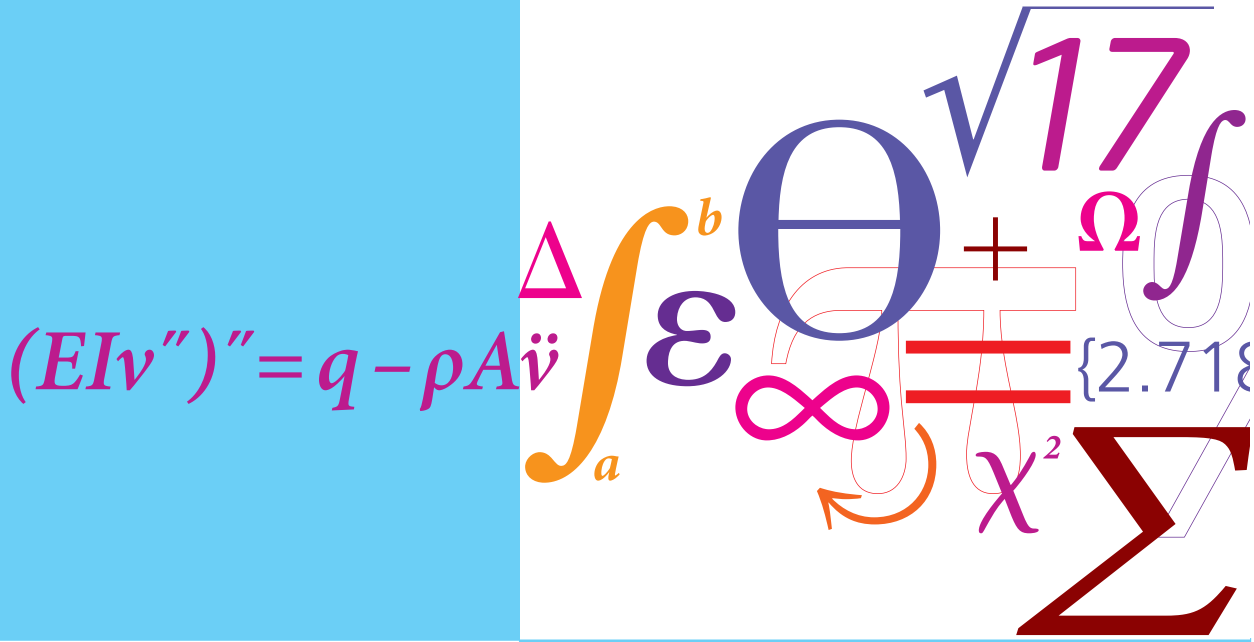
Automatic 2D Airfoil Generation, Evaluation and Optimisation using MATLAB and XFOIL

Master Thesis



Xavier Mauclère

MEK-FM-EP 2009-11

August 2009

DTU Mechanical Engineering
Department of Mechanical Engineering

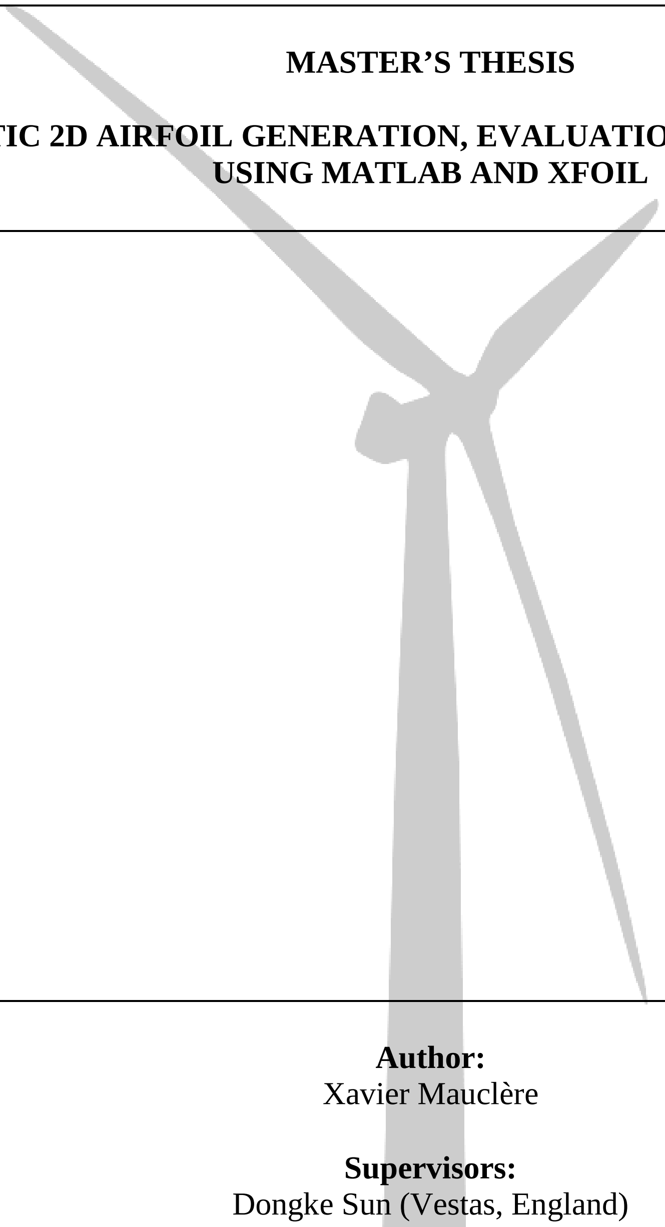
DTU Mechanical Engineering
Section of Fluid Mechanics
Technical University of Denmark

Nils Koppels Allé, Bld. 403
DK- 2800 Kgs. Lyngby
Denmark
Phone (+45) 45 25 43 00
Fax (+45) 45 88 43 25

www.mek.dtu.dk

MASTER’S THESIS

AUTOMATIC 2D AIRFOIL GENERATION, EVALUATION AND OPTIMISATION USING MATLAB AND XFOIL



Author:
Xavier Mauclère

Supervisors:
Dongke Sun (Vestas, England)
Ton Lubrecht (INSA Lyon, France)
Martin O. L. Hansen (DTU, Denmark)

Table of content

Abstract

A method for design of wind turbine airfoils is presented. This method is based on direct numerical optimisation of a B-spline representation of the airfoil shape. The XFOIL is used as flow solver during the optimisation. The generation and the optimisation processes are implemented in MATLAB, and an interface between MATLAB and XFOIL is created.

The airfoil profiles are represented by a B-Spline formulation based on 14 control points. The control point distribution, detailed in the present work, is able to reproduce different airfoils with various geometries with a good accuracy (maximal geometry difference less than 0.5%).

The design method is demonstrated by the improvement of the MH43 profile performances at different angles of attack under specific conditions: Reynolds number = 3.10^6 and Mach number = 0.2. First the lift-drag ratio at 2 degrees angle of attack is increased from 93 to 120. Then, lift-drag ratio at 5 degrees angle of attack is increased from 110 to 130. Finally, the lift-drag ratio is increased simultaneously at both angles: 2 and 5 degrees. This results in an increase in performances of the initial MH43 profile over the entire range of angles of attack.

Nevertheless, the program is not completed yet. Other validations have to be performed and several parameters still have to be implemented.

1. Introduction

Airfoil design for wind turbine blades is one of the keys of wind turbine development. Studies have shown that choosing airfoils with suitable characteristics decreases the cost of the produced energy. Design of modern airfoil profiles has to consider different variables such as the structural stiffness and the material that are most of the time conflicting with the aerodynamic performances. For example, thin airfoils are desirable for their high lift-to-drag ratio and roughness insensitiveness but can not guaranty the required blade stiffness for large rotors.

The airfoils currently used are pretty old like the NACA family which was originally developed for airplanes during the World War II. However aerodynamic performances and geometrical characteristics of airplane airfoils do not necessarily fit the wind turbine requirements. The NACA airfoils were developed for high Reynolds numbers and suffer from significant laminar separation bubbles when used on wind turbines at lower Reynolds numbers. The effect of these bubbles on the airfoil performance varies significantly with the roughness. Thus, airfoil families have to be developed specifically for wind turbine blades to improve rotor efficiency.

Methods for airfoil design can be classified into two categories: direct and inverse design. The direct airfoil design methods involve a given airfoil cross section and compute the pressure distribution under specified conditions (angle of attack, Reynolds number, Mach number). Thus the given airfoil shape is evaluated and then can be modified to improve its performances. Kristian S. Dahl and Peter Fuglsang have done remarkable work in this field [1] [6].

The variety of methods for inverse design works the other way round: the airfoil characteristics are imposed to determine the corresponding airfoil shape. In other words, the airfoil surface flow is defined at specified operational conditions and the shape that will generate these surface conditions (pressure distribution) is found. Until now, most of airfoil designs have been developed by use of inverse design methods. Nevertheless, the latter are not studied in this project since the objective is to develop a direct design method.

The baseline objective of the present work is to automatically generate airfoil profiles using control points and evaluate their aerodynamic performances. A further objective is to optimise the airfoils for the use on wind turbine blades. The design method is based on direct numerical optimisation. The airfoil generation, evaluation and optimisation is integrated using the MATLAB facilities. MATLAB is a robust and easy time software with a simple interface. Furthermore, it has a large panel of optimisation functions that are used for the present work. Airfoil characteristics are computed thanks to the flow solver XFOIL. The automatic airfoil optimisation process has to be directed from MATLAB. Hence, one of the keys of this project is to develop an interface between MATLAB and XFOIL.

The work presented here has been largely inspired by the article *Design of the Wind Turbine Airfoil Family RISØ-A-XX* written by Kristian S. Dahl and Peter Fuglsang on December 1998 at Risø National Laboratory, Roskilde, Denmark [1]. Basically, the aim of this project is to try to reproduce their work using a MATLAB interface instead.

2. Airfoil Generation

The aim of this section is to accurately generate any kind of airfoil profile on MATLAB. Since the airfoil shape is going to be modified and evaluated many times during the optimisation process, it has to be generated using as few input parameter as possible. Indeed describing the airfoil profile using one or two hundred coordinate points is not the easiest way to modify its entire geometry with minimum computational cost. Furthermore it is fundamental to obtain a smooth airfoil shape to guaranty better aerodynamic performance.

2.1. Geometry description

In the present work, the airfoil profile is generated using a B-spline formulation based on a set of control points. The B-spline method extrapolates a curve from given control points with respect to a given degree (or order) k:

Given *m* real valued t_i, called *knots*, with:

$$t_0 \leq t_1 \leq \ldots \leq t_{m-1}$$

A B-spline of degree k is a parametric curve [5],

$$S : [t_0, t_{m-1}] \rightarrow \Re^2$$

composed of a linear combination of basis B-splines $b_{i,k}$ of degree k

$$S(t) = \sum_{i=0}^{m-k} P_i b_{i,k}(t) \;,\; t \in [t_{k-1}, t_{m-k}]$$

The P_i's are called control points. There are m−k+1 control points. The m-k+1 basis B-splines of degree k can be defined using the Cox-de Boor recursion formula:

$$b_{j,0}(t) = \begin{cases} 1 & if \cdot t_j \leq t \leq t_{j+1} \\ 0 & otherwise \end{cases}$$

$$b_{j,k}(t) = \frac{t - t_j}{t_{j+k} - t_j} b_{j,k-1}(t) + \frac{t_{j+k+1} - t}{t_{j+k+1} - t_{j+1}} b_{j+1,k-1}(t)$$

If two knots t_j are identical, any resulting indeterminate forms 0/0 are deemed to be 0. Note that j+k+1 can not exceed m-1, which limits both j and k.

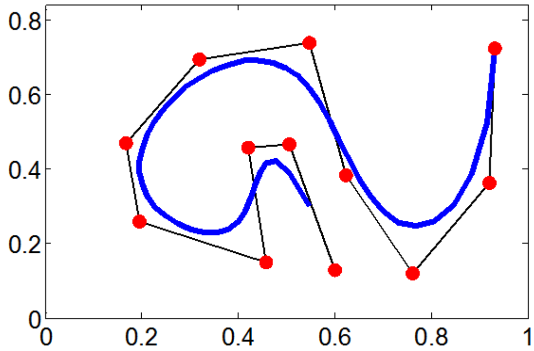


Figure 1: Illustration of a B-spline curve (blue line) on MATLAB generated by 12 control points (in red) with k=5

k determines how large a part of the entire curve is altered when a single control point is moved. High values of k lead to a smoother curve whereas small values of k tend to create abrupt angles. The figure bellow shows two B-spline curves that represent two airfoil shapes generated with 14 control points but different degrees of smoothness. It can be seen that for k=1 the B-spline curve (in blue) crosses all the control points. Thus the curve is very angular. Whereas for k=5, the B-spline curve is very smooth.

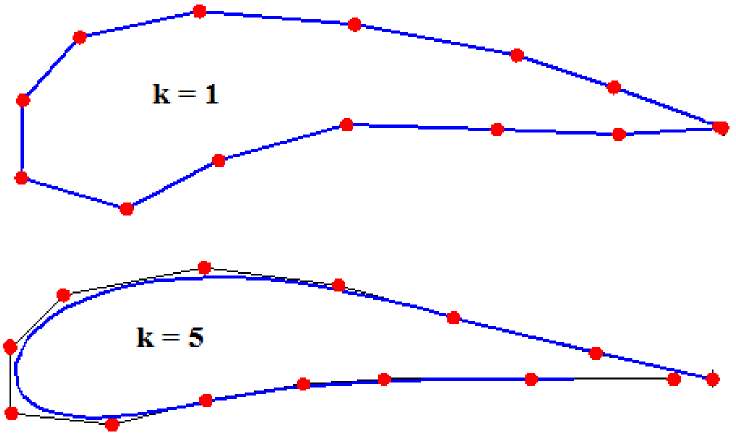


Figure 2: Effect of the degree k on a B-Spline curve generated by 14 control points on MATLAB

Thanks to the B-spline method, only the control point coordinates represent the design variables of the airfoil generation. For an airfoil shape, it is chosen that the single B-Spline curve is defined counter clockwise from the trailing edge. The deal now is to determine the number of control points to be used, their distribution as well as their degrees of freedom in order to be able to accurately reproduce any kind of airfoil shape with acceptable computational costs.

2.2. Validity of the geometry description

The control point distribution is supposed to be able to accurately generate any airfoil shape. Indeed it must not avoid the optimisation process finding an optimised profile. In the article *Design of the Wind Turbine Airfoil Family RISØ-A-XX* written by Kristian S. Dahl and Peter Fuglsang [1], 13 control points are chosen to generate the B-spline curve with a degree k=5. In the present work, it is decided to use the same degree (k=5) for it provides a very smooth airfoil. Nevertheless, the number of control points is studied in this section: 4 control point distributions are compared in order to determine which one of them is the best to reproduce various airfoil shapes. From a general point of view, most of the control points are fixed in the x-direction in order to limit the number of design variables. The different distributions are detailed in the figures below.

- **1st control point distribution:**

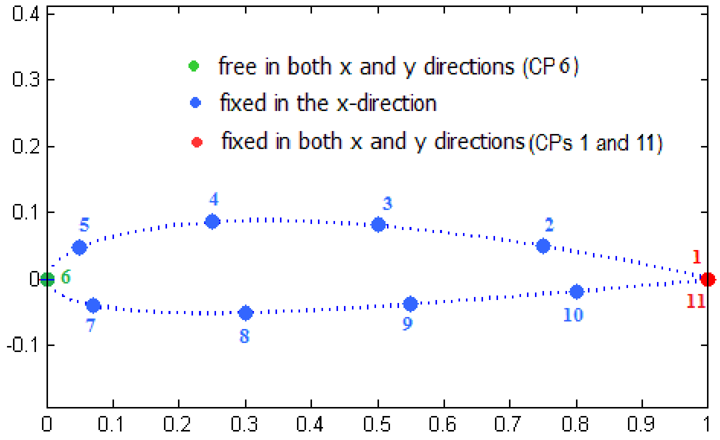


Figure 3

The B-spline curve is defined by **11** control points with different degrees of freedom.

Only **one** control point is used to define the **leading edge**. This point is free to move in both x and y directions.

Number of design variables = 10

- **2nd control point distribution:**

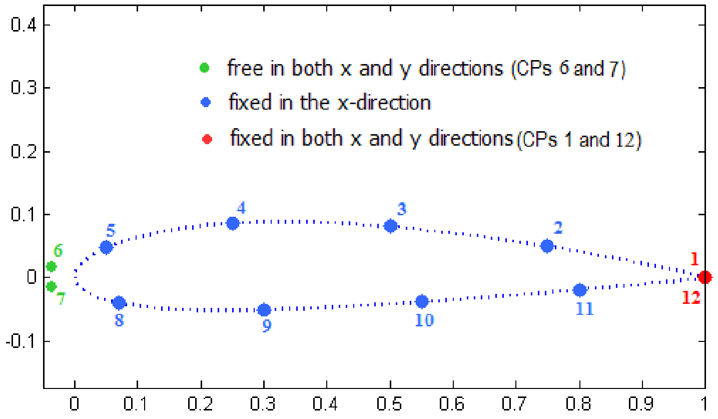


Figure 4

The B-spline curve is defined by **12** control points with different degrees of freedom.

In order to increase the accuracy at the leading edge, **two** control points are used to define this region. These points are free to move in both x and y directions.

Number of design variables = 12

- **3rd control point distribution:**

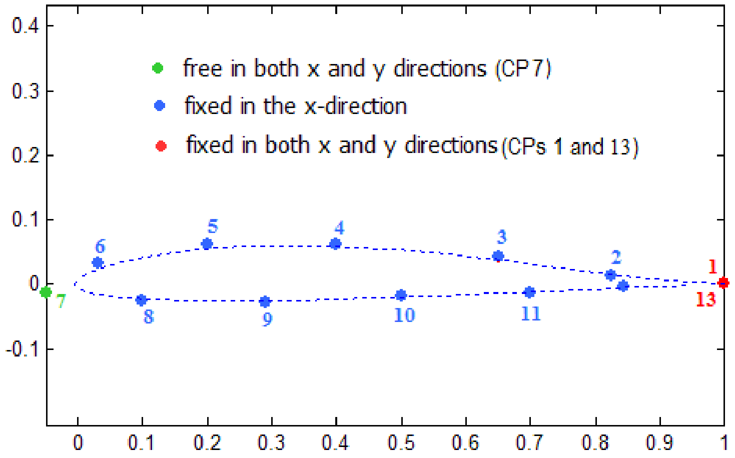


Figure 5

The B-spline curve is defined by **13** control points with different degrees of freedom.

Only **one** control point is used to define the **leading edge**. This point is free to move in both x and y directions.

The distribution is refined at the leading and trailing edges.

Number of design variables = 12

- **4th control point distribution:**

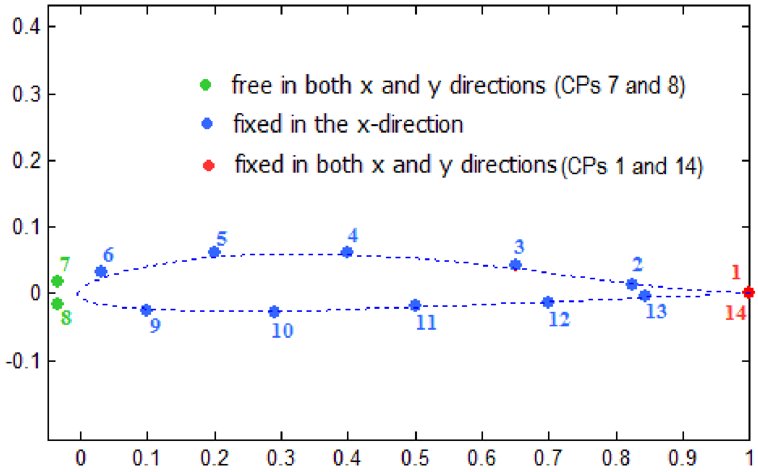


Figure 6

The B-spline curve is defined by **14** control points with different degrees of freedom.

In order to increase the accuracy at the leading edge, **two** control points are used to define this region. These points are free to move in both x and y directions.

The distribution is refined at the leading and trailing edges.

