INSTITUTO TECNOLÓGICO DE AERONÁUTICA

Alejandro Arturo Rios Cruz

Homework 03 - Aerostructural Optimization

AP-266

São José dos Campos
02/05/2018

1 Complex Step Test

The derivation of an equivalent test for the Finite Differences Test for verification of Forward AD but now using Complex Step Method instead of Finite Differences is presented below:

Using the Taylor series expansion to derive the complex step approximation for one variable using imaginary step:

$$f(x+ih) = f(x) + ih\frac{df}{dx} - \frac{h^2}{2}\frac{d^2f}{dx^2} + O(ih^3) \tag{1.1}$$

Taking the imaginary part from both sides:

$$Im[f(x+ih)] = h\frac{df}{dx} + O(h^3) \tag{1.2}$$

Dividing both sides by h:

$$\frac{Im[f(x+ih)]}{h} = \frac{df}{dx} + O(h^2) \tag{1.3}$$

Isolating $\frac{df}{dx}$:

$$\frac{df}{dx} = \frac{Im[f(x+ih)]}{h} - O(h^2) \tag{1.4}$$

For very small values of h:

$$\frac{df}{dx} \approx \frac{Im[f(x+ih)]}{h} \tag{1.5}$$

For multivariable functions:

$$\frac{\partial f_1}{\partial dx_1} \approx \frac{Im[f(x_1+ih, x_2)]}{h} \tag{1.6}$$

$$\frac{\partial f_2}{\partial dx_2} \approx \frac{Im[f(x_2+ih, x_1)]}{h} \tag{1.7}$$

The advantage of using the test based on the complex step method is that because there is no subtraction operation (differently from finite differences), the only source of numerical error is the truncated error (i.e. $O(h^2)$. When we decrease h to a very small value, it is ensure that the truncation error is of the same order as the numerical precision of the evaluation of the function.

To show this the next function (also used in other works for comparison) was evaluated using both, finite difference method and complex step method:

$$f(x) = \frac{e^x}{(cos(x))^3 + (sin(x))^3} \tag{1.8}$$

The results of the comparision are presented in figure 1.

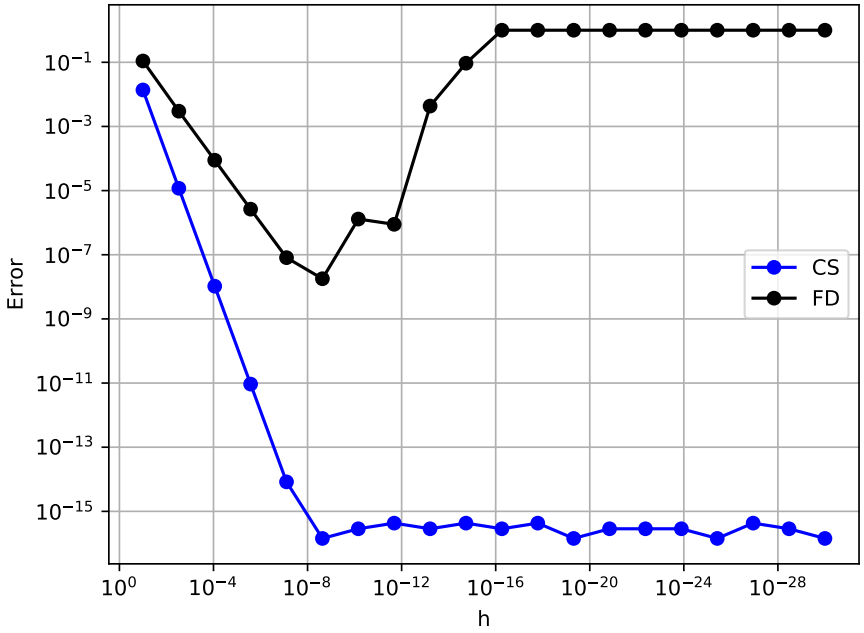

Figura 1 – Comparison between CS and FD relative error applied to the given function.

In the figure above is possible to observe how the FD method present start to increase the error value results for step values below $10^{-}8$. This behavior is due to the subtracting cancellation inherent to the method. In the other hand, the complex step method doesn't present this problem because there is no f(x) generating a subtracting cancellation error. It is also observable that the complex step method has a second order convergence.

