



HAL
open science

Arithmetic simulation and performance metrics

Jean-Marc Alliot, Géraud Granger, Jean-Marc Pomeret

► **To cite this version:**

Jean-Marc Alliot, Géraud Granger, Jean-Marc Pomeret. Arithmetic simulation and performance metrics. DASC 2002, 21st Digital Avionics System Conference, Oct 2002, Irvine, United States. 10.1109/DASC.2002.1067922 . hal-01021710

HAL Id: hal-01021710

<https://hal-enac.archives-ouvertes.fr/hal-01021710>

Submitted on 20 Nov 2014

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.

ARITHMETIC SIMULATION AND PERFORMANCE METRICS

Jean-Marc Alliot, Géraud Granger, Laboratoire d'Optimisation Globale, Toulouse, France

Jean-Marc Pomeret, Centre d'Etudes de la Navigation Aérienne, Toulouse, France

Abstract

Performance metrics are becoming a strategic issue, and they are getting more and more attention. However, defining such metrics is a difficult problem. In this paper, we show how arithmetic simulations can be used to give performance information. We also point out that many different metrics can be defined, each of them giving different results regarding efficiency. We conclude that an extreme caution must be applied when interpreting results.

Introduction

Performance review is now looked upon as a standard activity in the ATM community [9,10]. However, defining performance is not easy. The basic definition usually relies on the ability of a system to handle traffic without generating delays. But such a definition is utterly poor, as it is much more difficult to handle highly dense traffic over the core European area than sparse traffic over deserts in Africa. So, on the one hand, a sector might be generating delays just because its traffic is more difficult to handle. On the other hand it could have a very simple traffic to handle and then should generate no delay at all. Finally, it generates few delays because it is protected by another one. Defining lack of performance simply by generated delays doesn't work. Thus, a new notion steps forward: traffic complexity. But then again, defining complexity is not an easy task either. Many different complexity metrics can be used, each of them being apparently sensible, but all of them giving different results.

Beyond metrics definition, measurement is by itself a difficult process. While some very basic metrics (such as the number of aircraft controlled in one sector for one day, for example) can be easily computed by a statistical analysis of data samples, some others are not. For example, if we want to measure the number of "conflicts to solve" by

sector, analyzing radar samples won't help much as, if the controller is doing his job properly, there should be no remaining conflict.

In that case, the use of arithmetic, or fast time, simulations is required. Such simulators can handle one day of traffic in a few minutes. They however, introduce new weak points, as a simulation always relies on some hypothesis that may or may not be correct. In fact, as we will show in this paper, we have mainly the choice between simple reliable metrics, which may not be very meaningful, and more elaborate metrics, probably more meaningful, but not as reliable.

Metrics Considered

In this paper, we will concentrate on four different metrics: flights controlled per sector, number of conflicts per sector, number of potential conflicts per sector and number of maneuvers given to aircraft for solving conflict in one sector. The three last values are computed using a fast time simulator, while the first is extracted from flight samples. We also use in the rest of this paper the *sector volume*, which is the sector raw surface divided by the square of the radar separation minimum, multiplied by the number of available flight levels. This is roughly the number of aircraft that could be frozen and packed inside without violating standard separation constraints.

Number Of Flights By Sector

This metric is certainly the easier to define. Using real flight plans, we just compute the number of flights controlled by sectors. However, to have an idea of the CFMU regulations we also used initial flight plans to compare both results.

Number Of Conflicts By Sector

A significant part of the controller workload is conflict resolution. Then, we try to count the

number of standard separation violations by sector with aircraft following exactly their flight plan routes with no deviations. This is done by fast time simulation, the hypothesis of which being detailed thereafter.

Number Of Potential Conflicts By Sector

A large part of controllers workload comes from trajectory monitoring and conflict detection. Different studies show that only one conflict out of three to five detected and monitored would really result in separation violation. This is directly the consequence of uncertainties affecting aircraft trajectories (wind, unavailable FMS information, etc), and of human beings inability to handle complex numerical mathematics to compute trajectory predictions. Thus the number of potential conflicts is the number of conflicts detected, given a set of hypothesis regarding aircraft trajectories uncertainties, both in the horizontal and the vertical plane. There again, this value is estimated using fast time simulation.

