Genetic algorithms for optimal conflict resolution in Air Traffic

Nicolas Durand   Nicolas Alech   Jean-Marc Alliot   Marc Schoenauer

CENA*   ENSEEIHT†   IRIT‡   CMAPX§

Abstract

At the dawn of civil aviation, pilots resolved conflicts themselves because they always flew in good weather conditions with low speed aircraft. Nowadays, pilots must be helped by an air traffic controller on the ground who has a global view of the current traffic distribution in the airspace and can give indications to the pilots to avoid collisions. Solutions to conflicts are empirical, controllers are trained to react to certain types of conflicts and are limited by a workload. It is clear that if the ATC is overloaded, the sky is not. Conflict resolution is a trajectory optimization problem under constraints the complexity of which is so important that it has not been solved yet. Many attempts have been made to solve this problem with classical methods, such as gradient methods, reactive technics, expert systems, but most of them failed. In this paper, we show how genetic algorithms can be used to solve en-route aircraft conflict automatically to increase Air Traffic Control capacity in high density areas. Our main purpose is to find out the global optimum and not only a suitable solution, in a real time situation, with conflict free trajectories that respect both aircraft and pilot performances.

1 Introduction

The CENA is the institute in charge of studies and research for improving the French ATC systems. Studies on the use of genetic algorithms for conflict resolution and air space sectoring have given encouraging results [AGS93, DASF94a, DASF94b], and a new study has been funded to solve the conflict resolution for En Route Air Traffic Control. This paper summarizes the results of this study.

As there are lot of aircraft simultaneously in flight in the sky, a single controller is not able to manage all of them. So, airspace is partitioned into different sectors, each of them being assigned to a controller. Despite the traffic increase, aircraft conflict resolution is still done manually by controllers supervising control sectors. Solutions to conflicts are empirical, controllers are trained to react to certain types of conflicts and are limited by a workload. It is clear that if the ATC is overloaded, the sky is not. Conflict resolution is a trajectory optimization problem under constraints the complexity of which is so important that it has not been solved yet.

Many attempts have been made to try to reach one of the two objectives, automation and optimization, on which is this paper based. However, these two objectives must not be confused.

- AERA 3 [NFC+83, Nie89b, Nie89a] considers optimum results in the "Gentle-Strict" function for a two aircraft conflict resolution, but the "Manœuver Option Manager" only seeks for acceptable solutions and does not focus on the optimum.
- Karim Zeghal [Zeg93], with reactive technics for avoidance gives a solution to the problem of automation robust to disturbance, but completely disregards optimization. Furthermore,

*Centre d'Etudes de la Navigation Aérienne

†Ecole Nationale Supérieure d'Electronique, d'Electrotechnique, d'Informatique et d'Hydraulique de Toulouse

‡Institut de Recherche en Informatique de Toulouse

§Centre de Mathématiques appliquées de l'X the modelling adopted supposes a complete automation of both on board and on ground system.

- ARC 2000 [K+89, FMT93] optimizes aircraft trajectories when planning them without modifying already planned trajectories, global optimum is not reached.

The most important difficulty is to reach the following goals respecting realistic constraints :

- Constraints : Conflict free trajectories must respect both aircraft and pilot performances. Considering the evolution of ATC toward automation, trajectories must remain simple for controllers to describe as well as for pilots to understand and follow. Conflicts must be resolved, if possible horizontally for comfort and economical reasons.

- Goals : Conflict free trajectories must be as close as possible to optimal trajectories. They must remain sufficiently simple to allow extension to conflicts involving many aircraft and to be computed in a real time situation. Our main purpose is to find out the global optimum and not only a suitable solution.

To reach these goals and respect the constraints we established, we will first have to choose an adequate model, which had not been done in the first study in this field [DASF94a]. Then we will show why classical optimization tools such as gradient methods cannot solve our problem. After introducing Genetic Algorithm methods, we will show how we can solve complex aircraft conflicts very quickly.

