

SU2 and OpenFOAM validation for high lift configuration using HLPW5 cases

Ion MATEI^{*,1}, Laurentiu I. PADUREANU¹, Georgiana ICHIM¹

^{*}Corresponding author

¹INCAS – National Institute for Aerospace Research “Elie Carafoli”,
B-dul Iuliu Maniu 220, Bucharest 061126, 6, P.O. 76, Romania,
matei.ion@incas.ro, padureanu.laurentiu@incas.ro, ichim.georgiana@incas.ro

DOI: 10.13111/2066-8201.2025.17.4.7

Received: 15 September 2025/ Accepted: 22 October 2025/ Published: December 2025

Copyright © 2025. Published by INCAS. This is an “open access” article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

Abstract: Running Computational Fluid Dynamics (CFD) simulations include multiple parameters that influence numerical results such as grid type and refinement level, turbulence model, solver type, numerical scheme and other elements. For studying these parameters and their influence on CFD results, NASA organized Drag Prediction Workshop (DPW) and High Lift Prediction Workshop (HLPW). These events are attended by multiple research and industry entities that collaborate to improve industrial CFD simulation by studying the effect of various parameters on results. Such studies lead to new guidelines - best practices for numerical simulation engineers around the globe that would result in best accuracy with respect to experimental data. HLPW is focusing on aircraft configurations at low speed ($M=0.2$) with deployed slats and flaps. The current paper follows two configurations, one of a simple aircraft without high lift devices, in order to establish grid convergence for solvers at near stall angle of attack and the second configuration featuring a deployed slat, for which experimental results are available. For these two cases custom grids were generated and RANS simulation were performed with SU2 and OpenFOAM with respect to HLPW committee guidelines and suggestions. The set of results, comprising aerodynamic coefficients, pressure coefficient and flow patterns were analysed and compared with experimental data and selected results from the mentioned workshops.

Key Words: CFD, SU2, OpenFOAM, BETA CAE ANSA, Turbulence, HLPW5

1. INTRODUCTION

Solving the RANS equations is a complex task that requires sophisticated numerical programs developed over decades, each with an established user base. Industry-leading CFD solvers such as Ansys FLUENT, NASA FUN3D, and Siemens Star-CCM+ are recognized for their robustness, user-friendliness, and comprehensive documentation.

However, these programs come with significant costs and limitations, particularly regarding the number of cores and instances available for use. In contrast, several promising open-source CFD programs are available for users seeking free alternatives.

The most notable are OpenFOAM, the oldest and most widely used, and SU2, a more recent and modern approach to CFD.

Both are multiphysics simulation tools that extend beyond CFD applications and are designed to solve partial differential equations (PDEs). Both are primarily written in C++, with SU2 also incorporating Python, enabling the use of a Python wrapper for easier automation

and accessibility for users unfamiliar with coding. OpenFOAM can also be automated using C++ scripts, but this requires in-depth programming knowledge.

The main drawbacks of these open-source programs are their steep learning curve, limited documentation and tutorials, and the absence of a graphical user interface (GUI). These factors make setup and solution monitoring more challenging compared to commercial software. On the other hand, open-source tools offer greater control over numerical schemes and overall setup precision - a double-edged sword, as errors in setup may go unnoticed without built-in warnings or fail-safes. The purpose of this research was to familiarize ourselves with both programs, gain practical experience, and validate results by comparing them with data from CFD workshops available on NASA's website. The focus was on test cases from HLPW5, specifically case 1 (a near-stall grid convergence study) and case 2.2 (a challenging complex geometry configuration), both of which are difficult to run and converge but are accessible in term of computational power available at INCAS, cases 2.3 and 2.4 are too demanding in this regard and where not computed.

2. NUMERICAL MODEL

SU2 and OpenFOAM solve the same problems but employ different approaches. Both solvers use the Finite Volume Method; however, OpenFOAM is cell-centered (unknown quantities are calculated at the cell centres), while SU2 is node-centered (unknowns are calculated at the edge points).

For this study, the OpenFOAM ESI version (v2212) was used, and the rhoSimpleFoam solver was selected to obtain steady-state results for the cases analysed. In SU2, the SU2_CFD solver with a RANS model was utilized, featuring a density-based algorithm specifically designed for compressible flow. Both solvers are second order accurate in line with the mainstream solvers.

Navier Stokes equation [1] that are governing the flow are:

- **Continuity equation:**

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \mathbf{V}) = 0 \quad (1)$$

- **Momentum Equation:**

$$\frac{\partial}{\partial t}(\rho \mathbf{V}) + \nabla(\rho \mathbf{V} \otimes \mathbf{V}) = \rho \mathbf{f} - \nabla p + \nabla \bar{\tau} \quad (2)$$

where $\bar{\tau}$ is the viscous tensor and is expressed in terms of the strain rate tensor $\bar{s}$:

$$\bar{\tau} = 2\mu\bar{s} - \frac{2}{3}\mu\nabla\nabla \quad (3)$$

- **Energy Equation:**

$$\frac{\partial}{\partial t}(\rho \mathbf{E}) + \nabla(\rho H \mathbf{V}) = \rho \mathbf{f} \mathbf{V} + q_v + \nabla(\kappa \nabla T + \bar{\tau} \mathbf{V}) \quad (4)$$