Number Of Maneuvers By Sector

A more subtle indicator is the number and the category of maneuvers given inside a sector to solve conflicts. Some conflicts involving many aircraft can sometimes be solve by only one maneuver, while others may require many maneuvers. There again fast time simulation is used to estimate this parameter.

The OPAS Fast Time Simulator

Why using fast time simulations? There are different reasons that could be summarized as follow:

- there is sometimes no other way to compute some metrics. In real life, the number of conflicts is impossible to compute, as competent controllers are usually doing properly their job, thus suppressing conflicts. The only way to estimate the number of conflicts that would have happened is to simulate traffic without any control actions. The same goes with potential conflict

detection, as there is no way to be in the head of the controller.

- some metrics are too difficult to find out. For example, the number of maneuvers given could be found by a close examination of radar samples, but it would be difficult to find out which maneuvers are given to solve conflicts and which are given for other purposes.

In the remaining part of this section, the OPAS fast time simulator is presented. This simulator and its different modules have been presented in many different articles [3,4,5,6], and we will only concentrate on its main features.

General Principles And Conflict Detection

The simulator uses a tabulated model for aircraft performances: ground speed, vertical speed, and fuel burn are functions of altitude, aircraft type and flight segment (cruise, climb or descent.) The main dataset for aircraft flight performance is the base of aircraft data (BADA) performance summary tables derived from the total energy model of EUROCONTROL. 69 different aircraft types are described. Synonym aircraft are used to model the rest of the fleet.

Aircraft follow classical routes (from way-point to way-point). The flight model is simple: an aircraft first climbs up to its RFL, then remains leveled till its top of descent, then descends to its destination.

Flight plans are data of the COURAGE system, an archiving system of the operational French CAUTRA Air Traffic Control system.

Aircraft fly with a timestep that can be chosen at the start of the simulation. The timestep 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 radar separation minimum at at least one timestep. For most of our simulation, we use a 15s timestep. Separation minima are checked for each aircraft at each timestep.

Potential Conflict Detection

For potential conflict detection, the simulator does a trajectory prediction *each three minutes* by a

simulation of a given duration inside the global simulation. This duration is what we call the *anticipation*.

We assume during each of these detection simulations that there is an error about the aircraft future location because of ground and vertical speed prediction uncertainties.

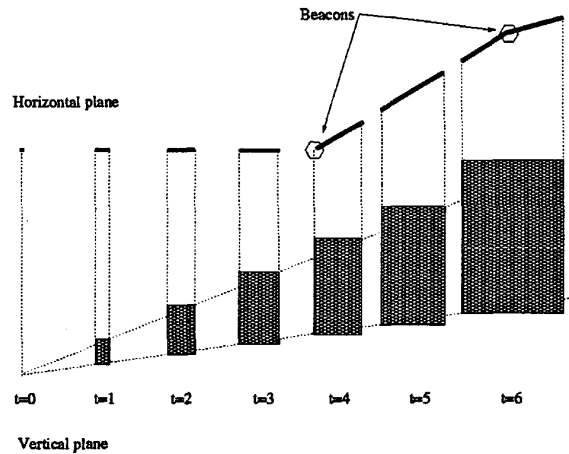


Figure 1. Modeling Of Speed Uncertainties (Standard Routes).

Then, an aircraft is represented by a point at the initial time of the conflict detection window. In the horizontal plane, the point becomes a line segment in the uncertainty direction (the speed direction here, see figure \ref{figure2}). The first point of the line "flies" at the maximum possible speed, and the last point at the minimum possible speed. These maximal and minimal speeds depend of course on the uncertainty chosen: for 5% uncertainty on ground speed, the first point will fly at a speed of $1.05v$ and the last point at $0.95v$, if v is the nominal speed of the aircraft.

