8th ANKARA INTERNATIONAL AEROSPACE CONFERENCE
10-12 September 2015 - METU, Ankara TURKEY

AIAC-2015-133

A SOLUTION ADAPTIVE MULTI-GRID EULER SOLVER ON TWO-DIMENSIONAL CARTESIAN GRIDS

Emre Kara[1] and Ahmet İhsan Kutlar[2]
University of Gaziantep
Gaziantep, Turkey

Mehmet Haluk Aksel[3]
Middle East Technical University
Ankara, Turkey

ABSTRACT

Cartesian grid method for inviscid flows was a very popular and powerful tool in late 80's for its robustness, quick solution convergence and automatic grid adaption around complex geometries. In last decade, by the help of ingenious approaches and fast computers, the method is born out of its ashes. In this paper, it is aimed to generate locally refined hierarchical Cartesian grids for two-dimensional irregular geometries to provide solutions, which are easy to realize and accurate in the case of inviscid compressible flows around such geometries. Automatic Cartesian grid generation methodology is implemented in object-oriented FORTRAN programming language. Dynamic quadtree data structure algorithm is used to store the parent/child and cell/neighbor connectivities. The grid typically begins with a single root cell, and grows by a recursive subdivision of each cell into its four children which is done in coarsening part of the FORTRAN 90 subroutines. The goals are to enhance automatic grid generation, to increase convergence rate with multi-grid method and to facilitate solution adaptation with least squares reconstruction scheme on inviscid flows. As a benchmark test case, supersonic flow around a Ni-bump construction gave the opportunity to test the shock wave capturing capability of GeULER for a solid body bordered by two solid walls.

INTRODUCTION

Cartesian grid method for inviscid flows was a very popular and powerful tool in late 80's for its robustness, quick solution convergence and automatic grid adaption around complex geometries. There has been a renewal of vigor towards the application of adaptive Cartesian grid generation algorithms especially to the problems involving complex geometries [De Tullio et al., 2007; De Zélicourt et al., 2009; Ebeida et al., 2010; Fujimoto et al., 2009; Hartmann et al., 2008; Liang 2012]. In comparison with body-fitted grid generation schemes, adaptation of inherently non-body-fitted Cartesian grid generation techniques to complex geometries is very easy to implement. With regard to traditionally used unstructured grid methods, the Cartesian grid methods use exclusive line iteration technology and multi-grid techniques on complex geometries. The procedure is also very quick in comparison with the other unstructured grid generation techniques. In external flow solvers, the Cartesian cell method is a beneficial technique for a fast representation of two dimensional flows [Kara, Kutlar, and Aksel, 2013a]. The employed cut-cell methodology can be found in the author's previous paper [Kara, Kutlar, and Aksel, 2013b]. Current study focuses on the generation of locally refined hierarchical Cartesian grids applicable to Euler equations involving two-dimensional irregular geometries providing solutions relatively easy to realize and accurate for the case of compressible flows. As the benchmark case, uniform channel flow over a circular arc bump is tested. The circular arc chord length is taken as equal to the channel height. A thin bump with thickness of 4 % is selected among Ni's studies [Ni, 1982] so that two oblique shock waves can be captured.

METHOD

Considering an arbitrary control cell the conservative integral form of Euler equations is generated using Gauss divergence theorem as follows:

$$\frac{\partial}{\partial t}\int_A \boldsymbol{Q}\mathrm{d}A + \int_S (\mathbf{F}\cdot\mathbf{n})\mathrm{d}S = 0 \qquad (1)$$

[1] Dr. in Mechanical Engineering Department, Email: emrekara@gantep.edu.tr
[2] Assoc. Prof. Dr. in Mechanical Engineering Department, Email: aikutlar@gantep.edu.tr
[3] Prof. Dr. in Mechanical Engineering Department, Email: aksel@metu.edu.tr

where A is the cell area, F is the inviscid flux vector, n is the normal vector, *S* is perimeter of the contour surrounding A. Q represents the vector of any conserved variable such as density, *ρ*, x- and y-velocity components, *u* and *v*, respectively or total energy, *E.*

Semi-discrete form of Eq. (1) can be written as the sum of the fluxes through each cell as follows:

$$A\frac{\partial \boldsymbol{Q}}{\partial t} + \sum_{f=1}^{4}(\mathbf{F}\cdot\mathbf{n})\Delta S = 0 \tag{2}$$

The residuals, *Res*, of each cell can be defined in the following equation:

$$\frac{\partial \boldsymbol{Q}}{\partial t} = -\frac{1}{A}Res(\boldsymbol{Q}) \tag{3}$$

Left hand side of Eq. (3) is discretized in time (Δt) as the finite difference of conserved variable between step number *m*+1 and step number *m*:

$$\frac{\partial \boldsymbol{Q}}{\partial t} = \frac{\boldsymbol{Q}^{m+1} - \boldsymbol{Q}^{m}}{\Delta t} \tag{4}$$

In the solver, explicit time stepping scheme is used and Taylor series expansion for this second order $O(Res^2)$ scheme is given by neglecting higher order terms:

$$Res(\boldsymbol{Q}^{m+1}) = Res(\boldsymbol{Q}^{m}) + \frac{\partial Res(\boldsymbol{Q}^{m})}{\partial \boldsymbol{Q}}(\boldsymbol{Q}^{m+1} - \boldsymbol{Q}^{m}) + O(Res^2) \tag{5}$$

