

MEC 6602E

Transonic Aerodynamics

NL-VLM code

Frédéric Plante Ph.D.

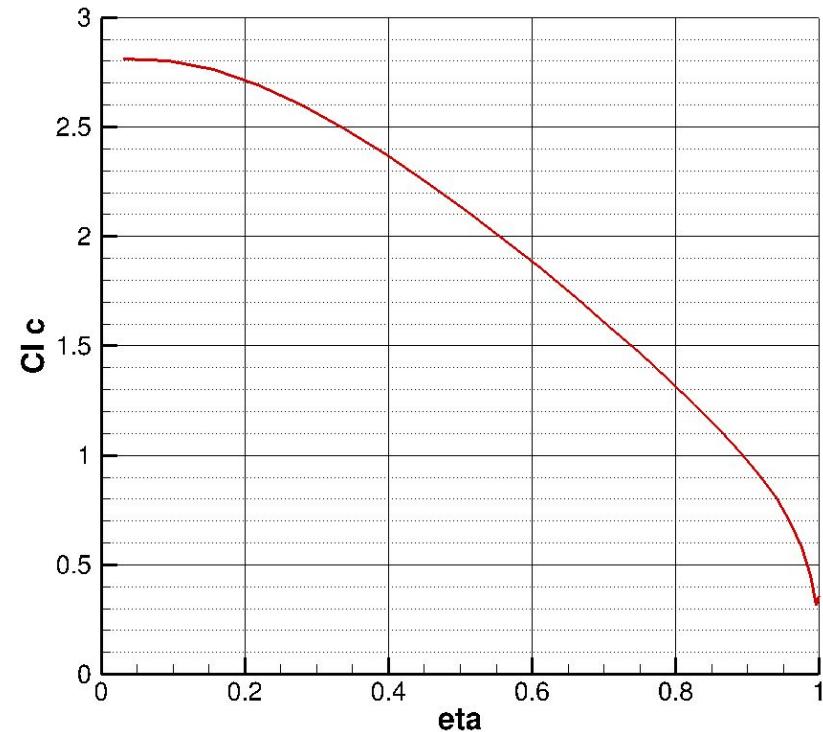
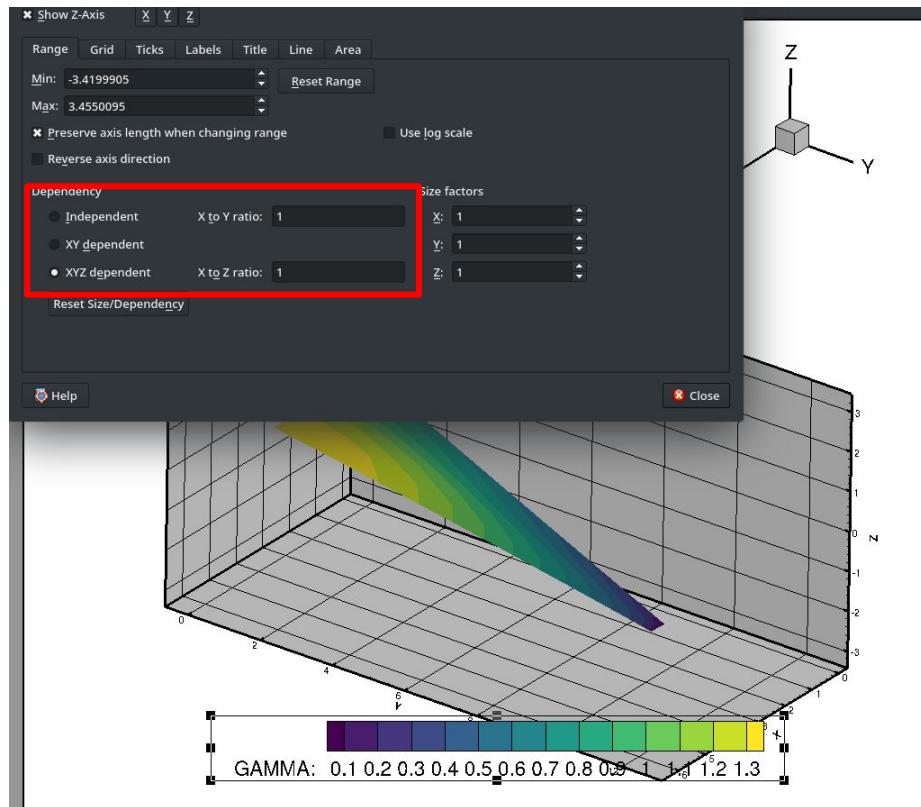
NL-VLM - How to run

