A Parallel Genetic Algorithm for Multi-Objective Optimization in Computational Fluid Dynamics

A Parallel Genetic Algorithm for Multi-Objective Optimization in Computational Fluid Dynamics

Edited by

John Wiley & sons

A Wiley-Interscience Publication
JOHN WILEY & SONS
New York • Chichester • Weinheim • Brisbane • Singapore • Toronto

Contributors

NATHALIE MARCO INRIA, 2004 Route des Lucioles, BP. 93, 06902 Sophia Antipolis Cédex-France

STEPHANE LANTERI INRIA, 2004 Route des Lucioles, BP. 93, 06902 Sophia Antipolis Cédex-France

JEAN-ANTOINE DESIDERI INRIA, 2004 Route des Lucioles, BP. 93, 06902 Sophia Antipolis Cédex-France

JACQUES PERIAUX Dassault-Aviation, AMD-BA, 78, Quai Marcel Dassault, B.P. 300, 92214 Saint-Cloud

Contents

Abstract :
We are developing and analyzing methodologies involving Genetic Algorithms for multi-objective optimum aerodynamic shape design. When several criteria are considered, the optimal solution is generally not unique since no point achieves optimality, w.r.t. all criteria simultaneously. Instead, a *set* of non-dominated solutions called *the optimal solutions of the Pareto set* can be identified.

0.1 INTRODUCTION

In everyday life, one is confronted with a variety of multi-objective optimization problems such as buying a new television set not too expensive and of good quality or chosing the optimal route to minimize time and maximize safety. In engineering, scientists attempt to build very performant spacecraft structures at low cost. In economics, one has to conciliate the utilities and the consumption of commodities to maximize the social welfare of each individual.

The common part of these optimization problems is that they have several criteria to be taken into consideration simultaneously. In these cases, methods of single objective optimization are no more suitable. Multi-objective optimization need be considered. A large number of application areas of multi-objective optimization has been presented in the literature.

In a simple optimization problem, the notion of optimality is straightforward. The best element is the one that realizes the minimum or the maximum of the objective function.

In a multi-objective optimization problem, the notion of optimality is not so obvious. If we agree in advance that we cannot link the values of the different objectives (i.e. if we refuse to compare apples and oranges), we must find a different definition of optimality, a definition which respects the entirety of each criterion. Then, the concept of Pareto optimality arises.

There is no solution that is the best for all the criteria, but there exists a set of solutions that are better than other solutions in all the search space, when considering all the objectives. This set of solutions is known as *the optimal solutions of the Pareto set* or *non-dominated solutions.*

There are two different ways to solve a mutli-objective optimization problem. The first consists in forming a linear combination of the different criteria with different weights and to minimize the resulting function. This approach has two main drawbacks : (1) not all solutions are found, (2) the objective function is meaningless unless the weigths are selected very intellegently.

In contrast, Genetic Algorithms (GAs) operate on a population of potential solutions (potential, because they belong to the feasible space). Then, during the optimization, GAs explore all the feasible space and build a database of solutions determining a cloud. At convergence, the cloud no more evolves and its convex hull determines the optimal Pareto set.

This article is organized as follows : in Section 2, a brief presentation of a multi-objective optimization problem is given ; in Section 3, a three-criteria optimization problem dealing with analytical functions is solved. Finally, in the last Section, we consider a multi-objective optimization problem in Computational Fluid Dynamics, in which the goal is to exhibit a family of profiles that have a smooth transition in shape from a first profile to a second one. The flows on these two profiles have been evaluated at two different Mach numbers and incidences.

