Research Article

Framework for the Shape Optimization of Aerodynamic Profiles Using Genetic Algorithms

D. López, C. Angulo, I. Fernández de Bustos, and V. García

Department of Mechanical Engineering, University of the Basque Country, UPV/EHU, Alameda Urquijo s/n, 48013 Bilbao, Spain

Correspondence should be addressed to C. Angulo; carlos.angulo@ehu.es

This study developed a framework for the shape optimization of aerodynamics profiles using computational fluid dynamics (CFD) and genetic algorithms. A genetic algorithm code and a commercial CFD code were integrated to develop a CFD shape optimization tool. The results obtained demonstrated the effectiveness of the developed tool. The shape optimization of airfoils was studied using different strategies to demonstrate the capacity of this tool with different GA parameter combinations.

1. Introduction

Most optimization techniques developed for computational fluid dynamics (CFD) analysis in the last two decades have focused on design problems in aerospace, aeronautic, and automotive applications. Developments in the fields of optimization techniques and turbulence models have facilitated the solution of more realistic aerodynamic design problems by computer simulation.

In recent decades, advances in hardware have allowed a considerable progress in flow prediction using Direct Numerical Simulation (DNS) and Large Eddy Simulation (LES) methods, which are mainly used in research. This constitutes a platform for understanding the physics of flow and the development of better Reynolds Averaged Navier-Stokes (RANS) turbulence models. Recently, some insight has been gained in the understanding of the physical mechanisms of transition from laminar to turbulent flow [1, 2], this enables important advance in transition prediction in the field of RANS models [3–6]. However, there is still an inability to accurately predict flow separation, which also depends on appropriate transition prediction.

The progress in flow prediction has permitted the development of aerodynamic shape optimization towards a practical and realistic design tool [7, 8]. Optimization methods can be classified into two categories: global search methods, which are generally stochastic, for example, genetic algorithms (GA) that explore the entire search space; and local

search methods, which are generally deterministic, including those based on gradient calculations or their approximations.

Global methods must perform a vast number of evaluations, whereas local methods are less expensive but they tend to become trapped in local extrema. GAs have demonstrated their robustness to avoid local extrema and numerical noise in aerodynamic optimization [9–12], as well as their validity in problems with constraints [13, 14]. Although GAs do not guarantee obtaining the global optimum, they constitute reliable tools to find a better design [15]. GAs are explorative methods; that is, they analyze many possible profile shapes. This is not a problem because every shape with a bad verification according to the evaluation function will be assigned a low aptitude value.

A shape optimization system was developed for fluid dynamic problems, which used the genetic code known as Oriented to Mechanical Engineering Genetic Algorithm (OMEGA), which is based on David Levine's PGA Pack Library [16]. This method has been applied to the optimization of bidimensional aerodynamic profiles with geometric constraints. The evaluation function analyzes the flow using the Fluent commercial code. A study was carried out to obtain efficient combinations of the basic GA parameters.

2. Problem Definition

A high Reynolds number subsonic flow optimization problem is considered. Two different fitness functions are used

to optimize two different aerodynamic features for a two-dimensional airfoil, which is subjected to constraints on its geometry to guarantee a threshold structural strength. The first function to maximize is c_l/c_d, the second is to minimize c_d for a given c_l value.

Advances in transition prediction [3, 4] have condensed empirical models such as the empirical model of Menter et al. [5] or the phenomenological model of Walters and Cokljat [6]. These models are designed to switch the turbulence, depending only on local variables, that is, those defined at each point, which reduces the computational cost and facilitates the implementation of CFD codes, while the effects of the models on the flow are constrained to the transition region. In this study, the transition model of Walters and Cokljat [6] was used because it has proven its ability to predict transitional flow behavior in flows around airfoils versus current generation RANS-based turbulence models. This model is based on a simple two-equation k-ω turbulence model [17] and a third equation is added to account for the physical effects of natural [18] and bypass [2] mechanisms of transition. Natural transition occurs for values of freestream turbulence intensity below 1%, while bypass transition happens above this value.

2.1. Governing Equations.
The mathematical model used in this study is based on the RANS system of differential equations for incompressible flow. The governing equations for the steady mean flow are expressed as follows.

Conservation of mass, consider

$$\frac{\partial U_i}{\partial x_i} = 0, \tag{1}$$

where the vectors U_i and x_i are the average velocity and position, respectively.

Conservation of momentum, consider

$$\rho U_j \frac{\partial U_i}{\partial x_j} = \frac{\partial}{\partial x_j}\left(-P\delta_{ij} + 2\mu S_{ij} - \rho\overline{u_i u_j}\right), \tag{2}$$

where ρ is air density, P is the average pressure, S_{ij} is the mean flow strain rate tensor, μ is the molecular viscosity, and $\delta_{ij} = 1$ for $i = j$; otherwise, $\delta_{ij} = 0$. Here, u_i is the velocity fluctuation, and the term $-\rho\overline{u_i u_j}$ is the Reynolds stress tensor, where the overline denotes average.

The k-k_l-ω turbulence model [6] was used in this study. It introduces the anisotropic component of the Reynolds stress tensor through the Boussinesq modeling hypothesis as follows:

$$-\rho\overline{u_i u_j} + \frac{1}{3}\rho\overline{u_k u_k}\delta_{ij} = 2\mu_t S_{ij}, \tag{3}$$

where μ_t is the turbulent viscosity. The effects of transition to turbulence and turbulence itself are approximated by three additional transport equations and influence the mean flow through the value of the turbulent viscosity. This is a carefully defined function that considers those effects.

2.2. Simulation.
All the simulations in the optimization runs presented in this paper were performed using the Fluent solver based on finite volume method. A second order upwind discretization scheme was used. The boundary conditions applied on the computational fluid domain can be seen in Figure 2. A constant velocity profile for the inlet semicircumference along with a turbulence intensity of 0.93%, and a constant pressure condition for the outlet semicircumference were imposed. The no slip boundary condition was imposed at the airfoil wall. All the meshes were constructed with a first cell y+ value around the unity. The same type of meshes were used for known NACA airfoils yielding errors around 25% and 5%, respectively, for the drag and lift efforts. The authors have checked that a higher density in the normal direction to the wall in the boundary layer can halve the error in both efforts, although this increases the CPU time per calculation.

2.2.1. Parameterization.
To ensure a successful optimization process, the search space must incorporate as much geometric flexibility as possible with as few design variables as possible. In this sense, Bezier curves provide soft and flexible shapes if the design variable intervals are defined appropriately. The upper and lower airfoil curves, extrados and intrados, are each defined by a Bezier polygon with five vertices. The design variables used are the vertical positions of the three intermediate Bezier polygon vertices, as shown in Figure 1, and their ranges of definition are shown on the right.