2 Modelling

2.1 Mathematical study

Lets consider two aircraft x and y flying at constant altitude and constant speed. Aircraft x and y are located by their position vector $\vec{x}(t)$ et $\vec{y}(t)$. At $t = t_0$ aircraft x and y are at $\vec{x}_0$ and $\vec{y}_0$. At $t = t_f$ aircraft would like to reach $\vec{x}_f$ and $\vec{y}_f$. At each t, the two aircraft must be horizontally distant one another from a separation norm d. Assume $\dot{\vec{x}} = \vec{u}$ and $\dot{\vec{y}} = \vec{v}$. We would like to minimize the increase of the trajectories length that is :

$$J = (\vec{x}_f - \vec{x}(t_f))^2 + (\vec{y}_f - \vec{y}(t_f))^2$$

under the following constraints :

$$\begin{aligned} ||\dot{\vec{x}}|| &= C_x \\ ||\dot{\vec{y}}|| &= C_y \\ ||\vec{x} - \vec{y}|| &\geq d \end{aligned}$$

Optimal Command Theory with State Constraints developed by Bryson and Ho [BH75], supplementary conditions exposed by Kreindler [E.K82], Bryson, Denham and Dreyfus [JWS63] leads to the following conclusions : At the optimum, as long as the norm separation constraint is not saturated, aircraft fly in straight line. When saturating, aircraft start turning, and as soon as the separation constraint is freed aircraft fly straight again. This result can easily be extended to the case of n aircraft, with $n \geq 2$. When moving only one aircraft, it can be proved that trajectories are regular, they do not include discontinuous points and the minimum increase of the trajectory can be implicitly found in an equation. Numerical resolution of this equation shows that the length of the conflict free trajectory increases all the more because :

- the angle of incidence between the two aircraft decreases.

- the speed ratio is close to one.

- aircraft are getting closer to the conflict point.

It can also be mathematically proved, that if aircraft parameters (speed and heading) are constant at intervals, and if aircraft trajectories don't loop, the set of conflict free trajectories has two connected components. In one of the two sets, one of the aircraft always lets the other one on its right side, whereas in the other set, it lets it on its left side.

More details can be found in [Dur94].

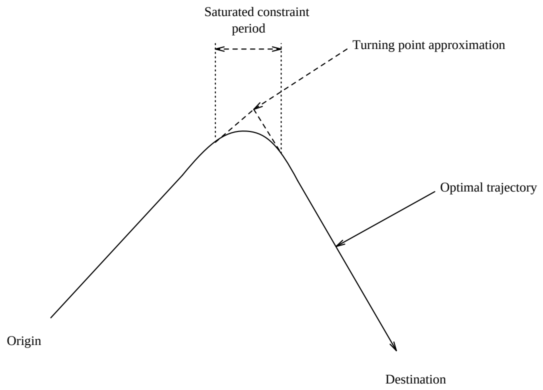

Figure 1: Turning point approximation.

2.2 The turning point modelling

The previous mathematical study leads naturally to approximate the conflict free trajectories by a turning point trajectory (figure 1). When only one of the two aircraft turns, it has been shown that the turning point approximation lengthens the optimal trajectory less than 1% if distance between the aircraft and the conflict point is greater than two separation norms and the angle of incidence between trajectories greater than 30 degrees. We can also prove that the offset modelling, which moves an aircraft to put it on a parallel route, is worse. It has the only advantage to linearize the separation constraints. The offset is thus very easy to calculate, but separation constraints must be checked during manœuvers and the complexity of the problem remains. Moreover, the offset modelling requires one more maœuver for the pilot. The turning point modelling reduces to $\Re^2$ the space solution set, which was a set of functions defined over a time interval. The whole information of a trajectory can thus be described by the coordinates of the turning point. In case this modelling was not sufficient for more complex conflicts, we could extend it by introducing more than one turning point per trajectory.

