UPAT ATLAS – M.ACH PROGRESS REPORT

Aerodynamics Subteam

Athena Vortex Lattice
AVL

THEODOROS ILIOPOULOS

MARCH 10, 2025

UNIVERSITY OF PATRAS

DEPARTMENT OF MECHANICAL AND AERONAUTICAL ENGINEERING

Contents

Introduction

This report aims to provide an introduction to the AVL software . Aerodynamic analysis requires tools that can quickly calculate and predict aerodynamic performance of aircraft configurations. AVL is an example of such a tool, which will be used in extent during the design methodology. This report will explore AVL, beginning with an introduction . A theoretical background of the main methods is also going to be presented. The main principles and capabilities of the software are going to be examined, along with the ways that the team can utilize it. Furthermore, specific use-cases are going to be highlighted and a user guide-session example is going to be provided. Details will be given for geometry file and the geometry file for UAV of 2024 ACC is going to be generated. For the UAV of acc2024, XFLR5 and AVL analyses are going to be run and compared, in order to derive valuable conclusions.

The report will be updated throughout the upcoming period.

Chapter 1 : Introduction to AVL

AVL (Athena Vortex Lattice) is a software developed by MIT professor Mark Drela. AVL is a command line interface (CLI) software which means that its environment is similar to command prompt. The software is used for aerodynamic analysis and flight-dynamic analysis of rigid aircraft of arbitrary configuration (arbitrary meaning it is up to the user to provide any geometry that he wants, but obviously logical choices). As the name states, AVL bases its calculation on the Vortex Lattice Method (VLM). A more detailed approach and explanation of the VLM will follow in order to gain an insight and a basic intuition into the principles and assumptions made as well as the capabilities it offers.

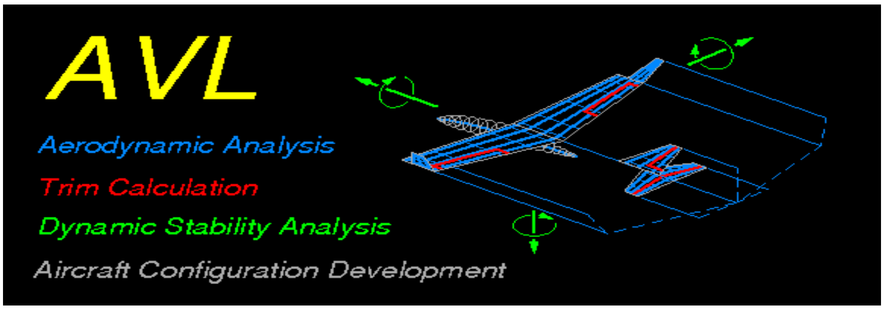

Figure 1: Capabilities of AVL

AVL is a low-fidelity aerodynamic tool that uses two types of analysis that are very useful, especially in the early stage of the design. These two types are:

- Aerodynamic analysis
- Flight Dynamic analysis/ stability

AVL can model lifting surfaces and slender bodies. That being said, AVL is used for 3D design and not airfoil design and optimization. Data derived from AVL can prove helpful for 2D design. Lifting surfaces are equivalent to any surface that provides lift and contributes to the aircraft's stability, basically the aircraft's wing and tail (vertical and horizontal). Although the tail has a much smaller contribution to the lift distribution compared to the wing, it is important to be modelled since it plays a crucial role in the aircraft's stability. Slender bodies are fuselages and nacelles. It is recommended to not model slender bodies, when their effect is small and thus can be neglected. In our case, and in the early stage of design, attention will mostly be given on the lifting surfaces and slender bodies will not be modelled until a later point of the design process . It should also be noted that AVL is not used for viscous drag calculation, thus drag will be computed in different methods (AVL can still contribute though, more on this later). Since fuselage (and potentially landing gear if not retractable) contributes to the viscous drag, it poses an extra reason for not including it into AVL calculations.

In general, although AVL is a CLI and not a GUI (graphical user interface) it proves valuable to early stage aerodynamic analysis. A similar software of the same nature is XFOIL which also (probably) will be used, mainly for 2D airfoil selection and design.

Chapter 2 : Vortex Lattice Method (VLM)

Vortex Lattice Method (VLM) is a numerical method that can be used to perform the aerodynamic analysis for 3D potential flows of lifting surfaces. A potential flow is a flow that is vorticity free (irrotational) meaning that $\nabla \times \vec{V} = 0$ and incompressible $\nabla \cdot \vec{V} = 0$. Since the velocity field is curl free then it can be written as the gradient of a potential function φ, $\vec{V} = \nabla\varphi$ (the curl of a gradient is always zero). The gradient of the potential function is substituted in the Euler equation (which is the simplified momentum equation for inviscid and incompressible flow) and the Laplace's equation is derived $\nabla^2\phi = 0$. If this Laplace equation is solved, the velocity vector field of the flow can be calculated through the potential function φ and then an aerodynamic analysis is easy to complete. The Laplace equation will be solved by using specific boundary conditions (such as tangential flow condition). So, since VLM solves potential flows and considering the nature of those problems, the main assumptions made are:

- **Quasi-steady flow**
  The conditions of the flow are considered to have a small change with time thus allowing the assumption of steadiness.
- **Irrotational /inviscid flow**
  Since VLM solves potential flows then the flow is considered to be vorticity free (irrotational $\nabla \times \vec{V} = 0$). It is important to note that this specific assumption and in general the potential flow is not true for all parts of the flow. As shown in *figure 2*, the flow is divided into the outer flow region and the boundary layer. The inviscid assumption made is true for the outer region of the flow (when Reynold's number is high, approximately greater than 10^6). The momentum equation, as derived from dimensional analysis, includes a term that corresponds to viscous effects ($\frac{1}{R_e} \cdot \nabla^2\vec{V}$). In a flow with high Re, the term can be neglected only if $\nabla^2\vec{V}$ is of order 1 (a value of the same order as other terms), which is true only for the outer flow and not for the boundary layer. This is the reason why outer flow is considered to be inviscid. Similarly, the flow is assumed to be irrotational/vorticity free, only in the outer flow. The rate of change of vorticity, when subject to dimensional analysis, has a similar term, as shown previously, which can only be neglected on the outer flow.
- **Thin lifting surfaces**
- **Small angles of attack α and small sideslip angles β.**

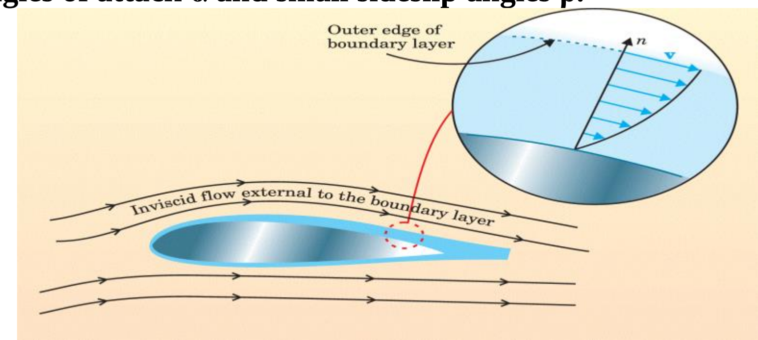

Figure 2: Outer flow and boundary layer

VLM is based on four important theories:

- Helmholtz's vortex theorems for inviscid, incompressible flow.
- Prandtl's Lifting Line Theory (LLT)
- Biot-Savart Law
- Kutta-Joukowski theorem

The solution of the Laplace equation can be obtained if a proper distribution of elementary solutions on the boundaries is found. Vortices constitute an elementary solution, and they can model lift. Helmholtz's theorems specify characteristics of vortices such as "*A vortex filament cannot start or end in a fluid (it must form a closed path or extend to infinity)",* which are used to develop the VLM method, with the particular theorem being the reasoning behind the nature of a horseshoe vortex. Biot-Savart law provides a calculation of the velocity induced by a vortex segment and thus, if integrated, for the whole vortex filament (a vortex filament is a vortex tube with infinitesimal cross section). Velocity induced by a vortex filament needs to be calculated in order to apply the boundary condition of flow tangency. LLT is the theory from which VLM originates. Horseshoe vortices and the theorems above- including the Kutta-Joukowski- were first used in LLT. Even though VLM originates from LLT, the two methods do not cover identical applications with VLM having a wider range of problems that can solve.

VLM Methodology:

Vortex Lattice Method discretizes the lifting surfaces in panels along the span and the chord. Each panel has a horseshoe vortex (as shown in figure 3). The superposition of all the horseshoe vortices of the panels will be used to calculate aerodynamic loads.

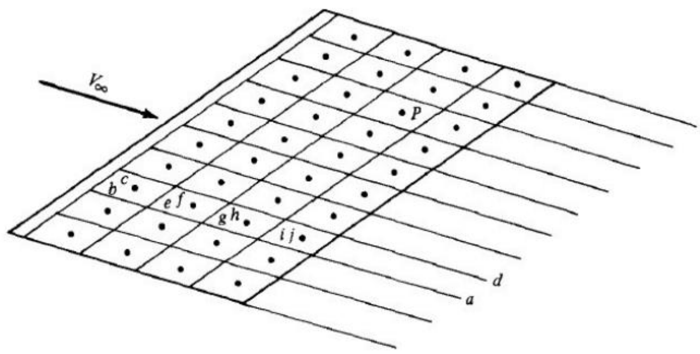

Figure 3: Discretization of a lifting surface and horseshoe vortices on each panel.

The main idea is to calculate the vortex strength distribution of the lifting surface so the aerodynamic loads can be found by the Kutta-Joukowski theorem. The vortex strength of each panel will be calculated by solving a system of equations. Each dot in *figure 3* corresponds to a control point, upon which a single equation is formed. The system is derived when the tangential flow condition (normal velocity component to be zero) is applied to each control point of a panel and when the Kutta condition is applied at the trailing edge. The normal velocity component that is induced by the horseshoe vortices is calculated with the Biot-Savart Law, as stated previously. The problem is formed in a way that the solution is relied only on the geometry. After solving the system, aerodynamic forces like lift and induced drag can be calculated, as well as moment.

In conclusion, VLM can be used to calculate the lift distribution, the induced drag and the moment of a lifting surface.

NEAR FIELD AND FAR FIELD CALCULATIONS:

AVL uses two ways to calculate aerodynamic forces. Forces can be computed either as near-field or far field. Near-field forces are forces related to the flow field that acts directly on the body surface while far-field forces are related with the flow field of a closed surface that is far from the body. The calculation of near field forces (when viscosity is neglected) is calculated by integrating the pressure components. In this case, though, induced drag is not calculated with great accuracy. Far field forces are computed via the momentum conservation which is an alternative way of calculating forces acting on the body. The far field approach is better for induced drag calculation. Trefftz plane is based on the far field approach and is also included in AVL (Trefftz plane plot).