In the integration of the flow solver, n-stage time stepping scheme using Runge Kutta (RK) method with the first order accuracy is used, which is given as:

$$\boldsymbol{Q}^{(0)} = \boldsymbol{Q}^{m};\ \boldsymbol{Q}^{(1)} = \boldsymbol{Q}^{(0)} - CFL\frac{\alpha_1\Delta t}{A}Res\left(\boldsymbol{Q}^{(0)}\right);\ \boldsymbol{Q}^{(2)} = \boldsymbol{Q}^{(0)} - CFL\frac{\alpha_2\Delta t}{A}Res\left(\boldsymbol{Q}^{(1)}\right)$$

....

$$\boldsymbol{Q}^{(m+1)} = \boldsymbol{Q}^{(n)} = \boldsymbol{Q}^{(0)} - CFL\frac{\alpha_n\Delta t}{A}Res\left(\boldsymbol{Q}^{(n-1)}\right) \tag{6}$$

where *n* is the RK stage number, *CFL* is the Courant-Friedrichs-Levy number and *α*'s are the stage coefficients.

During execution of the flow solver, Cartesian grid based structure makes it easier to locally refine or coarsen grids on the critical regions such as shock-based discontinuities, stagnation points, shear layers, etc. This phenomenon is called as solution adaptation which increases the accuracy of the results. Two characteristic lengths, ς, of the control cell are used to gain possession of changing conservative variables from one cell to its neighboring cells by:

$$\varsigma_{D,x} = |\nabla\bullet u|\ A^{0.5};\ \varsigma_{D,y} = |\nabla\bullet v|\ A^{0.5};\ \varsigma_{C,x} = |\nabla\times u|\ A^{0.5};\ \varsigma_{C,y} = |\nabla\times v|\ A^{0.5} \tag{7}$$

where subscripts $_D$ and $_C$ stand for divergence and curl of the velocity vectors, respectively. The standard deviation of these two characteristic lengths, are calculated for entire solution domain by:

$$\sigma_D = \frac{1}{N}\sum_{i=1}^{N}\varsigma_{D,x}{}^2 + \varsigma_{D,y}{}^2;\quad \sigma_C = \frac{1}{N}\sum_{i=1}^{N}\varsigma_{C,x}{}^2 + \varsigma_{C,y}{}^2 \tag{8}$$

where N is the total number of solution domain cells. Cells are flagged for refinement or coarsening using the threshold values t_r and t_c for refinement and coarsening respectively, following the criteria below:

$$\text{if } (\varsigma_{D,x} + \varsigma_{D,y} > t_r\sigma_D) \text{ and } (\varsigma_{C,x} + \varsigma_{C,y} > t_r\sigma_C) \text{ refine} \tag{9}$$

$$\text{if } (\varsigma_{D,x} + \varsigma_{D,y} < t_c\sigma_D) \text{ and } (\varsigma_{C,x} + \varsigma_{C,y} < t_c\sigma_C) \text{ coarsen} \tag{10}$$

t_r is taken as 1 and t_c is taken as 0.1 [Çakmak, 2009].

RESULTS

The numerical calculations are performed on the developed GeULER (cartesian Grid generator with eULER solver) code in a personal computer with an Intel® Core ™ I5 3470 3.20-GHz processor, 12 GB RAM and a FORTRAN Compiler. The five times solution-based refinement meshing result of GeULER is shown in Figure 1. One should note that higly-refined cells near the wall and near the shock locations are executed during the solver's iteration steps. Supersonic flow with 1.4 free-stream Mach number is incident over Ni-bump. Grid generator and flow solver inputs are given in Table 1.

