Proceedings of the 12th International Conference on Fluid Flow, Heat and Mass Transfer (FFHMT 2025)
July 15, 2025 - July 17, 2025 | Imperial College London Conference, London, United Kingdom
Paper No. 146
DOI: 10.11159/ffhmt25.146

The Curious Case Of A NACA 0012 Airfoil: Are We Learning Something New?

Rasha Al Jahdali[1], Lisandro Dalcin[1], Gianmarco Mengaldo[2], Matteo Parsani[1]
[1] Computer Electrical and Mathematical Science and Engineering Division (CEMSE)
King Abdullah University of Science and Technology (KAUST)
Thuwal, Saudi Arabia
rasha.aljahdali@kaust.edu.sa; dalcinl@gmail.com; matteo.parsani@kaust.edu.sa
[2] Department of Mechanical Engineering
National University of Singapore
Asian Institute of Digital Finance at NUS, Singapore, 9 Engineering Drive 1, 117575 Singapore, Singapore
mpegim@nus.edu.sg

Abstract - Numerical methods and "history-contextualized" high-performance computing have been the cornerstone of computational fluid dynamics (CFD) for the past 70 years, allowing solving of more and more complex models governing flow physics. In CFD, we are at a juncture where it has become possible to use so-called high-fidelity methods to simulate accurate complex flow models, allegedly with high accuracy. The primary method for building and quantifying confidence in modeling and simulation is to verify and validate computational tools. The focus of this paper is on a widely-used test case used for the verification of compressible flow solvers. We report the results for the compressible flow past a 2D NACA 0012 airfoil at Reynolds, Mach, and Prandtl numbers of $Re = 5{,}000$, $Ma = 0.5$, $Pr = 0.72$, at zero angle of attack. This test case is reported to be (laminar) steady in the literature. Is that the case? Is this test case teaching us something new about the large class of numerical discretizations used in this work?

***Keywords*:** Compressible Navier–Stokes equations, NACA 0012 airfoil, Computational Fluid Dynamics, High-fidelity.

1. Introduction

Commonly to many disciplines in science, three main approaches are used for studying and understanding fluid dynamics: experiments, theoretical analysis, and numerical simulations. In particular, the latter Swiss knife is at the core of computational fluid dynamics (CFD).

During the last 70 years, CFD has earned a respectable place alongside the established theoretical and experimental branches of fluid dynamics. In particular, over the previous two decades, CFD has become more powerful than ever because massively parallel computing has come within reach of every research group in academia and industry. Nowadays, CFD is the method that allows the user to perform virtual experiments that would be too expensive, difficult, dangerous, or impossible in the real world. Thus, CFD has the potential to provide unprecedented capabilities to push the boundaries of flow physics modeling, analysis, and design. However, such potential will be met if CFD becomes a truly predictive and high-fidelity computational tool that can handle many challenging applications requiring the treatment of increasingly complex physics in the presence of increasingly complex geometries [1].

Users and developers of computational tools always face a critical issue: How should confidence in modeling and simulation be critically assessed? Verification and validation of computational simulation tools are the primary methods for building and quantifying this confidence. Verification assesses the accuracy of the numerical solution of a model by comparison with known solutions. Validation is the assessment of the accuracy of a computational simulation by comparison with experimental data. In verification, the relationship of the simulation to the real world is not an issue. In validation, the relationship between computation and the real world, i.e., experimental data, is the issue.

During the last decade, with the increasing interest of the research community in the design and application of spatially high-order-of-accuracy discretization methods for CFD problems, there has been an imperative need to extend the verification methodology to this class of methods. This need was one of the main driving forces which led to a series of international workshops and mini-symposia on high-order CFD methods [2]. Early on, among an extensive set of laminar and turbulent test cases, the community considered the two-dimensional (2D) NACA 0012 test case at Reynolds, Mach, and Prandtl numbers of $Re = 5{,}000$, $Ma = 0.5$, $Pr = 0.72$, and zero angle of attack, i.e., $AoA = 0°$. This is one of the test cases proposed by Swanson and Turkel in 1985 [3] and extensively studied in [4] to evaluate an algorithm for solving the steady compressible Navier–Stokes equations.

