EngOpt 2008 - International Conference on Engineering Optimization
Rio de Janeiro, Brazil, 01 - 05 June 2008.

CFD Based Wing Shape Optimization Through Gradient-Based Method

Manuel J. Garcia, Pierre Boulanger*, Santiago Giraldo

EAFIT University, Cra 49 No 7 Sur 50, Medellin, Colombia, mgarcia@eafit.edu.co, sgirald6@eafit.edu.co,
*University Of Alberta, 2-21 Athabasca Hall, T6G 2E8, Canada, pierreb@cs.ualberta.ca

1. Abstract
This paper deals with the optimization of shape of aerodynamic profiles. The objective is to reduce the drag coefficient on a given airfoil while preserving the lift coefficient within acceptable ranges. A set of control points defining the geometry are passed and parameterized as a B-Spline curve. These points are modified automatically by means of CFD analysis. A given shape is defined by an user and a valid volumetric CFD domain is constructed from this planar data and a set of user-defined parameters. The construction process involves the usage of 2D and 3D meshing algorithms that were coupled into own-code. The volume of air surrounding the airfoil and mesh quality are also parametrically defined. Some standard NACA profiles were used by obtaining first its control points in order to test the algorithm.

Navier-Stokes equations were solved for turbulent, steady-state flow of compressible fluids using the k-epsilon model and SIMPLE algorithm. In order to obtain data for the optimization process an utility to extract drag and lift data from the CFD simulation was added. After a simulation is run drag and lift data are passed to the optimization process. A gradient-based method using the steepest descent was implemented in order to define the magnitude and direction of the displacement of each control point. The control points and other parameters defined as the design variables are iteratively modified in order to achieve an optimum. Preliminary results on conceptual examples show a decrease in drag and a change in geometry that obeys to aerodynamic behavior principles.
2. Keywords: Shape optimization, shape parametrization, gradient-based methods, CFD analysis, aerodynamics.

3. Introduction
With current development on computer performance, the conjunction of multiple disciplines at early design stages has become very attractive to engineers. Coupling CFD with FEA analysis to understand fluid-solid interactions is a common thread among engineers that pursue better design processes. Faster, more effective and less expensive product design methods push such disciplines towards optimization at early design stages.

With accurate physical models virtual shape designs can be previously tested before any manufacturing process takes place. Testing a real prototype is time and resource consuming. Therefore, reducing the design search space before manufacturing a prototype is an advantage to any engineer. Altering the shape of a design by means of computational analysis and optimization can prove its usefulness in a design process. With the rapid increase in capabilities of computers and algorithms, numerical fluid simulation of aerodynamic shapes offers a low-cost alternative to evaluate its performance.

This paper presents and discusses a method for the numerical evaluation of a given shape and its possible optimization regarding its aerodynamic performance. The computational domain is obtained by means of a b-spline curve shape parametrization, the control points are described in an input file and 2D geometry is constructed with a 2D mesher. A valid CFD domain is obtained from constructing a 3D geometry from the 2D information, additional parameters and a volumetric mesher. The aerodynamic information is obtained by solving the Navier-Stokes equations using the OpenFOAM (Opensource Field Operation And Manipulation) toolkit. The method is particularly useful to narrow a design search space for an aerodynamic shape, in which case the proof of concept that this paper presents is an airfoil.

This model could be used for initial approximations to improve aerodynamic behavior of a given shape. CFD simulation could deliver accurate predictions of how a given shape would perform under various boundary conditions, characteristic that makes this method more attractive. This type of multidisciplinary design optimizations have been implemented lately and can be implemented to various industrial applications as seen in [5] and [6].

4. Shape Optimization

4.1 Optimization

Optimization in a broader sense is maximizing or minimizing some function relative to some set, often representing a range of choices available in a certain situation. The function allows comparison of the different choices to determine which might fit better to the selection criteria [12]. The maxima or minima of the function represent an important feature of the structure (s) and its maximum performance under a given scenario. For example in a wing airfoil it can represent the minimum drag coefficient C_d for a given geometric configuration (s). Then, from eq. (1), the interest is to find the minimum of $f(s)$, that is to find a s^* that fulfills the restrictions in eq.(2).

$$f(s) = \min C_d \tag{1}$$

$$f(s^*) = \min\{f(s), \forall\ s \in \mathbf{V}\} \tag{2}$$

where $\mathbf{V}$ is the space of all the admissible geometries [8].

Shape optimization is the process of reaching an optimal shape through iteration over some design parameters. The optimization process couples a geometry definition and analysis code in an iterative process to produce optimum designs subject to various constraints [16]. These constraints allow us to find a bounded set that defines the optimal shape.

