



## City Research Online

### City, University of London Institutional Repository

---

**Citation:** Kassem, H. I., Liu, X. & Banerjee, J. R. (2016). Transonic flutter analysis using a fully coupled density based solver for inviscid flow. *Advances in Engineering Software*, 95, pp. 1-6. doi: 10.1016/j.advengsoft.2016.01.012

This is the accepted version of the paper.

This version of the publication may differ from the final published version.

---

**Permanent repository link:** <https://openaccess.city.ac.uk/id/eprint/14245/>

**Link to published version:** <https://doi.org/10.1016/j.advengsoft.2016.01.012>

**Copyright:** City Research Online aims to make research outputs of City, University of London available to a wider audience. Copyright and Moral Rights remain with the author(s) and/or copyright holders. URLs from City Research Online may be freely distributed and linked to.

**Reuse:** Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge. Provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

---

---

---

City Research Online:

<http://openaccess.city.ac.uk/>

[publications@city.ac.uk](mailto:publications@city.ac.uk)

---

# Transonic flutter analysis using a fully coupled density based solver for inviscid flow

H.I. Kassem, X. Liu and J.R. Banerjee

## Abstract

This paper focuses on the coupling between the high fidelity aerodynamic model for the flow field with the modal analysis of a typical wing section to carry out flutter analysis. This coupled aeroelastic model is implemented in one of the most widely used open source CFD codes called OpenFOAM. The model is designed to calculate the structural displacement in the time domain based on the free vibration modes of the structure by constructing the numerical model directly from the modal analysis. Essentially a second order ordinary differential equation is solved for each mode as a function of the generalized coordinates. A density based solver using central difference scheme of Kurganov and Tadmor is used to model the flow field. Two main cases of transonic flow over NACA 64A010 are modelled for a forced pitching oscillation airfoil and a self-sustained aerofoil respectively. The self-sustained two degrees of freedom case is modelled for three different possibilities covering damped, neutral and divergent oscillations. Predicted results show very good agreement with the numerical and experimental data available in the literature.

**Keywords:** Aeroelasticity, CFD, transonic flow, flutter.

## 1 Introduction

Aeroelasticity is the science of studying the interaction between three main forces namely; elastic, inertia and aerodynamics. Therefore aeroelasticity is an interdisciplinary field combining; fluid mechanics, solid mechanics and structural dynamics. In general, the interaction between these two or three areas is classified as aeroelastic problems. Aeroelastic research started in the late 1920's and the subject matter has matured enormously over the years and now there are many excellent texts on the subject [1, 2, 3, 4]. Insufficient or inaccurate prediction of aeroelastic characteristics of aircraft during the design process can lead to catastrophic incidents.

One of the most dangerous aeroelastic instabilities is, of course flutter. It is a self-excited oscillation of elastic body in fluid stream. This condition is usually defined by two important parameters namely the flutter speed and the flutter frequency. Flutter

speed defines the speed beyond which the aircraft becomes unstable. It means that if the aircraft flies at this speed it will have steady harmonic oscillation of constant amplitude. This self-excited oscillation will have a frequency which is called the flutter frequency. This point is the most critical point because if for any reason, free stream velocity exceeds the flutter speed, the system will have divergent oscillation and will eventually vibrate in a violent way which could lead to the destruction of the aircraft. The aeroelastic phenomenon, flutter is due to all three types of forces, namely elastic, inertia and aerodynamics. The fluid flow instead of playing its natural role to damp the structural vibration, it will feed the system instead with more and more energy until divergent oscillation occurs. The complexity of flutter analysis arises from the fact that flutter involves very strong coupling between fluid mechanics and structural dynamics. Therefore an accurate description of the flow field as well as structural dynamic behaviour together with a mechanism of coupling between the two is essential for flutter analysis.

Avoiding flutter is a mandatory requirement in any aircraft design process. Although flutter analysis is a relatively old problem in aviation, but it is still challenging, particularly with the advent of composite materials and requirement of high speeds. The main challenge for this problem is at the transonic flow region. The transonic flutter limit appears to be low in any flight range. Therefore for an aircraft the most critical flutter point generally arises when the flow is transonic. The phenomenon is called transonic dip which has featured in the literature many times [5, 6]. The transonic flow field is a transition between subsonic flow and supersonic flow exhibiting shock waves and highly non-linear behaviour.

The transonic flow being non-linear poses a great challenge over traditional linear theories [7] which fail to predict accurately the aerodynamic properties. Therefore solving the non-linear governing equations of fluid flow using numerical techniques has become essential [8, 9, 10, 3]. Despite the computational cost of using CFD, it is necessarily being used in the aeroelasticity field for greater accuracy and better flutter prediction. This has given birth to a new field in aeroelasticity called computational aeroelasticity which couples computational fluid dynamics (CFD) with computational structural dynamics (CSD) [11].

In the next section a concise theoretical background is given focusing on the governing equations of the aeroelastic system. Then the numerical methods and the implemented code are explained. Finally, the results of the two validation cases are discussed in detail. This paper is based on an earlier paper [12] but with some enhancement. The essential improvement in this paper appears in the results of the first case study which is improved considerably compared to the previous work. This improvement is mainly due to some refinement in the convergence criteria and better boundary condition for the slip moving wall.

Following the publication of the conference paper by the authors [12], an updated version of the software OpenFOAM-2.3 has now been used [13]. The newer version introduced many improvements, particularly in parallel running performance and the implementation of a new dynamic mesh solver. Also another important improvement

in this release is the inclusion of an enhanced ordinary differential equation solver library which is directly relevant to the present work [14]. Due these this modifications some of the implemented features by the authors have been updated in this paper.