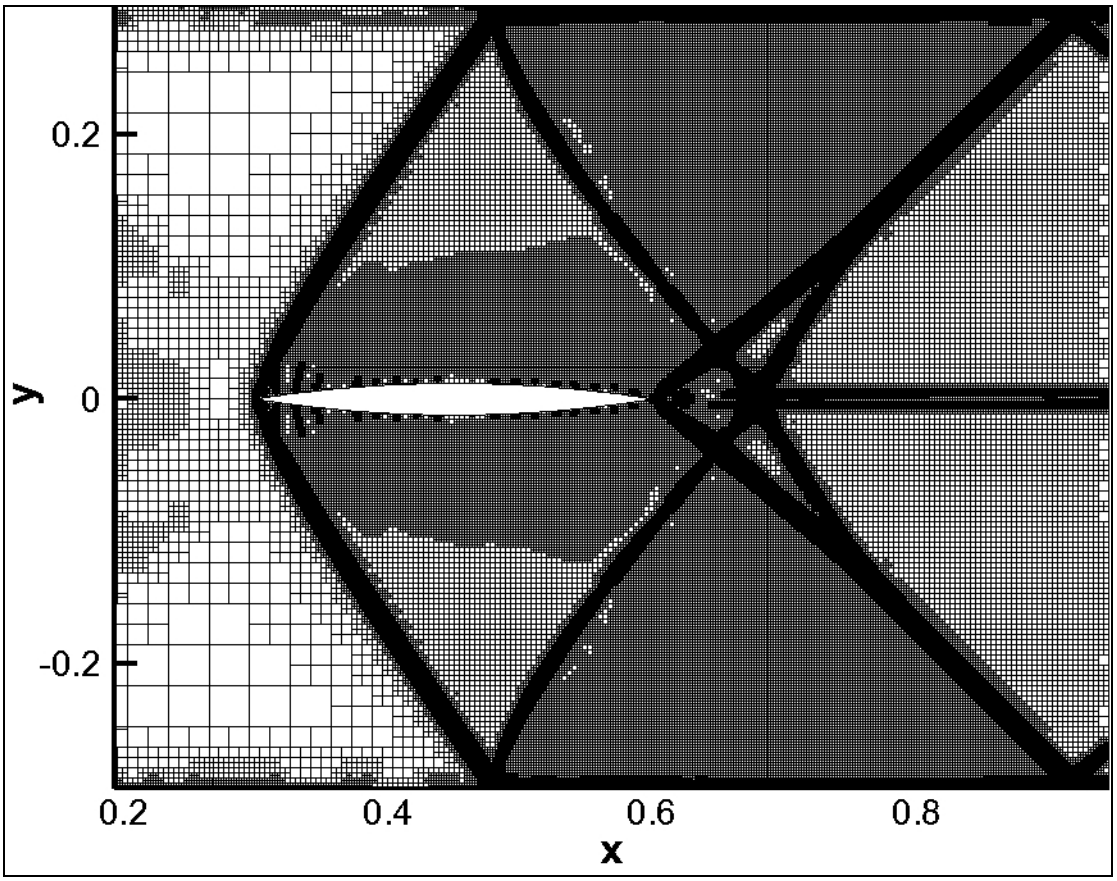

Figure 1: *Five level solution adaptation around circular arc bump with thickness ratio 4 %*

Cartesian Grid	
Outer boundary size factor	1.1
Number of successive divisions to generate uniform grid	2
Boundary size factor for box-adaptation (in x&y directions)	1.2
The size of the cells with respect to maximum body dimension	0.01
Cut-cell adaptation cycle	2
Curvature adaptation cycle	2
Initial cell number (before adaptation)	91856
Final cell number (after five cycles of solution adaptation)	1901241
Maximum level of cell	10
Boundary Conditions	
Mach Number	1.4
Angle of Attack (°)	0.0
Specific heat ratio of the fluid	1.4
Solution Parameters	
Solution scheme for the flux	AUSM
Multiplication criteria for refinement	0.05
Safety coefficient for CFL number	1.0
Number refinement and coarsening cycle	0,1,2,3,4,5
Multi-grid Cycle Number	7
Residual exponent for the limit of convergence	-14

Table 1: *Input file for Ni-Bump*

Figure 2 (a) represents the convergence results of supersonic Ni-bump test cases with V-cycle multi-grid scheme. The procedure iteratively solves the problem till logarithm of root-mean-square (rms) of the density residual drops below a predefined value (in Figure 2, 10^{-14} is selected). Figure 2 (b) shows the convergence of V-cycle and saw-tooth cycle multi-grid methods with respect to the solution without multi-grid method for no solution refinement but this figure does not elicit the difference between the multi-grid methods. As one can see in the Figure 2 (c), V-cycle converges much faster than saw-tooth cycle while using solution refinement techniques, and as expected, five solution refinement cycle with

V-cycle multi-grid method makes about four times faster convergence with respect to solution with no multi-grid.