Number of design variables = 14

2.2.1. Geometrical flexibility

A simple test is performed with MATLAB to test the flexibility of the distributions described above. In principle, any physically realistic shape should be achievable to allow design from an initial arbitrary shape. Hence, the aim of the test is to try reproduce different existing airfoil profiles using a given distribution and evaluate the maximal geometrical difference between the two profiles.

The geometrical difference between the two curves is evaluated vertically at 100 different points not equally spaced along the chord length. Indeed the density of evaluation points is higher at the leading and trailing edges. The test is directed by the MATLAB function ***lsqnonlin*** developed to solve nonlinear least-squares (nonlinear data-fitting) problems. The input parameters are the coordinates of the control points (according to their degrees of freedom) and the output is the vector of geometrical differences computed at each evaluation point. The function minimizes the difference between the two curves by gradually moving the control points (see figure below).

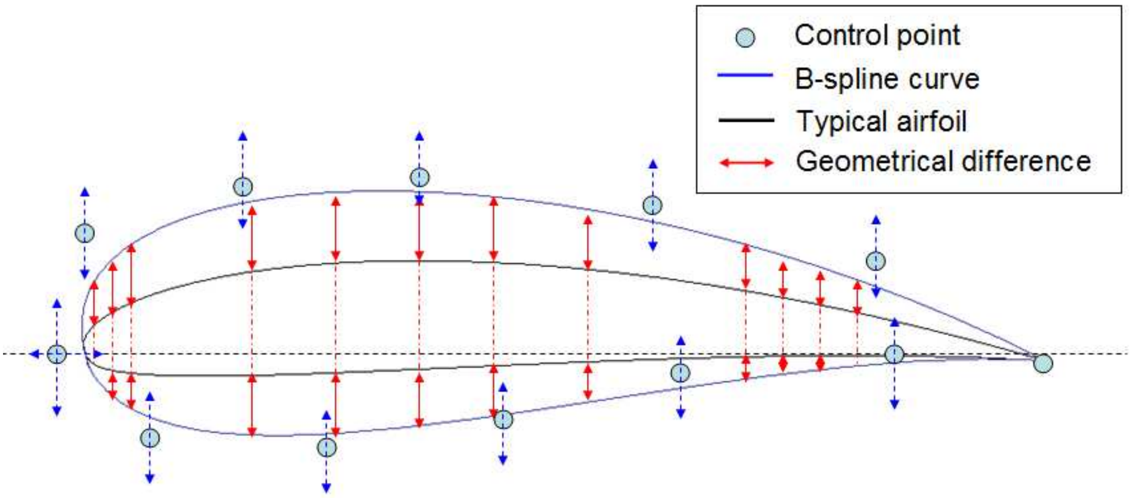
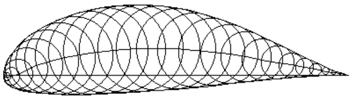


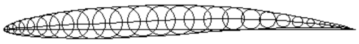
Figure 7: Test of the validation of the control point distribution

Four existing airfoils are selected to test the control point distributions. It is important to choose very different shapes in order to have relevant results. Therefore it is decided to reproduce:

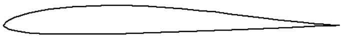
- the MH126 profile (thick airfoil)



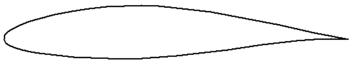
- the MH121 profile (thin and cambered airfoil)



- the MH43 profile (thin airfoil)



- the NACA 642415 (thick and cambered airfoil)



For each case, the maximal geometrical difference δ_{max} and its location on the profile are given in the table below. The airfoil profile is normalized.

	1st distribution (11 CPs)		2nd distribution (12 CPs)		3rd distribution (13 CPs)		4th distribution (14 CPs)	
	δ_{max}	**Loc.**	δ_{max}	**Loc.**	δ_{max}	**Loc.**	δ_{max}	**Loc.**
MH126	0.0052	TE	0.0054	TE	0.0049	LE	0.0021	TE
MH121	0.0017	LE	0.0014	TE	0.0014	TE	0.0014	TE
MH43	0.0030	LE	0.0027	LE	0.0027	LE	0.0016	LE
NACA 642415	0.0039	LE	0.0036	LE	0.0031	LE	0.0023	TE

TE = Trailing Edge
LE = Leading Edge

It can be deduced from the results above that the 4th control point distribution has the best geometrical flexibility. The maximal geometrical difference remains around 0.2% no matter the airfoil shape. Moreover, the location of δ_{max} is generally at the trailing edge. This means that the accuracy is even better at the leading edge. The latter affects a lot the aerodynamic characteristics of the airfoil profiles and thus has to be accurately defined. From a manufacturing point of view, it is difficult to build sharp and thin trailing edge. There is always a certain error between the theoretical and the real shape. This is why there is no particular interest in improving too much the accuracy at the trailing edge. Further more, including more control points or more degrees of freedom tends to increase significantly the computational costs. Obviously, better distributions based on 14 control points (CPs) can be found by changing slightly the CPs x-coordinates and may be the topic of a complementary study. Nevertheless the results shown above are considered accurate enough.

The plots corresponding to the selected CP distribution (the 4th distribution) are shown in the figures below. The initial airfoil (blue dashed line) is a NACA 2414. It is an arbitrary choice. Any airfoil shape could have been taken. On each plot is shown the maximal geometrical difference δ_{max}.

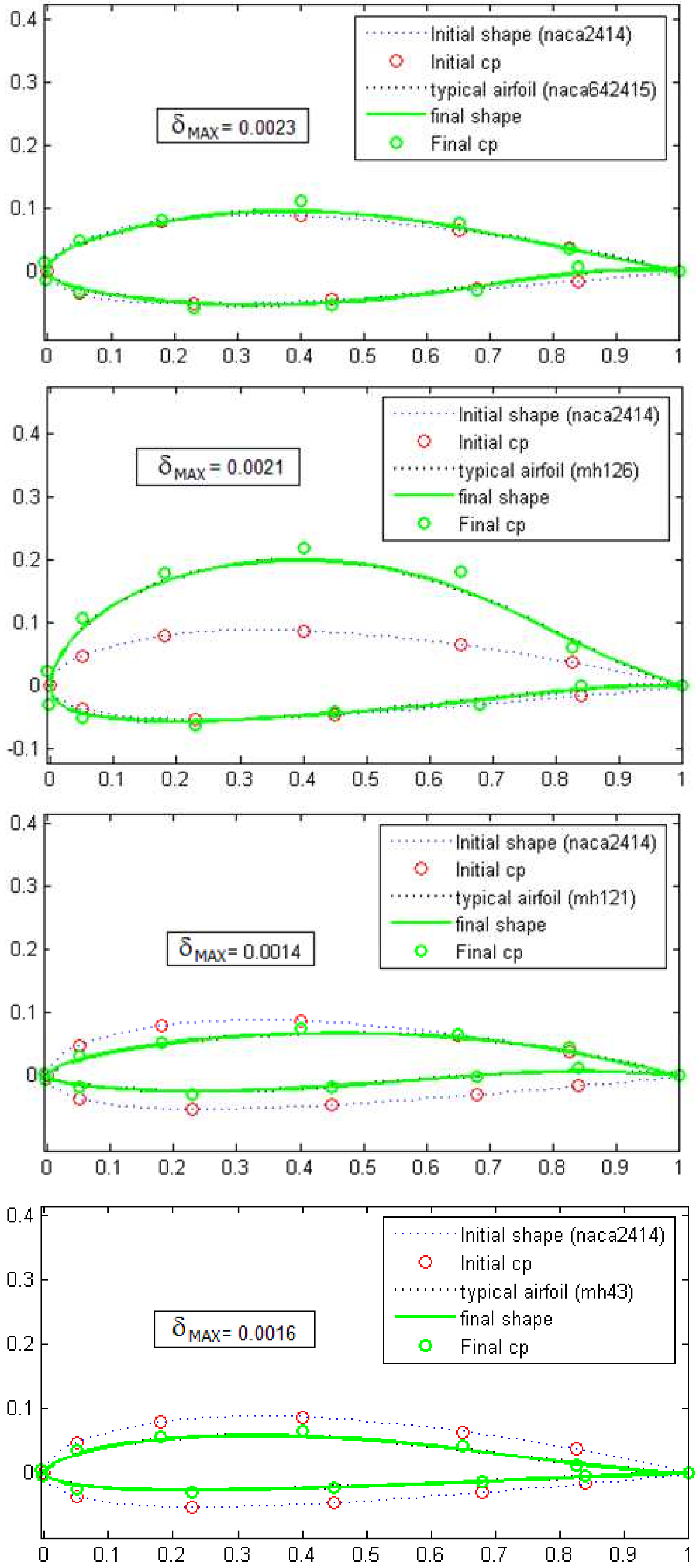


Figure 8: Flexibility of the CP distribution based on 14 CPs, k=5

As it is said above, the leading edge has to be accurately reproduced by the B-spline curve for most of the aerodynamic characteristics are induced by this region. Using two control points instead of one to define the leading edge increases significantly the accuracy of the curve. This is shown in the figure below. The green curve represents the airfoil generated by the 4th control point distribution which uses two control points at the leading edge (see Figure 6). Whereas the red curve represents the airfoil generated by the 3rd control point distribution which uses only one control point at the leading edge (see Figure 5). The green curve is much closer than the typical profile (dashed line). In this example, the typical profile to be reproduced is the MH121.

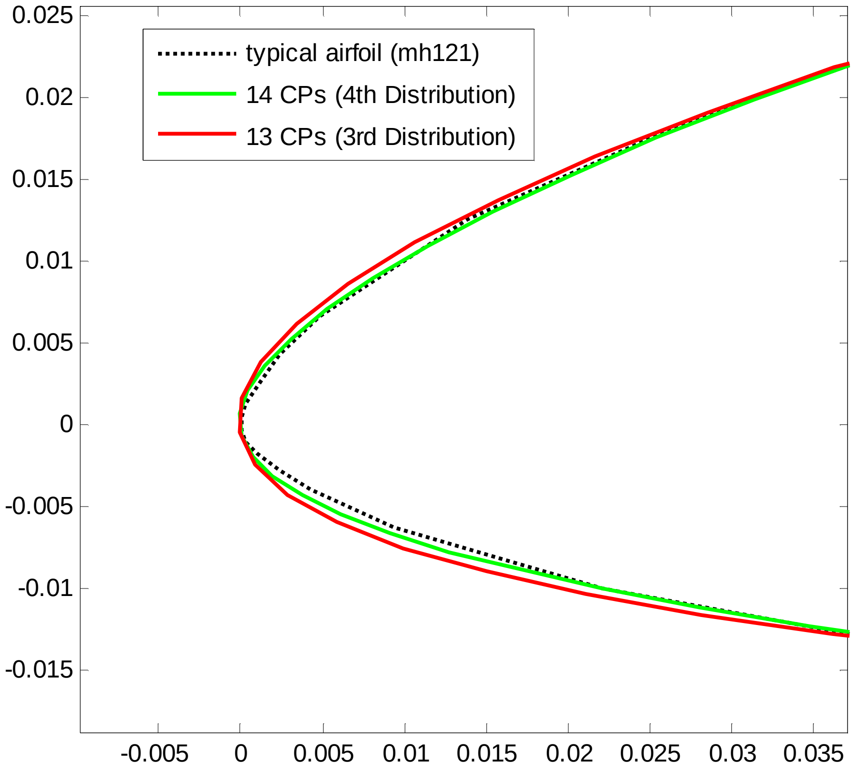


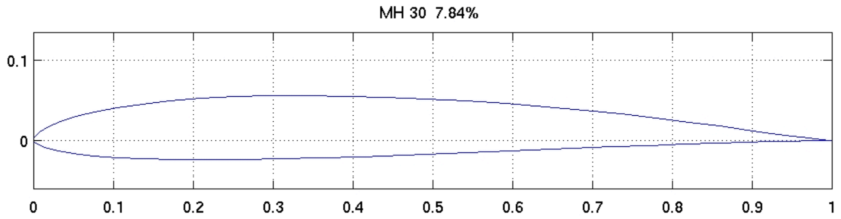
Figure 9: Comparison of the 3rd and the 4th control point distribution on the leading edge definition

To summarize, the 4th control point distribution is used to reproduce any kind of airfoil shape. This distribution is defined by 14 control points with two at the leading edge (free in both x and y directions) and has proved to have a good geometrical flexibility. It is now interesting to check whether this distribution is also able to reproduce the airfoil performance.

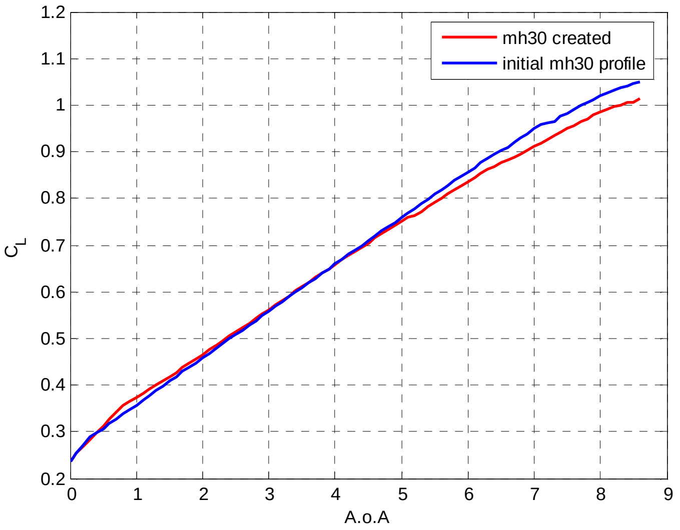
2.2.2. Airfoil performance check

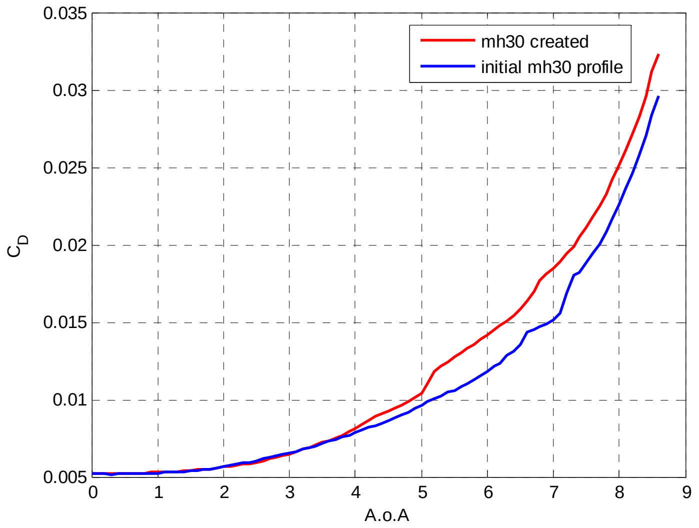
In the figures below, the red curve represents the characteristics of the airfoil profile generated by the 14 control points whereas the blue curve represents the real profile. The profile characteristics are determined with XFOIL. Two airfoil profiles are studied here: the MH30 and the MH43.

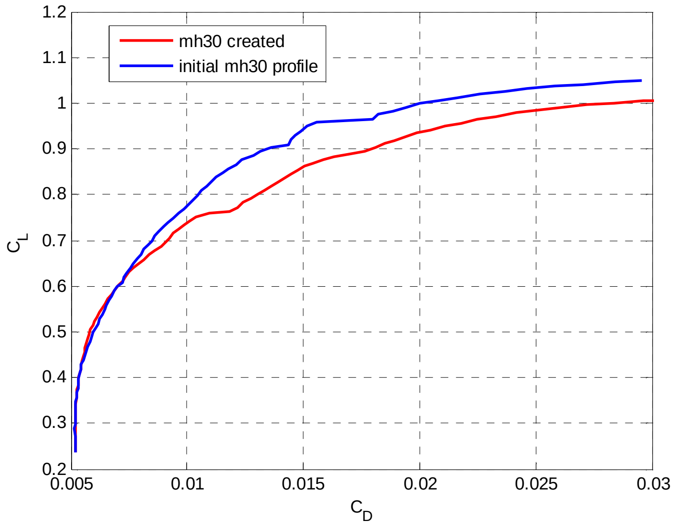
- **MH30 profile**



The control point distribution can reproduce the MH30 profile with a maximal geometry difference of 0.0017

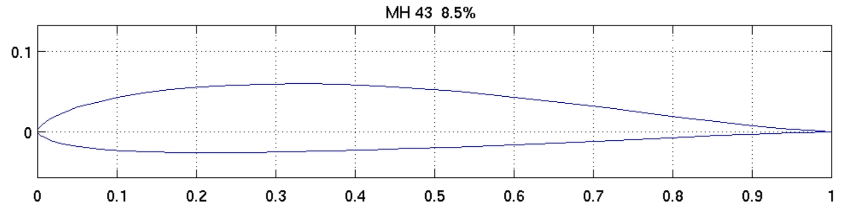




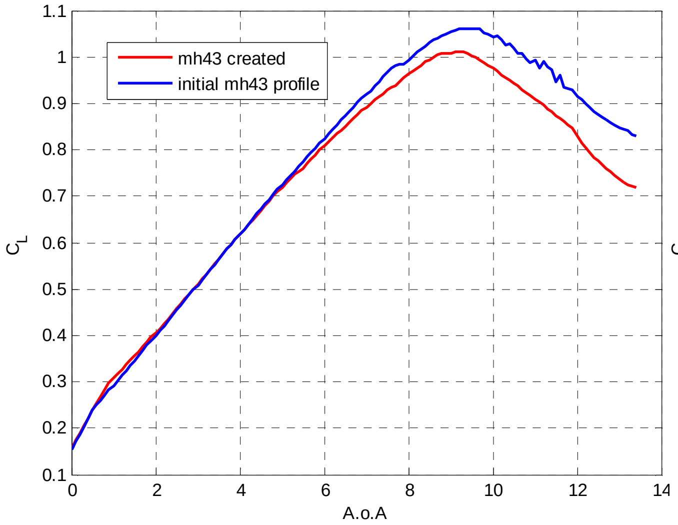


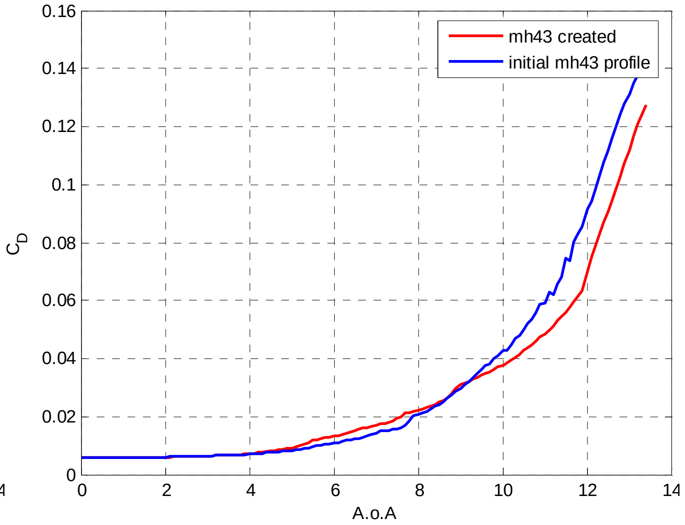
It can be seen that the MH30 airfoil created with the control point distribution has very close characteristics from the real MH30 at small angle of attack (until 5 degree angle of attack). For large angle of attack the polar differs but keeps the same progression.

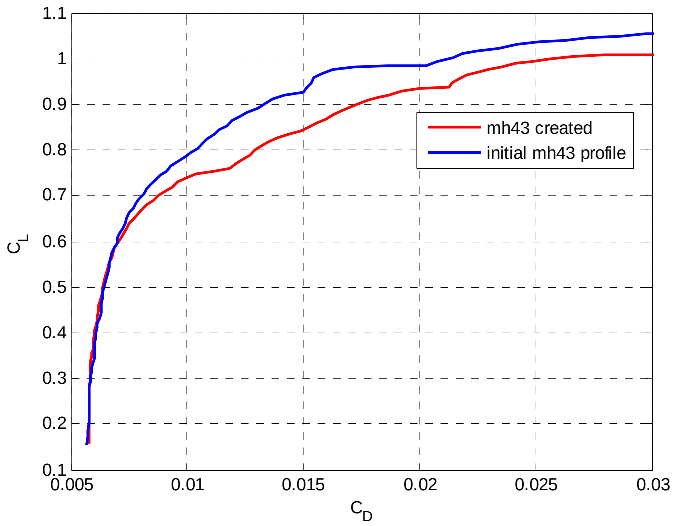
- **MH43 profile**



The control point distribution can reproduce the MH30 profile with a maximal geometry difference of 0.0016







In the same way, the B-spline representation of the MH43 profile has very close characteristics from the real MH43 profile at small angle of attack (until 6 degree angle of attack). For large angle of attack the polar differs but keeps the same progression. This difference in performance comes from the small geometrical difference between the real and created profiles. Nevertheless the airfoil performance remains close. Hence, it can be concluded that the 4th control point distribution (see Figure 6) is flexible enough to accurately reproduce the geometry and the performance of any airfoil shape.