When changing direction on a waypoint, the heading of the line segment "fastest point" changes as described on figure 1.

To check separation for two aircraft at time t , we compute the distance between the two line segments modeling the aircraft positions and compare it to the separation minima.

In the vertical plane, we use a cylindrical modeling (figure 1). 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. The radar separation minima used are 6 nautical miles in the horizontal plane and, 1000 ft under FL295 and 2000 ft above (no RVSM) in the vertical plane.

Conflicts detected can be merged: if a conflict is detected a time t_1 , and detected again three minutes later, the two conflicts are only considered as one.

Maneuvers For Conflict Resolution

In the horizontal plane, classical maneuvers given to aircraft are heading deviation. In the simulator, 10, 20 or 30 degrees deviations will be allowed. The deviation starts on a virtual waypoint created on the route (see figure 2). This waypoint is defined by the position of the head of the segment at some time t_0 . It ends on a second virtual waypoint, position of the head of the segment at time t_1 . An angle criteria is defined to find on which waypoint the modified and initial routes should connect.

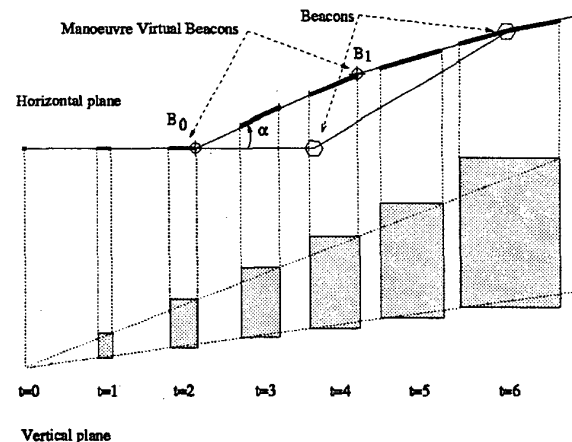


Figure 2. Horizontal Manoeuvre Modeling

A maneuver will be determined by:

- t_0 which defines the first virtual waypoint B_0 .
- the deviation angle α .
- t_1 which defines the second virtual waypoint B_1 .

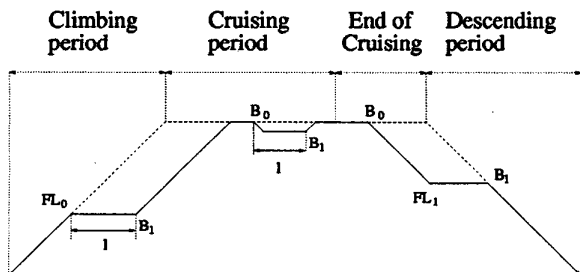


Figure 3. Vertical Manoeuvre Modeling

In the vertical plane, the aircraft trajectory is divided in 4 periods (figure 3):

- Climbing period. In this period, aircraft can be leveled at a lower than requested flight level to solve a conflict. The aircraft climb is stopped at flight level FL_0 and resumes climbing on a virtual waypoint B_1 as stated on figure 4. FL_0 and B_1 are defined by the position of the head of the uncertainty segment at time t_0 and t_1 .
- Cruising period. When aircraft have reached their desired flight level, they may be moved to the nearest lower level to resolve a conflict. Aircraft starts descending when reaching a virtual beacon B_0 and starts climbing at B_1 ($\alpha=0$), B_0 and B_1 are defined by the position of the head of the uncertainty segment at time t_0 and t_1 . An example of maneuver is represented on figure 5.
- End of Cruising period. When aircraft are about 50 nautical miles away from the top of descent, they may be moved to a lower level to resolve a conflict. Aircraft start descending on B_0 and are leveled at FL_1 ($\alpha=0$) (see figure 6). B_0 and FL_1 are defined by the position of the head of the uncertainty segment at time t_0 and t_1 .
- Descending period. During this period no vertical maneuver is possible.