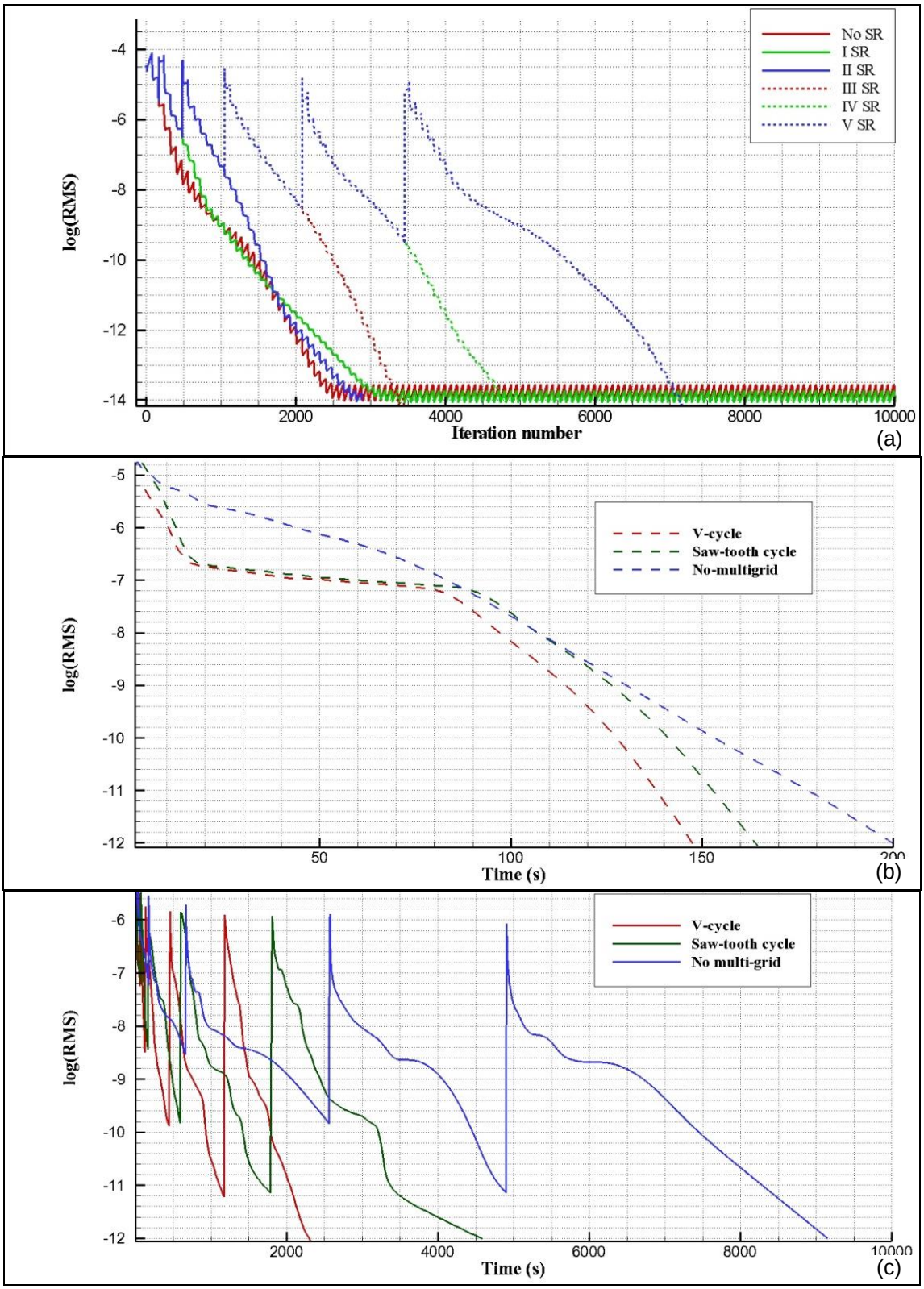

Figure 2: *Supersonic test case, Ni-bump : (a) Convergence histories for V-cycle; (b) residuals versus CPU time with no SR and (c) with five SR; $M_\infty = 1.4$, $\theta = 0.0°$*

The time and RAM use information for no solution refinement and five solution refinement levels is tabulated in Table 2. Interestingly, V-cycle halves the convergence rate of saw-tooth cycle for the wall bounded circular arc bump study. The reference studies of Çakmak [Çakmak, 2009] and Şahin [Şahin, 2011] encourage the use of saw-tooth cycle rather than V-cycle because of high-memory occupation of forcing function in V-cycle multi-grid scheme. The question is which one should be sacrificed;

elapsed time or high RAM requirement. That must be answered separately for each validation study and the most appropriate one should be decided.

Case	Refinement (Ref) no	Multi-grid scheme	Time	RAM Use
1	0	-	3 min. 20 sec.	0.18 Gb
2	0	V-cycle	2 min. 28 sec.	0.54 Gb
3	0	Saw tooth cycle	2 min. 45 sec.	0.43 Gb
4	5	-	152 min. 33 sec.	1.34 Gb
5	5	V-cycle	38 min. 39 sec.	4.93 Gb
6	5	Saw tooth cycle	76 min. 25 sec.	3.43 Gb

Table 2: *Comparison of the results for supersonic inviscid flow around Ni-Bump*

As shown in Figure 3 and 4, four test cases are performed to capture shock waves precisely. The final test case with five solution refinement levels (Figure 4 (b)) is isolated from others and compared with an experimental result [Ni, 1982] and a numerical result [Sert et al., 2004] for validation and verification, respectively.