Chapter 3 : How AVL works-Capabilities-Points of strength

AVL can be used to calculate direct aerodynamic forces and moment but can also calculate important parameters for stability analysis. In order for AVL to work and an analysis to be made the user must provide three files (2 of which are optional):

- **Geometry file**: The geometry file defines the geometry of lifting surfaces including an aircraft's wing, tail and their control surfaces. The format of the file is .avl. Geometry files are text based and can be written, edited and provided by the user. There are specific samples of the file structure that show how to properly write a geometry file to avoid any implications during the analysis. *Figure 4* shows how geometry is visualized through AVL's interface.
- **Mass file**: The mass file is optional and describes the mass and inertia properties of the aircraft configuration. A mass file is optional for an aerodynamic analysis but for a stability analysis it is mandatory. In general, it provides information for the configuration's mass, inertia, density, center of gravity (and more).
- **Run file**: This file is generated by AVL itself. It can be edited with a text editor, although this is not necessary. The parameter values in the file can be changed using AVL's menus, and the file can then be written again. A run file describes specific run cases that the aircraft will be subject to for an analysis. A run case, for example, can have specified parameters such as angle of attack, velocity, density, control surfaces angles etc.

The most important file is the geometry file and it is up to the user to create it. Through AVL the geometry can be visualized before running an analysis. In this way, a false geometry can be avoided while, also, analysis results that correspond to the exact configuration tested can be achieved.

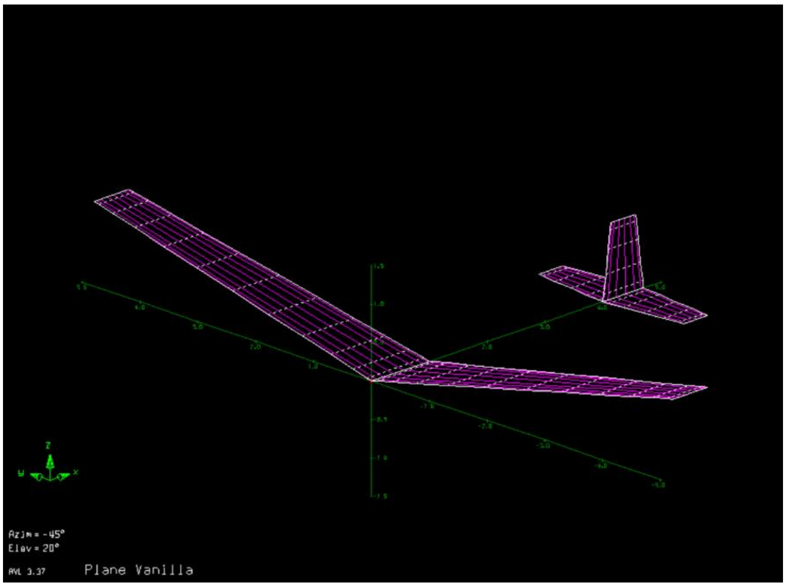

Figure 4: Geometry visualization through AVL interface.

Main capabilities of AVL:

AVL can compute lift distribution, induced drag, moment and can also plot Trefftz plane. It is important to note that the geometry file includes reference values (surface, chord, wingspan) and based on them coefficient of forces are derived. Main capabilities of AVL are:

- In the geometry file, the discretization can be chosen and visualized. Spanwise and chordwise discretization is selected by the user. Depending on the analysis made, this characteristic can prove valuable when there is specific points of interest along the surfaces.
- AVL includes many parameters that can be set to have specific values. Most important parameters are angle of attack α, sideslip β, roll rate (pb/2V), pitch rate (qc/2V), yaw rate (rb/2V), flap position, aileron position, elevator position, rudder position. Apart from setting up a run-case (i.e. takeoff), the user can also set constraints.
- Coefficients of lift, induced drag, moment are calculated for specified run-cases. Lift and induced drag are calculated with both the near-field and far-field approaches, thus there are two values computed (with slight differences). The above results can be saved in a txt file for further analysis.
- AVL has the capabilities to compute the forces either as body forces(total) or for each surface and each strip of the configuration. There are commands that can display these results. Knowing forces for each strip or surface of the configuration can be of great benefit (a method used by previous ACC teams will be reviewed later on).

- Far-field forces can be visualized via Trefftz's plane plot. Trefftz's plane has specific plots which are : (1) lift/span loading $\frac{c_l c}{c_{ref}}$ (green color) , (2) coefficient of lift along the span Cl (yellow) , (3) coefficient of lift along the span including the effect of sweep c_{l_L}(red) , (4) induced angle of attack along each section of the span (blue). The above plots are obtained for each lifting surface of the configuration (wing, tail etc.).

- Obtain derivatives that help with the stability analysis of the configuration .

- AVL is mainly used for 3D design but airfoil design is also possible. Airfoil analysis can be executed with a very high aspect ratio wing, that models an “infinite” wing.

The capabilities presented are some of the features that AVL has to offer. AVL has certain characteristics that can be very helpful for early stage design processes.

AVL’s points of Strength:

AVL is a low fidelity software, but it provides fast calculations with very low computational cost. Those characteristics are desired when a large number of analyses must be made. In the early stage of design, multiple combinations of geometries along with multiple different dimensions must be tested in order to decide on a final (candidate) design. Apart from that, an optimization of the design must be made. Optimization will also need to complete multiple analyses. Therefore, a software that can provide fast results with minimal computational cost is beneficial.

The command driven nature of AVL allows batch processing. The main difference of AVL from xlfr5 (CLI vs GUI) is the ability to integrate AVL analyses into script , automate operations and complete large batch processes at the same time. A pre-written script file that contains a specific command sequence can be used as input in order to automate a procedure. Matlab or python scripts can also integrate AVL. AVL will run iteratively in the background and data computed from the analyses will be derived and used in a desirable way, either for visualization or further analysis. An integration is valuable and almost necessary since manual usage can be time-consuming.

Another strong point of AVL is that it is highly editable. Geometry, mass and run files are txt based and can be edited at any moment (and fast) to create a different configuration or analyze a different run-case.

The characteristics noted above are some of the reasons why AVL is a suitable choice to work with .

Chapter 4 : AVL proven history

AVL has been used by two teams in the previous ACC competition of 2024. The two teams were Akamodell Stuttgart (winners of the competition) and Hermes NTUA. Both teams run AVL analyses for a large number of configurations since they followed a rigorous exploration of the optimal aircraft design. Calculations were made for lift distribution , induced drag as well as stability analysis and derivative computation (procedures done for wing and tail). Apart from that, particular flight states like takeoff where analyzed. AVL was heavily used to find the preferred performance of the aircraft. It should be noted that, the capabilities of mass estimation were used to assist with the structural design of the aircraft.

One interesting application of AVL was the calculation of viscous drag. The software does not calculate viscous drag in any way possible but can only include it in the forces, if it is manually input as a pre-specified value. Akamodell Stuttgart used coefficient of lift C_l values calculated at specific strips of the wing and imported them to xfoil in order to compute the local viscous drag. Summing up the results of all the ‘strips’ used, the total viscous drag of the wing is obtained. Team Hermes from NTUA used a similar method to compute the viscous drag.

Chapter 5 : User guide-Session examples

Download AVL and the source code in order to run the executable. The pop-up interface of AVL is shown on *figure 5.*

```
===================================================
 Athena Vortex Lattice  Program      Version  3.37
 Copyright (C) 2002   Mark Drela, Harold Youngren

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

 Caveat computor
===================================================

==========================================================
  Quit    Exit program

 .OPER    Compute operating-point run cases
 .MODE    Eigenvalue analysis of run cases
 .TIME    Time-domain calculations

  LOAD f  Read configuration input file
  MASS f  Read mass distribution file
  CASE f  Read run case file

  CINI    Clear and initialize run cases
  MSET i  Apply mass file data to stored run case(s)

 .PLOP    Plotting options
  NAME s  Specify new configuration name

AVL   c>  
```

Figure 5: AVL interface

Command LOAD followed by the geometry file name , will load the configuration. The command OPER gets the user to the main operating menu. As soon as OPER command is executed, the text window refreshes and the commands of the operating menu can be seen *(figure 6).*

```
Operation of run case 1/1:   -unnamed-
==========================================================

 variable          constraint
 ------------      ------------------------
 A lpha       ->  alpha       =   0.000
 B eta        ->  beta        =   0.000
 R oll  rate  ->  pb/2V       =   0.000
 P itch rate  ->  qc/2V       =   0.000
 Y aw   rate  ->  rb/2V       =   0.000
 D1  flap     ->  flap        =   0.000
 D2  aileron  ->  aileron     =   0.000
 D3  elevator ->  elevator    =   0.000
 D4  rudder   ->  rudder      =   0.000
 ------------      ------------------------

 C1  set level or banked  horizontal flight constraints
 C2  set steady pitch rate (looping) flight constraints
 M odify parameters

"#" select  run case           L ist defined run cases
 +  add new run case           S ave run cases to file
 -  delete  run case           F etch run cases from file
 N ame current run case        W rite forces to file

eX ecute run case              I nitialize variables

 G eometry plot                T refftz Plane plot

 ST  stability derivatives     FT  total   forces
 SB  body-axis derivatives     FN  surface forces
 RE  reference quantities      FS  strip   forces
 DE  design changes            FE  element forces
 O ptions                      FB  body forces
                               HM  hinge moments
                               VM  strip shear,moment

.OPER (case 1/1)   c>
```

Figure 6: Main operating menu

The parameter variables can be seen (angle of attack ,sideslip, roll rate etc) and their values can be set as constraints. By pressing the equivalent letter shown next to each parameter , the user can choose to set a specific constraint. For example, user can press D1 to change the flap angle. After pressing D1, the angle value will be the next input. The parameter variable table will refresh with the newly input values(*figure 7)*.

```
Operation of run case 1/1:   -unnamed-
==========================================================

 variable          constraint
 ------------      ------------------------
 A lpha       ->  alpha       =   0.000
 B eta        ->  beta        =   0.000
 R oll  rate  ->  pb/2V       =   0.000
 P itch rate  ->  qc/2V       =   0.000
 Y aw   rate  ->  rb/2V       =   0.000
 D1  flap     ->  flap        =   5.000
 D2  aileron  ->  aileron     =   0.000
 D3  elevator ->  elevator    =   0.000
 D4  rudder   ->  rudder      =   0.000
 ------------      ------------------------
```

Figure 7:Refreshed parameter table with flap angle 5° instead of 0°

It is important to note that specific parameters can be constrained indirectly by another variable. For example, angle of attack can be constrained by choosing a specific value for the coefficient of lift (C_L). Commands C1 and C2 (as seen in figure 6) allow the user to modify parameters in order to create a specific trimmed run-case. The term trim means that the aircraft will be in equilibrium. When C1 or C2 is chosen, specific constraints are input by the user (i.e. angle of attack, C_L), and AVL calculates the remaining parameters to achieve equilibrium.