## 2 Theoretical Background

### 2.1 Aerodynamic Model

The governing equations of the flow are the complete Euler equations [15, 16, 17]. If  $\rho$ ,  $\mathbf{u}$ ,  $p$  and  $\mathbf{E}$  are density, velocity, pressure and total energy respectively, the Euler equations in vector notation will then have the following form;

- Conservation of mass:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot [\mathbf{u}\rho] = 0 \quad (1)$$

- Conservation of momentum:

$$\frac{\partial(\rho\mathbf{u})}{\partial t} + \nabla \cdot [\mathbf{u}(\rho\mathbf{u})] + \nabla p = 0 \quad (2)$$

- Conservation of total energy:

$$\frac{\partial(\rho\mathbf{E})}{\partial t} + \nabla \cdot [\mathbf{u}(\rho\mathbf{E})] + \nabla \cdot [\mathbf{u}p] = 0 \quad (3)$$

where  $\nabla$  is the nabla vector operator,  $\nabla \equiv \partial_i \equiv \frac{\partial}{\partial x_i} \equiv (\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \frac{\partial}{\partial x_3})$ . Thus for any vector  $\mathbf{a}$ ,  $\nabla \cdot \mathbf{a}$  is the divergence defined by  $\nabla \cdot \mathbf{a} \equiv \frac{\partial a_1}{\partial x_1} + \frac{\partial a_2}{\partial x_2} + \frac{\partial a_3}{\partial x_3}$ . Also for any scalar  $s$ , the gradient is  $\nabla s \equiv (\frac{\partial s}{\partial x_1}, \frac{\partial s}{\partial x_2}, \frac{\partial s}{\partial x_3})$ . In equation (3),  $\mathbf{E} = e + \frac{|\mathbf{u}|^2}{2}$  with  $e$  the specific internal energy.

### 2.2 Aeroelastic Model

The typical wing section using two-dimensional model[4, 1, 3] is well established for studying two degrees of freedom wing dynamical system. This model considers the plunging ( $h$ ) and pitching ( $\alpha$ ) motions about the elastic axis of the wing. The governing equations of undamped motion are [18]:

$$m\ddot{h} + S_\alpha\ddot{\alpha} + K_h h = -L \quad (4)$$

$$S_\alpha\ddot{h} + I_\alpha\ddot{\alpha} + K_\alpha\alpha = M_{ea} \quad (5)$$

where  $m$ ,  $I_\alpha$  and  $S_\alpha$  are aerofoil mass per unit length, section moment of inertia about the elastic axis per unit length and static mass imbalance respectively.  $K_h$  and  $K_\alpha$  are bending and torsional spring stiffness whereas  $L$  and  $M_{ea}$  are the lift force (positive

up) and moment about the elastic axis (positive nose up). The plunging displacement  $h$  is positive down and the angle of attack  $\alpha$  is positive nose up and is in radians. Non-dimensionalizing the linear displacement by the aerofoil semichord ( $b$ ) in equations (4) and (5) and the time by the uncoupled natural frequency of the torsional spring ( $\omega_\alpha$ ) so that the dimensionless time is  $\tau = \omega_\alpha t$ . The governing equations (4) and (5) can now be reformulated in the following matrix form

$$[M]\{\ddot{q}\} + [K]\{q\} = \{F\} \quad (6)$$

where

$$[M] = \begin{bmatrix} 1 & x_\alpha \\ x_\alpha & r_\alpha^2 \end{bmatrix}; \quad [K] = \begin{bmatrix} (\frac{\omega_h}{\omega_\alpha})^2 & 0 \\ 0 & r_\alpha^2 \end{bmatrix} \quad (7)$$

$$\{F\} = \frac{U_\infty^2}{\pi\mu\omega_\alpha^2 b^2} \begin{Bmatrix} -C_l \\ C_m \end{Bmatrix}; \quad \{q\} = \begin{Bmatrix} \frac{h}{b} \\ \alpha \end{Bmatrix} \quad (8)$$

In equation (6),  $[M]$  and  $[K]$  are the mass and stiffness matrices, and  $\{F\}$  and  $\{q\}$  are the force and displacement vectors. The non-dimensional aerofoil mass ratio is  $\mu = \frac{m}{\pi\rho b^2}$  with  $x_\alpha$  and  $r_\alpha$  being the static unbalance and the radius of gyration respectively. The uncoupled natural frequencies in plunging and pitching motion are  $\omega_h$  and  $\omega_\alpha$ , respectively.  $C_l$  and  $C_m$  represent the lift and moment coefficients which have the same sign convention as the aerodynamic forces and moment  $L$  and  $M$ .

### 2.3 Modal Analysis

The main objective now is to solve equation (6) which represents the aerofoil motion in two degrees of freedom namely the heave and pitch. In order to solve the equations the modal analysis methodology is used. The main concept is representing the system displacements as a linear combination of the free vibration mode shapes through the use of generalized coordinates. In general if a combination of the first few modes of free vibration say  $N$  is used, then according to modal approach the displacement vector can be represented as

$$\{q\} = [\phi]\{\eta\} \quad (9)$$

where  $[\phi]$  is the modal matrix in which each column is an eigenvector of the free vibration analysis eigen-problem and  $\{\eta\}$  is the generalized coordinates. Premultiplying equation (6) by  $[\phi]^T$  and substituting using (9) and applying the eigenvectors orthogonality lead to a set of second order ordinary differential equations in generalized coordinates. Each equation is represented by its mode, say  $i^{th}$  mode [18, 19] to give

$$\ddot{\eta}_i + 2\zeta_i\omega_i\dot{\eta}_i + \omega_i^2\eta_i = Q_i; \quad i = 1, 2, \dots, N \quad (10)$$

where

$$Q_i = \{\phi\}_i^T \{F\} \quad (11)$$

$$\omega_i^2 = \{\phi\}_i^T [K] \{\phi\}_i \quad (12)$$

$$1 = \{\phi\}_i^T [M] \{\phi\}_i \quad (13)$$

and  $\zeta_i$  in equation (10) is modal damping which is not considered in (6). The modes are normalized in a way such that the generalized mass matrix became an identity matrix. In this report the structural system is considered as an undamped system. However, the damping is shown in equation (10) just for reference and showing how the system damping can be considered in the future work.

It is clear from the above equations that to calculate the system displacement vector from equation (9), modal matrix  $[\phi]$  and the generalized coordinates vector  $\{\eta\}$  should be obtained first. Determining the first  $N$  modes to formulate the modal matrix  $[\phi]$  can be accomplished by solving the eigen-problem for the free vibration system. This particular case has only two modes because the system is discrete with two degrees of freedom only. Then to get the generalized displacement vector  $\{\eta\}$ , equation (10) should be solved. It is a second order ordinary differential equation (ODE) in time. Here, it will be solved using numerical integration in time by Runge-Kutta scheme. In order to solve it, equation (10) should be reduced to two first order (ODE) in  $y_{1i}$  and  $y_{2i}$  using the transformation  $y_{1i} = \eta_i$  and  $y_{2i} = \dot{\eta}_i$  which leads to

$$\dot{y}_{1i} = y_{2i} \quad (14)$$

$$\dot{y}_{2i} = Q_i - 2\zeta_i\omega_i y_{2i} - \omega_i^2 y_{1i} \quad (15)$$

The system of equations (14) and (15) should be solved for each mode  $i$ . It is an initial value problem and therefore the initial conditions for  $y_{1i}$ ,  $y_{2i}$ ,  $\dot{y}_{1i}$  and  $\dot{y}_{2i}$  will be specified from the initial values of the generalized coordinates.