To the best of our knowledge, since the work of Swanson and Turkel in 1985 [3], all the results published for this test case, including those of the authors of this work, have been obtained using acceleration algorithms (e.g., h- or p-multigrid with Runge–Kutta "smoothers"), time stepping or pseudo time stepping techniques, and choices of parameters (e.g., large-time steps, damping parameters) adequate to try to converge quickly and efficiently to a steady-state solution. These approaches can also "suppress" any potential physical unsteadiness that the flow might have or develop, especially when simulating conditions very close to on-set instability. In this context, we ask our-self two questions: Is the compressible flow past a 2D NACA 0012 $Re = 5{,}000$, $Ma = 0.5$, $Pr = 0.72$, and $AoA = 0°$ truly steady? Can the computation of the numerical solution for this flow with a "time accurate" approach teach us more about the properties of the discretizations being used? Our reflection originates from "unexpected" numerical results obtained simulating this flow – a believed "must-pass" benchmark problem for validating any compressible Navier–Stokes solver with three compressible solvers. In this paper, we present results obtained from the high-order accurate entropy-stable spatial discretizations available in the SSDC [5] framework. Due to page limitations, we do not include similar results obtained with the provably linearly stable high-order accurate spatial discretizations implemented in PyFR [6] and Flexi [7]. We may be deceiving ourselves, but at this point, we find it intriguing to share the results we have observed in our ongoing study.

2. Governing equations

The flow simulated in the work is governed by the unsteady compressible Navier–Stokes with the assumption of a Newtonian working fluid. These equations are cast into the following compact conservation form:

$$\partial_t \mathbf{U} + \partial_j (\mathbf{F}_{\text{ivc}} + \mathbf{F}_{\text{vsc}}) = 0 \tag{1}$$

where $\mathbf{U} \in \mathbb{R}^{1\times\mathcal{N}}$ is the solution-vector, $\mathbf{F}_{\text{ivc}} \in \mathbb{R}^{\mathcal{D}\times\mathcal{N}}$ and $\mathbf{F}_{\text{vsc}} \in \mathbb{R}^{\mathcal{D}\times\mathcal{N}}$ are the inviscid and viscous flux vectors, respectively. $\mathcal{N}$ is the number of solution-variables and $\mathcal{D}$ is the spatial dimension. These quantities are defined as:

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho v_1 \\ \rho v_2 \\ \rho v_3 \\ \rho E \end{bmatrix}, \tag{2a}$$

$$\mathbf{F}_{\text{ivc}} = \begin{bmatrix} \rho v_1 & \rho v_2 & \rho v_3 \\ p + \rho v_1 v_1 & \rho v_1 v_2 & \rho v_1 v_3 \\ \rho v_2 v_1 & p + \rho v_2 v_2 & \rho v_2 v_3 \\ \rho v_3 v_1 & \rho v_3 v_2 & p + \rho v_3 v_3 \\ \rho v_1 H & \rho v_2 H & \rho v_3 H \end{bmatrix}, \; \mathbf{F}_{\text{vsc}} = \begin{bmatrix} 0 & 0 & 0 \\ -\tau_{11} & -\tau_{12} & -\tau_{13} \\ -\tau_{21} & -\tau_{22} & -\tau_{23} \\ -\tau_{31} & -\tau_{32} & -\tau_{33} \\ -v_i \tau_{i1} - \omega_1 & -v_i \tau_{i2} - \omega_2 & -v_i \tau_{i3} - \omega_3 \end{bmatrix}. \tag{2b}$$

