Construction of Conflict−Free Routes for Aircraft in Case of Free−Routing with Genetic Algorithms.

Ingrid Gerdes,
German Aerospace Research Establishment,
Institute for Flight Guidance,
Lilienthalplatz 7, 38108 Braunschweig,
Germany. Tel: (49) 531−295−2279,
email: ingrid.gerdes@dlr.de

1. Introduction:

Due to the continuously increasing air traffic, America and Europe will face serious congestion problems in the airspace over the next decades. To relieve the congestion it will be necessary to think not only on new equipment for aircraft but also on a new structure for the airspace instead of sectors and actual operational procedures (e.g. routes and semi circular cruising levels).

With the rising problems in mind a new idea evolved in the near past: Free−flight for en−route traffic. This would allow to use the whole airspace and therefore to increase the capacity of the airspace system. It would not be necessary any longer to force the use of the prescribed routing system and to leave the space between the routes unused. The main problems of free−flight are the construction of optimal conflict−free routes between given starting and destination points and the control of aircraft movements by the airspace controllers.

A route planning assistance tool for the controller would be helpful to cope with these problems. Otherwise it would be nearly impossible to guarantee conflict free **and** short routes in an overcrowded area. The development of such a new tool may be based on genetic algorithms. The theory of genetic algorithms deals with the application of biological principles − like survival of the fittest − to populations of artificial objects, like crossing and mutation of their genetic information, reproduction and genetic drift. The model ROGENA (free ROuting with GENetic Algorithms) described here, uses a special type of genetic algorithm for the simulation of aircraft movements in the airspace and the creation of conflict−free routes.

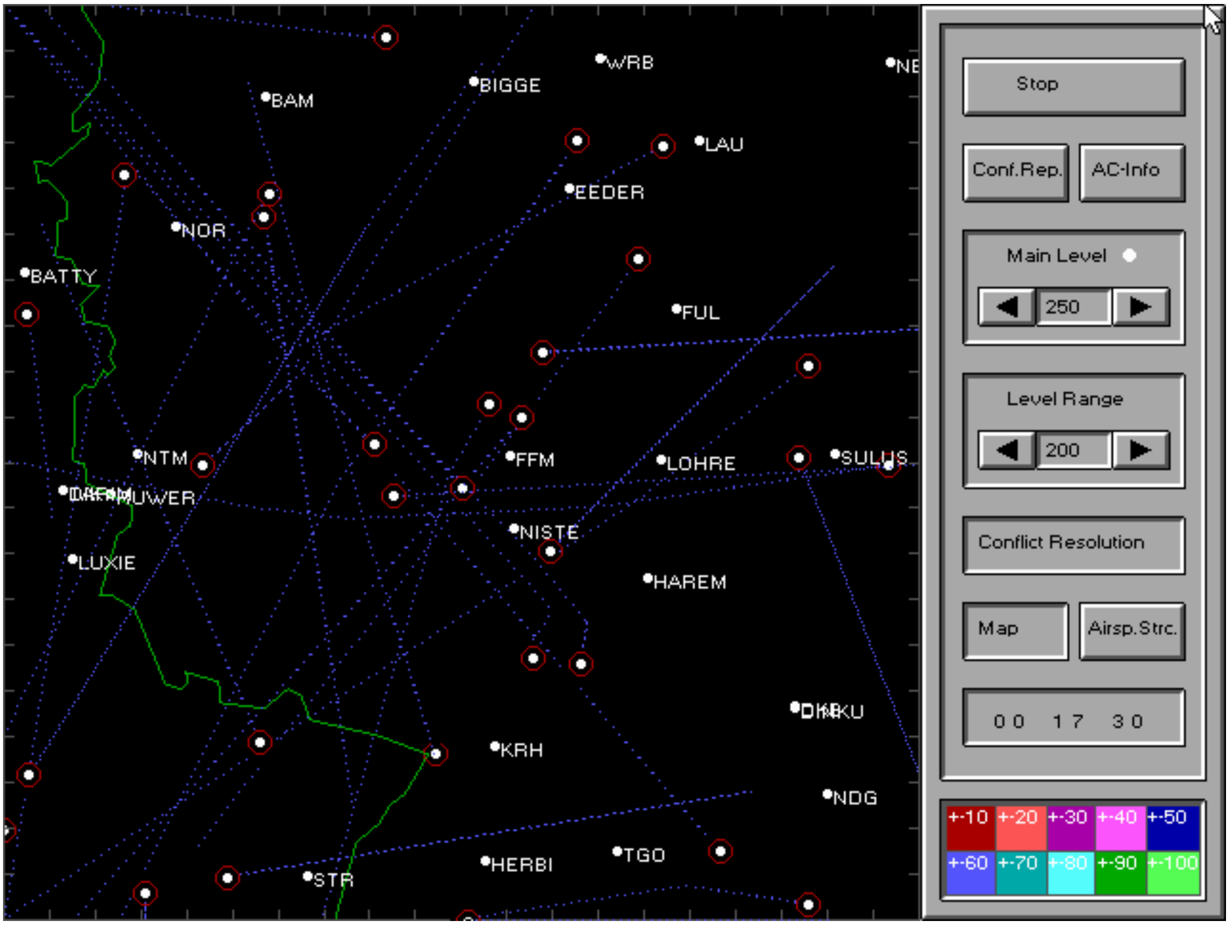

Figure 1: Main Screen of ROGENA with aircraft as white circles with half the minimum separation as red (grey) circle around them and the routes as dotted lines.