A run-case is an analysis of a specific flight-state that the user can fully edit either through the parameter variable table or the C1/C2 commands. To execute a run-case the command X must be input. After the execution , results can be obtained and saved in txt files. In figure 6, commands ST,SB,FT,FN,FS,FB,HM,VM correspond to computed values. When these commands are used , the user will be prompted to name a file where those values will be saved. The file will be txt based and will be saved in the same directory as the AVL executable.

One of the most useful commands is the G (Geometry plot) and T (Trefftz plane plot). The G command , plots the geometry of the configuration and has its own sub-menu, as seen in *figure 8 ,9 and 10*. The sub-menu of G command offers many capabilities that can be helpful and insightful.

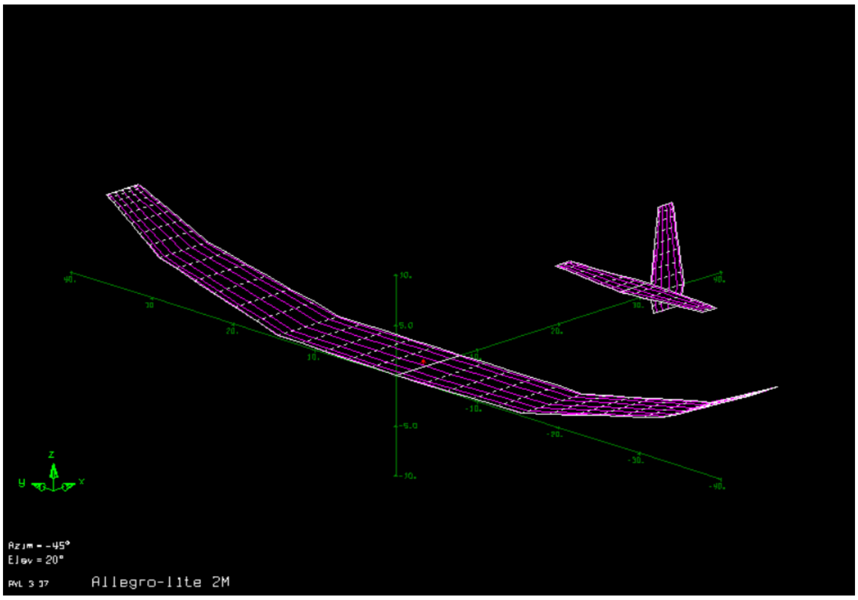

Figure 8: Geometry of allegro aircraft.

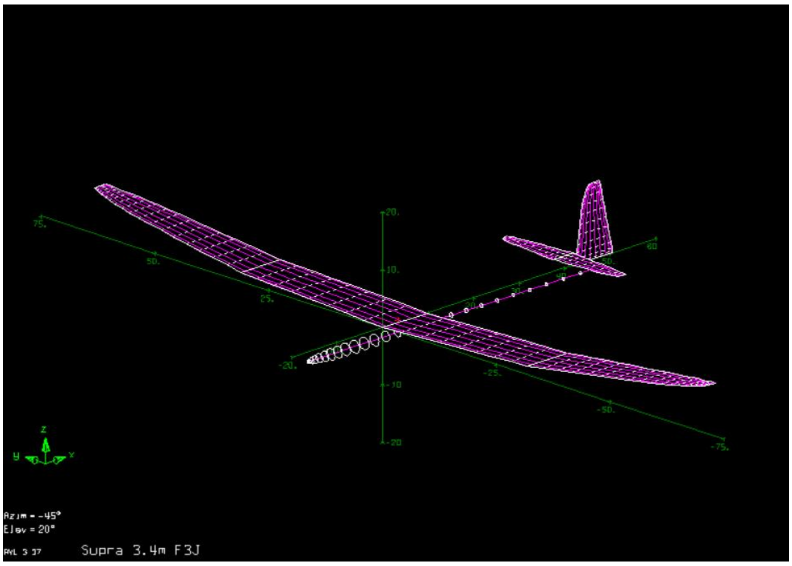

Figure 9:Geometry of supra aircraft .

```
.OPER (case 1/1)   c>  g

X-window size changed to  11.00" x -0.30"

=========================================
 K eystroke mode        V iewpoint
 A nnotate plot         O ptions
 H ardcopy plot         S elect surfaces
 Z oom                  U nzoom

 CH ordline    T        CA amber        F
 CN tlpoint    F        TR ailing legs  F
 BO ound leg   T        NO rmal vector  F
 LO ading      F        AX es, xyz ref. T

Geometry plot command: _
```

Figure 10:Sub-menu of geometry command.

K => Keystroke mode : When pressed ,the user can perform an interactive rotation of the aircraft and select preferred viewpoints.

CH,CA,TR,NO,LO are all commands that can be visualized in the initial geometry configuration(*figures 11,12,13 show NO,LO and TR).*

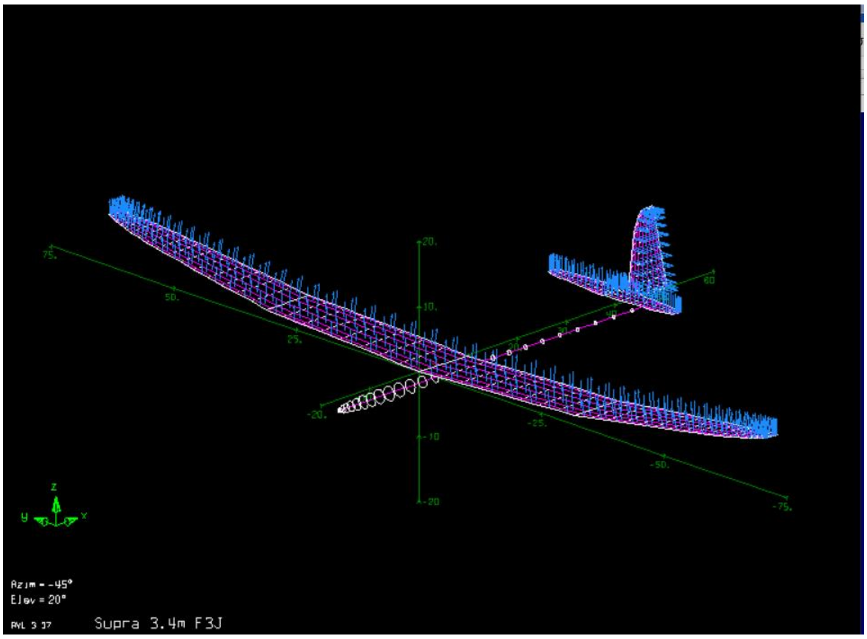

Figure 11:NO command, normal vector of each surface.

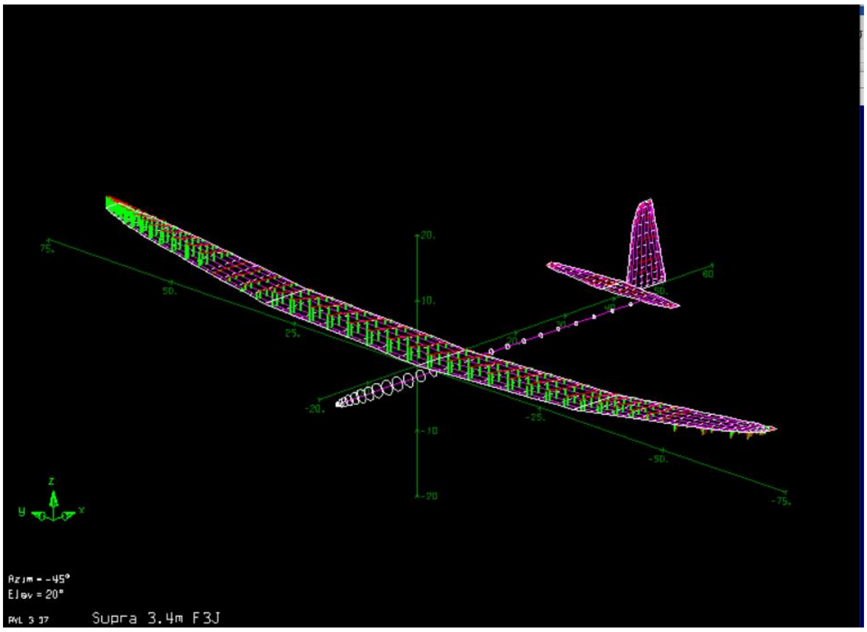

Figure 12: LO command, loading of the surface depending on the run-case.

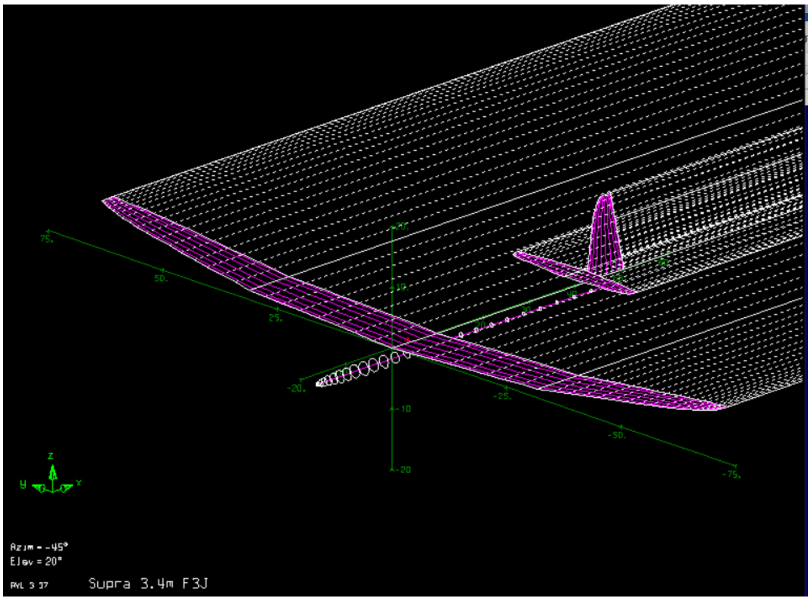

Figure 13:Trailing wake of the lifting surfaces.