2.2.2. Meshing Method.
During aerodynamic shape optimization in search spaces with high geometric flexibility, the design variables are allowed to vary in wide ranges, but when GAs are used, the geometries vary drastically from one evaluation to the next, so the reliability of the meshing method is a key issue. A suitable meshing method for CFD analysis in GA optimizations must be robust and accurate. The capability to mesh every geometry in the search space is necessary, but the meshes also have to be well shaped in the critical flow zones to provide accurate results. By contrast, GAs evaluate massive populations among a large number of generations, thereby incurring a huge overall cost per execution. Thus, the meshing method used in this study was designed to accomplish three objectives: the robustness of the mesh generation process, the accuracy of the meshes, and an affordable mesh cost.

Using this framework, a reliable meshing procedure for GA optimization problems was developed [19]. Instead of deforming the previous grid, a new mesh is reconstructed by dividing the domain into mesh blocks, which vary according to the geometry of the profile. However, since the advent of more precise RANS turbulence models, several modifications have been included in this procedure to obtain meshes that are capable of predicting flows more realistically. It is important for the CFD analysis to calculate the transition points at the extrados and intrados of the airfoil where the boundary layer turns from laminar to turbulent. If a completely turbulent boundary layer is calculated (like k-ε turbulence models) certain flow characteristics never occur in the simulation. Laminar separation bubble and the
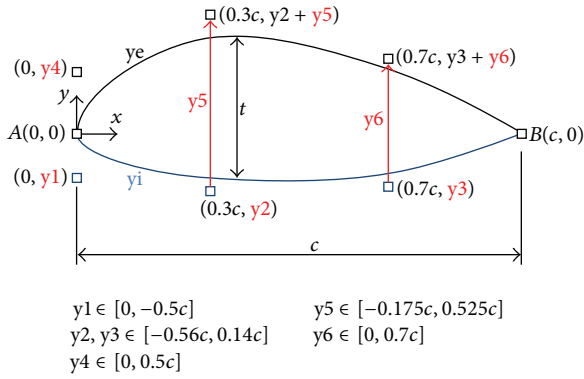
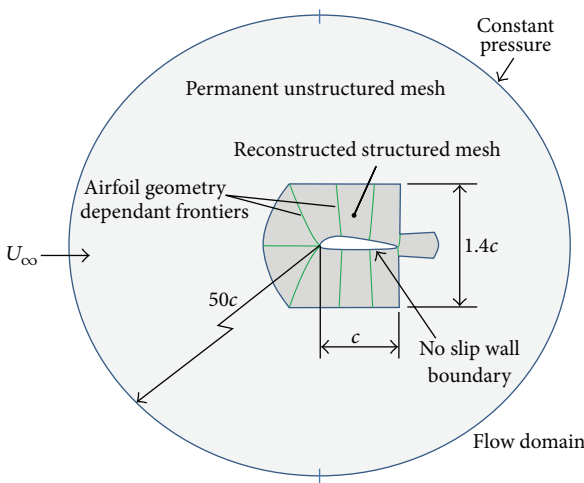FIGURE 1: Design variables used to define the profile and their ranges.

FIGURE 2: Flow domain for CFD analysis, type of boundary conditions and the structure of the mesh blocks.

associated change in pressure distribution are suppressed, and also predicted skin friction distribution and lift and drag values differ significantly. Moreover, the overall separation behavior is different.

As noted above, the new RANS transition models have facilitated the inclusion of transition predictions in CFD analysis in shape optimization. The correct use of these models requires very fine meshes in the wall region inside the boundary layer, with y+ values around the unity, so the number of cells across the boundary layer height and along the wall has increased greatly. As a secondary effect, this has generated an excessive number of transverse (to the flow) mesh lines in the outer region of the boundary layer, which is reduced by applying several layers of special-shaped cells that join two mesh lines in the transverse direction within the nearby boundary layer edge.

To minimize the shape distortion of nearby wall cells, the mesh block surrounding each wall was divided into three parts by tracing two new transverse block frontiers that emerge from the wall in a direction approximately perpendicular to it, as shown in Figure 2. In this manner, the direction of the surrounding transverse mesh lines is more constrained in the same way.

Given that the leading part of the airfoil is where the maximum slope variations are allowed (see Figure 1), cell shape control is increased in this region. Thus, the division of the wall surrounding mesh block is not divided in three equal ranges along the x direction, but instead by cutting at distances $0.2c$ and $0.66c$ from the leading edge. Here, c denotes the chord length of the airfoil.

Three transition cell layers were used in the second and third blocks, which reduced the mesh line density by eight. However, the flexibility of the geometry in the first part of the airfoil led to the possibility of excessively distorted cells in the third transition layer, the most exterior layer of the leading mesh block. To avoid this problem, a transition cell layer was eliminated in this block, which caused a relative increase in the mesh density in the outer area, as shown in Figure 3.

Unstructured meshes were used to reduce the mesh cost in the mesh blocks situated farther from the wall, where the flow variations are very small. This type of mesh permits a faster increase in the cell size in an outward direction. Thus, the overall mesh cost was reduced greatly while maintaining the accuracy (see Figures 2 and 3).

Structured mesh blocks were used in the airfoil surrounding the mesh blocks to guarantee the robustness of the grid generation process. These mesh blocks were regenerated for each new airfoil using the transfinite interpolation technique [20], which is a reliable technique whenever the curvilinear polygon defining the block is traced correctly and the cell length distribution along these curve-edges is adequate. Furthermore, transfinite interpolation permits highly anisotropic cell shapes (large aspect ratios) in the wall region of the boundary layer. These shapes facilitate a large reduction in the number of cells along the wall compared with isotropic shapes, because the flow gradients are several orders of magnitude smaller in the parallel direction than in the normal direction to the wall.

The block boundaries starting from the wall were designed according to the slope of the wall in the same way as a previous study [19], but the boundaries opposite to the wall are now straight and independent of the airfoil. As the family of the mesh lines "parallel" to the flow direction is calculated by linear interpolation between the wall and its opposite straight frontier, these mesh lines become more similar to the wall as one moves towards it. Now also all of the mesh blocks not adjacent to the airfoil become independent of its shape, and so they do not need to be reconstructed for each new airfoil.

These mesh blocks (except the wake region that had anisotropic cell shapes) were meshed with unstructured meshes using a mobile front technique [20], which can fail at the final stage of the meshing process. Then, to ensure robustness, the outer blocks of the domain were meshed before the start of the optimization process and they were kept permanent for each airfoil to avoid the possibility of failure when meshing an airfoil. This measure reduced the overall meshing time dramatically.