3. Airfoil Evaluation

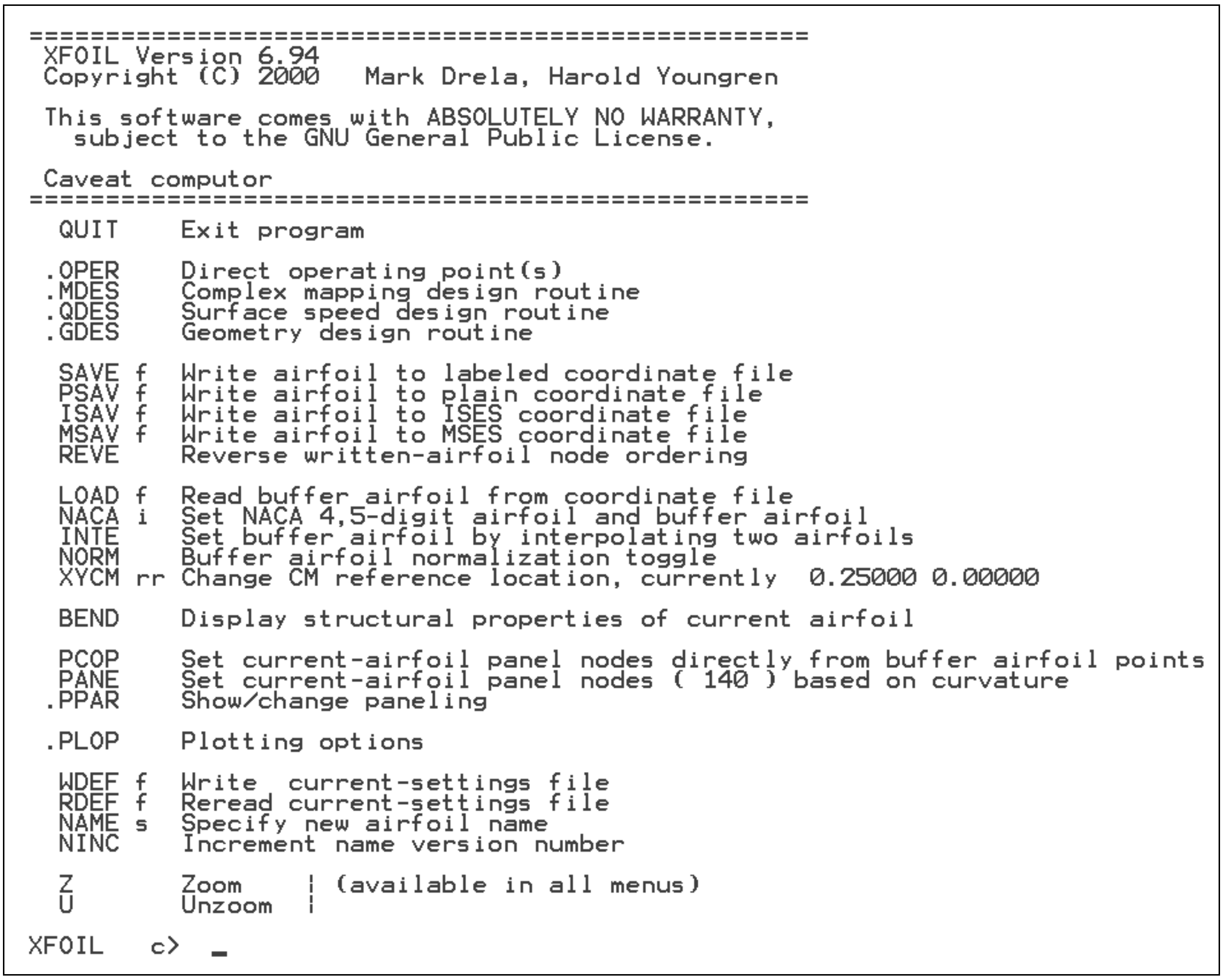
It has been shown in the previous part that any airfoil shape can be accurately reproduced using a B-spline formulation based on 14 control points. The next step now is to be able to evaluate the performance of the newly generated profile.

3.1. Flow solver

The task of the flow solver is to analyse and determine the airfoil characteristics. Since the optimisation process requires many evaluations and modifications of the airfoil profile, the computational costs of the flow solver has to remain relatively small. These are the reasons why XFOIL is used as flow solver in the present work.

XFOIL is a free software, written by Mark Drela. XFOIL is based on high order panel methods with the fully coupled viscous/inviscid interaction method used in the ISES code developed by Drela. It is a fast and robust solver.

The figure below shows the XFOIL home page. The different available commands are listed in the window and can be executed by typing their corresponding code.



```
===================================================
 XFOIL Version 6.94
 Copyright (C) 2000   Mark Drela, Harold Youngren

 This software comes with ABSOLUTELY NO WARRANTY,
   subject to the GNU General Public License.

 Caveat computor
===================================================
  QUIT    Exit program

 .OPER    Direct operating point(s)
 .MDES    Complex mapping design routine
 .QDES    Surface speed design routine
 .GDES    Geometry design routine

  SAVE f  Write airfoil to labeled coordinate file
  PSAV f  Write airfoil to plain coordinate file
  ISAV f  Write airfoil to ISES coordinate file
  MSAV f  Write airfoil to MSES coordinate file
  REVE    Reverse written-airfoil node ordering

  LOAD f  Read buffer airfoil from coordinate file
  NACA i  Set NACA 4,5-digit airfoil and buffer airfoil
  INTE    Set buffer airfoil by interpolating two airfoils
  NORM    Buffer airfoil normalization toggle
  XYCM rr Change CM reference location, currently  0.25000 0.00000

  BEND    Display structural properties of current airfoil

  PCOP    Set current-airfoil panel nodes directly from buffer airfoil points
  PANE    Set current-airfoil panel nodes ( 140 ) based on curvature
 .PPAR    Show/change paneling

 .PLOP    Plotting options

  WDEF f  Write  current-settings file
  RDEF f  Reread current-settings file
  NAME s  Specify new airfoil name
  NINC    Increment name version number

  Z       Zoom    | (available in all menus)
  U       Unzoom  |

XFOIL   c> _
```

Tableau 1: XFOIL home page

In the present work, only the viscous analysis is performed. According to [2], the boundary layers and wake are described with a two-equation lagged dissipation integral boundary layer (BL) formulation and an envelope e^n transition criterion, both taken from the transonic analysis/design ISES code. The entire viscous solution (boundary layers and wake) is strongly interacted with the incompressible potential flow via the surface transpiration model. This permits proper calculation of limited separation regions. The drag is determined from the wake momentum thickness far downstream. A special treatment is used for a blunt trailing edge which fairly accounts for base drag.

The total velocity at each point on the airfoil surface and wake, with contributions from the freestream, the airfoil surface vorticity, and the equivalent viscous source distribution, is obtained from the panel solution with the Karman-Tsien correction added. This is incorporated into the viscous equations, yielding a nonlinear elliptic system which is readily solved by a full-Newton method as in the ISES code. Execution times are quite rapid, requiring about few seconds.

For a given angle of attack α, Reynolds number and Mach number, *XFOIL* provides the pressure distribution, *CP(x)*, lift coefficient, *Cl*, and drag coefficient, *Cd*. Basically, XFoil finds the flow around the airfoil for the given angle of attack and a window pops up showing the pressure distribution, the section lift coefficient, the section moment coefficient and the angle of attack. The drag coefficient and the lift-to-drag ratio are also presented (see Figure 10). Both viscous and inviscid flow distribution are shown on the pop up window. The dashed lines represent the inviscid flow distribution. This provides an easy way to compare viscous and inviscid flow. In addition, numerous boundary layer parameters are calculated. Transition was modeled by the e^n method with n=9. The boundary layer is outlined around the airfoil in Figure 10.

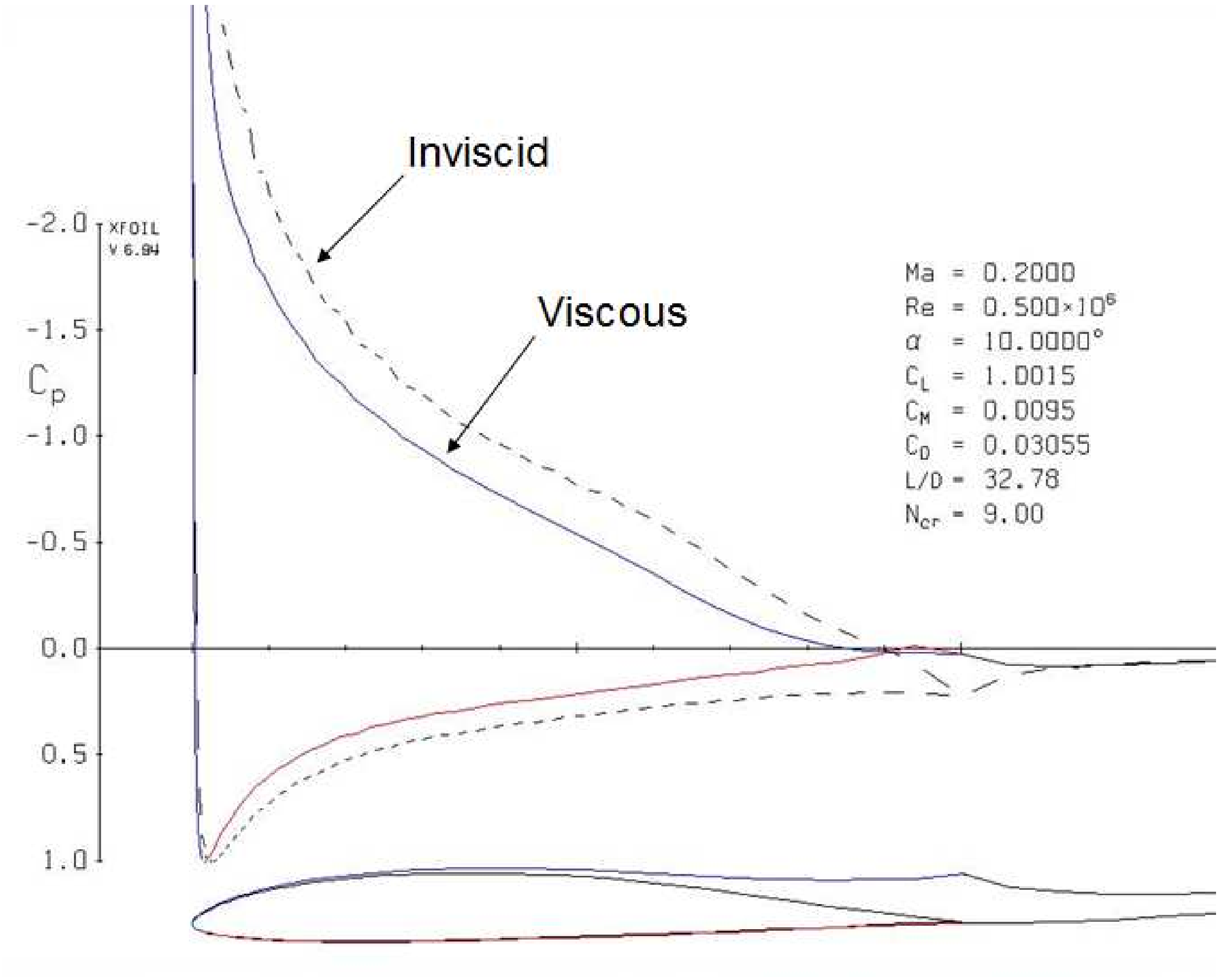


Figure 10: Example of results returned by XFOIL

The command screen (see Figure 11) returns the last iteration which provides more data about the airfoil like the point of transition to turbulent flow in the upper and lower surfaces. It also provides CDf and CDp, the friction drag and pressure drag respectively.

```
Side 1  free  transition at x/c =  0.0054   14
Side 2  free  transition at x/c =  1.0000   44
  1   rms: 0.5724E-05   max: 0.1692E-04   D at   17  1
      a = 10.000      CL =  1.0015
     Cm =  0.0095     CD =  0.03055   =>   CDf =  0.00531
  CDp =  0.02524
```

Figure 11: Illustration of the XFOIL command screen

XFOIL can be directed by typing manually a list of command or by loading an input file. This method is very convenient for iterative tasks. The input file is a text or data file that has to contain exactly the same syntax as the command window. The table below shows an example of input file that directs XFOIL to compute the characteristics of a given airfoil at Re=500000, Mach=0.2 for a sequence of angles of attack. The results are stored in a data file.

Table 2: Example of Input file	**Meaning**
`NORM`	Normalise the airfoil to be loaded
`LOAD Airfoils/airfoil.dat`	Load the airfoil coordinates
`OPER`	Toggle the operational mode
`VISC 5e+005`	Toggle the viscous mode and set Re=5e5
`M 0.2`	Set Mach Number = 0.2
`ITER`	Change viscous-solution iteration limit
`200`	
`PACC`	Toggle auto point accumulation to active polar
`OUTputs/output.dat`	Store the output in the file named output.dat

`ASEQ 0 10 0.2`
`PACC`
	Prescribe a sequence of alphas and launch the flow analysis

`QUIT`	Quit XFOIL

An example of XFOIL output file is shown is the table below. The output file can be either a text or a data file depending on the user's needs. In this example, the characteristics of a given airfoil are computed for a sequence of angles of attack (form 0 to 10 with an increment of 1) at Re=200000 and Mach=0.2.

Table 3 : Output file returned by XFOIL

```
       XFOIL         Version 6.94

 Calculated polar for:

 1 1 Reynolds number fixed          Mach number fixed

 xtrf =   1.000 (top)        1.000 (bottom)
 Mach =   0.000     Re =     0.200 e 6     Ncrit =   9.000

  alpha     CL        CD       CDp       CM     Top_Xtr  Bot_Xtr
 ------- -------- --------- --------- -------- -------- --------
   0.000   0.2339   0.00930   0.00403  -0.0549   0.9375   1.0000
   1.000   0.3642   0.00891   0.00354  -0.0570   0.8925   1.0000
   2.000   0.4591   0.00866   0.00323  -0.0514   0.8245   1.0000
   3.000   0.5558   0.00873   0.00319  -0.0465   0.7187   1.0000
   4.000   0.6512   0.00967   0.00367  -0.0421   0.5566   1.0000
   5.000   0.7419   0.01153   0.00490  -0.0381   0.3554   1.0000
   6.000   0.8300   0.01419   0.00685  -0.0348   0.1641   1.0000
   7.000   0.9080   0.01931   0.01158  -0.0296   0.0405   1.0000
   8.000   0.9777   0.02947   0.02250  -0.0233   0.0192   1.0000
   9.000   1.0292   0.04073   0.03523  -0.0161   0.0153   1.0000
  10.000   0.9988   0.05640   0.05244  -0.0064   0.0156   1.0000
```

The two last columns returned by XFOIL are the location of the transition point (laminar to turbulent transition) on both airfoil top and bottom faces. The position is normalised with the chord length.

In case of no convergence, no value is written in the output file. In most of the cases, convergence can not be achieved due to boundary layer separation and stall regions. Moreover, specific points can lead to numerical error (no convergence). In this case, preliminary computations have to be performed to force the convergence (if it is possible). Indeed when performing viscous analysis calculations, it is always a good idea to sequence runs so that alpha (the angle of attack) does not change too drastically from one case to another [2].The Newton solution method always uses the last available solution as a starting guess for a new solution, and works best if the change from the old to the new solutions is reasonably small. For this reason, it is best to perform difficult calculations by gradually increasing the angle of attack.

3.2. Validity of XFOIL results

Some polars were computed and compared with results from the literature in order to validate XFOIL results. This is done with two airfoil profiles: the MH126 and the MH30 at different Reynolds numbers. The profile geometries are shown directly on the polars.