2 Optimization with the Lifting-Line Theory

2.1 Python interface

The resulting input seeds from the reverse AD code, obtained changing the specified resb input seed, are presented below:

$$\mathbf{Xb} = \begin{bmatrix} -4.04950224 & 2.08471501 & 2.54897055 & 0.22408489 & -0.80826822 \\ -14.75943574 & 14.27401984 & -5.0427886 & 0.65624498 & 4.87195951 \\ 0.59555548 & -0.83782007 & 0.03601486 & -0.38566095 & 0.59191069 \end{bmatrix}$$

$$Gamab = \begin{bmatrix} 54.67416437 & -7.0955048 & 5.45562211 & 15.81322861 \end{bmatrix}$$

$$alpha0b = \begin{bmatrix} -98.64728057 & 16.85547484 & 0. & -19.72945611 \end{bmatrix}$$

$$chordsb = \begin{bmatrix} -8.30967337 & -2.40264656 & -1.53445338 & -2.88949738 \end{bmatrix}$$

2.2 Solving the LLT problem

The circulation values that cancel the residuals are presented below:

$$Gama = \begin{bmatrix} 0.23818756 & 0.26183574 & 0.26183574 & 0.23818756 \end{bmatrix}$$

The corresponding values of Sref, CL, CD, L and D are presented in table 1

Tabela 1 – Values of function obtained from appropriated values of Gama.

Variable	Value
Sref	8.0
CL	0.5000232936525408
CD	0.008104941727927098
L	4.1200164450136700
D	2.0000931746101633

2.3 Solving the Adjoint problem

The required values for this exercise are presented below:

$$\overrightarrow{\psi}_{C_L} = \begin{bmatrix} -0.06350275 & -0.02401232 & -0.02401232 & -0.06350275 \end{bmatrix}$$

$$\frac{dC_L}{d\mathbf{X}} = \begin{bmatrix} 3.18127304e{-}02 & -7.80508079e{-}03 & -4.80152993e{-}02 & -7.80508079e{-}03 & 3.18127304e{-}02 \\ 2.58640389e{-}01 & -2.73076792e{-}02 & -8.32667268e{-}17 & 2.73076792e{-}02 & -2.58640389e{-}01 \\ 2.06364752e{-}02 & -1.30147667e{-}02 & -1.52434170e{-}02 & -1.30147667e{-}02 & 2.06364752e{-}02 \end{bmatrix}$$

$$\frac{dC_L}{d\overrightarrow{\alpha}_0} = \begin{bmatrix} 1.25287471 & 1.34913032 & 1.34913032 & 1.25287471 \end{bmatrix}$$

$$\frac{dC_L}{d\overrightarrow{c}} = \begin{bmatrix} -0.66395098 & -0.81949111 & -0.81949111 & -0.66395098 \end{bmatrix}$$

To ensure that the results are correct rCL is also presented:

$$r_{CL} = \begin{bmatrix} -5.61616725e-17 & -1.13190707e-16 & -1.68701858e-16 & 1.65882932e-16 \end{bmatrix}$$

Repeating the same process to compute the total derivatives of Sref and CD gives:

The results for CD:

$$\overrightarrow{\psi}_{C_D} = \begin{bmatrix} -0.00559664 & -0.00644102 & -0.00644102 & -0.00559664 \end{bmatrix}$$

$$\frac{dC_D}{d\mathbf{X}} = \begin{bmatrix} -8.33240668e{-}03 & 1.95727664e{-}03 & 1.27502601e{-}02 & 1.95727664e{-}03 & -8.33240668e{-}03 \\ 5.63725428e{-}02 & 7.47167716e{-}02 & -4.16333634e{-}17 & -7.47167716e{-}02 & -5.63725428e{-}02 \\ -6.04878123e{-}03 & -4.93417262e{-}03 & 2.19659077e{-}02 & -4.93417262e{-}03 & -6.04878123e{-}03 \end{bmatrix}$$

$$\frac{dC_D}{d\overrightarrow{\alpha}_0} = \begin{bmatrix} 0.11041866 & 0.36188837 & 0.36188837 & 0.11041866 \end{bmatrix}$$

$$\frac{dC_D}{d\overrightarrow{c}} = \begin{bmatrix} -0.07350954 & -0.09973341 & -0.09973341 & -0.07350954 \end{bmatrix}$$

To ensure that the results are correct rCD is also presented:

$$r_{CD} = \begin{bmatrix} -0.00000000e+00 & -1.38777878e-17 & 3.55618313e-17 & 1.86482774e-17 \end{bmatrix}$$