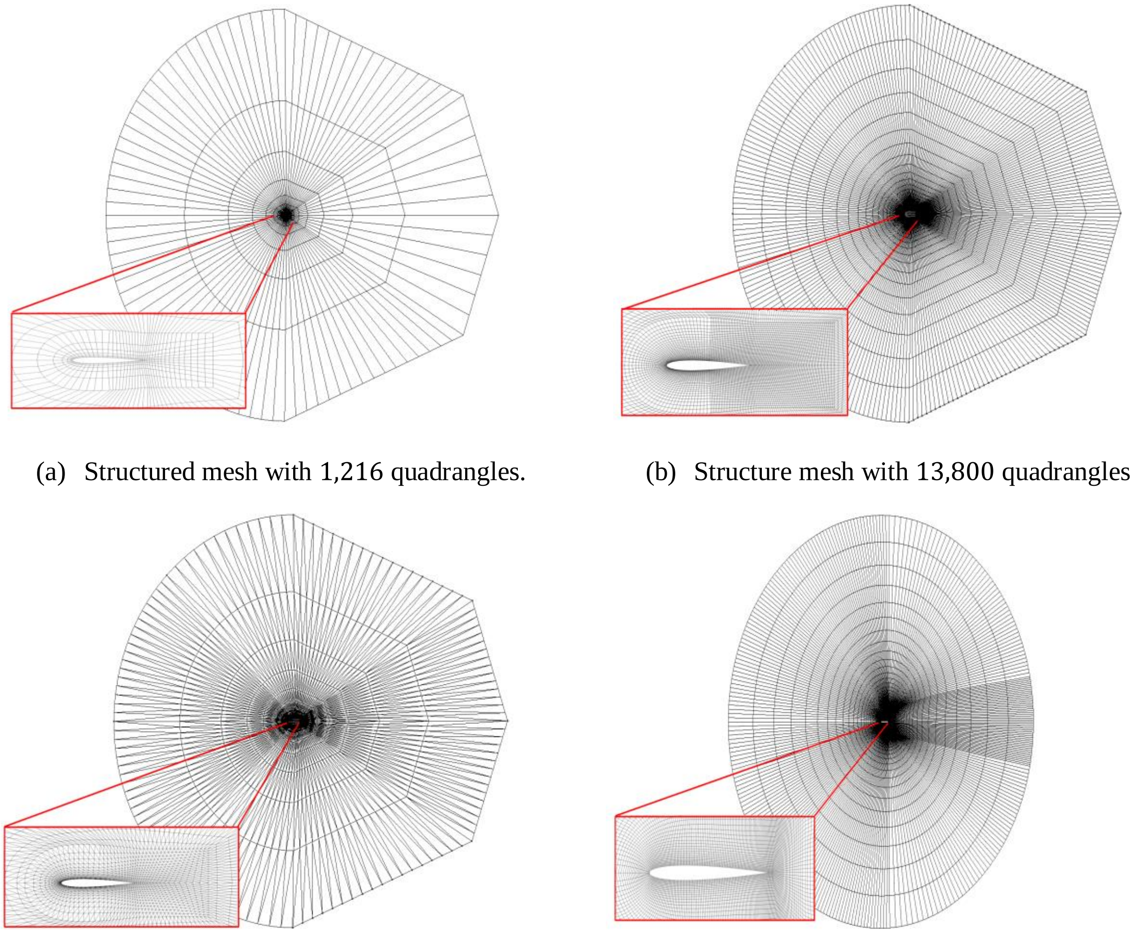
Einstein summation convention is used for repeated indices i. The symbols used in the equations above are as follows. ρ is the density, $\mathbf{v} = \mathbf{e}_i v_i$ is the velocity vector with $\mathbf{e}_i$ being the i-th orthonormal basis vector of Euclidean space, E is the total energy per unit mass, i.e., $E = e + \frac{1}{2} v_i v_i$ where e is the internal energy. For a calorically perfect gas, $e = \frac{RT}{\gamma - 1}$ where R is the gas constant and T is the temperature determined as $T = \gamma \mathrm{M}^2 p/\rho$. The total enthalpy H is defined as $H = E + \frac{p}{\rho}$ where p is the pressure, related to energy through the ideal gas law: $p = \rho(\gamma - 1)\left(E - \frac{1}{2} v_i v_i\right)$ where γ is the specific heat ratio. τ_{ij} are the components of the viscous stress tensor $\underline{\underline{\tau}}$. For compressible Newtonian fluids, we have $\tau_{ij} = 2\mu S_{ij}$ and $S_{ij} =$