2. Free−Flight Concept

The Free−Flight Concept [5] enables a better usage of the unused airspace between the standard flight routes. In the end it shall lead to a free selection of flight levels, speeds and routes for a better self−optimisation by the pilots. This concept will produce the needed flexibility to meet the capacity demand in the foreseeable future.

Restrictions will only occur

- to ensure minimum separations,
- to preclude exceeding airport or airspace capacity,
- to prevent unauthorised flights through Special Use Airspace (SUA) and
- to ensure the safety of flights (e.g. in case of bad weather).

Although free−flight does not lead to a global optimum for all aircraft routes, it is expected to reduce the amount of time, fuel, and the overall noise level due to a better distribution of noise.

Information on the position and on the short term intent will be transmitted by the pilots to the air traffic service providers and to other nearby aircraft to ensure an acceptable level of safety. The controllers then will intervene to resolve predicted conflicts. Due to a lack of prescribed routes new types of minimum separation criteria have to be introduced: Protected zone and alert zone (see

Figure 2). The protected zone can be described as a hockey puck with half the normal minimum separations as horizontal and vertical sizes. The shape of the alert zone is more complicated because it depends on the look ahead time, the speed and the performance parameters of the affected aircraft. Conflict avoiding actions are necessary in case of touching alert zones for two or more aircraft. If all manoeuvres for aircraft would be allowed, the alert zones would be very large. Therefore, manoeuvre limitations (rules of the road) e.g. for bank angles have to be defined.

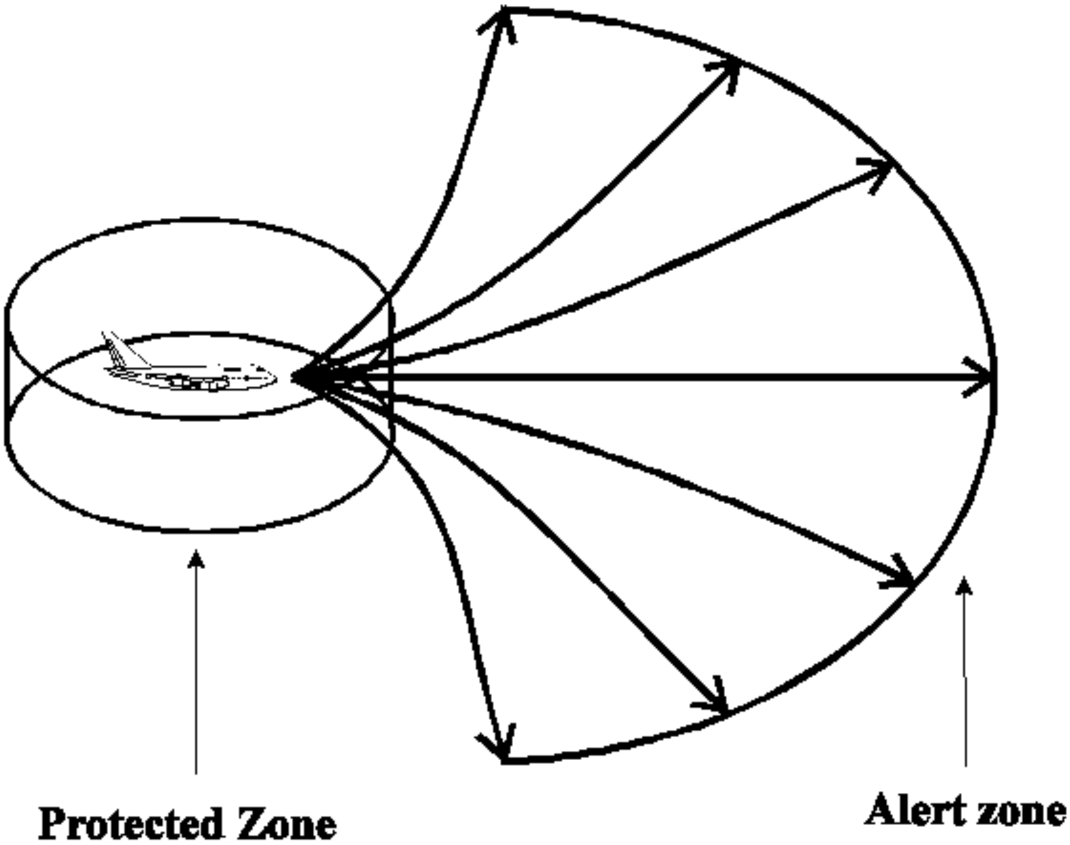

Figure 2: Free−Flight Concept with protected and alert zone.

3. Concept and Simulation Environment

3.1 General Concept

The main goal of ROGENA is to route new arriving aircraft through a given situation in the airspace (moving aircraft, restricted areas) as good as possible. Routes are described as sequence of way points whereby in the beginning the standard route for an arriving aircraft is the direct link between start and destination point. In case of separation violations new points can be inserted to avoid conflict situations with other aircraft.