The Sref function does not require an adjoint solution, given that the value of Sref is independent of the circulation. The results for Sref are presented below:

$$\overrightarrow{\psi}_{Sref} = \begin{bmatrix} 0 \end{bmatrix}$$

$$\frac{dSref}{d\mathbf{X}} = \begin{bmatrix} 0. & 0. & 0. & 0. & 0. \\ -1. & 0. & 0. & 0. & 1. \\ 0. & 0. & 0. & 0. & 0. \end{bmatrix}$$

$$\frac{dSref}{d\overrightarrow{\alpha}_0} = \begin{bmatrix} 0. & 0. & 0. & 0. \end{bmatrix}$$

$$\frac{dSref}{d\overrightarrow{c}} = \begin{bmatrix} 2. & 2. & 2. & 2. \end{bmatrix}$$

To ensure that the results are correct rSref is also presented:

$$r_{Sref} = \begin{bmatrix} 0. & 0. & 0. & 0. \end{bmatrix}$$

2.4 Twist optimization

Now we will perform the twist optimization for the wing plan-form presented in the previous exercise (2.1). The optimization problem is:

$$\begin{aligned} & C_D \\ \text{w.r.t} \quad & \alpha_0 \\ \text{s. t.} \quad & CL = 0.5 \end{aligned}$$

The SLSQP implementation algorithm from SciPy was selected for the constrained problem optimization. The obtained optimum values of α_0, CL and CD are presented on table 2.

Tabela 2 – Values of function obtained from apropriate values of Gama.

Variable	Value
alpha0 [deg]	-0.59705226 | 0.54270033 | 0.54270033 | -0.59705226
CL	0.4999999999992052
CD	0.00795774715456947

The optimal Gama and alpha0 are also shown in figures 2 and 3.

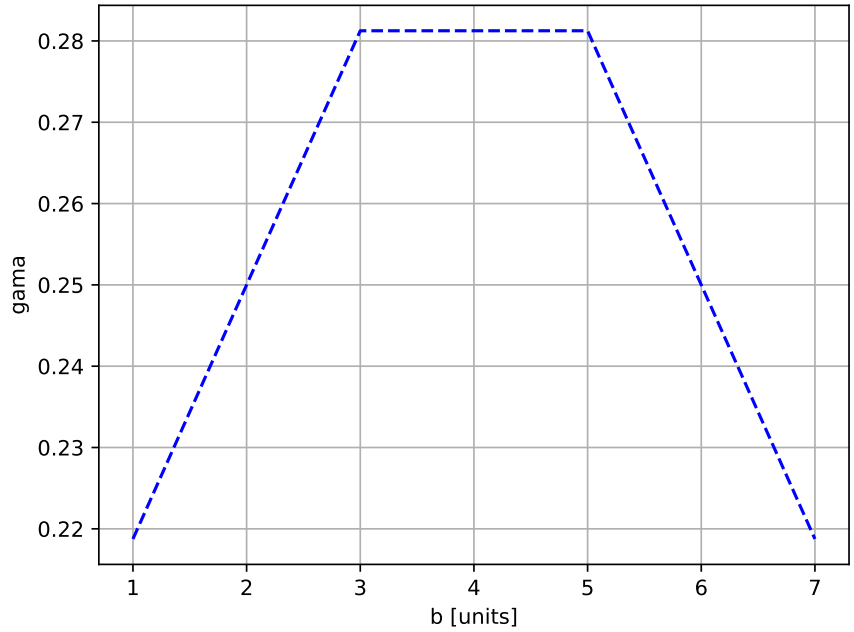

Figura 2 – Optimal Gama distribution for a wing with 4 horseshoe vortex.

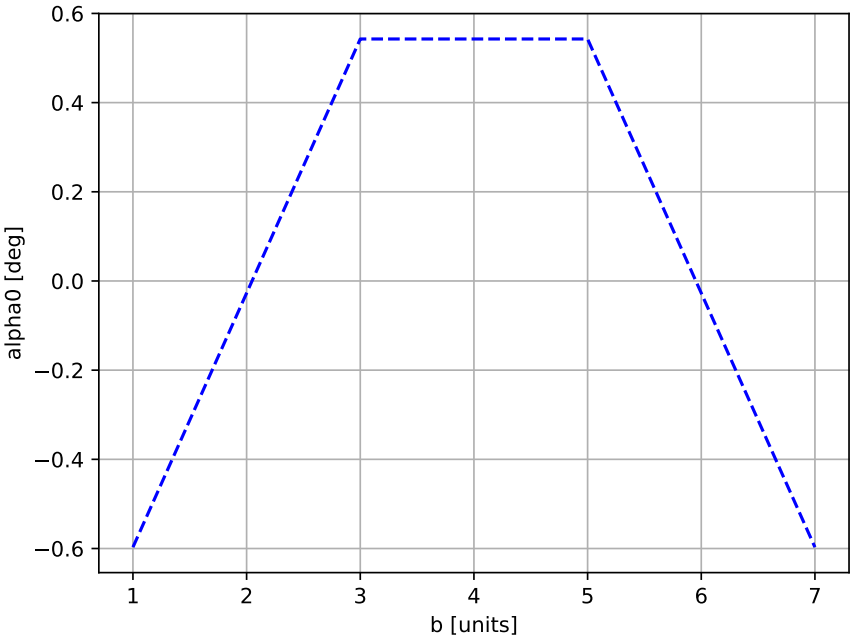

Figura 3 – Optimal alpha0 distribution for a wing with 4 horseshoe vortex.