2.3 The third dimension extension

The third dimension extension comes from the aircraft piloting constraints. When changing an aircraft level, we may not decide its climbing rate or descending rate for technical reasons. The only thing we may choose is the beginning of vertical evolution, the level reached and the length of the vertical offset. For practical reasons, we may not change the flight level of an aircraft already turned off or turn off an aircraft changing its level. The ENAC modelling gives us the flying parameters for many different types of aircraft.

2.4 Modelling uncertainties

Nowadays, aircraft are able to follow a route at a given altitude, but their speed cannot be precisely anticipated. In this paper, we will assume that there is an error about the aircraft location. Consequently, controllers prefer waiting to be sure two aircraft are involved in a conflict before solving it. For our modelling, we must take this into account. The conflict free trajectory must be robust to these uncertainties. When climbing or descending, aircraft follow climbing rates and descending rates that are fixed by technical constraints and on which uncertainties are more important. To take the uncertainty problem into account, and to be able to insure an aircraft a conflict free trajectory for the next 20 minutes for example, we will represent the aircraft by a point at the initial time. The point will move to a segment of a line in the uncertainty direction, the speed direction in our case. Checking the separation norm at time t amounts to saying that the two segments modelling the aircraft positions are separated from the separation norm (see figure3).

When generalizing to the third dimension, to remain realistic, we just extend the segment to the flight level in which the aircraft is expected. A level aircraft remains a segment whereas an aircraft climbing or descending from level a to level b will fill the whole rectangle between these two levels as shown on figure 3.

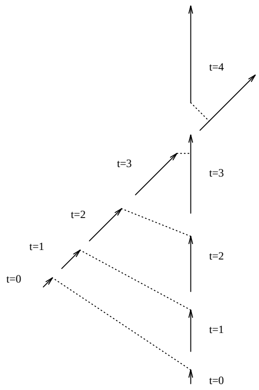

Figure 2: Modelling of speed uncertainties.

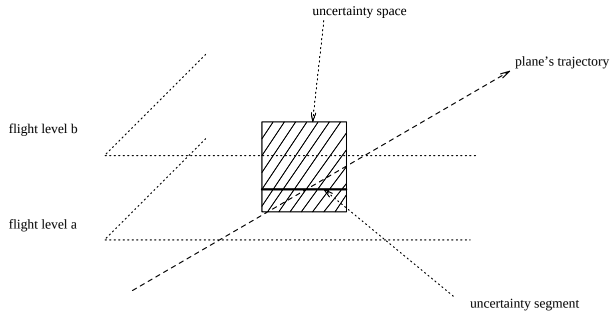

Figure 3: Modelling $3D$ uncertainties.

3 A combinatorial problem

3.1 Complexity of the problem

If we can easily prove that the minimized function is convex, the set of conflict free trajectories is not. It is not even connected. If trajectories don't loop, the set of conflict free trajectories has two connected components. For a conflict involving n aircraft there may be 2^n connected components in the free trajectory space which proves that the problem is NP. It is important to note that this complexity is independent of the modelling chosen. The offset modelling seems to be very attractive, because it linearizes constraints. Nevertheless, each constraint multiplies by two the number of linear programs to solve. Our problem involves $\frac{n\,(n-1)}{2}$ constraints. Moreover, linearizing the minimized function, multiplies by 2^n the number of linear program to solve (we minimize the sum of each aircraft offset which may be positive or negative). Finally, we will have to solve $2^{\frac{n\,(n+1)}{2}}$ linear programs, each one involving $\frac{n\,(n+1)}{2}$ linear constraints. For $n = 5$, we have 32768 linear programs to solve and 15 constraints in each program.