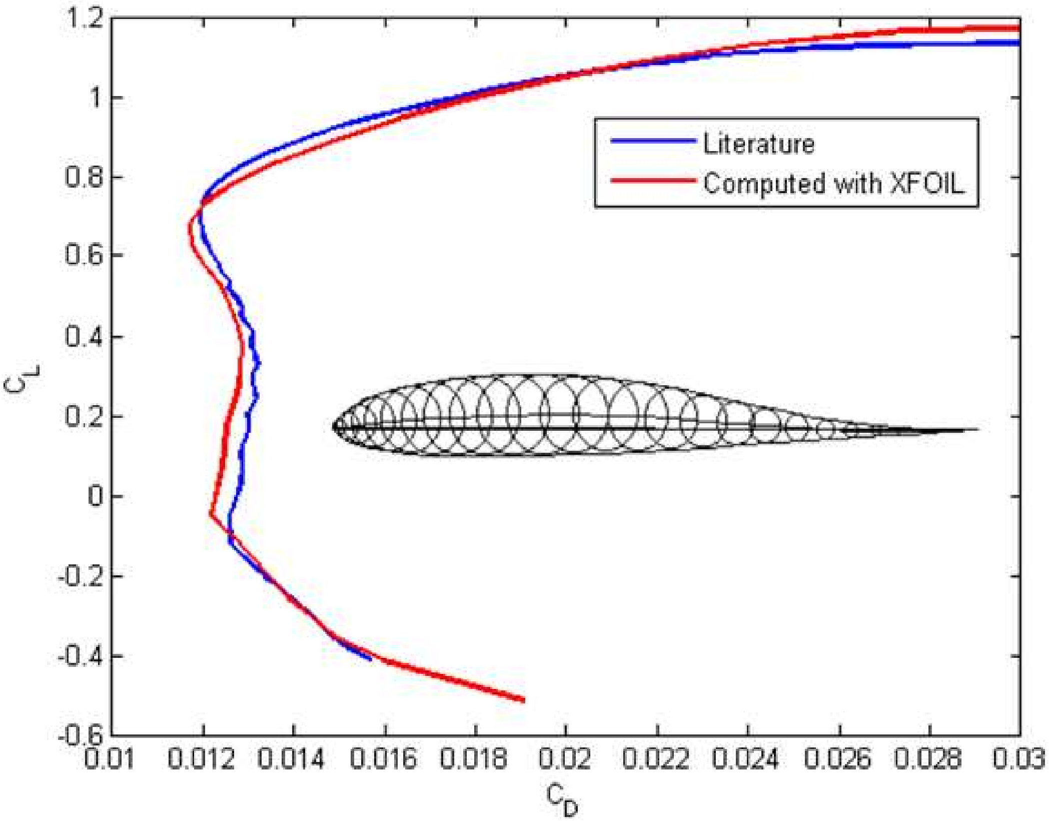


Figure 12: MH46 profile, Re=200000 and Mach = 0

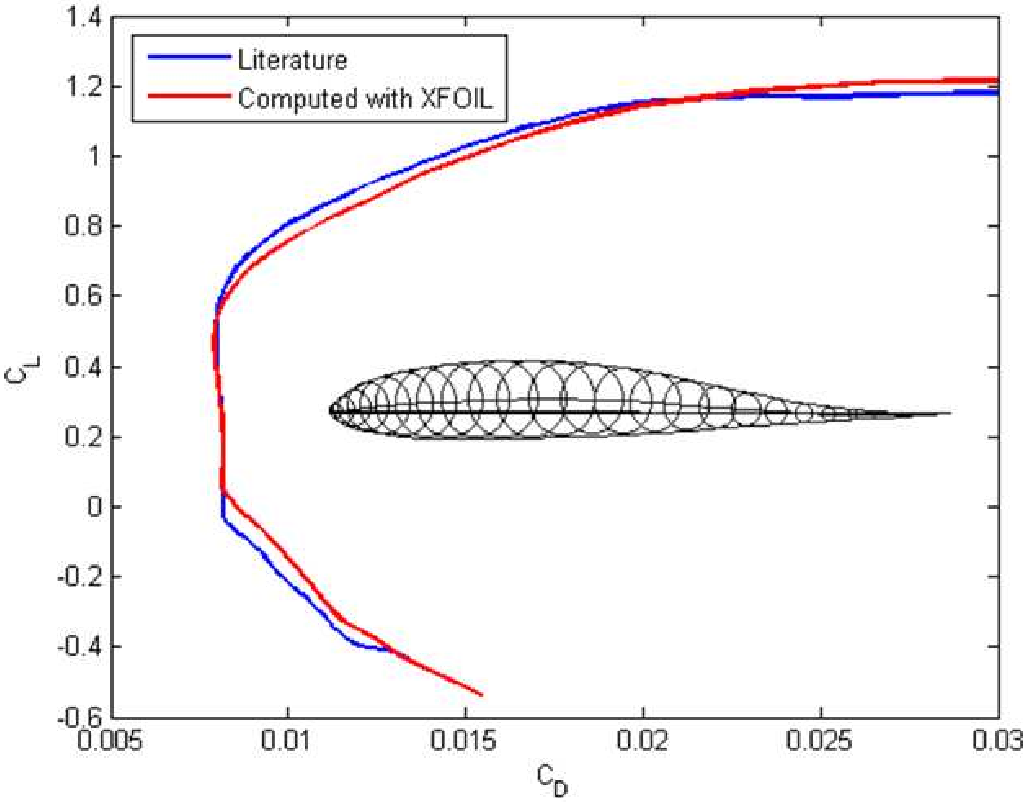


Figure 13: MH46 profile, Re=400000 and Mach = 0

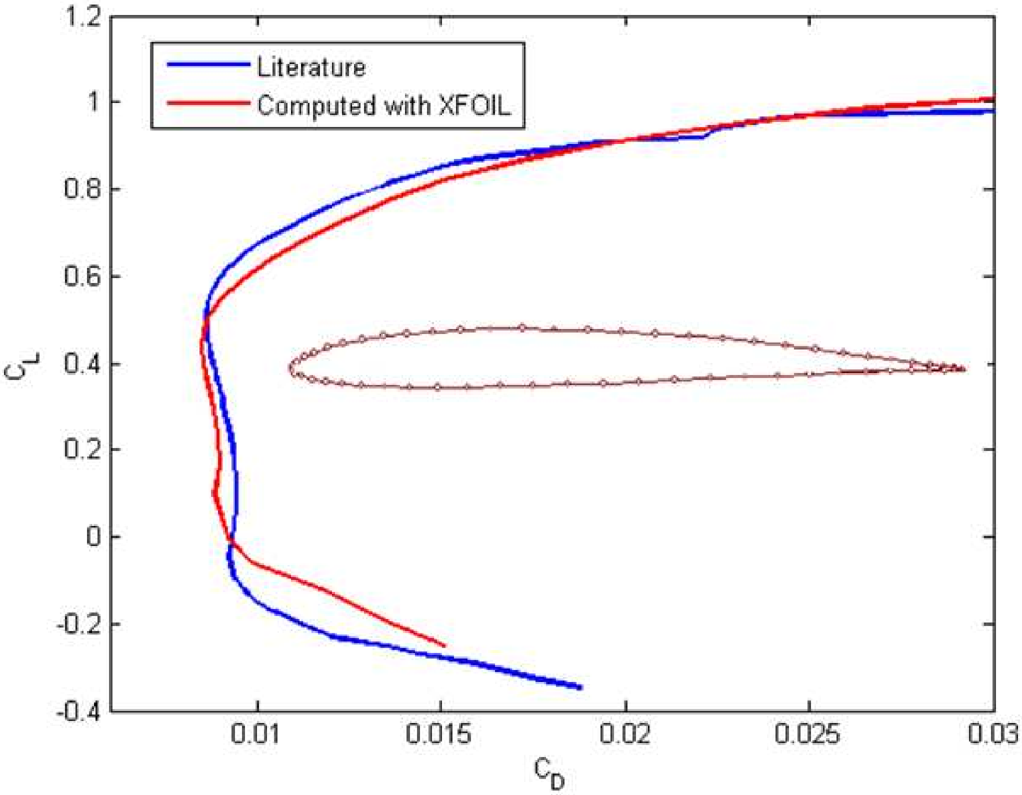


Figure 14: MH30 profile, Re=200000 and Mach = 0

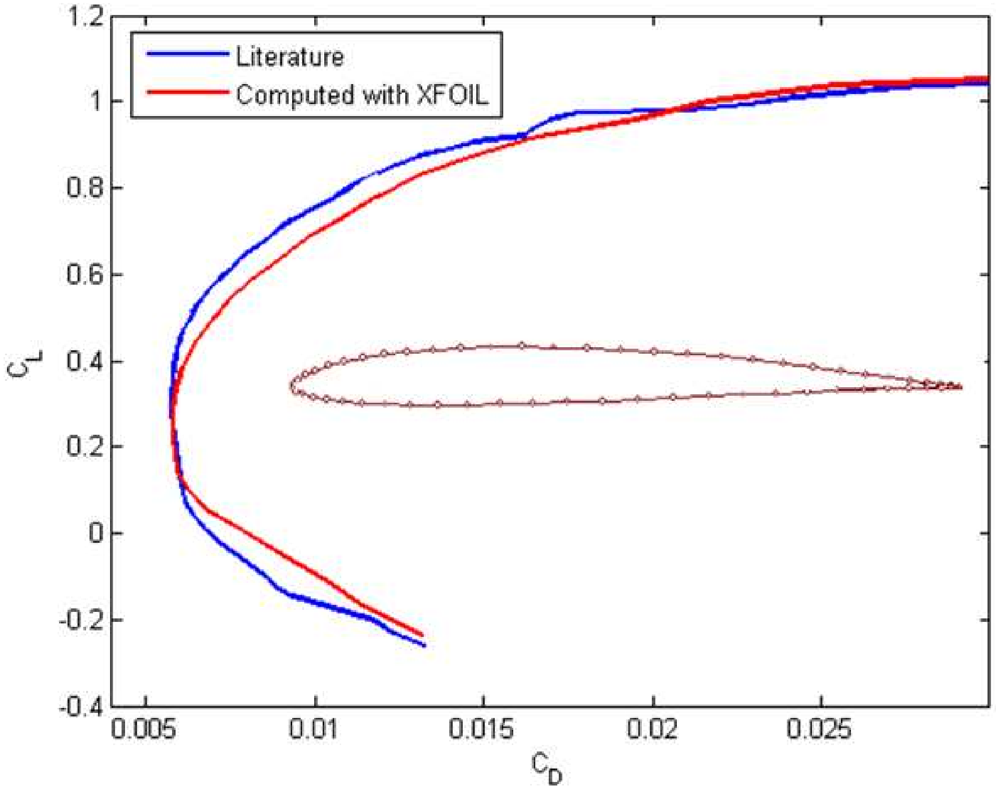


Figure 15: MH30 profile, Re=400000 and Mach = 0

The polars computed with XFOIL are very close to the literature. The small difference observed may come from the number of panels and nodes defining the airfoil geometries in XFOIL. Indeed, the number of coordinate points that defines the profile can slightly modify the polars especially if the leading edge region is not accurately defined. The polars shown in the figures above validate XFOIL results.

3.3. Interface MATLAB/XFOIL

The automatic airfoil optimisation process has to be directed from MATLAB. Hence, one of the keys of this project is to develop an interface between MATLAB and XFOIL.

As it is explained in the previous section, every XFOIL command can be automatically operated by creating an input file and loading it in XFOIL. For that purpose, specific functions are already developed in MATLAB to create different kind of files, including text, data and batch files. Thus, information can be written and save under a specific format in a specific directory. In the table below is shown the MATLAB functions used to create an input file (.dat file) and write XFOIL commands in it. In this example, the input file created orders XFOIL to compute the characteristics of a NACA2414 profile at 5 degrees angle of attack with Re=500000 and Mach number=0.2 (random values). The results are stored and saved in a data file, called *outputfile*.

Matlab script	Text file created

fid = fopen(inputfile.dat, 'wt');
fprintf(fid,'NACA2414 \n');
fprintf(fid,'OPER \n');
fprintf(fid,'VISC %12.4g \n',500000);
fprintf(fid,'M %12.4g \n', 0.2);
fprintf(fid,'PACC \n');
fprintf(fid,'outputfile.dat \n');
fprintf(fid, \n');
fprintf(fid,'A %12.4g \n', 5);
fprintf(fid,'QUIT \n');
fclose(fid);
	NACA2414
	OPER
	VISC 500000
	M 0.2
	PACC
	outputfile.dat
	
	A 5
	QUIT

The input file is then called in a batch file which is executed by the system.

Matlab script

```
%- create the run file (.bat):  -----------------------
fid = fopen(runfile, 'wt');
comand = ['xfoil < ',inputfile,' > ',outputscreenfile];
fprintf(fid,'%s\n',comand);
fclose(fid);

%- Execute the run file ------------------------------
system('cd (directory where XFOIL is installed)');
system(runfile);
```

Finally, once the results are saved in a text or data file, they are imported into MATLAB thanks to the MATALB function ***importdata***.

As it is explained in the previous section in case of no convergence of XFOIL computations, no data can be imported into MATLAB and an error message appears. Hence, a preliminary test has to check if convergence is reached. This test is based on the number of information contained in the output file.

To summarize, flow calculations can be entirely directed from MATLAB without typing manually any command in XFOIL. Every geometrical or physical parameters like the angle of attack, the Reynolds or the Mach numbers, are defined in MATLAB. Then different MATLAB functions automatically run XFOIL and import the desired results.

4. Airfoil Optimisation

It has been seen in the previous parts that any airfoil shape can be accurately reproduced using a B-spline formulation based on 14 control points. It has also been seen that the airfoil performance can be automatically determine with XFOIL thanks to an interface between MATLAB and XFOIL. Hence, these two programs are to be used now for the optimisation process.

4.1.General wind turbine airfoil requirements

Wind turbine airfoil profiles have slightly different requirements than airplane profile. Their characteristics depend on the specific rotor the airfoils are intended for (e.g. pitch regulated or stall control turbines). Nevertheless from a general point of view, some properties are desirable for most turbines and are discussed in [1] and [6].

A wind turbine blade is divided into three distinct parts: the root, the mid and the tip part. The general shape of the root part is strongly determined from the structural requirements. This section of the blade provides the entire blade its stiffness and prevents too large tip deflection. This is the reason why the root thickness is increased at the expense of aerodynamic performance (e.g. lift-drag ratio). Indeed, in this region, the structural requirements are more important than the aerodynamic performance. In the opposite, the tip part of the blade is strongly determined by the aerodynamic performance.

The local shape of the airfoil and the location of the maximum thickness are also important parameters. Indeed, this represents the geometrical compatibility of the entire blade. Having the same airfoil family along the blade enables a smooth and fairly good interpolation of the aerodynamic performance between the root, mid and tip parts. Moreover to ensure the blade stiffness, a spar is inserted inside the blade and this would not be possible without geometrical compatibility. Hence, an airfoil family must have the maximum thickness at the same location.

Concerning the aerodynamic performance, the lift-drag ratio should be high for airfoils located on the outer part of the blade for maximum power production. In case of pitch regulation and active stall regulation, the lift-drag ratio should be high at and near the operational point. For stall regulation, the lift-drag ratio should be high in the entire operational range, in other words, at angles of attack below the maximum lift coefficient. The operational point has to be close to the maximum lift to ensure the highest possible lift-drag ratio for most angles of attack until stall.

Off-design characteristics are also important because of the large variation of angle of attack during normal operation. Special study of the flow separation has to done concerning stall regulation. Indeed, to limit the risk of stall induced vibrations, the flow at maximum lift should separate from the trailing edge. To ensure well defined stall, there should be a sudden movement of the transition point from the trailing edge to the leading edge. For traditional pitch regulation, stall should be avoided in operation.

In natural conditions, bugs and dirt often soil wind turbine blades at the leading region and increase leading edge roughness [1]. This causes early transition from laminar to turbulent flow and an eventual jump in boundary layer momentum thickness. Thus, it affects a lot the airfoil performance by reducing the maximum lift, lower the lift curve slope and increasing the skin friction. It is very important that the maximum lift remains insensitive to leading edge roughness, especially for stall regulation. It is possible to simulate leading edge roughness in XFOIL by prescribing transition to 1% after the leading edge on the suction side and 10% at the pressure side [6].

Another point is important to consider in wind turbine airfoil design: the noise emission. The dominant noise from wind turbine is the broadband trailing edge noise coming from the passage of the turbulent flow over the trailing edge. This type of noise depends mainly on the free stream velocity but also on the thickness and the shape of the boundary layer at the trailing edge [6]. The latter are directly related to the airfoil shape. It is possible to design airfoils to limit noise emission at the most critical operational condition by reducing the boundary layer thickness.

From a general point of view, the desirable characteristics involve both structural and aerodynamic properties that are often in conflict with each other. For example, high lift-drag ratio is in contrast to high airfoil thickness whereas high maximum lift is in contrast to insensivity to leading edge roughness [1]. The aim of the design algorithm is thus to find an airfoil that is a compromise between these conflicting requirements.

4.2. Design Strategy

In the present work, the basic idea of the airfoil optimisation is to use the B-spline formulation of the airfoil to pull on the different control points to modify the blade profile in order to reach some objectives. Hence, the degrees of freedom of the control points are the only design variables. As far as the objectives are concerned, they can be either structural or aerodynamic. The optimisation process is iterative and each iteration loop involves several flow calculations. Thus, after every change in the geometry profile, XFOIL is called inside the optimisation algorithm to evaluate the characteristics of the newly created shape. Based on XFOIL results, the design algorithm develops an optimisation strategy to reach, if it is possible, the different objectives. The design algorithm is detailed in the next section.

The design strategy is illustrated in Figure 16.

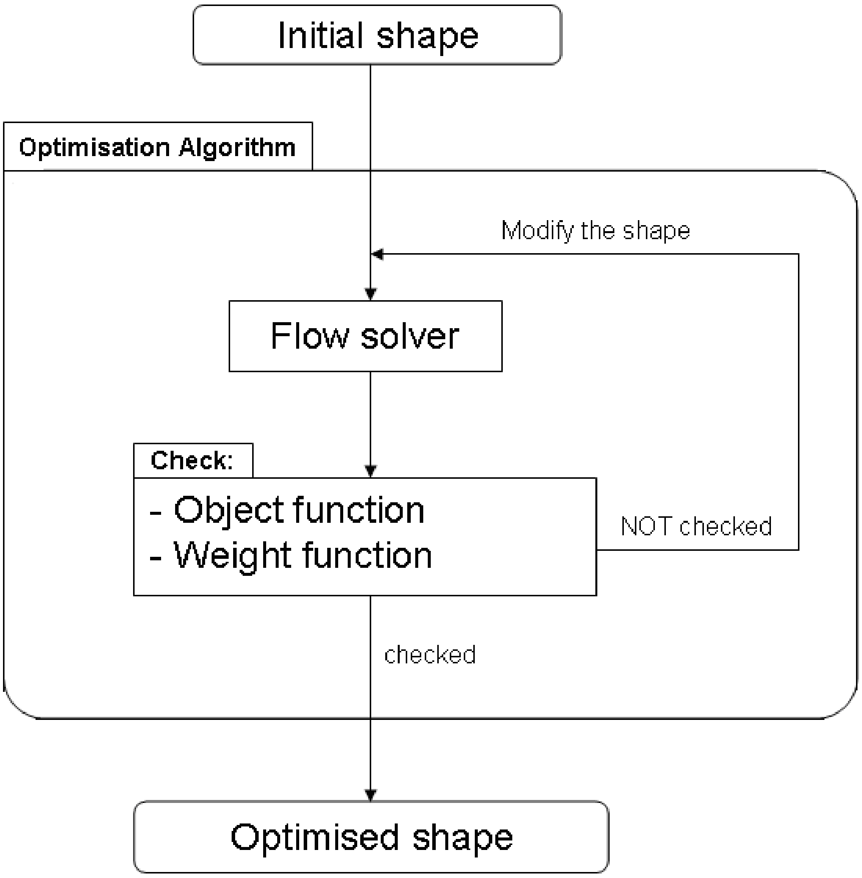


Figure 16: Flow chart of the design method

As a first step, the user has to define an object and a weight function. The object function regroups all the different objectives that the optimisation process has to satisfy, or at least approach as close as possible. If some objectives are more important than others, weight factors have to be defined in the weight function but this is the responsibility of the designer. The objectives can be both aerodynamic (e.g., lift-drag ratio for one or more angles of attack) and structural (e.g. airfoil thickness at a certain chord-wise position). It can happen that design constraints are imposed to the optimisation problem. For example a maximum/minimum lift-drag ratio or airfoil thickness is sometimes required depending on the specific airfoil is intended for. In this case design constraints are also included in the object function. Further details on the weight and object functions are given in the next sections.

The initial airfoil shape can theoretically be any shape. Nevertheless choosing an initial shape relatively close to the optimised profile decreases considerably the computational cost. For example, if the objectives tend to create a thin airfoil profile, the optimisation process would converge to a solution faster with a thin initial shape.

Once the objectives are reached or once the solution that best approaches the objectives is found, the optimisation algorithm stops and returns the final optimised profile.

The figure below is an example of the different optimisation steps: every airfoil profiles created during the optimisation process are plotted on the same figure such that the profile evolution can be clearly observed. The initial airfoil is the external thick and symmetric blue profile. In this example, the optimisation objective is to increase the lift-drag ratio of the initial airfoil. The optimisation algorithm makes the airfoil profile becomes thinner and more cambered until it reaches the given objective. It can be clearly seen from this figure that the algorithm varies the control point coordinates (circles) only along their degrees of freedom. In this example the B-spline formulation is the 4th CP distribution (see 2.2) based on 14 CPs and k=5. Hence, there are 14 design variables.

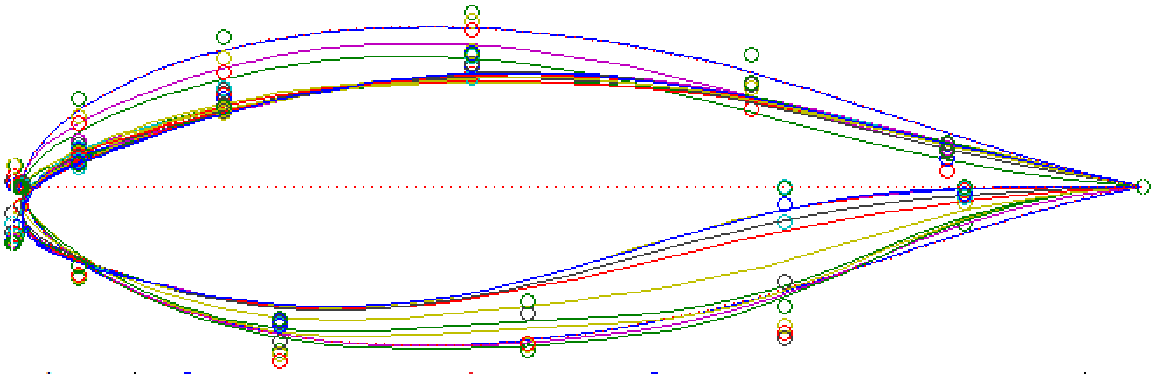


Figure 17: Example of the different optimisation steps

The initial and final profiles are both plotted in Figure 18.

