A NUMERICAL ALGORITHM TO SIMULATE COMPRESSIBLE FLOW WITH SPATIAL-TIME ADAPTIVE PROCEDURE

Gustavo Bono, bonogustavo@gmail.com
Federal University of Pernambuco,
Graduate Program in Mechanical Engineering
Acadêmico Hélio Ramos S/N, 50740-530 Recife, Brazil

João R. Masuero, joao.masuero@ufrgs.br
Armando M. Awruch, amawruch@ufrgs.br
Federal University of Rio Grande do Sul,
Center of Applied and Computational Mechanics,
Av. Osvaldo Aranha, 99, 90035-190 Porto Alegre, Brazil

Tales L. Popiolek, dmttales@furg.br
Federal University Foundation of Rio Grande,
Mathematical Department,
Av. Itália km 8, Campus Carreiros, 96201-900 Rio Grande, Brazil

Abstract. *An algorithm to simulate numerically three dimensional high compressible flows, with a spatial – time adaptive procedure is presented in this paper. The Finite Element Method (FEM) and an explicit one-step Taylor-Galerkin scheme are used for space discretization and time integration, respectively. The numerical solution behavior is analyzed during the time marching process using several error indicators to map regions where physical phenomena presenting spatial high gradients take place and then an adaptive mesh procedure is applied to these regions to increase accuracy of the simulation. An explicit scheme applied to refined meshes can lead to excessive CPU time due to CFL stability constrains. Computational cost may be reduced using a multi-time integration technique with sub-cycles. The capability and efficiency of the time-spatial adaptive procedures are compared with experimental data and with those obtained when a unique global time step is used.*

Keywords*: compressible flows, finite element method, spatial-time adaptive strategy*

1. INTRODUCTION

The prediction of aerodynamic parameter for realistic configurations is essential in the assessment of performance of new designs. Accurate determination of aerodynamics is critical to the low-cost development of new system vehicle (aircraft and spacecraft). The demand to solve finely detailed models with realistic configurations for transonic and supersonic flows has challenged many researchers to come up with new and efficient tools. Specifically, adaptive methods for space discretization and time integration have demonstrated to be useful means to obtain efficient solutions of flow simulations.

Mesh adaptation methods can be separated into three general techniques: *r*, *h* and *p*. In adaptive meshes using the *r* technique, the number of nodes in the computational domains remains fixed and are simply redistributed, so that regions with some specified characteristics are better solved. In *h*-refinement or mesh enrichment, nodes are added to regions of relatively large solution error by locally dividing the elements which make up the mesh or by embedding finer meshes in these regions. In adaptive meshes using the *p* adaptation method, the degree of the basis function is locally adjusted to match the variation in problem solution. Although the above methods were initially designed to be applied in a separate manner, the combination of such strategies may lead to very effective schemes. An adaptive mesh strategy, such as the *h*-refinement technique, which was adopted in this work, has the potential to give numerically accurate and computationally efficient solution, because the mesh is only locally adapted where it is necessary.

Time integration may be performed in one of the two classical approaches, explicit or implicit techniques. Implicit methods are computationally more expensive in terms of computer memory, but they have less stringent stability bounds with respect to explicit schemes. Explicit methods are relatively simple to implement, and they are easily cast in a form suitable for efficient parallel codes, but they are limited by very small time steps due to the Courant-Friedricks-Lewy (CFL) stability condition, which depends on the elements dimension. In many cases it is necessary to use a very small global time step due to the need to accurately capture some phenomena exhibiting high gradients in parts of the finite element domain (requiring very small elements) of one or more variables.

To improve the performance of explicit schemes, specially in unsteady transient flows, an adaptive technique with sub-cycles to integrate in time may be used. Several mixed time integration methods have been implemented and they are based in previous work in the field of structural dynamics. These methods are efficient because they use different time steps in different parts of the mesh, adapted to the local physics or local numerical restrictions. Within the field of computational fluid dynamics (CFD) and in the context of the Finite Element Method (FEM) an early work was

developed by Löhner *et al.* (1984). In this work, the reduction in processing time was about 2 and 4 times for an inviscid two-dimensional problem with an explicit-explicit method (sometimes referred as a multi-time stepping technique using sub-cycles). Chang *et al.* (1993) proposed an implicit-explicit method for viscous and inviscid two-dimensional problems, where the time adaptive algorithm was between 1.1 and 7.1 times faster than the algorithm which uses an unique global time step. In this kind of method, the implicit and explicit integration methods are combined in different parts of the mesh to obtain an optimal time integration scheme. Based on the work of Belytschko and Gilbertsen (1992), Teixeira and Awruch (2001) implemented the sub-cycles technique for inviscid compressible problems with tetrahedral elements. The algorithm had the possibility to control the time step using the same indicators of flow characteristics employed by Argyris *et al.* (1990) to adapt finite element meshes. The time step in the elements is reduced when the indicators identify possibilities of numerical instability or a significative accuracy loss of the solution.