3.2 A global optimization problem

Using classical methods, such as gradient methods for example, becomes useless for our problem, because of the arbitrary choice of the starting point required by these methods. Each connected component may contain one or several local optima, and we can easily understand that the choice of the starting point in one of these components cannot lead by a classical method to an optimum in another component. We can thus expect only a local optimum. Practical attempts done on LANCELOT *Large And Nonlinear Constrained Extended Lagrangian Optimization Techniques* [CGT92] have confirmed this problem and high lighted others. Convergence is very slow, particularly when introducing the speed constraint. Solving a five aircraft conflict without vertical offset takes a week on a Sun work station. This approach was abandoned. The combinatory induced by the offset modelling is so important that we cannot expect to find the global optimum efficiently. Moreover, the separation during manœuvers must be checked afterwards. For these reasons, classical methods were abandoned.

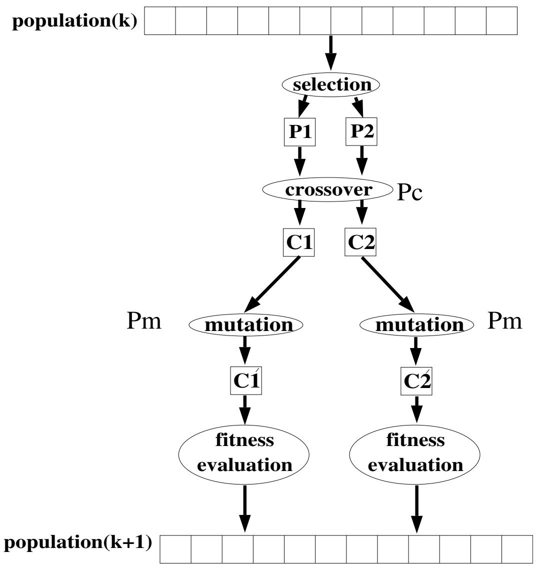

Figure 4: GA principle

4 Genetic Algorithms

4.1 Principles

We are using classical Genetic Algorithms and Evolutionary Computation principles such as described in the literature [Gol89, Mic92]; Figure 4 describe the main steps of GAs.

First a population of points in the state space is randomly generated. Then, we compute for each population element the value of the function to optimize, which is the *fitness*. In a second step we select[1] the best individuals in the population according to their fitness. Afterwards, we randomly apply classical operators of crossover and mutation to diversify the population (they are applied with respective probabilities P_c and P_m). At this step a new population has been created and we apply the process again in an iterative way.

This GA can be improved by including a Simulated Annealing process after applying the operators [MG92]. For example, after applying the crossover operator, we have four individuals (two parents $P1$,$P2$ and two children $C1$,$C2$) with their respective fitness. Afterwards, those four individuals compete in a tournament. The two winners are then inserted in the next generation. The the winners are selected as follows : if $C1$ is better than $P1$ then $C1$ is selected. Else $C1$ will be selected according to a probability which decreases with the generation number. At the beginning of the simulation, $C1$ has a probability of 0.5 to be selected even if its fitness is lesser than the fitness of $P1$ and this probability decreases to 0.01 at the end of the process. A description of this algorithm[2] is given on figure 5.

Tournament selection brings some convergence theorems from the Simulated Annealing theory. On the other hand, as for Simulated Annealing, the (stochastic) convergence is ensured only when the fitness probability distribution law is stationary in each state point [AK89].

4.2 Coding our problem

As our problem was to solve conflicts between aircraft using a 3 dimensionnal representation we did not use binary chromosomes. Finding a coding represents one of the main problems while using a GA. Using a bad coding can break all the efficiency of the algorithm.

An example of chromosome is given in figure 6. This chromosome matches with other data : an Origin and a Destination point for each aircraft. The initialization of the population creates 100 or 200 chromosomes. The parameters are initialized as follows :

[1] Selection aims at reproducing better individuals according to their fitness. We tried two kinds of selection process, Roulette Wheel Selection" and "Stochastic Remainder Without Replacement Selection", the last one always works out better.

[2] We are using our own GA simulator, which includes some goodies usually not available on public domain GA, such as Simulated Annealing, very simple parallelism, etc.

