Journal of Applied Fluid Mechanics, Vol. 12, No. 2, pp. 539-549, 2019.
Available online at www.jafmonline.net, ISSN 1735-3572, EISSN 1735-3645.
DOI: 10.29252/jafm.12.02.29156

A Navier-Stokes Solver for Compressible Turbulent Flows on Quadtree and Octree Based Cartesian Grids

E. Kara[1†], A. İ. Kutlar[1] and M. H. Aksel[2]

[1] *University of Gaziantep, Gaziantep, 27310, Turkey*
[2] *Middle East Technical University, Ankara, 06800, Turkey*

†*Corresponding Author Email: emrekara@gantep.edu.tr*

(Received May 1, 2018; accepted October 12, 2018)

ABSTRACT

Cartesian grids represent a special extent in unstructured grid literature. They employ chiefly created algorithms to produce automatic meshing while simulating flows around complex geometries without considering shape of the bodies. In this article, firstly, it is intended to produce regionally developed Cartesian meshes for two dimensional and three dimensional, disordered geometries to provide solutions hierarchically. Secondly, accurate results for turbulent flows are developed by finite volume solver (GeULER-NaTURe) with both geometric and solution adaptations. As a result, a "hands-off" flow solver based on Cartesian grids as the preprocessor is performed using object-oriented programming. Spalart-Allmaras turbulence model added Reynolds Averaged Navier Stokes equations are solved for the flows around airfoils and wings. The solutions are validated and verified by one two dimensional and one three dimensional turbulent flow common test cases in literature. Both case studies disclose the efficaciousness of the developed codes and qualify in convergence and accuracy.

Keywords: Cartesian grid generation; Finite volume solver; Turbulent flows; Object-oriented programming; RANS equations; Spalart-Allmaras (SA) turbulence model.

NOMENCLATURE

A	surface area	c_{v2}	a constant in Spalart-Allmaras model
C_d	drag coefficient	c_{v3}	a constant in Spalart-Allmaras model
C_l	lift coefficient	c_{w1}	a constant in Spalart-Allmaras model
C_p	specific heat at constant pressure	c_{w2}	a constant in Spalart-Allmaras model
C_{suth}	sutherland constant	c_{w3}	a constant in Spalart-Allmaras model
De	destruction term	d_{wall}	distance to wall in Spalart-Allmaras model
Di	diffusive term		
F	inviscid flux term	e	specific internal energy
G	viscous flux term	f_{v1}	a function in Spalart-Allmaras model
E	total energy	f_{v2}	a function in Spalart-Allmaras model
H	total time step	f_w	a function in Spalart-Allmaras model
M	Mach number	g	a function in Spalart-Allmaras model
M_∞	Mach number at reference state	g_n	a function in Spalart-Allmaras model
Q	conserved variable	h	semi-height of the tunnel
P	production term	n	normal vector
Pr	Prandtl number	p	static pressure
Re_a	Reynolds number based on speed of sound	q	diffusive flux
		r	a function in Spalart-Allmaras model
S	source term	t, *turb*	turbulent
$\tilde{S}$	spalart-Allmaras modified vorticity	u	velocity vector in x-direction
$\bar{S}$	modified vorticity variable	v	velocity vector in y-direction
T	fluid temperature	w	velocity vector in z-direction
V	control volume	x	Cartesian coordinate perpendicular to yz
c	airfoil chord length	y	Cartesian coordinate perpendicular to xz
c_p	pressure coefficient	z	Cartesian coordinate perpendicular to xy
c_{v1}	a constant in Spalart-Allmaras model	γ	specific heats ratio

θ	angle of attack
λ	bulk viscosity
μ	viscosity (absolute)
ν	viscosity (kinematic)
ρ	density
$\tilde{S}$	a constant in Spalart-Allmaras model
$\bar{S}$	tensors of viscous stresses
φ	modified χ
χ	a variable in Spalart-Allmaras model

1. INTRODUCTION

The "Cartesian grid" term belongs to earlier times of 1970's; Peskin (1972) firstly used this term in cardiac mechanics and 2D blood flow simulations. The obvious advantages of Cartesian grid generation approach over the conventional body conforming methodology are summarized as:

- Regardless of the intricacy of the submerged bodies on their boundary, the Cartesian grid remains the same excluding boundaries that are in the neighbourhood of the solid wall where cut-cells are deployed.
- Any kind of adaptation is very easy to execute, i.e. around a shock wave, finer mesh can be obtained automatically. Moreover, computational time is saved without excessive computational cell count and without losing accuracy.
- The governing equations are solved on a Cartesian grid based computational domain. By this method, grid generation greatly reduces in complexity; therewith relatively simpler solutions of the governing equations are maintained in Cartesian coordinates.
- Moving boundaries and topological changes are also simulated and captured easily in Cartesian grids.
- User intervention is held in minimum by Cartesian methods permitting automatic grid generation. Both grid generation and adaptation processes are automated. The proper definition of the problem is the user's only task. Therefore, there is a continuing effort for further automating the grid generation and adaptation processes.

Purvis and Burkhalter (1979) originally defined Cartesian grid methods in their study for 2D potential flow solutions; but the approach used in this study is based upon exquisite state-of-art work of Clarke *et al.* (1986) who expanded Peskin's method to inviscid, two-dimensional, steady flows. In the following year, Gaffney *et al.* (1987) improved this approach for 3D Euler solver on Cartesian grids. In the following decade (De Zeeuw, 1993; Pember *et al.*, 1993; Aftosmis *et al.*, 1998) Cartesian grid generation methods are automated and numerical solution schemes are utilized with adaptive refinement/coarsening ability. In 2000's, essential studies were performed on elaborated applications (Ji *et al.*, 2010; Berger *et al.*, 2012; Liu *et al.*, 2013) and consequentially, in the last decade by the progress in computing efforts and capabilities of computers (Kupiainen and Sjögreen, 2009; Sang and Yu, 2011; Liang, 2012) Cartesian methods are started to handle extremely complicated geometries by the considerable progresses that are yet to come in robustness, computational efficiency and accuracy of the Cartesian based flow solvers.

In this study, main focus is the development of a geometric/solution adaptive, automatically generated compressible flow solver that have ability in solving turbulent flows in both 2D and 3D over different types of elements. An explicit time marching scheme is used in finite volume discretization. Convergence is accelerated employing local time stepping. Object-oriented programming instructions are implemented in order to construct a Cartesian grid generator as the preprocessor of the finite volume solver that is capable of producing accurate solution of turbulent flow around 2D airfoils and 3D bodies. The solver is created to generate the solutions of turbulent flow over two-dimensional airfoils and three-dimensional bodies. Spalart-Allmaras (SA) turbulence model is selected as a turbulence modelling scheme, is successful than algebraic models while being computationally cheaper than Direct Numerical Simulation (DNS) and sophisticated two-equation models. Viscous terms of governing equations and SA turbulence model are subjoined and added into the flow solver.