$\frac{1}{2}\left(\partial_i v_j + \partial_j v_i\right) - \frac{1}{3}\partial_k v_k \delta_{ij}$, where μ is the dynamic viscosity which is assumed to be constant. $\omega_j = \lambda\,\partial_j T$ is the j-th component of the heat flux vector where $\lambda = \frac{\gamma R}{\gamma - 1}\frac{\mu}{\mathrm{Pr}}$ is the molecular conductivity.

3. Results and discussion

We solve the system of Equations (1) with the definitions (2) to investigate the flow around a 2D NACA0012 profile at a Reynolds' number of $Re = 5{,}000$, Mach's number of $Ma = 0.5$, and Prandtl number of $Pr = 0.72$. The angle of attack is $AoA = 0°$. In this preliminary work, we perform the time advancement flow problem using the 3(2) pair of explicit Runge–Kutta scheme of Bogacki and Shampine (3BS scheme), and the first-order backward difference methods, also known as backward Euler schemes.

We use four structured grids with different element densities and element types, all refined smoothly near the airfoil trailing edge and wake. The grids are generated using Gmsh [9]; smooth quadratic and cubic elements are used to represent the airfoil geometry. Figure 1 reports a visual view of these grids. Because of the page limit and to avoid unnecessary repetitions, below, we report solely the results computed using the entropy stable SSDC solver [5]. However, analogous results have been obtained using the provably linearly stable high-order accurate algorithms implemented in the PyFR [6] and Flexi [7] solvers.



(a) Structured mesh with 1,216 quadrangles.

(b) Structure mesh with 13,800 quadrangles.

(c) Structure mesh with 5,904 triangles.

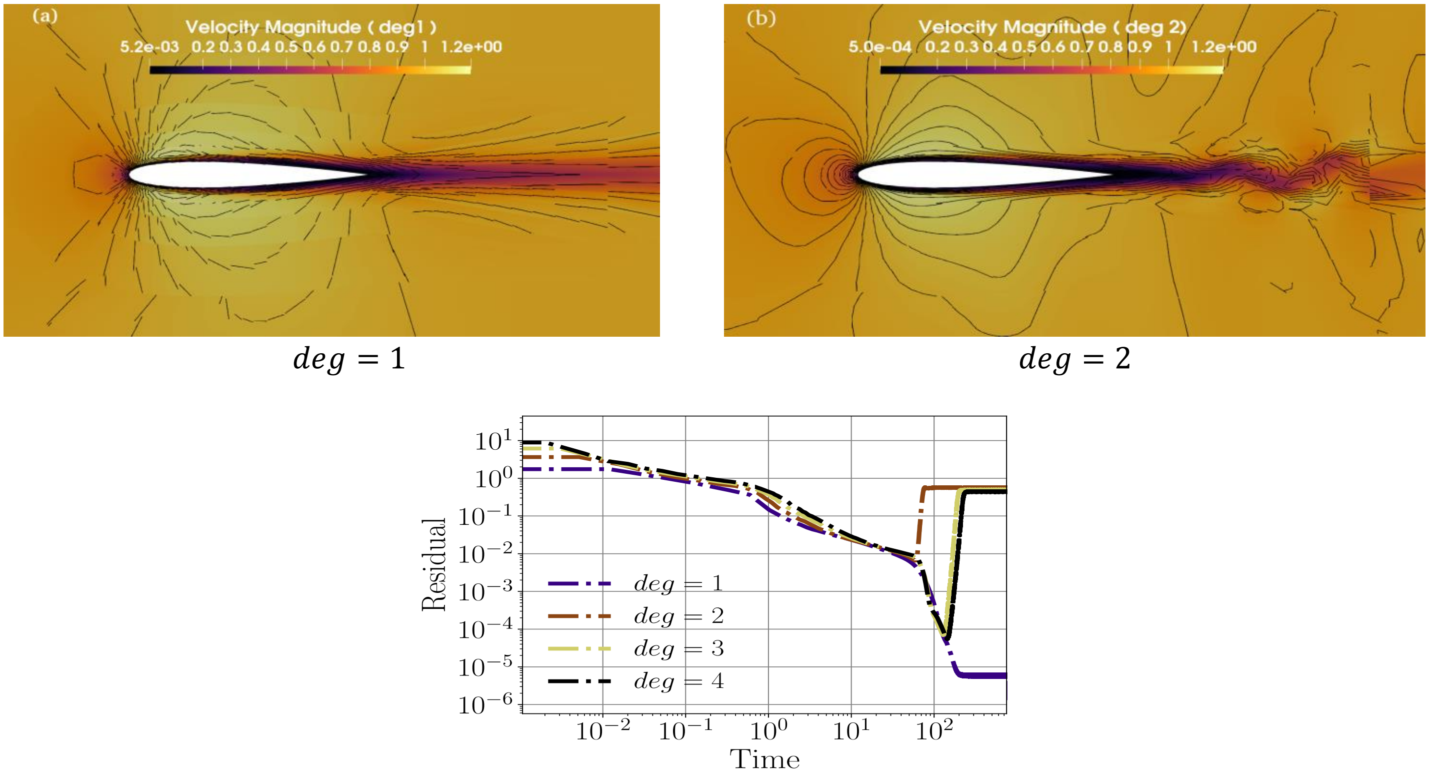
(d) Structure mesh with 15,274 quadrangles.