The Trefftz plane plot is important because it provides specific plots, as mentioned in unit 3. In *figure 14*,a Trefftz plane plot for a simple run-case is visualized.

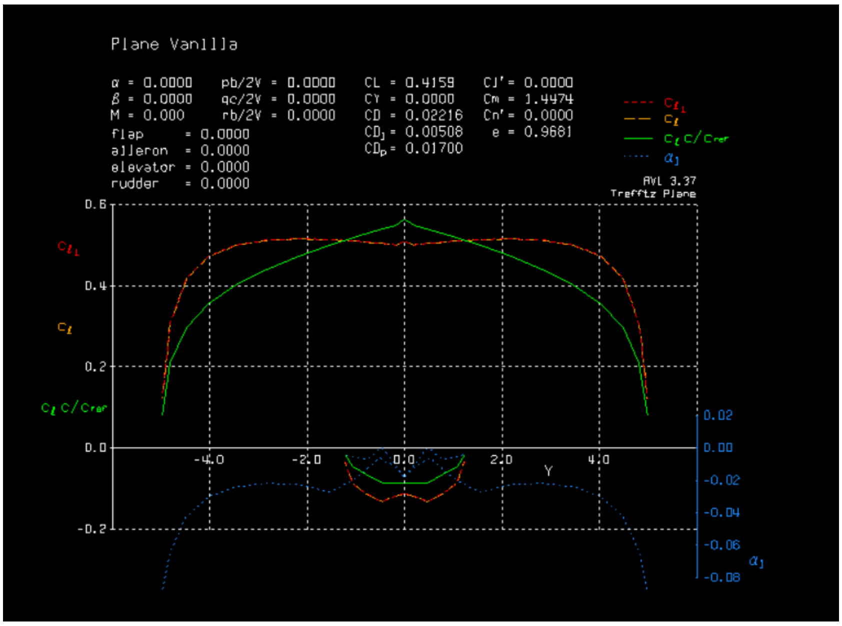

Figure 14:Trefftz plane for vanilla aircraft.

- Green plot corresponds to lift/span loading $\frac{c_l c}{c_{ref}}$
- Red plot corresponds to coefficient of lift along the span including the effect of sweep c_{l_L} (Velocity used in Kutta-Joukowski is V_T=V·cos(sweep))
- Yellow plot corresponds to coefficient of lift along the span without the effect of sweep c_{l_L} (Velocity used in Kutta-Joukowski V=$V_{freastream}$)
- Blue plot corresponds to the induced angle α_i.

It is worth noting that the Trefftz plane includes all lifting surfaces and that is why plots that represent the tail's behavior can be seen.

The visualizations presented above were taken from the AVL editor with screenshots for timesaving. However, they can be obtained outside the AVL interface and saved as postscript file. This can be done with the H (hardcopy) command, which saves the content to ps file. Later on , with the use of a third party software or web tool, the postscript file can be converted to any file format is convenient. In *figure 15* an aircraft configuration is presented in the form of png (converted form).

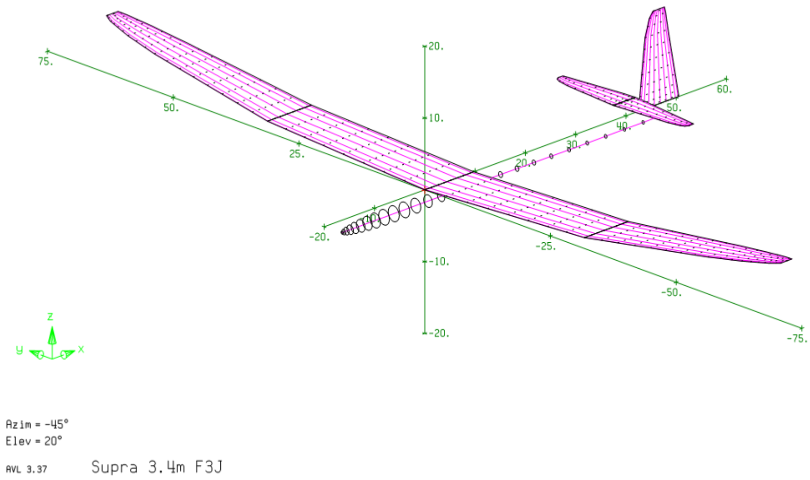

Figure 15: Converted file

Along with the AVL executable, there are many other files downloaded. Those files include a number of aircraft configurations as well as wing that can be tested to gain further experience with the software. Two session examples are also included and they provide command sequences to model an analysis.

Chapter 6: Geometry file details

The geometry file is the most important and needs to be defined properly. If not defined correctly , the analysis will not have the desirable results. The utter goal is for AVL to run automatically. For this to happen the geometry file must be carefully created , which means the structure and parameters should be examined in detail. However, whether AVL runs manually or automatically , a geometry file holds the same value. The structure of the file should be specific and for that reason sample files are provided. The geometry file contains information for all the characteristics mentioned below:

- Mach number (can also manually be changed for analysis).
- Reference values , Sref, Cref, Bref needed for calculations.
- Profile drag (manual input ,as previously said it cannot be calculated).
- Surface definition-Surface discretization into panels.
- Airfoil selection.
- Incidence angle of lifting surface.
- Section definition-Division of surfaces to sections with their own characteristics.
- Sweep, twist, dihedral, chord, wingspan are input via the coordinates provided.
- Control surface definition.
- Design parameters that change values of variable for quick analysis at runtime.
- Body definition -Fuselage discretization.
- Reference points on the coordinate system(usually correspond to cg coordinates).

In order for a geometry file to be created, the above characteristics must be known. Attention to detail should be given, since a small mistake can lead to heavily inaccurate analysis outcomes. In *figure 16* a sample structure of an aircraft configuration is provided. The specific aircraft configuration is part of the pre-designed aircrafts that are included along with the software’s downloaded source files. Each term will be analyzed and certain details will be pointed out. It should be noted that there have been small modifications to reduce the size of the sample. The sample is given in order to understand better the structure of the geometry file , that being said, in some cases, the structure can take different forms apart from the one shown.

```
Plane Vanilla
0.0   # Mach
0 0 0.0 # IYsym IZsym Zsym
9.0 0.9 10.0 # Sref Cref Bref
0.50 0.0 0.0 # Xref Yref Zref

SURFACE
Wing
8 1.0 12 1.0 # Nchordwise Cspace Nspanwise Sspace
0.0 # YDUPLICATE
2.0 # ANGLE
SECTION
0.0 0.0 0.0 1.0 0.0 0 0 # Xle Yle Zle Chord Ainc Nspanwise Sspace
AFILE
sd7037.dat
CONTROL
flap 1.0 0.75 0.0 0.0 0.0 1.0
CONTROL
aileron -1.0 0.75 0.0 0.0 0.0 -1.0
CLAF
1.0
SECTION
0.2 5.0 1.0 0.6 0.0 0 0 # Xle Yle Zle Chord Ainc Nspanwise Sspace
AFILE
sd7037.dat
CONTROL
flap 1.0 0.75 0.0 0.0 0.0 1.0
CONTROL
aileron -1.0 0.75 0.0 0.0 0.0 -1.0
CLAF
1.0

SURFACE
H-stab
6 1.0 6 1.0 # Nchordwise Cspace Nspanwise Sspace
0.0 # YDUPLICATE
4.0 0.0 0.0 # TRANSLATE
SECTION
0.0 0.0 0.0 0.7 0.0 0 0 # Xle Yle Zle Chord Ainc Nspanwise Sspace
CONTROL
elevator 1.0 0.7 0.0 1.0 0.0 1.0
SECTION
0.14 1.25 0.0 0.42 0.0 0 0 # Xle Yle Zle Chord Ainc Nspanwise Sspace
CONTROL
elevator 1.0 0.7 0.0 1.0 0.0 1.0

SURFACE
V-stab
```

```
6 1.0 5 1.0 # Nchordwise Cspace Nspanwise Sspace
4.0 0.0 0.0 # TRANSLATE
SECTION
0.0 0.0 0.0 0.7 0.0 0 0 # Xle Yle Zle Chord Ainc Nspanwise Sspace
CONTROL
rudder 1.0 0.5 0.0 0.0 1.0 1.0
SECTION
0.14 0.0 1.0 0.42 0.0 0 0 # Xle Yle Zle Chord Ainc Nspanwise Sspace
CONTROL
rudder 1.0 0.5 0.0 0.0 1.0 1.0
```

Figure 17: Sample geometry file

HEADER:

At the beginning of the file there shall always be a header. The header includes the following(*table 1* has details about the parameters used):

- 1st line: Name of the configuration. (Attention: the name used to load the file is the name of the file ,not the name provided in the header unless they are the same).
- 2nd line: Mach number is provided . For low speed analyses is preferred to input Mach number equal to zero.
- 3rd line: Provide the axis of symmetry of the configuration.
- 4th line: Provide the reference values for the configuration.
- 5th line: Provide the reference point of the configuration. The reference point is the location about which moments and rotation rates are calculated. Usually it represents the center of gravity.

Details about header parameters	
IYsym	Symmetry about the X-Z plane. Can take the value of 1.0 for symmetry, 0.0 for no symmetry and -1.0 for anti-symmetric configuration. Symmetry about the X-Z plane is always around y=0 point. If IYsym and IZsym are used then aerodynamic analysis is made for half of the configuration. This means that it can only be used when the user is certain the forces are also symmetric/antisymmetric. Not very helpful for most cases.
IZsym	Symmetry about the X-Y plane. Takes the same values as IYsym. The difference is that the point of symmetry is now provided by the user. Zsym corresponds to that point.
Zsym	Point of symmetry about the X-Y plane.
Sref	Corresponds to the reference area of the wing. Based on this value, the aerodynamic forces are non-dimensionalized and coefficients are derived. Because the main lifting surface is the wing , Sref Cref Bref all correspond to the main wing.
Cref	Same purpose as Sref, corresponds to the reference chord length. The MAC is usually used as the Cref. Specifically used for pitching moment.

Bref	Same purpose as Sref, corresponds to the reference wingspan. Important for roll and yaw moments.
X,Y,Z ref	It is the point about which the moment and rotation rates are defined. It is recommended to correspond to the configuration's center of gravity.
Optional parameters	
CDp	Manually input the profile drag of the configuration and it will be included to the calculations. If the user does not want to include profile drag, then the parameter can be ignored.
COMPONENT	The 'COMPONENT' keyword is used to group surfaces into one "virtual" surface. For example it can be used in a T-tail in order to group the horizontal and vertical tail. By not grouping these kinds of geometries, errors will occur .These errors relate with the location and interaction of vortices and control points.