The meshes comprise approximately 12000 finite volumes or cells (see Figure 3). To confirm the grid-independence of the results, a second mesh was constructed with twice the
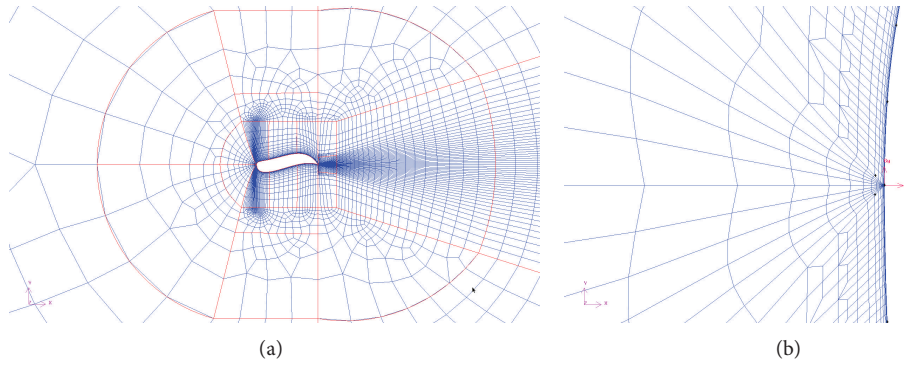
(a)

(b)

Figure 3: (a) Mesh around an aerodynamic profile and (b) detail near its leading edge.
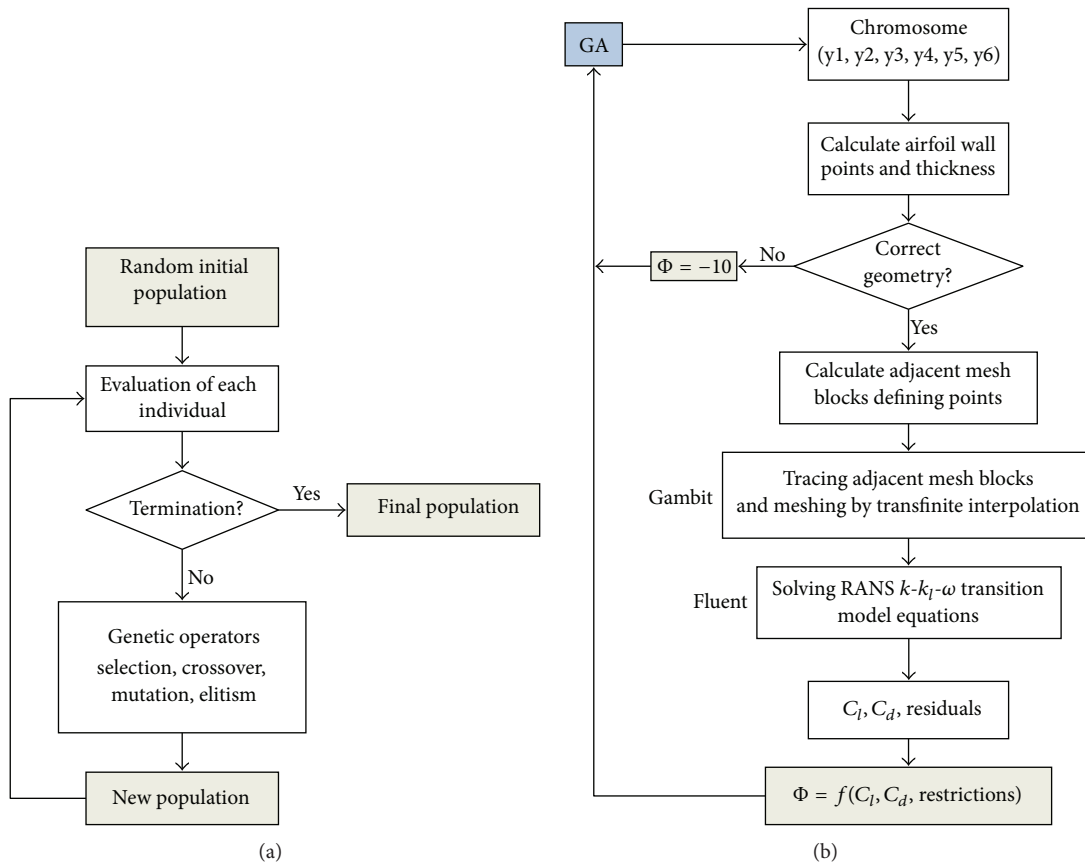

(a)

(b)

Figure 4: (a) Optimization scheme and (b) details of evaluation procedure used for each airfoil.

resolution in the streamwise and wall-normal directions, and the results were compared to those using the original mesh. In all cases, the results had negligible differences so were judged to be mesh-independent.

The validity of the meshing method was checked by numerous runs. Moreover, the method could be adapted to other problems and different domains.

The sequence used to mesh the domain around an airfoil is showed in the flowchart in Figure 4. Each airfoil shape is generated using NURBS curves (Not Uniform Rational B-Splines) by creating the files containing discrete data points

on the airfoil surface, which were defined as a Bezier curve (Figure 1). These files are imported into Gambit, a general purpose preprocessor for CFD analysis, which creates the mesh of the outer flow domain. Next, the blocks immediately surrounding the airfoil are designed and meshed, before the whole mesh is exported.

2.3. Optimization. Optimization was performed using a simple GA that follows the sequence shown in Figure 4(a). First, a seed number s is provided by the user, which is used to generate a univocal sequence of random numbers that form

the genes of the p chromosomes in the initial generation population. This is denoted as $P(t = 0)$, where P means population and $t = 0$ is the number of the generation. The seed number could also be generated randomly, but the former option was preferred because it facilitated the performance of different experiments starting with the same initial population to investigate the separate effects of different combinations of GA parameters on exploratory and exploitative behaviors during search. The aim was to tune them to obtain the best results for a given objective function.

After generating the initial population $P(0)$, each individual was evaluated (as explained in paragraph 3, see Figure 4(b)), before selection, crossover, and mutation operators are applied to this population to obtain the next generation of individuals $P(1)$.

The sequence used to obtain $P(1)$ is as follows. First, a new population $P'(1)$ is generated after p tournaments. Next, with a crossover rate c, approximately $pc/2$ pairs of individuals are crossed, each of which yielded two offsprings that substituted their parents and joined with the noncrossed individuals to form population $P''(1)$. The two-point crossover type was used. Next, a percentage $m \cdot 100$ of the genes were mutated in this population. Finally, the duplicated individuals in the current population were mutated repeatedly until no identical individuals remained. This operation increases the exploration of the search space. After the first generation $P(1)$ has been obtained, the same process is repeated until the termination criterion is satisfied. The criterion used in this study was 100 generations.