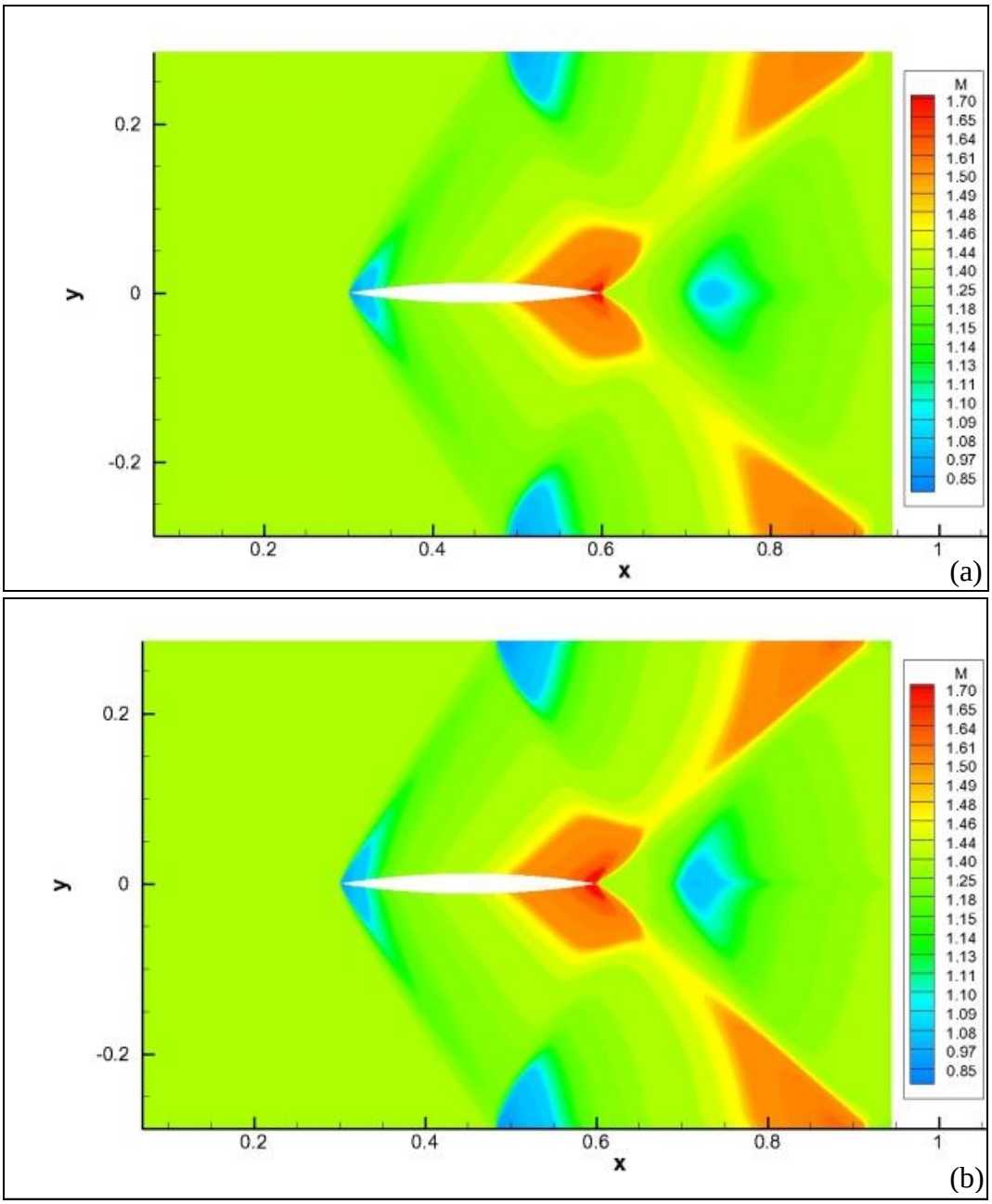

Figure 3: *GeULER results of Mach number contours around Ni-bump using AUSM with (a) no solution refinement (b) one solution refinement; M_∞=1.4 and θ=0.0°*