The general initial conditions are:

$$h(0) = h_0; \quad \alpha(0) = \alpha_0 \quad (16)$$

$$\dot{h}(0) = \dot{h}_0; \quad \dot{\alpha}(0) = \dot{\alpha}_0 \quad (17)$$

$$\{\eta_0\} = [\phi]^{-1} \{q_0\} \quad (18)$$

$$\{\dot{\eta}_0\} = [\phi]^{-1} \{\dot{q}_0\} \quad (19)$$

## 2.4 Fluid Structure Coupling

As mentioned before, closely coupled interaction is considered in this study. Two levels of coupling are required for which the first one is essentially time coupling

carried out by integrating the aerodynamic forces over the aerofoil at every time step to calculate the force vector  $\{F\}$ . The second level of interaction is coupling between the structural displacements and the fluid solver. For the case in hand where the aerofoil cross section is considered to be rigid (non-deformable), the aerofoil position will be updated at every time step according to the calculated  $C_l$  and  $C_m$ . By knowing  $h$  and  $\alpha$  from equation (9) the new location  $P_1$  for point  $P_0$  on the aerofoil is obtained from

$$\{P_1\} = [R]\{P_0\} + \{h\} \quad (20)$$

where  $\{h\}$  is the displacement vector in the plunging direction and  $[R]$  is the rotation matrix involving an angle  $\alpha$  around the elastic axis. For an aerofoil in the  $xy$ -plane, the rotation matrix by an angle  $\alpha$  in radian around a unit vector in the  $z$  direction through the elastic axis is

$$[R] = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (21)$$

### 3 Numerical Scheme and Background Information

The previous section was a general introduction about the mathematical foundation of the aeroelastic problem of a typical aerofoil section. This model is well established as an educational tool in the literature as a representative model for studying the stability and response problem of wings [1, 3]. In this section, the current implementation and the other implemented models for aeroelastic study will be introduced and discussed. This section will also highlight the main challenges of modelling fluid-structure interaction.

There are three main elements of the current problem. First, solving the governing equations of the fluid flow using the finite volume method. The governing equations will be solved numerically for a finite number of control volumes representing the flow domain (discretization). Computational fluid dynamics techniques involve pre-processing stages for creating a mesh and defining the boundary and initial conditions, which is followed by the solving stage when iterative numerical algorithms are used and finally the post-processing of the result takes place. The second element in this problem is the structural model which has already been mentioned in detail in section 2.2. The third element is coupling between structure and fluid which was outlined in section 2.4. In this section, the main aspects of these elements will be discussed in greater details including a description of the developed code.

#### 3.1 The Fluid Solver "rhoCentralFoam"

The main purpose of this work is to predict the transonic flutter. In this regime the flow is highly non-linear and unsteady. Moving and oscillating shock waves are the dominant features of transonic flow field. In order to predict such complex flow field

with high fidelity, a special technique should be applied to solve the governing equations outlined in section 2.1. In general there are two main approaches to solve these equations in CFD. These are essentially either pressure based solver or density based solver. The main difference between them is that the latter solves the continuity equation as a function of density, directly coupled with the rest of governing equations. This is in contrast to the pressure based solver which solves a pressure correction equation which is derived from the momentum and continuity equations [16]. This pressure correction works as a constraint on the velocity field to satisfy the continuity equation. Therefore it is called segregated solver.

Both approaches are available in OpenFOAM for high speed compressible flow. The pressure based solver called sonicFoam and the density based solver called rhoCentralFoam. The advantages and disadvantages of each method are well known amongst the CFD community [15, 16]. The main advantage for pressure based solver is that it requires less computational resources than the density based solver due to the segregation between the governing equations. The obvious advantage of density based solver is the coupling between the governing equations which leads to better non-oscillating solution, specially when discontinuities are involved due to shock waves. In this report only rhoCentralFoam solver is used because of its advantages over sonicFoam for transonic flow under consideration. The implementation of rhoCentralFoam reveals some of the advantages of Riemann solver [17]. A full comparison between the implemented two methods in OpenFOAM showed better results of rhoCentralFoam solver over sonicFoam in different high speed compressible flow cases [20]. The density based solver, rhoCentralFoam uses central difference schemes based on Kurganov and Tadmor formulation introduced in 2000 [21]. It was implemented in OpenFOAM by Greenshields et al. in 2009 [17]. It is a semi-discrete, non-staggered central scheme.

## 3.2 Dynamic Mesh

Solving a particular case involves a moving solid object requiring some special strategy to include this movement. In this respect, finite volume methods are usually used for solving fluid dynamics governing equations at fixed cells in space (control volumes). When the solid objects start to move there will be a relative velocity between the boundaries and the mesh cells. There are two approaches to solve this problem. The first approach relies on calculating the movement of the mesh according to its boundary displacement but maintaining the same number of grid cells. The second approach is to calculate the new position of each grid cell with the possibility of removing or adding new cells as required. These two techniques already implemented in OpenFOAM are particularly useful [22, 23].

In this study, the first approach is used which basically solves Laplace equation for the grid displacement at every time step [22, 23]. A diffusion coefficient for the mesh movement is the only parameter that should be specified by the user. Before describing the governing equation of moving grid it is useful to examine the main dif-

ferences between static and dynamic mesh. Basically it is the relative speed between the boundary and the mesh which has a direct relation with the flux through each finite volume cell. Ignoring this relative speed could lead to numerical error in the solution. Preventing this numerical problem requires applying the space conservation law (SCL) which states [22, 15]

$$\frac{d}{dt} \int_V dV - \oint_S n \cdot v_s dS = 0 \quad (22)$$

where  $V$  is an arbitrary moving volume,  $n$  is the unit vector normal to the surface and  $v_s$  is the surface speed. The above condition applied in OpenFOAM solvers by a function called `makeRelative`. In OpenFOAM the name of the solvers which are capable of handling dynamic meshes includes "DyM". For example the variant of `rhoCentralFoam` solver which is used in this study for dynamic mesh is `rhoCentralDyMFoam`. Now attention is turned to the Laplace displacement mesh motion solver in OpenFOAM, which solves for independent displacement vector  $d$  defined by

$$r(t + \Delta t) = r(t) + d \quad (23)$$

where  $r$  is the point position vector. Thus Laplace equation for mesh motions with  $k$  as diffusion coefficient is

$$\nabla \cdot (k \nabla d) = 0 \quad (24)$$

Equations (22) and (24) illustrate the main difference between static mesh solvers and dynamic mesh solvers in OpenFOAM. Also a special boundary condition for moving walls velocity associated with dynamic mesh solver has to be used. It is called `movingWallVelocity` which makes the normal flux to the wall equal to zero. Also there are other dynamic mesh solvers in OpenFOAM which solve for the grid points velocity [22, 24].

