1 XFOIL aerodynamic analysis: `run_xfoil.sh`

The goal of an aerodynamic design optimization problem is the improvement of the baseline geometry performance by changing its shape. When a robust version of this problem is faced, an optimal solution that is less vulnerable with respect to uncertainties in operating conditions and geometric shape has to be obtained. The baseline airfoil is the NACA 2412. The airfoil is parametrized as a linear combination of an initial geometry $(x_0(s), y_0(s))$, and some modification functions $y_i(s)$. Moreover, to describe geometry uncertainties, further $z_i(s)$ modification functions are introduced. So, the airfoil shape, including uncertainties, is described by

$$x(s) = x_0(s), \qquad y(s) = k\left(y_0(s) + \sum_{i=1}^{n} w_i y_i\right) + \sum_{j=1}^{m} U_j z_j \tag{1}$$

where the airfoil shape is controlled by the design parameters w_i and by the scale factor k. The uncertainty on shape and thickness of the airfoil is described by the U_j random variables.

The `run_xfoil.sh` script builds a parametric airfoil using Equation 1 and starting from the NACA 2412 baseline. Then, it computes the airfoil performance using an aerodynamic analysis code, namely Prof. Drela's XFOIL code. It is based on a second order panel method interactively coupled to a boundary layer integral module. Moreover, the laminar to turbulent flow transition is predicted using the e^N method.

In order to calculate the aerodynamic performance, the operating conditions (Mach numeber and Reynolds number) must be specified. In addition, further parameters that have to be fixed are lift coefficient, c_l, and optionally, maximum thickness of the airfoil, t/c.

All mentioned parameters are specified in an input file (`eval_obj.in`) and the results of the simulation are written in an output file (`eval_obj.out`). The format of the files is explained in the following sections.

Input file: `eval_obj.in`

The input file allows the specification of:

- Operating conditions: they are defined by the variables `MACH` (Mach number), `Reynolds_number` (Reynolds number) and `CL` (lift coefficient). The specified lift coefficient is obtained by changing the angle of attack of the airfoil.
- Constraints on laminar to turbulent flow transition: the constraints are defined using the variables `XTRUP` (boundary layer transition point on the airfoil upper surface) and `XTRLO` (boundary layer transition point on the airfoil lower surface). `XTRUP` set equal to 0.5, for example, means that the transition point cannot be set at an x/c greater than 0.5.

- Geometric constraints: `MAX_THICK` fixes the maximum airfoil thickness

Furthermore, design parameters (`W`) and uncertainties on shape (`U`) are specified in the input file. Vectors `W` and `U` can have a maximum size of 20 elements (from 0 to 19). The number between parentheses determines the type of modification function. If a particular index is not specified, the weight of the related modification function is set equal to 0.

An example of a possible input file is given in Figure 1.

```
CL = 0.5
MACH = 0.0
Reynolds_number = 500000
XTRLO = 0.95
XTRUP = 1.0
MAX_THICK = 0.12
W(0) = 1.6592688171214585
W(1) = -0.25971899556688016
W(2) = 0.48881342283888141
W(3) = 0.14247126031957394
W(4) = 0.39947881671820701
W(5) = -0.20633251349356299
W(6) = -0.87011425209234317
W(7) = 0.42458107394040112
W(8) = 0.7677216309396836
W(9) = -0.64594578458833207
U(10) = 0.042274686344264678
U(11) = 0.19844599191645305
U(12) = 0.061807615900600277
U(13) = 0.0028126036172765947
U(14) = 0.16218435710928967
U(15) = -0.027786505828329044
U(16) = 0.13324753950076356
U(17) = -0.027167195033320934
U(18) = 0.11419903712794166
U(19) = 0.023761492429478476
```

Figure 1: Example of input file

Output file: `eval_obj.out`

The information given in the output file is shown in Figure 2.

```
ERROR = 0
MACH = 0.0
a = 1.196
CL = 0.5000
Cm = -0.0806
CD = 0.682937E-02
CDf = 0.00469
CDp = 0.00214
Airfoil_maximum_thickness = 0.124268262E+00
Airfoil_leading_edge_radius = 0.790555338E-02
Airfoil_trailing_edge_angle = 0.168762553E+02
```

Figure 2: Example of output file

- `ERROR` is a flag that if it is set equal to 1 it means that the aerodynamic analysis performed by XFOIL did not converged.
- `MACH` reports, for the sake of completeness, the Mach number at which the simulation has run.
- `a` is the airfoil angle of attack at which the required lift coefficient was obtained.
- `CL` is the lift coefficient actually obtained (in some cases it might be sightly different from the specified one).
- `CD` is the drag coefficient. In addition, its decomposition in friction drag (`CDf`) and pressure drag (`CDp`) is provided.
- `Airfoil_maximum_thickness` is the actually obtained maximum thickness of the airfoil. Its value might be different with respect to the one set in input when the uncertainty on shape (vector `U`) is considered. Indeed, the modification functions related to the uncertainty are applied to the airfoil after the scaling to the assigned maximum thickness (see Equation 1).
- `Airfoil_leading_edge_radius` and `Airfoil_trailing_edge_angle`, as their names indicate, are, respectively, the leading edge radius and the trailing edge angle of the airfoil.

Furthermore, a plot reporting the pressure coefficient, c_p, the output airfoil, and some relevant results are presented in the file *plot.ps*. It can be visualized with any postscript viewer (*gv plot.ps*, *gs plot.ps* or *evince plot.ps*) and an example is reported in Figure 3.