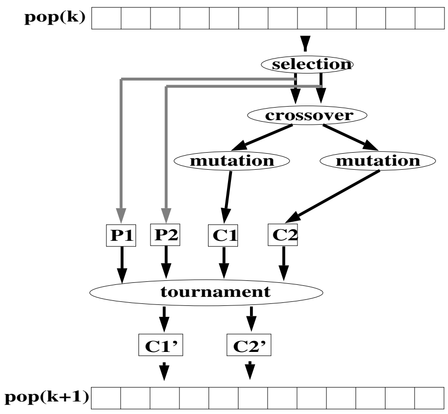

Figure 5: GA and SA mixed up

- Vertical Evolution
  This parameter can be :
  1. **GOES_DOWN**
  2. **STAYS**
  3. **GOES_UP**

  The value is set at random.
- Heading[3]
  We have the initial heading of the aircraft, we set a new one using a random number generator. We add to the initial value the maximum turning angle multiplied by a random number.
- Time[4]
  Stands for the time the aircraft will fly following the previously specified heading. After that the aircraft will turn and head to its Destination point.
- Verticaltime[5]
  This represents the duration of the Vertical Evolution, if there is one.
- Leveltime[6]
  This represents the duration the aircraft will fly at the same level, after its vertical evolution. Then the aircraft may have to change once again its flight level, for instance if it has been climbing too much it may need to climb down for a while.
- Consumption of fuel
  This value is updated during the flight. We will talk about it later.

A aircraft which does not remain at the same level during its flight is not changing its heading, this is easy to understand for practical reasons. A pilot will not accept changing its altitude and also its heading, for the comfort of the passengers and also because of his own lazyness.

A problem is given by a set of aircraft entering an aerian space in which conflicts are going to occurr. We have for each aircraft :

1. Its type.
   For instance a Boeing 747[7]

[3] For a aircraft whose vertical evolution is STAYS
[4] idem
[5] For a aircraft which goes up or down
[6] idem
[7] This is not something useless. We are using a realistic model to make our aircraft fly, this model is also based on the characteristics of the aircraft, which are read from a specified file

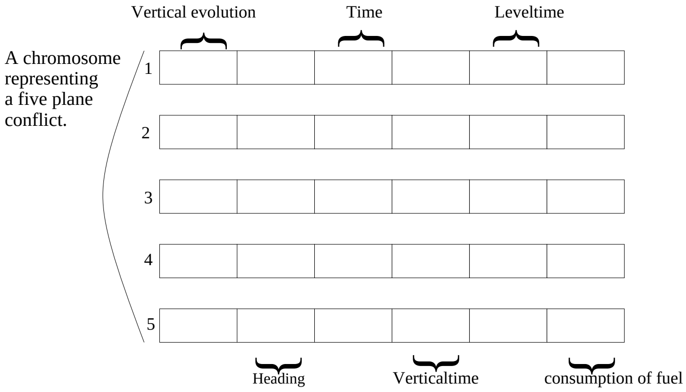

Figure 6: Structure of the chromosome

2. Its position
   - Its latitude, in degrees.
   - Its longitude, in degrees.
   - Its Altidude, in feet.
3. Its heading, in degrees.
4. Its destination point altitude.[8]

Other global data is given :

1. The duration of our study.
2. The time step used during the simulation.
3. The horizontal separation, in Nm.
4. The vertical separation, in Nm.
5. The percentage error about aircraft speed.[9]
6. The maximum turning angle, in degrees.

The main issue was to know how we were going to compute the *fitness* of a chromosome. We have a poly criteria problem to solve, in fact the following criteria have to be matched together to give us a single fitness function :

- A difference of flight level at the end of the simulation.
  For instance a aircraft was bound to reach flight level 300 and it only reached level 290.
- The distance remaining to a aircraft to fly once it has changed its heading.
- The consumption of fuel induced by a modification of the flight level.