Fig. 1: Overview of the four grids.

To study the convergence to a steady-state solution, we present the time evolution of the residual, i.e., L^2-norm of the term $\partial_j(\mathbf{F}_{\text{ivc}} + \mathbf{F}_{\text{vsc}})$ in Equation (1) for the momentum in the horizontal direction, i.e., the direction of the incoming flow upstream of the NACA0012 airfoil. Note that for a steady-state solution, all the components of the residual vector for the conserved variable vector should be zero, or, for a steady solution computed numerically, the residual components should be close to machine precision.

Solid wall no-slip wall boundary conditions are imposed on the airfoil surface. Uniform laminar flow corresponding to $Re = 5{,}000$, $Ma = 0.5$, and $Pr = 0.72$ is imposed at the incoming flow. We use a “sponge zone” approach to suppress spurious reflections at the outlet boundary. The initial condition is also set to a uniform flow. All the simulations are performed in double (machine) precision.

We perform the simulations using four solution polynomial degrees: $deg = 1, 2, 3,$ and 4, corresponding to second-, third-, fourth-, and fifth-order accurate spatial discretizations. Figures 2a and 2b show the iso-contours of the velocity magnitude for the solutions computed with $deg = 1$ and $deg = 2$ using the grid with 1,216 quadrangle elements (see Figure 1a). The first results are computed using the 3BS scheme for time advancement. The time evolution of the residual for all the solution polynomial degrees used is shown in Figure 2c. The solution for $deg = 1$ in Figure 2a appears laminar and steady. However, upon inspection of Figure 2c, we observe that the residual associated to $deg = 1$ decreases monotonically and then stagnates right below 10^{-5}. It remains like that for the rest of the simulation. Thus, the solution computed with $deg = 1$ cannot be considered a converged steady-state solution. Furthermore, we tested the same setup for $deg = 1$ using two levels of uniform mesh refinement and the residual behaves in the same way stagnating monotonically around 10^{-5}, and no pronounced unsteadiness/asymmetry in the wake can be detected. In contrast, for all the other degrees used, i.e., $deg \geq 2$, we observe that the residual decreases, and then after a small oscillation, it rapidly increases until it stagnates just below a value of 10^0, i.e., $O(1)$. The attained value of the residual is practically the same for all the degrees $deg \geq 2$. The unsteadiness in the solution is visible in the wake in Figure 2b. To conclude this first part of the results, we highlight behavior discrepancies among the same family of spatial discretizations in this first instance. It seems that for $deg \geq 2$, some physical or numerically-driven instability appears and triggers the residual growth, leading to a visible unstable wake. Could this indicate a flow on the verge of becoming (laminar) unsteady? Or is this and an indication that the schemes with $deg \geq 2$ in SSDC [5], PyFR [6], and Flexi [7] are exhibiting some unexpected behavior? An ongoing detailed analysis should be able to provide more information, and by the time of the presentation of this work, we should be able to provide complete or almost complete answers to these questions.



