

Military Technical College
Kobry El-Kobba,
Cairo, Egypt



11-th International Conference
on Aerospace Sciences &
Aviation Technology

APPLICATION OF GENETIC ALGORITHMS WITH THE AID OF PANEL METHOD TO OPTIMIZE MISSILE AERODYNAMIC SHAPE

Eng. Mutawkil* A. Suliman

ABSTRACT

The use of genetic algorithms with the aid of the panel method to determine high-efficiency missile aerodynamic shape has been examined. The capability of these algorithms to determine highly efficient and robust missile aerodynamic designs is demonstrated by giving a variety of design variables and constraints. Results indicate that the genetic algorithm is clearly capable of designing aerodynamic shapes that perform well in either single or multiple goal applications.

KEY WORDS

Genetic algorithm, Panel method, Missile, Optimization, CFD, Aerodynamic shape

NOMENCLATURE

b: Span.

C_D, C_L : Drag and lift coefficients

CFD: Computational fluid dynamic.

C_r, C_t : Root and tip chord.

GA: Genetic Algorithm.

L: Length.

p_c : Crossover probability.

p_m : Mutation probability.

R: Caliber

X_{cg}, X_{cp} : Location of center of gravity and center of pressure from nose tip.

X_{Static} : Static margin.

TR: Taper ratio

Λ : Sweep back angle.

* Lecturer, Dpt. Of Mech. Aeronautical & Aerospace, Karary Academe of Technology, Khartoum, Sudan.

INTRODUCTION

The design of missile configurations is one of the most interesting and challenging fields "perhaps the most complex" for the aeronautical design engineer [1]. In recent years, researchers have applied gradient-based optimization schemes to aerodynamic design, but these methods are subject to undesirable restrictions. For instance, gradient-based optimizers must start with a specified set of initial parameters, which can bias future solutions toward a local optimum in the vicinity of the starting point. Gradient search procedures work most efficiently when there are a small number of design variables and when the variables are essentially independent of each other. As the number of design variables increases and coupling of the variables occurs (most often the case for complex aerodynamic designs), gradient-based algorithms do not have the ability to recombine disparate solutions to form solutions that sample a new portion of the optimization space [2].

GENETIC ALGORITHM

GA is a search optimization method, which simply works by selecting a group of solutions from a population of candidate solutions and evolving them through its main operator's crossover and mutation. So it is better than the traditional search method by its capability to evolve the candidate solution and its ability to solve both single and multi-objective (goal) problems.

Genetic algorithms have been applied to aeronautical problems in several ways, including parametric and conceptual design of aircraft, preliminary design of turbines, topological design of non-planar wing and aerodynamic design using Computational Fluid Dynamics (CFD) [3].

Encoding of Chromosomes:

The chromosome should be in some way containing information about solution, which it represents. There are many ways to encode chromosomes. In encoding of the design parameters, the number of codes (strings or solutions) available is:

$$\text{number of solutions} = m^n \dots\dots\dots (1)$$

- Where,
- m: being the number of possible alphabetic in encoding type
- n: being the number of bits (genes) used to represent a parameter.
- n = 2 for binary, 10 for Permutation, and 26 for English alphabetic.

Binary encoding

Binary encoding is the most common, mainly because first works about GA used this type of encoding. In *binary encoding* every chromosome is a string of bits, 0 or 1 as shown below.

Chromosome A	101100101100101011
Chromosome B	111111100000110000

Binary encoding gives many possible chromosomes even with a small number of alleles. On the other hand, this encoding is often not natural for many problems and sometimes corrections must be made after crossover and/or mutation [4].

GA Operators

1. Crossover:

After encoding crossover can be done. Crossover selects parent of chromosomes and creates a new offspring. There are many ways in how to make crossover, for example single or multi crossover points.

Chromosome 1	11011 0010011 0110
Chromosome 2	11011 1100001 1110
Offspring 1	11011 1100001 0110
Offspring 2	11011 0010011 1110

Crossover is made in hope that the new offspring will have good parts of old chromosomes and may be the new chromosomes will be better. However it is good to leave some part of population survive to next generation. Crossover can be rather complicated and very depends on the encoding of chromosome. Specific crossover made for a specific problem can improve performance of the genetic algorithm.