When elitism is applied, the most fit individual in a generation $P(i)$ is automatically introduced into the next one $P(i + 1)$. This option increases the exploitative pressure and accelerates convergence, although it reduces the diversity in populations throughout evolution. Thus, the search is prone to becoming trapped in local extrema if small populations are used.

When elitism is not used, the algorithm tends to converge if a good individual is found, but this does not reduce the exploration of other possibilities in the search space. A more precise definition of the other parameters was required to achieve this behavior. This is the antagonism between the principal aspects of GAs: search and tuning. It is not possible to tune a solution if there is a tendency to search the entire space, and vice versa. Thus, the user has to find an adequate balance in each actual case.

3. Framework

In this study, the optimization by means of GAs of expensive nonanalytical objective functions is tackled. Each individual is an airfoil and its fitness value is a point in the search space. The fitness functions used in aerodynamic shape optimization problems aim to represent certain aspects of the airfoil's aerodynamic performance that need to be improved. Thus, the direct problem was tackled and several fitness functions were used, which were defined as functions of the aerodynamic loads imposed by the flow on the airfoil, that is, the aerodynamic coefficients c_l and c_d. Restrictions

on the airfoil thickness were also included to guarantee a threshold structural strength. An expensive CFD analysis has to be performed to evaluate an airfoil. This required the laborious implementation of a sequence of tasks using different techniques to generate an automatic process. This framework is shown in the flowchart in Figure 4. The most expensive and complex tasks are meshing and solving the flow model, which are achieved using two commercial codes, that is, the preprocessor Gambit and the CFD solver Fluent, respectively.

As shown in Figure 4, the GA optimizer sets the pace of the optimization process, which included each individual evaluation and assignment of a fitness value. This comprises the concatenation of calls to the different tools and codes that perform each task. The user entered the relevant parameters for the GA and these codes were controlled automatically using script files after the optimization commenced.

The airfoil evaluation process is governed by the GA, which operates as follows. Two sets of points are calculated using the set of Bezier vertices provided by the GA. The first set consists of two rows of 50 points, which are interpolated to define, respectively, the upper and lower curves of the airfoil. If these curves intersected with each other, the airfoil geometry is incorrect and it is assigned a negative fitness value, so it is omitted from the CFD analysis. The second set of points is used to define the boundaries of the mesh blocks adjacent to the airfoil. Next, both sets of point coordinates are written to the text journal file of Gambit, which also includes the commands used to build the mesh blocks next to the airfoil and to fill them with structured meshes. Finally, Gambit exports the whole mesh to the Fluent code.

Fluent conducts the processing and postprocessing tasks specified in its journal file. A second order discretization scheme is used. The airfoil characteristics c_l and c_d are obtained and the corresponding fitness value Φ is calculated as a function of these data and a constraint on its thickness (see Figure 4).

4. Results

A preliminary experiment was performed to identify efficient combinations of parameters for the basic GA. The aim was to optimize the shape of the bidimensional aerodynamic profiles for incompressible flow with a Reynolds number of 6×10^6 based on the chord length.

Two optimization criteria were used. The first is to maximize the ratio of the lift relative to the drag force, and the second consists on minimizing c_d for a given value of c_l. In both cases, the angle of attack is zero and the airfoil's thickness t is constrained to be above $0.12c$, where now c is the chord length and t is the maximum vertical distance between the intrados and the extrados along the chord.

These functions are defined in the following form:

$$\Phi = f\left(c_l, c_d\right) \cdot p_t \cdot p_a \cdot p_s, \tag{4}$$

where f is the aerodynamic function that needs to be optimized, and the variables p are penalty functions that
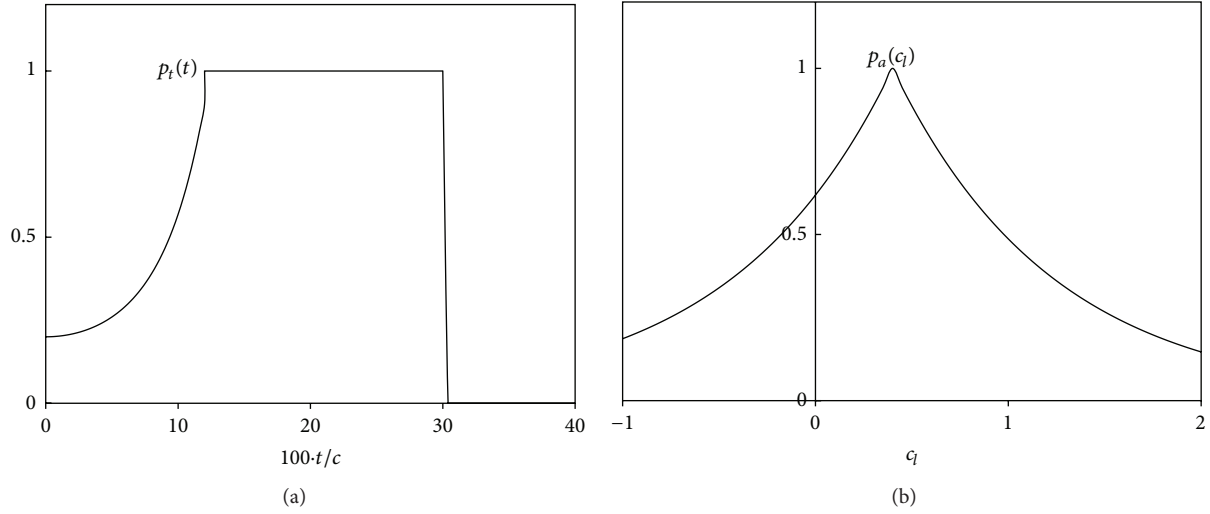
FIGURE 5: Penalty factors for the thickness (a) and lift (b) constraints.

include nonlinear constraints of different type. Their values are defined in the range $[0, 1]$ and they were assigned 1 if the constraint was satisfied, but reduced from this value according to the degree to which the constraint was exceeded, where 0 was the minimum value. The penalty function p_t introduces the airfoil thickness constraint, p_a is the necessary condition for the aerodynamic loads, and p_s penalizes the airfoils where the flows are suspected to be solved incorrectly.