Analoguous to the free−flight concept ROGENA does not create an overall optimum for all scheduled flights. In addition ROGENA assumes that the positions of all aircraft are always known. Therefore, all possible conflicts will be identified.

3.2 Formalization of Airspace and Routes

For the application of genetic algorithms it is necessary to formalize airspace and routes. Therefore, the northern part of Germany was covered by a 80 x 80−square grid with square size of 2.5 x 2.5 NM. Then all grid nodes were numbered from 1 to 6561 plus the flight level multiplied by 6561/10.

A flight route for an aircraft contains a sequence of ten so−called way points (numbered grid points) in addition to start and destination points. This sequence is the chromosome for the genetic algorithm (see Figure 3). From pevious investigations it was found that it is usually not necessary to use all way points in each route. Furthermore, the number of runs (generations) of the genetic algorithm, which is necessary to create an optimal route increases dramatically with the prescribed number of nodes. Therefore, an eleventh gene was added at the end of the chromosome containing the number of points actually used for this special route. Together with the route this number will undergo an evaluation process.

A minimum separation of 5 NM horizontally, 1000 feet below flight level 300 and 2000 feet above flight level 300 vertically is used for the conflict detection algorithm of ROGENA. For this conflict detection algorithm each route is handled as a sequence of time−dependent vectors in space. For each vector pair of different routes, position and time of the nearest approach are calculated.

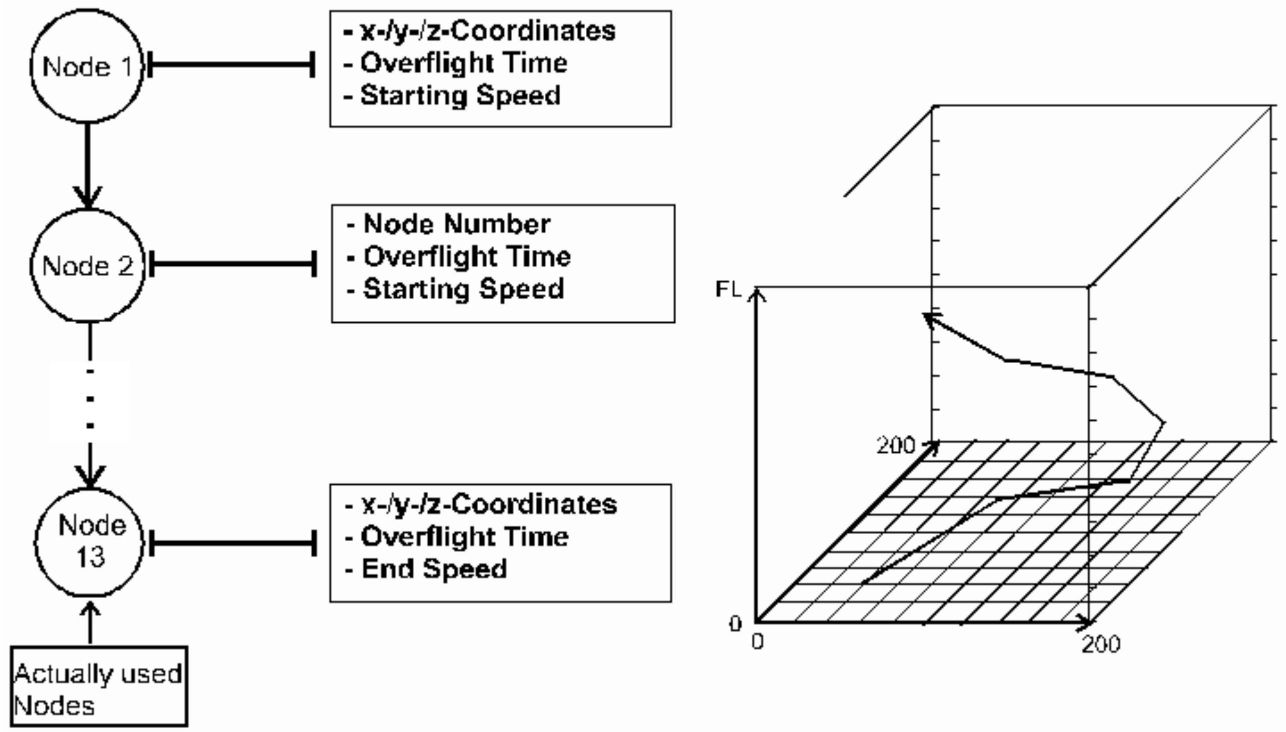

Figure 3: Formalized route.

Furthermore, for a good modelling of reality the simulation has to meet the following conditions:

1. Steady increase and decrease of the simulated flight speed by calculating the increase or decrease per time unit for the actual part of the route.
2. Avoidance of restricted areas because of bad weather, military area etc. by calculating the crossing points between the routes and the bounding polygons of the restricted areas.
3. Possibility of holding actions by introducing waiting times in case of two similar nodes following each other in the route sequence.

4. Usage of aircraft dependent climb and descent profiles by implementing the aircraft performance parameters and models of the BADA database (see section 3.3).