$deg = 1$

$deg = 2$

Residual for $deg = 1,2,3$, and 4.

Fig. 2: Iso-contours of the velocity magnitude for solution polynomial degree (a) $deg = 1$, and (b) $deg = 2$ computed using the 3BS scheme for time advancement. History of the residual in the wind (i.e., horizontal) direction for all solution polynomial degrees. The simulations ran on the 1,216 quadrangle elements (see Figure 1a).

We now turn our attention to the behavior of the numerical solution when we use different the four grids with different mesh resolutions and element types shown in Figure 1. We use again the 3BS scheme for time advancement. Figure 3 shows the residual history using $deg = 2$ for all the four grids. Also, in this case, the residual decreases, and then after a small oscillation, it rapidly increases until it stagnates just below a value of 10^0, i.e., $O(1)$, leading to an unsteady wake, as shown in the iso-contour of the velocity magnitude in Figure 3. The described residual behavior is apparent for all the grids, but for that one tagged as "mesh3" in Figure 3. This grid corresponds to the mesh shown in Figure 1c, composed of triangular elements. The residual obtained using the "mesh3" grid decreases monotonically and plateau around a value of 10^{-1}, one order of magnitude lower than the value reached using the other grids. Nevertheless, the numerically computed solution is not a converged steady-state solution. It is important to note that if we use the same problem settings and perform a simulation using the "mesh3" grid and one level of uniform mesh refinement, the residual behaves like the residual of the other three meshes, i.e., it decreases, experiences a quick oscillation, then grows and reaches approximately the value 10^0 and plateaus there.

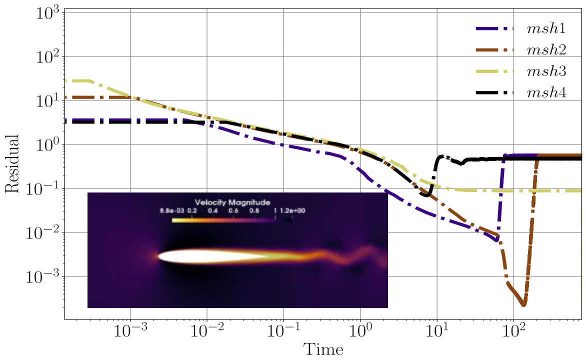


Fig. 3: History of the residual in the wind (i.e., horizontal) direction for $deg = 2$ solution polynomial degree computed with the 3BS scheme and the four grids shown in Figure 1.

To conclude the sets of results, we now focus on the behavior of the algorithms/solvers and solution when using the backward Euler (BE) scheme with a relatively "small" and a relatively "large" time step that is $d\Delta t = 1e - 3$ and $\Delta t = 1e - 1$, respectively. For this test, we use the structured grid with the highest density of elements shown in Figure 1d.

The last results we want to discuss are shown in Figure 4. This figure contains the history of the residual in the wind (i.e., horizontal) direction for $deg = 2$ solution polynomial degree and the contour plots of the velocity magnitude in two different solution states.