In order to couple the structural dynamics with the fluid flow solver, a new boundary condition is developed called `elasticDisplacement`. The main function of `elasticDisplacement` boundary condition is to calculate the force coefficients over the aerofoil and calculate the corresponding displacement according to the free vibration natural modes of the system.

## 4 Results and discussion

In this section a number of different cases will be investigated to test the source code. A wide range of operating conditions is modelled to display the potential of the method presented in this study.

### 4.1 Case A: Pitching NACA 64A010 Aerofoil

The first test case (case A) is a pitching aerofoil about the quarter chord in transonic flow free stream. The aerofoil section is NACA 64A010. This particular case is one

of the widely used cases to validate transonic CFD codes. The experimental work was carried out by Davis [25], Alonso et al. [18] as well as by Chen et al. [26] in order to validate their CFD codes. Table 1 gives the operating conditions for this case.

Description	Variable	Value
Aerofoil		NACA 64A010
Mean angle of attack	$\alpha_m$	$0^\circ$
Angle of attack amplitude	$\alpha_A$	$\pm 1.01^\circ$
Free stream Mach number	$M_\infty$	0.8
Reynolds number	$Re$	$1.256 \times 10^7$
Reduced frequency	$k$	0.202
Pitch axis from leading edge	$x_p$	25% of chord

Table 1: Characteristics of test case A.

In this case GMSH has been used [27] instead of using one of OpenFOAM meshing utilities. GMSH has a graphical user interface which gives more control, and thus accelerates the mesh generation process. Figures 1 and 2 show the complete mesh and the mesh around the sharp trailing edge respectively. The computational domain is  $15c \times 10c$  with 39,006 grid cells.

Figure 3 shows the lift coefficient versus the angle of attack. The results from this work are in good agreement with the experimental results [25]. Although this figure shows an excellent agreement along the pitching cycle, it also shows that the model did not predict very well the peak points. In general these results are comparable with the results reported in literature [18, 26]. McMullen et al.[28] modelled this case with a grid independent study and also reported under and over predictions for the lift coefficients. Obviously, increasing the grid quality will increase the accuracy but it is not the main reason for these differences. It is probably due to ignoring the viscous effect for a streamer-line objects like aerofoils. In such cases the forces arising from shear stress may have a noticeable contribution. More investigations using different grids and turbulence models may clarify and pin-point the reason. Figure 4 shows the Mach contours at the maximum angle of attack.