Restricted areas are implemented as polygons of points together with start and end times as well as vertical extension.

3.3 Flight Schedules and Aircraft Performance

This simulation is driven by a flight schedule which contains the necessary data for all flights within the simulation run. Included is information on the x−, y− and z−co−ordinates of start− and destination points, way points, starting time, speeds, climb and descent rates, and a number as characteristic of the general importance (preference value) of the flight, e.g. an airbus will have a higher preference type than a small general aviation aircraft.

Climb− and descent−rates are calculated on the basis of the BADA−database [4] in dependence of the aircraft type. This preference value is important for replanning of flights.

3.4 Controller Influence

The tool ROGENA can be used in two different ways: Automated mode and interactive mode with controller influence. In case of the second option the controller has the possibility to decide from his point of view whether the ROGENA−generated conflict−free route is good enough or not (Accept/Reject).

Furthermore, he can temporarily remove an aircraft from the simulation, which has conflicts with the new one (Reassignment). Then the system is checked for those aircraft which have a conflict with the direct link for the new aircraft. If there are aircraft with a lower preference value than the new aircraft, they are shown to the controller in the order of the lowest preference value and the highest number of conflicts. This allows the controller to select an appropriate aircraft. It gets a new route after a conflict−free route for the new aircraft is found. Routes for aircraft which are already moving in the observed sector are normally not automatically changed by ROGENA until the controller decided otherwise.

4. Route Planning with ROGENA

4.1 Planning Process

Each time when a new aircraft enters the simulation the following process starts:

> Step 1: Check of the direct link between start and destination points for conflicts with other aircraft and restricted areas. If there are no conflicts assign the direct link as flight route for the new aircraft and resume normal simulation. If there are conflicts continue with step 2.
>
> Step 2: Start of the genetic algorithm of ROGENA (GA Main). Creation of a first generation of routes between the start and destination points with way points close to the direct link. Evaluation of these routes in dependence of the length and the number of conflicts. Selection of 20 routes which will survive this run without being

changed and 40 routes which will undergo crossover or mutation (see section 4.2). Repetition of this process (selection, genetic operators, evaluation) until an appropriate conflict−free route is found (decided manually by the controller or automatically by the program). If a route is found resume normal simulation with the new route for the new aircraft. If after a certain number of runs no short and conflict free route with a good evaluation value can be found, the controller may advise the program to continue with step 3.

Step 3: Removal of another aircraft − which has conflicts with the new one − from the planning process. Creation of new routes for the new aircraft and the removed aircraft according to step 2. If there is no aircraft with a lower preference value assign the best route found in step 2 to the new aircraft and resume simulation.

4.2 Genetic Algorithm of ROGENA

The principle idea of the genetic algorithm of ROGENA is based on the modGA of Michalewicz [3]. Compared to the standard evolutionary algorithm modGA avoids the problem of premature convergence, frequently caused by the dominance of particularly fit individuals (super individuals) in the selection phase (see section 5.1).

Figure 4 shows the structure of this special algorithm. ModGA requires the selection of chromosomes for 3 special groups (see Figure 4): One for the application of the **crossover** operator, for **mutation** operators and to remain **unchanged**. In ROGENA each of these three groups consists of 20 chromosomes or routes because this number has led to the best results in the course of detailed tests.

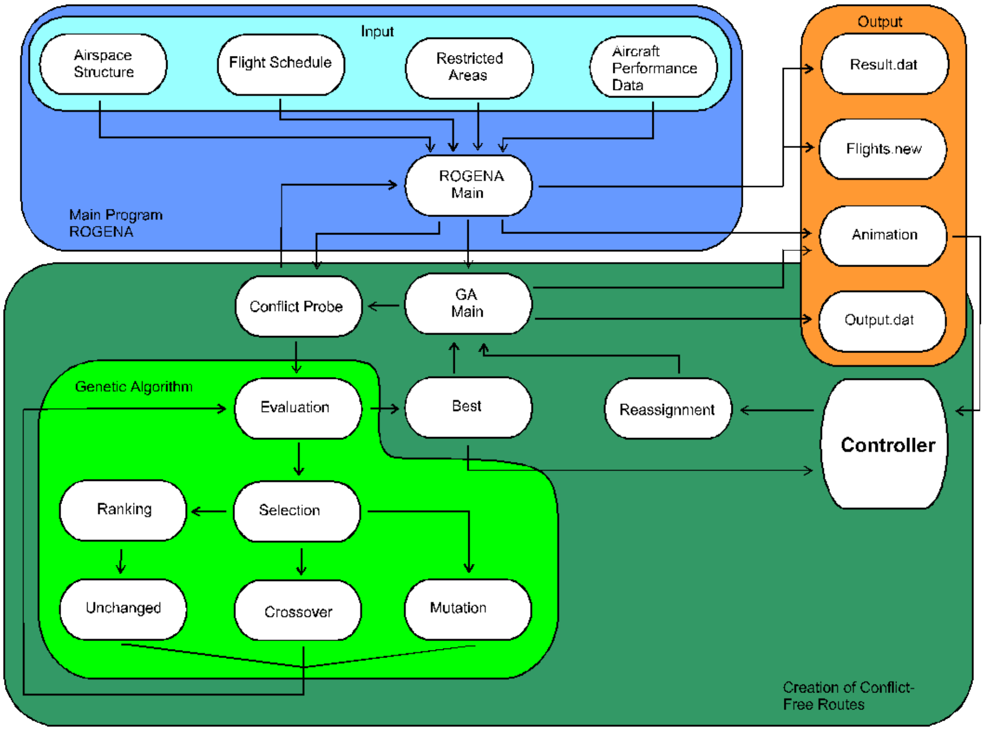

Figure 4: Structure of ROGENA.