The curve of the residual is a composition of different curves obtained using different algorithmic setups. The first part of the curve (i.e., the black dash-dot portion) is obtained by starting from a uniform flow condition and advancing the time using the BE scheme with a time step of $\Delta t = 1e - 3$. We observe that the residual behaves like the results shown in Figures 2c and 3, that is, it decreases, and then after a small oscillation, it rapidly increases until it stagnates just below a value of 10^0, i.e., $O(1)$. At that point, we restart the simulations using the same spatial and temporal discretizations and a larger time step, i.e., $\Delta t = 1e - 1$. The brown dash-dot line illustrates the behavior of the residual as a function of time. We observe that rapidly, the value decreases and plateaus near 10^{-12}, practically zero machine precision. A residual value of 10^{-12} is the lowest value in double precision we can achieve with our problem setup, computing hardware, and any acceleration technique such as h- and p-multigrid. Thus, if the solution is steady-state, at this level, the BE scheme used as a smoother should have removed any error. The solution obtained at this point is a "converged" and steady-state solution, as shown in the first contour plot of the velocity magnitude tagged with (a) in Figure 4 and corresponding to the residual point indicated with a red X. This solution has been extensively published in the literature since the work of Swanson and Turkel in 1985 [3]. The right panel of Figure 5 shows the corresponding lift and drag coefficients.

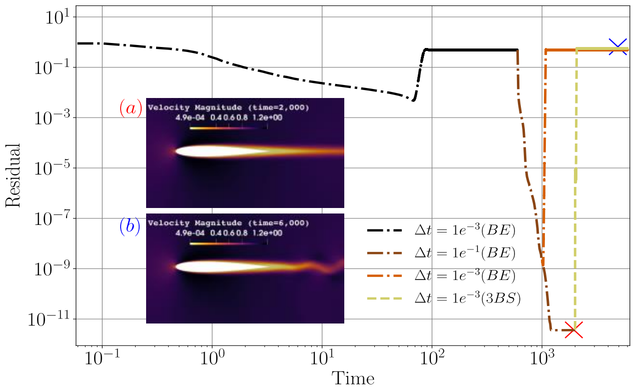


Fig. 4: History of the residual in the wind (i.e., horizontal) direction for $deg = 2$ solution polynomial degree computed with the BE scheme and the grid shown in Figure 1d.

The steady-state values of these two coefficients are close to those published in the literature. An additional mesh refinement would yield values very close to those indicated in [4]. However, this is not the main objective of this short overview of our results.

The brown line is intercepted by an orange dahs-dot line, which, for a moment, we neglect to focus on the next segment of the residual curve, indicated with the yellow dashed line. This yellow portion shows what happens when we restart the simulation with the 3BS scheme using the largest admissible time step (admissible for stability). A similar behavior can be observed by restarting with a smaller time step of the order 10^{-8}. We observe that the residual grows rapidly and plateaus at the same level attained by the BE when run with a $\Delta t = 1e - 3$. This state corresponds to an unsteady solution as shown in the second contour plot of the velocity magnitude tagged with (b) in Figure 4. The left panel of Figure 5 shows the corresponding lift and drag coefficients, which are not constant but time-dependent. In particular, the lift coefficient is symmetric concerning the horizontal axis, which defines the zero value, i.e., its average value. Nevertheless, as expected, their variation is very small.

Finally, we can focus on the orange portion of the curve. This part is obtained by restarting the simulation with the BE scheme and using a time step of $\Delta = 1e - 3$, i.e., what we consider a “small” time step for this problem. The solution state from which we restarted corresponds to a value of the residual approximately equal to 5×10^{-10}. As for the case represented by the yellow curve, the residual rapidly increases and plateaus close to the value characteristics of the black and yellow lines, leading to an unsteady solution.