Equations (1), (2) and (3) can be written in a vectorial form as:

$$\frac{\partial}{\partial t} \begin{bmatrix} \rho \\ \rho \mathbf{V} \\ \rho \mathbf{E} \end{bmatrix} + \nabla \cdot \begin{bmatrix} \rho \mathbf{V} \\ \rho \mathbf{V} \otimes \mathbf{V} + p \bar{\mathbf{I}} - \bar{\boldsymbol{\tau}} \\ \rho \mathbf{V} H - \kappa \mathbf{t} - \boldsymbol{\tau} \mathbf{V} \end{bmatrix} = \begin{bmatrix} 0 \\ \rho \mathbf{f} \\ \rho - \mathbf{f} q_v \end{bmatrix} \quad (5)$$

where $\bar{\mathbf{I}}$ - is identity matrix of third order.

Or, in compact form:

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \mathbf{F} = \mathbf{Q} \quad (6)$$

where $\mathbf{U}$ is the vector of conservative variables, $\mathbf{F}$ is the flux vector, and $\mathbf{Q}$ is the source vector.

By averaging these equations, we obtain the RANS equations, which require a closure model represented by the turbulence model.

For both software options, the Spalart-Allmaras one-equation turbulence model is used. However, SU2 employs more modern modifications that enhance the robustness and accuracy of the model: the rotation correction (denoted by R in the turbulence model name) and the quadratic constitutive relation (denoted by QCR-2000) [2]. Thus, OpenFOAM uses the standard SA model, while SU2 uses the SA-QCR2000-R model.

2.1 Numerical Model in OpenFOAM

OpenFOAM's rhoSimpleFoam [3] solver, a compressible, steady state solver with turbulence modelling capabilities was used with SIMPLEC (Semi-Implicit Method for Pressure Linked Equations Consistent) algorithm, a pressure based algorithm that was used in the past only for incompressible flows but was modified and extended to compressible flow scenarios. A simplified overview of how this method works is [4]:

1. Advance to the next iteration $t = t^{n+1}$.
2. Initialise u^{n+1} and p^{n+1} using latest available values of u and p .
3. Construct the momentum equations.
4. Under-relax the momentum matrix.
5. Solve the momentum equations to obtain a prediction for u^{n+1} .
6. Construct the pressure equation.
7. Solve the pressure equation in consistent form for p^{n+1} .
8. Correct the flux for ϕ^{n+1} .
9. Under-relax p^{n+1} .
10. Correct the velocity for u^{n+1} .
11. If not converged, go back to step 2.

This algorithm is fast, robust, and works well for coupled flows. It also has the advantage of a low memory footprint compared to density-based solvers, as it operates only with vectors, not matrices (a characteristic of density-based solvers).

Additionally, Rhie-Chow interpolation is implemented to address the instabilities that may arise when using a pressure-velocity algorithm on a collocated grid (where variables are stored at cell centers).

For gradient computation, the Green-Gauss method with linear interpolation is used. For velocity, pressure, and energy, a second-order upwind scheme with a cell limiter is applied. Overall, a robust and second-order numerical setup was employed to achieve a stable and accurate solution with minimal impact on the results.

2.2 Numerical Model in SU2

The SU2_CFD module is tailored for solving CFD problems and was designed for a wide range of cases, from subsonic to supersonic Mach numbers. The solver uses an implicit, density-based approach with multigrid capabilities to speed up convergence. The implicit solver offers less dissipation and allows for higher possible CFL (Courant) numbers.

For gradient reconstruction, the Weighted Least Squares (WLS) scheme was chosen. For the convective scheme, the Upwind SLAU2 scheme - a member of the AUSM family—was used due to its high robustness and good convergence characteristics.

Second-order accuracy is achieved by using Monotone Upstream-centred Schemes for Conservation Laws (MUSCL) with the Van Albada edge limiter for flow variables. For turbulence quantities, the Nishikawa_R5 limiter, a modified Venkatakrishnan-based limiter, is selected. Turbulence quantities are solved separately from flow variables to make the code more modular and facilitate changes to the turbulence model.

The numerical viscous fluxes are determined by averaging the flow variables, flow derivatives, and transport properties between two neighbouring cells. The resulting linear matrix system can be solved using available linear solvers (FGMRES, BCGSTAB, SMOOTHER) [5].

The settings chosen for the current case are FGMRES with a LU-SGS preconditioner, which works well with Upwind AUSM schemes. All of the above yields a highly efficient, second-order accurate RANS solver that works well for the selected cases.

3. GRID GENERATION

For a successful RANS analysis, the grid is the main ingredient that needs to be perfected. For this purpose, the ANSA v24.1 program was used—an advanced pre-processor that is part of the Beta CAE Systems suite. ANSA is a Swiss Army knife tool with broad applications across multiple industries, including automotive, aerospace, marine, and bioengineering. It can generate grids for CFD, FEM, impact, and other applications, and it can output grids compatible with a wide range of programs (such as NASTRAN, ANSYS Fluent, ANSYS Mechanical, LS-DYNA, ABAQUS, OpenFOAM, TAU, and SU2). Additionally, ANSA can set up cases where possible.

Grid generation begins with geometry import. The geometry is available on the workshop website [6] in both IGES and STEP formats.