Table 1: Parameters used in the header.

LIFTING SURFACES:

After the header the surfaces are defined. Every lifting surface of the configuration is modeled as a surface (wing, tail). Steps in defining the surfaces are explained below in order, as seen in the sample structure.

1. Start with the command SURFACE to state the beginning of a surface definition.
2. State the name of the surface.
3. Input the discretization of the surface. The surface will have a specific amount of horseshoe vortices and their distribution is chosen by the user (Nchordwise Cspace Nspanwise Sspace).
4. Keyword ANGLE will be used to set an angle of incidence for the whole surface. This command can be skipped if incidence angle is zero.
5. Keyword SECTION defines a particular section of the surface. Every surface must have two sections. In the sections, important dimensions in the form of coordinates are input, while also section discretization can be an optional addition (Xle Yle Zle Chord Ainc Nspanwise Sspace).
6. Keyword AFILE is used to input the airfoil of the section and is followed by the airfoil file (i.e. naca0012.dat).
7. All sections of a surface are defined one after the other .Each section requires the steps 5 and 6 in order to be defined successfully.

In *table 2* details about the surface parameters are given. Notes for potential errors and potential use cases are given in *table 3.* Analysis results heavily depend on the surface parameters thus more attention is needed. The keyword sequence presented above needs to be completed for any surface definition. That means wing, vertical tail and horizontal tail have the same sequence format.

Details about Surface parameters	
SURFACE	Always start the definition with the SURFACE keyword.
Nchordwise	The amount of horseshoe vortices in the chordwise direction.
Cspace	Defines the spacing between the vortices placed chordwise. The values given are ±3.0 , ±2.0 , ±1.0 and 0.0 . Each of these values represent a particular distribution. Discretization will be further analyzed later on.
Nspanwise	The amount of horseshoe vortices placed in the spanwise direction.
Sspace	Defines the spacing between the vortices places spanwise. Everything that is true about Cspace are also true for Sspace.
ANGLE	Keyword used to define the incidence angle for the whole surface. It is followed by the angle value in degrees. It is best suited to define incidence angle while Ainc is better for twist definition.
SECTION	To define a surface at least two sections are needed. The two sections correspond to the definition of the root ant tip. Any change in airfoil, taper, twist, dihedral will require more section definitions (i.e. sweep starting at 30% of wing requires more than 2 sections).
TRANSLATE	TRANSLATE keyword should be used to move the whole surface to a specific point in space (i.e. wing is places 0.5m behind the fuselage nose). It basically moves the Xle,Yle,Zle of all surface sections. (Optional)
SCALE	Scale the geometry by a factor applied in all x,y,z coordinates. Must be used before TRANSLATE.
Xle,Yle,Zle	The coordinates of the airfoil's leading edge.
Chord	Defines the chord of the specific section. If a wing only has two sections then by defining root and tip chord, taper ratio is indirectly defined too.
Ainc	Defines the incidence angle of the specific section only. Twist can be defined with this keyword. If you use this keyword, it is better to not use the ANGLE parameter unless they define different things.
Nspan Sspace	These two keywords can be used ONLY if the discretization is not already defined. Usually they can be ignored
AFILE	In order to define a section's airfoil, keyword AFILE is used. The keyword is followed by the airfoil .dat file. The airfoil file must be in the same directory as the executable otherwise the directory should be provided.
YDuplicate	Y position of X-Z plane about which the current surface is reflected to make the duplicate geometric-image surface.
COMPONENT	Used when a wing or tail consists of more than one surface(i.e. winglets).

Table 2:Details about surface parameters.

Notes	
Note 1	Leading edge coordinates are used to define correctly the geometry of each surface. From these coordinates, twist , dihedral, sweep can indirectly be defined.
Note 2	Sections should be defined from root to chord in order for the analyses results to be correct. For the vertical tail, it is preferred to have an order from top to bottom, although it is not absolutely necessary
Note 3	Yduplicate and IYsym must not be used at the same time because an error will occur. Yduplicate is preferred.
Note 4	If Yduplicate is used then the values provided in Xle,Yle,Zle should be half (i.e. Yle should be half the span since the section is symmetrical).
Note 5	Airfoil file must have correct values. An incorrect .dat file will cause unnatural analysis results. The “SINVRT: spline inversion failed” message during the loading of the geometry file, indicates a problem with the values provided in the airfoil file.
Note 6	Values that vary along the surface will be linearly interpolated (i.e. incidence angle and chord). That is one of the reasons two sections are needed. Two known values and between them there is linear variation.

Table 3: Notes and tips for surface parameters usage.

CONTROL SURFACES:

To define control surfaces either these are ailerons, rudder or elevators the following keyword sequence must be completed. Flaps and slats can also be defined in the same way.

1. Enter keyword CONTROL
2. Follow the CONTROL keyword with the name of the control surface(i.e. aileron).
3. Input the gain value , the Xhinge value ,the XYZhvec value and the SgnDup value.
4. Follow the same procedure for every control surface definition.

Table 4 includes details for the keywords used and *table 5* has notes that help with the completion of a correct definition.

Details about Control Surface parameters	
CONTROL	Keyword used to initialize control surface definition
Gain	Control deflection gain, units: degrees
Xhinge	It is the x/c location of the hinge. If it is positive then the movement will extend to the trailing edge and if it is negative it will extend to the leading edge.
XYZhvec	It represents the vector of the hinge axis about which the surface rotates.
SgnDup	If the surface is duplicated, then it shows the sign of deflection for the duplicated surface. For example elevators must have the same deflection that's why the value is +1.0 .However, ailerons have opposite deflection and the value is set to -1.0.
CLAF	It is an optional parameter that scales the derivative of lift coefficient in terms of AoA (dCl/dα).

Table 4:Details for control surfaces definition.

Notes	
Note 1	Control surface should be defined in every section of the surface that it exists. For example if the aileron is placed near the end of the wing, then after the tip section definition the control surface definition must follow. If part of the control surface existed in another section , then the control surface would also be defined again in that surface.
Note 2	Gain can be different for each section of the control surface. It will be linearly interpolated.
Note 3	A different SgnDup can model non-symmetric control effects,. However, such thing is a bit off scope.
Note 4	The CLAF keyword adds a scaling factor in order to include a more precise modeling of the thick airfoils. If not included then the slope will be equal to 2π as found in thin airfoil theory. Should be used when the airfoils are thick.

Table 5:Notes on control surfaces definition.

Discretization of the lifting surfaces:

When the lifting surfaces are defined, the discretization of them is also chosen. Discretization is critical and special care should be given , in order to achieve good analyses results. In this part of the report, some advices are going to be provided. These advices will help the user take the proper decisions according to his purpose.

- What is the number of horseshoe vortices that should be used to discretize the surface chordwise and spanwise?

In general, the amount of vortices should be larger for spanwise direction that chordwise direction. Depending on the analysis accuracy and the analysis run-time there are two ranges of vortices:

Standard analysis that provides good results and sufficient accuracy:

Nchordwise ➔ 8-12 horseshoe vortices

Nspanwise ➔ 10-20 horseshoe vortices

Best analysis that provides the most accurate results AVL can give:

Nchordwise ➔ 12-15 horseshoe vortices

Nspanwise ➔ 20-30 horseshoe vortices

The amount of horseshoe vortices depends on the use-case of the geometry file.

- What should the distribution of the vortices be along the span and the chord?

Figure 18 presents the scenarios for the distribution spacing of the vortices. Parameters Sspace and Cspace define the distribution spanwise and chordwise and their values should be chosen accordingly.

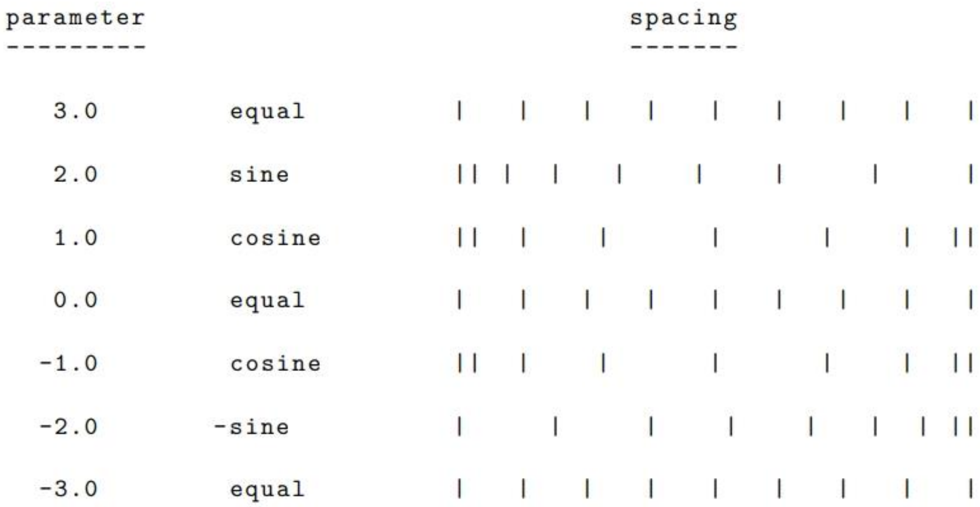

Figure 18:Potential distributions of vortices.

Some distributions give better results depending on the geometry of the configuration. The most efficient distribution in most cases is the cosine (1.0) spanwise and chordwise. The -sine distribution is also very efficient especially for wings that are NOT elliptical, straight or slightly tapered. The most important thing to understand is that the distribution should be denser in areas that rapid changes occur in the flow. Such areas are usually the leading and trailing edges as well as the points of maximum camber, wingtips , control surfaces breaks and taper breaks (the last two define where the control surfaces and the wing's taper starts or ends). As seen in *figure 18* cosine and sine distribution meet this criterion and that's one of the main reasons for their frequent usage.

When using VLM, and according to the official AVL user guide, there are certain rules that must be taken into consideration:

- "*A trailing vortex leg must not pass close to a downstream control point , otherwise the solution will be incorrect"*. This means that if surfaces align in the x axis, then they must have the same spanwise vortex spacing.
- In general, vortex spacing in the spanwise direction should be smooth and not have any sudden changes. More vortices (Vortex bunching) should be applied at wingtips or control surface ends for better results.
- Cosine spacing is usually the best for control surfaces.
- If vortex spacing needs to be refined then refinement should take place in both spanwise and chordwise direction.

In the next chapters ,the geometry file of the previous ATLAS-M.ACH aircraft is going to be created. After the geometry visualization, an aerodynamic analysis is going to be performed. The results are going to be compared and verified with the XFLR5 software.