2. MATERIALS AND METHODS

In this chapter, Reynolds-averaged Navier-Stokes equations (RANS) with SA model (RANS-SA) are presented in integral definition as the compressible flow governing equations. The finite volume statement of 3D RANS-SA equations is attained by cell-centred scheme. Riemann Solver of Roe (Toro, 2009) and Advection Upstream Splitting Method (AUSM) of Liou are the assigned flux construction schemes in which the flux computations through cell boundaries are estimated. Full Approximation Storage (FAS) (Ashford, 1996) multi-grid scheme is used to increase the convergence rate. Second order scheme is occupied and followed for the derivation of conserved variables in both two and three dimensions and used in conjunction with multi-grid techniques. Solution adaptation and primitive variables are calculated by the least squares reconstruction scheme. Details about mesh generation techniques and their implementation rules can be found in Kara *et al.* (2015).

2.1 Flow Solver

Gauss divergence theorem is applied on an arbitrarily selected control volume, V, using RANS-SA equation, which is generated as follows:

$$\frac{\partial}{\partial t}\int_V \mathbf{Q}dV + \int_A (\mathbf{F}\,.\,\mathbf{n})dA = \int_A (\mathbf{G}\,.\,\mathbf{n})dA + \int_V S\,dV \quad (1)$$

where, G stands for the viscous flux vector, F stands for the inviscid flux vector, S stands for the source term, V stands for the cell volume, and Q stands for the conserved variable vector:

$$\mathbf{Q} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \\ \rho \tilde{\nu} \end{bmatrix} \quad (2)$$

where, u, v and w are the velocity components in three dimensions, x, y and z directions, respectively, ρ stands for the density, E stands for the total energy and $\tilde{\nu}$ stands for the SA working variable. Inviscid flux vectors, F, are as follows in equations Eqs. (3):

$$\mathbf{F} = \mathrm{F_x}\,\mathbf{i} + \mathrm{F_y}\,\mathbf{j} + \mathrm{F_z}\mathbf{k} \quad (3.a)$$

$$\mathrm{F_x} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ \rho uH \\ \rho u\tilde{\nu} \end{bmatrix};\ \mathrm{F_y} = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ \rho vH \\ \rho v\tilde{\nu} \end{bmatrix};\ \mathrm{F_z} = \begin{bmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ \rho wH \\ \rho w\tilde{\nu} \end{bmatrix} \quad (3.b)$$

where, p stands for the pressure (static) and H stands for the enthalpy (total).

$$E = e + \frac{u^2 + v^2 + w^2}{2} \quad (4)$$

$$H = E + \frac{p}{\rho} \quad (5)$$

Combining the equation of state and Eqs. (4) and (5) gives:

$$p = (\gamma - 1)\left(\rho E - \frac{\rho\left(u^2 + v^2 + w^2\right)}{2}\right) \quad (6)$$

where, e stands for the specific internal energy and γ stands for the specific heats ratio. Viscous flux vector, G, is given in Eqs. (7):

$$\mathbf{G} = \mathrm{G_x}\,\mathbf{i} + \mathrm{G_y}\,\mathbf{j} + \mathrm{G_z}\mathbf{k} \quad (7.a)$$

$$\mathrm{G_x} = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{yx} \\ \tau_{zx} \\ u\,\tau_{xx} + v\,\tau_{xy} + w\,\tau_{xz} - q_x \\ \frac{1}{\sigma}\rho(\nu + \tilde{\nu})\frac{\partial \tilde{\nu}}{\partial \mathrm{x}} \end{bmatrix} \quad (7.b)$$

$$\mathrm{G_y} = \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{zy} \\ u\,\tau_{yx} + v\,\tau_{yy} + w\,\tau_{yz} - q_y \\ \frac{1}{\sigma}\rho(\nu + \tilde{\nu})\frac{\partial \tilde{\nu}}{\partial \mathrm{y}} \end{bmatrix} \quad (7.c)$$

$$\mathrm{G_z} = \begin{bmatrix} 0 \\ \tau_{xz} \\ \tau_{yz} \\ \tau_{zz} \\ u\,\tau_{zx} + v\,\tau_{zy} + w\,\tau_{zz} - q_z \\ \frac{1}{\sigma}\rho(\nu + \tilde{\nu})\frac{\partial \tilde{\nu}}{\partial \mathrm{z}} \end{bmatrix} \quad (7.d)$$

The viscous stresses, τ_{ij}, are defined by:

$$\tau_{xx} = \lambda\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) + 2\mu_{eff}\,\frac{\partial u}{\partial x} \quad (8.a)$$

$$\tau_{yy} = \lambda\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) + 2\mu_{eff}\,\frac{\partial v}{\partial y} \quad (8.b)$$

$$\tau_{zz} = \lambda\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) + 2\mu_{eff}\,\frac{\partial w}{\partial z} \quad (8.c)$$

$$\tau_{xy} = \tau_{yx} = 2\mu_{eff}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) \quad (8.d)$$

$$\tau_{xz} = \tau_{zx} = 2\mu_{eff}\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) \quad (8.e)$$

$$\tau_{yz} = \tau_{zy} = 2\mu_{eff}\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) \quad (8.f)$$

$$\lambda = -\frac{2}{3}\mu_{eff} \quad (8.g)$$

where effective viscosity, μ_{eff}, is the total of eddy (turbulent) viscosity, μ_t and the molecular viscosity, μ. λ is the bulk viscosity defined by Stoke's relationship (Eq. 8.g). The diffusive fluxes, q_i's, are defined by Fourier's heat conduction law:

$$q_x = C_p\left(\frac{\mu}{\mathrm{Pr}} + \frac{\mu_t}{\mathrm{Pr_t}}\right)\frac{\partial T}{\partial x} \quad (9.a)$$

$$q_y = C_p\left(\frac{\mu}{\mathrm{Pr}} + \frac{\mu_t}{\mathrm{Pr_t}}\right)\frac{\partial T}{\partial y} \quad (9.b)$$

$$q_z = C_p\left(\frac{\mu}{\mathrm{Pr}} + \frac{\mu_t}{\mathrm{Pr_t}}\right)\frac{\partial T}{\partial z} \quad (9.d)$$

In heat flux terminology, γ/Cp = μ/Pr is replaced by γ/Cp = μ/Pr + μ_t/$\mathrm{Pr_t}$. When perfect gas behaviour is assumed:

$$\frac{\mu}{\mu_\infty} = \frac{T_\infty + C_{suth}}{T + C_{suth}}\left(\frac{T}{T_\infty}\right)^{3/2} \quad (10)$$

where, T is the fluid temperature, μ_∞ is the reference viscosity (dynamic), T_∞ is the reference temperature and C_{suth} stands for an effective temperature called Sutherland constant. For air, C_{suth} is taken as 110.4 K and T_∞ is taken as 273.15 K.