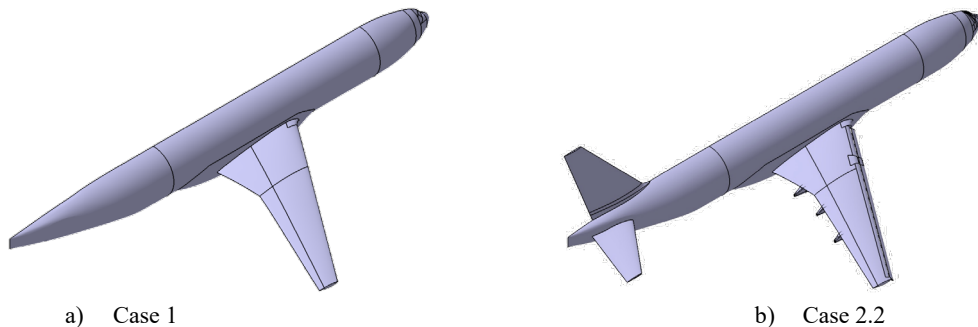


Fig. 1 Geometries of test cases.

After importing the geometry, the cleanup process begins. During this stage, surfaces are assigned PIDs, and additional curves on surfaces are deleted to facilitate mesh generation. Special geometric features are identified to apply targeted mesh treatments, particularly for

sharp edges (SE), leading edges (LE), and trailing edges (TE). This allows the use of anisotropic cells (stretched elements), which reduce the number of elements required for capturing curvature at the LE, refining at the SE, and meshing the TE without resorting to a high cell count.

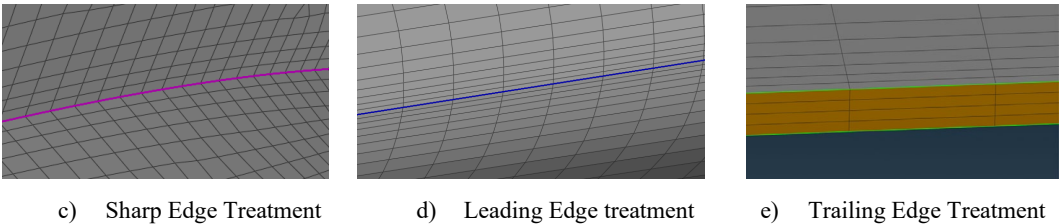


Fig. 2 Special Feature Treatment.

Following this step, the domain for the simulation is created. This is a rectangular domain with edges of $1000 \cdot C_{ref}$, where C_{ref} is the reference chord, equal to 7 meters. Domain creation is automated, and if the geometry is correctly positioned, the domain and aircraft are sealed together to form a watertight geometry. Next, meshing scenarios are created: Surface Scenario, Layer Scenario, and Volume Scenario. These scenarios allow the specification of target sizing and curvature refinement for each PID assigned during geometry cleanup. For layer generation—a crucial step in grid generation for CFD - the first cell height and growth ratio are defined, along with a stopping criterion for layer generation based on the aspect ratio, $AR = \frac{h}{\sqrt{(A)}}$, with a target value of 0.8. This ensures that the elements are nearly cubic, enabling the outer volume mesh to connect seamlessly to the layer mesh and minimizing numerical diffusion caused by abrupt changes in cell size. For volume meshing, the algorithm, maximum element size, growth ratio, and number of buffer zones (the number of volume elements before doubling the element size) are specified. The Hexa-interior option is best suited for CFD applications, as it minimizes the number of elements while maintaining high grid resolution. Once all meshing scenarios are set, the mesh can be generated with a single click. However, it is essential to run the surface scenario first and inspect the resulting mesh to ensure sufficient quality for further layer mesh generation.

The grid sizing were derived from the HLPW4 grid generation guidelines and are presented in Tables 1 and 2.

Table 1. Recommended surface Mesh Characteristics [7]

Mesh	Fuselage Cell Size	Chordwise Spacing	Spanwise spacing	Cells on TE	Mesh Factor
A	$\leq 2.25\% C_{REF}$	$\leq 0.225\% Local\ chord$	$\leq 0.225\% Semi-span$	4	-
B	$\leq 1.5\% C_{REF}$	$\leq 0.15\% Local\ chord$	$\leq 0.15\% Semi-span$	6	2/3
C	$\leq 1.0\% C_{REF}$	$\leq 0.1\% Local\ chord$	$\leq 0.1\% Semi-span$	9	2/3
D	$\leq 0.75\% C_{REF}$	$\leq 0.075\% Local\ chord$	$\leq 0.075\% Semi-span$	12	3/4

Table 2. Recommended Volume Mesh Characteristics [7]

Mesh	Initial Wall Spacing [mm]	Estimated y^+	Boundary Layer Growth Rate	Mesh Factor
A	0.060706	2.25	1.25	-
B	0.04064	1.5	1.16	2/3
C	0.026924	1	1.1	2/3
D	0.02032	3/4	1.03	3/4

In total, six grids were generated: four for case 1 (A, B, C, D) and two for case 2.2 (A, B). Thanks to the automatic grid refinement feature in ANSA, only mesh A was required for both cases, significantly reducing the workload for grid generation. The number of elements and volume are summarized in Table 3, and a section view of the mesh for case 2.2 is shown in Figure 3.