4.1. Optimization of c_l/c_d with a Constraint on the Airfoil Thickness.
The first objective was to find airfoil shapes with a maximum lift per drag ratio and a thickness above $0.12c$. Thus, the objective function was defined in the following form:

$$\Phi = \left(\frac{c_l}{c_d} \right) \cdot p_t(t) \cdot p_s(r_k), \tag{5}$$

where

$$p_t(t) = \begin{cases} 0.2 \exp\left[0.01045t^2\right] & \text{if } 0 < t < 0.12c, \\ 1 & \text{if } 0.12c < t \leq 0.3c, \\ 0 & \text{if } t > 0.3, \end{cases} \tag{6}$$

$$p_s(r_k) = \begin{cases} 1 & \text{if } r_k \leq 5 \cdot 10^{-4}, \\ 0 & \text{if } r_k > 5 \cdot 10^{-4}, \end{cases} \tag{7}$$

where the two penalty factors $p_t(t)$ and $p_s(r_k)$ were defined carefully with respect to the constraint on the airfoil (maximum) thickness and to eliminate incorrectly calculated airfoils. The parameter r_k denotes the value of the residual of the turbulent kinetic energy equation obtained during the last iteration of the simulation.

In search spaces with constraints, the optimum solution may be found in the boundaries where the constraint conditions are satisfied. For this problem in particular, given a certain camber and an airfoil thickness distribution, the ratio of c_l/c_d is increased as the maximum thickness t is reduced.

This can be seen for example, by comparing the experimental data of decreasing thickness airfoils naca2424, naca2412, and naca2408 [21]. This fact was considered to define $p_t(t)$ in (6), which is shown in Figure 5. The aim of this term was to direct the evolution towards airfoils with a thickness value over $t = 0.12c$, but also to conserve some useful genetic information from slightly thinner airfoils.

A severe penalty was used for excessively thin profiles, which was motivated by structural requirements and the relative increase in the ratio c_l/c_d as the thickness declined. The function p_t increases with the thickness to direct the population towards thicker airfoils, but without completely eliminating thinner ones with a good ratio c_l/c_d because they could add useful genetic information to the new generation. Moreover, a 10% amplitude jump was included at the thickness value $t = 0.12c$ to highlight this boundary. Thicknesses above $0.3c$ are considered excessive.

The factor $p_s(r_k)$ was used to qualify the reliability of the solution to eliminate airfoils from the evolution process in the cases the flow was not calculated correctly. When the errors overestimate the aerodynamic performance they are very detrimental to the search process.

After performing CFD analysis using many airfoils with the k-k_l-ω transition turbulence model, it was observed that simulations rarely produced great errors in the aerodynamic loads. And this usually occurred in simulations that gave flows with large separated regions. It was found that in contrast with the correct simulations, all of the incorrect ones had final values for the residual in the k_t equation that exceeded 5×10^{-4}. Therefore, correct flags were given when the residuals in the k_t equation fell below this value. This prevented the contamination of the genetic search with fake "superindividuals" which would have produced bad results.

The GA used a tournament selection operator and a generational replacement model. Tests were performed using different combinations of population sizes, crossover rates, and mutation rates, as shown in Table 1. Two-point-type

TABLE 1: Results for each combination of GA parameters.

Population size	40				100	
Mutation rate	0.0025		0.025		0.01	0.025
Crossover rate	0.5	0.65	0.5	0.65	0.65	0.65
Best individual found	(a) 75.53	(a) 82.30	(a) 82.64	(a) 68.84**	(e) 89.04	(e) 87.72
	(b) 81.72	(b) 77.32	(b) 88.02	(b) 86.60		
	(c) 82.07	(c) 76.27	(c) 87.95	(c) 88.11		
				(d) 87.43		
	(a)* 88.98	(a)* 81.91	(a)* 89.51	(a)* 90.77		
Best individual average	79.64	79.28 (a, c)	86.20	82.74		
				87.38 (b, c, d)		

( ) Initial population. * Elitism. ** Anomalous evolution.
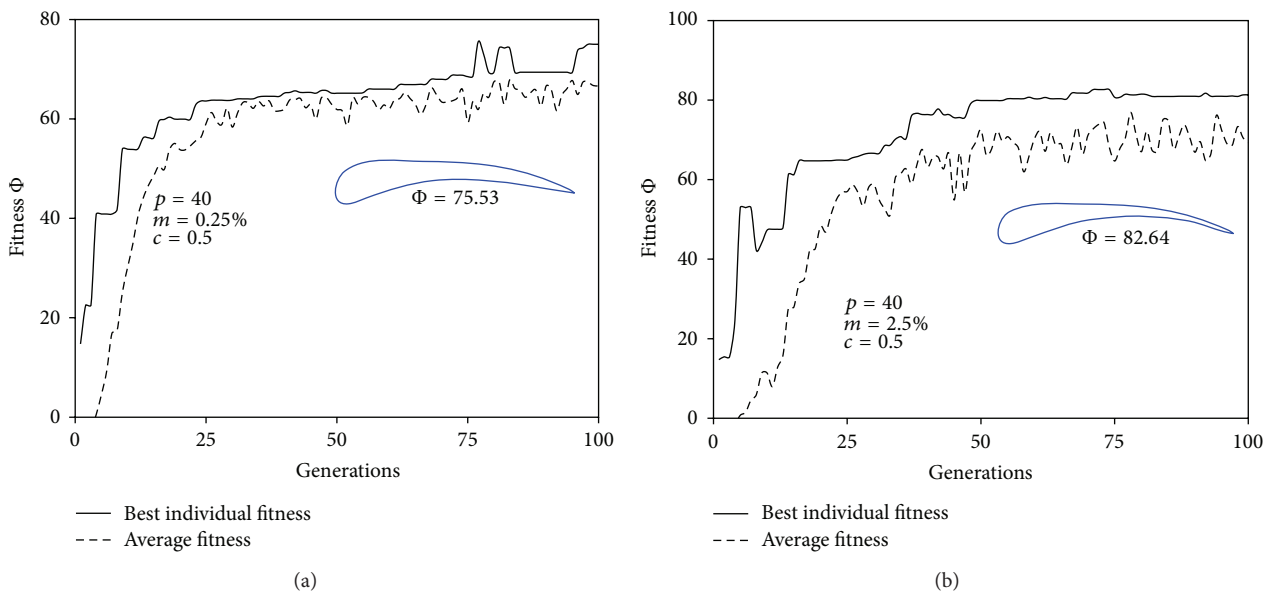

(a)

(b)

FIGURE 6: Typical evolution pattern for low and high mutation rates on the same initial population (a) for a population size of 40 individuals and a crossover rate $c = 0.5$.

crossover was used, and a simple mutation operator that flips bits at random positions along the chromosome was applied. At least three runs were performed for each combination. The termination criterion was established based on the number of generations, which was 100 in this case.