In this paper, results for two problems are presented to demonstrate the application of the unstructured mesh refinement and the adaptive time integration technique (which is applied here for a case characterized by a steady state solution). The space-time adaptive method shown in this work is an optimized version of the code employed by Bono *et al.* (2008). Results for transonic and supersonic flows are presented for a clipped delta wing and a canard-wing-body configuration, respectively, to demonstrate an application of the adaptive procedure in space and time and to assess the accuracy of the computed results by comparisons with experimental data, when they are available.

2. NUMERICAL METHOD

A numerical solution is obtained by solving the 3D unsteady compressible Euler/Navier-Stokes equations written in conservative form. An explicit one-step scheme is employed for solving the compressible inviscid/viscous flow problems. The one-step scheme is similar to that presented by Donea (1984). To develop the scheme, we consider a Taylor expansion of the unknown variables $\mathbf{U}(\mathbf{x},t)$ in time $t = t^{n+1}$. Applying the classical Bubnov-Galerkin weighted residual method in the context of the finite element method (FEM), the spatial discretisation is obtained. This approach could be interpreted as the finite element version of the Lax-Wendroff scheme, used in finite differences. This time integration provides second-order accuracy in time. The formulation exclusively employs tetrahedral finite elements which provide second-order spatial accuracy. Linear unstructured finite elements is chosen due to its flexibility in creating mesh for complex geometries and exactly integrated without numerical quadrature.

The consistent mass matrix is substituted by the lumped mass matrix and then these equations are solved with an explicit scheme. More details can be found in Bono (2008). The explicit character of the algorithm implies that it will be subjected to the Courant-Friedrichs-Lewy stability criterion. At supersonic speeds, an additional numerical damping is necessary to capture shocks and to smooth local oscillations in the vicinity of shocks. An artificial viscosity model, as proposed by Argyris *et al.* (1990), due to its simplicity and efficiency in terms of CPU time, is adopted here.

3. THE ADAPTIVE PROCEDURE IN THE FINITE ELEMENT DOMAIN

The unstructured mesh adaptation has the potential to give numerically accurate and computationally efficient solutions, because the mesh is only locally refined at the places of interest. An adaptive mesh strategy basically is characterized by error indicators, an adaptive criterion and a refinement scheme.

Errors indicators are used to identify the characteristics and behavior of the numerical solutions in order to determine regions of the computational domain where a refinement process is necessary, looking for an accurate solution. In this work, these error indicators take into account regions with velocity gradients, pressure gradients and specific mass gradients. The criterion for mesh adaptation is based in the normal distribution of the error indicators and their mean values and standard deviation. The adaptive process was performed using the *h*-refinement method. Elements refined are divided in eight new elements; his type of refinement is defined as a regular refinement, and it is represented by 1:8. To close the refinement scheme and to avoid hanging nodes, it is necessary to perform irregular refinements in neighbour elements, represented by 1:2, 1:3 or 1:4. Elements having less than four edges divided by new nodes, created as a consequence of the adaptation scheme applied to their neighbour elements, are submitted to irregular refinements. However, if an element has four or more edges divided by new nodes, it is submitted to a regular refinement. In order to improve the geometric quality of the elements in the finite element mesh and to smooth the transition among elements of different size a smoothing technique with node re-allocations could be included. Details of the error indicators, mesh adaptation and the refinement process can be found in Popiolek and Awruch (2006).

4. THE MULTI-TIME INTEGRATION TECHNIQUE USING SUB-CYCLES

The adaptive technique to integrate in time the Euler and Navier-Stokes equations is very suitable for meshes having many groups of elements with different size and for problems where it is convenient to conserve the variable *time* with its real physical meaning.

The adaptive procedure to integrate in time the governing equations can be summarized by the following steps:

1 – The procedure is initiated determining the critical time step for each element i (Δt_{Ei}) and the corresponding minimum value Δt_{Emin} is found. To each element is then addressed an integer multiple value n_{Ei}, which corresponds to the relation

$$n_{Ei} = \text{int}\left(\frac{\Delta t_{Ei}}{\Delta t_{E\min}}\right) \quad \text{with } i = 1,\ldots, nel \tag{1}$$

where *nel* is the total number of elements.