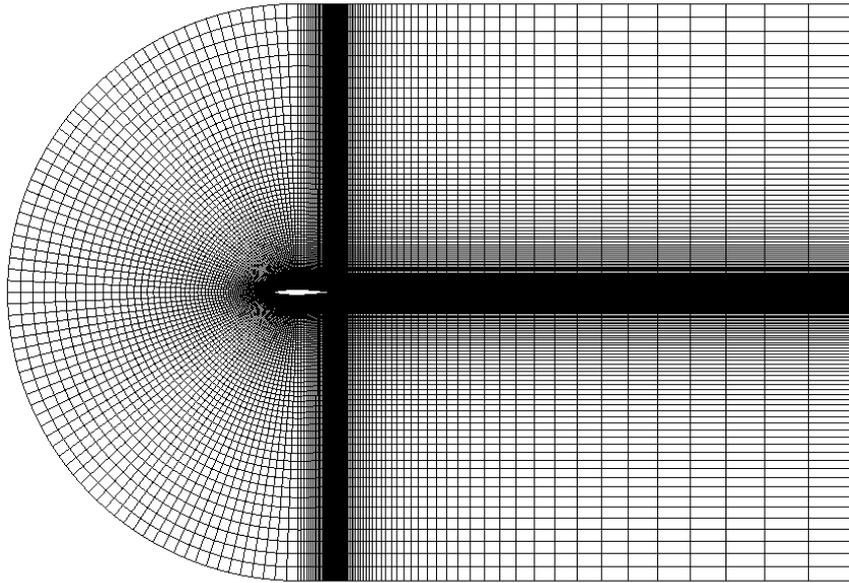


Figure 1: C-mesh type around NACA 64A010

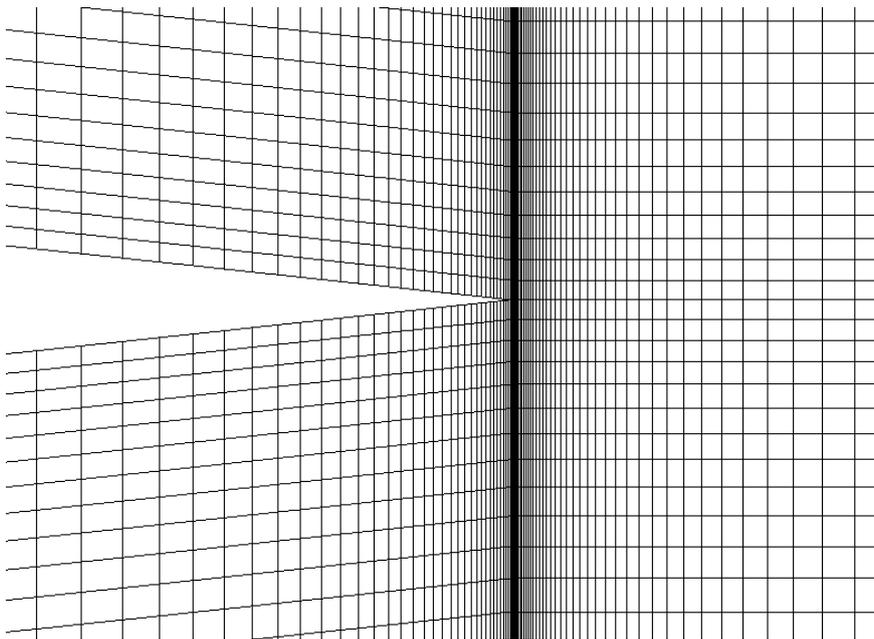


Figure 2: Mesh around NACA 64A010 tail

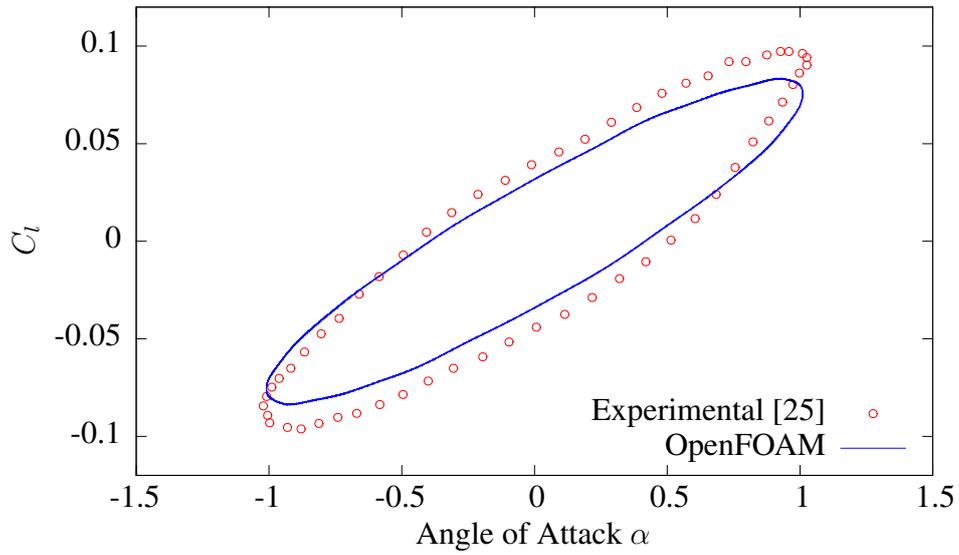


Figure 3: Instantaneous lift Coefficient

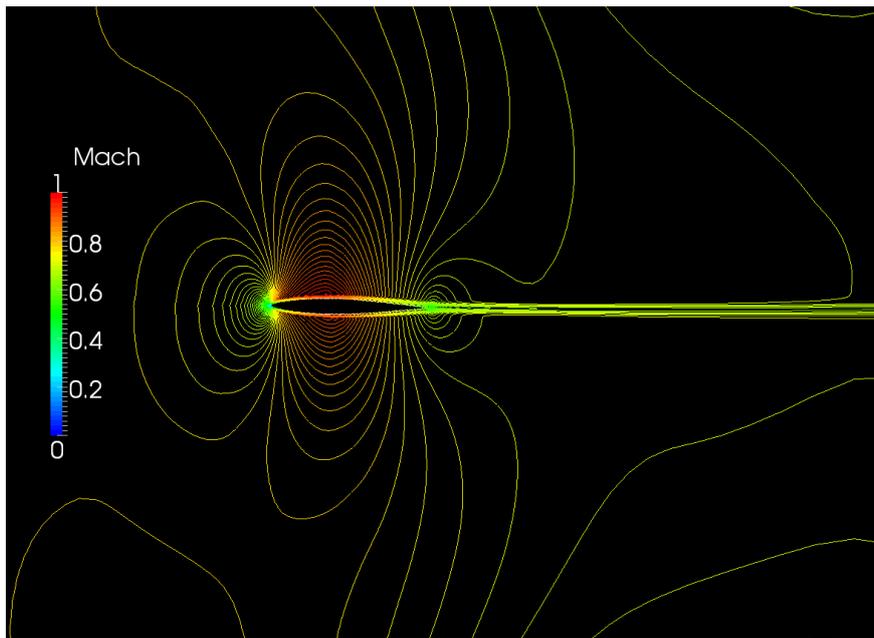


Figure 4: Mach Contours at  $\alpha = 1.01^\circ$

## 4.2 Case B: Self-Sustained NACA 64A010

In this case, the modal analysis was used to calculate the aerofoil displacement. Again the NACA 64A010 was used as in case A. However, three different operating conditions are modelled for this case [18]. The elasticDisplacement boundary condition was used. Table 2 shows the selected operating conditions. The structural model fol-

lows the one which was introduced by Isogai [5, 6]. The modelling for each condition was done using three stages, namely, fixed aerofoil, pitching aerofoil around the elastic axis and finally self-sustained aerofoil. The same mesh from case A was used to save computational time. A fifth-order Runge-Kutta with adaptive time step developed by Cash and Karp [29] was selected. It is one of the OpenFOAM ODE solvers for non-stiff systems.

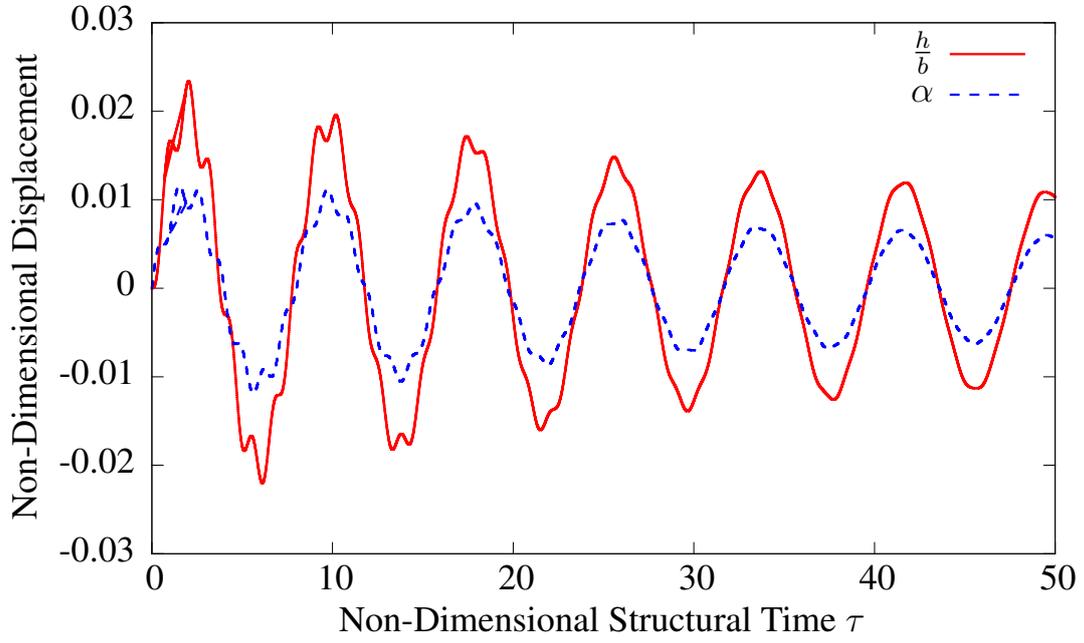
Description	Variable	Value
Aerofoil		NACA 64A010
Mean angle of attack	$\alpha_m$	$0^\circ$
Angle of attack amplitude	$\alpha_A$	$\pm 1.01^\circ$
Free stream Mach number	$M_\infty$	0.85, 0.825, 0.875
Speed Index	$V^* = \frac{U_\infty}{\omega_\alpha b \sqrt{\mu}}$	0.439, 0.612, 1.420
Aerofoil mass ratio	$\mu$	60
Reynolds number	$Re$	$1.256 \times 10^7$
Static unbalance	$x_\alpha$	1.8
Squared radius of gyration	$r_\alpha^2$	3.48
uncoupled natural freq. in plunge	$\omega_h$	100rad/s
uncoupled natural freq. in pitch	$\omega_\alpha$	100rad/s
Pitch axis from leading edge	$x_p$	-50% of chord