It was noted that the best solutions in populations with 40 chromosomes were obtained using a mutation rate of 2.5%, and that slightly better aptitude values could be obtained using populations of 100 individuals. However, moderate crossover rates did not appear to affect the maximum fitness significantly.

Figure 6 shows two particular runs starting from the same initial population for two different mutation rates. Both runs are representative of their respective typical evolution patterns.

When low mutation rates ($m = 0.25\%$) are used (see Figure 6(a)), the average fitness in each generation keeps near the best individual fitness. There is little difference among the

individuals in the population and so the risk to get trapped in local maxima is relatively high. This is consistent with the results collected in Table 1.

When a higher mutation rate $m = 2.5\%$ is used, also an (nonmonotonical) increase in both the average and the maximum fitness occurs. This feature indicates that although this mutation rate is relatively high, the GA does not yet behave as a random search and that constructive blocks remain. There is a wider diversity in the population and the average fitness evolves at a bigger distance below the maximum individual fitness. The former curve has now a more rugous aspect, these bigger variations from one generation to the next being consistent with a higher frequency of the random changes imposed to the genes.

High mutation rates reduce the risk of deceptive search of the GA but do not completely avoid it. In this sense the run marked with (**) in Table 1 experimented an atypical evolution shown in Figure 7. The premature appearance
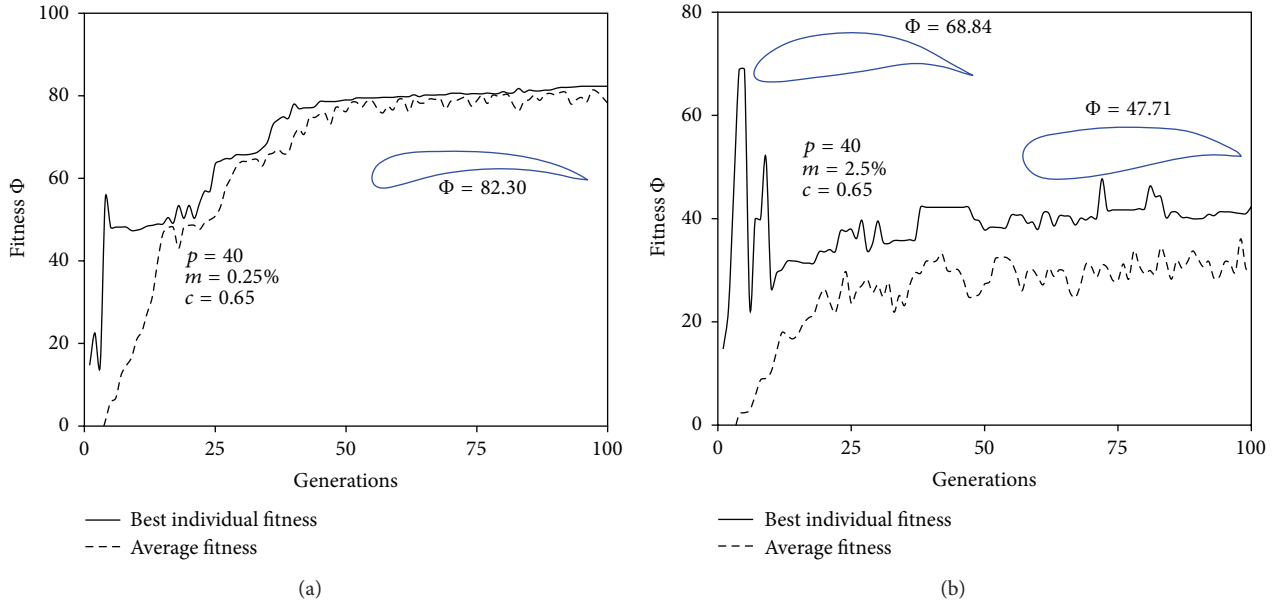
(a)

(b)

FIGURE 7: Search evolution for two runs performed on the same initial population for different mutation rates, (a) low and (b) high. The population size is 40 and the crossover rate is $c = 0.65$. The last run showed an anomalous behavior.

of a high fitness individual in the third generation (that lasted for two generations) guided the population towards nonprosperous regions where they got trapped.

The computing platform used in the analysis is a PC with 12 CPUs with 3 GHz speed and 96 Gb of RAM. In an optimization run using a population of 40 individuals along 100 generations, the number of function evaluations needed is usually in the range between 1600 and 2500 because some individuals are repeated in the next generation.

The use of elitism accelerated the convergence towards a high maximum in the landscape, as shown in the run selected in Figure 8. The populations become nearly converged around a third of the maximum number of generations. Consistent with the related reduction in the exploration, the results in Table 1 show that the use of elitism does not guarantee the achievement of the global maximum. However, in three among four runs it outperformed the best individual fitness obtained in the corresponding runs without elitism but with the same GA parameters and initial population.

4.2. Drag Minimization for a Given Value of c_l and Minimum Airfoil Thickness.
To minimize the c_d, the maximization of $1/c_d$ is tackled. The objective function was defined in the following form:

$$\Phi = \left(\frac{1}{c_d} \right) \cdot p_t(t) \cdot p_a(c_l) \cdot p_s(r_k), \quad (8)$$

where

$$p_a(c_l) = \begin{cases} \exp\left[1.2\left(c_l - 0.4\right)\right] & \text{if } c_l < 0.4, \\ \exp\left[-1.2\left(c_l - 0.4\right)\right] & \text{if } c_l \geq 0.4, \end{cases} \quad (9)$$
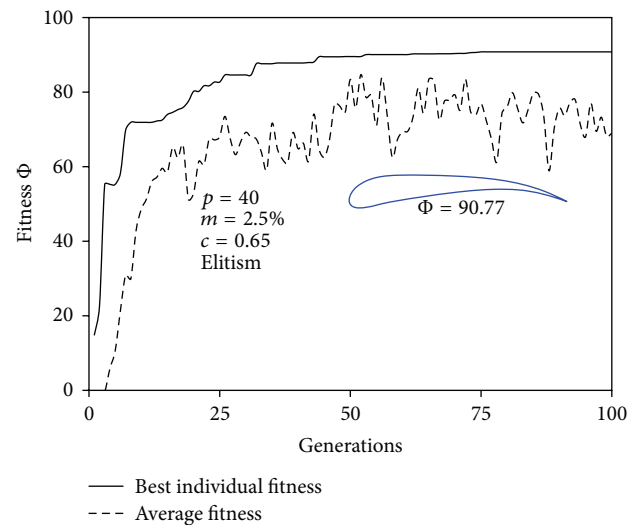

FIGURE 8: Typical convergence pattern with elitism. Search evolution for the best fitness value in Table 1.