The size of the population is set to 60 chromosomes. The information on the way points is integer coded. The co−ordinates for the start and destination points are float values.

However, problem specific modifications of the modGA have to be included for a significant improvement of performance. In case of one of these modifications, the best five routes of every generation are stored automatically for the next generation.

Another very important adaptation is the creation of the first population. Instead of creating the way points of the first routes completely by random they are arranged randomly close to the original route. Therefore, routes with a high number of way points can be as short as routes with only a few way points. Since the chance of survival (evaluation value) of a route depends mainly on its length, routes with a high number of used way points are not killed within the first population as it would be in case of complete randomly generated routes. The resulting evolutionary algorithm has proven to avoid the problem with super individuals.

For each route of the population an evaluation value is calculated which is used to assess the chance to survive the selection process and to live within the next generation of routes. It is calculated by means of an evaluation function which depends on

- the length of the route,
- the differences to the optimal climb and descent rates, and
- the differences between the *prescribed minimum separations* of the observed aircraft to all other aircraft and the *actual separations* at the time of closest approach.

Furthermore, a weight function for the lack of separation is part of the formula. It increases with the number of the generation to ensure a high number of conflict−free routes at the end.

generation n:

$$EVAL(route_{i,n}) = length(route_{i,n}) + \sum_{j=1}^{k} \max(0, minsep - dist(route_{i,n}, route_{ac\,j})) * weight(n) +$$

$$\sum_{l=1}^{Lsektor} (\left| optimal\ c/d\ rate_{ac\,type,l} - actual\ c/d\ rate_{ac\,type,l} \right|);$$

Routes with conflicts are not automatically removed because they often have the potential to change to a good route (short and conflict−free) or to jump over a barrier of conflicts. With this definition of the evaluation function, routes which fail to fulfil the separation standards only a little bit have only a small penalising factor.

The selection process for the three groups mentioned before is mainly carried out with 'stochastic sampling with replacement' [1]. An additional condition of this type of genetic algorithm is that for the group for which the chromosomes remain *unchanged* all selected routes have to be different. This helps to prevent super individuals from taking over the population. For each generation the best five routes are stored in this group.

The operators crossover and mutation are carried out only for the chromosomes in their group and only on those way points which are a part of the actual route (see Figure 5). The used *crossover* type is two−point−crossover, but reduces to the normal one−point−crossover in cases where only one way point is in use for both involved chromosomes. Two−point crossover means that the way points of two routes between two cutting positions are exchanged. The cutting positions are selected randomly. The meaning of this is that parts of routes are exchanged.