Table 2: Characteristics of test case B.

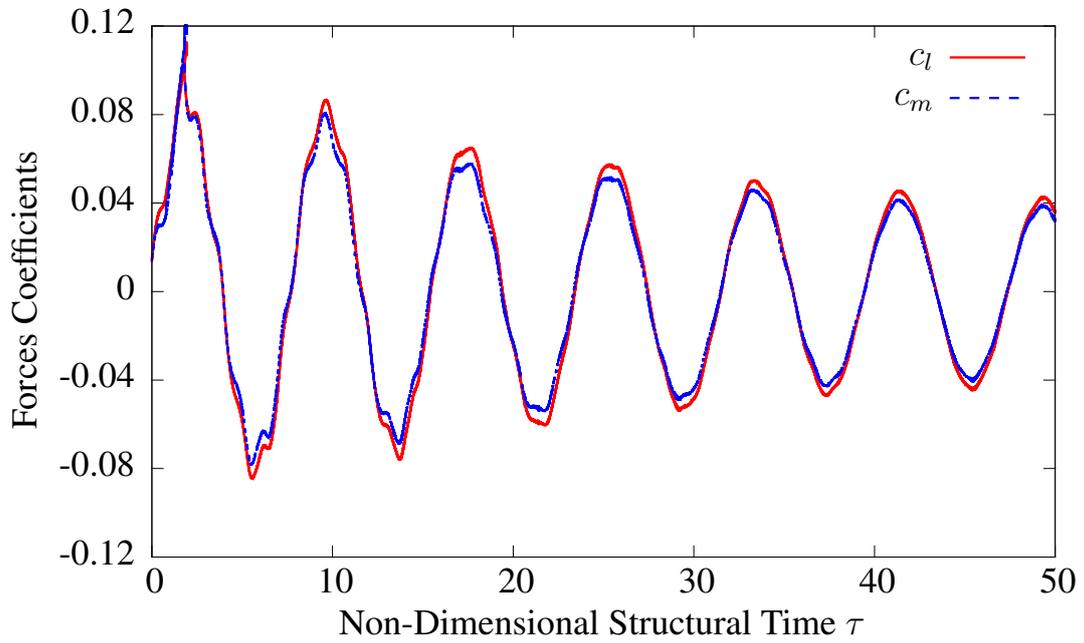
Figures 5, 6 and 7 show the responses and the forces for the three operating conditions. It is clear that Figure 5 represents a damped response, whereas Figure 7 shows a divergent response. Both are in very good agreement with [18, 26]. It was expected that Figure 6 would predict the flutter point as reported by Alonso et al. [18], but as it turned out the flutter point was missed only by a small margin. Nevertheless, the trend to predict the flutter speed is sufficiently clear.

## 5 Conclusions

In this paper the main aspects of computational aeroelasticity are discussed. The newly implemented code in OpenFOAM for coupling the fluid-structure interaction based on free vibration natural modes of an oscillating aerofoil is highlighted. One case for forced pitching aerofoil has been investigated and the predicted results are compared with experimental measurements from the literature. A second case for self-sustained aerofoil based on the newly developed code has also been studied. All these case studies are verified. Results from the implemented code showed good agreement with experimental data and numerical predictions found in the literature.

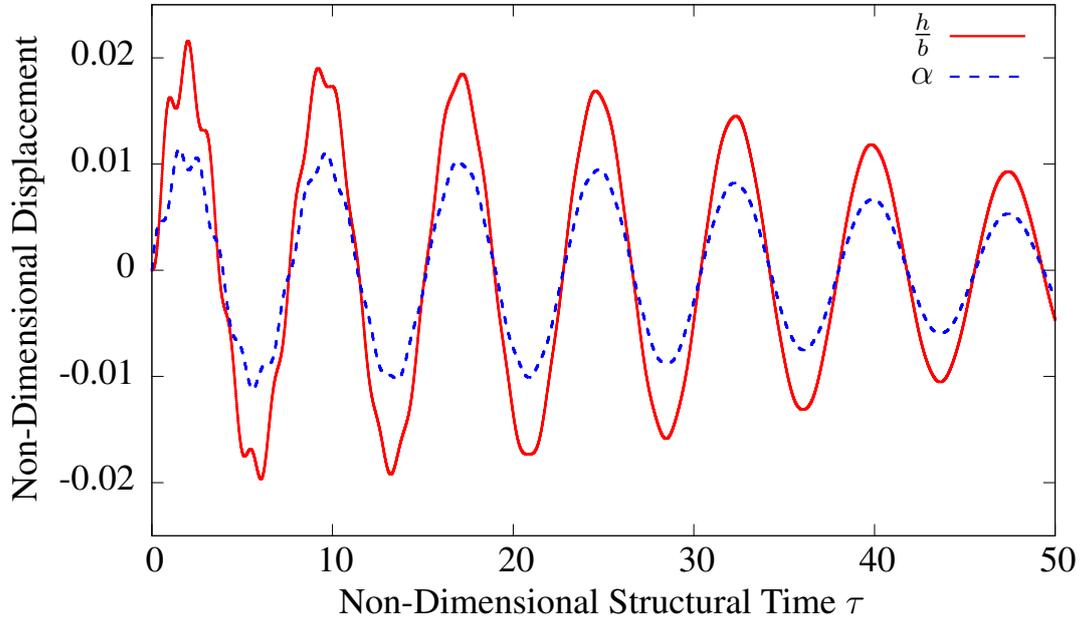


(a) Displacements.