0.2 MULTI-OBJECTIVE OPTIMIZATION PROBLEM

0.2.1 A general multi-objective problem

A general multi-objective optimization problem consists of a number of objectives to be optimized simultaneously and is associated with a number of inequality and equality constraints (for a theoretical background, see for example the book of Cohon [1]). It can be formulated as follows :

$$\text{Minimize / Maximize} \quad f_i(x) \quad i = 1, \cdots, N$$

$$\text{subject to :} \begin{cases} g_j(x) = 0 & j = 1, \cdots, M \\ h_k(x) \leq 0 & k = 1, \cdots, K \end{cases} \qquad \text{constraints}$$

The f_i are the objective functions, N is the number of objectives, x is a p-dimensional vector whose p arguments are known as decision variables. In a minimization problem, a vector x^1 is partially less than another vector x^2 when :

$$\forall i \;\; f_i(x^1) \leq f_i(x^2) \quad (i = 1, \cdots, N) \;\; \text{and there exists at least one } i \text{ such that} f_i(x^1) < f_i(x^2) \;\; .$$

We say that *the solution x^1 dominates the solution x^2*.

Definition of non-dominance for 2 criteria in the case of a minimization :

$$\begin{cases} \text{Minimize} \quad f(x) = (f_1(x), f_2(x)) \\ \\ \text{such that} \quad x \in X, \;\text{the feasible region.} \end{cases}$$

x^1 and x^2 are two solutions to be compared. Then, x^1 dominates x^2 iff :

$$f_1(x^1) < f_1(x^2) \quad \text{and} \quad f_2(x^1) \leq f_2(x^2) \quad \text{or} \quad f_1(x^1) \leq f_1(x^2) \quad \text{and} \quad f_2(x^1) < f_2(x^2)$$

0.2.2 Ranking by fronts

GAs operate primarily by selection of the individuals according to their fitness values. But, in a multi-objective optimization problem, several criteria are assigned and *not* a single fitness value. Therefore, to evaluate the individuals, we need to build a fitness value, called *dummy fitness.*
After the application of the definition of non-dominance, the chromosomes are classified by fronts. The non-dominated individuals belong to front 1 (or rank 1), the following non-dominated individuals belong to front 2 (or rank 2), ... the worst individuals belong to front f, where f is the number of fronts (or ranks).
As an example, we consider a situation where the population of individuals is sorted according to two criteria (J_1, J_2). On Figure 0.1(left), the case of a population of

30 individuals corresponding to values of the criteria J_1 and J_2 picked randomly is presented. As a result, the figure indicates the 8 different fronts. On this example, the points belonging to fronts number I, II, III, IV have more chances to be reproduced than points belonging to fronts number $VII, VIII$.

Once the individuals have been ranked, they can be assigned the following values :

$$f_i = \frac{k}{number\ of\ the\ rank\ to\ which\ i\ belongs} \quad k > 0 \quad \text{in case of minimization}$$

$$f_i = k * (number\ of\ the\ rank\ to\ which\ i\ belongs) \quad k > 0 \quad \text{in case of maximization}$$

(i is the number of the individual.)

Because GAs operate on a whole population of points, they allow to find a number of optimal solutions of the Pareto set. A first application of GAs on multi-objective optimization problems has been studied by Schaffer in 1984 (see his experiments, called Vector Evaluated GA (VEGA), in his dissertation [15]). Schaffer's objective was to minimize a cost and maximize a reliability. He distributed the population in two half populations and optimized the two objectives on each half. After a lot of generations, the population converged towards the optima of each sub-region. This method gave a new avenue in the research of a method to solve a multi-objective problem. The results were encouraging but the obtained solutions only converged to the extreme points of the Pareto set (the first point is the first optimized criterion and the second point is the second optimized criterion), where one objective is maximal, since it never selects according to trade-offs among objectives. VEGA's results implied that a speciation through the population was developped. In order to get diversity (i.e to maintain individuals all along the non-dominated frontier), a non-dominated sorting procedure in conjunction with a sharing technique was subsequently implemented, first by Goldberg in [5] and more recently by Horn in [8], and Srinivas in [16]. Then, the goal has been to find a representative sampling of solutions all along the Pareto front.

Our results in multi-objective optimization are based on the algorithm of Srinivas and Deb [16], called the Non-dominated Sorting Genetic Algorithm (NSGA). In the next paragraph we expose this method.