2 – After the local time step has been determined for each element, the elements are collected into g groups based on their relative local time step

$$2^{(g-1)} \le n_{Ei} < 2^{g} \qquad \Delta t_g = 2^{(g-1)}\Delta t_{E\min} \qquad \Delta t_{Ei}^{g} = \Delta t_g \quad \text{with } i = 1,\ldots, nel \tag{2}$$

where Δt_g is the time step of the group g and Δt_{Ei}^{g} is the time step of element i that belongs to group g.

The time step Δt_N for each node N is computed considering the smallest time step corresponding to all elements e connected to node N. Then, the time step for each node is determined by

$$\Delta t_N = \min_e\left(\Delta t_{Ee}^{g}\right) = 2^{g_N - 1}\Delta t_{E\min}\,, \qquad g_N = \min_e\left(g_e\right) \tag{3}$$

and belongs to nodal group g_N which is the minimum value of the element groups g_e of all elements connected to node N.

3 – Finally, the time step for each element Δt_{Ei}^{R} is re-evaluated considering the smallest time step corresponding to all nodes m belonging to this element.

$$\Delta t_{Ei}^{R} = \min_m\left(\Delta t_{Nm}\right) \tag{4}$$

This re-evaluation address each element to a group g that is the smallest group of all nodes connected to the element.

The adaptive method for time integration implemented here has three clocks: the first clock is assigned to the current time (*time*), the second is assigned to each node group (t_N) and the third clock is assigned to each element group (t_G). These clocks, when compared to the current time, indicate when a group of nodes or elements is ready to be updated. The current time is updated with the minimum time step Δt_{Emin} and each nodal or element group with its own time step Δt_g. The initial value of all clocks is zero.

When the current time clock reaches a node group clock or an element group clock, these clocks are updated. Each element group whose clock is equal to the current time $t_G = time$ is updated. After all elements groups have been updated, the nodal loop is executed. Each node group whose clock is equal the current time $t_N = time$ is updated. The nodal variables of all nodes belonging to groups whose $t_N > time$ are linearly interpolated to obtain values to this time, resulting in great reduction of the computational effort, and making that all variables are referred to the physical problem clock (*time)*.

The control time step or master time increment, which is taken as being equal to the largest time step in the element groups, determines when a cycle has been completed. At this time level, all variables are referred to the same time, without any interpolation.

This updating procedure can be visualized easily with a simplified example represented in Figure 1. Consider a mesh which has five one-dimensional elements in three groups (g = 1, 2 and 3) with the following initial time step $1\Delta t$, $2\Delta t$, $2\Delta t$, $4\Delta t$ and $4\Delta t$. Applying stages 1, 2 and 3, previously described, the following node and element groups are obtained: $\Delta t_N = \{1, 1, 2, 2, 4, 4\}$ and $\Delta t_G = \{1, 1, 2, 2, 4\}$, respectively. It should be noted that the time step of the cycle is $\Delta t_c = 4\Delta t$. Considering the graphical representation in Figure 1, nodes are represented by solid diamonds, and elements by lines connecting the corresponding nodes. Time is represented by the vertical axis. The nodal time is represented by black circles and white circles represent the time where linear interpolation was used to obtain the nodal variables at the current time level. In the first integration step, Figure 1, the current time clock is t and the nodes and elements in all groups are updated with their corresponding time steps. Each group clock advances to different time levels. The current time clock advances to $t+\Delta t$ and all nodes whose group clock is ahead this time level have their variables linearly interpolated. It should be noted that nodes 5 and 6 as well as element 5 need only one updating to complete the whole cycle because they are integrated with the maximum time step. In the integration step two, the current time clock is $t+\Delta t$, therefore only nodal and element groups with time step Δt must be updated; for all other nodal groups interpolation is used. To update element 2, interpolated values of the variables of node 3 are necessary. In the third integration sub-cycle, the current time clock is $t+2\Delta t$, nodes 1 to 5 and elements 1 to 4 are updated. As before,

to update element 4 in group 2 interpolated values of the variables of node 5 are needed. Finally, time cycle is completed with the fourth and last sub-cycle, Figure 1. Only nodes and elements in group 1 are updated.