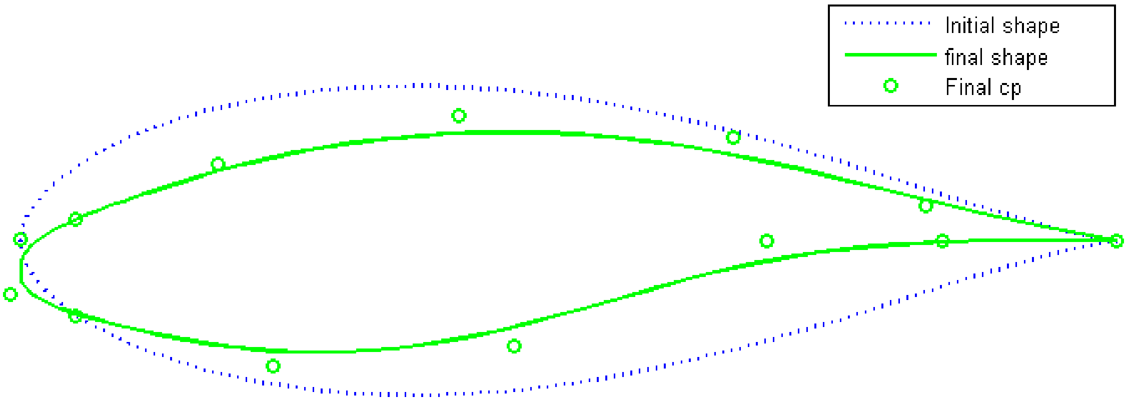


Figure 18: Initial and final profiles of the optimisation process

The overall operational conditions are defined by the Reynolds number based on the chord length c and the wind speed V:

$$\mathrm{Re} = \frac{Vc}{\nu} \quad \text{where } \nu \text{ is the kinematic viscosity of air}$$

The Reynolds number for an airfoil profile of a wind turbine blade depends on the spanwise location and on the size of the wind turbine which determines the chord and the rotational speed. The maximum Mach number is usually around 0.2, thus the flow can be considered incompressible.

4.3.Design algorithm

There are basically two possible choices for the optimisation algorithm. Either a gradient based method or global methods such as evolutionary type algorithms can be used. Evolutionary algorithms have the advantage to be less sensitive to local minima. Nevertheless they are time consuming and constraints have to be included as a penalty term on the objective function. On the other hand, gradient based methods allow multiple constraints but lack global optimality [1].

It has been chosen in [1] and [6] to use a traditional simplex optimisation algorithm based on sequential linear programming with move limits in a standard bound formulation. In the present work, it has been decided to use a pre-existing MATLAB function from the optimisation toolbox. This function, called ***fgoalattain***, is specifically developed for multi-objective goal attainment problem and involves gradient based methods. Furthermore optimisation algorithms developed and proposed by MATLAB have been chosen for their robustness and iterative efficiency [3].

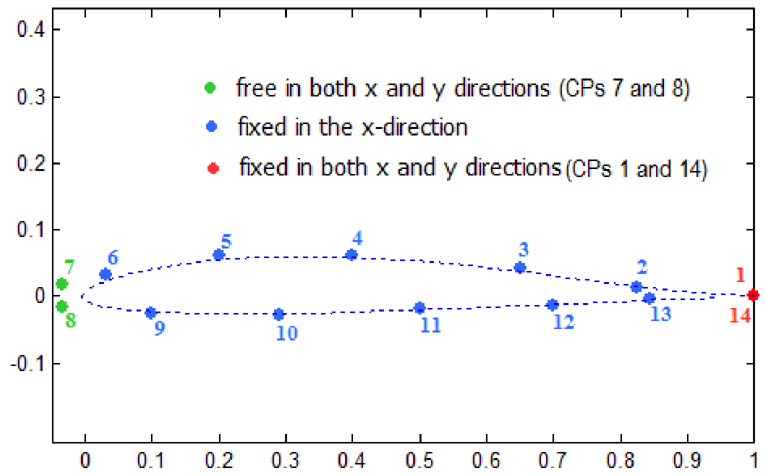
From a general point of view, multi-objective optimisation concerns the minimization of a set of objectives simultaneously. The formulation implemented in ***fgoalattain*** is the goal attainment problem of Gembicki [3]. Thus, the airfoil optimisation problem can be formulated as follows:

$$\underset{x,\gamma}{Minimise} \cdot \gamma \qquad \text{such that} \qquad \begin{array}{l} F(x) - weight \cdot \gamma \leq goal \\ lb \leq x \leq ub \end{array}$$

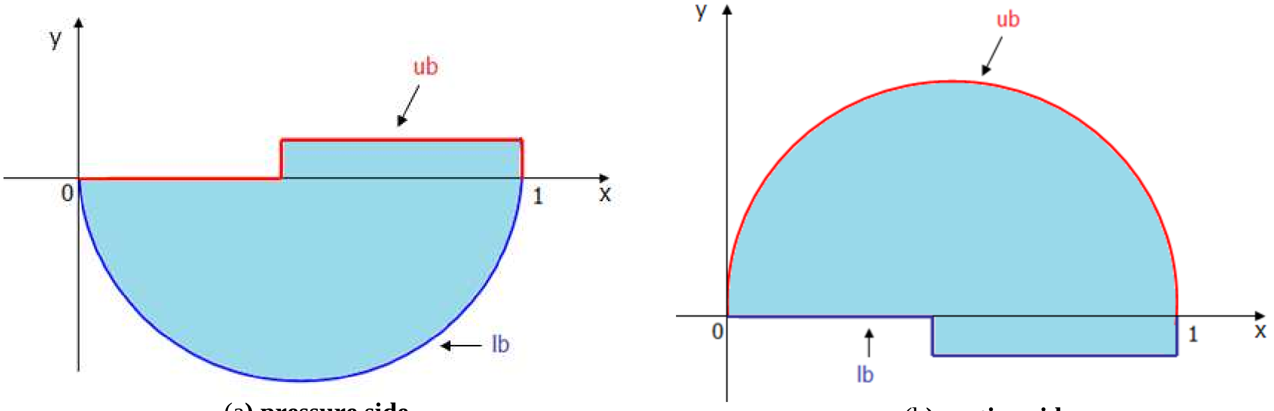
Where $F(x), x, goal, \gamma, weight$, lb and ub are vectors of the same length. x is the design vector which is bounded by a lower bound lb and an upper bound ub. In this implementation, the slack variable γ is used as a dummy argument to minimize the object function vector F(x) simultaneously. γ is an internal variable; $goal$ is a set of values that the objectives attain. Generally, prior to the optimization, it is not known whether the objectives will reach the goals (under attainment) or be minimized less than the goals (over-attainment). A weighting vector, $weight$, controls the relative under-attainment or over-attainment of the objectives.

Design variables and boundaries

The vector x contains the control point coordinates that are free along the x- or/and the y-direction. As it is explained in part 2, the CP distribution is formed by 14 points with different degrees of freedom (see the table below). Therefore, the design vector x contains 14 variables and is bounded by two vectors lb andub. Theses two vectors limit the geometrical displacements of the control points. As the thickest profile possible can not exceed a circular profile, the CPs are free to move in a circle of diameter 1. However, in order to limit computational costs, the CPs are divided into two groups: the CPs from the suction side and the CPs of the pressure side. Making this distinction, the maximum displacements of each group can be restricted to two half circles (corresponding to the blue areas in Figure 19).



$$x = \begin{pmatrix} CP7x \\ CP8x \\ CP2y \\ CP3y \\ CP4y \\ CP5y \\ CP6y \\ CP7y \\ CP8y \\ CP9y \\ CP10y \\ CP11y \\ CP12y \\ CP13y \end{pmatrix}$$



(a) **pressure side**

(b) **suction side**

Figure 19: Maximum displacement of the CPs of the pressure side (a) and the suction side (b)

Object function and goal vector

The objectives of the optimisation process have to be defined in the $goal$ vector. For example, if the objective is to reach lift-drag ratio of 100 at 5 degrees angle of attack and a ratio of 90 at 7 degrees, then the $goal$ vector is written:

$$goal = \begin{pmatrix} 100 \\ 90 \end{pmatrix} = \begin{pmatrix} C_L / C_D\big|_{\alpha=5} \\ C_L / C_D\big|_{\alpha=7} \end{pmatrix}$$

The order of the different objectives does not matter as long as it is coherent with the rest of the code.

If constraints should be applied, they have to be mentioned in the $goal$ vector as well. For example, if the airfoil profile should be normalised, then another objective can be defined as chord=1. This is formulated as follows in the $goal$ vector:

$$goal = \begin{pmatrix} 1 \\ 100 \\ 90 \end{pmatrix} = \begin{pmatrix} chord \\ C_L / C_D\big|_{\alpha=5} \\ C_L / C_D\big|_{\alpha=7} \end{pmatrix}$$

The goal vector represents the objective that the object function $F(x)$ has to reach. $F(x)$ is computed at each iteration of the optimisation process by an internal function inside ***fgoalattain***. In other words, at each change in CP coordinates, an internal function computes the characteristics of the newly created profile and regroups them in $F(x)$ (in the example mentioned above the characteristics are the chord length and the lift-drag ratio at 5 and 7 degrees).

The task of the optimisation function ***fgoalattain*** is to gradually move the design variables to minimize the difference $F(x) - goal$, and if possible, reach the objectives.

Weight function

A weighting vector $weight$ is defined to control the relative under-attainment or over-attainment of the objectives in ***fgoalattain***. When the values of $goal$ are all nonzero, to ensure the same percentage of under- or over-attainment of the objectives, the weighting function is set to $weight = abs(goal)$. In the present work, it is decided that the objectives have the same percentage.

When the weighting function $weight$ is positive, ***fgoalattain*** attempts to make the objectives less than the goal values. To make the objective functions greater than the goal values, weight is set to be negative rather than positive.

Options

Different options are available when using ***fgoalattain***. For example, it is possible to make an objective as near as possible to its corresponding goal value by using the option called *GoalsExactAchieve*. Another option turned out to be very useful: the option *DiffMinChange*. This option defines the minimum change in variables for finite-difference gradients. However the default value in MALTAL is too small (10^{-8}). Hence the change in the geometry profile is too small to affect the airfoil performance and convinces the optimiser that convergence is reach. Therefore, in order to avoid this problem, higher value of *DiffMinChange* has to be defined. In the present work, it has been set: $DiffMinChange = 10^{-2}$.

Local minima

The drawback of the algorithm implemented in ***fgoalattain*** is the sensitivity to local minima. Indeed ***fgoalattain*** might give only local solutions. This means that the optimised airfoil profile returned by the optimisation process might not be the optimum solution. Nevertheless, if the optimisation process has converged, this means that the objectives have been reached. In this case, it does not really matter if the solution is global or local.

In case of no convergence, it is possible to perform a second optimisation and define the final profile of the first optimisation as the initial profile of the second. The value of $DiffMinChange$ can be reduced to 10^{-3}. This method can sometimes avoid local minima.

XFOIL convergence

It has been explained in part 3 that XFOIL calculations might not converge. In this case, no result is returned and arbitrary values have to be assigned to the different terms of object function. It has been decided in the present work to set the lift coefficient and the lift-drag ratio to zero when XFOIL does not converge.

5. Program validation

Examples of airfoil profile optimisation are detailed in this section. Theses examples concern only the improvement of the MH43 profile at different angles of attack for given Reynolds and Mach numbers. Furthermore, every design objectives described in 4.1 are not all included in the following optimisations, mainly due to a lack of time and computing power. Thus, in the following examples, it has been decided to improve only the lift-drag ratio and the lift coefficient in order to validate the optimisation code.

The lift-drag ratio is to be improved at different angles of attack and two geometrical constraints are to be imposed to the final design. First, the chord length has to remain equal to chord=1 in order to normalise the final profile. Then, as a spar is supposed to be inserted in the final profile, a minimum thickness is imposed to the profile. Neither the shape nor the dimensions of the spar are realistic because the optimisation presented here is just an example of the automatic 2D airfoil optimisation. The spar is assumed to be rectangular and its dimensions are set randomly.

First, the optimisation is performed at 2 different angles of attack separately (2 and 5 degrees). It consists in improving the lift-drag ratio at these specific angles of attack independently. Finally, the optimisation is performed at 2 different angles of attack at the same time (2 and 5 degrees). The effect of the geometrical constraints due to the presence of the spar on the optimised airfoil profile is shown in each case.

The optimisation process is performed in each case with:

Mach number = 0.2
Re = 3 000 000

Theses values correspond to the typical operational conditions of a 30m diameter wind turbine. At these values, the characteristics of the MH43 profile are given in the figure below.

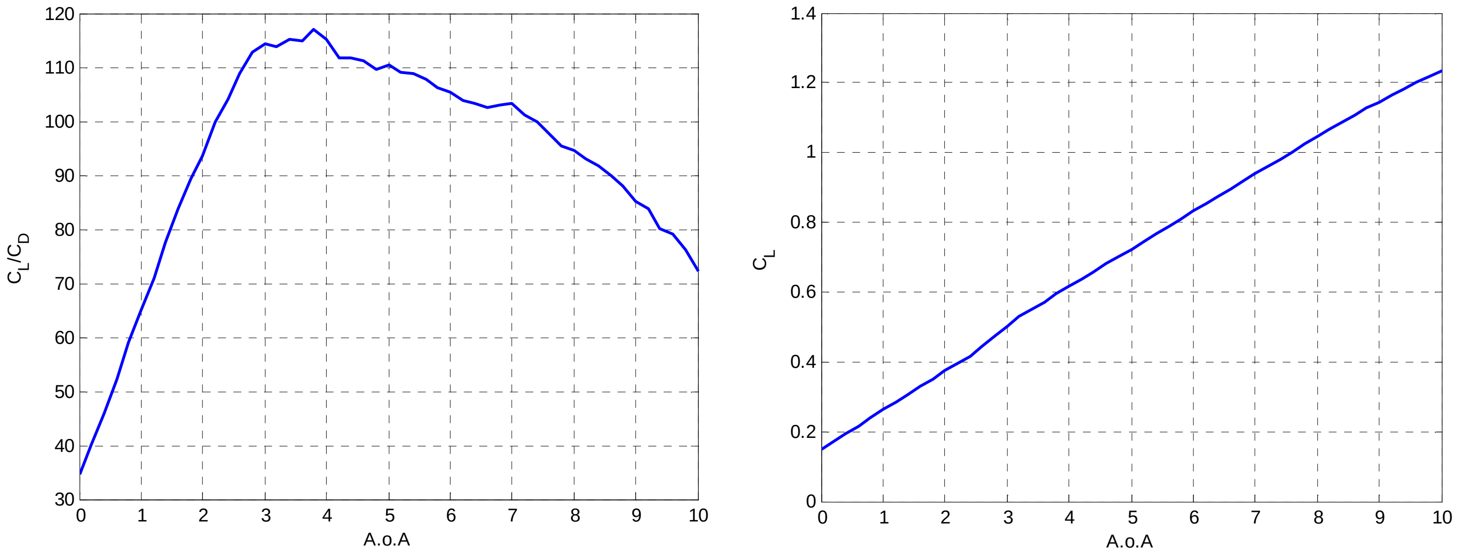


Figure 20: MH43 profile characteristics at Re=3e6 and Mach = 0.2

5.1.Improvement of the lift-drag ratio at 2 degrees angle of attack

At α = 2 degrees, the initial MH43 profile has a lift-drag ratio of 93 (see Figure 20). The aim of the optimisation is to increase this ratio to 120 and keep the same chord length. Thus, the ***Goal*** vector given to the optimisation function ***fgoalattain*** is written as follow:

$$Goal = \begin{pmatrix} chord \cdot length = 1 \\ \left.\dfrac{C_L}{C_D}\right|_{\alpha=2} = 120 \end{pmatrix}$$

The spar is supposed to be rectangular. Its position and its dimensions are taken arbitrarily to illustrate the capacity of the design algorithm to include geometrical constraints in the optimisation process. The spar is defined its four corners (see the figure below). Given XsparLE and XsparTE (x coordinates of the corners), the four y coordinates of the corner are defined as objectives to be reached in the ***Goal*** vector.

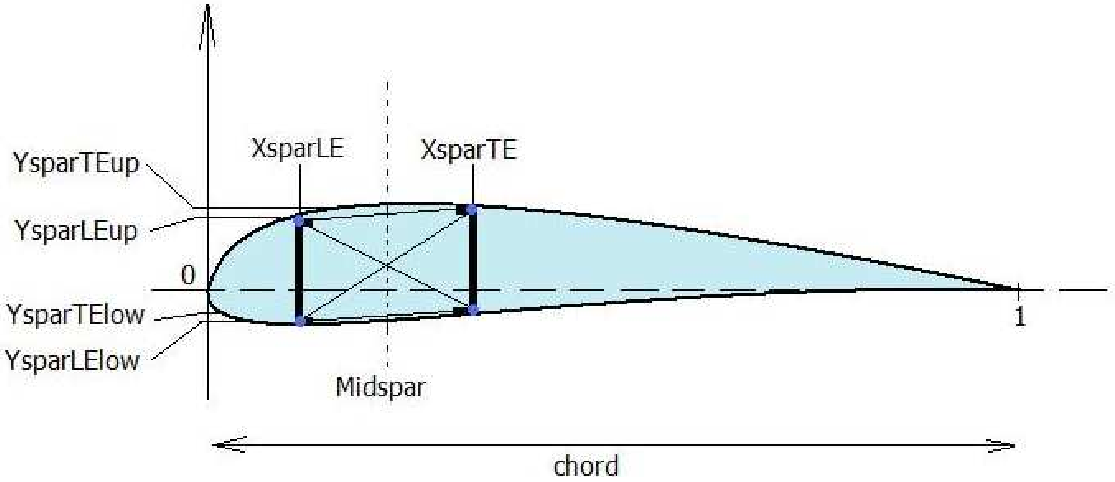


Figure 21: Illustration of the rectangular spar

The results of the optimisation are given in the table below.

Results:	
Considering the spar constraints:	Without considering the spar constraints:
$\begin{pmatrix} chord \cdot length = 1 \\ \left.\dfrac{C_L}{C_D}\right\|_{\alpha=2} = 122{,}0388 \end{pmatrix}$	$\begin{pmatrix} chord \cdot length = 1 \\ \left.\dfrac{C_L}{C_D}\right\|_{\alpha=2} = 120{,}2579 \end{pmatrix}$

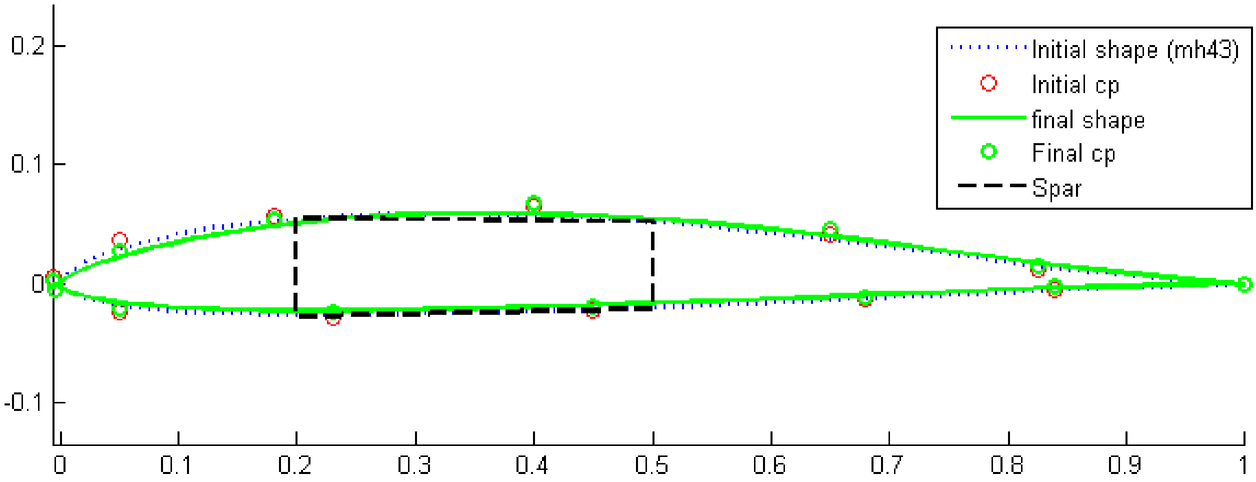


Figure 22: Optimisation at alpha=2 degrees WITHOUT considering the spar constraints

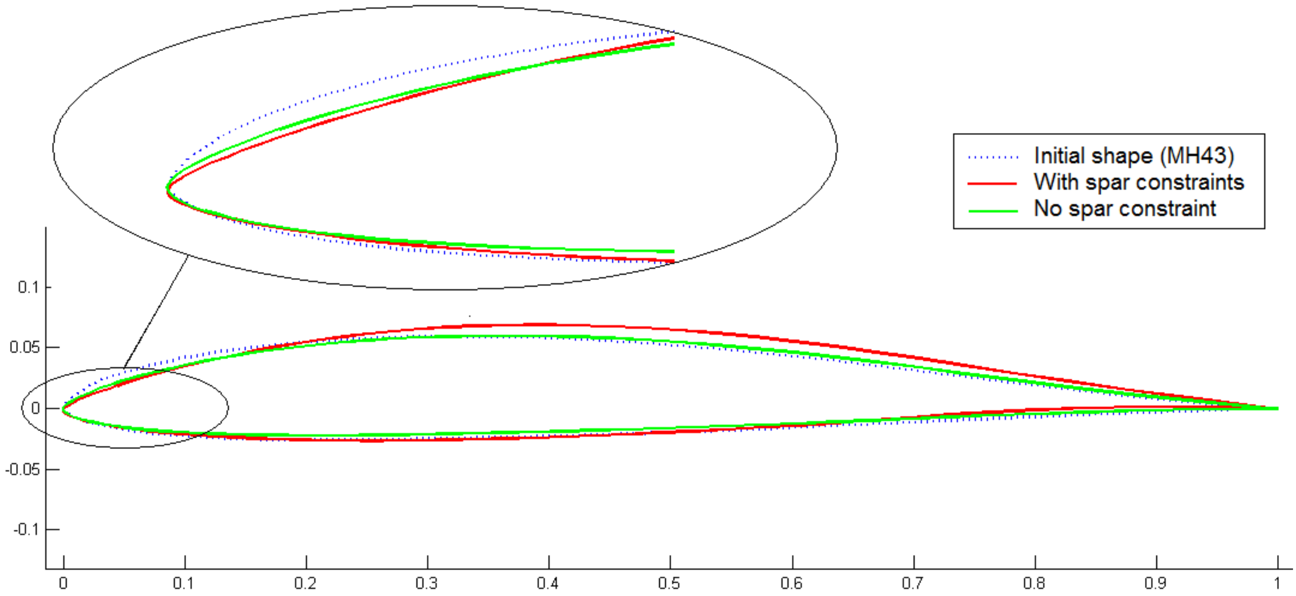


Figure 23: Optimisation at alpha=2 degrees, effect of the spar constraints on the final profile

The aerodynamic characteristics of the optimised airfoil (considering the spar constraints) are compared to the initial MH43 characteristics and shown in the figures below. It can be seen that the MH43 performance have been enhanced at a 2 degrees angle of attack but drop suddenly soon after. Indeed the airfoil profile is designed for a single and small angle of attack. The sharp nose is efficient for very small angles but leads to early separation at the leading edge when incidence increases (see Figure 26).

