Aerodynamics Assignment

Drag Polar for Finite Wing Analysis using Vortex Lattice Method

SC20M001 - Srinivasan S
SC20M006 - Chaitanya G
SC20M008 - Krishna Vamsi C

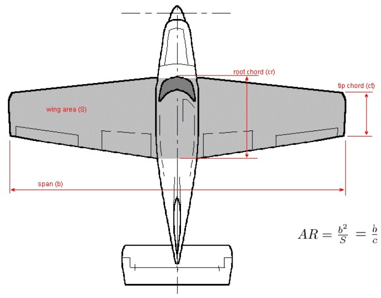
An assignment submitted for the course of Aerodynamics

07-12-2020

Aerospace Engineering
Indian Institute of Space Science and Technology
Valiamala

Contents

1 Geometry of a finite Wing



1.1 Wing

(i) Chord Length:
→ It is defined as the length from the Leading edge to the Trailing edge of a Wing and is denoted by 'c'.
→ In this experiment Chord Length 'c' is taken as the mean chord input from the user in MATLAB.

(ii) Aspect Ratio:
→ It is defined as the ratio of Wing span to the Chord length and is denoted by 'AR'.
→ In this experiment Aspect Ratio 'AR' is taken as the input from the user in MATLAB.

(iii) Wing Span:
→It is defined as the length measured from the root to the tip of the wing and is denoted by 'b'.
→In this experiment Wing Span 'b' is calculated using the input from the user in MATLAB.

$$b = c \times AR$$

(iv) Angle of Attack:
→It is defined as the angle between the freestream to the chord of the Wing and is denoted by 'α'.
→In this experiment Angle of Attack 'α' is restricted from 1 to 5.

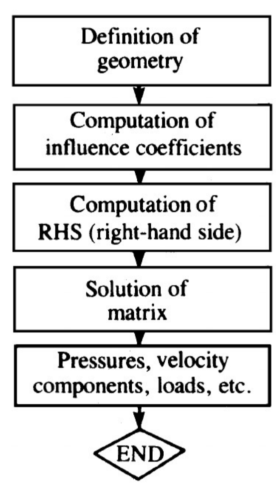
2 Procedure used to solve the problem

2.1 Vortex Lattice Method by Horseshoe Elements

→ We are approximating the lifting properties of a wing by a single lifting line , an approximation that will allow a closed-form solution.
→ Inspite of the considerable simplifications in this model it captures the basic features of three-dimensional lifting flows and predicts the reduction of lift slope and the increase in induced drag with decreasing aspect ratio.
→ 'n' chord wise vortices are used here.
→To solve the Laplace equation, the vortex line is a solution of the below equation and the only boundary condition that needs to be satisfied is the zero normal flow across the thin wing's solid surface.

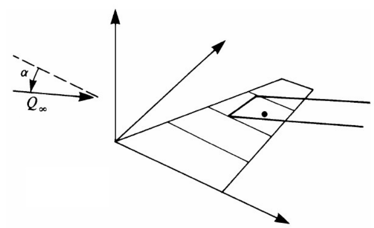
$$\nabla(\Phi + \Phi_\infty) \cdot n = 0$$

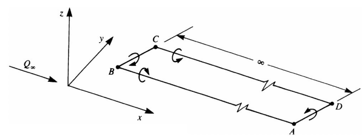
→ The wing is placed on the x-y plane and hence the boundary condition requires that the sum of the normal velocity component induced by the wing's bound vortices w_b, by the wake w_i and by the free-stream velocity Q_∞, will be zero.
→ Based on the discussed horseshoe element and on the above boundary condition we will construct a numerical solution, following the below flowchart:

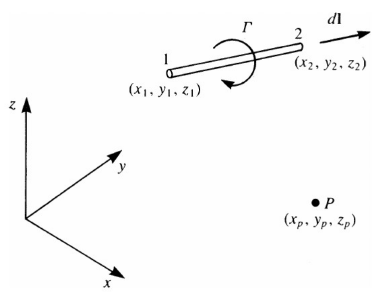


2.2 Selection of Singularity Element

→ A Flat finite wing of negligible thickness is selected to study the effect of Aspect Ratio on performance of using Vortex Lattice Method.
→ To solve this problem the horseshoe element shown in below fig is selected.