2.5 Increasing the problem complexity

The optimization problem is increased in complexity in this exercise by increasing the number of vortex for the aerodynamic problem. Solving the optimization using 8 and

16 horseshoe vortex the resulting optimum values of α_0, CL and CD for each case are presented in table 3 and in the vectors numbered as 2.1 and 2.2.

Tabela 3 – Values of function obtained from appropriated values of Gama.

Variable	Value nvor = 8	Value nvor = 16
CL	0.4999999999999771	0.500000000000029
CD	0.008841941282910146	0.00936205547647221

$$alpha0(nvort = 8) = \begin{bmatrix} -1.30168995 \\ -0.0284285 \\ 0.65720297 \\ 0.96884057 \\ 0.96884057 \\ 0.65720297 \\ -0.0284285 \\ -1.30168995 \end{bmatrix} \tag{2.1}$$

$$alpha0(nvort = 16) = \begin{bmatrix} -2.023389 \\ -0.93873865 \\ -0.21732611 \\ 0.30210862 \\ 0.6803883 \\ 0.94679695 \\ 1.11732055 \\ 1.20064798 \\ 1.20064798 \\ 1.11732055 \\ 0.94679695 \\ 0.6803883 \\ 0.30210862 \\ -0.21732611 \\ -0.93873865 \\ -2.023389 \end{bmatrix} \tag{2.2}$$

Figures 4 and 5 show the evolution of the circulation and the local angle of attack varying n_vort from 8 to 40 with a step of 8.

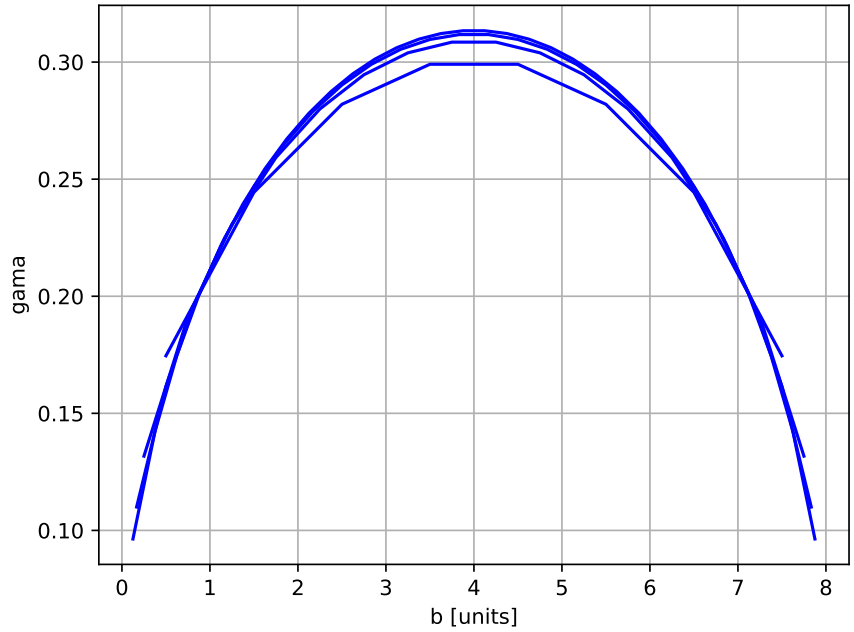

Figura 4 – Optimal Gama distribution for different values of n_vort (from 8 to 40 with a step of 8).