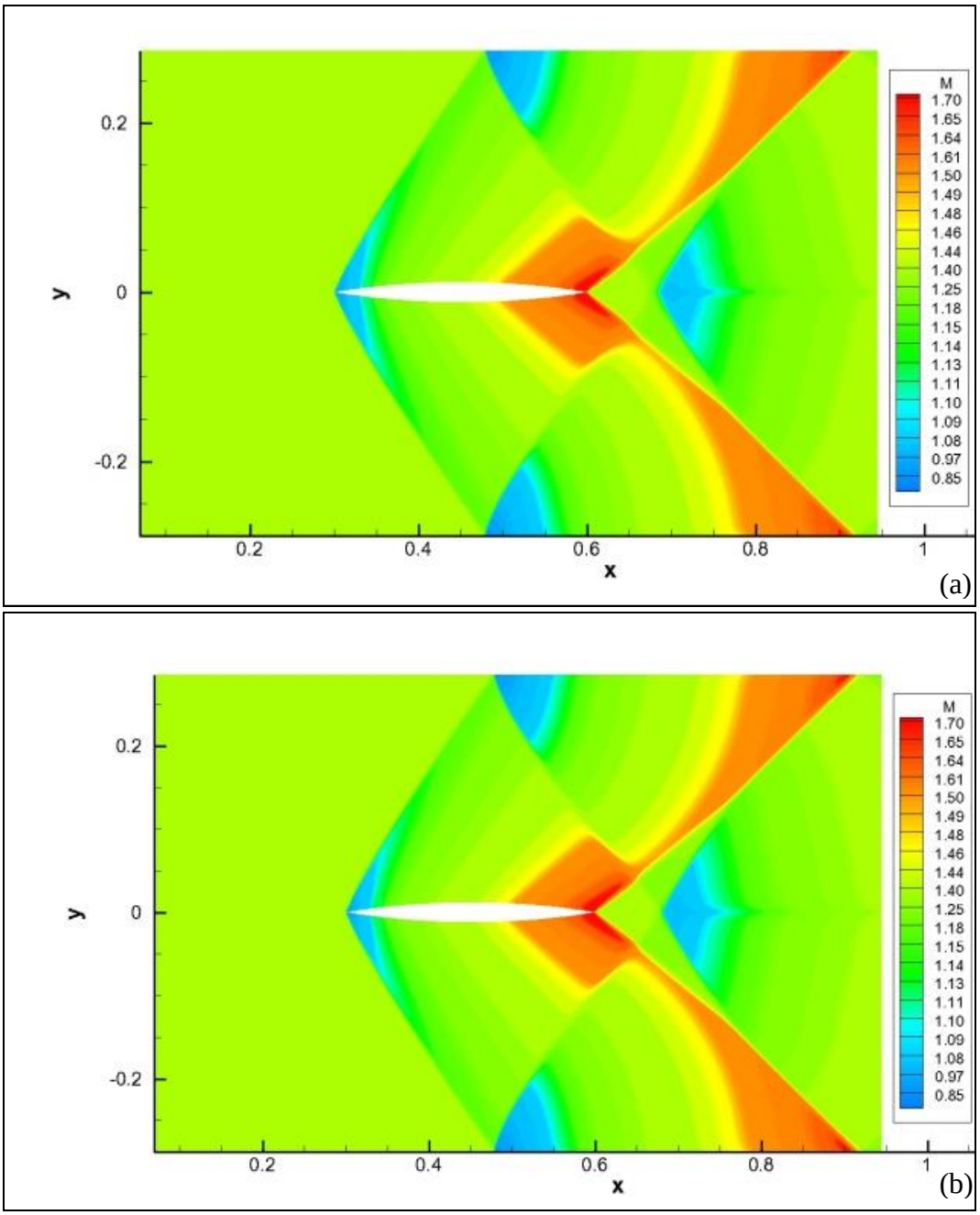

Figure 4: *GeULER results of Mach number contours around Ni-bump using AUSM with (a) three solution refinement, (b) five solution refinement; M_∞=1.4 and θ=0.0°*

As shown in Figure 5, GeULER results capture both oblique shock waves in the same locations as these two references. The incident flow is compressed at the leading edge of the Ni-bump. As Mach contours show, the flow increases speed from M=1.2 to 1.6 over the surface of the bump. This acceleration weakens by this wave emittance along the surface so that the leading-edge shock softens and curves in downstream. If the walls are investigated, the leading shock wave bounces back off the walls and this reflected shock weakens the expanding flow by passing through it. At the trailing edge of the Ni-bump, a weaker (with respect to leading edge shock) recompression shock wave reflects. Entire domain is supersonic with maximum Mach number of 1.7 that occurs after the trailing edge of the Ni-bump and minimum Mach number of 0.85 near the leading edge.