Turbulent viscosity, μ_t, is empirically modelled and calculated from SA working variable, $\tilde{\nu}$, of the turbulent transport equation. One should note that using this turbulent transport equation, the continuity equation, three momentum equations and energy equation, the number of equations reaches six which is sufficient to solve and close the system with five unknown primitive variables, ρ, u, v, w, p and turbulent viscosity μ_t. Last term in RANS-SA equation, the source vector, S, is given as:

$$\mathbf{S} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & S_t \end{bmatrix}^T \tag{11}$$

Detailed derivations and implementation techniques of SA working variable, $\tilde{\nu}$ and source term, S_t, are given in Appendix.

2.2 Computational Hierarchy

User intervention is held in minimum by Cartesian methods permitting automatic grid generation. Both grid generation and adaptation processes are automated. The proper definition of the problem is the user's only task. Cartesian grid algorithms are implemented in order to improve computational efficiency. Initial step is the creation of the domain and uniform mesh generation. The determination of cell types by inside-outside test is the second step. Afterwards, geometric adaptation consisting of four parts; box, cut-cell, split-cell and curvature adaptation (only in two dimensions), is applied to the uniform mesh. About 100 different solid body models were tested for the correct grid generation code. Rest of the code contains the flux computations which are not detailed in this article, can be found in the previous study (Kara *et al.*, 2016).

The developed/coded final program is named GeULER-NaTURe (Cartesian **G**rid generator with **eULER**-**Na**vier Stokes **TUR**bulent flow solv**e**r). Following initialization of the program, a data file is called for needed grid generator/solution information (see Tables 1-2-3), grid generator and flow solver starts automatically and successively till initially provided break point of the program is reached. Then, the output files (graphs, tables, contours etc.) are exported in predefined formats.

In Fig. 1, the flow chart of GeULER-NaTURe is presented. The steps are explained in authors' previous studies (Kara *et al.*, 2015; Kara *et al.*, 2016).

All cells are flagged for refinement using a predefined limiting/threshold value. Although the desired residual value is reached, if this value for the refinement is not achieved, the solution is sent for the refinement cycle until the threshold value is passed.

3. RESULTS

In this chapter, the GeULER-NaTURe code is validated and verified for turbulent flows with two test cases. The computational hierarchy shown in Fig. 1 is followed. The numerical calculations are performed in a personal computer with a dual Core™ 3.20-GHz processor (I5), 12 GB RAM and a FORTRAN Compiler. As the two-dimensional test case, a flow around NACA0012 airfoil at a fairly high and real-life-flight Reynolds number of 9 million and Mach number, 0.799, at the corrected angle of attack of 2.26º (corrected for interference of the wind tunnel wall) is examined in GeULER-NaTURe to provide pressure distribution around the airfoil comparing with NASA's experimental benchmark data. In the second turbulent flow test case, validation of GeULER-NaTURe results is executed by comparing them with NASA's experimental benchmark data from NASA's AGARD report of the transonic flow at free stream Mach number, 0.8395, and real-life-flight free stream Reynolds number of 11.7 million around three-dimensional ONERA M6 wing.

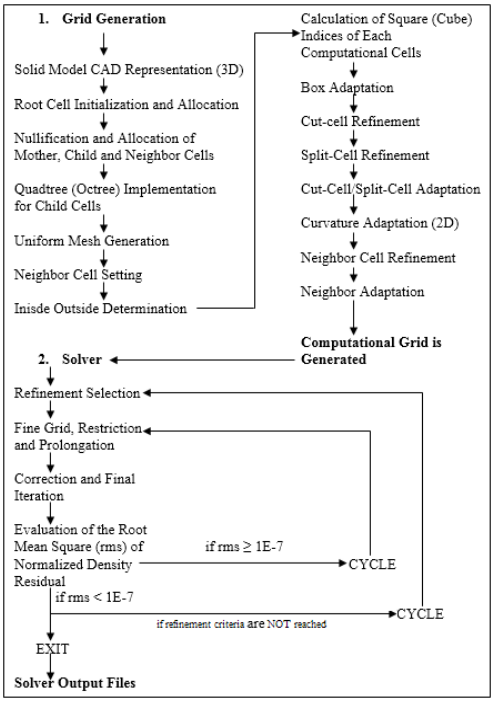

Fig. 1. Flow chart of GeULER-NaTURe.

3.1 NACA 0012: M_∞=0.799, θ=2.26º, Re=9x10^6

The performance of GeULER-NaTURe flow solver in predicting separated flows is tested in the case of high Re, steady, 2D flow around NACA0012 airfoil. The experimental data from NASA's Langley 8-Foot Transonic Pressure Tunnel (Harris, 1981) is used to validate the GeULER-NaTURe results in turbulent flow. Because of tunnel's large span-chord ratio and small side-wall-boundary-layer effects, the data of this experimental study is accepted as a benchmark test case for predicting separated flows. GeULER-NaTURe solution of chord-wise pressure distributions for NACA 0012 airfoil at a fairly high and real-life-flight Reynolds number of 9x10^6 and

Mach number, 0.799, at the corrected angle of attack of 2.26º is compared with the experimental study of Harris (1981) and computational results of Aybay (2004). Harris (1981) used the analysis of Barnwell (1978) to calculate the incremental difference in the angle of attack (in degrees), $\Delta\alpha = -0.125*C_l*c/h$, where C_l is the lift coefficient, c is the chord length, h is the semi-height of the tunnel. Details can be found in Barnwell (1978).

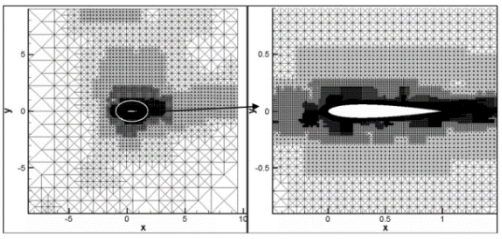

Fig. 2. Mesh generation after three level solution adaptation around NACA 0012.

Eventual, solution adapted mesh constructed around the geometry is shown in Fig. 2 above. In Fig. 3, for initial cell dimensions of 0.05 and 0.02, lift coefficient converges around 0.48 and 0.45, respectively. For the initial cell dimension of 0.01, it converges around 0.39 which is the exact experimental result of Harris (1981).

Boundary conditions, Cartesian grid statistics, and GeULER-NaTURe inputs are summarized in Tables below.