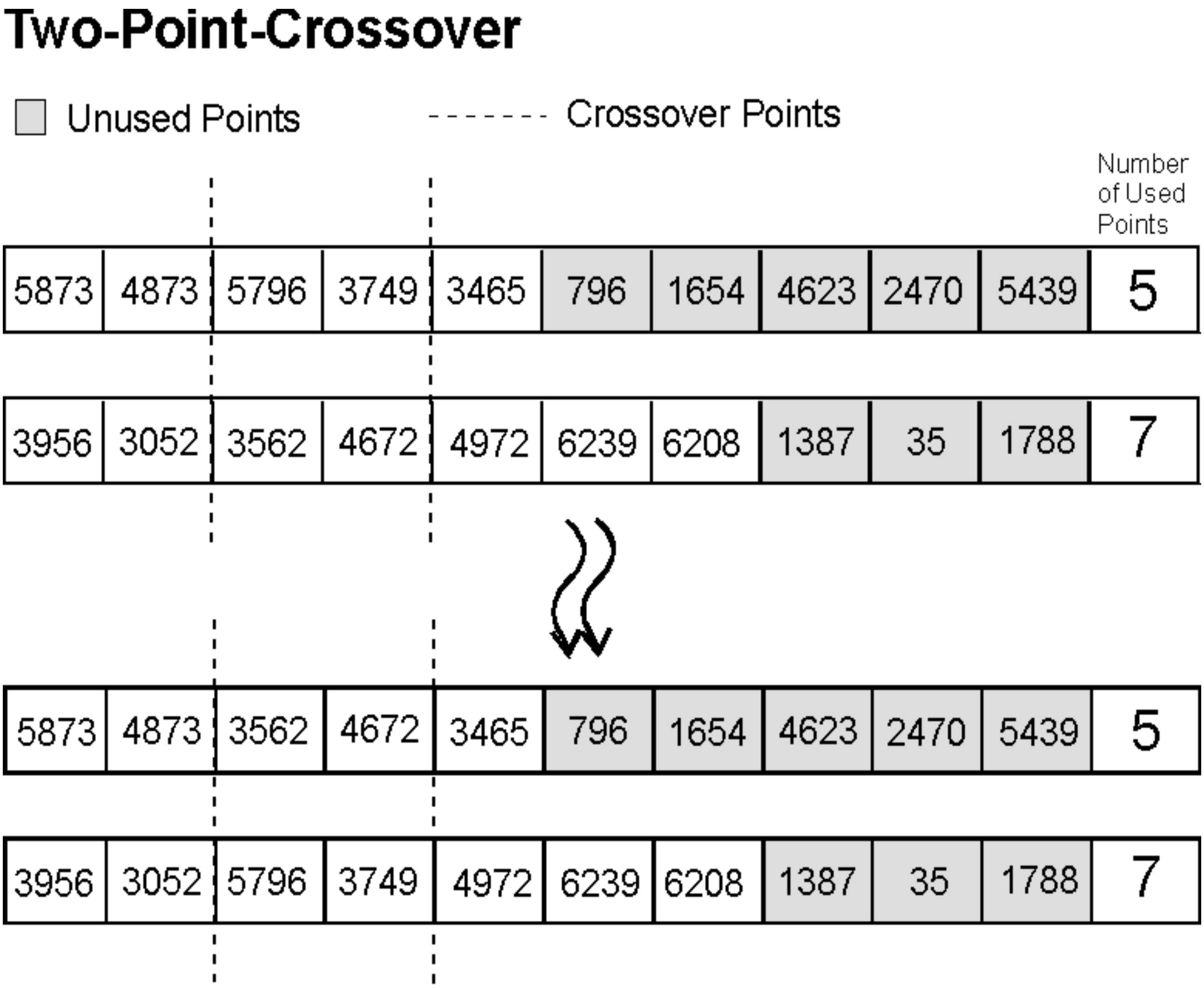

Figure 5: Example for crossover.

For having a good balance between local search in the neighbourhood of good way points and random search for finding completely different good routes, two types of *mutation* are implemented in ROGENA. The first one is similar to hillclimbing and moves the selected point to a neighbouring node. The other operator is the normal mutation where the original point is substituted by a randomly selected new one.

Furthermore, the number of actual used way points is mutated with a small probability to prevent the population from loosing routes with other node numbers than numbers of the best routes. In this case a position is selected and a way point becomes deleted or inserted at this position.

5. Comparison with other Optimisation Algorithms

5.1 Standard Genetic Algorithm

In section 4.2 it was stated that the standard genetic algorithm (stdGA) would have much more problems with super individuals and loss of genetic diversity than the modified genetic algorithm used for ROGENA. To examinine this problem, several tests were carried out.

For this test a standard algorithm was implemented into ROGENA with the same evaluation function, mutation and crossover operators as before. The difference to the modGA lies in the number of routes and way points the genetic operators were applied to.

In case of the standard algorithm each route of the 60 routes of a population has a probability to undergo the crossover operator, and each way point of every route has a probability to be mutated. By using the same probabilities as for the modGA it was not possible to create suitable results. Therefore, it was necessary to adapt the probabilities to this type of genetic algorithm.

For the comparison of both algorithms 10 test scenarios were created with one conflict for each scenario. Ten runs of each scenario were carried out and the results shown in Table 1 are the average values of these ten runs. For this and all other types of scenarios described in this paper no reassignment was used, i.e. there were no unsolvable conflicts within all scenarios.

The results for the modGA are only slightly better than for the standard Genetic Algorithm. This shows that the parameters for stdGA were carefully adjusted.

Average Length (NM) for Scenario										
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
modGA	93.13	139.31	156.50	154.90	141.83	178.01	210.06	51.72	202.24	115.95
stdGA	93.29	139.35	156.59	155.10	141.83	178.3	210.06	52.21	202.63	116.02

Table 1: Comparison of modGA and stdGA.

In spite of this the results for the average length of the whole population are very different. An example for this is shown in Figure 6 and Figure 7. The curves consists of the values of the best runs for scenario 9. It is easy to see that the population of the modGA is much more inconsistent and includes a lot of different and sometimes bad routes. The results for the stdGA are very close to the best route and the standard deviation is very small. This shows the lack of divergence within the population of stdGA. In case of deceptive problems − where normally suboptimal routes are found first − the stdGA will not be able to find his way back to the best route because the whole population consists of almost similar routes.