$p_a(c_l)$ is the constraint in the lift value used to get airfoils that approximately satisfy the condition $c_l = 0.4$. For the calculations, two different initial populations of 40 individuals were used along with the same combinations of GA parameters used above. Two clear differences can be distinguished in the evolutionary behavior of these runs compared with those of the previous criterion. First, the convergence is slower (as seen in the run collected in Figure 9(b), which has a typical evolution pattern) and there is a bigger tendency to get trapped in local maxima. An expression of this is the thickness of the best airfoil in the run represented in Figure 9(a).

TABLE 2: Results for each combination of GA parameters.

Population size	40				100
Mutation rate	0.0025		0.025		0.025
Crossover rate	0.5	0.65	0.5	0.65	0.65
Best individual found	(f) 70.57	(f) 105.88	(f) 117.37	(f) 111.78	(g) 122.32
	(a) 89.31	(a) 85.68	(a) 123.18	(a) 111.63	
Best individual average	79.94	95.78	120.28	111.70	122.32

( ) Initial population.

(a)

(b)
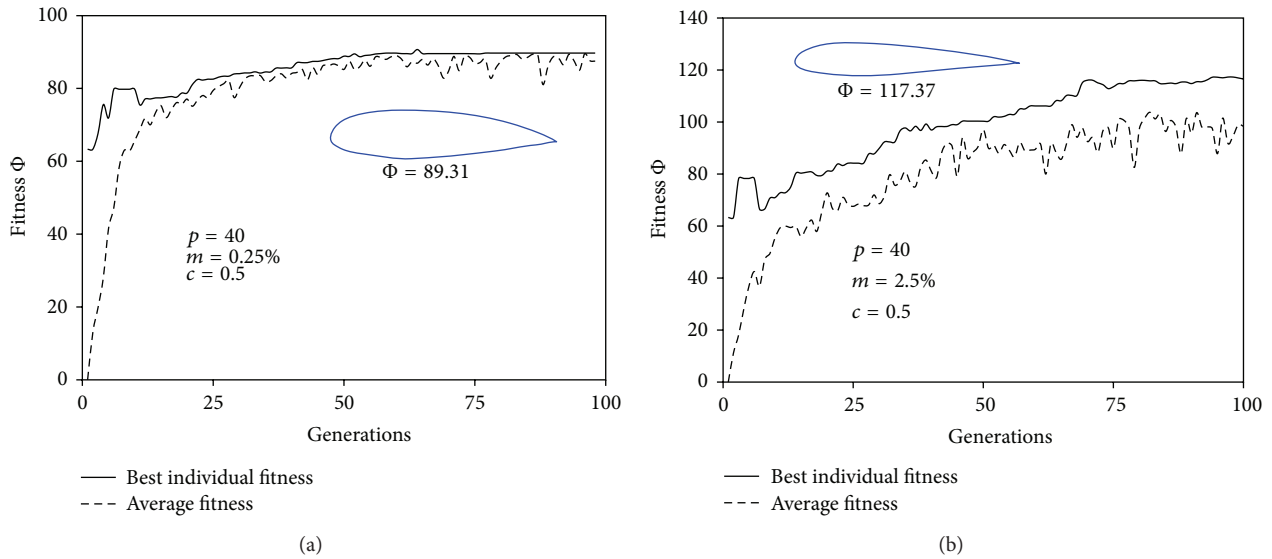
FIGURE 9: Search evolution for two runs performed on the same initial population for different mutation rates. The population size is 40 and the crossover rate is $c = 0.5$.

However, the effects of the different combinations of the GA parameters in the search appear to be qualitatively similar to the ones observed with the previous objective function. In this sense, the results of the few runs collected in Table 2 suggest that high-mutation rates give the best results, and also that bigger populations could improve the best individual fitness (see Figure 10(a)). The search evolution for the best fitness value in Table 2 is shown in Figure 10(b).

The increased tendency observed in the GA to get trapped in local maxima is consistent with the current more abrupt six-dimensional landscape. The introduction of the lift constraint in the objective function as the multiplying factor $p_a(c_l)$ defined in [0, 1] (see Figure 5) excavates valleys in the six-dimensional landscape corresponding to the six design variables that define the profile. Now, there are more local maxima, and so the difficulty for the GA to get the highest peak is increased. The authors opine that a less steep definition of the factor p_a (see Figure 5) could substantially improve the results.

Finally, the use of metamodels can reduce the CFD runs as in the case of artificial neural networks, utilized to create surrogate models. In this way, the total computational time of optimization can be reduced. However, it is important to assess the effectiveness of the metamodel using validation.

5. Conclusions

A CFD shape optimization tool was developed by integrating the genetic code known as OMEGA, which is based on David Levine's PGA Pack Library, and a commercial CFD code known as Fluent. This tool facilitated shape optimization for aerodynamic profiles. A current generation RANS turbulence model that predicts transition from laminar to turbulent flow in the boundary layer was employed in order to achieve accurate results. A fast and reliable meshing procedure was developed that generates low-cost meshes with small heights and little distortion in the boundary layer. The system robustness was verified for the specific optimization problem of an aerodynamic profile.

In addition, a preliminary experiment was performed to identify efficient parameters for the optimization algorithm with respect to this problem. Two different objective functions with nonlinear constraints were defined to find improved airfoils for two aerodynamic goals. Different
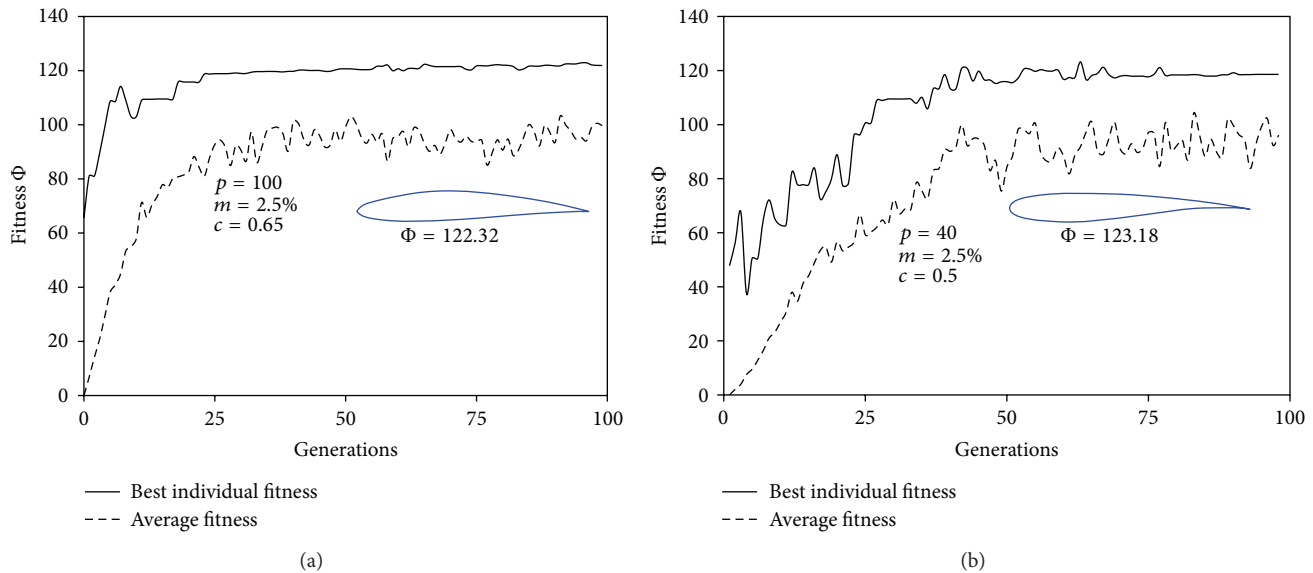
FIGURE 10: (a) Search evolution for a run with a population of 100 individuals and a crossover rate $c = 0.65$ and mutation rate $m = 2.5\%$. (b) Search evolution for the best fitness value in Table 2.