Single point crossover

It is the simplest way. In this type one crossover point is selected, binary string from beginning of chromosome to the crossover point is copied from one parent and the rest is copied from the second parent

$$11001011 + 11011111 = 11001111 \text{ \& } 11011011 \text{ (2)}$$

2. Mutation (Bit inversion):

After crossover is performed, then mutation takes place. Mutation is made to prevent falling GA into local extreme, or to prevent falling of all solutions in population into a local optimum of solved problem. Mutation changes randomly the new offspring. Mutation can then be shown as following:

Original offspring	110111100001111
Mutated offspring	110011100011111

Mutation depends on the encoding as well as the crossover.

Flip bit

A mutation operator, who is simply inverts the value of the chosen gene (0 goes to 1 and 1 goes to 0). This mutation operator can only be used for binary encoding genes.

Selection:

As we already know from above, chromosomes are selected from the population to be parents to crossover. The problem is how to select these chromosomes. According to Darwin's evolution theory the best ones should survive and create new offspring. There are many methods how to select the best chromosomes, for example roulette wheel selection, Boltzman selection, tournament selection, rank selection, steady state selection and some others.

Roulette wheel selection

In this type parents are selected according to their fitness. The better the chromosomes are, the more chances to be selected they have. Imagine a *roulette wheel* where are placed all chromosomes in the population, every one has its place big accordingly to its fitness function. Then a marble is thrown there and selects the chromosome. Chromosome with bigger fitness will be selected more times.

Elitism:

When creating new population by crossover and mutation, there will be a big chance, that the best chromosome can be loose. *Elitism* is the name of method, which first copies the best chromosome (or a few best chromosomes) to new population. Elitism can very rapidly increase performance of GA, because it prevents losing the best found solution. Surely *elitism* should be used (if other methods are not used for saving the best-found solution).

Genetic Parameters:

- a. *Crossover probability* p_c : Says how often will be crossover performed. If crossover probability is 100%, then all offspring is made by crossover. If it is 0%, whole new generation is made from exact copies of chromosomes from old population. Crossover rate generally should be high, about 80%-95% [4].
 - b. *Mutation Probability* p_m : Says how often will be parts of chromosome mutated. If mutation probability is 100%, whole chromosome is changed, if it is 0%, nothing is changed. Mutation rate should be very low. Best rates reported are about 0.5%-1% [4].
 - c. *Population Size*: It may be surprising, that very big population size usually does not improve performance of GA. Some research shows that best population size depends on encoding (*on size of encoded string*). Two times the number of genes is the best population size.
-

Termination

Termination is the criterion by which the genetic algorithm decides whether to continue searching or stop the search. Each of the enabled termination criterions is checked after each generation to see if it is time to stop. *Generation Number* is a termination method that stops the evolution when the user specified the maximum numbers of evolutions has been run. This termination method is always active [5].

PANEL CODE

The panel code, which is used here, is Woodward panel method code (Woodward code [6], is a panel code developed by Boeing aircraft company to calculate pressure distribution and aerodynamics coefficient for wing body combination by using singularity distribution on the body surface) after some modifications. In spite of that the panel method is computationally expensive it give a good results.

DESIGN VARIABLES

In this research the external shape is determined from a number of design variables, ten for wing-body, and fifteen for wing-body-tail. These variables include total length, nose length, boat-tail length, caliber, base diameter and wing/tail geometry, etc. (see table1), within design constraints to these parameters.

SIMPLE OPTIMIZATION ALGORITHM:

As shown in figure (2) the simple optimization algorithm work as follow:

- 1) Start with a randomly generated population of n candidate solution to a problem (chromosome) each chromosome consists of all design variables in a certain arrangement.
- 2) Calculate the fitness $f(x)$ of each chromosome x in the population by using panel method.
- 3) Select the best fitness chromosome as a temporary solution of the problem.
- 4) Compare the temporary solution with the previous solution and determine the correct solution.
- 5) Encoding all the design variables with their arrangement as a zeros ones string (chromosome).
- 6) Repeat the following steps until n offspring have been created.
 - a. Select a pair of parent chromosomes from the current population, the probability of selection being an increasing function of fitness.
 - b. With probability p_c cross over the selected pair.
 - c. Mutate the two offspring at each locus with probability p_m "mutation probability".

