R&D Planning Optimization and Modeling team.

Nicolas Durand graduated from the Ecole polytechnique de Paris in 1990 and the Ecole Nationale de l’Aviation Civile (ENAC) in 1992. He has been a design engineer at the Centre d’Etudes de la Navigation Aérienne (now DSNA/DTI R&D) since 1992, holds a Ph.D. in Computer Science (1996) and got his HDR (French equivalent of tenure) in 2004. He is currently the deputy head of the DSNA/R&D department.