From a general point of view, the new designed airfoil has better characteristics for angles of attack below the design angle but looses its performance at larger incidences.

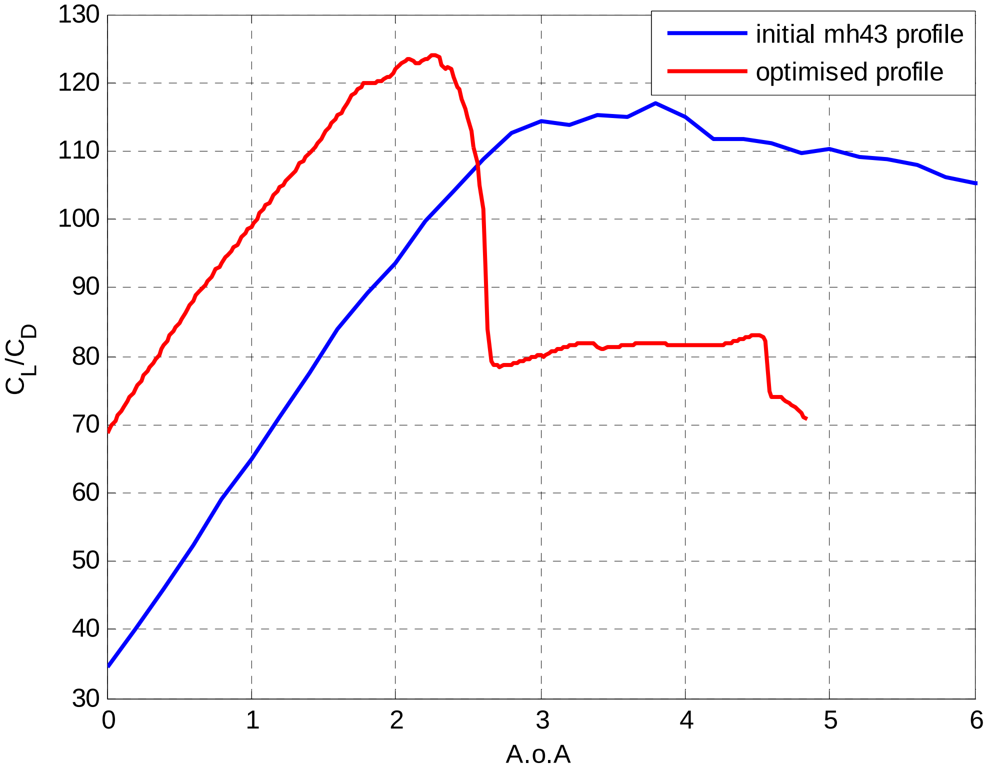


Figure 24: Lift/Drag ratio versus the angle of attack, Re=3000000 and Mach=0.2

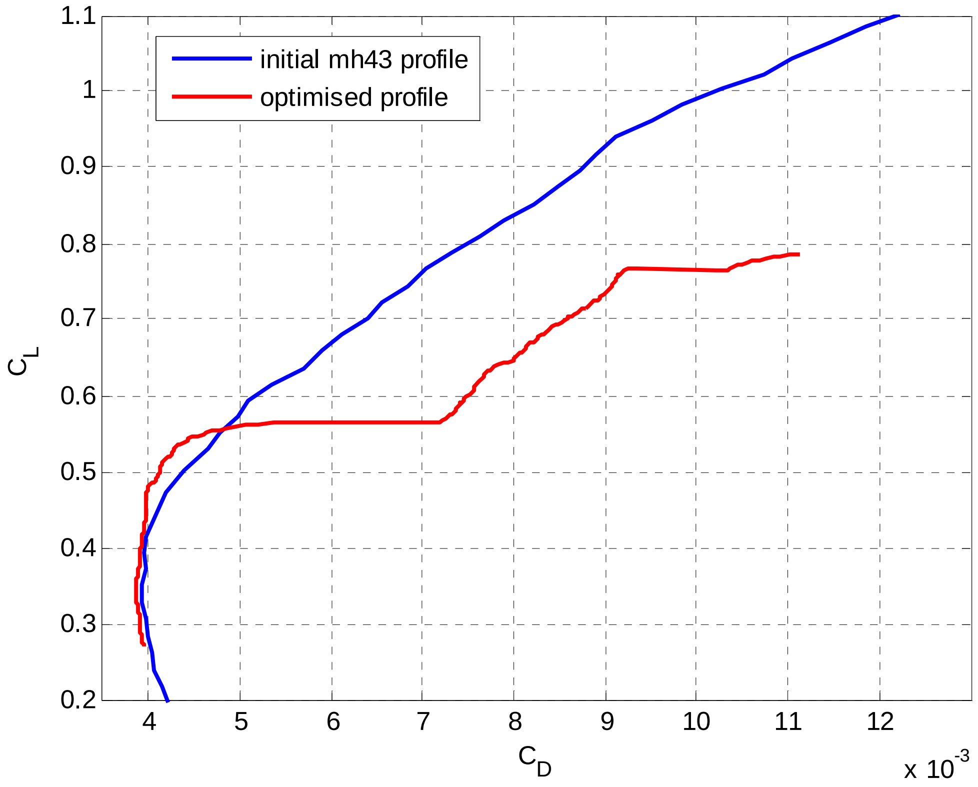


Figure 25: Lift Coefficient versus Drag Coefficient, Re=3000000 and Mach=0.2

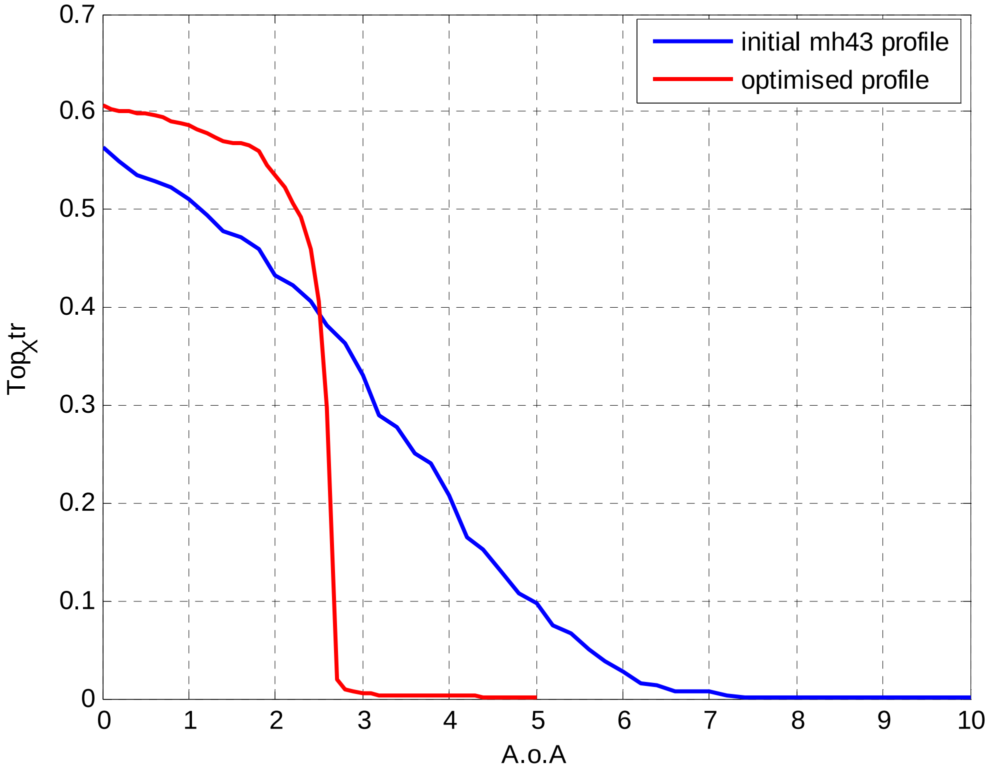


Figure 26: Location of the transition point on the suction side versus the angle of attack

5.2.Improvement of the lift-drag ratio at 5 degrees angle of attack

At α = 5 degrees, the initial MH43 profile has a lift-drag ratio of 110 (see Figure 20). In the same way as in the previous section, the aim is to increase this ratio to 130 and keep the same chord length. Thus, the ***Goal*** vector given to the optimisation function is written as follow:

$$Goal = \begin{pmatrix} chord \cdot length = 1 \\ \left.\frac{C_L}{C_D}\right|_{\alpha=5} = 130 \end{pmatrix}$$

The results of the optimisation are given in the table below.

Results:	
Considering the spar constraints:	Without considering the spar constraints:
$Goal = \begin{pmatrix} chord \cdot length = 1 \\ \left.\frac{C_L}{C_D}\right\|_{\alpha=5} = 130{,}0388 \end{pmatrix}$	$Goal = \begin{pmatrix} chord \cdot length = 1 \\ \left.\frac{C_L}{C_D}\right\|_{\alpha=5} = 131{,}0906 \end{pmatrix}$

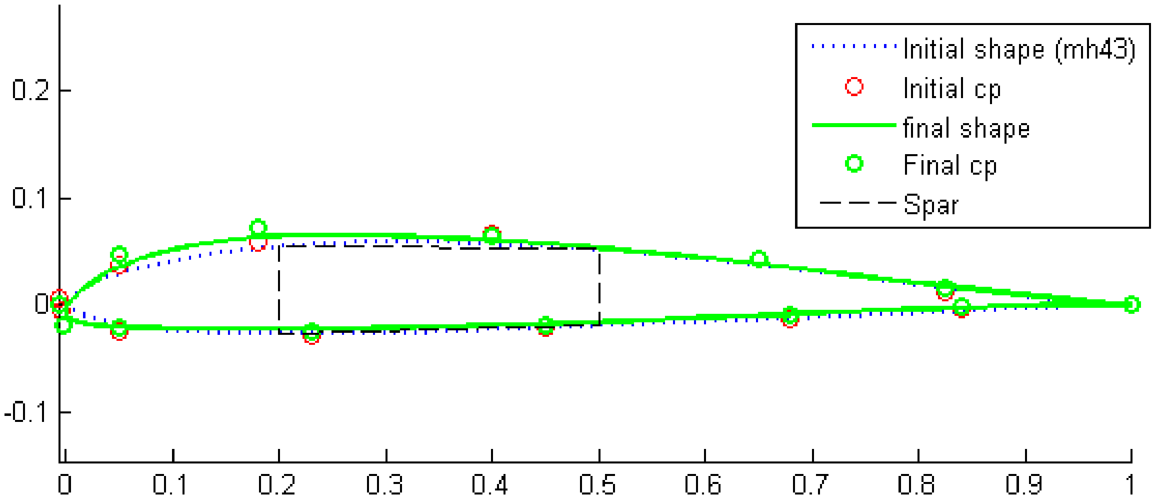


Figure 27: Optimisation at alpha=5 degrees WITHOUT considering the spar constraints

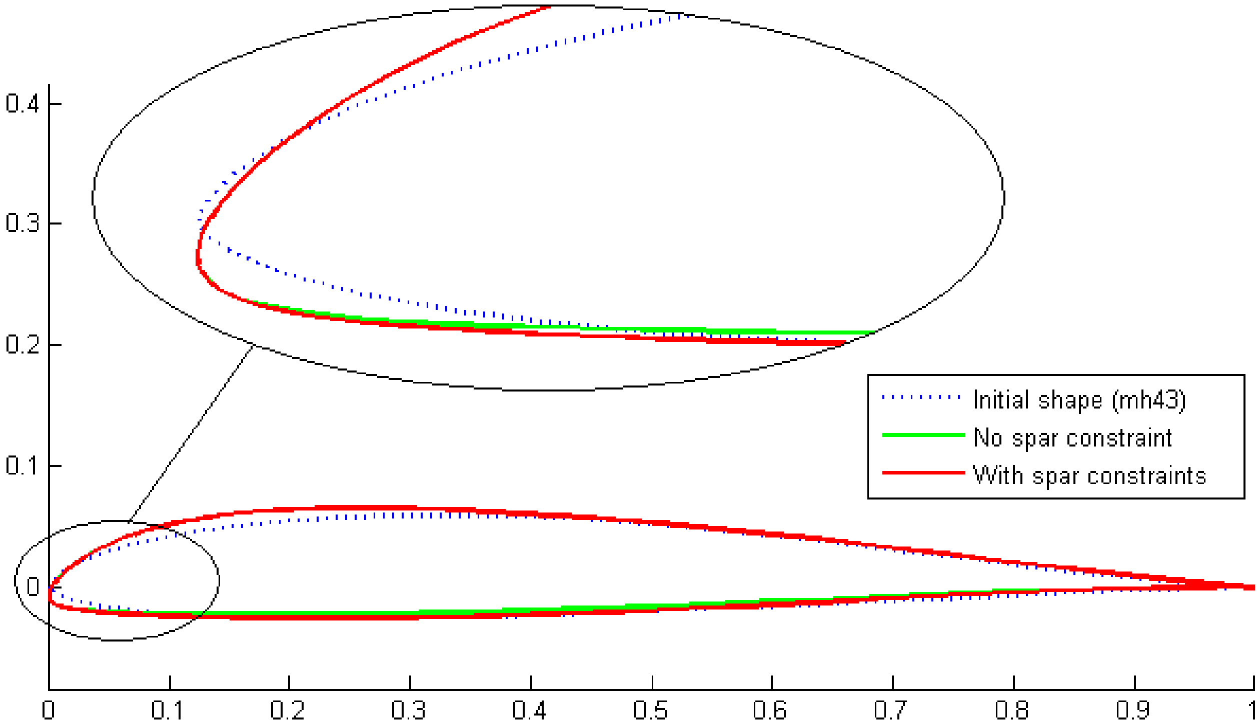


Figure 28: Optimisation at alpha=2 degrees, effect of the spar constraints on the final profile

The geometrical difference between the final profiles when the spar dimensions are taken into account or not is not obvious. The leading edge and the suction are very similar. Only the maximum thickness varies between the two optimised profiles. Nevertheless the geometrical difference with the initial MH43 profile appears clearly on the figure above. Indeed leading edges of the optimised profiles are much more cambered and thicker. This leads to better performance at large angles of attack. However, due the thicker leading edge, the drag coefficient is higher than the initial MH43 profile for small incidences (see Figure 30). This is why the optimised profiles have lower lift-drag ratio below 4 degrees angle of attack.