0.2.3 Non-dominated Sorting Genetic Algorithms

The non-dominated sorting procedure requires a ranking selection method which emphasizes the optimal points. The sharing technique, or niche method, is used to stablilize the subpopulations of the "good" points.

We use this method, because one of the main defects of GAs in a multi-objective optimization problem is the premature convergence. In some cases, GAs may converge very quickly to a point of the optimal Pareto set and as the associated solution is better than the others (it is called a "super individual"), it breeds in the population and, after a number of generations the population is composed of copies of this individual only ! The non-dominated sorting technique with the niche method avoids this situation because as soon as a solution is found in multiple copies, its fitness value is decreased and in the next generation, new solutions that are different can be found, even if they are not so good. The fitness values decrease because the different niches to which belong the different optimal solutions are exhibited.

Technical aspects of the method :

- From an initial population of solutions, one first identifies the non-dominated individuals (according to their criteria) as explained in paragraph 0.2.1. These individuals belong to the rank number 1. Recall that all the points belonging to front 1 are better than those not belonging to the front and are not comparable among themselves.
  These individuals have large probabilities to be reproduced.
- Then assign the same dummy fitness f_i to the non-dominated individuals i of front 1 (generally, the dummy fitness f_i is equal to 1).
- To maintain the diversity, the dummy fitness of the individuals is then shared : it is divided by a quantity proportional to the number of individuals around it, according to a given radius. If the individual has a lot of neighbours, this means that there is a lot of similar solutions. Then, the fitness is going to be shared in order to create some diversity in the next generation.
  We call this quantity a **niche** and it is evaluated as follows :

$$m_i = \sum_{j=1}^{M} Sh(d(i,j))$$

  where i is the number of the individual, M is the number of the individuals belonging to the current front (or rank).
  $Sh(d)$ is the **Sharing function**, linear decreasing function of d defined by :

$$\begin{cases} Sh(0) = 1 \\ \\ Sh(d(i,j)) = \begin{cases} 1 - \dfrac{d(i,j)}{\sigma_{share}} & \text{if } \ d(i,j) < \sigma_{share} \\ 0 & \text{if } \ d(i,j) \geq \sigma_{share} \end{cases} \end{cases}$$

  σ_{share} is the maximum phenotypic distance allowed between any two individuals to become members of a niche. It has no universal value ; it is problem-dependent. In [7], Goldberg proposes to implement an adaptive niching scheme to overcome the limitations of fixed sharing schemes.
  $d(i,j)$ is the distance between two individuals i and j. In some multi-objective optimization problems, the *Hamming distance* is used. It is a genotypic (at the string level) distance and it consists in counting the number of different bits in the two strings, divided by the string length. Such a distance has been used in electromagnetics system design optimization (see [12]) and in problems of finding an optimal distribution of active control elements in order to minimize the backscattering of a reflector in computationnal electromagnetics (see [4]). In this case, the Hamming distance has a true sense because the bits 1 and 0 do not correspond to a real codage but to the fact that the control elements are active (1) or not (0). When using real-value binary coding, the

Hamming distance may be biased because two neighbouring solutions may have a very important Hamming distance (e.g : 0111 and 1000 correspond to 7 and 8, which are neighbours but their Hamming distance is maximal !). Then, in these cases, a phenotypic (at the real parameter level) distance is used, the *Euclidian distance*. is the number of individuals belonging to the current front. Then after having created a niche for each individual of the current front, the new dummy fitness $\frac{f_i}{m_i}$ is assigned.

- After this sharing applied, the non-dominated individuals of Front 1 are temporarily ignored from the population. The following non-dominated individuals belong to Front 2. One evaluates the minimum of the quantities $\frac{f_i}{m_i}$ taken over all chromosomes of Front 1. Then this value is assigned as a dummy fitness to the non-dominated individuals of Front 2. Then, a niche is evaluated for chromosomes of Front 2, and new dummy fitness values.
- This process is continued until the whole population has been visited.
- As all the individuals in the population have now a dummy fitness value, selection, crossover and mutation can be applied. The flowchart of the method is presented in Figure 0.1(right) (inspired from [16]).