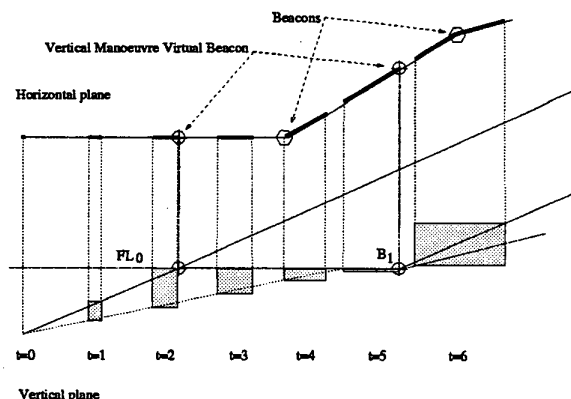


Figure 4. Vertical Maneuver During The Climbing Period

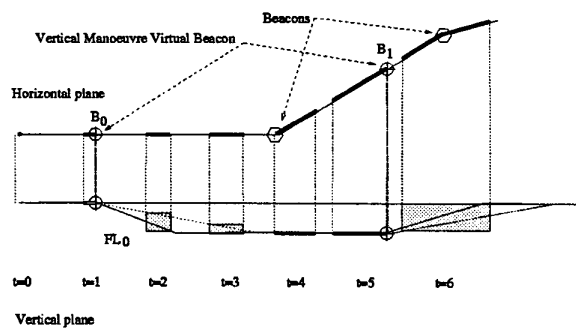


Figure 5. Vertical Maneuver During The Cruising Period

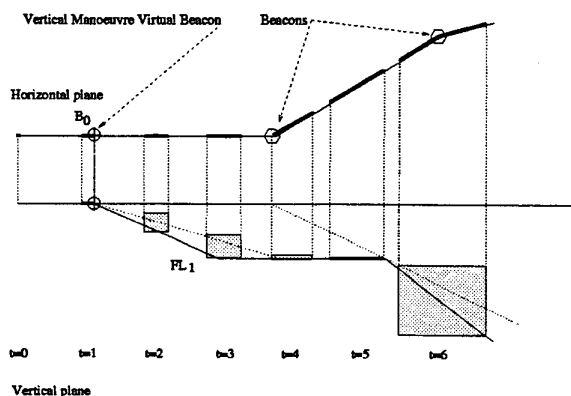


Figure 6. Vertical Maneuver During The End Of Cruising

No maneuver will be simultaneously done in the horizontal and vertical plane.

These manoeuvres were chosen in order to be as close as possible to the controller's behavior.

Maneuver Decision Time

Because of uncertainties, a conflict that is detected early before it could occur may finally not happen. Consequently, deciding to move an aircraft in that case could sometimes 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. With the turning point modeling, 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.

Choosing The Model

Initially, aircraft are allowed to use their flight plan routes.

If we do not want to call into question previous maneuvers and be able to solve very large conflicts, we must try to start maneuvers as late as possible with respect to the aircraft constraints. This argument is enforced by the fact that we allow aircraft to have large uncertainties on their speeds¹.

For example, the first trajectory of figure 7, at $t=0$, cannot be modified before $t=\delta$. At the end of the first optimization run, at $t=\delta$, the current position of the aircraft is updated. The maneuver that occurred between $t=\delta$ and $t=2\delta$ is kept as a constraint for the second optimization run (on the example, no maneuver is decided). In the above example, we can see that the maneuver described on line 2 (resulting from an optimization at $t=\delta$) is more penalizing than the maneuver described on

¹ We do not plan to solve conflicts by speed modifications. Theoretical study shows that optimal En Route conflict resolution by speed modifications would require large anticipation time (anticipation time depends on different parameters such as angle of convergence, speed margins for each aircraft, standard separation etc; more details can be found in [2]). This is quite unrealistic due to aircraft speed uncertainties.

line 3 (resulting from an optimization at $t=2\delta$). This phenomenon occurs because of uncertainties. If uncertainties on speed are important, having a small δ will be very helpful to minimize the resolution costs in the real time situation.