- 7) Replace the current population with the new population.
- 8) Go to step 2

RESULTS AND DISCUSSION

The results were carried out for single and multiple goal cases as shown in below.

Maximum Lift Coefficient

In this case it was tried to see whether the GA would maximize the fins areas. Fin placement does not play a significant part in the design consideration for this case. Also, because drag force is not considered for this case, the base radius can be anything and nose can be very blunt. Figure (3) graphically show the design history for the missile. In this figure, and subsequent figures of this nature, only the design that produced the highest aggregate lift force coefficient at each displayed generation is shown. Although these configurations design may make the missile look a little strange at first, one must focus on the wing/tail planform geometries because they are the most important parameters for now. From these qualitative representations of the design variable histories it is clear that the GA learned to make the wing and tail areas large. Although early attempts used small and highly swept wings, they were later replaced by straight and large wings. The missile body also grew in length during the process, which contributes to higher normal force coefficients. The nose and boat-tail length was also minimized in order to increase the body length, further helping to maximize the normal force coefficient.

Convergence history is important to the understanding of the behavior of a GA. Figure (4), show the convergence history for this case. The maximum line denotes the performance of the best single member of the population. Average denotes the performance of the average member of the population. From this figure and subsequent figures of this nature it is clear that by using elitism the maximum solution always in progress if it is not constant.

Minimum Drag Coefficient Case:

The inclusion of a requirement for low drag force coefficients should necessitate two primary design changes over the preceding high lift coefficient case. *First*, the nose of the missile should elongate to minimize large wave drag due to a blunt nose. *Second*, the nozzle exit radius should be equal to or less than the radius of the body. It might be expected that there would be wing/tail leading-edge sweeps to forestall the sharp transonic drag rise that accompanies the transition from subsonic to supersonic flight. Also wings and tails areas should be minimize. Figure (5), show that as generations number increase there will be progress and the new generations have long nose and small sweeps fins and base radius less than or equal to body radius. Figure (6) show the convergence history for this case. In these figures the maximum fitness line denotes the inverse of the drag coefficient, so if the corresponding drag coefficient for each generation is down,

the resultant line will be start from high drag for earlier generation and decreases (minimized) as generations proceed.

Maximum Lift and Minimum Drag Coefficients Case:

This case will be optimized by, maximizing C_L and minimizing C_D . So the excepted configuration is a combination of the above two cases geometry. Drag coefficient minimization need sweep wings, but the GA might find that high normal force coefficients due to unswept wings might out weigh the delay in axial force coefficient rise that sweep would certainly produce. The objective function for this case can be:

$$f = \frac{C_L}{C_D} \dots\dots\dots (3)$$

Or when using one as the maximum value the objective function can be:

$$f = 1 - \frac{C_D}{C_L} \dots\dots\dots (4)$$

The GA design history figure (7) show that the algorithm produced good high C_L/C_D designs. As expected with the following properties:

- 1) The wings and tails became exactly those that were designed under the high C_L case (taper ratio included).
- 2) The body radius was not maximized.
- 3) The base radius became less than the body radius.
- 4) The nose grew in length to minimize bluntness (to reduce the axial force coefficient).

Figure (8) show the convergence history for this case.