Remark : In the case of a minimization problem, the fact that we sort the population by assigning a greater dummy fitness to the best individuals (individuals belonging to Front 1 have a greater fitness than individuals belonging to Front 2 which have a greater fitness than the ones belonging to Front 3 ...) implies that the Pareto optimization becomes a *maximization problem*.

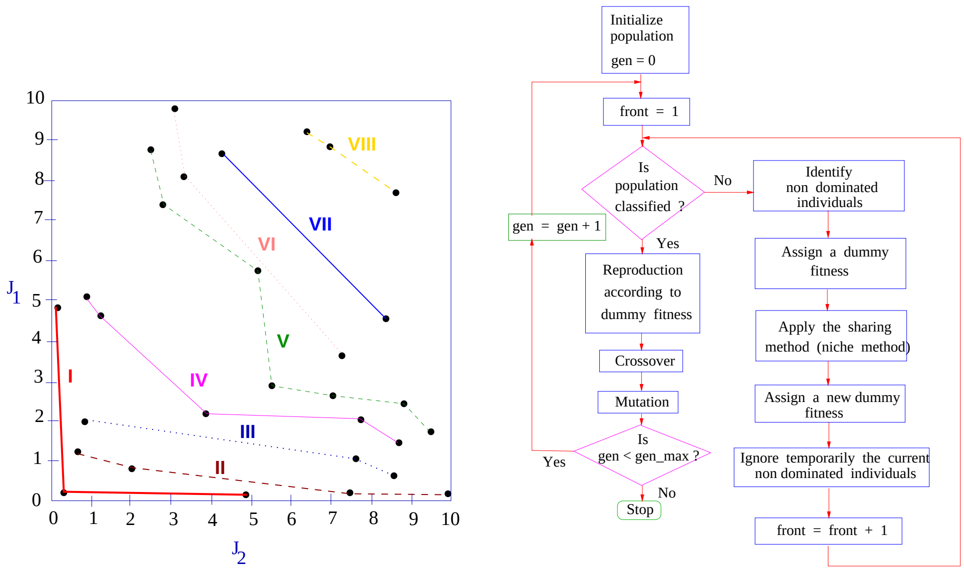

Fig. 0.1 Population ranking by fronts (left). Flowchart of the Non-dominated Sorting GA (right).

0.3 THREE-OBJECTIVE OPTIMIZATION PROBLEM

Three analytical functions depending on two parameters are considered below. The goal is to minimize these three functions simultaneously :

$$\begin{array}{rl} f_1(x,y) = & x^2 + (y-1)^2 \\ f_2(x,y) = & x^2 + (y+1)^2 + 1 \quad \text{with} \quad (x,y) \in [-2,2] \times [-2,2] \\ f_3(x,y) = & (x-1)^2 + y^2 + 2 \end{array}$$

Analytical solution :
The goal is to minimize $f(x,y) = \alpha f_1(x,y) + \beta f_2(x,y) + \gamma f_3(x,y)$ with $\alpha, \beta, \gamma \in [0,1]$ and $\alpha + \beta + \gamma = 1$.
The optimal Pareto front is obtained by solving :

$$\left\{ \begin{array}{l} \dfrac{\partial f(x,y)}{\partial x} = 0 \\ \dfrac{\partial f(x,y)}{\partial y} = 0 \end{array} \right. \Rightarrow \left\{ \begin{array}{l} 2\alpha x + 2\beta x + 2\gamma(x-1) = 0 \\ 2\alpha(y-1) + 2\beta(y+1) + 2\gamma y = 0 \end{array} \right. \Rightarrow \left\{ \begin{array}{l} x = \gamma \\ y = \alpha - \beta \end{array} \right.$$

Then, $x^{opt} \in [0,1]$ and $y^{opt} \in [-1,1]$, that means, $f_1^{opt} \in [0,5]$, $f_2^{opt} \in [1,6]$, $f_3^{opt} \in [2,4]$.