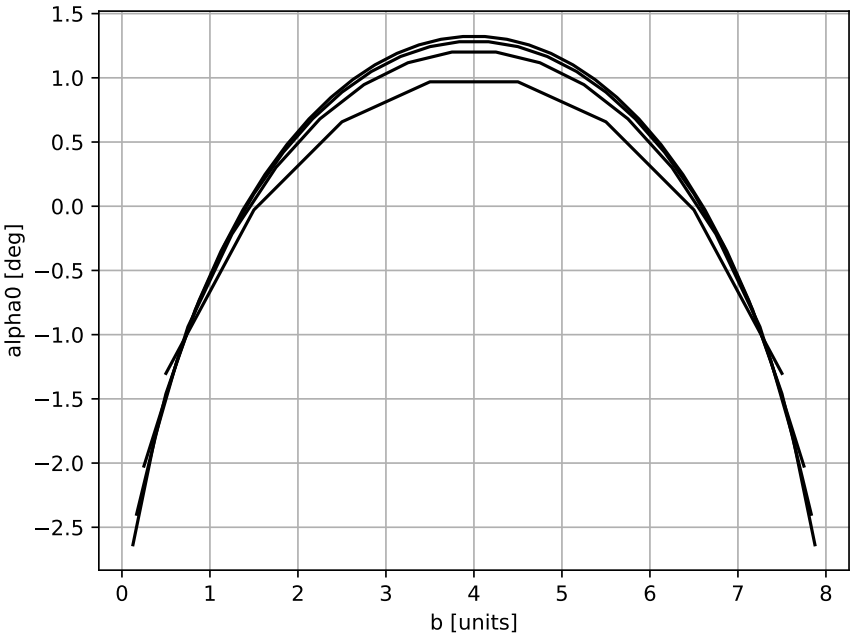

Figura 5 – Optimal alpha0 distribution for different values of n_vort (from 8 to 40 with a step of 8).

Similar figures but for the evolution of CD and CL are shown in figures 6 and 7.

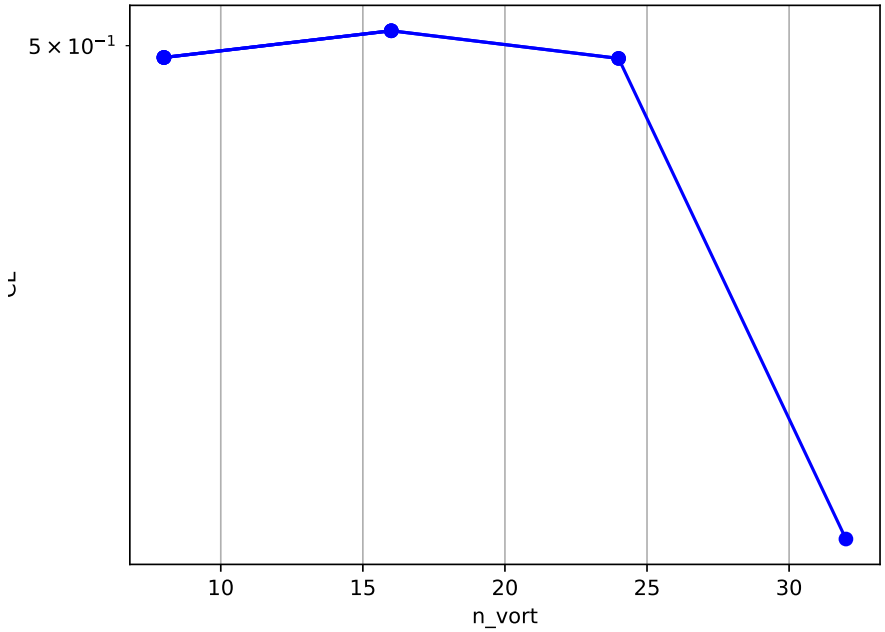

Figura 6 – Lift coefficient variation for different values of n_vort (from 8 to 40 with a step of 8)

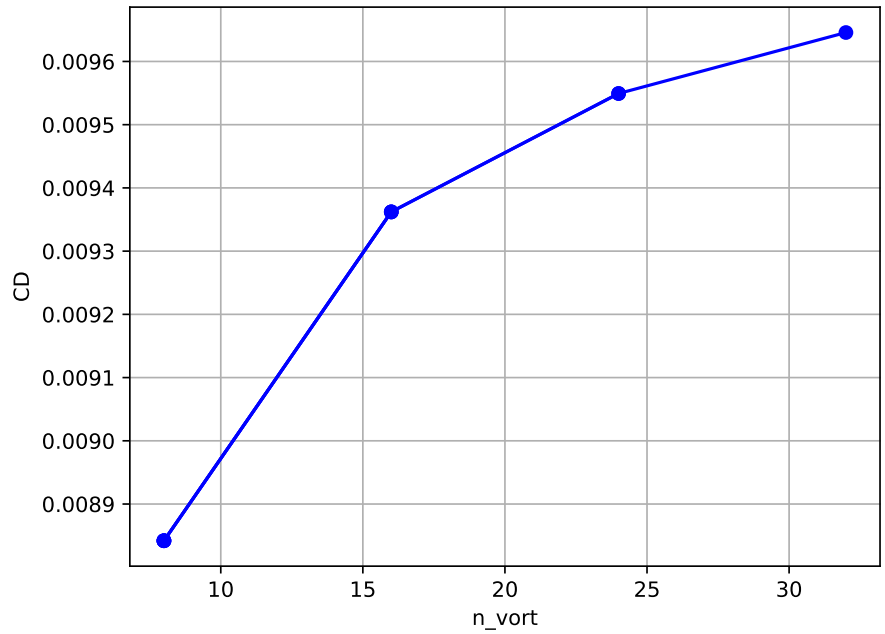

Figura 7 – Drag coefficient variation for different values of n_vort (from 8 to 40 with a step of 8)