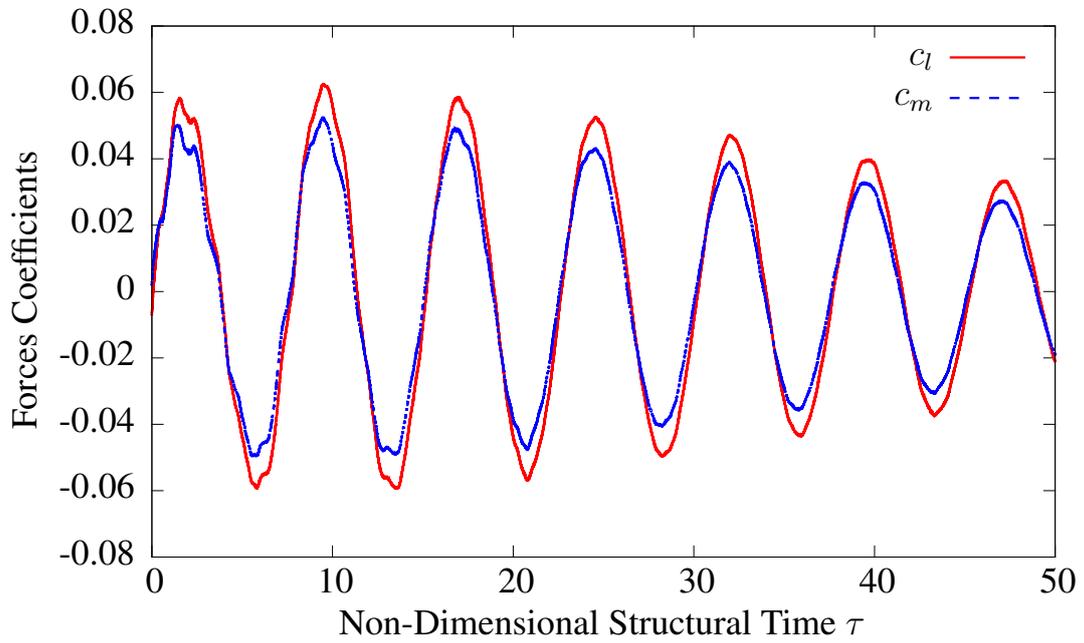


(b) Forces Coefficients.

Figure 5: Damped Response.  $M_\infty = 0.85$ ,  $V^* = 0.439$

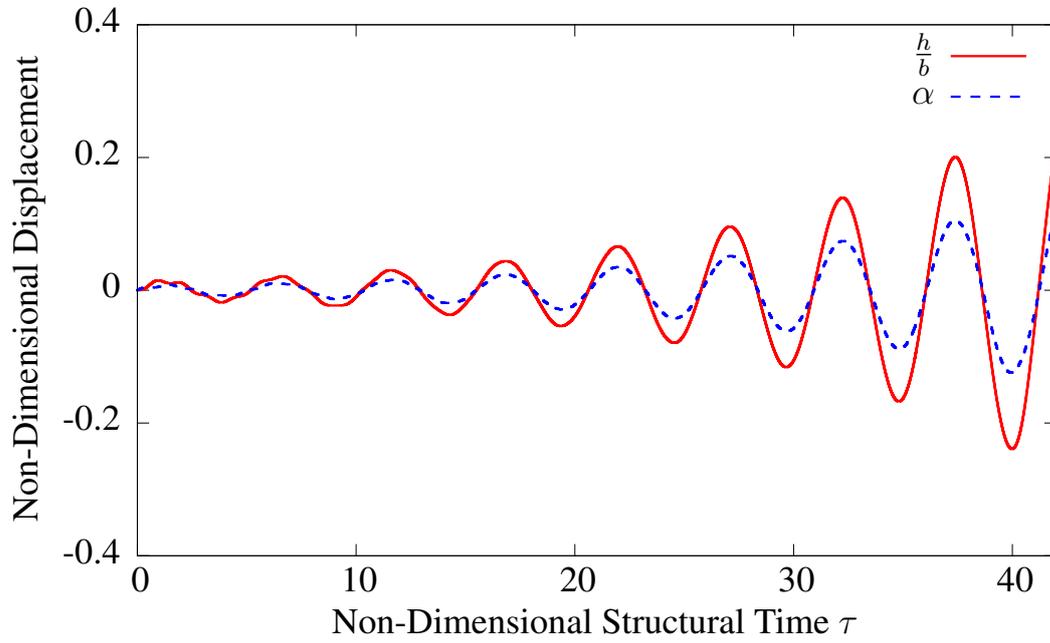


(a) Displacements.

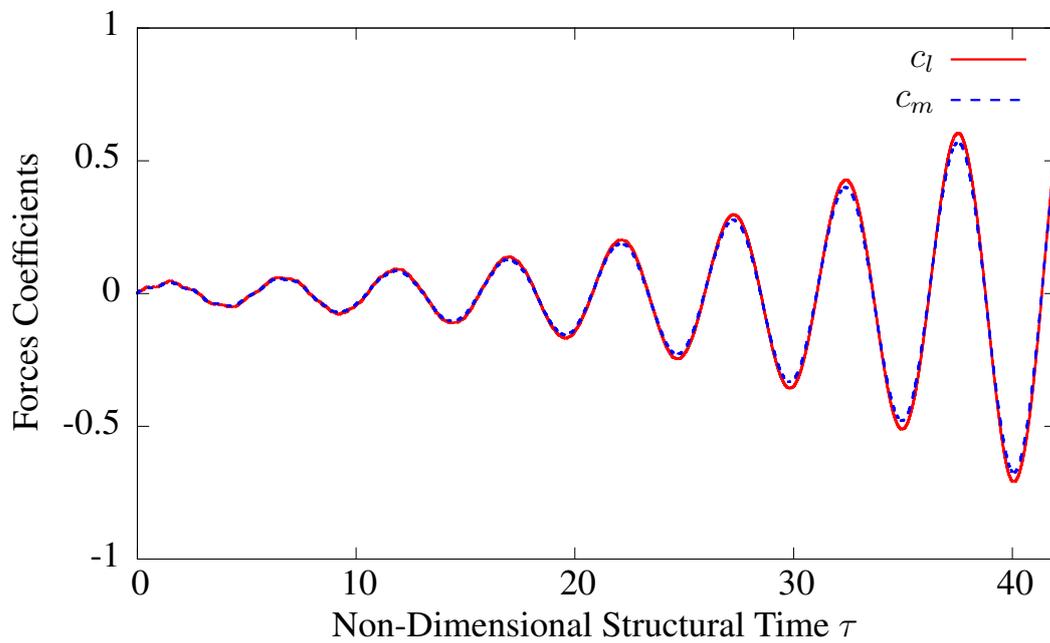


(b) Forces Coefficients.