Table 3. Mesh Family Characteristics

Characteristic	1-A	1-B	1-C	1-D	2.2-A	2.2-B
Surface elements [$\cdot 10^3$]	66.5	122	220	441	440	950
Volume elements [$\cdot 10^6$]	5	10	17.5	36.5	25.7	68

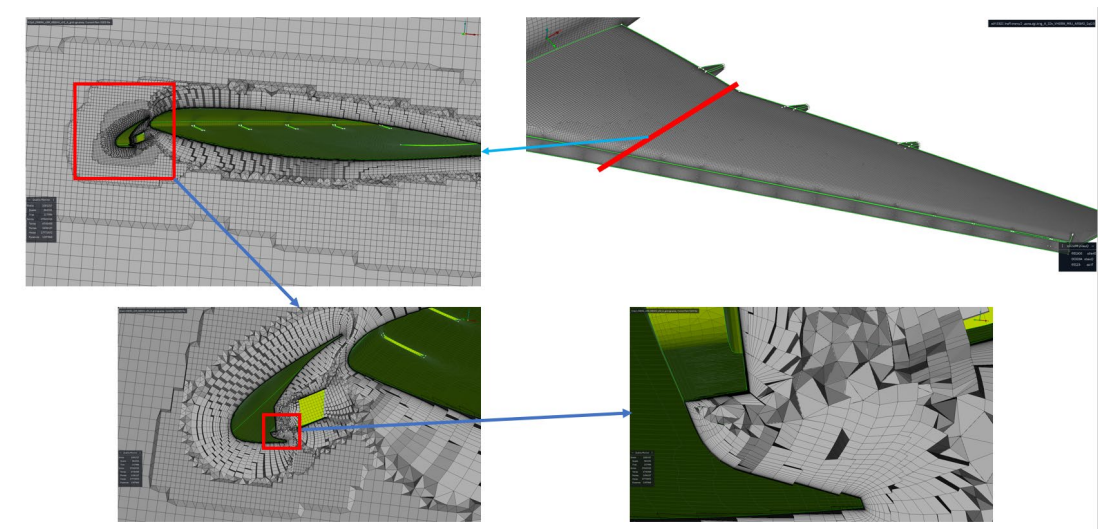


Fig. 3 Cross section of mesh for case 2.2.

4. RESULTS

For both OpenFOAM and SU2, convergence was achieved after a maximum of 5,000 iterations, with all residuals dropping by a minimum of four orders of magnitude and entering a plateau when force convergence was achieved.

Case 1 results were compared to those of the participants of HLPW5, as experimental results were not yet available.

Case 2.2, on the other hand, had extended wind tunnel testing, with experimental coefficient data, pressure distribution, and flow visualization available, which helped compare the results and determine whether the CFD results captured the underlying physics of the flow.

Between starting the solution from a previous converged case (hot start) and a cold start (starting from a uniform value field), the former was preferred due to better results.

Flow conditions and case setup were in accordance with HLPW5 requirements, with freestream pressure and temperature equal to those of the standard atmosphere at 0 meters altitude, and viscosity adjusted to achieve the desired Reynolds number. Table 5 shows the required angle of attack (AoA) and Reynolds number for simulations.

Table 4. Principal flow parameters for CFD cases

Case 1	Case 2.2
$\alpha = 11^\circ$	$\alpha = 6^\circ, 12^\circ, 16^\circ, 17.7^\circ$
$MRe = 5.6$	$MRe = 5.9$

In total, 8 simulations were carried out for Case 1 and 10 for Case 2, with 2 additional simulations for OpenFOAM due to the high variation in angle of attack from case to case, which caused some simulations to stall early.

In terms of computational cost, approximately 8,000 core-hours, or 130 hours of wall time, were consumed on a dual Intel Xeon Gold 6430 workstation.

4.1 Case 1 Results

All the results were processed using Tecplot 360 EX 2019 and the post processing information available on the HLPW5 website, where multiple macros that automate results extraction are provided.

Below (see Figs. 4, 5, 6), the results for the force coefficients are shown, with the results of this study on the left and the data from the HLPW5 participants on the right for comparison.

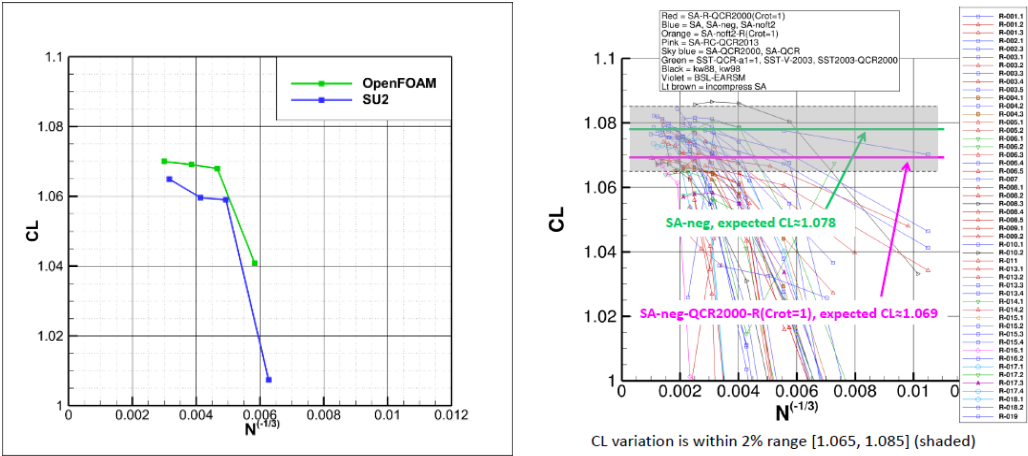


Fig. 4 CL Grid convergence case 1

The lift coefficient (CL) shows decreasing variation with grid refinement, indicating grid convergence for both solvers. The results are in line with and comparable to other HLPW5 participants' data, providing confidence in SU2 and OpenFOAM to be used in near stall condition.