Table 1 Boundary conditions

Mach Number (Free stream), M_∞	0.799
Reynolds Number, Re	9 x 10^6
Angle of Attack, °	2.26
Prandtl Number, Pr	0.72
Turbulence Prandtl Number, Pr_t	0.90
Specific heats ratio of the fluid, γ	1.4
Temperature (Free stream), T_∞	273.15

Table 2 Grid statistics (Cartesian)

Outer boundary size factor	20
Boundary size factor	2.5 x 1.5
Number of successive divisions	4
Size of small cells per maximum body dimension on each plane	0.01, 0.02, 0.05

Table 3 Solution parameters

Numerical flux solution scheme	AUSM or Liou
No. of refinement-coarsening cycle	2
Multiplication no. for refinement	0.05
Order of the solver	2nd
Residual exponent for convergence	-5

In Fig. 4 below, the computed pressure coefficient distributions by GeULER-NaTURe flow solver on the suction (upper) surface and the pressure (lower) surface by using three different initial small cell dimensions, namely 0.05, 0.02 and 0.01 are given together with the experimental benchmark study of Harris (1981). Predictions shown in Figs. 4(a) and 4(b) for the lower surface are seen to be almost identical except the overshoot at the critical shock occurrence on the upper surface and cannot capture the characteristic plateau distribution after the shock. The slight overshoot of the critical shock on the suction surface is the result of staircase phenomenon of cut-cells near the solid boundary (Kidron *et al.*, 2009). The Cartesian methods suffer from this phenomenon and can overcome it if meshing near the boundary layer is designed body-conformal but this cancels out the advantageous automated grid generation specialty of Cartesian methods. The solution given in Fig. 4(c) shows a better prediction of the flow characteristics but there is a small deterioration in the solution of the noncritical pressure surface side. The concave behaviour of the curve of case 4 between x/c = 0.35 and 0.45 can be interpreted as a response to sharp pressure drop at x/c = 0.45, and this phenomenon consolidates the effectiveness of GeULER-NaTURe turbulent flow solver.

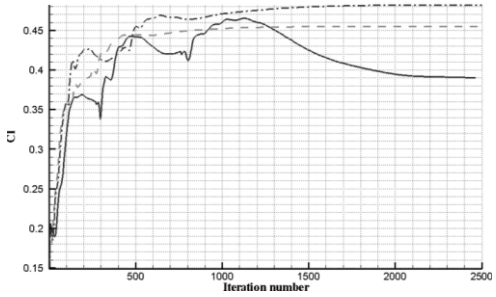

Fig. 3. Transonic, turbulent flow around NACA0012: Convergence histories of the lift coefficient; M_∞ = 0.799, θ = 2.26º; Re = 9x10^6.

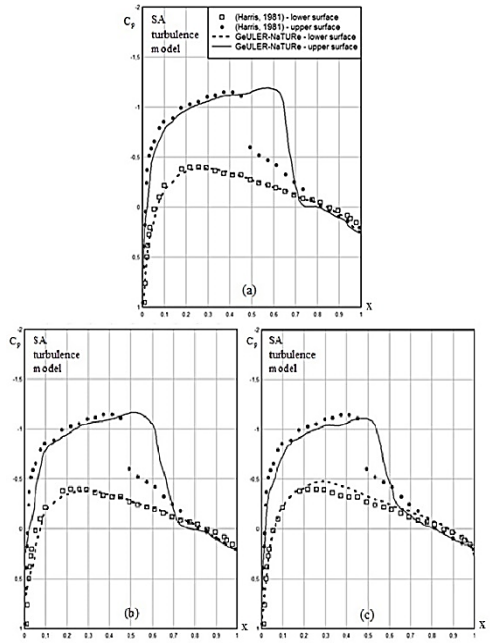

Fig. 4. Pressure coefficient (c_p) distribution results of Harris (1981) compared with GeULER-NaTURe around NACA0012 with SA turbulence model having (a) 0.05 units, (b) 0.02 units and (c) 0.01 units; M_∞ = 0.799, θ = 2.26º; Re = 9x10^6.

Table 4 Comparison of the GeULER-NaTURe C_d and C_l results with Thibert *et al.* (1979) at eight different angle of attacks, °, for transonic test case of NACA 0012; M_∞= 0.75, Re = 4x10^6.

Mach Number	Angle-of-Attack, °	Cd (Thibert)	Cd (GeULER-NaTURe)	Cl (Thibert)	Cl (GeULER-NaTURe)
0,753	-1,67	0,0096	0,0168	-0,269	-0,324
0,756	-0,84	0,0073	0,0112	-0,114	-0,165
0,755	-0,05	0,0068	0,0092	0,010	-0,010
0,754	0,74	0,0072	0,0106	0,124	0,145
0,752	1,49	0,0088	0,0149	0,253	0,290
0,753	2,26	0,0143	0,0240	0,399	0,427
0,754	3,02	0,0246	0,0375	0,523	0,561
0,752	3,79	0,0372	0,0552	0,620	0,691

Mach contours of GeULER-NaTURe are very similar to the figure of a commercial CFD solver package, FloEFD (Mentor Graphics, 2010) shown in Fig. 5. As Mach contours show, the flow increases speed from M_∞ = 0.9 to 1.34 over the upper surface, whereas change of the velocity along the lower surface is much smaller. Steep Mach gradient is captured around 60 % of the mid-chord for both cases.

Figure 6 shows the pressure coefficients and their distribution computed on both sides of the airfoil with computational results taken from Aybay (2004) together with the computational results of GeULER-NaTURe flow solver. Aybay (2004) obtained computational results by utilizing the viscous solution with triangular unstructured mesh and body-conformal grid near the surface by the Spalart-Allmaras model with tripping terms included. The shock location prediction of Aybay (2004) is around 65 % chord airfoil which is worse than GeULER-NaTURe solver. The results of Aybay (2004) are similar to GeULER-NaTURe results in the characteristic of the flow after the location of the shock; GeULER-NaTURe is ahead in predicting the exact location of the first point of the shock appearance. Experiments indicate that the flow separates from suction surface of NACA0012 around the middle of the chord (x/c = 0.55) after a strong shock. GeULER-NaTURe flow solver captures this shock-induced separation around x/c = 0.54 which is comparably better than Aybay's study (2004) that captures the separation around x/c = 0.60. Smooth change of the c_p along pressure surface of NACA0012 is resembled by GeULER-NaTURe flow solver. As a final note, in experiments of Harris (1981) the boundary layer is tripped at 5 % of NACA0012 chord, so that flow is fully turbulent downstream of this point.

As the supplementary two-dimensional study, GeULER-NaTURe simulation results and the experimental validation study of Thibert *et al.* (1979) are given on Table 4 as drag coefficient C_d and lift coefficient C_l values at eight different angle of attacks changing between -1.67 and 3.79, with SA turbulence model having M_∞ = 0.75, Re = 4x10^6. Change of C_d and C_l versus angle-of-attack results are depicted in Fig. 7 that shows the qualitative match of GeULER-NaTURe results with respect to the reference study both for C_d and C_l.