High Normal Force, Low Axial Force, Minimal Static Margin Case:

This case includes a measure of the stability of the airframe in addition to the aerodynamic performance. The aerodynamic performance goal is calculated as before by calling the panel code, whereas the static margin goal, which is to minimize the static margin, is calculated as

$$X_{Static} = |X_{Cp} - X_{Cg}| \dots\dots\dots (5)$$

Where the X locations, are measured positive aft of the nose tip. And the objective function for this case can be. The genetic algorithm will attempt to find solutions that will perform reasonably well in each of the two goal areas, but it can be expected that, some of the earlier aerodynamic performance exhibited will be lost to the desire to have a minimal static margin. The objective function for this case can be:

$$f = \frac{C_l}{C_d} + \frac{1}{X_{Static}} \dots\dots\dots (6)$$

Or when using one as the maximum value the objective function can be:

$$f = 1 - 1 / \left(\frac{C_l}{C_d} + \frac{1}{X_{Static}} \right) \dots\dots\dots (7)$$

Figure (8) shows the convergence history for the maximum among the generations in each goal area. Clearly, some aerodynamic performance was sacrificed for good static margin performance. For this case several of the better solutions are obtained, and allowing the designer to choose from a variety of similar performing solutions, based perhaps on some other criteria such as manufacturing simplicity or weight. These solutions have different levels of static margin and aerodynamic performance. All of the solutions have large tails to increase C_L/C_D . This result should have been expected because large tails would serve also to help the static margin performance. Also better static margin solutions have more highly swept wings and more taper than the higher C_L/C_D solutions. Clearly, the GA learned to keep the wing areas fairly large keeping the center of pressure far enough aft to keep the static margin reasonable.

CONCLUSION:

It is clear that it is possible to couple Computational Fluid Dynamics (CFD) panel code, with Artificial Intelligence (AI) Genetic Algorithm (GA) in a single computer program. This program can predict aerodynamic (external) shapes of a wing-body or a wing-body-tail missile configuration that can perform well in either single or multiple goal applications. Also It is clear from the results that genetic algorithm always search for the optimum solution, so that the curves of maximum fitness always in progress until it reach its global maximum value.

REFERENCES:

- [1] S. S. Chin: "Missile Configuration Design, McGraw-Hill, 1961.(1)
- [2] M. B. Anderson, J. E. Burkhalter and R. M. Jenkins: "Missile Aerodynamic Shape Optimization Using Genetic Algorithms", AIAA Journal of Spacecraft and Rockets, Vol. 37, NO.5, September-October, 2000.
- [3] Melanie Mitchell: "An Introduction to Genetic Algorithms", First MIT Press paperback edition, 1998.
- [4] "Introduction to Genetic Algorithm" Marek Oitko, 1998. Available as <http://cs.felk.cvut.cz/~xobitko/ga/>
- [5] Genetic Server/Library "What are Genetic Algorithms" Neuro Dimension, Inc. 2001. Available at <http://www.nd/products/genetic.htm>
- [6] F. A. Woodward: "An Improved Method for the Aerodynamic Analysis of Wing-Body-Tail Configurations in Subsonic and Supersonic Flow, Part 1", NASA CR-2228, 1973