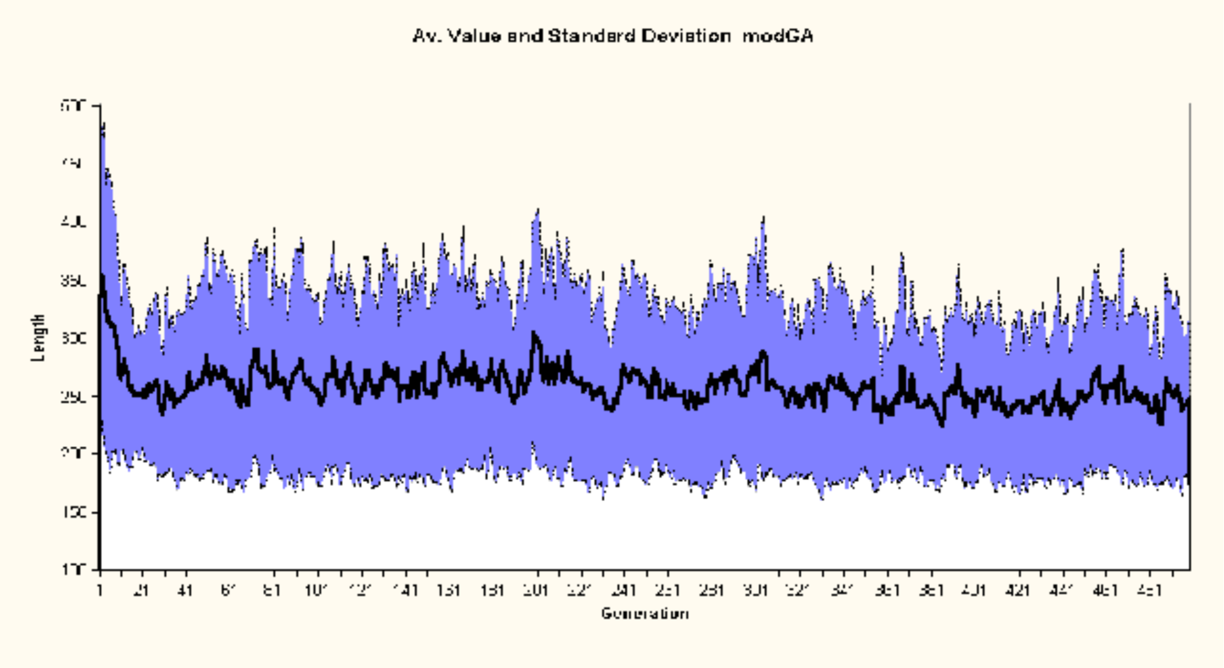

Figure 6: Average route length and standard deviation for modGA.

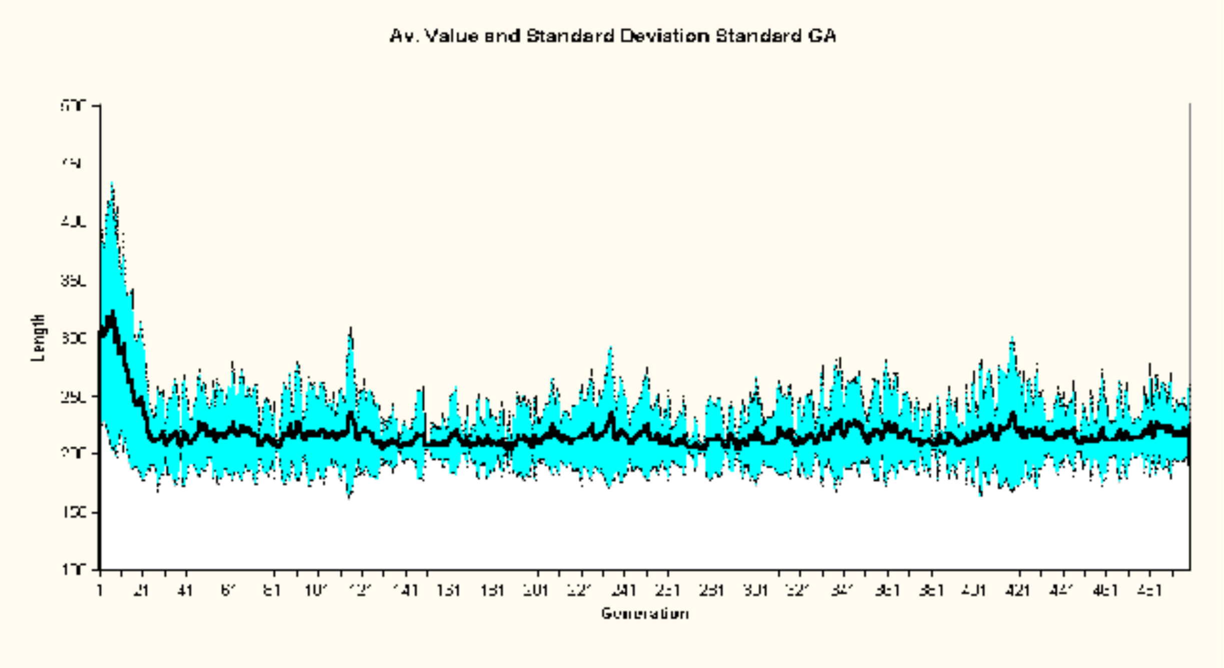

Figure 7: Average route length and standard deviation for stdGA.

5.2 Hillclimbing and Simulated Annealing

As example for other optimisation algorithms hillclimbing and simulated annealing [2] were selected. All simulations were carried out with a flight schedule of 40 flights and an average of 17 conflicts. For both algorithms six simulation runs were performed. The algorithms have according to their definition a population size of one route. In order to compare hillclimbing and simulated annealing with the implementation of the genetic algorithm 40 different starting routes for each conflict were generated for both algorithms. For each of these various starting routes two hundred optimisation steps (generations) were conducted. For each step one way point of the route was randomly selected and then changed to one of the neighbouring eight way points. If this led to an improvement, the old way point was replaced by the new one, otherwise in case of hill−climbing not. Simulated annealing has, in dependence of the number of already accepted non−improvements, a certain probability to accept the worse route. The results of hill−climbing and simulated annealing algorithm were compared with the results of the ROGENA algorithm explained before.