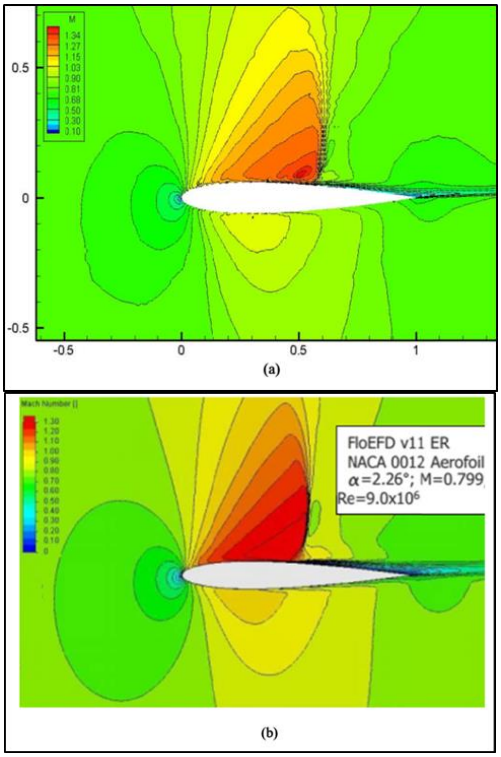

Fig. 5. Mach number contours of (a) GeULER-NaTURe with SA turbulence model having 0.01 units and (b) FloEFD (Mentor Graphics, 2010) around NACA 0012; M_∞ = 0.799, θ = 2.26°; Re = 9x10^6.

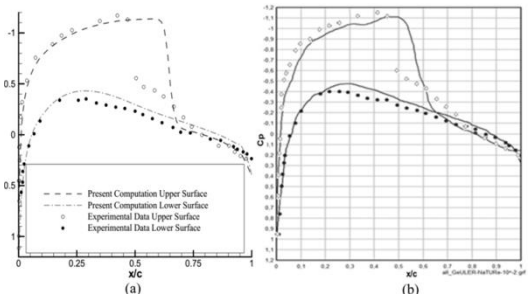

Fig. 6. Pressure coefficient (c_p) distribution of (a) Aybay (2004) compared with (b) GeULER-NaTURe around NACA0012 airfoil with two solution refinements and SA turbulence model with 0.01 units; M_∞ = 0.799, θ = 2.26°; Re = 9x10^6.

Since no trip terms are included in Spalart-Allmaras version of the GeULER-NaTURe solver, it was expected to have a slightly delayed separation and a squeezed boundary layer after the separation point. SA turbulence model with tripping terms, modified one-equation models or two-equation models can be integrated into GeULER-NaTURe flow solver for predicting shock location comparatively well as a future work.

3.2 ONERA M6: M_∞=0.8395, θ=3.06º, Re=11.7x10^6

As in Kara *et al.* (2016), AGARD report (Schmitt and Charpin, 1979) is taken as the reference experimental data. The surface grid structure is shown in Fig. 8. Surface mesh of ONERA M6 wing is eight times denser than the surface mesh used in authors' previous study of inviscid, laminar flow (Kara *et al.*, 2016). This is because the boundary definition is more crucial for viscous, turbulent solution than for inviscid, laminar solution. Wing geometry and grid statistics can be found in Kara *et al.* (2016).

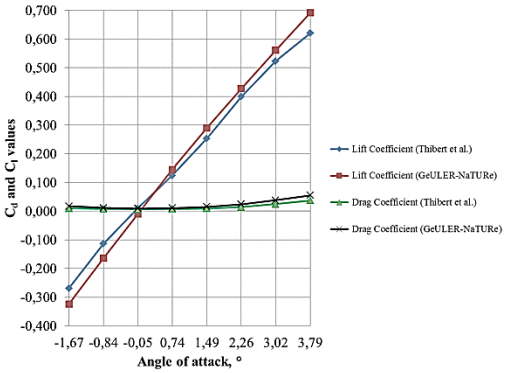

Fig. 7. C_d and C_l values of Thibert *et al.* (1979) in comparison with GeULER-NATURe turbulent flow solver results around NACA0012 airfoil at eight different angle of attacks having M_∞ = 0.75, Re = 4x10^6.

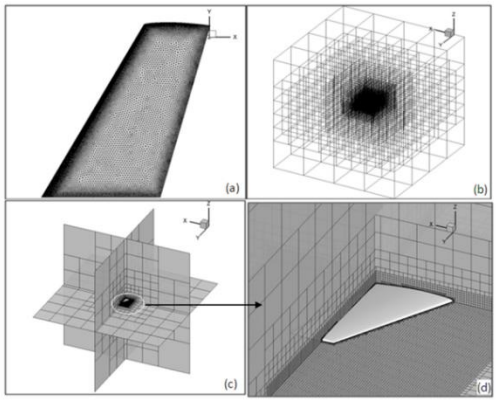

Fig. 8. (a) Magnified outlook of the surface grid of the ONERA M6 wing on XY plane, (b) isometric view of the Cartesian grid solution domain around the wing, (c) sliced expression of the whole domain wing-centred, (d) close-up view on the grid.

The spanwise locations of pressure distribution around ONERA M6 wing are shown in Fig. 9. The lambda shock and its characteristic shape (Gaffney *et al.*, 1986) are distinguished better by turbulence solution in Fig. 9 than inviscid, laminar solution in Fig. 12 and also in Kara *et al.* (2016). The lambda-shaped dark blue region on ONERA M6 wing is built by weak leading-edge shock and strong mid-chord shock near the tip, the location of which is predicted better than inviscid, laminar solution. The isobar patterns of pressure distribution agree better with the results shown in Dadone and Bernard (2007). The shockwave locations on the upper surface of the ONERA M6 wing move downstream in comparison with the inviscid, laminar solution. This is due to the presence of boundary layer.

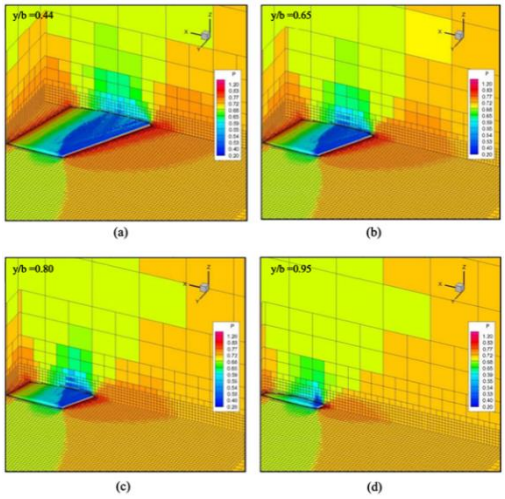

Fig. 9. Spanwise slices on transonic ONERA M6 wing, (a) y/b=0.44, (b) y/b=0.65, (c) y/b=0.80, (d) y/b=0.95; M_∞=0.8395, θ = 3.06°, Re = 11.7 x 10^6, (viscous turbulent).