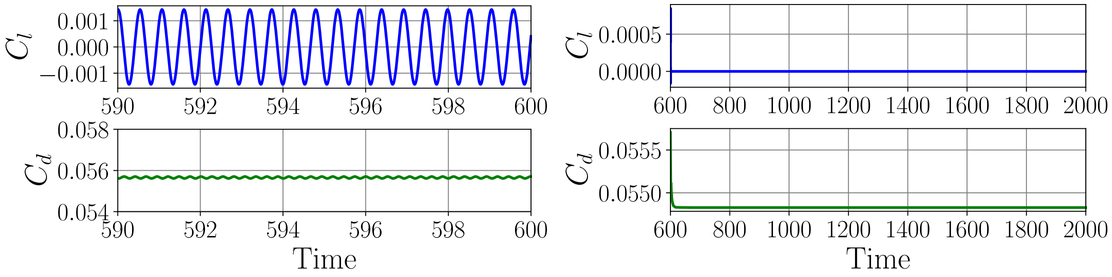


Fig. 5: Lift and drag coefficients C_l and C_d.

4. Conclusion

In this work, we have presented some of the results of an ongoing analysis that focuses on a geometrically simple yet interesting compressible laminar flow around a 2D NACA 0012 airfoil at $Re = 5{,}000$, $Ma = 0.5$, $Pr = 0.72$, and $AoA = 0°$. This flow problem has been widely used for a long time to assess the accuracy and reliability of compressible solvers for both low- and high-order accurate CFD solvers. The results presented in this work have been validated using three fully independent compressible flow frameworks, namely SSDC [5], PyFR [6], and Flexi [7]. However, we only presented the results computed with the SSDC solver due to page limitations. The findings have led us to ask ourselves the following questions: Is the compressible flow past a 2D NACA 0012 $Re = 5{,}000$, $Ma = 0.5$, $Pr = 0.72$, and $AoA = 0°$ truly steady? Can the computation of the numerical solution for this flow with a "time accurate" approach teach us more about the properties of the discretizations being used? In the near future, we should be able to provide some answers to these questions.

Acknowledgements

This work was funded by King Abdullah University of Science & Technology (KAUST).

References

[1] J. P. Slotnick, A. Khodadoust, J. Alonso, D. Darmofal, W. Gropp, E. Lurie, D. J. Mavriplis, "CFD Vision 2030 Study: A Path to Revolutionary Computational Aerosciences", NASA Contractor Report, no. NASA/CR–2014-218178, 2014.

[2] NASA Glenn Research Center, https://www1.grc.nasa.gov/research-and-engineering/hiocfd/, 2017.

[3] R. C. Swanson and E. Turkel "A multistage time-stepping scheme for the Navier–Stokes equations.", 23rd Aerospace Sciences Meeting, AIAA Paper 85-0035, 1985.

[4] R. C. Swanson, "Comparison of NACA 0012 Laminar Flow Solutions: Structured and Unstructured", NASA Technical Memorandum, no. NASA/TM–2016–219003, 2016.

[5] M. Parsani, R. Boukharfane, I. R. Nolasco, D. C. Del Rey Fernández, S. Zampini, B. Hadri, and L. Dalcin, "High-order accurate entropy-stable discontinuous collocated Galerkin methods with the summation-by-parts property for compressible CFD frameworks: Scalable SSDC algorithms and flow solver", Journal of Computational Physics, vol. 424, pp. 109844, 2021.

[6] F.D. Witherden, A.M. Farrington, and P.E. Vincent, "PyFR: An open source framework for solving advection–diffusion type problems on streaming architectures using the flux reconstruction approach", Computer Physics Communications, vol. 185, no. 11, pp. 3028-3040, 2014.

[7] N. Krais, A. Beck, T. Bolemann, H. Frank, D. Flad, G. Gassner, F. Hindenlang, M. Hoffmann, T. Kuhn, M. Sonntag, and C.-D. Munz, "FLEXI: A high order discontinuous Galerkin framework for hyperbolic-parabolic conservation laws", Computers & Mathematics with Applications, vol. 81, pp. 186-219, 2021.

[8] C. Geuzaine and J.-F. Remacle. "Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities", International Journal for Numerical Methods in Engineering, vol. 79, no. 11, pp. 1309-1331, 2009.