Solution by GAs :
The decision variables x and y are both coded by binary strings of length equal to 26 (corresponding to an accuracy of 10^{-7}) and a population of 40 chromosomes. The probability of crossover is $p_c = 0.9$, the probability of mutation is $p_m = 0.05$ and $\sigma_{share} = 12$. An elitist strategy has been considered and a selection by 2-point tournament has been used.
Figure 0.2 (left) depicts the evolution of the population in the objectives space from the first generation to the last one. We note that the optimal Pareto set is located between $f_1 \in [0,5]$, $f_2 \in [1,6]$ and $f_3 \in [2,4]$.
On Figure 0.2 (right), the optimal Pareto set obtained after 100 generations is also plotted ; there are 1692 individuals on the set.

0.4 MULTI-OBJECTIVE OPTIMIZATION IN SHAPE OPTIMUM DESIGN

0.4.1 The optimization problem

Two reference Euler-flow solutions are first obtained :
a subsonic flow at a free-stream Mach number of 0.2 and an incidence of 10.8^o on a "high-lift" profile (see Figure 0.3) ; the corresponding calculated pressure is denoted P_1. Secondly, a transonic flow at a free-stream Mach number of 0.77 and an incidence of 1^o is computed on a "low-drag" profile (see Figure 0.3) ; the corresponding calculated pressure is denoted P_2.
Then, we consider the two-objective optimization problem consisting in minimizing the following two cost functionals :

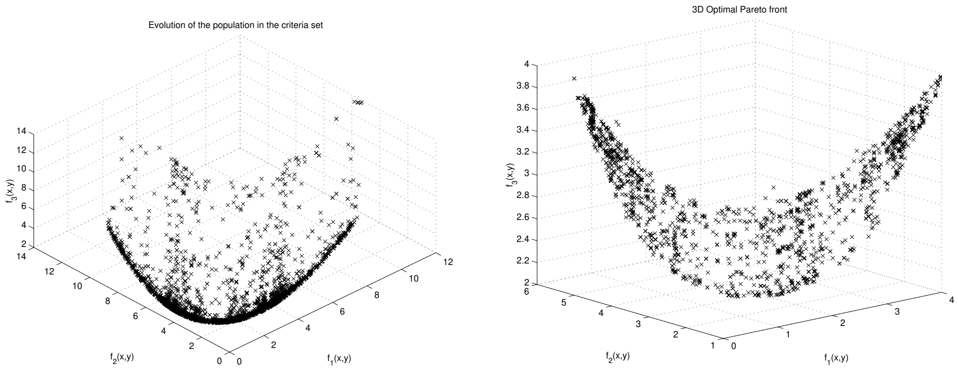

Fig. 0.2 Evolution of the population from the first generation to the 100^{th} (left) and the Optimal 3D Pareto front (right).

$$\begin{cases} J_1(W(\gamma)) = \displaystyle\int_\gamma (P(W) - P_1)^2 \, d\sigma & at\ Mach = 0.2\ and\ \alpha = 10.8^o \\ J_2(W(\gamma)) = \displaystyle\int_\gamma (P(W) - P_2)^2 \, d\sigma & at\ Mach = 0.77\ and\ \alpha = 1^o \end{cases}$$

Following Poloni in [14], we want to determine a whole set of profiles existing between the low-drag profile and the high-lift profile. For each chromosome determining the profile of an airfoil γ, two Euler flows are computed at the two previously mentioned regimes.

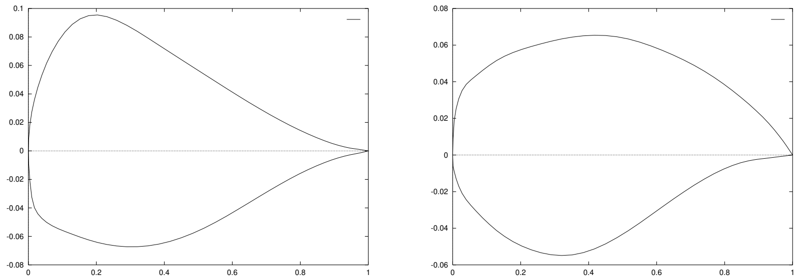

Fig. 0.3 High-lift and low-drag profiles.