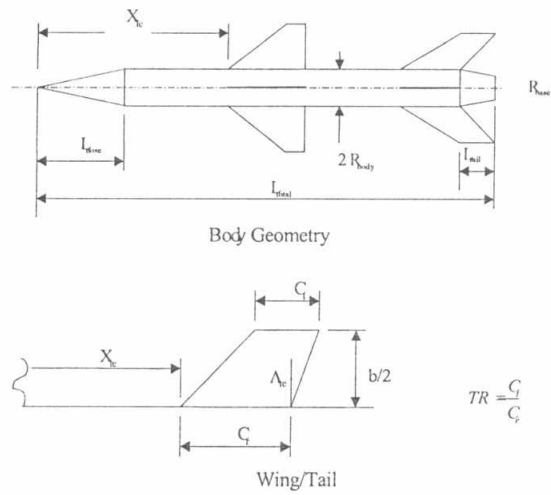


Fig. 1. Design geometrical variables

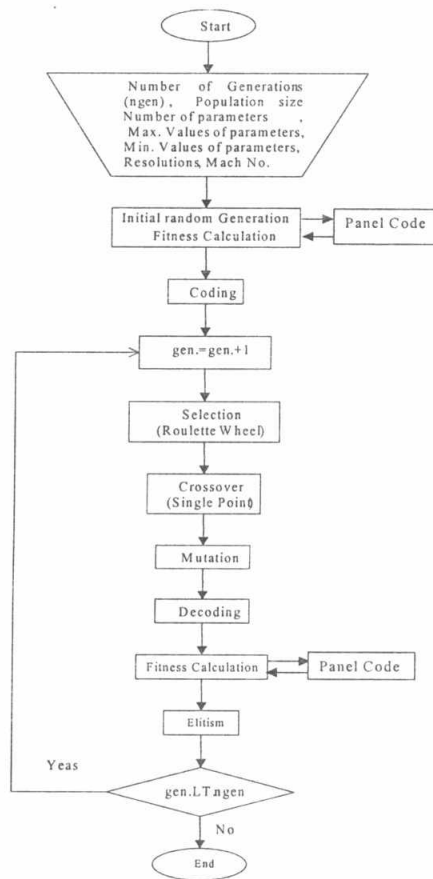


Fig.2. Simple optimization algorithm

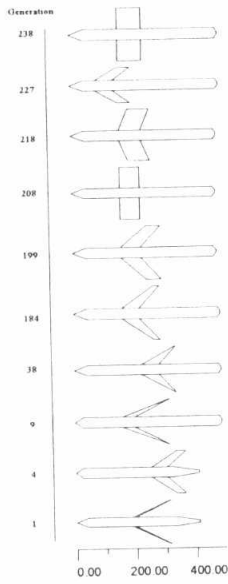


Fig. 3. External shape history for maximum lift coefficient

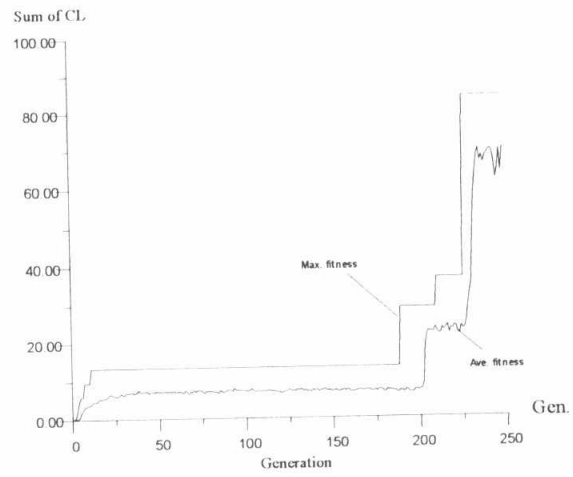


Fig. 4. Convergence history for maximum lift coefficient

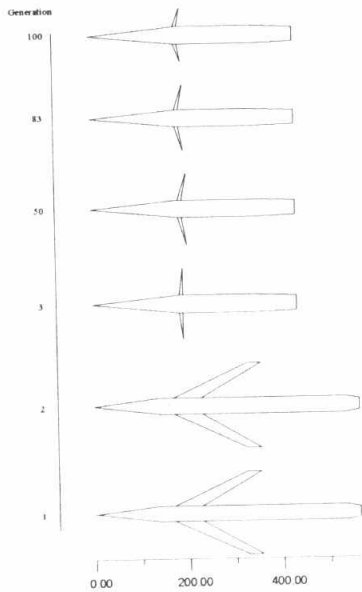


Fig. 5. External shape for minimum drag coefficient

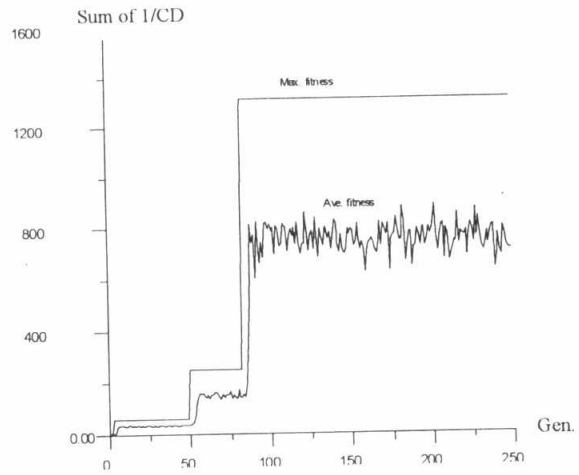


Fig. 6. Convergence design for minimum drag coefficient

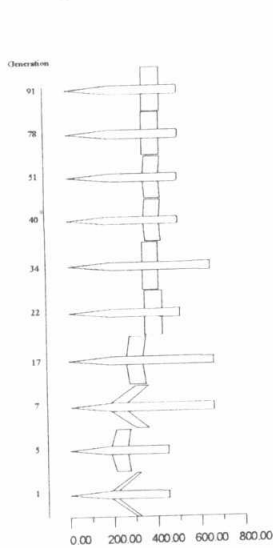


Fig.7. External shape for maximum lift coefficient and minimum drag coefficient

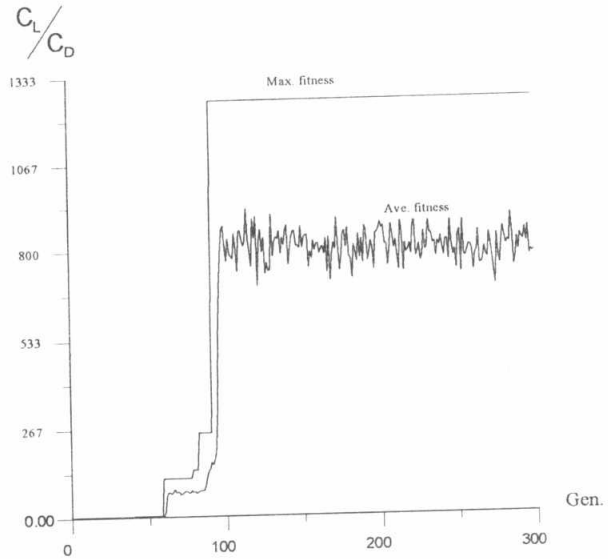