Shape optimization results depend on the accuracy and relevancy of the chosen design parameters. These parameters ensure that an overall optimum is reached. Systems using multiple design variables are more demanding in terms of computational time. Here low fidelity modeling offer an alternative to explore designs in a conceptual fashion. Coarser approximations offer valuable information in situations where little initial knowledge is available [5].

4.2. Shape Parametrization

Engineers have been attracted by geometry parametrization methods in recent years, specially in the context of multidisciplinary design optimization (MDO). Samareh [14] identified three categories of parametrization methods in the context of MDO. These include the discrete approach, CAD-based approaches, and free-form deformation methods.

A correct "real world" system model must incorporate the physics and geometry-based parametrizations that represent the underlying physics of the system's aerodynamics, thermodynamics, and mechanical behavior [5]. Having a parametric and programmatic approach makes the optimization process more efficient in terms of calculation times, parameter input tasks and computational manipulation of the data. A geometry defined using simple parameters that yield a detailed definition of a surface model is an ideal shape parametrization. Such shape parametrization reduce significantly design cycle times. It is a common practice to define geometries using curves such as Bezier curves, B-Spline curves and non-uniform rational b-splines *(NURBS)*. The Fig.(1) shows a representation of a wing with Bezier control points. Bezier curves allow an easy representation of airfoil upper and lower portions [4]. Bezier curve parametrization can be defined as:

$$x(t) = \sum_{k=0}^{n} B_n^k(t) x_k \qquad y(t) = \sum_{k=0}^{n} B_n^k(t) y_k \tag{3}$$

in which the parameter t varies from 0 to 1, n is the degree of the parametrization, and

$$B_n^k(t) = C_n^k t^k (1-t)^{n-k} \tag{4}$$

is a Bernstein polynomial, $C_k^n = \frac{n!}{k!(n-k)!}$, and $P_k = (x_k\ y_k)$ $(k = 0, 1, .., n)$ is a generic control point. The grid sensitivity is affected by the type of parametrization. A CFD environment is used to evaluate the design variables for fluid-structure optimization problems. The accuracy of the obtained values when evaluating the objective function is dependent on the mesh quality and complexity of the geometry. Surface forces on an airfoil will be highly sensible to this aspect. The type of parametrization should be selected from an analysis of the sensitivity of the grid. The values of the state variables (governing equations) as well as the design parameters should be a key aspect to select parametrization, otherwise gradient errors could be induced [13].

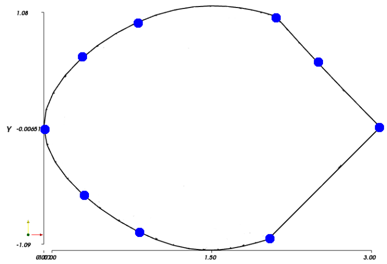


Figure 1: b-Spline shape parametrization through few control points

Leaving CAD software handle the mathematics of the geometric aspects has become a common practice. The integration of geometric kernels through programmable interfaces allow to parametrize and manipulate surfaces and bodies [5].

The balance between robustness and flexibility is a key issue to determine the parametrization method, these decisions are strongly dependent on the goal of the design activity. Although some parametrization methods may well be able to generate radical new shapes, this might not be suitable for designs where the aim is to meet an specific criterion. Elevated degrees of freedom in the control parameters will lead to a poor efficiency caused by the large search space that arises in the optimization process [16].

4.2. Objective Functions

Given a set of design parameters a target function can be defined. This target function must comply with both geometric and dynamic parameters, hence the importance of the incorporation of accurate physics models. In a CFD-based design process target functions must be evaluated several times until design specifications are met [19].

An equation that represents all the criteria defining the characteristics to improve in the design must be introduced. This equation will allow to determine the quality of the shape generated. The minimization of this expression (objective function) will lead to an optimum design [7].

A possible definition of the objective function (OF) is an aggregation function with weighted coefficients c_k -see Eq.(5)-, defining the relevance factors of each single objective F_k with K weighting coefficients c_k, k=1,...,K [10].

$$F_{aggr} = \sum_{k=1}^{K} c_k F_k(x) \tag{5}$$

In an design optimization process multiple objective functions can be involved, the complexity of the parametrization scales and different approaches must be used. In aerodynamic design it is common to use basis functions that describe the expected behavior given specific conditions, thus a single optimization function might fail to express the desired objective. One optimization approach defines the geometry of an airfoil as the linear combination of the optimization functions (in terms of aerodynamic parameters) and a set of perturbation functions, defined either analytically or numerically. These coefficients of the perturbation functions involved are then considered as the design variables. A set of such orthogonal basis functions are the functions to be evaluated to test a design alternative [16].

4.3. Optimization Methods.

The methods available to optimize a given geometry are numerous, and different strategies can be used to find the minimum of the objective function. We focus on some available and commonly used for airfoil design. A review of the optimization methods in airfoil design can be found in [15, 18].

