Universitat Politècnica de Catalunya
Escola Superior d'Enginyeries Industrial, Aeroespacial i Audiovisual de Terrassa

Aerodynamics Project: The constant strength vortex method and Prandtl's lifting line model

Aerodinàmica, Mecànica de Vol i Orbital
2024-25 Q2

02/06/2025

Group 12

Lluís Ceinos, Marc
Gutiérrez Portillo, Guillem
Poposki Stanojkovska, Vedran

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Contents

1 Introduction to the Problem

The objective of this project is to study the flow around the surfaces of a glider airplane composed by a wing, a canard and a vertical tail plane, all attached thanks to a thin, negligible, rigid rod, as shown in Figure 1. Both wing and canard have zero sweep angle and a trapezoidal shape, with their geometries defined by the following values:

$$b = 24m; cr = 1.8m; ct = 1.2m$$

$$b_h = 6m; cr_h = 1m; ct_h = 0.6m$$

The airfoils employed are the NACA 22112 and NACA 0012 for the wing and canard respectively. Incidence angles are 0° for the wing and 3° for the canard.

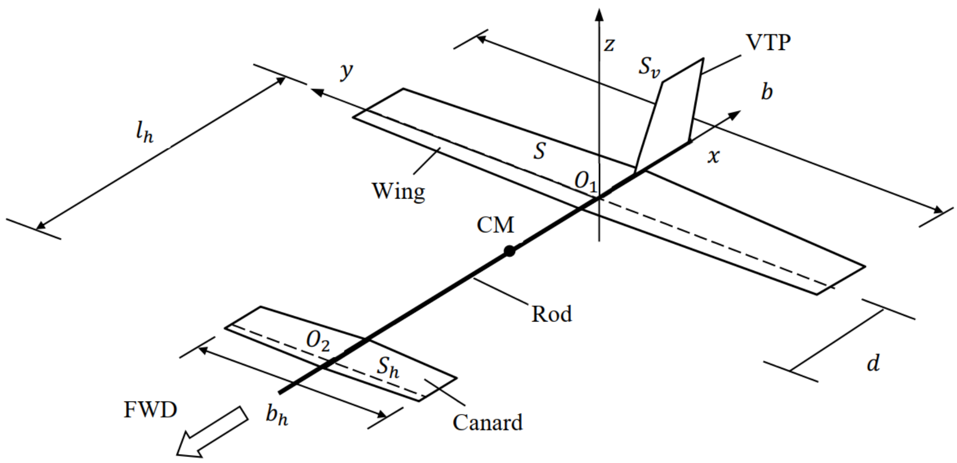

Figure 1: Configuration of the studied glider aircraft.

The problem is divided into the following parts:

- **Part 1:** Implementation of the constant strength vortex method. This part is itself divided in two, firstly the wing is studied with the objective to compute relevant aerodynamic coefficients, then the flow around the canard is calculated, considering it a combination of two airfoils, in order to understand the aerodynamic effects of having two areodynamic components very close together as can be observed in Figure 2.

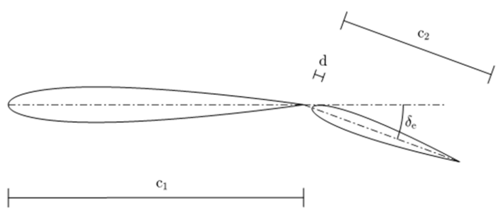

Figure 2: Configuration of the canard, with two NACA 0012 in succession.

- **Part 2:** Implementation of Prandtl's lifting line model.

2 Part 1

This first part of the problem is intended to compute and verify the numerical computation of convective and diffusive terms using a known analytical solution. The goal is to demonstrate second-order convergence by computing numerical errors at different grid resolutions and checking their convergence.

2.a Algorithm

Firstly, the intial condition is established a periodic velocity field as explained in the introduction. Then, the convective and diffusive terms for u and v are computed using symbolic differentiation.

The numerical computation is done for different grid resolutions ranging from 5 to 2560. The code loops over these grid sizes, repeating the following algorithm:

1. Addition of halo and calculation of cell size.
2. Computing numerical velocity field using set velocity field function with analytical u and v as inputs.
3. Computing numerical convective and diffusive terms using the dedicated functions and the calculated velocity field.
4. Compute analytical convective and diffusive fields on the grid
5. Compute the maximum error for each computed field convective u, convective v, diffusive u and diffusive v.
6. Store errors in preallocated storage arrays.
7. Plot errors.

Custom functions have been developed to compute, among other things, the convective and diffusive terms. The calculation of the convective term is done by firstly interpolating the velocities at the faces, then computing flow terms and the convective terms for each velocity direction. In the case of the diffusive term, it is first approximated using second-order finite differences (forward and backward).

It is also important to highlight the importance of performing a halo update after each domain-wide computation. This means that at the halo update function is called at the end of each of the other implemented functions, updating the values of the cells at the edges of the domain.

2.b Validation strategy

FUMABLE?

2.c Results

RESULTS AND FIGURES

3 Part 2

This second part of the problem aims to implement the pressure-velocity coupling and show how an arbitrary velocity field, after substracting the gradient of a certain type of field, can become a null-divergence one.

3.a Algorithm

The implemented algorithm follows a simplified version of a predictor-corrector scheme. The objective is to begin with a velocity field containing artificial divergence and apply a correction using a scalar field (pseudo-pressure) whose gradient will subtract the divergence component from the velocity.

1. Create coordinate positions, initialize null velocity field and introduce artificial divergence at a specific grid point by setting its horizontal and vertical velocities to non-zero values.

2. Compute the divergence of the predictor velocity field using a custom function.

3. Construct Laplacian matrix for pressure Poisson equation, solve a linear system for pseudo-pressure using the Laplacian equation and the divergence field in vector form.

4. Convert the pseudo-P vector back to 2D field format, then compute its gradients. Correct the velocity field using the calculated pseudo-P gradients. Then, compute the divergence of this new velocity field.

5. Compare maximum values of divergence before and after correction.

6. Plot the results.

A relevant remark about this code are that the divergence and gradient are calculated in the physical domain excluding the halo, which is calculated after. Both divergence and gradient are calculated using finite difference approximation, forward difference in the case of the gradient and backward for the divergence, ensuring consistency with the staggered-grid scheme.

3.b Validation strategy

SCZENSNY

3.c Results

RESULTS AND FIGURES

4 Conclusions

The numerical resolution of the incompressible Navier-Stokes equations has been implemented and validated in three stages.

In the first part, a known analytical solution was used to validate the implementation of convective and diffusive terms. By computing numerical errors over a range of grid sizes, the error plot confirmed a second-order convergence rate. This validates the finite-difference discretization schemes used for both convective and diffusive terms.

The second part focused on enforcing incompressibility using a projection method. Starting from a velocity field with artificial divergence. The pseudo-pressure gradients aligned correctly with the divergence sources, and the visualizations confirmed that the velocity field became divergence-free after correction.

The final part combined the previous components into a full time-dependent simulation . The temporal evolution was validated against the analytical solution, showing a close match to the analytical one in both velocity and pressure over time, and the convergence study confirmed second-order spatial accuracy. Divergence remained low throughout the simulation, confirming the effectiveness of the projection step. Additionally, particle trajectory animations showed smooth motion consistent with the velocity field.

Overall, the solver performs accurately and reliably across all tested aspects: It achieves second-order convergence in space, it correctly enforces the divergence-free condition through a pressure correction step and it reproduces time-dependent flow behavior with accurately compared to analytical references.