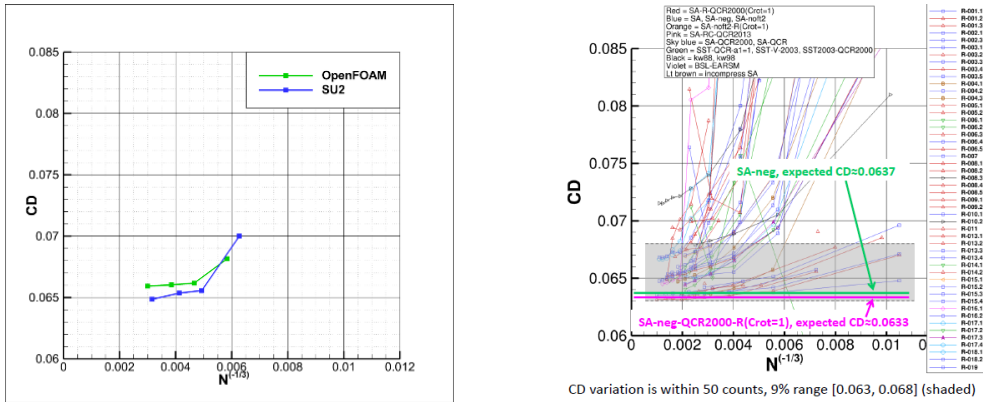


Fig. 5 C_D Grid convergence case 1

The drag coefficient (C_D) value, as expected is more sensible to grid resolution, with additional refinement levels needed for a 100% confidence in grid convergence. In terms of trend and value itself results are in line with expected value with a deviation of only 3% from the results on the finest grid.

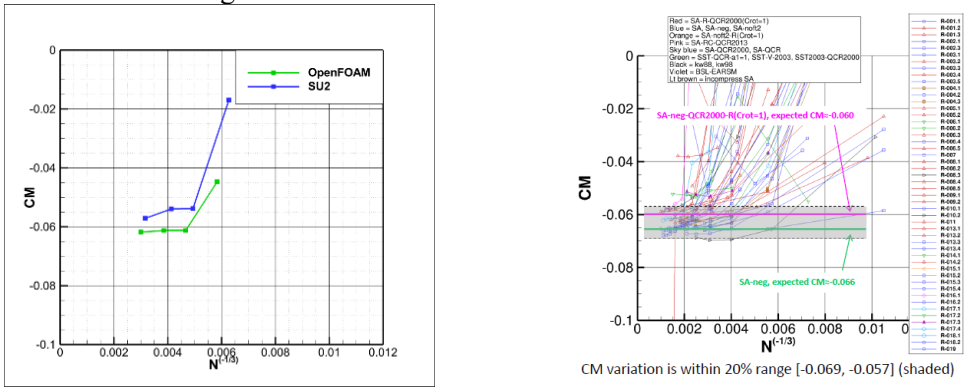


Fig. 6 C_m Grid convergence case 1

C_M is similar to C_L , good agreement with other participants, low variation between grid B, C, D showing an asymptotic grids convergence. Sizable percentual differences compared to expected value ($\sim 10\%$) but is mainly due to low overall value of the moment coefficient.

Case 1 results all fall within the grey band, where most of the HLPW5 data lies. The results improve monotonically with refinement, demonstrating that the SU2 and OpenFOAM implementations of the turbulence model are consistent and align with the expected behavior for CFD solvers (the slope of variation with grid size has the same sign). n terms of grid convergence, although it was not fully achieved and running on a finer mesh could be beneficial, the results presented in the study are more than satisfactory for identifying the behavior of the solvers and their usability - especially considering the use of a custom grid and the constraints of computing power.

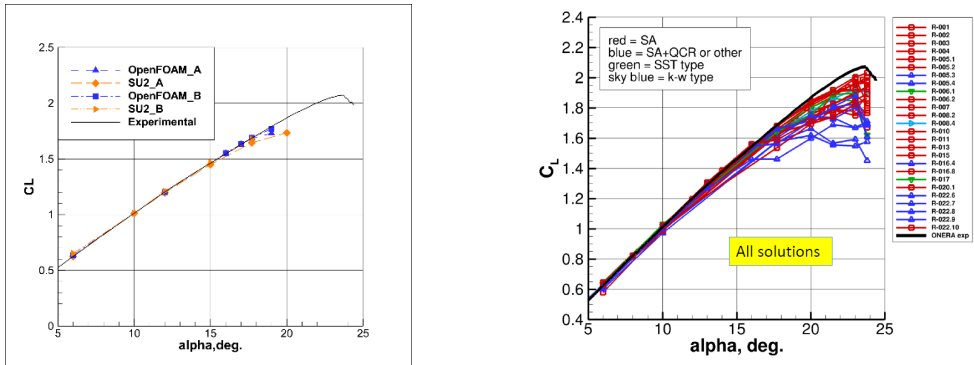
4.2 Case 2.2 Results

The experimental results for case 2.2 come from a testing campaign conducted by ONERA in the ONERA F1 Pressurized Low-Speed Wind Tunnel. The wind tunnel is specialized for studying take-off and landing configuration. The full model configuration in the tunnel is presented in Figure 7. The campaign provided force data, pressure data, and flow visualization for near-stall cases.