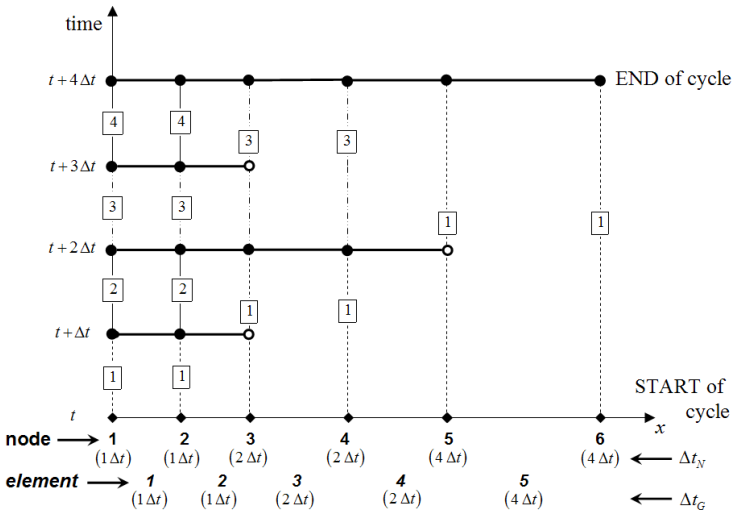

Figure 1. Graphical representation of the sub-cycle method with three groups of elements

The theoretical gain in CPU time (speed-up) using sub-cycles can be calculated by the expression developed by Belytschko and Gilbertsen (1992)

$$speed-up = \frac{t^{nsc}}{t^{sc}} = \frac{NSC}{\sum_{k=1}^{NSC} PESC_k / 100} \tag{5}$$

where t^{nsc} and t^{sc} are the time required to solve a problem with a global time step (without sub-cycles) and using multi-time steps (with sub-cycles), respectively. The number de sub-cycles is *NSC* and the percentage of elements updated in the sub-cycle *k* is *PESC*. In this simplified example, fourth sub-cycles are required (*NSC = 4*) and the percentages of elements for groups *g* = 1, 2 and 3 are 40%, 40% and 20%, respectively. Therefore, the theoretical gain in CPU time is 1.54.

5. RESULTS

The evaluation of the efficiency and performance of an adaptive refinement method and a sub-cycle technique is a complicated task due to the numerous aspects that must be considered. In what follows we try to address at least some of them with the help of classical three-dimensional problems in CFD, such as a clipped delta wing (CDW) in transonic flow and a canard-wing-body (CWB) configurations in supersonic flow.

Comparisons are made with experimental and numerical data to determine the accuracy, the capability and the performance of the adaptive methodologies in space and time.

5.1. Clipped Delta Wing (CDW)

This wing has been studied experimentally by Bennett and Walker (1999). The wing is characterized by an aspect ratio equal to 1.242, a swept leading edge with 50.4 deg, an unswept trailing edge, and a taper ratio which is taken equal to 0.1423. The airfoil is thus a symmetrical circular arc section with t/C = 0.06. Case No. 9E15 corresponding to the reference mentioned above was numerically simulated, which is characterized by an inviscid flow at a free stream with Mach number equal to 0.901 and an angle of attack equal to 4.24 deg. The spatial domain is shown in Figure 2, being the dimensions R = R_1 = 30, L = 55 and L_1 = 15.5. The free stream conditions are given by the specific mass $\rho_\infty = 1.0$, the total energy $e_\infty = 2.1916$ and the pressure $p_\infty = 0.71428$. They were applied on the surface edges defined by AB and BDEF. In ACEF, symmetric boundary conditions were applied and the zero normal velocity condition was prescribed over the CDW. Finally, in CD a prescribed value for pressure $p_\infty = 0.71428$ was applied, and no other boundary conditions were specified.

The flow around the clipped delta wing was simulated with an initial mesh and with two successive mesh refinements, identified by R1 and R2, respectively. The meshes M1, M1R1 e M1R2 consisting of 76523, 273689 and

613017 elements, respectively. The adaptive mesh technique is employed with the following errors indicators: velocity, pressure and specific mass gradients.