Figure 6: Damped Response.  $M_\infty = 0.825$ ,  $V^* = 0.612$



(a) Displacements.



(b) Forces Coefficients.

Figure 7: Divergent Response.  $M_\infty = 0.875$ ,  $V^* = 1.420$

## References

- [1] R. Bisplinghoff, H. Ashley, R. Halfman, *Aeroelasticity*, Dover Books on Aeronautical Engineering Series. Dover Publications, 1996.
- [2] Y.C. Fung, *An Introduction to the Theory of Aeroelasticity*, Dover Phoenix Edition: Engineering. Dover Publications, Incorporated, 2002.
- [3] R. Clark, D. Cox, H. Curtiss, E. Dowell, J. Edwards, K. Hall, D. Peters, R. Scanlan, E. Simiu, F. Sisto, *A Modern Course in Aeroelasticity*, Solid Mechanics and Its Applications. Kluwer Academic Publishers, 2006.
- [4] W. Rodden, *Theoretical and Computational Aeroelasticity*, Crest Publishing, 2011.
- [5] K. Isogai, “On the Transonic-Dip Mechanism of Flutter of a Sweptback Wing”, *AIAA Journal*, 17(7): 793–795, July 1979.
- [6] K. Isogai, “Transonic dip mechanism of flutter of a sweptback wing. II”, *AIAA Journal*, 19(9): 1240–1242, Sept. 1981.
- [7] O.O. Bendiksen, “Review of unsteady transonic aerodynamics: Theory and applications”, *Progress in Aerospace Sciences*, 47(2): 135–167, Feb. 2011.
- [8] O.O. Bendiksen, G. Seber, “Fluidstructure interactions with both structural and fluid nonlinearities”, *Journal of Sound and Vibration*, 315(3): 664–684, Aug. 2008.
- [9] R.M. Bennett, J.W. Edwards, “An overview of recent developments in computational aeroelasticity”, *AIAA paper*, (98-2421), 1998.
- [10] G.P. Guruswamy, “A review of numerical fluids/structures interface methods for computations using high-fidelity equations”, *Computers & structures*, 80(1): 31–41, 2002.
- [11] D.M. Schuster, D.D. Liu, L.J. Huttshell, “Computational aeroelasticity: success, progress, challenge”, *Journal of Aircraft*, 40(5): 843–856, 2003.
- [12] H.I. Kassem, X. Liu, J.R. Banerjee, “Flutter Analysis using a Fully Coupled Density Based Solver for Inviscid Flow”, in B.H.V. Topping, P. Ivnyi, (Editors), “*Proceedings of the Twelfth International Conference on Computational Structures Technology*”. Civil-Comp Press, Stirlingshire, UK, Paper 146, 2014. doi:10.4203/ccp.106.146
- [13] “OpenFOAM-2.3.0 user guide”, February 2014.
- [14] “OpenFOAM-2.3.0 release notes”, February 2014. <http://www.openfoam.org/version2.3.0/index.php>
- [15] J.H. Ferziger, M. Perić, *Computational methods for fluid dynamics*, Volume 3, Springer Berlin, 1996.
- [16] H.K. Versteeg, W. Malalasekera, *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*, Pearson Education Limited, 2007.
- [17] C.J. Greenshields, H.G. Weller, L. Gasparini, J.M. Reese, “Implementation of semi-discrete, non-staggered central schemes in a colocated, polyhedral, finite volume framework, for high-speed viscous flows”, *International journal for numerical methods in fluids*, 63(1): 1–21, 2010.
- [18] J. Alonso, A. Jameson, “Fully-implicit time-marching aeroelastic solutions”,

- AIAA paper*, (94-0056), 1994.
- [19] F. Liu, J. Cai, Y. Zhu, H.M. Tsai, A.S. F. Wong, “Calculation of Wing Flutter by a Coupled Fluid-Structure Method”, *Journal of Aircraft*, 38(2): 334–342, Mar. 2001.
  - [20] L.F.G. Marcantoni, J.P. Tamagno, S.A. Elaskar, “HIGH SPEED FLOW SIMULATION USING OPENFOAM”, in *Mecánica Computacional Vol XXXI*, pages 2936–2959. Asociación Argentina de Mecánica Computacional, 2012.
  - [21] A. Kurganov, E. Tadmor, “New High-Resolution Central Schemes for Nonlinear Conservation Laws and Convection-Diffusion Equations”, *Journal of Computational Physics*, 160(1): 241–282, 2000.
  - [22] H. Jasak, Z. Tukovic, “Automatic mesh motion for the unstructured finite volume method”, *Transactions of FAMENA*, 30(2): 1–20, 2006.
  - [23] H. Jasak, “Dynamic Mesh Handling in OpenFOAM”, *43rd AIAA Aerospace Sciences Meeting and Exhibit*, pages 1–10, 2008.
  - [24] H. Jasak, A. Jemcov, Z. Tukovic, “OpenFOAM: A C++ library for complex physics simulations”, in *International Workshop on Coupled Methods in Numerical Dynamics, IUC, Dubrovnik, Croatia*, pages 1–20, 2007.
  - [25] S.S. Davis, “NACA 64A010 (NASA Ames Model) oscillatory pitching”, *AGARD Report*, 702, 1982.
  - [26] X. Chen, G. Zha, Z. Hu, M.T. Yang, “Flutter Prediction Based on Fully Coupled Fluid-Structural Interactions”, in *9th National Turbine Engine High Cycle Fatigue Conference*, 2004.
  - [27] C. Geuzaine, J.F. Remacle, “Gmsh: A 3-D finite element mesh generator with built-in pre-and post-processing facilities”, *International Journal for Numerical Methods in Engineering*, 79(11): 1309–1331, 2009.
  - [28] M. McMullen, A. Jameson, J. Alonso, “Application of a non-linear frequency domain solver to the Euler and Navier-Stokes equations”, *AIAA paper*, 120, 2002.
  - [29] J.R. Cash, A.H. Karp, “A variable order Runge-Kutta method for initial value problems with rapidly varying right-hand sides”, *ACM Transactions on Mathematical Software (TOMS)*, 16(3): 201–222, 1990.