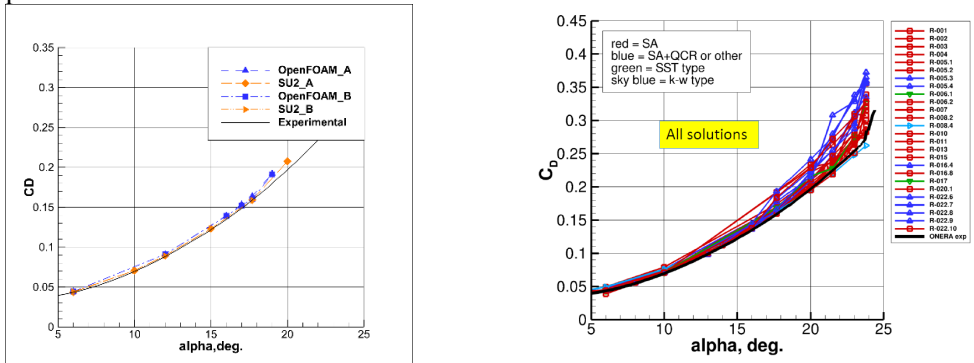


Fig. 7 Full configuration Experimental model in Tunnel [8]

One of the most important aspects of high-lift configuration is determining the maximum lift coefficient. Upon inspecting the results, we can see that both SU2 and OpenFOAM exhibit early stall on both Grid A and Grid B. This could suggest that a more refined mesh is necessary, or – as is the case with multiple participants – that RANS simulations are not sufficient to capture the flow characteristics. Specifically, strut support interference may be overestimated in RANS, leading to early stall. On the bright side, the results up to an angle of attack (AoA) of 17.7° coincide with the experimental data and other participants and the divergence from experimental trendline is for majority of participants at 20° similar to SU2 and OpenFOAM results.

Fig. 8 C_L Results Case 2.2

In terms of C_D , the values closely follow the experimental values up to 17.7° , similar to the C_L results, and then diverge - though not by a large margin - consistent with other participants' data.

Fig. 9 C_D Results Case 2.2

The moment coefficient seems to be the more sensitive coefficient with a pitch breaking happening after the 17.7° . With moment coefficient all the participants encounter this, even

on those where early stall in lift coefficient didn't occur. Comparing the results for $AoA \leq 17.7^\circ$. There is a good agreement between experimental data and CFD results.

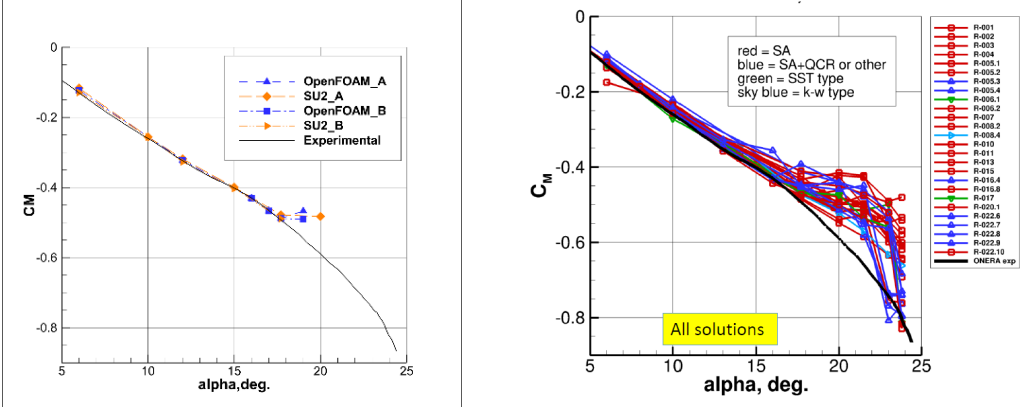


Fig. 10 C_M Results Case 2.2

Now, proceeding to a comparison of the pressure distribution between experimental and CFD data, three locations (Fig. 11) for 6° and 17.7° were chosen.

There is good agreement between both solver results and experimental data for row A and row D at both angles of attack.

For row I, a difference in pressure distribution is visible on the lower surface of the wing, which is probably caused by the wind tunnel walls interfering with the flow. On the upper surface, divergence from the experimental results is not present.