Fig.8. Convergence history for maximum lift coefficient and minimum drag coefficient

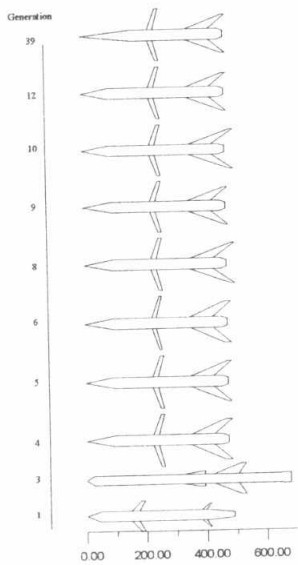


Fig.9. External shape history for maximum lift coefficient minimum drag coefficient and minimum static margin

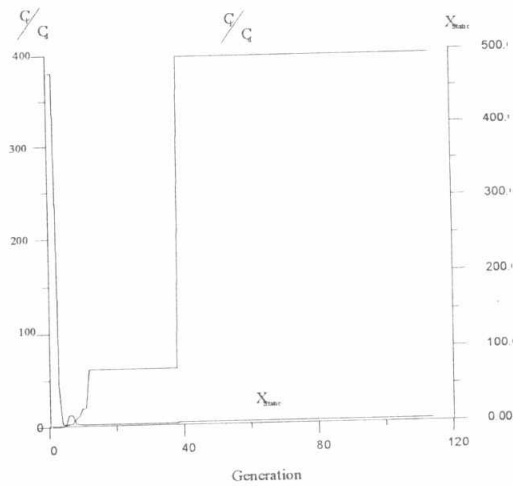


Fig. 10 Convergence history for maximum lift coefficient minimum drag coefficient and minimum static margin

Table I. Design parameters

Parameter	Minimum (in)	Maximum (in)	Resolution	Number of genes
L_{nose}	20	200	5	6
L_{body}	300	700	5	7
L_{tail}	10	100	5	5
R_{body}	16	20	0.25	5
R_{base}	4	20	0.25	7
X_{lew}	20	350	2	8
b_w	5	80	1	7
C_{rw}	5	80	1	7
Λ_{tew}	0	75	5	4
TR_w	0	1	0.05	5
X_{let}	300	690	2	8
b_t	5	80	1	7
C_{rt}	5	80	1	7
Λ_{tet}	0	75	5	4
TR_t	0	1	0.05	5