A commonly used approach for optimization processes is evolutionary algorithms. Specifically, the use of Genetic Algorithms (GAs), which are semi-stochastic semi-deterministic optimization methods presented as natural evolution. The GAs are based on the evaluation of a set of solutions (population). Random operations of selection, crossover and mutation are applied to the population. The probability of survival of new individuals depends on their fitness: the best are kept with a high probability, the worst are rapidly discarded [6].

The steepest descent method is an iterative procedure used to accomplish optimization. Starting from some initial geometry the best direction to move towards is the direction of the steepest descent. This direction corresponds to the negative of the gradient. That is:

$$s^{n+1} = s^n - \beta \frac{\nabla(f(s^n))}{\| \nabla(f(s^n)) \|} \tag{6}$$

where β is a constant to be determined (in an optimum way) and the gradient at s^n of $f(s)$ can be approximated by forward-difference:

$$\left.\frac{\partial f(s)}{\partial s_i}\right|_{s_n} = \frac{f(s^n + \triangle s_i) - f(s^n)}{\triangle s_i} \tag{7}$$

or with a second order central-difference approximation [11]:

$$\left.\frac{\partial f(s)}{\partial s_i}\right|_{s_n} = \frac{f(s^n + \triangle s_i) - f(s^n - \triangle s_i)}{2\triangle s_i} \tag{8}$$

This method assumes the existence of only one local minimum (unimodal functions). Under the existence of several valleys the algorithm will converge to the closest local minimum -see Fig.(2)-, which is not necessarily the global minimum of this quadratic shape. This could or could not be the case and that is why it is important to analyze the general shape of the objective function. Rapid approximation to an optimum is crucial to decide upon the optimization technique. The steepest descent proves to be applicable in rapid optimization at initial design stages.

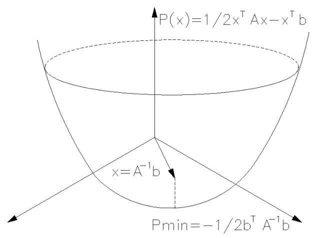


Figure 2: Steepest descent on a given space (local-global minima) [9].

6. Characteristics of Aerodynamic Profiles

6.1. Aerodynamic Forces and Moments. Aerodynamic forces are the reactions of a body moving through a fluid. Pressure and shear stress distributions on the body surface are the acting forces that make flight possible. Different force scenarios can be observed depending on the orientation of the body. Pressure (p) acts in the normal direction to the body and shear (τ) acts tangentially -see Fig.(3)-. The effect of p and τ distributions integrated over the complete body surface is a resultant aerodynamic force R and moment M on the body. R can be split into two components lift (L) and drag (D) [2].

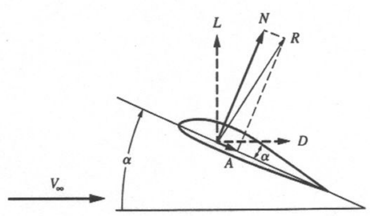


Figure 3: Wing airfoil aerodynamic forces decomposition[1].

6.2. Aerodynamic Relations. In aeronautics a wing profile or foil is a curved shaped contour able to generate lift by generation of a distributed pressure on its surface. The geometrical relations between the profile regions have specific names. Depending on the definition of these parameters the aerodynamic behavior of the foil will be altered [2].

The Lift Coefficient (C_l) relates the lift force (L) with body shape and fluid properties. The drag coefficient (C_d) relates drag force (D) with the shape of the wing and the fluid properties. The moment coefficient (C_m) relates the twisting moment (M) to a point, usually this point is set to 1/4 of the mean chord from the leading edge. For all coefficients: ρ: Density, V: Velocity, S: Reference area. The equations of the coefficients are the following.

$$C_l = \frac{2L}{\rho V^2 S} \; ; \; C_d = \frac{2D}{\rho * V^2 S} \; ; \; C_m = \frac{2M}{\rho * V^2 S}$$

The defined coefficients determine the correct behavior of wing profiles. An appropriate combination of these variables for a given attack angle and speed yield smoother results. The most common designation for standard airfoils is NACA. NACA stands for National Advisory Committee for Aeronautics. This North American institution began the early studies on aerodynamics in 1915. In those early stages, they defined a numerical designation for each airfoil –a four digit number that represented the airfoil section critical geometric properties–[3]. Nowdays five digit and six digit definitions have been implemented.

The four digit nomenclature NACA XYZZ describes the airfoil as follows: The first digit X, defines the maximum camber (or slight elevation), the second digit Y, determines the maximum camber position, the third and fourth digits ZZ, determine the thickness of the airfoil - see [17]-.