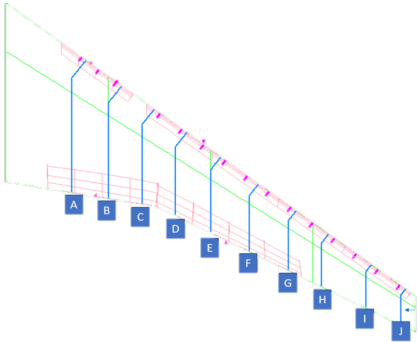


Fig. 11 Wing cuts positions

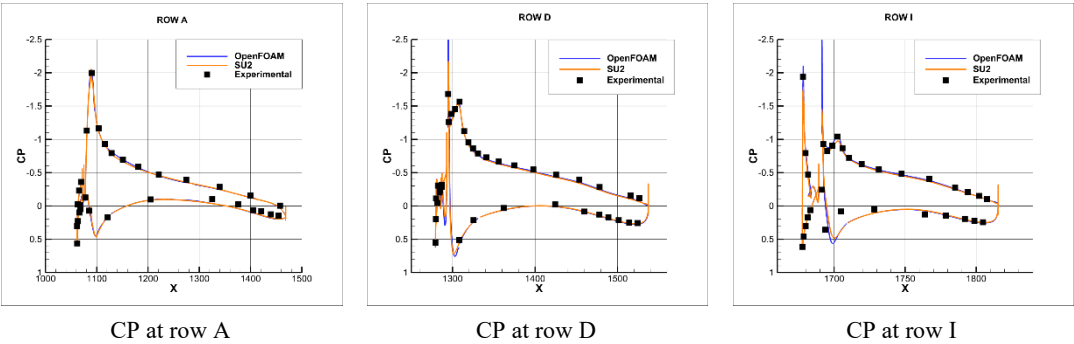
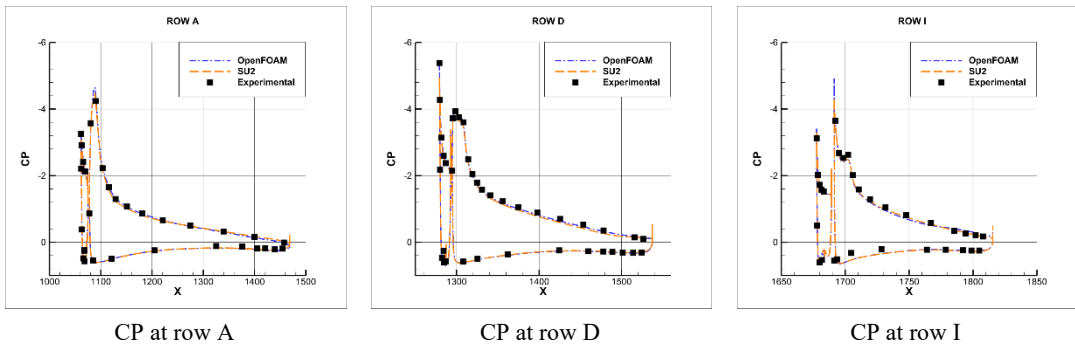


Fig. 12 CP comparison for $AoA=6^\circ$

Fig. 13 CP comparison for $AoA=17.7^\circ$

Oil flow visualization is a good method for gaining insight into flow behavior and comparing numerical and experimental images. The numerical images were generated in ParaView 5.11.2, an open-source, free data visualization tool widely used in the CAE (Computer-Aided Engineering) domain.

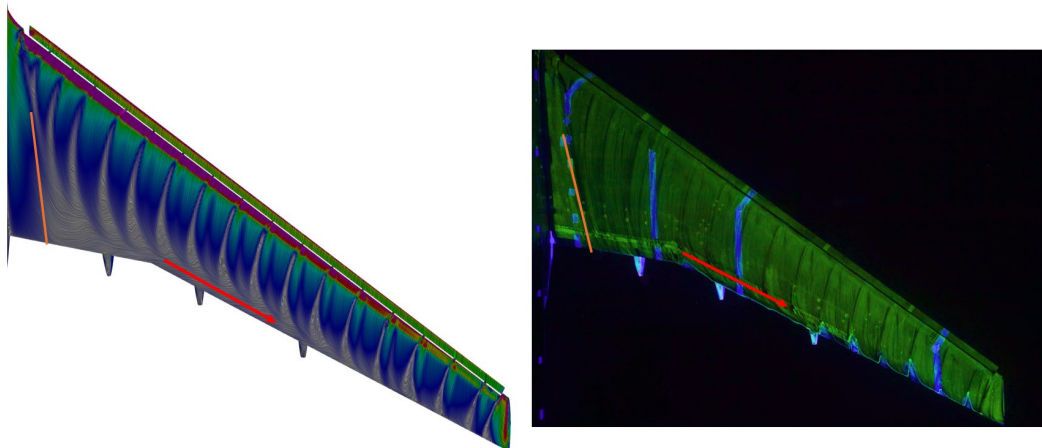
ParaView offers interactive 3D visualization, easy scripting with Python, data extraction, and 2D plotting. It is also parallelized and can efficiently handle cases with hundreds of millions of cells.

For this case, the Surface LIC (Line Integral Convolution) utility was used to visualize the flow field at the wing surface, producing an oil-flow-like image that is shown side by side with a oil flow visualisation image at the $AoA=17.7^\circ$ for both SU2 and OpenFOAM.

In Fig. 14, a side-by-side comparison is shown between SU2 results (LIC representation of wall shear stress and friction coefficient contours) and the experimental image.