The last figure shows that the number of vortex selected for the optimization have impact in the obtained value of drag coefficient. A greater number of vortex will produce

more accurate results. Then, a convergence analysis is necessary to know which value of n_vortex to use.

2.6 Comparing with analytic solutions

In this exercise the results of the optimized wing in terms of circulation will be compared with the analytical solution of an elliptical lift distributed wing. The results of this comparison for different number of horseshoe vortex discretization (4, 8 ,16 and 40) are presented in figure 12.

the elliptical Γ distribution and compare it against the distributions of the optimized configurations obtained with 4, 8, 16, and 40 horseshoe vortex.

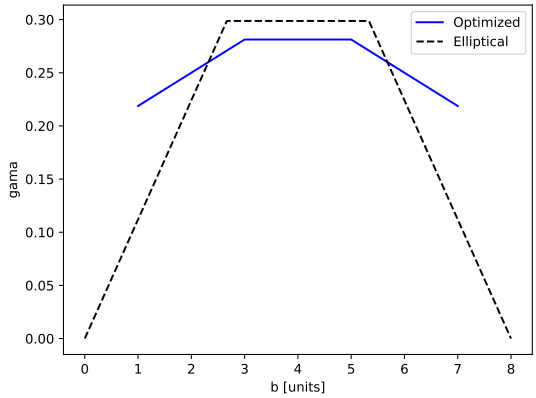

Figura 8 – n_vort = 4

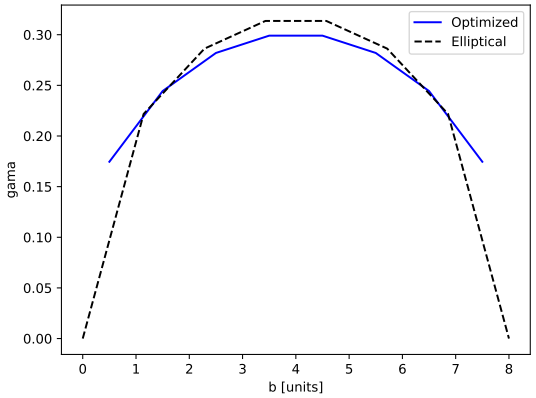

Figura 9 – n_vort = 8

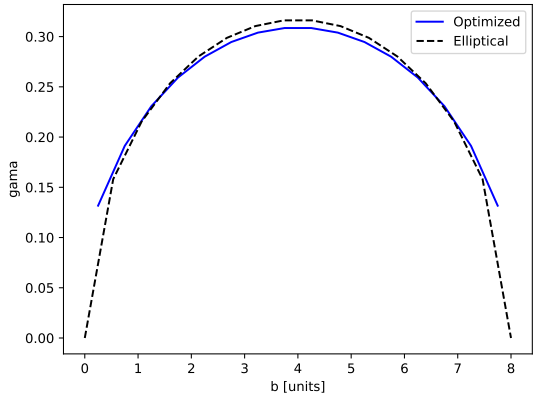

Figura 10 – n_vort = 16

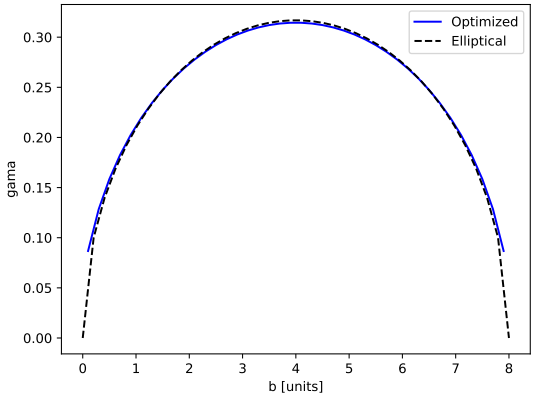

Figura 11 – n_vort = 40

Figura 12 – Comparison of circulation for the theoretical elliptical loaded wing vs the circulation of the optimized wing.