7. Method Description

The main objective is to produce a method that optimizes the shape of an aerodynamic profile. To achieve this, we must define first the set of needs towards the construction of the method. The needs are listed below:

- The geometry has to be represented as a data set that is portable between the multiple tools.
- The geometric characteristics of the body must be parametric - to allow portability and increase simplicity-.
- The geometry has to be prepared for CFD analysis, including generation of the domain and boundary conditions.
- A CFD solver must be chosen to fulfill fluid analysis requirements.
- The selected CFD solver must be adapted to allow evaluation of the optimization function.
- The optimization function must be evaluated and updated during run time.
- The geometry has to be manipulated based on the optimization method and criteria.

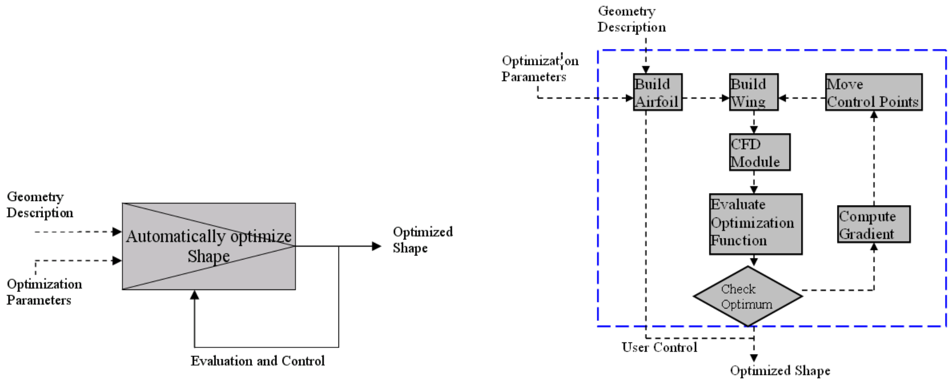
7.1. Main Function and Flows Abstraction (Black Box)

If the process is contemplated as a black box it can be simplified to basic input/output (I/O) variables. The main I/O variables will begin the definition of the structure of the method.

During any design process, feedback is an important step before any design alternative is chosen. The expertise of the designer can accelerate an optimization process avoiding stagnation in local minima. This reason supports the need to offer a degree of interaction between the optimization process and the user. The combination of an experienced designer, a solid design process and effective tools guarantee the achievement of an optimal solution.

Finding optimal solutions with the minimal effort is foremost the objective to achieve, and reducing the problems to its general form help to address their solution. Basically, the process can be reduced to the following need:

- An application that integrates a set of tools to test and analyze automatically an aerodynamic profile.



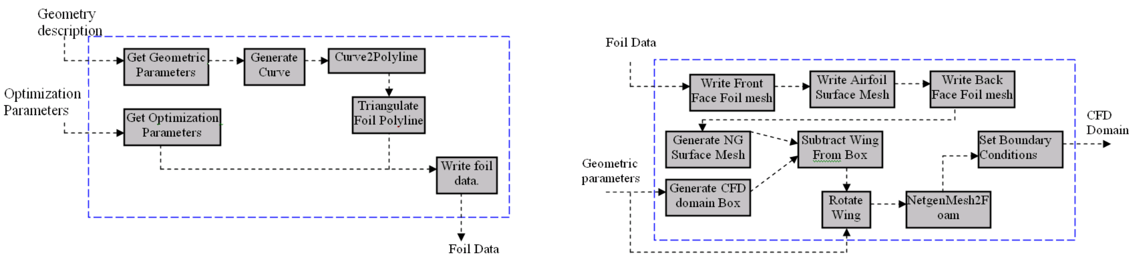
(a) Black Box Representation of the Method (b) First Level Functional Synthesis

Figure 4: Main function and first level decomposition

Then a black box representation like in Fig.(4(a)) states the design process.

7.2. Functional Synthesis
Further segregation proves itself useful in the definition of the tasks to perform. The problem proposed is presented as a general structure before contemplating any data types (or specific solution methods). Specific solutions to each particular need are only selected after a generalized segregation process. This strategy also reveals the subdivision of functions and the expected interaction between them, then data structures can be considered. The data structure must obey to the set of needs and how they develop throughout the problem, not the other way around.
In Fig.(4(b)) an overall decomposition in basic sub-modules can be appreciated. The process can be subdivided into functions which form the underlying structure of the optimization. Both geometric and optimization parameters must be set in the early stage. The first module takes care of the initial geometric representation and conservation of the parameters. A following module expands the geometry to a 3d body from parameters specified earlier. Then, one module groups CFD pre-processing, solving and post-processing using the geometry provided by the previous module. The post-processed data is fed to a module that evaluates the optimization function. At this point a check point is set to identify wether the optimum has been reached. The gradient is calculated in another module in order to set the direction for the geometric change. In a subsequent module, the gradient is used to modify the shape by moving the control points. At last this modified shape is the input for another optimization cycle. When the module structure is considered the tools that must be developed to solve each portion of the problem arise clearly.