Chapter 7: ATLAS geometry file- Creation of geometry in both AVL and XFLR5

The geometry file is going to be created based on the values presented in *table 6,7 and 8* where all the needed coordinates and values are provided. The structure is going to be derived by the examination made in chapter 6 , while also taking into account the notes that were previously presented.

Main wing			
Airfoil	NACA 6412	**Angle of incidence (°)**	2.1
Aerodynamic twist (°)	-2	**Wingspan b (m)**	2.98
Surface Area (m^2)	0.48	**Dihedral**	0
Aspect Ratio	18.48	**Sweep at 25% (°)**	≈0
MAC (m)	0.167	**Flaperon start point (x/c)**	55%
Root chord (m)	0.215	**Flaperon end point (x/c)**	90%
Tip chord (m)	0.105	**Flaperon length (x/c)**	20%
Taper Ratio	0.5	**Root twist (°)**	-2°

Table 6: Main wing properties

Horizontal Stabilizer			
Airfoil	NACA 0008	**Angle of incidence (°)**	0
Aerodynamic twist (°)	0	**Wingspan b (m)**	0.377
Surface Area (m^2)	0.04	**Dihedral**	0
Root chord (m)	0.138	**Taper ratio**	0.54
Tip chord (m)	0.075	**Aspect Ratio**	3.53
Distance of horizontal tail LE from wing LE at x axis (m)	0.986	**Distance of horizontal tail LE from wing LE at z axis (m)**	0.373

Table 7: Horizontal stabilizer properties.

Vertical Stabilizer			
Airfoil	NACA 0008	**Wingspan b (m)**	0.373
Surface Area (m^2)	0.064	**Dihedral**	0
Root chord (m)	0.215	**Taper ratio**	0.6
Tip chord (m)	0.129	**Aspect Ratio**	2.17
Distance of vertical tail LE from wing LE at x axis (m)	0.906	**Distance of vertical tail LE from wing LE at z axis (m)**	0

Table 8:Vertical Stabilizer properties.

Based on these properties, the geometry file is created as seen in *figure 20.* The geometry file includes the surfaces of the wing and the vertical and horizontal tail. As mentioned above, the T-tail has been defined as one component for better analyses results. ANGLE was used to define the incidence angle of the wing while Ainc was used to individually define twist in root and tip. A linear interpolation happens for the rest of the surface. In *figure 19* ,the geometry is visualized via AVL's environment.

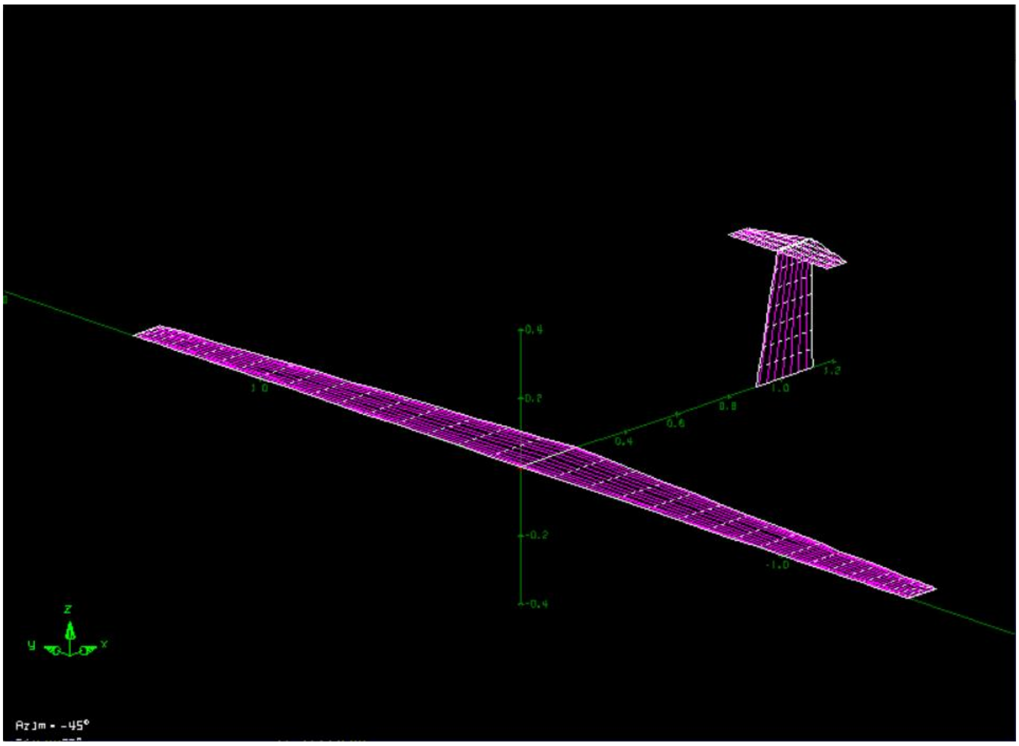

Figure 19: ATLAS 2024 configuration

```
ACC_2024_ATLAS
#Aircraft Name

#Mach
0.0

#IYsym   IZsym  Zsym
  0        0     0

#Sref    Cref     Bref
0.48     0.167    2.98

#Xref    Yref     Zref
0.0      0.0      0.0

#=======================================
SURFACE
Wing_geom

#Nchordwise   Cspace  Nspanwise   Sspace
13              1.0      19       -2.0

#Mirror geometry
YDUPLICATE
0.0

#Surface angle of incidence
ANGLE
2.1
#-----------------------------------------
SECTION
#Xle  Yle  Zle  Chord   Ainc  Nspanwise   Sspace
0.0   0.0  0.0  0.215    2.0     0          0

#Airfoil
AFILE
naca6412.dat
#-----------------------------------------
```

```
SECTION
#Xle  Yle   Zle  Chord   Ainc  Nspanwise   Sspace
0.0   1.49  0.0  0.1075   0.0     0          0

#Airfoil
AFILE
naca6412.dat
#-----------------------------------------
SURFACE
Horizontal tail

#Nchordwise   Cspace  Nspanwise   Sspace
7             0.0      19         0.0

#Component definition
COMPONENT
1
#Mirror H tail
YDUPLICATE
0.0

#Surface angle
ANGLE
0.0
#-----------------------------------------
SECTION
#Xle    Yle   Zle    Chord  Ainc  Nspanwise   Sspace
0.986   0.0   0.373  0.138  0.0     0           0

#Airfoil
AFILE
naca0008.dat
#-----------------------------------------
SECTION
#Xle    Yle       Zle    Chord    Ainc  Nspanwise   Sspace
0.986   0.1885    0.373  0.07452  0.0     0           0

#Airfoil
```

```
AFILE
naca0008.dat
#-----------------------------------------
SURFACE
vertical_tail

#Nchordwise   Cspace  Nspanwise   Sspace
7               0.0      7          0.0

#Component definition
COMPONENT
1

#Surface angle
ANGLE
0.0
#-----------------------------------------
SECTION
#Xle    Yle   Zle    Chord  Ainc  Nspanwise   Sspace
0.986   0.0   0.377  0.129  0.0     0           0

#Airfoil
AFILE
naca0008.dat
#-----------------------------------------
SECTION
#Xle    Yle   Zle    Chord   Ainc  Nspanwise   Sspace
0.906   0.0   0.0    0.215    0.0     0           0

#Airfoil
AFILE
naca0008.dat
#-----------------------------------------
```

Figure 20: Geometry file for ATLAS 2024 UAV

The configuration can easily be created in XFLR5 through its graphic user interface, with no need of any external files. In *figure 21* and *figure 22* the configuration of the lifting surfaces can be seen.