- Non-linear VLM code couples a VLM code for wing analysis with 2D/2.5D polar including non-linear effect;
- The code is called from a python script (ex.: [runIsoAoA.py](#) and [runIsoCl.py](#));
- Comments in the code describe the input;
- You can modify the python script to carry out your optimization. A simple parametrization in twist and planform is given. You can make it more complex and even modify the function in the VLM code to test other designs;
- The [Polar.py](#) provides function to read an angle of attack polar (Polar.dat) containing non-linear lift and drag coefficient. It also provides means to inform if the AoA is post stall or buffet onset;
- Call the scripts in python3.
- Beware, the code extrapolate outside of the provided polar.

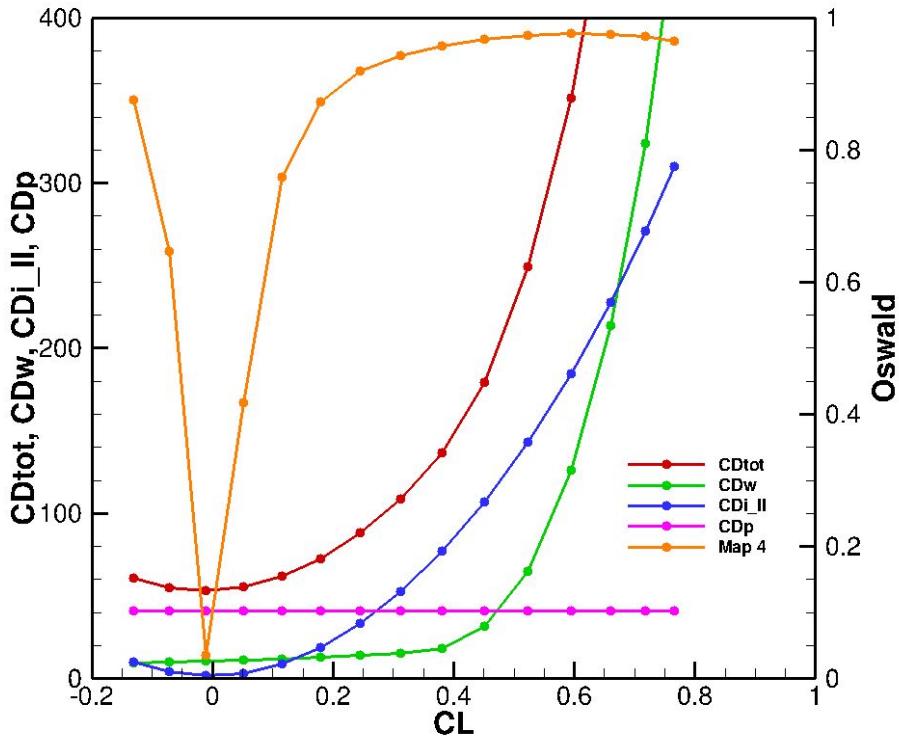
NL-VLM outputs

- 3D solution file to see the planform and vortex strength distribution (make sure the axis are proportional in tecplot visualisation);
- Spanload solution file for X-Y plot of lift and drag distribution;
- Convergence file of the NL-VLM iterations;
- force file for the complete 3D polar

Solution visualization



Drag decomposition



- Drag coefficient in count (1 count is 10 000CD);
- Oswald is the efficiency of the wing to generate lift $CD_i = CL^2 / (\pi AR e)$ with e the Oswald
- CD_p is a simple flat plate drag with a factor
- CD_w is the integration of the lift polar from Polar.dat
- DCD_{i_II} is the lifting line theory induced drag coefficient computed for the VLM spanload