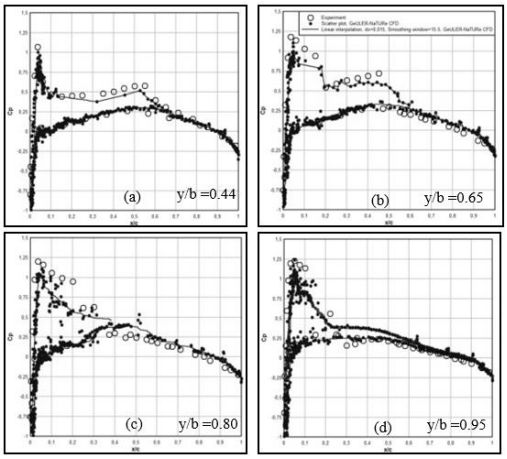

Fig. 10. GeULER-NaTURe viscous, turbulent flow results of pressure coefficient (c_p) distributions versus x/c on the surface of transonic ONERA M6 wing test case at (a) y/b=0.44, (b) y/b=0.65, (c) y/b=0.80, (d) y/b=0.95; M_∞=0.8395, θ = 3.06°, Re = 11.7 x 10^6.

Two strong shockwaves are correctly resolved by GeULER-NaTURe turbulent flow solver. Pressure recovery between these two shockwaves on the upper surface after strong shock appearing in mid-chord region of the wing is correctly predicted and resolved at y/b = 0.44 station, between x/c = 0.35 and 0.55; on y/b = 0.65 station, between x/c = 0.2 and 0.45 as shown in Figs. 10(a) and 10(b). This phenomenon indicates the solution accuracy of GeULER-NaTURe turbulent flow solver. At y/b = 0.80 station, smooth change of the pressure from leading edge to the mid-chord of the wing is qualitatively assessed as in the case of the flow characteristics at y/b = 0.95 station. In short, at four spanwise locations, namely y/b=0.44, y/b=0.65, y/b=0.80 and y/b=0.95, turbulence solution (Fig. 10) of the surface pressure coefficients is in better agreement with experimental results in comparison with the surface pressure coefficients obtained from inviscid, laminar solution (Fig. 11) (Kara *et al.*, 2016). The reason for the scattering behaviour of the results (especially near the tip of the ONERA M6 wing, i.e. at y/b = 0.80 and y/b = 0.95 stations) just after the first shock appearance is the inheriting staircase phenomenon (Kidron *et al.*, 2009) of Cartesian methods that solution refinements are required to eliminate oscillations in the solution. At station y/b = 0.65, inviscid, laminar solution (Kara *et al.*, 2016) was not submitted, since the pressure coefficient scattering cannot be captured because of the low resolution.

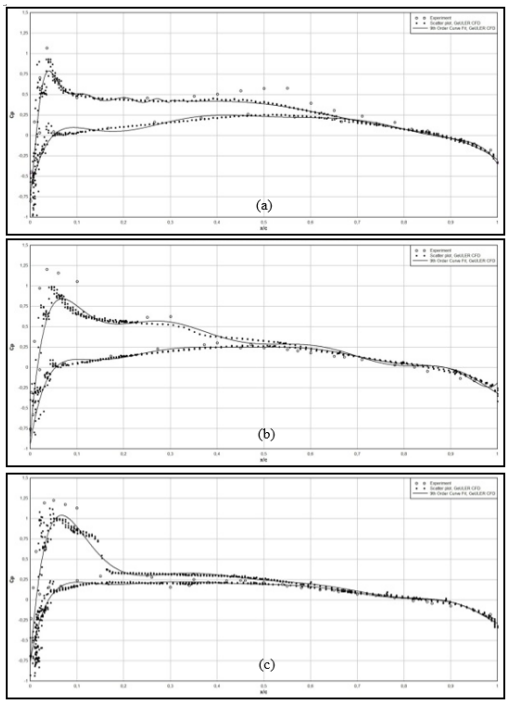

Fig. 11. GeULER-NaTURe inviscid, laminar flow results of pressure coefficient (c_p) distributions versus x/c on the surface of transonic ONERA M6 wing test case at (a) y/b=0.44, (b) y/b=0.80, (c) y/b=0.95; M_∞=0.8395 and θ = 3.06° (Kara *et al.*, 2016).

4. CONCLUSIONS

In this article, it is intended to produce regionally developed Cartesian meshes for 2D and 3D disordered geometries to provide solutions hierarchically in the case of viscous compressible flows around such geometries. A "hands-off" flow solver based on Cartesian grids as the preprocessor is performed using object-oriented programming. Euler equations, Navier-Stokes equations and RANS equations with SA turbulence model are solved for the flows around airfoils and wings.

In predicting separated flow, the performance of the GeULER-NaTURe flow solver with the addition of one-equation SA turbulence model is investigated by computing high Reynolds number steady flow around NACA 0012 airfoil at a fairly high and real-life-flight Reynolds number of 9 million and Mach number of 0.799 with the corrected angle-of-attack of 2.26º. GeULER-NaTURe solutions are compared with the experimental benchmark study, computational studies and Mach number contour of a commercial CFD software package. Smooth change of the pressure coefficient along lower surface of the airfoil is resembled by GeULER-NaTURe flow solver.

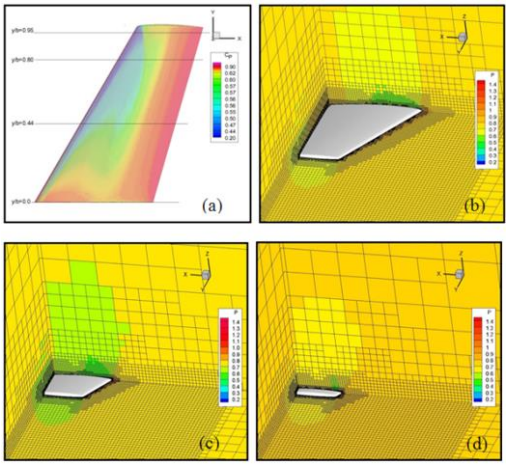

Fig. 12. Spanwise slices on transonic ONERA M6 wing, (a) y/b=0.44, (b) y/b=0.65, (c) y/b=0.80, (d) y/b=0.95; M_∞=0.8395, θ = 3.06°, (inviscid, laminar), (Kara *et al.*, 2016).

The effectiveness of GeULER-NaTURE turbulent flow solver for the real flows around three-dimensional bodies is tested by comparing the computational results with NASA's experimental study around three-dimensional ONERA M6 wing at free stream Mach number of 0.8395 and real-life-flight free stream Reynolds number of 11.7 million. The lower surface shows highly consistent agreement of pressure distribution throughout all locations as in the case of inviscid, laminar solution; moreover, the lambda-shaped region on the surface of the ONERA M6 wing is predicted well unlike inviscid, laminar solution. The shockwave locations on the upper surface of the ONERA M6 wing move downstream with the GeULER-NaTURe solution in comparison with the inviscid, laminar solution (Kara

et al. 2016). This is due to the presence of boundary layer. Two strong shockwaves are correctly resolved by GeULER-NaTURe turbulent flow solver. Pressure recovery between these two shockwaves on the upper wing surface after the strong shock appearing in mid-chord region of the wing is correctly predicted and resolved at y/b = 0.44 station, between x/c = 0.35 and 0.55; on y/b = 0.65 station, between x/c = 0.2 and 0.45 as shown in Figs. 10(a) and 10(b). This phenomenon indicates the solution accuracy of GeULER-NaTURe turbulent flow solver. At y/b = 0.80 station, smooth change of the pressure from leading edge to the mid-chord of the wing is qualitatively assessed as in the case of the flow characteristics at y/b = 0.95 station. In short, at four spanwise locations, namely y/b=0.44, y/b=0.65, y/b=0.80 and y/b=0.95, turbulence solution of the surface pressure coefficients is in better agreement with experimental results in comparison with the surface pressure coefficients obtained from inviscid, laminar solution.

