User guide for Aerospace Simulation Web Application

Contents

1 Introduction

The Aerospace Simulation Web Application provides a comprehensive, browser-based interface for performing aerodynamic and aeroelastic simulations. Built using Python and the Dash framework, it integrates multiple simulation engines, visualization tools, and user-friendly configuration forms. This guide explains how to install, use, and interpret results using this application.

2 Installation and Setup

2.1 System Requirements

- Windows 10 or newer (64-bit recommended)
- Python 3.13.2 or higher
- Internet browser

2.2 Cloning the Repository

To obtain the code, ensure you have Git installed on your system. If not, download it from `https://git-scm.com/` and clone the repository using the following command in the terminal :

```
git clone https://github.com/Aminelmk/AeroBoreal.git
```

This will create a folder named `AeroBoreal` containing the application code. Keep the repository structure intact to ensure correct file path referencing.

2.3 Python and Dependencies

Ensure Python is installed and accessible from the terminal. From the project directory, install all required packages using:

```
pip install -r requirements.txt
```

2.4 Git Extension Requirement

If your IDE or workflow requires Git integration, ensure the appropriate Git plugin or extension is installed. For example, Visual Studio Code users can install the GitLens extension from the VS Code Marketplace.

2.5 Launching the App

Navigate to the project folder in your terminal and run:

```
python main.py
```

This will start a local web server at `http://127.0.0.1:8050`. Open this URL in your browser.

3 Application Overview

3.1 Main Modules

The home page provides access to the following simulation components:

- **Mesh 2D**: Structured mesh generation for airfoil analysis
- **Mesh 3D**: Panel mesh generation for wings
- **Euler 2D**: Inviscid flow simulation around airfoils
- **VLM-Structure 3D**: Coupled aerodynamic-structural simulation
- **Pressure Visualization**: Display of 2D/3D simulation results

4 2D Mesh Generation

4.1 Geometry Definition

Airfoil Input Methods:

- **NACA 4-digit**: Specify camber, position, and thickness
- **CST**: Upload coordinates and fit a Class-Shape Transformation
- **B-Spline**: Upload coordinates and fit a B-spline curve (user defines number of knots and spline degree)

4.2 Mesh Parameters

Conformal mapping
Conformal mapping grid is only compatible with symmetric NACA airfoil. If another type of airfoil is generated, the *Generate Mesh* button will be disabled.

- **Resolution**: Choose grid resolution

Elliptic mesh
Elliptic grid is compatible with most airfoil shape.

- **Resolution**: Choose grid resolution
- **Farfield**: Choose the distance of the farfield
- **Solver Settings**: Set max iterations and convergence tolerance

4.3 Saving and Using the Mesh

When generating the mesh, it is automatically saved in Plot3D format in a temporary directory. This mesh will be used by default by the Euler CFD solver. It is possible to save the mesh elsewhere using the download button.

5 3D Mesh Generation

5.1 Wing Geometry Types

- **Custom**: Define aspect ratio, sweep, dihedral, taper, and twist
- **Elliptic**: Preset elliptical loading with custom span
- **CRM or Lovell**: Built-in reference wing models

5.2 Discretization and Export

Choose the number of panels in the X (chordwise) and Y (spanwise) directions. The exported mesh file `mesh3d.x` is used by the coupled solver.

5.3 Structural Settings

- Toggle structural node visualization
- Set the number of structural elements and their chordwise position (e.g., quarter-chord)

6 Euler 2D Solver

6.1 Freestream Configuration

Input fields:

- Mach number M_∞
- Angle of attack α
- Pressure p_∞ [Pa]
- Temperature T_∞ [K]

The pressure and temperature do not influence the Euler simulation. The coefficients are adimensionlized using $q_\infty = \frac{1}{2}\rho_\infty U_\infty^2$.

6.2 Solver Controls

The Euler CFD solver uses the JST dissipation scheme.

- CFL number, number of iterations, k2/k4 dissipation terms. For a user provided n coefficient, the κ of the JST scheme are defined as follow: $\kappa_2 = n \times \frac{1}{4}$ and $\kappa_4 = n \times \frac{1}{64}$.
- Option to enable/disable multigrid and residual smoothing

7 Coupled 3D Simulation

7.1 Aerodynamic Inputs

- Mach number, angle of attack, pressure, and temperature
- Number of Euler profiles along the wing span
- Table of spanwise coordinates and corresponding airfoil files

7.2 Structural Inputs

- Number of structural elements
- Table with: cross-sectional area A, Young's modulus E, Poisson ratio ν, moments of inertia I_y, I_z, torsional constant J

7.3 Solver Execution

The solver iteratively couples aerodynamic and structural computations. Outputs include a VTU file and spanwise data used for post-processing and visualization.

8 Visualization

8.1 Euler 2D Results

- Contour plots of pressure, Mach number, density ρ, velocity components u and v
- Streamlines and mesh visualization toggle
- Display of aerodynamic coefficients: C_L, C_D, C_M

8.2 3D Pressure Visualization

- Toggle between undeformed and deformed geometries
- Visualization of pressure, pressure coefficient C_p, lift, and drag distributions
- Interactive plots: T_z, R_y, C_l, and α_e along the wing span
- Clickable panel interface for detailed information

9 Conclusion

This application enables users to generate meshes, configure simulation parameters, run both CFD and aeroelastic solvers, and analyze results—all from a single, user-friendly web interface. It is optimized for subsonic and transonic flows and assumes compressible, inviscid flow for Euler simulations and linear structural behavior for the coupled solver. The tool is ideal for preliminary design and educational purposes.