The reference profile is the profile of a naca0012 airfoil.

0.4.2 The 2D Euler flow solver

The underlying flow solver solves the 2D Euler equations :

$$\frac{\partial W}{\partial t} + \vec{\nabla}.\vec{\mathbf{F}}(W) = 0 \quad , \quad W = \left(\rho \ , \ \rho\vec{U} \ , \ E\right)^T \quad , \quad \vec{\nabla} = \left(\frac{\partial}{\partial x} \ , \ \frac{\partial}{\partial y}\right)^T$$

where $\vec{\mathbf{F}}(W) = (F_1(W), F_2(W))^T$ is the vector of convective fluxes whose components are given by :

$$F_1(W) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(E+p) \end{pmatrix} \quad , \quad F_2(W) = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(E+p) \end{pmatrix}$$

In the above expressions, ρ is the density, $\vec{U} = (u, v)^T$ is the velocity vector, E is the total energy per unit volume and p denotes the pressure which is obtained using the perfect gas state equation $p = (\gamma_p - 1)(E - \frac{1}{2}\rho \parallel \vec{U} \parallel^2)$ where $\gamma_p = 1.4$ is the ratio of specific heats. The flow domain Ω is a polygonal bounded region of $I\!R^2$. Let $\mathcal{T}_h$ be a standard triangulation of Ω. At each mesh vertex S_i, a control volume C_i is constructed as the union of local contributions obtained from the set of triangles attached to S_i. The union of all these control volumes constitutes a dual discretization of the domain Ω. The spatial approximation method adopted here makes use of a mixed finite element/finite volume formulation. Briefly speaking, for each control volume C_i, one has to solve for n *one-dimensional Riemann problems* at the control volume boundary (finite volume part), n being the number of neighbors of S_i. The spatial accuracy of the Riemann solver depends on the accuracy of the interpolation of the physical quantities at the control volume boundary. For first order accuracy, the values of the physical quantities at the control volume boundary are taken equal to the values in the control volume itself. Extension to second order accuracy can be performed via a "Monotonic Upwind Scheme for Conservative Laws" (MUSCL) technique of Van Leer[17] . It consists in providing the Riemann solver with a linearly interpolated value, taking into account some gradient of the physical quantities (finite element part). Time integration relies on a linearised implicit formulation which is best suited to steady flow simulations [3] :

$$\left(\frac{\text{area}(C_i)}{\Delta t^n} + J(W^n)\right)(W^{n+1} - W^n) = -\Psi(W^n)$$

where $\Psi(W)$ is the flux depending on a numerical flux function and on the vector $F(W)$ of the convective fluxes. $J(W^n)$ denotes the associated Jacobian matrix. At each time step, the above linear system is solved using Jacobi relaxations. We refer to [9] and [11] for a more detailed description of the flow solver.

0.4.3 Parallelization strategy

The main concern related to the use of GAs in optimum shape design is the computational effort needed. In difficult problems, GAs need bigger populations and this translates directly into higher computational effort. In the case of the multi-objective optimization problem, two objectives in our case, we have to solve two

Euler flows for each individual, which means that the computational effort is very high. One of the main advantages of GAs is that they can easily be parallelized. At each generation, the fitness values associated to each individual can be evaluated in parallel. We will not go into the details for the parallelization strategy we have adopted, it is not the topic of our paper. A complete description can be found in [11] or [10]. In summary, we have employed a two-level parallelization strategy : the first level is the parallelization of the flow solver, which combines domain partitioning techniques with a message-passing programming model (where calls to the MPI library are used). The second level is the parallelization of GAs. We have chosen to exploit the notion of process groups which is one of the main features of the MPI environment. Two groups are defined, each group is responsible for the evaluation of the criteria for one individual, and each group contains the same number of processes. Then, two individuals are taken into account in parallel. For more details, one can refer to [11].