From the last figure, it result interesting to observe that the tendency of the optimization is to twist the wing in such a way (increasing alpha at the root and decreasing

at the tips)that the circulation distribution approximates to the elliptical distribution. In other words, the optimization tendency is to proof what was before demonstrated by Prandtl. By increasing the vortex discretization to a higher number may result in the convergence of both curves, analytic and optimized. Another way to approximate both circulation distributions, without the need of increase too much the number of vortex will be modify the linear distribution of the discretization using a cosine distribution instead.

A comparison of the CD values of the numerical optimization against the analytic result are presented in figure 13.

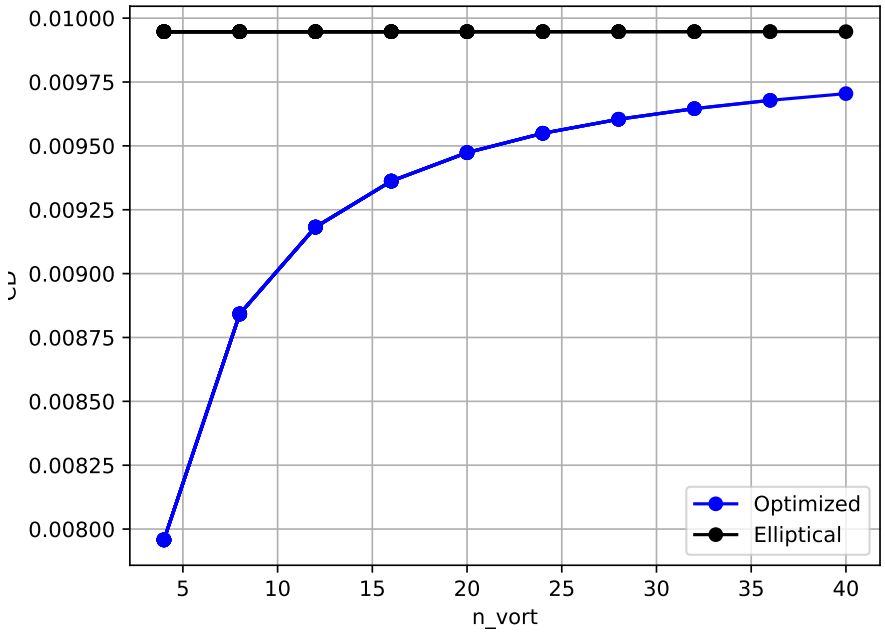

Figura 13 – Drag coefficient variation for different values of n_vort (from 8 to 40 with a step of 8)

In the figure above it is observable that the tendency of the optimized wing is to approximate to the analytic result. Nevertheless, it can be notice that the optimized wing results always in better (lower) values of drag coefficient. This result may be consequence of two unfair factors in terms of comparison. The first one, is that if we compare the drag coefficient for a given number of vortex (preferably small) the optimizer will have advantage and can provide a wing with less drag than the analytic wing. This is because neither the optimized nor the analytical wings can represent an elliptical distribution with just 4 vortex or discretization. The second unfair factor is related to the way that the drag coefficient is calculated. The analytical CD is calculated with an equation that doesn't takes into account the discretization of the wing. For this reason we have observed a constant drag coefficient in figure 13. In the other hand, the optimization drag coefficient evaluation is function of the discretization. This explains the behavior observed in figure 13. If the discretization is increased to a very high value, both curves in figure 13 will

eventually meet. This is better shown in figures 16

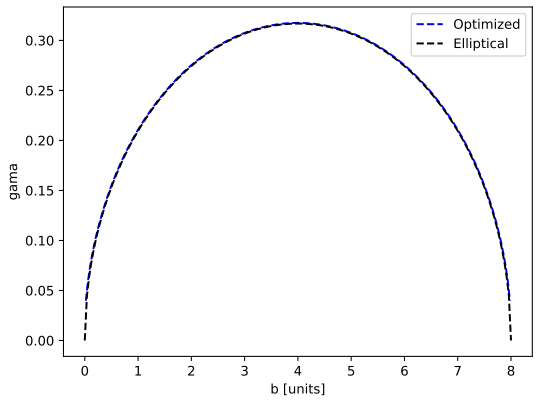

Figura 14 – n_vort = 200

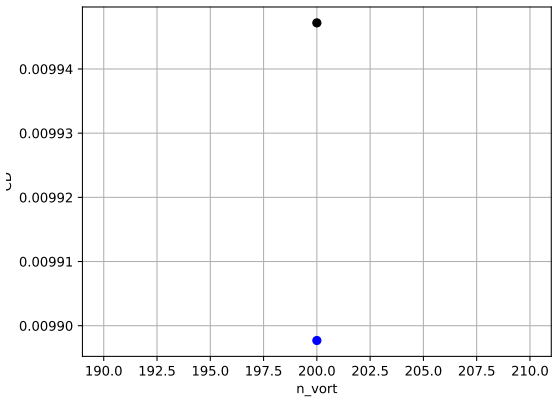

Figura 15 – n_vort = 200

Figura 16 – Comparison results using n_vort = 200.