The difference of flight level at the end of the simulation has been heavily penalized so that each aircraft reaches its flight level destination. The solutions creating conflicts have not been eliminated, but they are heavily penalized, because if we want the GA to work we have to let it reproduce, cross and mutate chromosomes. Chromosomes creating conflicts are often near a suitable solution so they have to survive, for a while, among the population. The problem which occured when introducing the third dimension was to evaluate, maybe not with accuracy but with efficiency, the cost of a vertical evolution, the algorithm must try to keep the aircraft as much as possible at the same level because of the problems caused by a flight level modification :

[8] We only need to know the altitude of the destination point because the other coordinates are computed by our model using the heading of the aircraft and its characteristics.

[9] There is always an error, we can not spot a aircraft exactly.

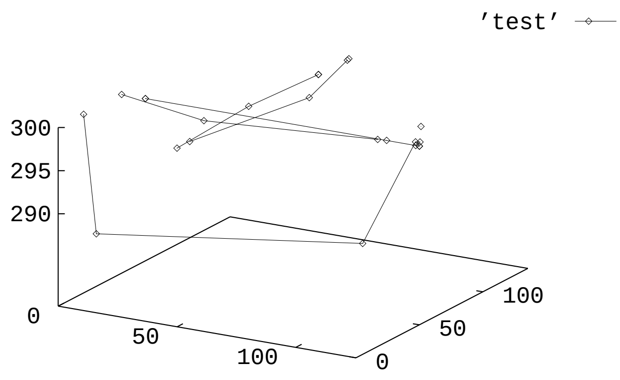

Figure 7: Efficiency of a flight level modification a.

- Consumption of fuel
- Discomfort for the passengers
- Difficulty to know, with accuracy, how the aircraft climbs

The consumption of fuel has been computed as follows :

1. A aircraft staying at the same level, as a aircraft going down, does not increase its consumption.
2. A aircraft going up increases its consumption by a fixed value.
3. A aircraft going up and then going down, as a aircraft going down and then going up, increases its consumption by twice the normal value, in order to penalize that kind of solution which implies useless changes of altitude.

In the final fitness evaluation we use the square of the consumption average. The average consumption is computed as shown below :

$$\left(\frac{consumption * timestep}{totaltime}\right)^2$$

This average consumption is multipliated by a well chosen parameter, so that if we want to solve a conflict without making any aircraft change its altitude we only have to penalize the consumption of fuel more heavily, which means changing the value of the parameter, and so a flight level modification will be catastrophic for the fitness value. Then this chromosome will be quickly eliminated.

4.3 Results

We have to consider the efficiency that a flight level modification for a single aircraft can have on a conflict, as it is shown on figure 7. This 3-dimensional representation shows how a aircraft avoids the conflict by going down 10 levels[10].

We can also see that the now 4 aircraft and bidimensional conflict is very well solved, each aircraft almost reaches its destination point, figure 8 is the bi-dimensional representation of the conflict. This is a perfect example of the interest of having a aircraft climbing down in order to solve the conflict more easily. Moreover the flight level modification is done with efficiency, we

[10] 10 levels represent 1000 Feet, which was precisely the vertical separation used for the simulation.

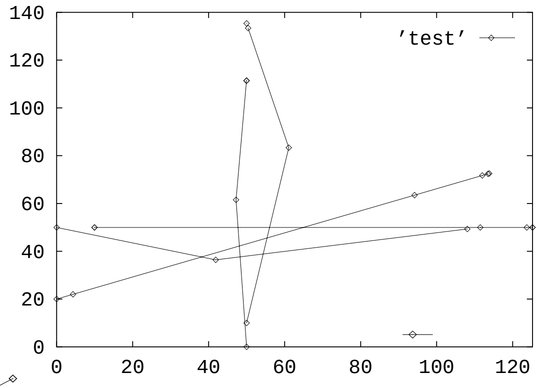

Figure 8: Efficiency of a flight level modification b.