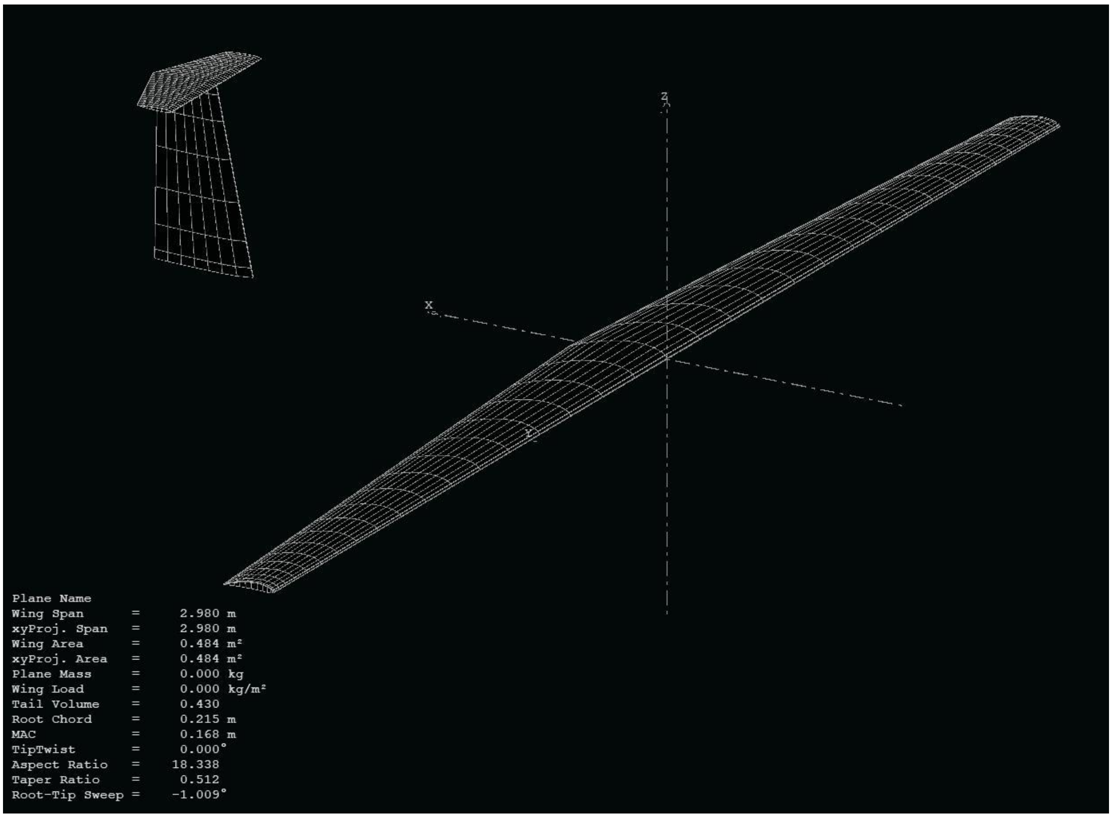

Figure 21: XFL5 meshed configuration

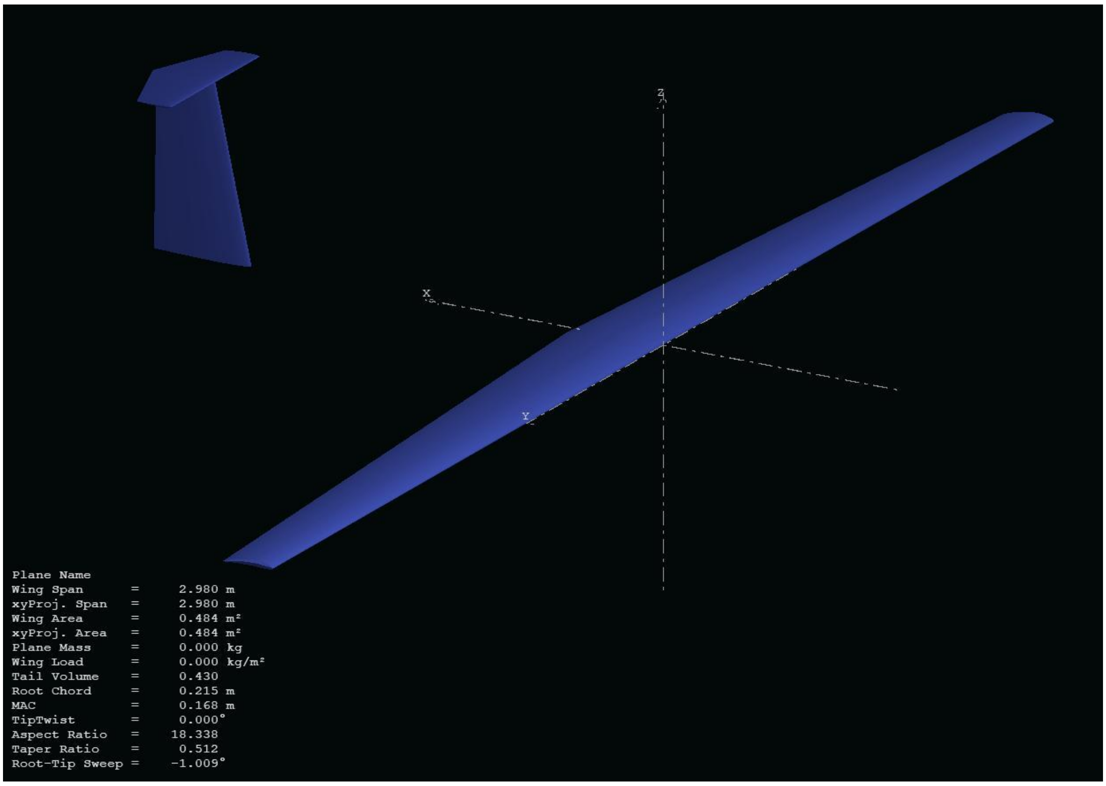

Figure 22: XFLR5 3D configuration

Chapter 8: AVL and XFLR5 analyses-Comparison of results

As soon as the configuration was created in both software, some draft analyses were made. These analyses were more aerodynamic oriented than stability, thus CL was the main quantity taken into consideration. In first glance, the values diverged at a rate of ≈50%. The divergence were significant which led to the questioning of the geometry file and the analysis procedure. After some "debugging" the root of the problem was found and fixed. The airfoil file (.dat) had a mismatch of values in the trailing edge which caused a problem in AVL analysis. Even though the error was frustrating, there was an important outcome to consider. AVL displays an irregular sensitivity on airfoil data , in comparison to XFLR5 that is not severely impacted. The root cause of the problem lies in the different methods of analyzing the .dat file and is not a main concern. What should be understood is the fact that .dat files should be cross-checked before used to avoid false analyses.

The analyses where run again and this time results displayed a smaller deviation but still concerning(≈9%). After some iterations and changes in the analyses procedure, on both AVL and XFLR5, a conclusion was derived. AVL and XFLR5 process camber and thickness differently. This springs from the different methods that are used. XFLR5 appears to have more accuracy , possibly due to xfoil's integration. The conclusion was first derived when alternating from symmetrical and non symmetrical airfoils, where symmetrical airfoils showed very little to none deviation. It is important to note that the deviation does not depend solely on AVL but also on XFLR5. For that reason, airfoils and 3D lifting surfaces were refined with more panels for better accuracy. The refined XFLR5 geometry was compared with AVL results , as previously done, and deviation appeared decreased by ≈30%.

In order to cross check the reason behind the deviation, a few analyses were run to understand how camber and thickness affected (individually) the results. During this process, it was highlighted that AVL analyses have a larger effective angle of attack compared to XFLR5. For example, the CL value of XFLR5 at 0° is equal(≤1% deviation) with AVL's CL value for -0.25°. The remarks described were examined for the airfoils:

- NACA 2421-NACA 2412-NACA 2408 to showcase how thickness affects the results (*figure 23).*
- NACA 6412-NACA 4412-NACA 2412 to showcase how camber affects the results (*figure 24).*

The figures that were derived show that indeed camber and thickness , both individually and combined, affect the deviation %. In general, from the comparison made the following outcomes were noted:

- AVL tends to overestimate aerodynamic forces, especially in smaller angles of attack. This is partially due to how camber and thickness is modelled.
- The overestimation matches values that have increased angles of attack by 0.15°-0.5°, depending on the thickness and camber. For example the values calculated with XFLR5 for AoA 0° correspond to the values of AVL for -0.2°. The larger the angle of attack is , the bigger the angle of overestimation is.

- For increased camber , the overestimation angles are greater.
- Symmetrical airfoils have almost zero deviation.
- Induced angle differs between the two software and requires further examination.
- In higher angles of attack the deviation of aerodynamic forces is decreased. However, the overestimation angle is slightly increasing.
- The same remarks can be made for induced drag. Deviation can reach ≈10% for geometries with high camber airfoils.