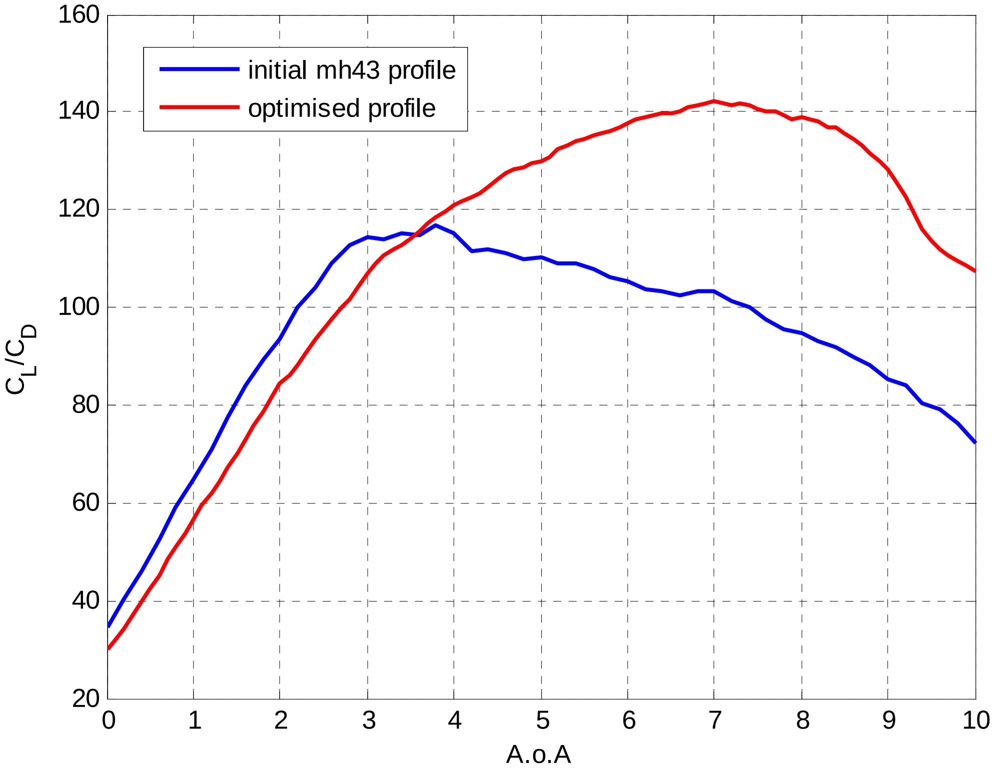


Figure 29: Lift/Drag ratio versus the angle of attack, Re=3000000 and Mach=0.2

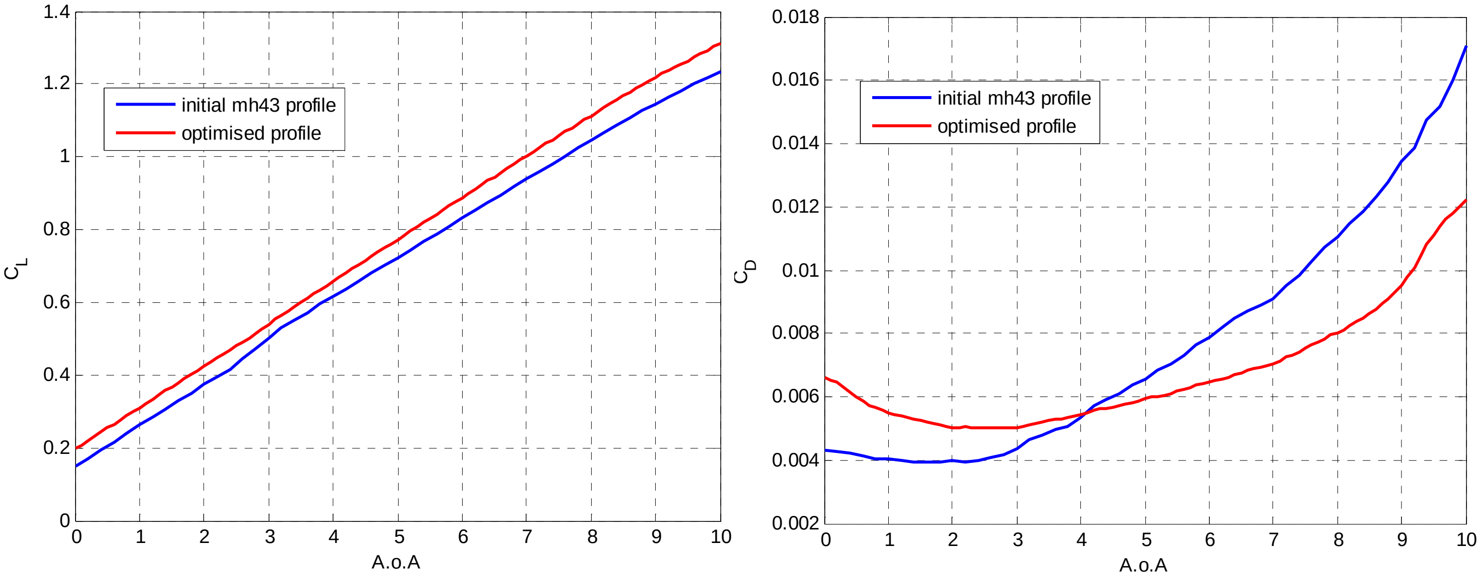


Figure 30: Lift and Drag coefficients versus the angle of attack

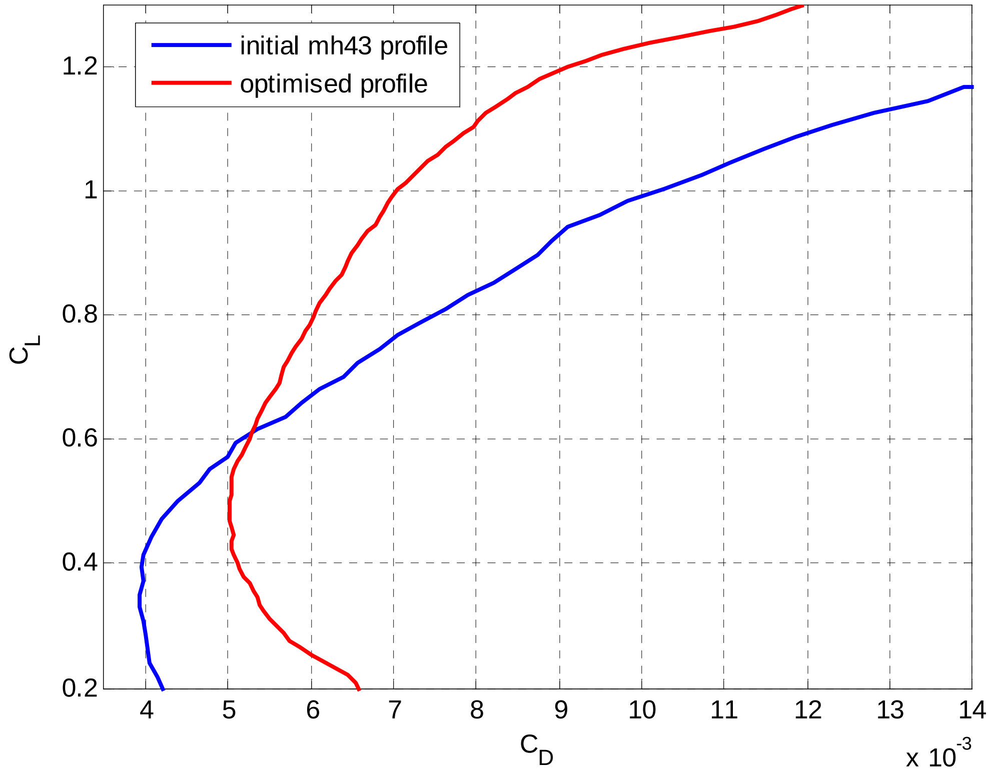


Figure 31: Lift coefficient versus Drag coefficient

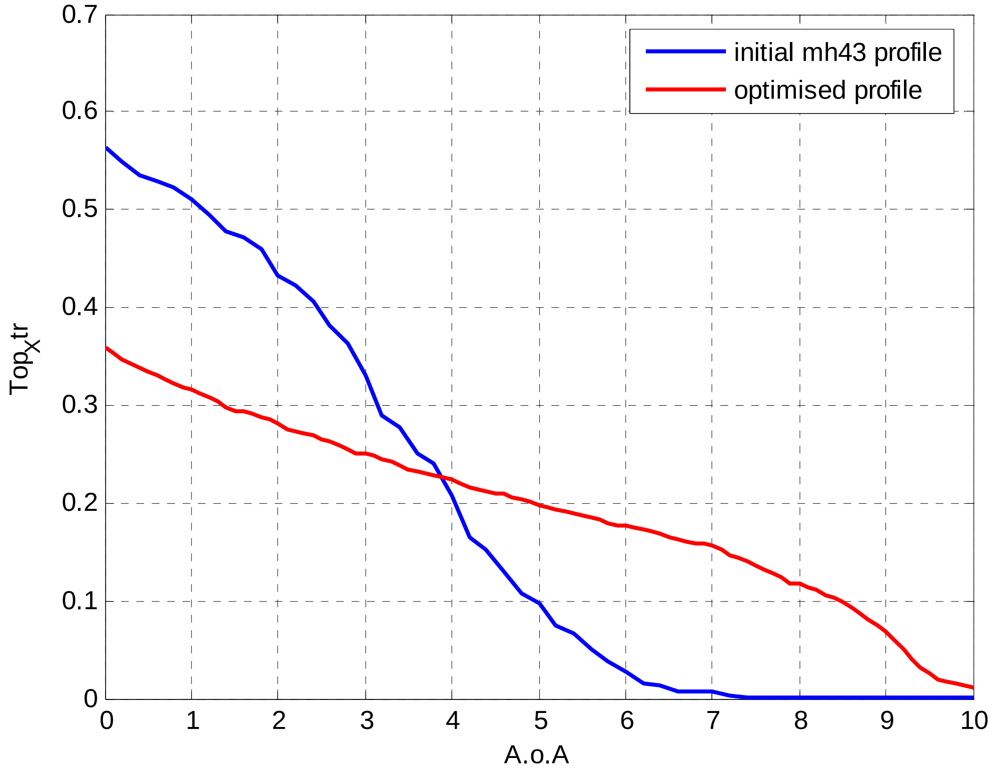


Figure 32: Location of the transition point on the suction side versus the angle of attack

5.3.Improvement of the lift-drag ratio at 2 AND 5 degrees angle of attack

In this section, the optimisation is performed at two angles of attack at the same time. In other words, the optimisation function is trying to simultaneously increase the lift-drag ratios at these two angles. According to Figure 20, the maximum lift-drag ratio of the initial MH43 profile is reached at α = 3.5 degrees. Improving the characteristics around this angle would tend to improve the profile performance over a large range of angles of attack. Thus the ***Goal*** vector given to the optimisation function is written as follow:

$$Goal = \begin{pmatrix} chord \cdot length = 1 \\ \left.\frac{C_L}{C_D}\right|_{\alpha=2} = 110 \\ \left.\frac{C_L}{C_D}\right|_{\alpha=5} = 120 \end{pmatrix}$$

Results:	
Considering the spar constraints:	Without considering the spar constraints:
$Goal = \begin{pmatrix} chord \cdot length = 1 \\ \left.\frac{C_L}{C_D}\right\|_{\alpha=2} = 110{,}0001 \\ \left.\frac{C_L}{C_D}\right\|_{\alpha=5} = 1203{,}6635 \end{pmatrix}$	$Goal = \begin{pmatrix} chord \cdot length = 1 \\ \left.\frac{C_L}{C_D}\right\|_{\alpha=2} = 110{,}0011 \\ \left.\frac{C_L}{C_D}\right\|_{\alpha=5} = 120{,}5164 \end{pmatrix}$

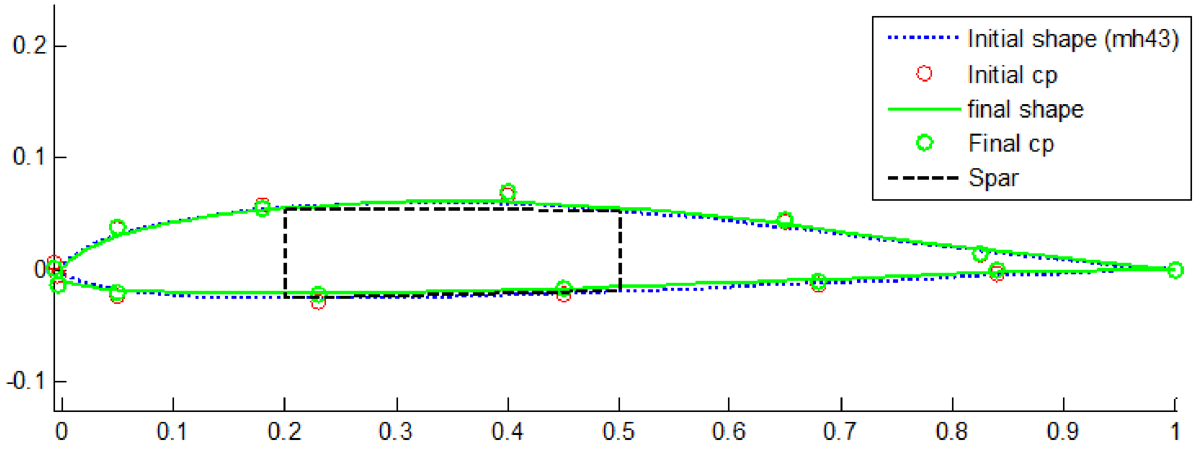


Figure 33: Optimisation at alpha=2 AND 5 degrees WITHOUT considering the spar constraints

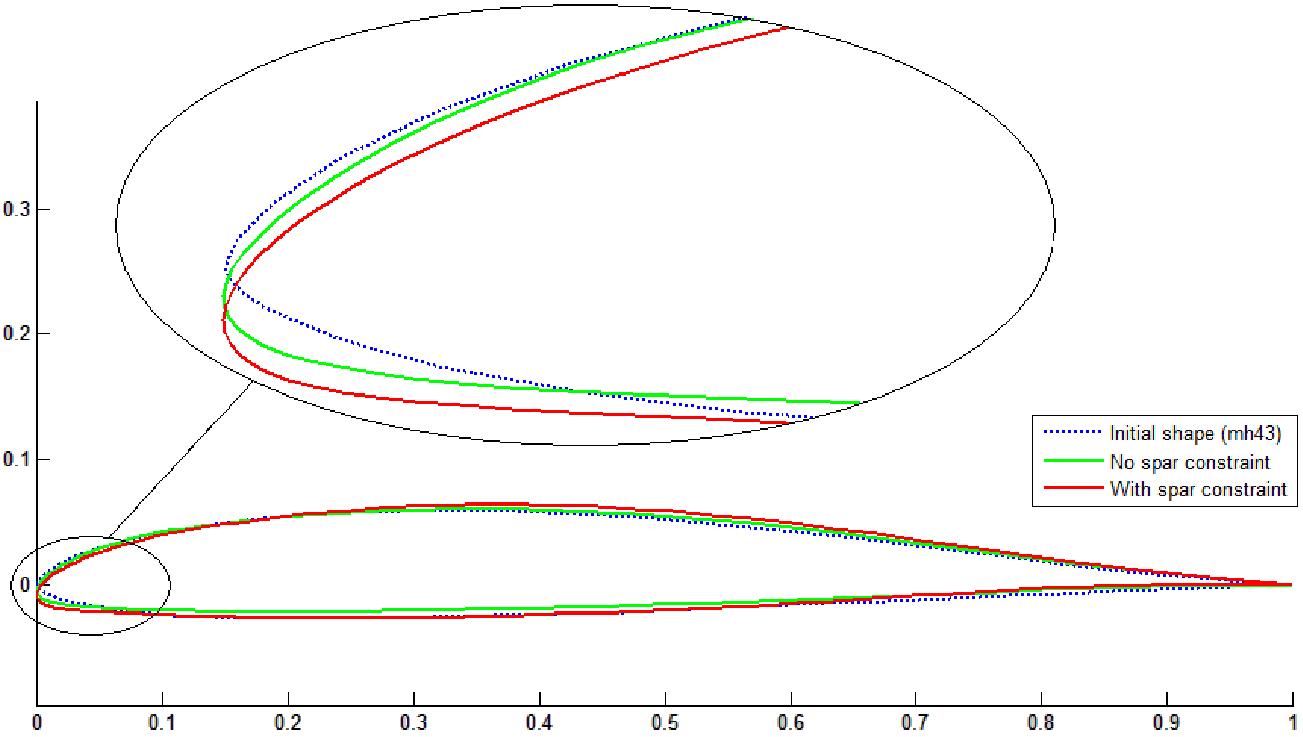


Figure 34: Optimisation at alpha=2 AND 5 degrees, effect of the spar constraints on the final profile

The aerodynamic characteristics of the optimised airfoil (considering the spar constraints) are compared to the initial MH43 characteristics and shown in the figure below. It can be seen that the MH43 performances have been enhanced for all angles of attack.