penalty functions were included in their definition to improve the convergence to optimum solutions. In both cases, the use of populations of 40 individuals along with a high mutation rate and a moderate crossover rate showed a good behavior.

The future application of this type of technique to the multiobjective optimization of airfoils with different angles of attack was considered.
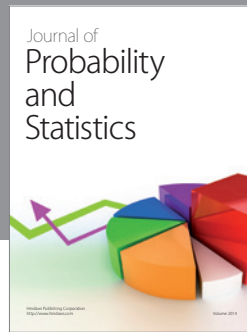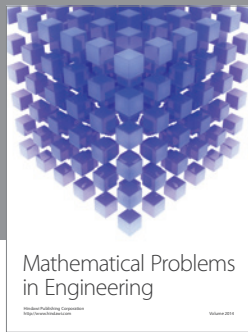
Acknowledgments

References

[1] G. Kalitzin, X. Wu, and P. A. Durbin, "DNS of fully turbulent flow in a LPT passage," *International Journal of Heat and Fluid Flow*, vol. 24, no. 4, pp. 636–644, 2003.

[2] T. A. Zaki and P. A. Durbin, "Mode interaction and the bypass route to transition," *Journal of Fluid Mechanics*, vol. 531, pp. 85–111, 2005.

[3] D. K. Walters and J. H. Leylek, "A new model for boundary layer transition using a single-point RANS approach," *Journal of Turbomachinery*, vol. 126, no. 1, pp. 193–202, 2004.

[4] Y. B. Suzen, P. G. Huang, L. S. Hultgren, and D. E. Ashpis, "Predictions of separated and transitional boundary layers under low-pressure turbine airfoil conditions using an intermittency transport equation," *Journal of Turbomachinery*, vol. 125, no. 3, pp. 455–464, 2003.

[5] F. R. Menter, R. Langtry, and S. Vlker, "Transition modelling for general purpose CFD codes," *Flow, Turbulence and Combustion*, vol. 77, no. 1–4, pp. 277–303, 2006.

[6] D. K. Walters and D. Cokljat, "A three-equation eddy-viscosity model for reynolds-averaged navier-stokes simulations of transitional flow," *Journal of Fluids Engineering*, vol. 130, no. 12, Article ID 121401, 14 pages, 2008.

[7] A. Shahrokhi and A. Jahangirian, "Airfoil shape parameterization for optimum Navier-Stokes design with genetic algorithm," *Aerospace Science and Technology*, vol. 11, no. 6, pp. 443–450, 2007.

[8] J. D. Lee and A. Jameson, "Natural-laminar-flow airfoil and wing design by adjoint method and automatic transition prediction," in *Proceedings of the 47th AIAA Aerospace Sciences Meeting*, pp. 1–25, Orlando, Fla, USA, January 2009.

[9] S. Obayashi and T. Tsukahara, "Comparison of optimization algorithms for aerodynamic shape design," *AIAA Journal*, vol. 35, no. 8, pp. 1413–1415, 1997.

[10] R. A. E. Makinen, J. Periaux, and J. Tovainen, "Multidisciplinary shape optimization in aerodynamics and electromagnetics using genetic algorithms," *International Journal for Numerical Methods in Fluids*, vol. 30, no. 2, pp. 149–159, 1999.

[11] J. Burman and B. R. Gebart, "Influence from numerical noise in the objective function for flow design optimization," *International Journal of Numerical Methods for Heat and Fluid Flow*, vol. 11, no. 1, pp. 6–19, 2001.

[12] D. Quagliarella and A. Vicini, "Viscous single and multicomponent airfoil design with genetic algorithms," *Finite Elements in Analysis and Design*, vol. 37, no. 5, pp. 365–380, 2001.

[13] F. Schmid, K. Hirschen, S. Meynen, and M. Schäfer, "An enhanced approach for shape optimization using an adaptive algorithm," *Finite Elements in Analysis and Design*, vol. 41, no. 5, pp. 521–543, 2005.

[14] S. Peigin and B. Epstein, "Robust optimization of 2D airfoils driven by full Navier-Stokes computations," *Computers and Fluids*, vol. 33, no. 9, pp. 1175–1200, 2004.

[15] J. H. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, Mich, USA, 1975.

[16] V. Gomez, I. Fernandez de Bustos, J. Aguirrebeitia, C. Angulo, and R. Aviles, "A modified genetic algorithm suitable for

nonstationary environments," in *Proceedings of the 4th World Congress of Structural and Multidisciplinary Optimization (WCSMO-4)*, pp. 1–6, Dalian, China, 2001.

[17] D. C. Wilcox, *Turbulence Modelling for CFD*, DCW Industries Inc., 1998.

[18] H. Schlichting and K. Gersten, *Boundary Layer Theory*, Springer, Berlin, Germany, 2000.

[19] D. López, C. Angulo, and L. Macareno, "An improved meshing method for shape optimization of aerodynamic profiles using genetic algorithms," *International Journal for Numerical Methods in Fluids*, vol. 56, no. 8, pp. 1383–1389, 2008.

[20] J. F. Thompson, B. K. Soni, and N. P. Weatherill, *Handbook of Grid Generation*, CRC Press, Boca Raton, Fla, USA, 1999.

[21] I. H. Abbott and A. E. von Doenhoff, *Theory of Wing Sections: Including a Summary of Airfoil Data*, Dover, New York, NY, USA, 1959.