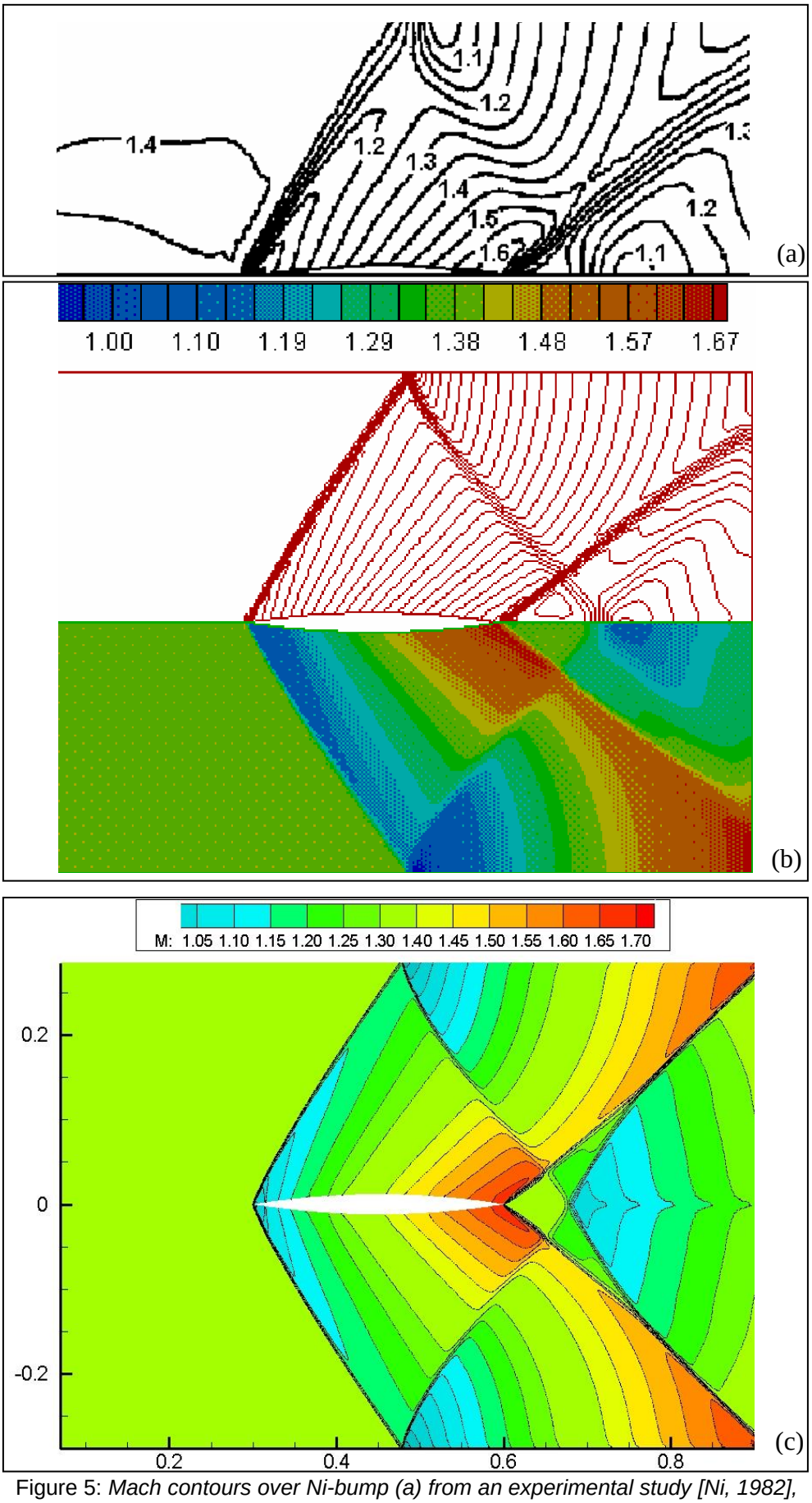

Figure 5: *Mach contours over Ni-bump (a) from an experimental study [Ni, 1982], (b) from a numerical study [Sert et al., 2004] and (c) from GeULER solution; M_∞=1.4 and θ=0.0°*

A subsonic flow at free stream Mach number of 0.11 around multi-element BOEING Model TR-1332 airfoil (see Figure 6) is tested to predict pressure distributions around it with very low Mach number of 0.11. Mach contours of the flow are depicted by using the developed *GeULER* code. Details can be found in authors' most recent paper [Kara et al., 2015].

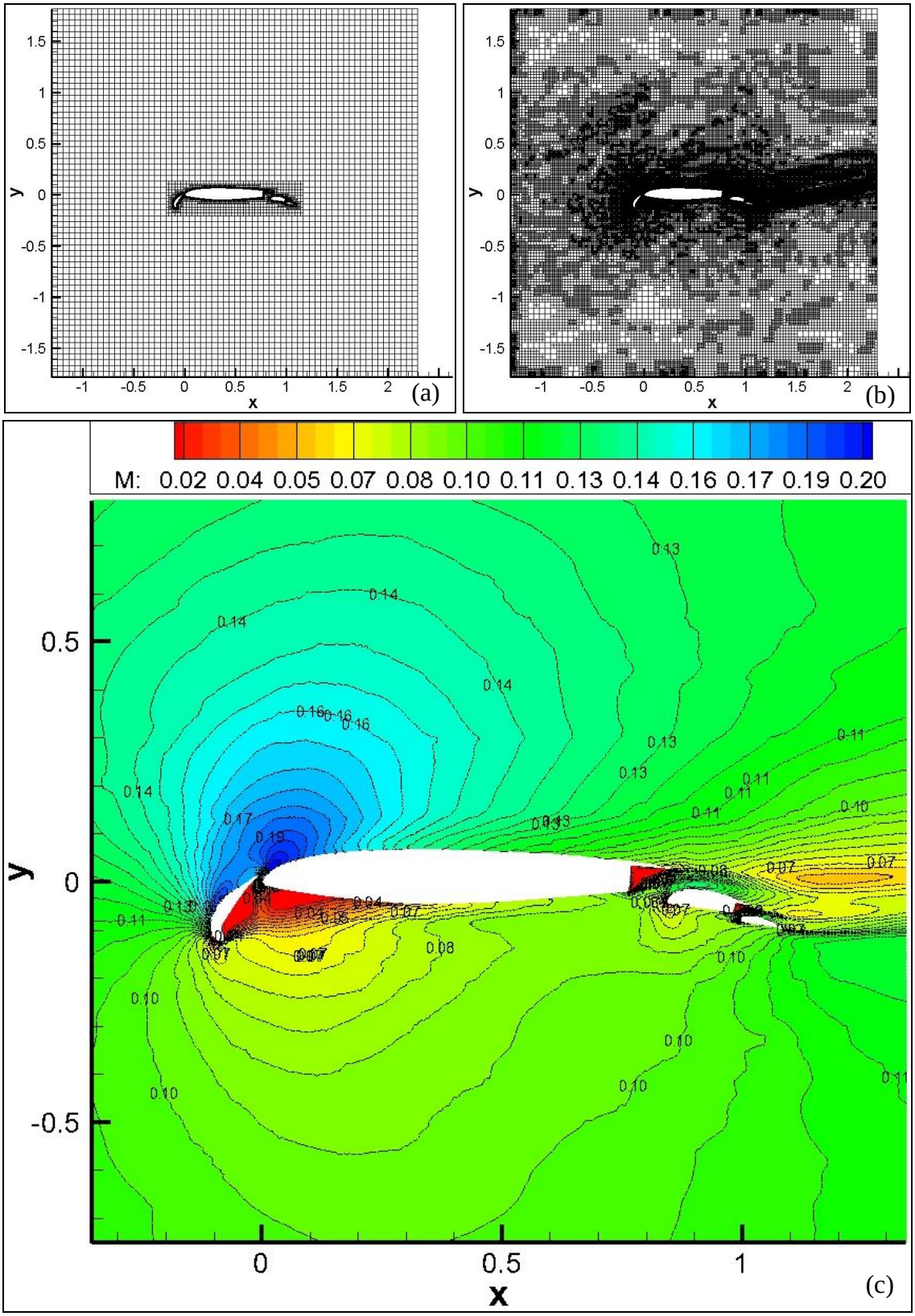

Figure 6: *(a) Box adapted grid, (b) solution adapted grid and (c) Mach contours around BOEING Model TR-1332 by GeULER code [Kara et al., 2015]*