A comparison of the number of function evaluations required by the optimizer versus the numbers of design variables is shown in figure 17.

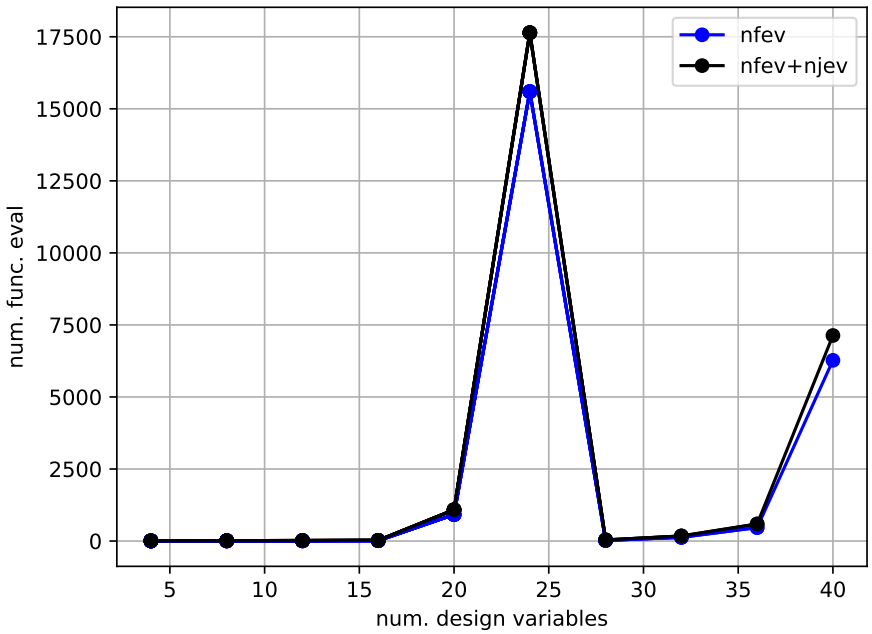

Figura 17 – Number of function evaluations required by the optimizer vs numbers of design variables.

Another figure increasing the step to avoid the n_vort generating the outliers in the previous figure is presented in 18.

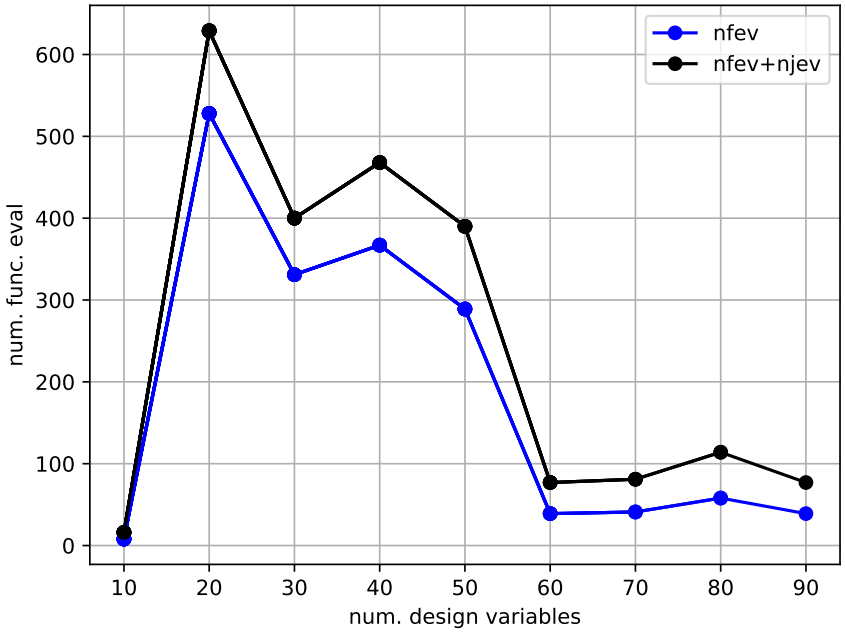

Figura 18 – Number of function evaluations required by the optimizer vs numbers of design variables.

From both figures above it is observable that the number of function evaluations doesn't precisely scale with the number of design variables. This shows the advantages of using the reversal adjoint method in such type of optimization problems.