First tests demonstrated, that also for hill−climbing and simulated annealing it would be necessary to allow for a mutation the number of way points. Because each complete simulation of these two algorithms is done with 40 different starting routes, there will be otherwise an average of only four routes with the correct number of way points for the creation of the optimal route.

The results of the tests are shown in Table 2. The column 'Distance Ratio of Type to Direct' shows the value of the ratio of the length of the conflict−free routes created by ROGENA / hill−climbing / simulated annealing to the length of the direct link between start and destination point. The other columns show the average increase in the route length per aircraft and per conflict.

It can easily be observed that the values for hill−climbing in column three and four are more than four times higher than the values for the genetic algorithm, and two point five times higher in comparison to the results of simulated annealing. Furthermore, one of the simulation runs with hill−climbing was not able to solve all conflicts. Therefore, the value for hill−climbing is obtained by averaging the data of 5 simulation runs.

Type	**Distance Ratio of Type to Direct**	**Average Increase [NM/Aircraft]**	**Average Increase [NM/Conflict]**
Hill−climbing	1.02230	3.47	8.49
Simulated Annealing	1.00897	1.39	3.28
ROGENA	1.00504	0.78	1.75

Table 2: Comparison between hill−climbing, simulated annealing and the genetic algorithm of ROGENA.

5.3 Combinations of Evolutionary Algorithms with Hillclimbing

Two hybrid systems of evolutionary algorithms with a hillclimbing algorithm were tested, one with two versions:

1. the first version has a hillclimbing algorithm with 200 runs for each route in the beginning and then 200 generations with the evolutionary algorithm,
2. the second has the inverse order and
3. the third is like the second but with only 100 runs for the hillclimbing algorithm and 300 generations with the genetic algorithm.

The first version was carried out to test whether a better starting set could improve the results of the genetic algorithm. The other two versions should adjust the created routes in the best way around the conflicts. The results from Table 3 have shown that a number of 200 runs with the evolutionary algorithm is not enough for getting a good starting set for the hillclimbing algorithm.

	Average Number of Nodes (Best Route)	Average Increase of Route Length	Length of New Routes (SUM)
Version 1	1.49	2.40 NM	2013.28 NM
Version 2	1.55	2.19 NM	2010.56 NM
Version 3	1.59	1.99 NM	2007.85 NM
ROGENA	1.69	1.88 NM	2006.05 NM

Table 3: Results for different hybrid versions of the evolutionary algorithm of ROGENA with hillclimbing.

Furthermore, the results show that version 1 has the worst results while the results of version 2 are comparable to those of the normal ROGENA algorithm. Version 3 has results very close to those of modGA of ROGENA but is not better. Therefore, no clear argument for the introduction of a hybrid system can be found.

6. Simulations with Real Data

The tested scenario was composed of real aircraft trajectories which were extracted from collected radar data of the northern part of Germany. Each of the three flight schedules contained the data of a special flight level. Starting time for each aircraft was the actual time when the aircraft crossed the border of the grid or reached the actual flight level, respectively. Start and destination points were the positions where the aircraft entered and left the grid. The speed was average speed on the route as far as the flight was conducted within the actual flight level. The comparison was made between the measured length of the radar tracks and the routes generated by ROGENA.

The results in Table 4 show only a small number of conflicts. The 'Number of Conflicts' stands for conflict situation in case of using the direct links between start and destination points. There are no conflicts when using the standard routes and the routes of ROGENA but the routes created by ROGENA are shorter than for standard routes. This confirms the assumption that free−routing works well and would not lead to a high number of conflicts. In case of the scenario without conflicts the gain is caused by using the more direct routes of ROGENA. However, we have the highest gain in scenarios with more conflicts.

Number of :		Route Length (Sum)		ROGENA in % of Traffic	Average Decrease
A/C	**Conflicts**	**Traffic**	**ROGENA**		**(NM) per AC**
12	0	834.43	820.29	98.31	1.18
22	2	1647.70	1602.07	97.31	2.02
31	2	1747.24	1714.00	98.10	1.07

Table 4: Comparison between the length of standard routes and the more direct routes generated by ROGENA.
Traffic: Real traffic scenario.

7. References

[1] Goldberg, D.E. (1989): Genetic Algorithms in Search, Optimization and Machine Learning. Addison Wesley, Reading/Massachusetts.

[2] Heistermann, Jochen: Genetische Algorithmen. Teubner Verlag, ISBN: 3−8154−2057−1.

[3] Michalewicz, Z. (1992): Genetic Algorithms + Data Structures = Evolution Programs. Springer Verlag, Berlin.

[4] User Manual for the Base of Aircraft Data (BADA) Release 2.3, EUROCONTROL, Experimental Center.

[5] RTCA Task Force 3, Free−Flight Implementation, Draft Final Report, October 1995.