As the future works, some suggestions are given as follows: (1) Integrating special meshing techniques such as grid stitching, cell merging-cell linking to the code can be assigned as a future work in authors' viewpoint. (2) Parallel programming can be applied on the GeULER-NaTURe so that more turbulence models can be added in it. (3) Faster convergence rates can be reached by the use of implicit time stepping techniques. (4) Different solver schemes other than Roe's Riemann Solver and AUSM can be added for faster solutions.

To sum up, in current research a novel and original GeULER-NaTURe mesh based solver using object-oriented programming is developed and executed. Performance of the program is justified via several test cases in both two- and three-dimensions in literature. All results disclose the efficaciousness of the developed codes and qualify in convergence and accuracy.

APPENDIX

The SA model is a one-equation model. The model determines the eddy viscosity using a parabolic partial differential equation. The method is originated from the work of Spalart and Allmaras (1992) with fully-turbulent conditions. Some modifications are used which are based on the studies available in literature (Oliver, 2008; Moro *et al.*, 2011; Allmaras *et al.*, 2012; Burgess and Mavriplis, 2012) for better numerical performance. The baseline, non-dimensional integral form of transport equation of SA turbulence model, excluding laminar suppression and trip terms, is expressed as (In closed form adapted from Eq. (1)):

$$\frac{\partial}{\partial t}\int_V Q_{turb}dV + \int_A F_{turb}dA = \int_A G_{turb}dA + \int_V S_{turb}\,dV \quad \text{(A.1)}$$

explicitly integrated as:

$$\frac{d}{dt}\int_V \rho\tilde{\nu}dV + \int_A \rho\tilde{\nu}\left(\mathbf{u}.\mathbf{n}+\mathbf{v}.\mathbf{n}+\mathbf{w}.\mathbf{n}\right)dA \quad \text{(A.2)}$$

$$= \mathrm{Re}_a^{-1}\sigma^{-1}\int_A \rho\left(\tilde{\nu}+\nu\right)\left(\nabla\tilde{\nu}.\mathbf{n}\right)dA \quad \text{(A.3)}$$

$$-Re_a^{-1}\sigma^{-1}\int_V \left(\tilde{\nu}+\nu\right)\nabla\nu\nabla\rho\mathrm{d}V + Re_a^{-1}\sigma^{-1}c_{b2}\int_V \rho\left(\nabla\nu\right)^2\mathrm{d}V \quad \text{(A.4)}$$

$$+Re_a^{-1}c_{b1}\int_V \rho\tilde{\mathrm{S}}\,\tilde{\nu}\mathrm{d}V \quad \text{(A.5)}$$

$$+\,\mathrm{Re}_a^{-1}c_{b1}\int_V \rho c_{w1}f_w\left(\frac{\tilde{\nu}}{d_{wall}}\right)^2 dV \quad \text{(A.6)}$$

where $\mathrm{d}_{\mathrm{wall}}$ is the closest distance to the wall Re_a is the Reynolds number based on speed of sound and the modified vorticity, $\tilde{\mathrm{S}}$ is defined as:

$$\tilde{\mathrm{S}} = |\omega|\mathrm{Re}_a + \bar{\mathrm{S}};\quad \bar{\mathrm{S}} = \frac{\tilde{\nu}}{\kappa^2\,\mathrm{d}_{wall}{}^2}\mathrm{f}_{\mathrm{v2}};\quad \mathrm{f}_{\mathrm{v2}} = 1-\frac{X}{1+X\,f_{v1}} \quad \text{(A.7)}$$

The magnitude of vorticity, $|\omega|$ is defined in two dimensions and three dimensions, respectively, as follows:

$$|\omega| = abs\left(\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}\right) \quad \text{(A.8.a)}$$

$$|\omega| = \sqrt{\left(\frac{\partial u}{\partial y}-\frac{\partial v}{\partial x}\right)^2+\left(\frac{\partial v}{\partial z}-\frac{\partial w}{\partial y}\right)^2+\left(\frac{\partial w}{\partial x}-\frac{\partial u}{\partial z}\right)^2} \quad \text{(A.8.b)}$$

The Eqs. (A.3) and (A.4) are diffusive terms, Di, (A.5) is the production term, P and (A.6) is the destruction term, De, respectively. Source term is the differentiation of total of these three terms with respect to cell volume. The destruction term includes wall-related function, f_{w} defined as:

$$f_w = g\left(\frac{1+c_{w3}{}^6}{g^6+c_{w3}{}^6}\right)^{1/6} \quad \text{(A.9.a)}$$

where:

$$g = r + c_{w2}\left(r^6-r\right);\quad r = \min\left(\frac{\tilde{\nu}}{\tilde{\mathrm{S}}\kappa^2\mathrm{d}_{\mathrm{wall}}{}^2}, r_{lim}\right) \quad \text{(A.9.b)}$$

The kinematic eddy viscosity is obtained from:

$$\nu_t = \tilde{\nu}\,f_{v1};\quad f_{v1} = \frac{X^3}{X^3+c_{v1}{}^3};\quad X = \frac{\tilde{\nu}}{\nu} \quad \text{(A.10)}$$

The closure coefficients used in Eqs. from (A.2) through (A.10) are:

$$c_{b1} = 0.1355;\sigma = \frac{2}{3};\kappa = 0.41;c_{b2} = 0.622;c_{v1} = 7.1 \quad \text{(A.11)}$$

$$c_{w1} = \frac{c_{b1}}{\kappa^2} + \frac{1+c_{b2}}{\sigma}; c_{w2} = 0.3; c_{w3} = 2;\ r_{lim} = 10 \quad \text{(A.12)}$$

In Eq. (A.7), $\tilde{S}$ is defined that it could become negative and in some conditions divergence becomes inevitable. Ashford (1996) modified $\tilde{S}$ $\tilde{S}$ of the production term (A.5) to assure non-negative values, thus yielding better numerical results:

$$\tilde{S} = |\omega| Re_a f_{v3} + \bar{S} \quad \text{(A.13.a)}$$

where:

$$f_{v2} = \left(1 + \frac{X}{c_{v2}}\right)^{-3};\ f_{v3} = \frac{(1+Xf_{v1})(1-f_{v2})}{\max(X, 0.001)} \quad \text{(A.13.b)}$$

where $c_{v2} = 5$.