CONCLUSION

In the present study, modern Cartesian cut-cell based grid generation techniques are applied to two-dimensional inviscid compressible flows around irregular geometries. A quadtree data structure is employed to store the information with an easy path for parent-child and cell-neighbor relations. Ray-Casting method, cut-cell determination, curvature adaptation algorithms are studied on adaptively refined schemes. The grid generator is joined to Euler solver for compressible flow solutions. Five stage Runge-Kutta time stepping scheme is used in temporal discretization algorithm. Spatial discretization is constructed using cell-centered finite volume definitions of Euler equations in two-dimensions. Gradients of the corresponding primitive variables are reconstructed by least-squares algorithm and its gradient limiter. GeULER is constructed by Liou's Advection Upstream Splitting Method (AUSM) numerical construction scheme. Full Approximation Storage (FAS) multigrid scheme is used to increase the convergence rate. Second order flux construction scheme is employed for the conserved variables in two dimensions and used in conjunction with multi-grid techniques. The capability of GeULER is validated by test case of Ni-Bump. The continuing study is currently being extended to three dimensional compressible flows.

References

Çakmak, M. (2009) *Development of a Multi-grid Accelerated Euler Solver on Adaptively Refined Two and Three-dimensional Cartesian Grids*, Master's dissertation, Middle East Technical University, Ankara, Turkey, 2009.

De Tullio, M. D., De Palma, P., Iaccarino, G., Pascazio, G., Napolitano, M. (2007) *An Immersed Boundary Method for Compressible Flows Using Local Grid Refinement*, Journal of Computational Physics, Vol.225-2, p:2098–2117, 2007.

De Zélicourt, D., Ge, L., Wang, C., Sotiropoulos, F., Gilmanov, A., Yoganathan, A. (2009) *Flow Simulations in Arbitrarily Complex Cardiovascular Anatomies–An Unstructured Cartesian Grid Approach*, Computers & fluids, Vol.38-9, p:1749–1762, 2009.

Ebeida, M. S., Davis, R. L., Freund, R. W. (2010) *A New Fast Hybrid Adaptive Grid Generation Technique for Arbitrary Two-Dimensional Domains,* International Journal for Numerical Methods in Engineering, Vol.84-3, p:305–329, 2010.

Fujimoto, K., Fujii, K., Wang, Z. J. (2009) *Improvements in the Reliability and Efficiency of Body-Fitted Cartesian Grid Method,* In: 47th AIAA Aerospace Sciences Meeting, AIAA Paper, Vol.1173, p:1–11, 2009.

Hartmann, D., Meinke, M., Schröder, W. (2008) *An Adaptive Multilevel Multigrid Formulation for Cartesian Hierarchical Grid Methods,* Computers & fluids, Vol.37–9, p:1103–1125, 2008.

Kara, E., Kutlar, A.İ., Aksel, M.H. (2013a) *Quad-Tree Based Geometric Adapted Cartesian Grid Generation, Recent Advances in Continuum Mechanics, Hydrology and Ecology: Proceedings of the 8th International Conference on Continuum Mechanics (CM '13*), p:15-19, Rhodes Island, Greece, Jul 2013.

Kara, E., Kutlar, A.İ., Aksel, M.H. (2013b) *A Quad-Tree Based Adaptive Cartesian Grid Generator with Applications on Multi-Element Airfoils,* Proceedings of the 7th Ankara International Aerospace Conference (AIAC'13), AIAC-2013-027, Ankara, Turkey, Sep 2013.

Kara, E., Kutlar A.İ., Aksel, M.H. (2015) *A Solution Adaptive Cartesian Grid Based Euler Solution for Compressible Flow around BOEING TR-1322 Multi-element Airfoil*, Nevşehir Bilim ve Teknoloji Dergisi, Vol.4–1, p:69–80, 2015.

Liang, Q. (2012). *A Simplified Adaptive Cartesian Grid System for Solving the 2D Shallow Water Equations,* International Journal for Numerical Methods in Fluids, Vol.69–2, p:442–458, 2012.

Ni, R.N. (1982) *A Multiple-grid Scheme for Solving the Euler Equations*, *AIAA Journal*, Vol. 20, p: 1565–71, 1982.

Sert C., Aksel, M.H., Dener C. (2004) *Object-oriented Multi-block Approach for the Solution of the Euler Equations*. Modeling, Simulation & Control B, Vol:73-3, p:1-24, 2004.

Şahin, M.S. (2011) *Development of a two-dimensional Navier-Stokes Solver for Laminar Flows Using Cartesian Grids*, Master's dissertation, Middle East Technical University, Ankara, Turkey, 2011.