7.2.1 2D Geometry Generation Module Function Segregation



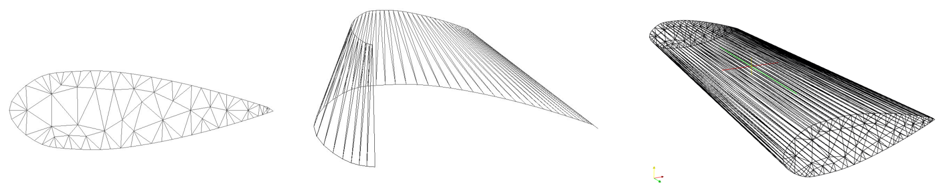
(a) Build Airfoil Module Function Decomposition (b) Build Wing Module Function Decomposition

Figure 5: Geometry generation modules function decomposition

In Fig.(5(a)) the functions to generate an airfoil from basic geometric data is presented. Geometry description and optimization parameters are defined in a text file for the algorithm to take as input, parameters such as: Control points for the upper and lower surface of the profile, length of the wing, attack angle, 2D triangulation parameters, surface and volumetric meshing parameters, maximum number of iterations and gradient related coefficients. Then, the process described in Fig.(5(a)) takes place. The geometric description and optimization parameters are read into the data structure. Following, a B-Spline curve is generated from the control points. A polyline approximation of the curve is generated for further triangulation purposes. A triangulation of the foil is generated and stored as the 2D shape as seen in Fig.(6(a)).

7.2.2 CFD Domain Generation Module Function Segregation

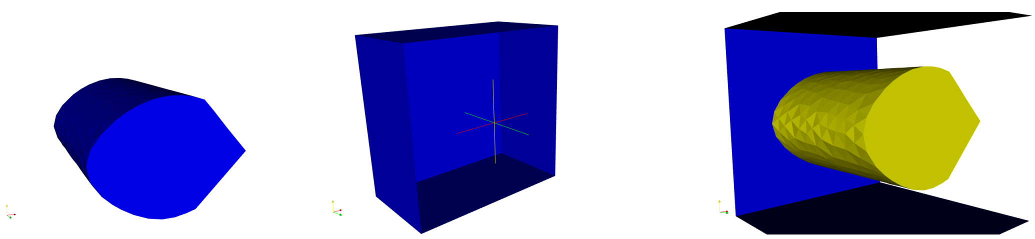
In Fig.(5(b)) the module that describes the 3D geometry generation is presented. The 2D airfoil pro-



(a) 2D front face (b) 2D face protrusion (partial) (c) 3D Geometry

Figure 6: 3D geometry generation process

duced by the previous module is passed. With this data defined, both front and back faces of the mesh can be generated. Using the boundary polyline of the triangulation a surface that connects the front and back faces is constructed -see Fig.(6)-. The surface data is passed to Netgen to obtain a surface mesh in their own format -see Fig.(7(a))-. On a parallel strain, a box is created to represent the CFD domain, and subsequently a surface mesh of this box is obtained in Netgen format -as seen in Fig.(7(b))-. A boolean subtraction is done to obtain the CFD domain -see Fig.(7(c))-. Then, the normal of the wing surface mesh are inverted and a correct CFD domain is available. If an attack angle different than 0 is defined the wing is rotated inside the domain. After the CFD surface mesh is fully defined a volumetric mesh is generated, this volumetric mesh is exported from Netgen to OpenFOAM format. Finally, the boundary conditions are set directly on the OpenFOAM mesh, patch labels, names and physical types are defined here.
In order to obtain a feasible simulation meshing processes have to be coupled with CFD domain generation. The correct balancing of the geometric and meshing parameters play an important role in the generation of a valid CFD domain.

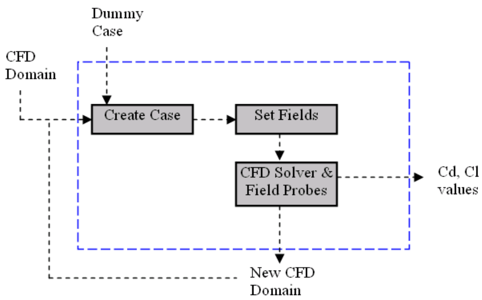


(a) Meshed 3D geometry (b) CFD domain box (c) CFD domain

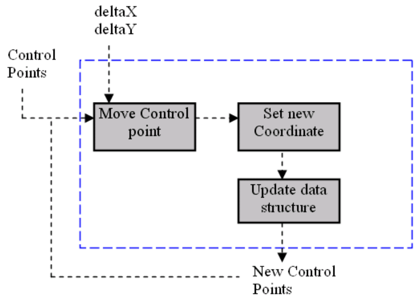
Figure 7: CFD Domain generation process

7.2.3 CFD Analysis Module Function Segregation

In Fig.(8(a)) an overview of the CFD simulation and post-processing is presented. Some basic necessary data are placed in a dummy case which is common to all simulation scenarios. This folder contains data