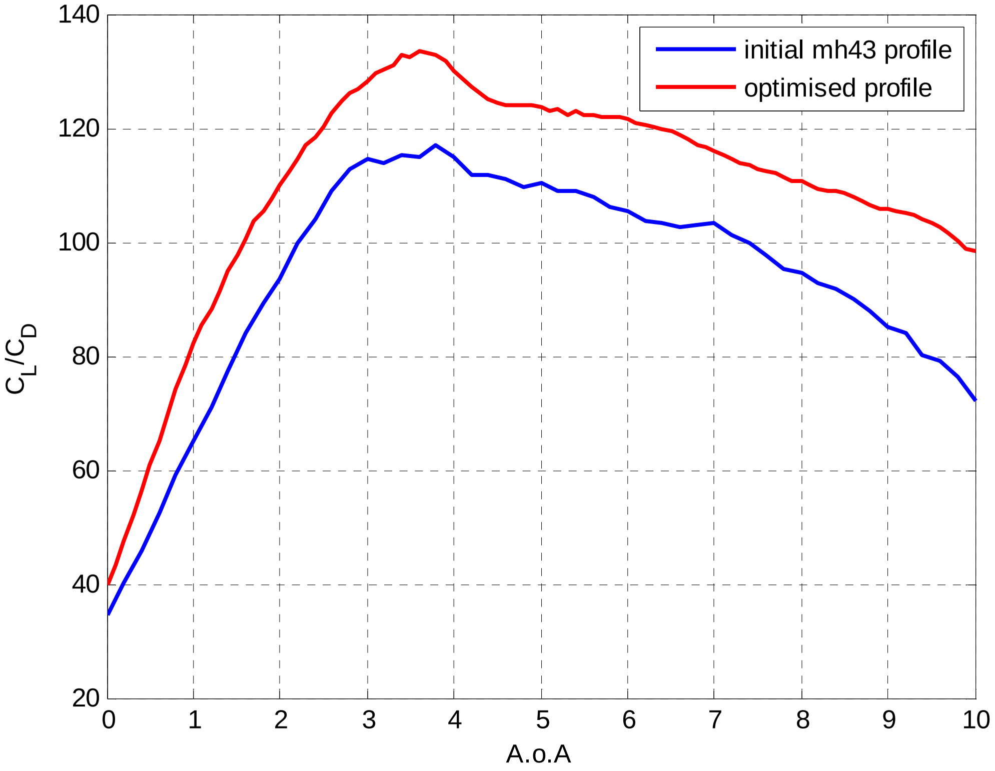


Figure 35: Lift/Drag ratio versus the angle of attack, Re=3000000 and Mach=0.2

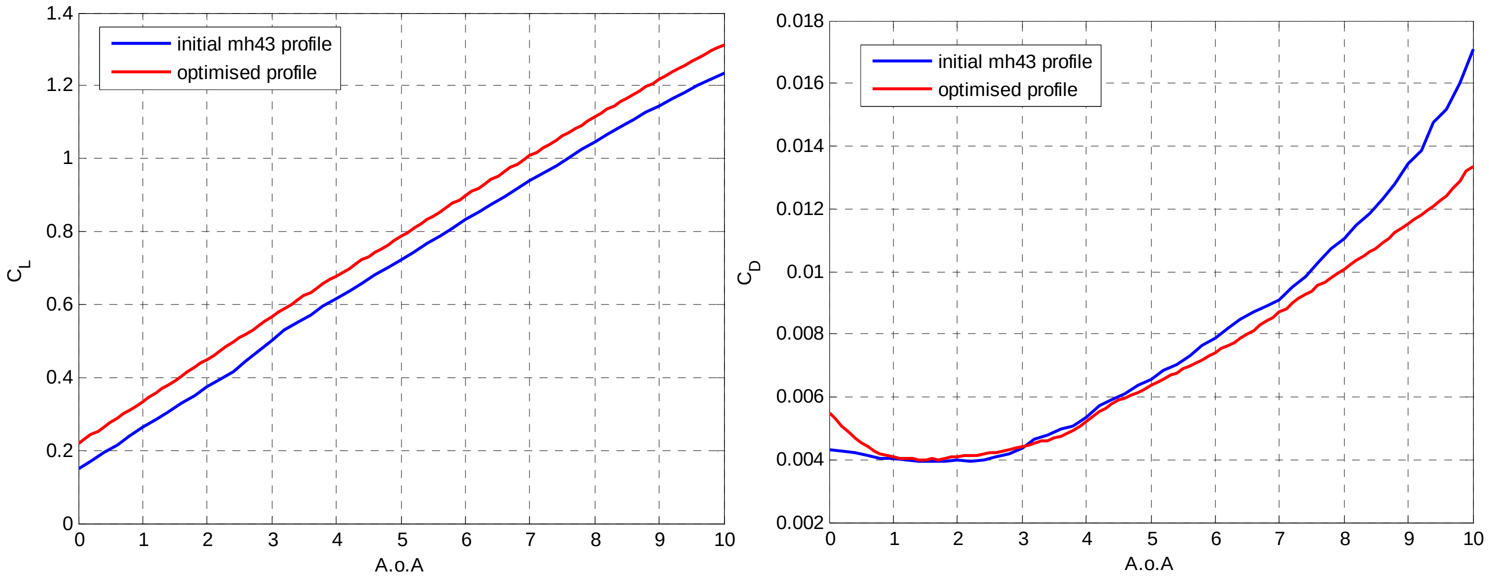


Figure 36: Lift and Drag coefficients versus the angle of attack

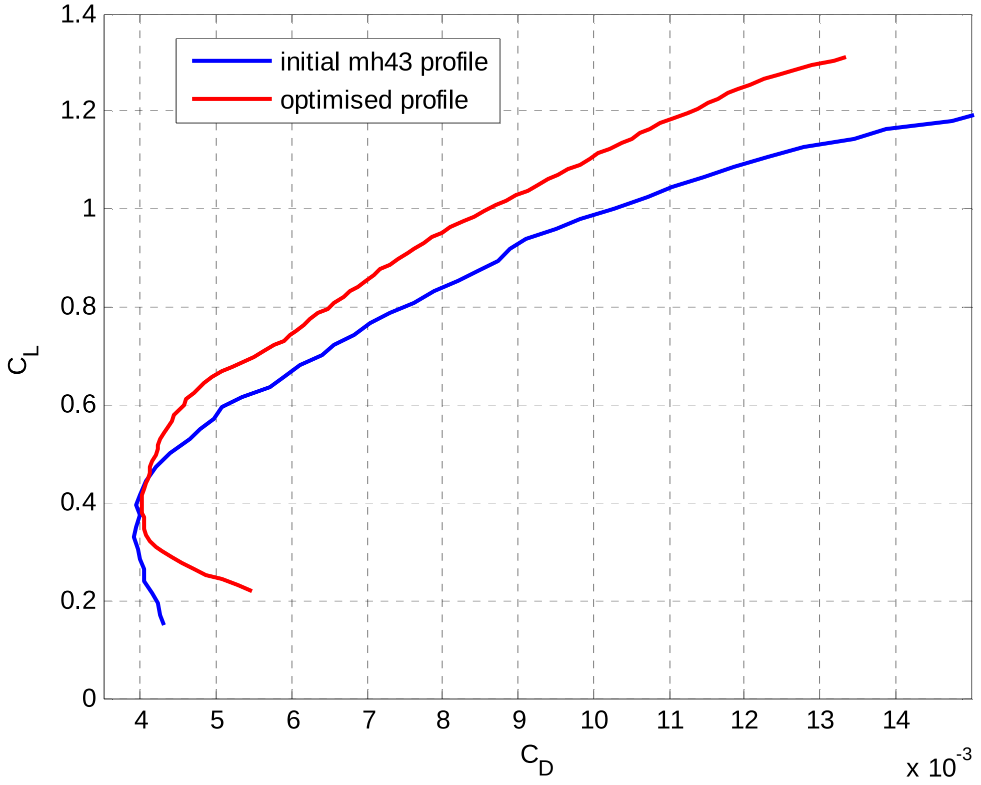


Figure 37: Lift coefficient versus Drag coefficient

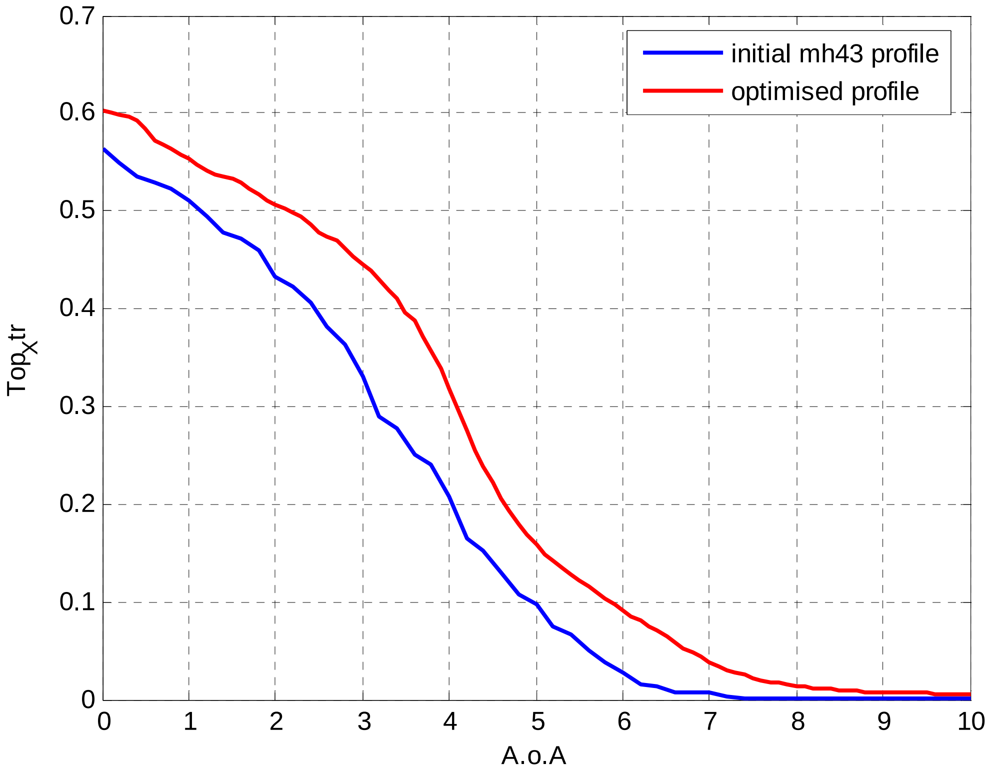


Figure 38: Location of the transition point on the suction side versus the angle of attack

In this particular case, optimising the MH43 profile at 2 and 5 degrees is sufficient to enhance the profile performance on the entire range of angles of attack. Nevertheless, the optimisation is made at Re=3000000 and Mach=0.2 and it is possible that the performance would not have been as much enhanced at different values of Re and Mach. Moreover though the optimised profile has better lift-drag ratio, it may have lower roughness insensitiveness. This is why a roughness analysis is performed and shown in Figure 39.

It is explained in part 4.1, that in natural conditions bugs and dirt often soil wind turbine blades and affect significantly the blade performance. It is possible to simulate this phenomenon in XFOIL by forcing transition laminar-turbulent to 1% of chord after the leading edge on the suction side and 10% at the pressure side. The results are shown in Figure 39. Clean and dirty (dashed line) performance of both MH43 and optimised profile are plotted in the figure. It can be seen that both profile are very sensitive to leading edge roughness but it turns out that the optimised profile keeps better performance than the MH43 profile with a rough leading edge.

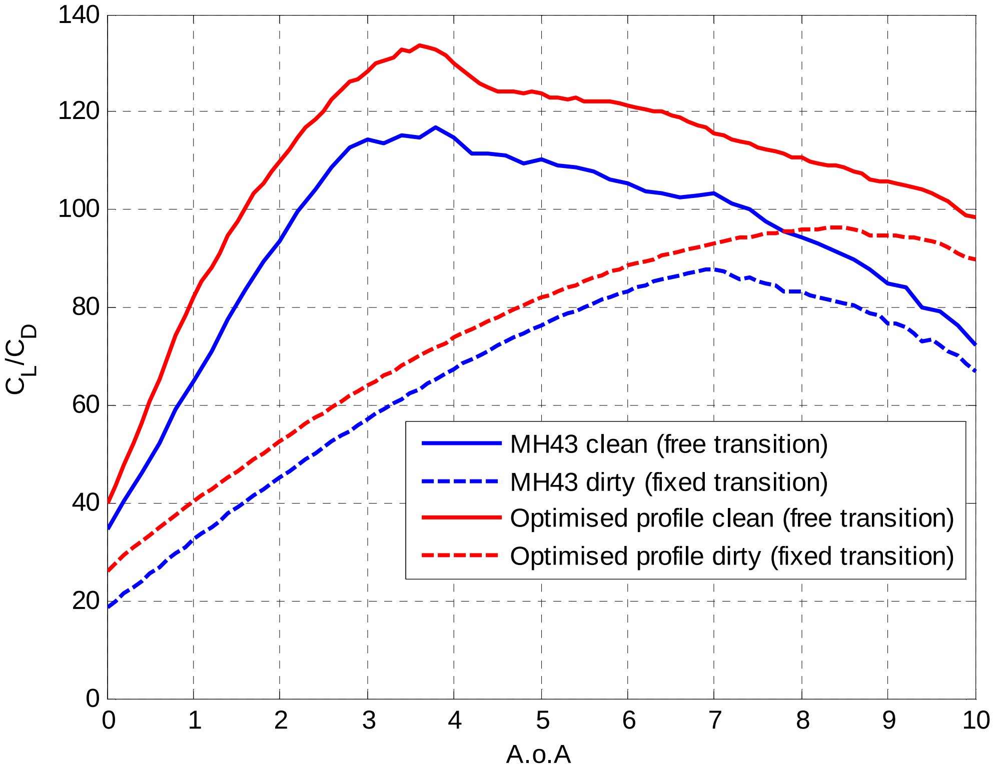
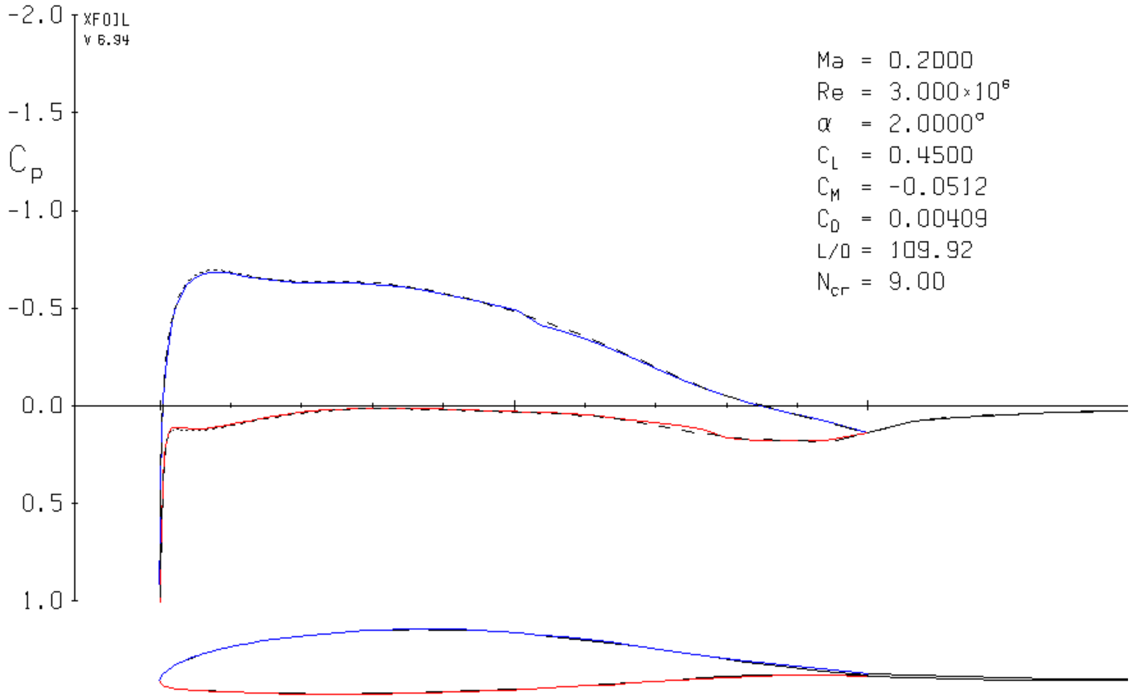


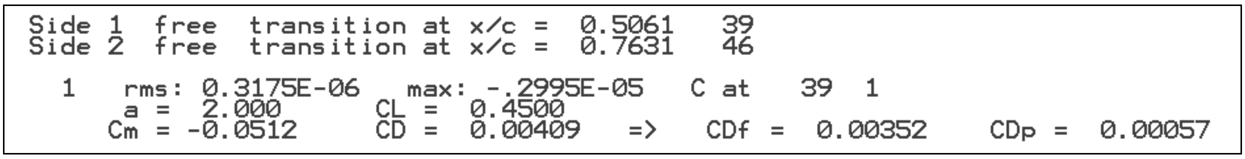
Figure 39: Roughness sensitiveness analysis

Theses examples of local optimisation of the MH43 profile show some very interesting results that are validated by XFOIL. To summarize this last section, the optimisation program has optimised the MH43 profile at the specific operational conditions Re=3e6 and Mach=0.2. The MH43 performances have been enhanced on a large incidence interval (from 0 to 10 degrees angle of attack). The optimised profile turns out to be roughness sensitive but not worse than the initial MH43 profile. The entire optimisation has been automatically directed from MATLAB.

In the following figures are shown the pressure distributions on the optimised airfoil at the design angles of attack (2 and 5). The dashed lines represent the inviscid flow distribution. This provides an easy way to compare viscous and inviscid flow. Furthermore the boundary layer is outlined around the airfoil. The XFOIL command screens are given in the tables below their corresponding figures. They show the last iteration which provides the point of transition to turbulence (x/c) on both extrados (side1) and intrados (side2). It also provides CDf and CDp, the friction drag and pressure drag respectively.

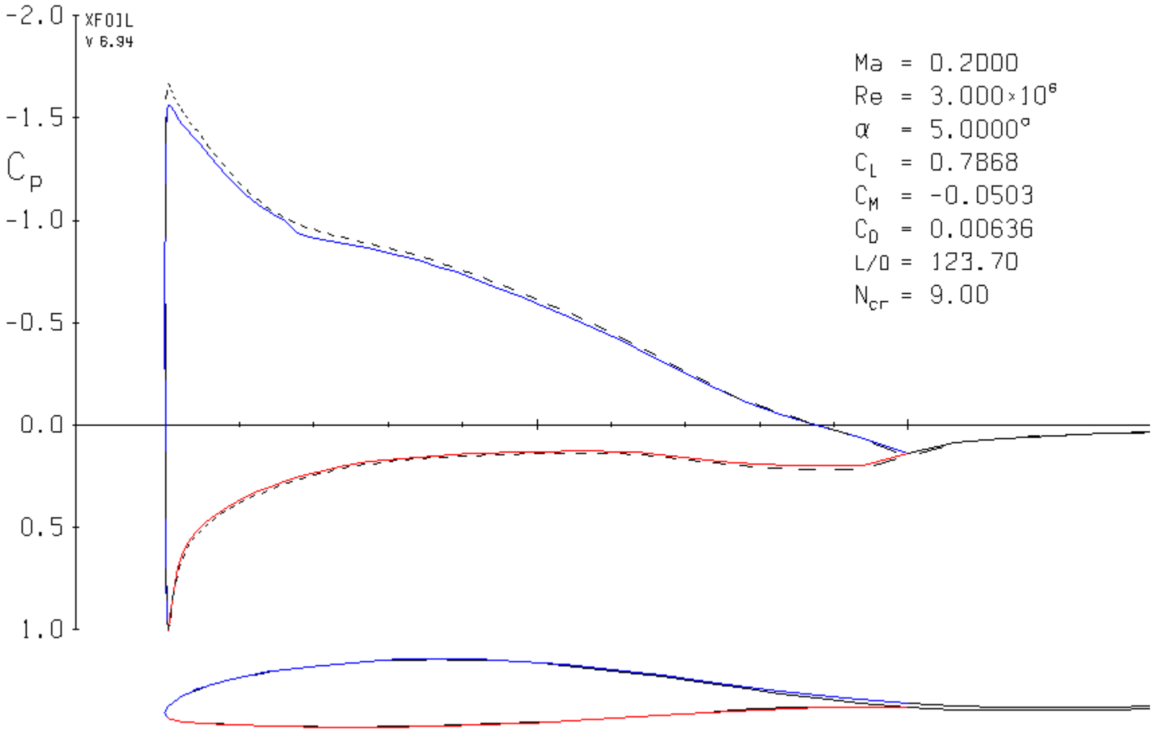
Concerning noise emission, there exists a specific command in XFOIL that computes and returns different parameters of the boundary layer, like δ* or θ*, at several locations on the airfoil profile. This command is very useful but such analysis is not performed in the present work. This would constitute a future work on this project.

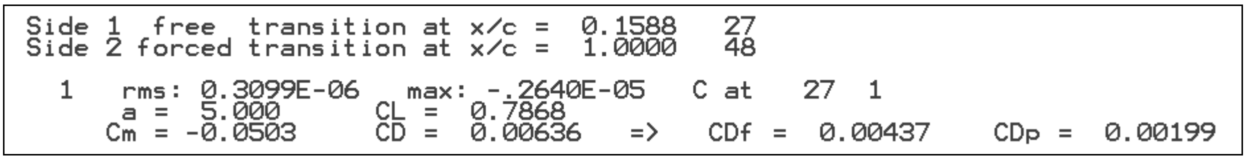




```
Side 1  free  transition at x/c =  0.5061   39
Side 2  free  transition at x/c =  0.7631   46

  1    rms: 0.3175E-06   max: -.2995E-05   C at   39   1
        a =  2.000        CL =  0.4500
       Cm = -0.0512       CD =  0.00409    =>   CDf =  0.00352    CDp =  0.00057
```





```
Side 1  free   transition at x/c =  0.1588   27
Side 2 forced transition at x/c =  1.0000   48

  1    rms: 0.3099E-06   max: -.2640E-05   C at   27   1
        a =  5.000        CL =  0.7868
       Cm = -0.0503       CD =  0.00636    =>   CDf =  0.00437    CDp =  0.00199
```

6. Future work and limitations

The optimisation program presented in this project is not completed yet and many other tests and validations have to be performed.

Most of the time, the performance of the optimised profile is worse out of the design range of angles of attack. This is why it is important to accurately define the desired characteristics and constraints on a large range of angles of attack. In the present work, two aerodynamic characteristics were defined as objectives. A future work would be to include more such objectives to be able to design the entire polar. Nevertheless, this would require high computer costs, the main limitation of this project.

A further objective of the future work would be to include the transition laminar-turbulent position as a design objective. This is a very important parameter especially for the design of stall regulation and active control wind turbine blades. The basic idea would be to impose a minimum distance to the transition point in order to prevent stall up to a given angle of attack. This would be very easy to implement in the design code since the position of the transition point is provided in the XFOIL output file (see Table 3, part 3.1). Once again, this has not been implemented in this project to limit computational costs.

Still concerning the boundary layer, a special analysis on roughness sensitiveness can be easily added to the design code. The aim would be to impose aerodynamic characteristics as design objectives to the corresponding “dirty” profile (rough leading edge).

Noise emission analysis could be another future work. The aim of this study would be to impose a maximum boundary layer thickness at the trailing edge.

Finally, it would be interesting to compare XFOIL results with another flow solver like FLUENT or EllipSys2D.

Concerning the limitations of this design code, the algorithm shows some difficulties to converge when the optimisation objectives impose lower lift coefficient than the initial airfoil profile. This problem does not exist anymore when the design objectives concern the lift-drag ratio. This has to be investigated as a future work.

One of the main limitations of this design code is the design algorithm itself. It is explained in part 4.3 that the optimisation algorithm implemented in the MATLAB function ***fgoalattain*** is sensitive to local minima. As long as the optimisation process reaches the objectives, it does not matter if the solution is a local or a global minimum. But when no solution is found, it does not necessarily mean that no feasible solution exists. In most of the time, some optimisation parameters need to be changed (e.g. the minimum change in variables for finite-difference gradients). This requires some level of aerodynamic and algorithmic expertise and common sense on the part of the user.

7. Conclusion

A method for design of wind turbine airfoils is presented. This method is based on direct numerical optimisation of a B-spline representation of the airfoil shape. The generation and the optimisation processes are implemented in MATLAB, and an interface between MATLAB and XFOIL is created.

Airfoil curves are generated based on 14 control points using the B-Spline approach. It has been shown that two control points, free to move in both x and y directions, can reproduce the leading edge more accurately than with a single control point. Most of the control points are fixed in x-direction in order to limit the number of design variables. The control point distribution defined in the present work is able to reproduce different airfoils with various geometries with maximal geometry difference less than 0.5%.

The design method is demonstrated by the improvement of the MH43 profile performances at different angles of attack under specific conditions: Reynolds number = 3.10^6 and Mach number = 0.2. First the lift-drag ratio at 2 degrees angle of attack is increased from 93 to 120. Then, lift-drag ratio at 5 degrees angle of attack is increased from 110 to 130. Finally, the lift-drag ratio is increased simultaneously at both angles: 2 and 5 degrees. In this particular case, this results in an increase in performances of the initial MH43 profile over the entire range of angles of attack. Moreover a roughness analysis is performed in XFOIL and shows that the optimised profile is less sensitive to leading edge roughness than the initial MH43 profile.

To conclude, the design code presented here has proved to be able to design and optimised airfoils at some specific points and under specific operational conditions. Nevertheless, the program is not completed yet. Other validations have to be performed and several parameters still have to be implemented. For example, objectives should be defined on a large interval of angles of attack and transition properties as well as boundary layer characteristics should be included in the objectives. Finally it would be necessary to check XFOIL results with another flow solver like FLUENT or EllipSys2D.

References

[1] Kristian S. Dahl, Peter Fuglsang, *Design of the Wind Turbine Airfoil Family RISØ-A-XX*, Risø National Laboratory, Roskilde, Denmark, December 1998

[2] Mark Drela, MIT AERO & Astro, Harold Youngren, Aerocraft, Inc., *XFOIL 6.9 User Primer*, 30 November 2001

[3] The MathWorks, http://www.mathworks.com/ 1994-2009 The MathWorks, Inc.

[4] Press W. H.,Flannery B. P. , Teukolsky S. A. and Vetterling W. T. *Numerical recipes: The Art of Scientific Computing*, Third Edition. Cambridge University Press, 2007

[5] Carl de Boor (1978), *A Practical Guide to Splines*. Springer-Verlag. pp. 113–114.

[6] Peter Fuglsang, Christian Bak, *Status of the RISØ Wind Turbine Airfoils*, Risø National Laboratory, Roskilde, Denmark, presented at the European Wind Energy Conference, Madrid, 16-19 June 2003