The red arrow highlights the high spanwise flow at the trailing edge (TE) in the middle of the wing, while the yellow arrow indicates the flow angle at the wing root. The overall flow pattern is captured correctly, showing the interference from the slat support on the wing. The flow angle at the root is underpredicted by SU2, with the experimental image showing a slightly higher angle.

For the spanwise flow, SU2 now overpredicts compared to the experiment. Overall, the results from SU2 are reasonable.

Fig. 14 SU2 vs experimental at $AoA=17.7^\circ$

In Fig. 15, the results from OpenFOAM show similar trends to those from SU2. There is a stronger secondary interference from the slat braces.

The same observations apply to OpenFOAM: a slight underprediction of the root angle and an little overprediction of the spanwise flow at the midspan.

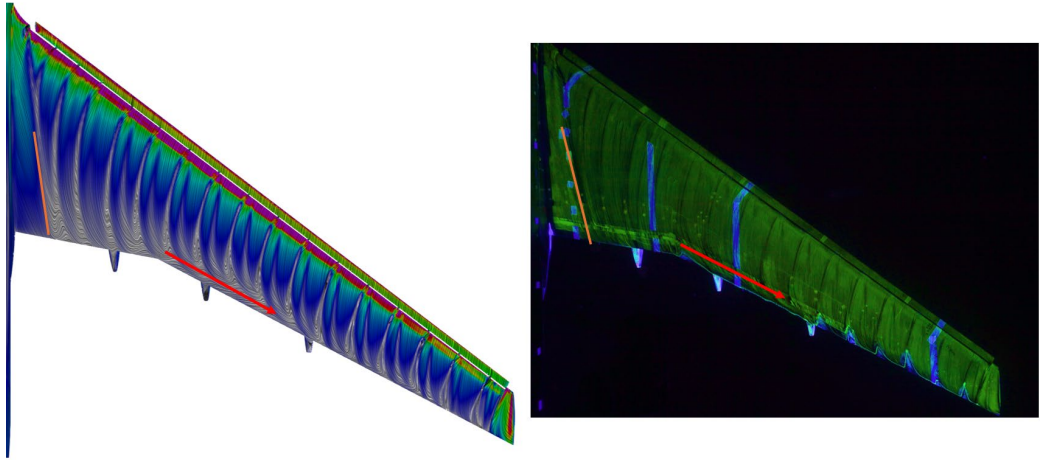


Fig. 15 OpenFOAM vs experimental at AoA=17.7°

5. CONCLUSIONS

Open-source CFD tools, such as OpenFOAM and SU2, offer powerful capabilities for simulating fluid flows and related phenomena. However, their accuracy and reliability must be assessed against known results to ensure they can be trusted for engineering and research applications. The primary goal of this study was to gain hands-on experience with open-source computational fluid dynamics (CFD) software and to validate these tools using HLPW5 cases. These HLPW5 cases provide extended results from both experimental campaign and other CFD solvers, allowing for a direct comparison of results.

In addition with the power of ANSA pre-processor a custom grid family was generated that performed on par with other grid families from the workshop. Practically, a complete loop of engineering applications was covered using the resources available at INCAS. On the results side, they showed similar results to commercial solvers that are widely used in the CFD domain. Although SU2 and OpenFOAM have different approaches to solving RANS, their results were similar and in line with expected outcomes. Finally, the experience gained from this study is important for further expanding our capabilities and resources that can be used by the Aerodynamics and CFD departments.

REFERENCES

- [1] C. B. Sterian Dănăilă, *Metode Numerice în Dinamica Fluidelor*, București: Editura Academiei Române, 2003.
- [2] B. Diskin, Y. Liu and M. Galbraith, High-Fidelity CFD Verification Workshop 2024: Spalart-Allmaras QCR2000-R Turbulence Model, *SciTech*, no. High-Fidelity CFD Verification Preworkshop, 2023.
- [3] E. Foundation, *OpenFOAM: User Guide: rhoSimpleFoam*, ESI Foundation, 2018. [Online]. Available: <https://www.openfoam.com/documentation/guides/latest/doc/guide-applications-solvers-compressible-rhoSimpleFoam.html>. [Accessed 22 9 2025].
- [4] E. Foundation, *OpenFOAM: User Guide: Solver SIMPLE*, ESI Foundation, 2018. [Online]. Available: <https://www.openfoam.com/documentation/guides/latest/doc/guide-applications-solvers-simple.html>. [Accessed 24 09 2025].
- [5] S. Foundation, *SU2-Docs*, 2025. [Online]. Available: https://su2code.github.io/docs_v7/home/. [Accessed 24 09 2025].

- [6] L. Wang, *Geometry Files 5th AIAA High Lift Prediction Workshop*, NASA, 16 09 2024. [Online]. Available: <https://hiliftpw.larc.nasa.gov/Workshop5/geometries.html>. [Accessed 2025 09 2025].
- [7] L. Wang, *Mesh generation Guidelines HLPW4*, NASA, 06 12 2020. [Online]. Available: <https://hiliftpw.larc.nasa.gov/Workshop4/Mesh-Generation-Guidelines-V111-3-Final.pdf>. [Accessed 24 09 2025].
- [8] S. Mouton and G. Charpentier, Testing the Full-Span High-Lift Common Research Model at the ONERA F1 Pressurized Low-Speed Wind Tunnel, *AIAA*, 2024.