In full-scale aerospace applications, typical upper limit of $X = 10^5$ or $2\ x\ 10^5$ is used. In modified vorticity model $\tilde{S}$ cannot have negative values for physically relevant flow conditions and should never fall below $0.3|\omega|$. This suggestion is not true for all conditions so that Allmaras *et al.* (2012) suggested a new modification to prevent negative values of $\tilde{S}$:

$$\tilde{S} = \begin{cases} |\omega| + \bar{S} & , \bar{S} \geq -c_{v2}|\omega| \\ |\omega| + \dfrac{|\omega|\left(c_{v2}^2 |\omega| + c_{v3}\bar{S}\right)}{(c_{v3} - 2c_{v2})|\omega| - \bar{S}} & , \bar{S} < -c_{v2}|\omega| \end{cases} \quad \text{(A.14)}$$

where $c_{v2} = 0.7$ and $c_{v3} = 0.9$.

REFERENCES

Aftosmis, M. J., M. J. Berger and J. E. Melton (1998). Robust and efficient Cartesian mesh generation for component-based geometry. *AIAA Journal* 36(6), 952-960.

Allmaras, S. R. and F. T. Johnson (2012). Modifications and clarifications for the implementation of the Spalart-Allmaras turbulence model. In *Proceedings of Seventh International Conference on Computational Fluid Dynamics (ICCFD7)*, Hawaii, USA.

Ashford, G. A. (1996). An Unstructured Grid Generation and Adaptive Solution Technique for High Reynolds Number Compressible Flows. Ph. D. thesis, University of Michigan, Michigan, USA.

Aybay, O. (2004). Implementation of Spalart Allmaras Turbulence Model to a Two-Dimensional Unstructured Navier-Stokes Solver. Master's thesis, Middle East Technical University, Ankara, Turkey.

Barnwell, R. (1978). Design and performance evaluation of slotted walls for two dimensional wind tunnels. NASA Tech.

Berger, M. J., M. J. Aftosmis and S. R. Allmaras (2012). Progress towards a Cartesian cut-cell method for viscous compressible flow. In *Proceedings of 50th AIAA Aerospace Sciences Meeting*, Nashville, Tennessee, USA.

Burgess, N. K. and D. J. Mavriplis (2012, July). High-order discontinuous Galerkin methods for turbulent high-lift flows. In *Proceedings of Seventh International Conference on Computational Fluid Dynamics (ICCFD7)*, Hawaii, USA.

Clarke, D. K., H. A. Hassan and M.D. Salas (1986). Euler calculations for multielement airfoils using Cartesian grids. *AIAA Journal* 24(3), 353-358.

Dadone, A. and G. Bernard (2007). Ghost-cell method for analysis of inviscid three-dimensional flows on Cartesian-grids. *Computers and Fluids* 36(10), 1513-1528.

De Zeeuw, D. L. (1993). A Quadtree-based Adaptively-refined Cartesian-grid Algorithm for Solution of the Euler Equations. Ph. D. thesis, University of Michigan, Michigan, USA.

Gaffney, R. L., H. A. Hassan and M. D. Salas (1987). Euler calculations for wings using Cartesian grids. In *Proceedings of 25th AIAA Aerospace Sciences Meeting*, Reno, Nevada, USA.

Harris, C. D. (1981, April). Two-dimensional aerodynamic characteristics of the NACA0012 airfoil in the Langley 8-foot transonic pressure tunnel. NASA Tech.

Ji, H., F.S. Lien and E. Yee (2010). Numerical simulation of detonation using an adaptive Cartesian cut-cell method combined with a cell-merging technique. *Computers and Fluids* 39(6), 1041-1057.

Kara, E., A. İ. Kutlar and M. H. Aksel (2015). A solution adaptive multi-grid Euler solver on two-dimensional Cartesian grids. In *Proceedings of 8th Ankara International Aerospace Conference (AIAC)*, Ankara, Turkey.

Kara, E., A. İ. Kutlar and M. H. Aksel (2016). An octree-based solution-adaptive Cartesian grid generator and Euler solver for the simulation of three-dimensional inviscid compressible flows. *Progress in Computational Fluid Dynamics* 16(3), 131-145.

Kidron, Y., Y. Mor-Yossef and Y. Levy (2009) Turbulent flow predictions using a Cartesian flow solver. In *Proceedings of 19th AIAA Computational Fluid Dynamics Conference*, San Antonio, Texas, USA.

Kupiainen, M. and B. Sjögreen (2009). A Cartesian embedded boundary method for the compressible Navier-Stokes Equations. *Journal of Scientific Computing* 41(1), 94-117.

Liang, Q. (2012). A simplified adaptive Cartesian grid system for solving the 2D shallow water equations. *International Journal for Numerical Methods in Fluids* 69(2), 442-458.

Liu, J., N. Zhao, O. Hu, M. Goman and X. K. Li (2013). A new immersed boundary method for

compressible Navier-Stokes equations. *International Journal of Computational Fluid Dynamics* 27(3), 151-163.

Mentor Graphics: FloEFD validation and test-cases. [online]http://www.cadflo.com/fileadmin/user_upload/6._addition._FloEFD_Aero_Valid_External.pdf / (Accessed 28 September 2018).

Moro, D., N. C. Nguyen and J. Peraire (2011). Navier-Stokes solution using hybridizable discontinuous Galerkin methods. In *Proceedings of 19th AIAA Computational Fluid Dynamics Conference*, Honolulu, Hawaii, USA.

Oliver, T. A. (2008). A High-order, Adaptive, Discontinuous Galerkin Finite Element Method for the Reynolds-averaged Navier-Stokes Equations. Ph. D. thesis, Massachusetts Institute of Technology, Cambridge, USA.

Pember, R. B., J. B. Bell, P. Colella, W.Y. Crutchfield and M. L. Welcome (1993). Adaptive Cartesian grid methods for representing geometry in inviscid compressible flow. In *Proceedings of 11th AIAA Computational Fluid Dynamics Conference*, Orlando, Florida, USA.

Peskin, C. S. (1972). Flow patterns around heart valves: A numerical method. *Journal of Computational Physics* 10(2), 252-271.

Purvis, J. W. and J. E. Burkhalter (1979). Prediction of critical Mach number for store configurations. *AIAA Journal* 17(11), 1170-1177.

Sang, W. and J. Yu (2011). Numerically analyzing more efficiently high-lift aerodynamics of wing/body model with omni-tree Cartesian grids. *Aerospace Science and Technology*, 15(5), 375-380.

Schmitt, V. and F. Charpin (1979). Pressure Distributions on the ONERA M6-Wing at Transonic Mach Numbers, Experimental Data Base for Computer Program Assessment. AGARD AR-138 Report, New York, USA.

Spalart, P. R. and S. R. Allmaras (1992). A one-equation turbulence model for aerodynamic flows. In *Proceedings of 30th Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, USA.

Thibert J., M. Grandjacques and L. H. Oham (1979). *NACA 0012 Airfoil.* AGARD Advisory Report, New York, USA.

Toro E. F. (2009). Riemann Solvers and Numerical Methods for Fluid Dynamics. Springer-Verlag, Berlin, Germany.