0.4.4 Results

0.4.4.1 Data and parameters The "high-lift" mesh is made of 2130 nodes, the "low-drag" of 3847 nodes and the reference Naca0012 mesh of 1351 nodes.
Shape parametrization :
The individuals represent airfoil shapes. Following a strategy often adopted, the shape parametrization procedure is based on Bézier curves. Eigth order Bézier representation has been used (in extrados and intrados) : 2 fixed points, P_0 and P_8, and 7 control points. The values $x_i \in [0, 1]$ are fixed and the only parameters that vary are the ordinates.
The GA :
Results have been obtained with a population of 40 individuals, after 100 generations. The probabilities of crossover and mutation have been set to $p_c = 0.8$ and $p_m = 0.007$, in addition, $\sigma_{share} = 0.7$. The selection operator is the binary tournament with an elitist strategy.

0.4.4.2 Numerical results Results have been obtained on a SGI Origin 2000 system equipped with Mips R10000/195 Mhz processors. By working with 2 groups and 4 processors by group, the CPU time was about 2 hours.
Figure 0.4(left) visualizes the evolution of the population in the course of the generations until a steady state is reached. We note the cloud of points and its convex hull. The optimal Pareto set is shown on Figure 0.4(right), it describes the convex hull. After 100 generations, there are 131 individuals on the Pareto front. All these individuals correspond to optimal solutions of the mutli-objective minimization problem. On Figure 0.5, a sample of the different profiles corresponding to the optimal solutions of the two-objectives problem is depicted.

0.5 CONCLUDING REMARKS

The multi-objective optimization process by a genetic algorithm is carried independently of the decision maker's eventual options : a database is constructed as the

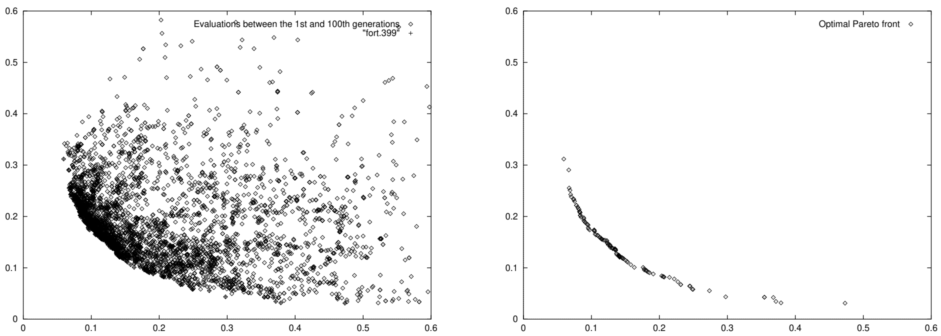

Fig. 0.4 Evolution of the individuals in the objectives plane, from the first generation to the last one (100 generations) (left) and ptimal Pareto set with 131 individuals uniformly distributed (right).

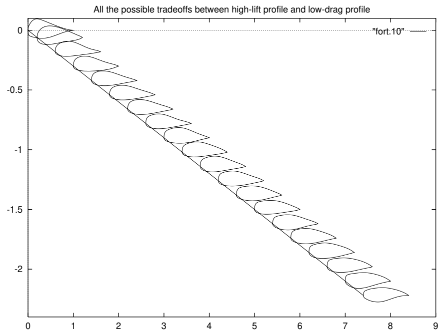

Fig. 0.5 A sample of profiles between high-lift and low-drag.

population evolves and the pareto front is achieved at convergence of this database. In this way, the approach offers an alternative to optimization with penalty functions.

In some cases, it is difficult to identify numerically all feasible optimal solutions, particularly those corresponding to trade-offs between the objectives. A sharing procedure with a classification by fronts can then be introduced to increase diversity between the optimal solutions all along the Pareto front ([6],[2],[13],[7] and we have tested the method of [16]).