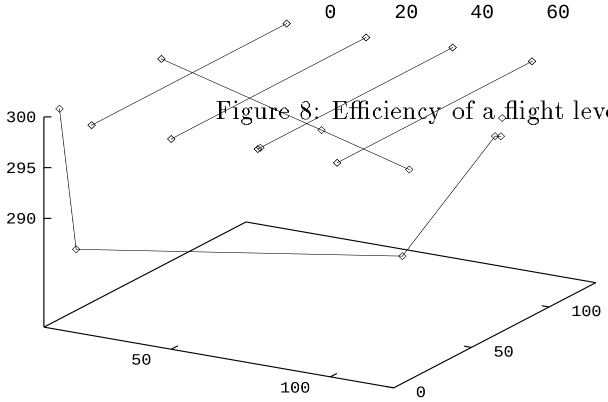

Figure 9: Artificial conflict result.

can see that the aircraft avoids the conflict as closely as possible. It only modifies its altitude to avoid a conflict.

We can give another exemple : 6 aircraft flying in an artificial way. 2 aircraft flying North-South, 2 other ones flying South-North and the 2 last ones crossing all the trajectories East-West and West-East with heading of 30 degrees. The result is quite obvious : one aircraft goes down in order to avoid the conflict, and the other ones are going straight, without changing their headings. This is figure 9.

We can give the results of the algorithm : the *average* and *best fitness* evolution these are figures 10 and 11.

5 Conclusion

We mentionned before the fact that we were using a realistic model to make our aircraft fly, this model is able to compute many parameters such as wind speed, characteristics of the aircraft[11] , ability to change its flight level. It also computes TAS[12] , CAS[13], GS[14], and all the parameters which are necessary to make an aircraft fly. It is important to have this realistic model working for us because it is a first step in the resolution of real conflicts with real aircraft. Our next purpose is to foresee a real conflict happening during a traffic day, and to solve it in real time to make the aircraft avoid each other when the conflict happens. This will be possible precisely because we are using our realistic model, an aircraft is represented in the chromosome and we are able to update our aircraft with a chromosome. The GA works on chromosomes and we use these chromosomes to make real[15] aircraft fly. Moreover the model is the same that is used by another tool we have which make aircraft fly using real flight plans.

[11] For instance we cannot set the rate of an aircraft it is automatically set by the model.
[12] True Air Speed
[13] Calibrated Air Speed
[14] Ground Speed
[15] real for our simulation

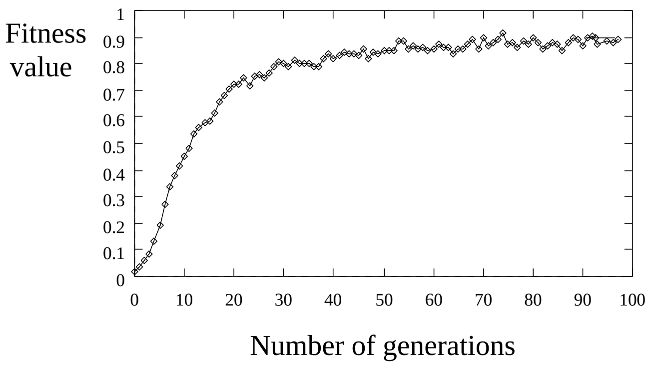

Figure 10: Average fitness evolution.

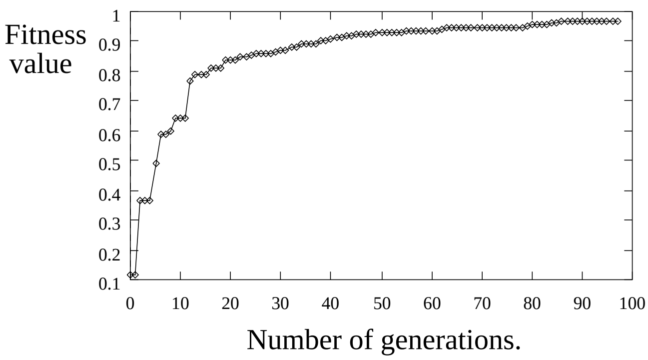

Figure 11: Best fitness evolution.