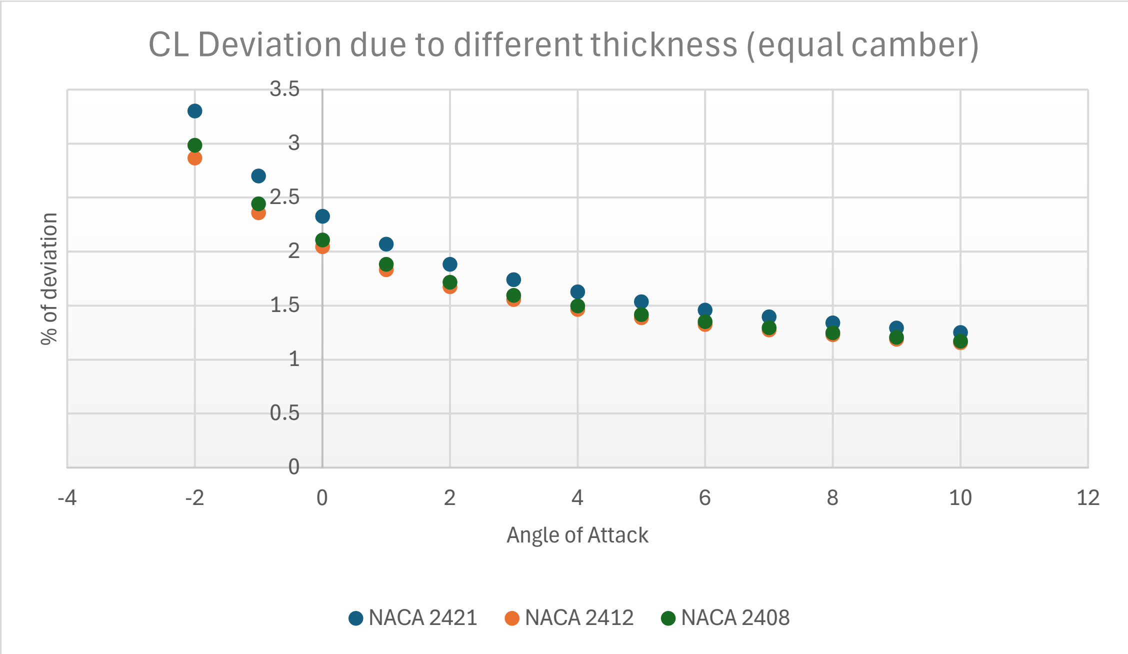

Figure 23:CL Deviation as a function of thickness

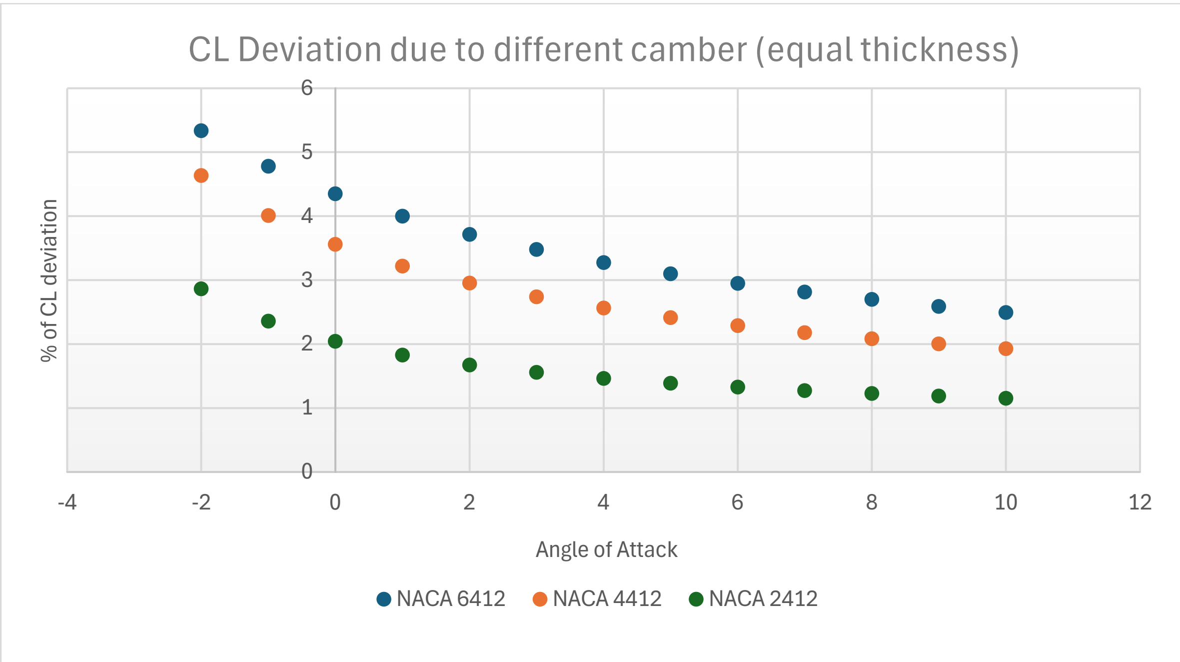

Figure 24: CL deviation as a function of camber

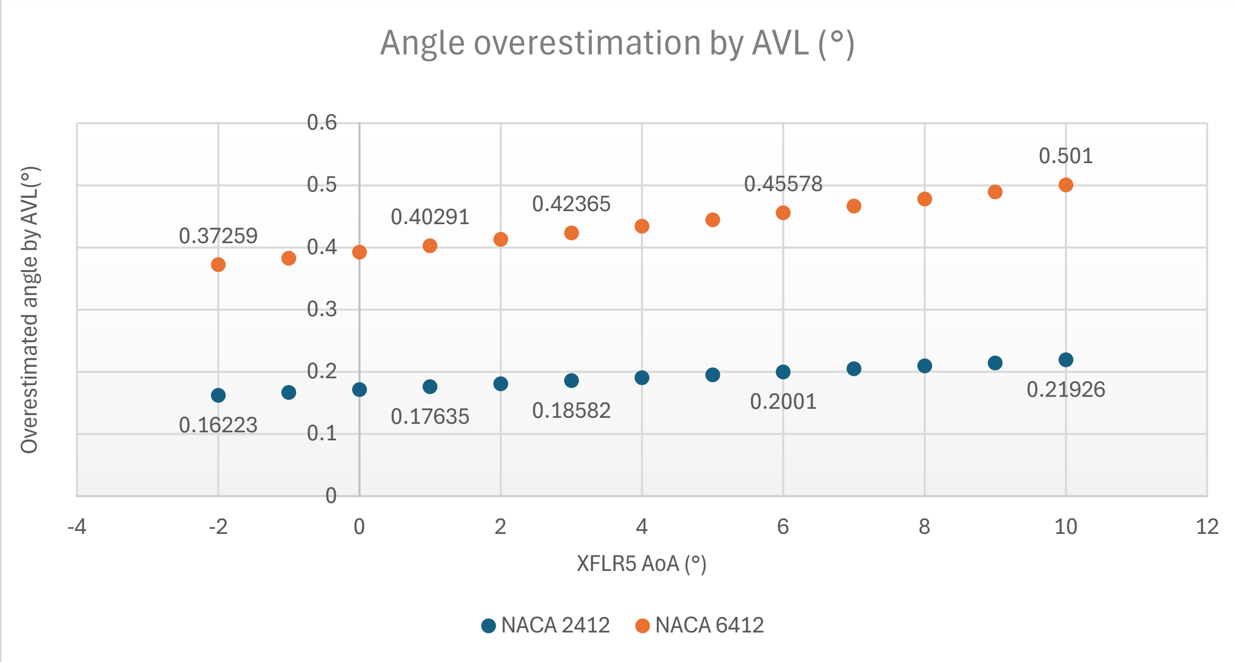

Figure 25: Angle overestimation by AVL.

In *figure 25* the overestimation of the AoA by AVL is visualized. The values calculated for 1° angle of attack on XFLR5 are corresponding to the AVL values for 0.403°,so when an 1° analysis is made on AVL the results taken are actually for a bigger AoA(≈1.4°). This can also be seen in *figure 26*.

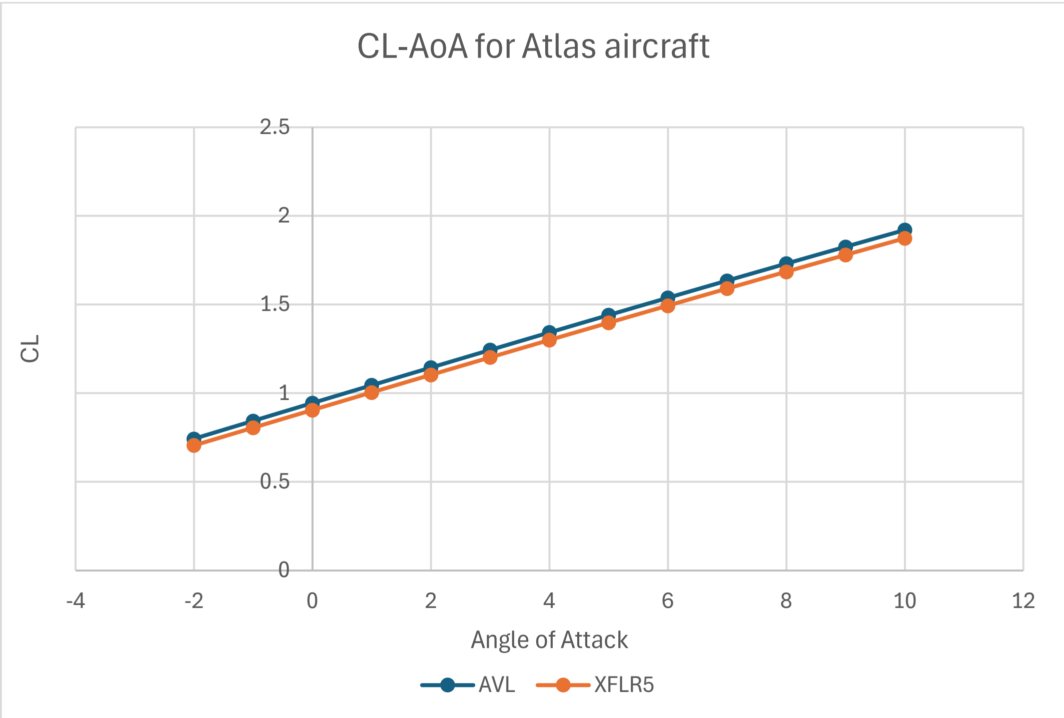

Figure 26:CL-AoA for ACC2024 Atlas UAV.

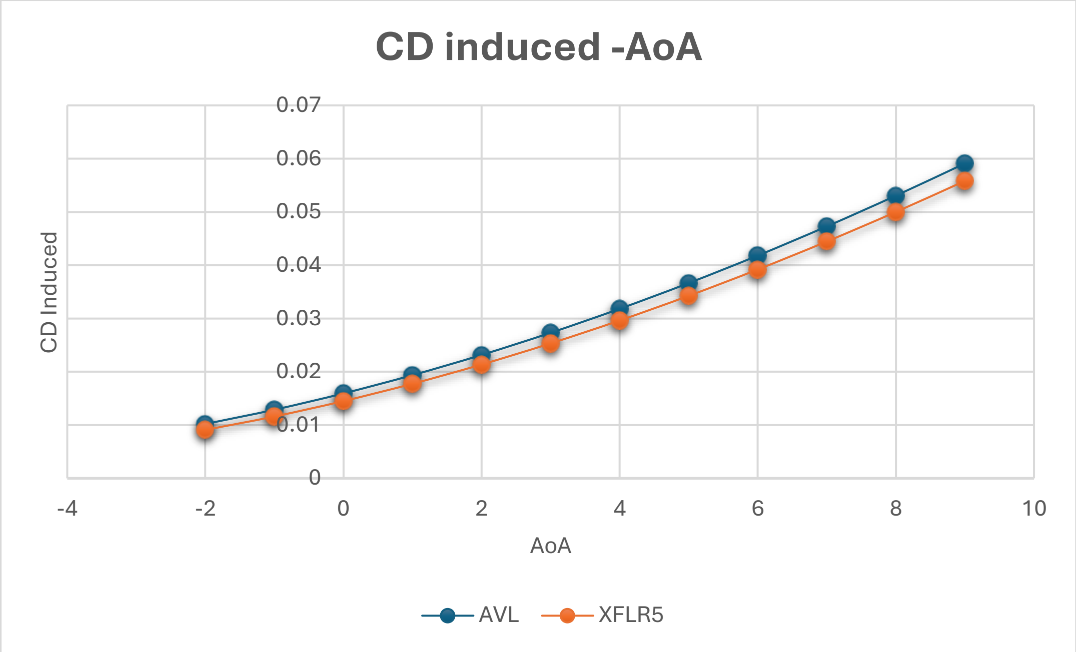

Figure 27:CD induced-AoA

Conclusion

AVL is a software with multiple capabilities. Aerodynamic analysis and stability analysis are the most attractive attributes it offers and can be very helpful, especially if integrated in an automated routine-script. XFLR5 is one of the best low fidelity software and its results can be sufficiently accurate for preliminary analyses. In order to understand the accuracy of AVL , a comparison between the two software were made for the UAV of the previous competition, designed by the team. From this comparison the conclusions derived were:

- AVL and XFLR5 model camber and thickness with different methods, thus results display a deviation , especially when high camber airfoils are used (≤7%).
- AVL estimates differently the angle of attack which leads to results that correspond to a slightly higher AoA than what is actually tested. The values of overestimated angle are not random but they follow a trend. It is possible to account for these values in a script and achieve exact similarity with xflr results(≤1% deviation).
- The higher the AoA the more the over estimated angle increases . However, deviation between CL values seem to have the opposite trend, which means that deviation will decrease while AoA increases.
- Geometry file for AVL is not hard to create but requires attention in specific parts such as the discretization of the lifting surfaces. For this matter , XFLR5 can also be consulted since it provides automatic discretization suggestion. In general, for the same discretization AVL appears to be slower than XFLR5. However, AVL does not necessarily require the same discretization as XFLR5 ,since results can be quite accurate for less dense discretization.
- The comparison was made for aerodynamic analysis so if stability is going to be used , further examination will be needed.
- It is possible to add a correction angle to get better accuracy, but since the angle is dependent on both airfoil and geometry characteristics, more tests will be required.

Notes & Future Work

A routine for AVL is going to be developed in python. The main idea is to automate AVL analyses for multiple geometries. A geometry file will be created by specific parameters that will work as input in the script. The automation will be mainly focused on aerodynamic analysis and not stability .

The effect that deviation has should be further examined. Examination should include the effect on other important parameters ,i.e. stability .

References - Bibliography

- **Katz, J., & Plotkin, A. (2001). *Low-speed aerodynamics* (Vol. 13). Cambridge university press.**
- **Drela, M. (2014). *Flight vehicle aerodynamics*. MIT press.**
- **AVL user guide , AVL**
- **Budziak, K. (2015). Aerodynamic analysis with athena vortex lattice. *Project, Hamburg University of Applied Sciences, Hamburg*, *29*.**