→ This element consists of a straight bound vortex segment BA that models the lifting properties and of two semi-infinite trailing vortex lines that model the wake.
→ The simple horseshoe vortex is kept and are shed at 20×b.
→ The span of the finite wing is considered as $\frac{-b}{2}$ to $\frac{b}{2}$.
→ To align the Trailing vortices parallel to XY plane the coordinate frame is changed.
→ The Flat finite wing is divided into 'n' Horseshoe elements.
→ Collocation points are selected at a distance of $\frac{3c}{4}$ in each panel.
→ Consider a vortex segment as shown in the above figure from point 1 to point 2. Then the velocity at an arbitrary point P (Collocation point) can be obtained by the below equation:

$$q_{1,2} = \frac{\Gamma}{4\pi}\frac{r_1 \times r_2}{|r_1 \times r_2|^2} r_0 \cdot (\frac{r_1}{r_1} - \frac{r_2}{r_2})$$

→ The velcity vector is obtained by taking the cross product of $r_1 \times r_2$ using the below equation:

$$r_1 \times r_2 = \begin{vmatrix} i & j & k \\ (x_p - x_1) & (y_p - y_1) & (z_p - y_1) \\ (x_p - x_2) & (y_p - y_2) & (z_p - y_2) \end{vmatrix}$$

2.3 Discretization of Geometry (and Grid Generation)

→ The wing is divided into n=100 panels (convergence results are better at n=100 for the code developed)

2.4 Influence Coefficients

→ The straight bound vortex $\Gamma(y)$ in this case is placed along the y axis and at each spanwise station the leading edge is one-quarter chord ahead of this line and the local trailing edge is three-quarter chord behind the vortex line.

$$\frac{\partial \Phi_{wing}}{\partial z} + \frac{\partial \Phi_{wake}}{\partial z} + Q_\infty \alpha$$

→ To obtain the influence coefficient matrix the below equation is used:

$$\omega_b + \omega_i + Q_\infty \alpha$$

→ where $w_b = \frac{\partial \Phi_{wing}}{\partial z}$ and $w_i = \frac{\partial \Phi_{wake}}{\partial z}$

→ The velocity induced by the horseshoe vortex element no. 1 at collocation point no. 1 is indicated for example as $(u, v, w)_{11}$.

where ij=11 denotes that i=1 is the no. of horseshoe vortex and j=1 is the no. of collocation point.

→ For any collocation point the no normal flow across the wing boundary condition can be written as

$$(u, v, w)_{ij} = HSHOE(x_1, y_1, z_1, x_{B1}, y_{B1}, z_{B1}, x_{A1}, y_{A1}, z_{A1}, x_{A'1}, y_{A'1}, z_{A'1}, x_{B'1}, y_{B'1}, z_{B'1}, \Gamma = 1)$$

→ Γ= 1 is used to evaluate the influence coefficient due to a unit strength

vortex.
$\rightarrow$ The velocities at one collocation point due to all vortex lines are obtained using 1st "for loop" for collocation points in MATLAB as:

$$\begin{pmatrix} a_{11} \\ a_{21} \\ a_{31} \\ - \\ - \\ - \\ a_{n1} \end{pmatrix}$$

$\rightarrow$ Similarly the velocity induced by all the vortices at the second collocation point is obtained and this continues until all the velocities at n- Horshoe vortices $\times$ n-collocation points are obtained. This is done by coding 2nd "for loop" for vortex elements in MATLAB.
$\rightarrow$ The final matrix can be formed by considering Γ=1 using the individual column matrices and equating to -$Q_\infty \cdot n_{1\rightarrow n}$ as follows:

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & - & - & -a_{1n} \\ a_{21} & a_{22} & a_{23} & - & - & -a_{2n} \\ a_{31} & a_{32} & a_{33} & - & - & -a_{3n} \\ - & - & - & - & - & - \\ - & - & - & - & - & - \\ - & - & - & - & - & - \\ a_{n1} & a_{n2} & a_{n3} & - & - & -a_{nn} \end{pmatrix} \cdot \begin{pmatrix} \Gamma_1 \\ \Gamma_2 \\ \Gamma_3 \\ - \\ - \\ - \\ \Gamma n \end{pmatrix} = \begin{pmatrix} -Q_\infty sin\alpha \\ -Q_\infty sin\alpha \\ -Q_\infty sin\alpha \\ - \\ - \\ - \\ -Q_\infty sin\alpha \end{pmatrix}$$

where n is the no. of collocation points
$\rightarrow$ The influence coefficients are defined as

$$a_{ij} \equiv (u, v, w)_{ij} \cdot n_i$$

2.5 Establish RHS

$\rightarrow$ The LHS of the influence matrix is obtained and to establish the RHS of the matrix i.e the freestream matrix, we now consider the normal component of the freestream to find out $-Q_\infty \alpha$.
$\rightarrow$ The normal velocity components of the free-stream flow $Q_\infty \cdot n_i$ are known and moved to the right-hand side of the equation:

$$R.H.S \equiv -(U_\infty, V_\infty, W_\infty) \cdot n_j$$

$$\Rightarrow a_{ij} \equiv (u, v, w)_{ij} \cdot n_i$$

$$\Rightarrow b_{ij} \equiv (u, v, w)^*_{ij} \cdot n_i$$

→ Here b_{ij} is the normal component of the wake-induced downwash that will be used for the induced-drag computations and $(u, v, w)^*_{ij}$.