(a) CFD Module Decomposition



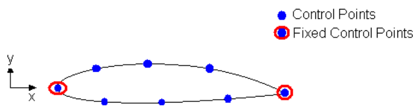
(b) Geometry Manipulation Module Function Decomposition

Figure 8: Shape analysis and modification modules

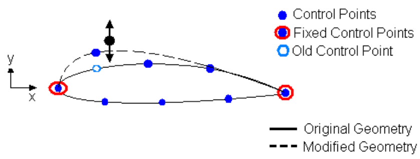
to set fluid properties and boundary conditions - like inlet wind velocity -. After setting the fields belonging to the case, the data is passed to the CFD solver. Instead of undertaking the task of developing a CFD solver, some open source codes like OpenFOAM offer a variety of solvers and tools for retrieving data. The structure of such software and its Object Oriented Programming (OOP) nature enables the user to implement a customized CFD solver up to his needs. In the present method an adaptation of a turbulent steady-state solver was implemented. The Semi-Implicit Method for Pressure Linked Equations (SIMPLE) algorithm is used in this particular solver. The solver was further modified to allow field data extraction to calculate drag and lift coefficients. Each simulation scenario is run using the modified simpleFoam solver, adding an utility to calculate drag an lift after the simulation has finished.

7.2.4 Geometry Manipulation Module Function Segregation

In Fig.(8(b)) the basic functions to alter the shape are identified. Once a case has been fully analyzed its shape must be altered to obtain a new set of control points for a new analysis cycle. The current control points are moved a distance deltaY (currently deltaX is fixed to zero) and then a structure is updated in order to regenerate the whole CFD domain. The distance to move the points can also be defined in the input file. Here in this module and in the way the shape is modified the performance of the application can be of great sensitivity. The process can be observed in Fig.(9).



(a) Control point defined shape



(b) Control point manipulation

Figure 9: Control point definition and manipulation process

7.2.5 Optimization Module Function Segregation

In Fig.(10) the layout for the core of the optimization process is displayed. The drag and lift coefficients calculated on the post-processing stage of the CFD module are taken as input. The objective function that has been previously defined is evaluated in this stage. The values of the previous evaluated objective function for each node are used in order to obtain the gradient via finite difference. The magnitude of the gradient is then calculated to normalize the direction in which the points should be moved. With the parameters for the optimization the set of points is moved in the opposite direction of the gradient and the set of points acquire a new shape.

When the described modules are implemented a CFD-Based optimization method is created. In or-

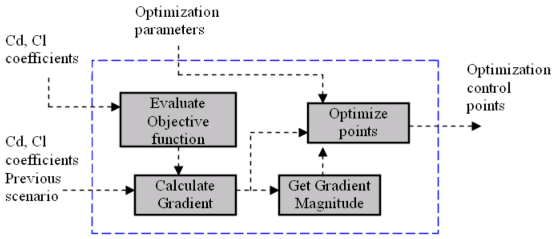


Figure 10: Shape Optimization Module Function Decomposition

der to achieve an optimal shape, modifications have to be done on the geometry to estimate the possible change in shape. The manipulation of the geometry can increase in difficulty as the number of parameters and control points rise. Increased numbers of control points will generate larger search spaces, since each control point displaced means a complete CFD simulation with the correspondent objective function evaluation. Greater search spaces of the design variables will affect the convergence and speed of the method. The importance of the parameters used to displace the control points in the opposite gradient direction is noticed also in the rate of convergence and speed of the method. Variables such as meshing parameters, discretize the CFD domain in smaller elements (therefore increasing the number of them) increasing the time consumed by each simulation to complete.

8. Preliminary Results

The proof of concept test showed a significant change of shape, such behavior is not as visible in other examples. The test shown had the following configuration:

1. **Control Points:** 6 control points for the upper and lower surface.
2. **B.C. and Solver:** Inlet velocity of 50 m/s at 0° attack angle and k-epsilon turbulence model with default coefficients for air at $0°C$. Besides the inlet and outlet, the walls had symmetry condition to emulate outside flow.
3. **Mesh:** Between 4000 to 5000 elements composed each CFD domain.

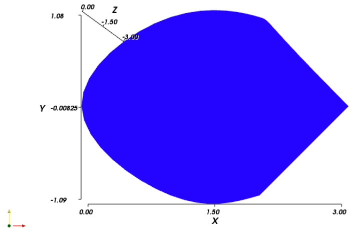
It was important to have a smooth shape defined with as few control points as possible. Fewer control points reduce the size of the search space. Smaller search spaces yield faster results. It was important that the Re number was high enough to ensure steady-state turbulent simulation scenario. Remember that each displaced control point generates a new CFD domain with approximately the same number of elements. For each iteration, each CFD scenario must be solved and evaluated.