The final step will be to mix GAs and this tool completely, to have a simulation making the aircraft fly following real flight plans and avoiding each other by means of GAs.

References

[AGS93] Jean-Marc Alliot, Hervé Gruber, and Marc Schoenauer. Using genetic algorithms for solving ATC conflicts. In *Proceedings of the Ninth IEEE Conference on Artificial Intelligence Application*. IEEE, 1993.

[AK89] Emile Aarts and Jan Korst. *Simulated annealing and Boltzmann machines*. Wiley and sons, 1989. ISBN: 0-471-92146-7.

[AS92] Jean-Marc Alliot and Thomas Schiex. *Intelligence Artificielle et Informatique Théorique*. Cepadues, 1992. ISBN: 2-85428-324-4.

[BH75] Bryson and Ho. *Applied Optimal Control*. Hemisphere Publishing Corporation, New York, 1975.

[CGT92] A.R. Conn, Nick Gould, and Ph. L. Toint. A comprehensive description of LANCELOT. Technical report, IBM T.J. Watson research center, 1992. Report 91/10.

[DASF94a] Daniel Delahaye, Jean-Marc Alliot, Marc Schoenauer, and Jean-Loup Farges. Genetic algorithms for partitioning airspace. In *Proceedings of the Tenth IEEE Conference on Artificial Intelligence Application*. IEEE, 1994.

[DASF94b] Daniel Delahaye, Jean-Marc Alliot, Marc Schoenauer, and Jean-Loup Farges. Genetic algorithms for air traffic. In *Proceedings of the CAIA 94 Conference on Artificial Intelligence Application*. CAIA, 1994.

[Dur94] Nicolas Durand. Conflict free trajectory modelling for en route control. Technical report, Draft, January 1994.

[E.K82] E.Kreindler. Additional necessary conditions for optimal control with state-variable inequality constaints. *Journal of Optimization theory and applications*, 38(2):241–250, october 1982.

[FMT93] Xavier Fron, Bernard Maudry, and Jean-Claude Tumelin. Arc 2000 : Automatic radar control. Technical report, Eurocontrol, 1993.

[Gol89] David Goldberg. *Genetic Algorithms*. Addison Wesley, 1989. ISBN: 0-201-15767-5.

[JWS63] A.E.Bryson Jr, W.F.Denham, and S.E.Dreyfus. Optimal programming problems with inequality constraints i: Necessary conditions for extremal solutions. *AIAA Journal*, 1:2544–2550, november 1963.

[K+89] Fred Krella et al. Arc 2000 scenario (version 4.3). Technical report, Eurocontrol, April 1989.

[MG92] Samir W. Mahfoud and David E. Goldberg. Parallel recombinative simulated annealing: a genetic algorithm. IlliGAL Report 92002, University of Illinois at Urbana-Champaign, 104 South Mathews Avenue Urbana IL 61801, April 1992.

[Mic92] Zbigniew Michalewicz. *Genetic algorithms+data structures=evolution programs*. Springer-Verlag, 1992. ISBN: 0-387-55387-.

[NFC+83] W.P. Niedringhaus, I. Frolow, J.C. Corbin, A.H. Gisch, N.J. Taber, and F.H. Leiber. Automated En Route Air Traffic Control Algorithmic Specifications: Flight Plan Conflict Probe. Technical report, FAA, 1983. DOT/FAA/ES-83/6.

[Nie89a] W.P. Niedringhaus. Automated planning function for AERA3: Manoeuver Option Manager. Technical report, FAA, 1989. DOT/FAA/DS-89/21.

[Nie89b] W.P. Niedringhaus. A mathematical formulation for planning automated aircraft separation for AERA3. Technical report, FAA, 1989. DOT/FAA/DS-89/20.

[Zeg93] Karim Zeghal. Techniques réactives pour l'évitement. Technical report, ONERA, June 1993.