2.6 Solve Linear Set of Equations

→ Using the LHS and RHS of the coefficient matrix, Gamma values are formulated using Matrix inversion method.
→ These formulated n unknown equations are solved.

2.7 Secondary Computations: Vortex Strength, Lift and Drag

→ Using the solutions of the n unknown equations the secondary information required in the problem i.e. C_l, C_d are determined and tabulated by adding up the individual contributions from the panels by using the below mentioned equations.

$$\Delta L_j = \rho Q_\infty \Gamma_j \Delta y_j$$

$$\Delta D_j = -\rho w_{ind_j} \Gamma j \Delta y_j$$

where $\Delta y = \frac{b}{N}$

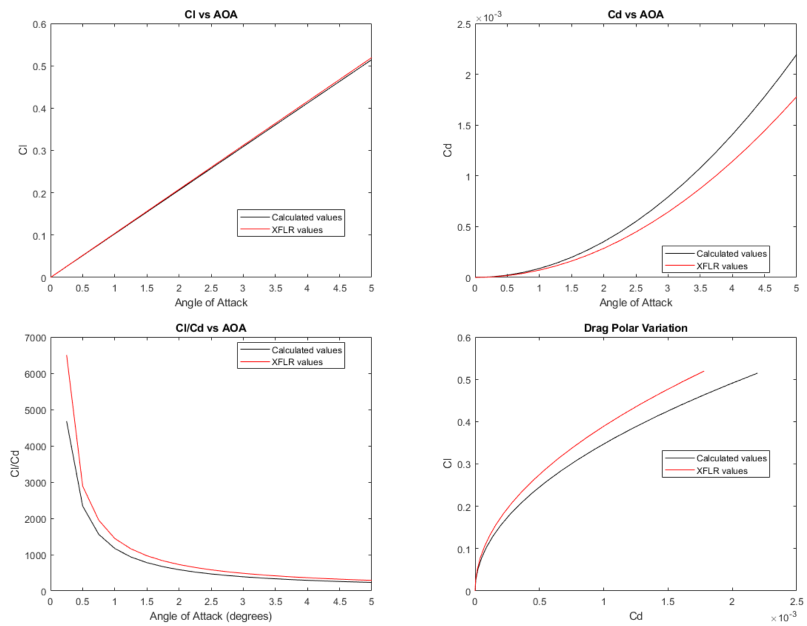
Δy_j is the is the panel bound vortex projection normal to the free stream.

3 Results

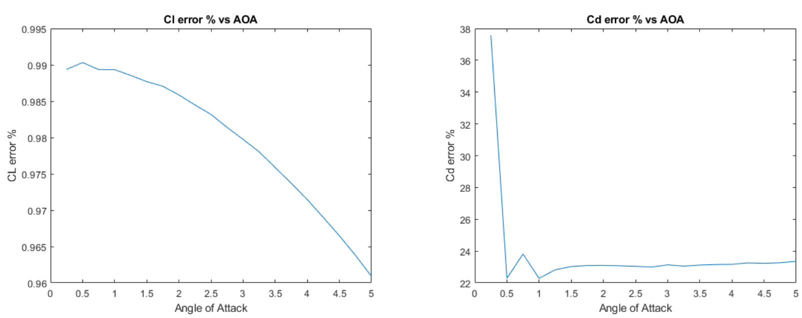
3.1 Input given in Matlab

$\rightarrow AspectRatio = 60$
$\rightarrow Chord = 1m$

3.2 Plots



3.3 Error



4 Conclusions

(1) For a increase in the Aspect Ratio 'AR' by user input, the error in C_l and C_d of MATLAB decreases when compared with XFLR5

(2) For a increase in the Angle of Attack, the error in C_l and C_d of MATLAB increases when compared with XFLR5

(3) C_l - Error variation is less compared to C_d - Error variation after comparing the MATLAB results with XFLR5.

(4) For less number of panels 'n' the C_l error and C_d error is getting reduced while comparing with XFLR5 results.
$\rightarrow$ For example $n = 15 \Rightarrow C_l$ error <5% and C_d error <5%