The presented example was run on a dual core 2.0 GHz CPU with 2 GB of ram yielding solution times around 2 minutes for the whole simulation process using reduced CFD domains. Here in Fig.(11) the initial geometry and its final state is shown, when both shapes are superimposed the change in shape is noticeable. The behavior of the drag coefficient through the iterative process can be observed in Fig.(12) and Table (1). Some perturbations during the optimization process increase the drag coefficient -thus the behavior-, moving a control point is a tool to evaluate wether the drag coefficient increased or decreased and use it to obtain the direction of the gradient. With the gradient direction a modified shape can be obtained and finally, an optimized shape throughout the iterations is reached.

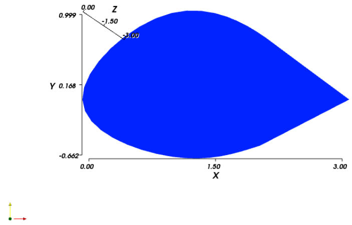
9. Conclusions And Future Work

9.1 Conclusions.

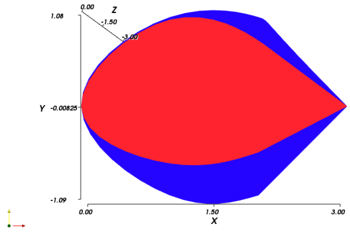
The process of constructing the shape in 3d from 2d control points and later on the CFD domain became a time consuming task. A parametrical definition of the geometry was implemented and allows the generation of various shapes with a valid CFD domain. The CFD domain size is quite large when treating outside flows (like the ones when simulating an airfoil moving through air) and causes simulation times to increase considerably.



(a) Initial shape



(b) Final shape



(c) Shape comparison

Figure 11: Proof of concept optimization process

A method for the CFD analysis and simulation of aerodynamic profiles is presented on this work. The aim was to develop an infrastructure for shape optimization of an aerodynamic profile using a gradient-based method. The solution varies depending on the selection of the optimization criteria and constants. When pursuing an objective like reducing drag, lift is proportionally reduced as well since they are directly related. Weight coefficients play an important role in maintaining such balance.

An optimal combination of forces is the objective of the method. An accurate definition of the constants that define the weight of the objectives becomes a crucial step. When only a minimization of drag is treated, the geometry differs greatly from the initial, but the significant loss of lift leads to profiles with no capacity to sustain flight conditions optimally. For this matter, a penalization of lift and minimal area acceptable were introduced.

The possibility of exploring drag minimization exclusively in aerodynamic design poses interesting results. Using the developed method with switches for specific design scenarios (automotive, hydrodynamic) offers feasible and improved shapes.

One of the main concerns during the tests was the quality of the mesh. The high curvature involved in the leading edges of airfoils forced the need for increased number of surface elements, this translated into elevated number of elements in the CFD volumetric mesh, inducing high demands on computational operations and amounts of disk space.

Coarser meshes allow the designer to have valuable initial approximations. If an accurate mesh grade is defined, even if coarse, the designer can obtain shape optimization within minutes. The ability of the code to define a shape with few control points presents and important advantage when the optimization process occurs. The dimension of the search space plays a vital role when defining the evolution of the shape, having fewer control points reduces the size of the search space, enabling the designer to obtain initial approximations faster. As well, fewer control points defining a given shape compromise the complexity of the geometry.

The optimized shape varies depending on the operation conditions derived from the CFD boundary conditions. For the same shape under different velocities or attack angles, the optimum shape achieved is diferent.

Setting the conditions of the turbulent solver are highly case-dependent as well. In this work the k-epsilon model was used due to its robustness in practical application. When high Reynolds numbers

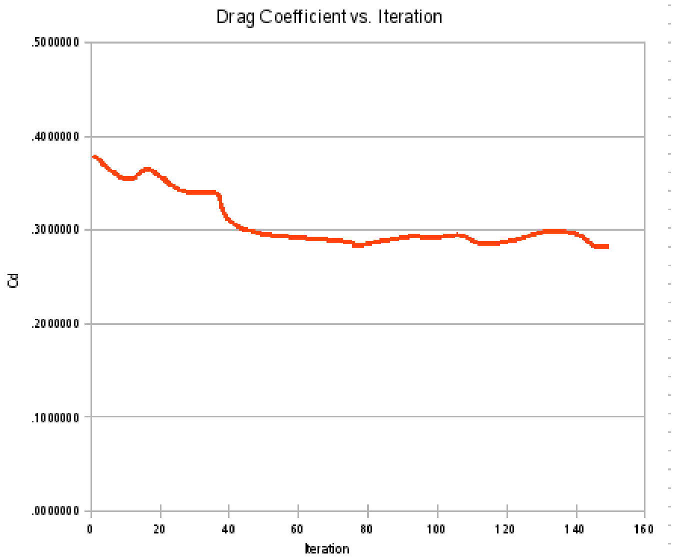


Figure 12: Evolution of drag coefficient in the optimization process

Table 1: Results of the drag coefficient through the iterations.