In this contribution, we have demonstrated the application of a GA for a practical 2D airfoil design optimization according to two criteria related to subsonic and supercritical conditions.

0.6 BIBLIOGRAPHY

REFERENCES

1. J.L. COHON, "*Multiobjective Programming and Planning*", Academic Press, New-York, R. Bellman Editor, 1978.
2. K. DEB and D.E. GOLDBERG, "An investigation of niches and species formation in genetic function optimization", In J.D. Schaffer, editor, *Proceedings of the Third International Conference on Genetic Algorithms*, pages 42–50, San Mateo (CA), Morgan Kaufmann, 1991.
3. L. FEZOUI and B. STOUFFLET, "A class of implicit upwind schemes for Euler simulations with unstructured meshes", *J. of Comp. Phys.*, 84:174–206, 1989.
4. R. GLOWINSKI, J. PERIAUX, M. SEFRIOUI, and B. MANTEL, "Optimal backscattering of a reflector by means of Genetic Algorithms", In *ECCOMAS 96*, New-York, John Wiley and Sons, 1996.
5. D.E. GOLDBERG, "*Genetic Algorithms in Search, Optimization and Machine Learning*", Addison-Wesley Company INC., 1989.
6. D.E. GOLDBERG and J. RICHARDSON, "Genetic Algorithms with sharing for multimodal function optimization", In *Genetic Algorithms and Their Applications : Proceedings of the Second International Conference on Genetic Algorithms*, pages 41–49, San Mateo (CA), Morgan Kaufmann, 1987.
7. D.E. GOLDBERG and L. WANG, "Adaptive niching via coevalutionary sharing", In C. Poloni D. Quagliarella, J. Périaux and G. Winter, editors, *Genetic Algorithms and Evolution Strategies in Engineering and Computer Science*, pages 21–38, Trieste (Italy), John Wiley and Sons, 1997.
8. J. HORN and N. NAFPLIOTIS, "Multiobjective Optimization Using the Niched Pareto Genetic Algorithm", In *Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*, Piscataway (NJ), Volume 1 (ICEC '94), 1994.
9. S. LANTERI, "Parallel solutions of compressible flows using overlapping and non-overlapping mesh partitioning strategies", *Parallel Comput.*, 22(7):943–968, 1996.
10. N. MARCO and S. LANTERI, "Parallel Genetic Algorithms applied to Optimum Shape Design in Aeronautics", In M. Griebl, C. Langauer and S. Gorlatch, editors, *Euro-Par'97 Parallel Processing*, pages 856–863, Passau (Germany), Springer Verlag, 1997.
11. N. MARCO and S. LANTERI, "A two-level parallelization strategy for Genetic Algorithms applied to shape optimum design", INRIA Report No 3463 - Submitted to *Parallel Comput.*, 1998.
12. E. MICHIELSSEN and D.S. WEILE, "Electromagnetic System Design Using Genetic Algorithms", In M. Galàn G. Winter, J. Périaux and P. Cuesta, editors, *Genetic Algorithms in Engineering and Computer Science*, pages 345–368. John Wiley and Sons, 1995.
13. C.K. OEI, D.E. GOLDBERG, and S.J. CHANG, "Tournament selection, niching and the preservation of diversity", IlliGAL Report No 91011, 1992.

14. C. POLONI, "Hybrid GA for Multi-Objective Aerodynamic Shape Optimization", In M. Galàn G. Winter, J. Périaux and P. Cuesta, editors, *Genetic Algorithms in Engineering and Computer Science*, pages 397–415. John Wiley and Sons, 1995.

15. J.D. SCHAFFER, "Some experiments in machine learning using Vector Evaluated Genetic Algorithms", Unpublished doctoral dissertation, Vanderbilt University, Nashville (TN), 1984.

16. N. SRINIVAS and K. DEB, "Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithm", *Evolutionary Computation*, 2(3):221–248, 1995.

17. B. VAN LEER, "Towards the ultimate conservative difference scheme V : a second-order sequel to Godunov's method", *J. of Comp. Phys.*, 32:361–370, 1979.