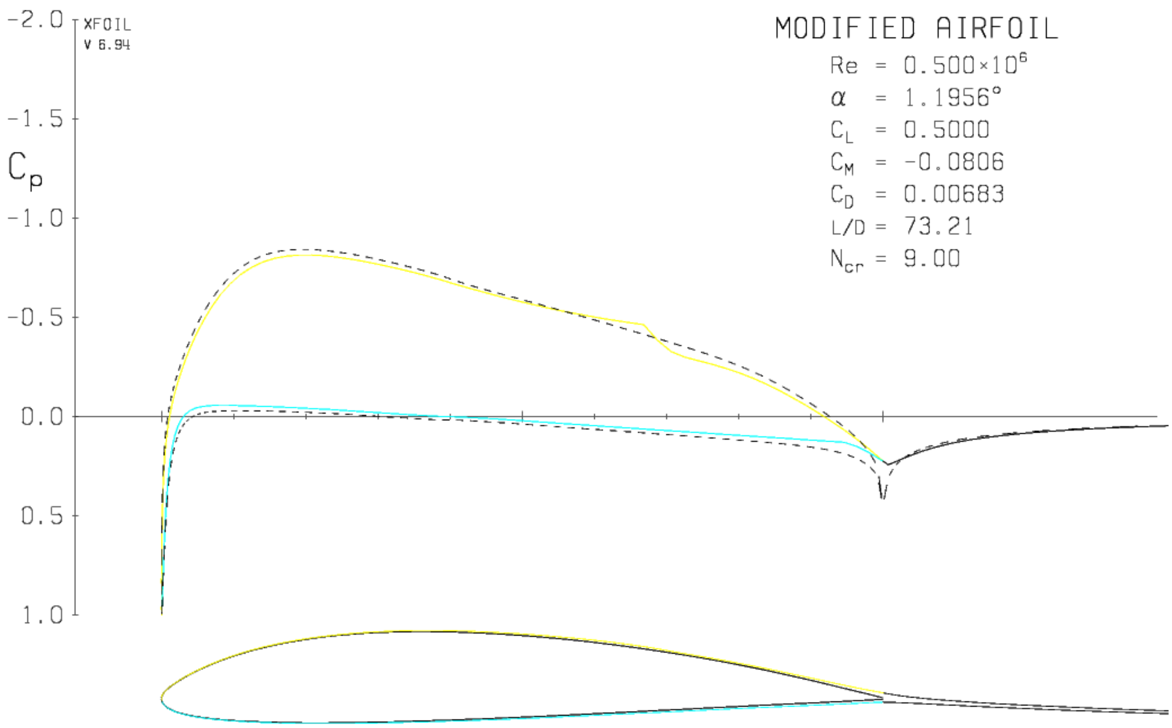

Figure 3: *plot.ps* example

Finally, launching the script with the flag `-dbg` (`run_xfoil.sh -dbg`) inhibits the cancellation of the intermediate data and executable files. This may be useful for debugging purposes.

2 R script: `adg.risk_fnct.R`

The script `adg.risk_fnct.R` allows the calculation of the risk functions, Value at Risk (VaR) and Conditional Value at Risk (CVaR), as well as their confidence intervals by means of the Bootstrap method.

The following arguments can be given as input to the script:

- `--Nboot=someValue`

  This argument must be numeric. It indicates the number of iterations of the bootstrap method. If it is equal to 0 no bootstrap is performed.

- `--DB=someValue`

  This argument must be a string. The name of the file containing the Data table must be introduced. In the examples the file is called `DATABASE.des`.

- `--Vname=someValue`

  This argument is again a string character. Here the name of the variable to be extracted and elaborated from the file `DATABASE.des` is introduced. Usually, in `DATABASE.des`, `OB_00000` is the label of the objective function.

- `--alpha=[0,1]`

  This argument must be numeric. It is the percentile value for the calculation of the risk functions VaR and CVaR. It is a number in the range [0,1].

- `--subset=<n>`

  This is a numeric argument. It performs the calculation of the risk functions by taking randomly n samples of all the provided data instead of taking the complete set.

- `--plot=[TRUE,FALSE]`

  This argument gives the possibility of producing pdf plots in output with the obtained result. As default is set equal to `TRUE`.

- `--help`

  This argument prints a text giving help for the use of the script.

After explaining each of the arguments, let us give an example:

```
adg.risk_fnct.R --Nboot=10000 --DB=DATABASE.des --Vname=OB_00000
--alpha=0.9  --subset=10 --plot=TRUE
```

An example of the information given as output is shown in the following figure.

```
Vname =  OB_00000
alpha =  0.9
DB size =  10
 VaR =  0.00613905
CVaR =  0.00616482


 VaR_Bootstrap:

ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot(data = f, statistic = varboot, R = Nboot, alpha = alpha)


Bootstrap Statistics :
      original       bias     std. error
t1* 0.00613905 3.800632e-06 1.384749e-05
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 10000 bootstrap replicates

CALL :
boot.ci(boot.out = b1, type = "perc")

Intervals :
Level     Percentile
95%   ( 0.0061,  0.0062 )
Calculations and Intervals on Original Scale
VaR.ci  0.95    0.00612551    0.00616482


CVaR_Bootstrap:

ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot(data = f, statistic = cvarboot, R = Nboot, alpha = alpha)


Bootstrap Statistics :
      original        bias     std. error
t1* 0.00616482 -9.700608e-06 1.345123e-05
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 10000 bootstrap replicates

CALL :
boot.ci(boot.out = b2, type = "perc")

Intervals :
Level     Percentile
95%   ( 0.0061,  0.0062 )
Calculations and Intervals on Original Scale
CVaR.ci  0.95    0.00612816    0.00616482
```

Finally, an example of the produced pdf plots is provided in the following figure.