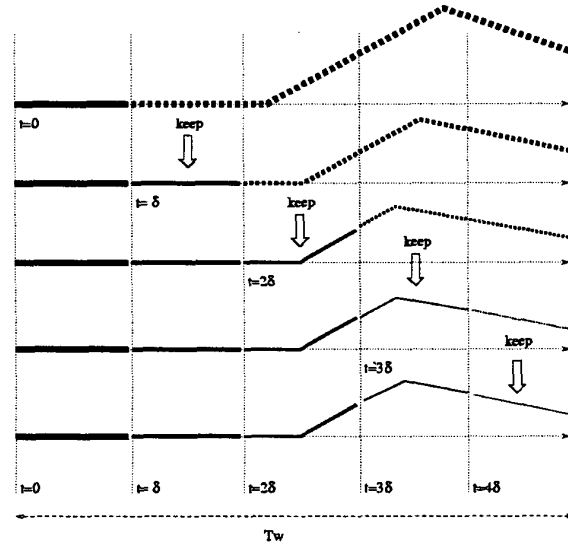


Figure 7. The Model And Real Time Optimization

Pilots should only be given maneuver orders that will not be modified; if no conflict occurs, no order will be given.

Results

As stated above, we used data (flight plans, airspace, etc) for the French airspace. There are two main reasons:

- we have full access to these data, and have a quite good knowledge of their reliability and their weaknesses
- it would have probably been more interesting to try to compare airspaces of different countries, but results could have been misinterpreted, and would have been much more subject to controversy. Our goal here was to discuss from a general standpoint the problems linked to performance metrics and fast time simulations, to develop a general methodology, but not to start a

discussion about the efficiency of European national air traffic control systems. The present work can be considered as an initial experiment, leaving the field open to more ambitious projects.

Validation

Using fast time simulations introduces biases. Then, it is mandatory to verify that results fit the reality of traffic before going any further. Then results of fast time simulations must be compared to real radar data statistics. We used IMAGE radar statistics to compare values such as Standard Transit Time (STT), Standard Traveled Distance (STD), and number of aircraft controlled. These were the first results generated, and they are presented in table 1. The error column indicates the difference between radar statistics and fast time simulation results.

Table 1. Comparison Of Fast Time Simulation Results And Real Data For The 5 French ACC

	STT	Error	STD	Error	Acft	Error
Aix	30mn	3%	182nm	2%	3442	2%
Bordeaux	30mn	3%	194nm	1%	2274	3%
Brest	28mn	2%	197nm	1%	2431	2%
Paris	21mn	5%	117nm	3%	3887	4%
Reims	18mn	5%	112nm	3%	2564	3%

Differences are insignificant and can be either due to the simulator or to radar data, which are not completely reliable and easy to interpret.

Computing Metrics

We present here the results of simulations for the French sectors. In table 2 and 3, we give the 5 highest ranked sectors regarding the 4*2 indicators defined above.

Table 2. Sector Ranking (P:Paris, Br:Brest, R:Reims, A:Aix)

	Flights	Real	Potential	Manœuvres
1	TC,(P) 1544	SE,(R) 135	DS,(P) 477	SE,(R) 158
2	DS,(P) 1217	DS,(P) 117	LN,(P) 470	XS,(Br) 135
3	LN,(P) 1211	UE,(R) 92	SE,(R) 295	AR,(P) 130
4	TS,(P) 884	ZS,(Br) 91	TC,(P) 192	UE,(R) 130
5	SE,(R) 718	LN,(P) 91	TP,(P) 189	UR,(R) 127

Ranking by number of flights : see column 1 of tables 2 and 3.

Ranking by number of conflicts: see column 2 of tables 2 and 3.

Ranking by number of potential conflicts: see column 3 of tables 2 and 3.

Ranking by number of manoeuvres: see column 4 of tables 2 and 3.