Iteration	Cd
0	0.3804260
1	0.4233110
2	0.3858450
:	:
12	0.3535870
:	:
38	0.3372310
:	:
149	0.2815080

lead to the assumption of a steady state flow condition, the k-epsilon model is used widely.

9.2 Future Work.
Further testing with shapes other than airfoils are required to prove the flexibility of the application.

Since the optimum shape reached is higly dependent on the operation conditions (derived from the CFD boundary conditions), an interactive process to select this conditions would prove itself useful.

After proving the concept of CFD-based shape optimization in a single CPU, the code will undergo changes to allow parallelization. Parallelization will allow more complex geometries, more refined domains and shorter simulation times.

The current meshing process has to be reviewed for efficient adaptation to CFD optimal conditions, possible use of more specific airfoil hexahedra mesh generation will be explored using OpenFOAM's embedded mesher “blockMesh“.

Exploring the integration with a CAD platform under Linux (the native environment of the application) in C/C++ programming language would scale the usability of this application. Currently the API of ProEngineer offers an alternative to such challenge.

References

[1] . @Aerospaceweb. Lift and drag vs. axial force. Web, 2007. Available at: `http://www.aerospaceweb.org/question/aerodynamics/q0194.shtml`.

[2] J. D. Anderson. *Fundamentals Of Aerodynamics*. Mc Graw-Hill, 3rd edition, 2001.

[3] R. E. Bilstein. *Orders of Magnitude: A History of the NACA and NASA, 1915-1990*. National Aeronautics and Space Administration, Office of Management, Scientific and Technical Information Division, 1st edition, 1989.

[4] J.-A. Désidéri and A. Janka. Multilevel shape parameterization for aerodynamic optimization - application to drag and noise reduction of transonic/supersonic business jet. *European Congress on Computational Methods in Applied Sciences and Engineering ECCOMAS*, 2004.

[5] C. Dye, J. B. Staubach, D. Emmerson, and C. G. Jensen. Cad-based parametric cross-section for gas turbine engine mdo applications. *Computer Aided Design and Applications*, 4(1-4):509–518, 2007.

[6] B. Epstein and S. Peigin. Optimization of 3d wings based on navier-stokes solutions and genetic algorithms. *International journal of Computational Fluid Dynamics*, 20(2):75–92, 2006.

[7] L. Ferrano, J.-L. Kueny, F. Avellan, and L. Pedretti, Camille Tomas. Surface parameterization of a francis runner turbine for optimum design. *22nd IAHR Symposium on Hydraulic Machinery and Systems*, 2004.

[8] M. Garcia. *Fixed Grid Finite Element Analysis in Structural Design and Optimisation*. PhD thesis, Department of Aeronautical Engineering, The University of Sydney, March 1999.

[9] M. J. Garcia. Lecture notes on numerical analysis, 2007.

[10] K. C. Giannakoglou. Design of optimal aerodynamic shapes using stochastic optimization methods and computational intelligence. *Progress in Aerospace Sciences*, 38:43–76, 2002.

[11] R. Haftka and Z. Güdal. *Finite Element Method: volume 3, Fluid Dynamics*. Kluwer Academic Publishers, 3rd edition, 1992.

[12] R. T. Rockafellar. Fundamentals of optimization, 2007.

[13] I. Sadrehaghighi, R. E. Smith, and S. N. Tiwari. Grid sensitivity and aerodynamic optimization of generic airfoils. *journal of Aircraft*, 32(6), 1995.

[14] J. A. Samareh. Survey of shape parameterization techniques for high-fidelity multidisciplinary shape optimisation. *AIAA*, 39(5):877–884, 2001.

[15] W. Shyy, N. Papila, R. Vaidyanathan, and K. Tucker. Global design optimization for aerodynamics and rocket propulsion components. *Progress in Aerospace Sciences*, 37:59–118, 2001.

[16] W. Song and A. J. Keane. A study of shape parameterisation methods for airfoil optimisation. *AIAA*, page 1, 2004.

[17] J. Trapp and R. Zores. Applet to generate naca 4 digit airfoils. Web, November 2007. `http://www.pagendarm.de/trapp/programming/java/profiles/NACA4.html`.

[18] A. Vicini and D. Quagliarella. Airfoil and wing design through hybrid optimization strategies. *AIAA*, pages 27–29, 1998.

[19] J. Wu, K. Shimmei, K. Tani, S. Joushirou, and K. Niikura. Cfd-based design optimization for hydroturbines. *journal Of Fluids Engineering*, 129/159, 2007.