An excellent example is the TC sector, which is the one having the largest number of controlled flights, but doesn't even appear in the "flight per volume" indicator (rank 10), or in the "number of conflicts to solve" indicator (rank 12). Only one sector (DS) appears in the top five of six classifications (but not in the manoeuvres classification), the rest being scattered.

Table 3. Sector Ranking

	Fl/Vol	Conf./Vol	Pot./Vol	Man./Vol
1	DS,(P) 6.16	ZS,(Br) 0.68	DS,(P) 2.41	ZS,(Br) 0.90
2	ZS,(Br) 5.01	DS,(P) 0.59	ZS,(Br) 0.92	XS,(Br) 0.75
3	XS,(Br) 3.20	XS,(Br) 0.38	LN,(P) 0.72	XH,(R) 0.38
4	UZ,(P) 2.27	XN,(R) 0.23	XS,(Br) 0.65	SU,(P) 0.31
5	TU,(P) 2.18	XH,(R) 0.21	XN,(R) 0.32	B2,(A) 0.31

Then, after a review of these results, it seems extremely difficult to find the most "loaded" or "efficient" control sectors. Moreover, even the simplest indicator (number of flights going through one sector) should be examined with great care, as it is highly probable that some flights crossing sectors such as TC (arrivals) or DS (departures) are not controlled by these sectors but directly by the approaches of Paris airports. And we are not through yet...

AIR TRAFFIC COMPLEXITY INDICATORS & ATC SECTORS CLASSIFICATION

*Raphaël Christien, Azzedine Benkouar, Thomas Chaboud, Pierre Loubieres,
Eurocontrol, Brétigny-sur-Orge, France*

Abstract

It is a widely held view that complexity is a key factor that significantly affects the work of an air traffic controller, which, in turn affects capacity. A better understanding of what makes the controllers' work complex will improve current and future Air Traffic Management (ATM) capacity, analysis, airspace planning, and future Air Traffic Control (ATC) development.

This paper describes our approach to develop a macroscopic model that will give us an automatic and non-subjective method to classify sectors according to their complexity.

The first step was to identify the complexity indicators. We combined ATC operational advice with statistical analysis to compile a list of relevant complexity indicators. Clearly, these indicators, their influence and interaction vary amongst sector types. Hence, our next step was to classify our sectors into a small number of homogenous groups, or clusters to arrive at sectors' typology. We used two approaches to classify the sectors. The first was based on a K-means classification and the second was by descendant hierarchical clustering – divisive segmentation.

Our study shows that our model gave us a meaningful typology and understanding of our sectors' complexity and that we can improve future controller workload and sector capacity predictions at a macroscopic level.

Introduction

Current capacity restrictions are implicit contributors to safety. They help to ensure that the controllers' work remains within certain limits.

As traffic numbers increase, greater demands will be placed on the air traffic control system to

increase sector capacity without increasing controller workload.

The subjective workload of the controller is one of the defining elements of sector capacity. The controller's workload is difficult to quantify and cannot simply be defined as a set of tasks. However, one of the most influential factors on controller workload is the complexity or degree of difficulty of the situation in which (s)he has to perform those tasks.

There are many elements that are commonly accepted as indicators of ATC complexity. For example: traffic pattern, mix of aircraft types and performance characteristics, number of flights in the sector, size and shape of sector, interface and interactions with adjacent sectors, separation standards, route crossing or convergence points, technical system limitations, restricted airspace, inadequate procedures, etc.

If we have a better understanding of what makes one area or situation more complex than another we can improve our predictions for sector capacities now and in the future.

Usual methods to evaluate sectors' capacities are of two kinds: on one hand we have microscopic models to analyse specific areas and local procedures; accurate but costly. On the other hand, there are macroscopic models, fast and easy to use but with limited accuracy.

Hence, we propose an adapted macroscopic model for each group of sectors sharing the same class of complexity indicators and subsequently similar capacities.

In this paper we will describe our method of classification to create groups of sectors with homogenous complexity. The classification process helps us to understand the similarities and differences between areas.