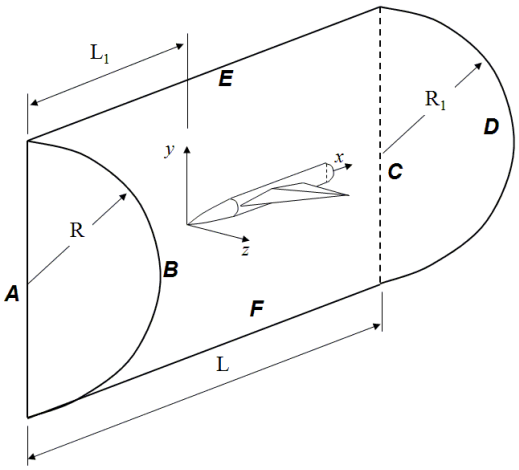

Figure 2. CDW computational domain

In Figure 3, the surface pressure distributions obtained with the initial and final mesh are compared with experimental data obtained by Bennett and Walker (1999) for the case 9E15. Note that the upper shock is barely visible in the results obtained with the initial coarse mesh. As expected, considerable improvement in the resolution of the shocks can be observed when an adaptive mesh is employed. In general, the surface pressure distributions obtained with the final mesh M1R2 (two refinement levels) compare well to those obtained from experimental data, although some differences are observed on the leading edge.

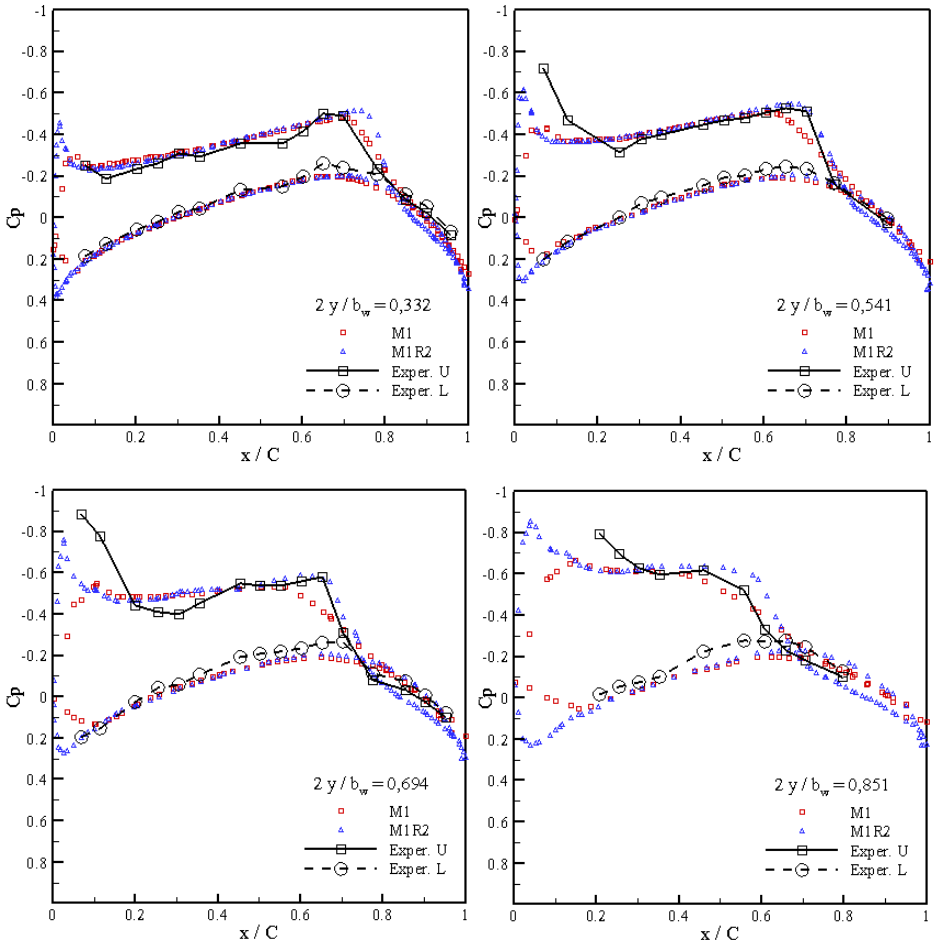

Figure 3. Comparisons of steady pressure coefficient distributions using initial and final meshes in the CDW

To demonstrate the advantages when adaptive procedures in space and time are combined, Figure 4 shows the pressure coefficient distributions on the upper surface of the clipped delta wing corresponding to the final mesh (M1R2). The results obtained with the sub-cycle technique to integrate in time were identified using the index *sc*.

The surface pressure coefficients obtained without the adaptive procedure to integrate in time (using for all elements a global unique time step) compare well to those obtained including this technique, excepting small discrepancies that are evident near the leading edge region. These oscillations occur in regions of the mesh where there is an abrupt transition from a fine to a coarse mesh.

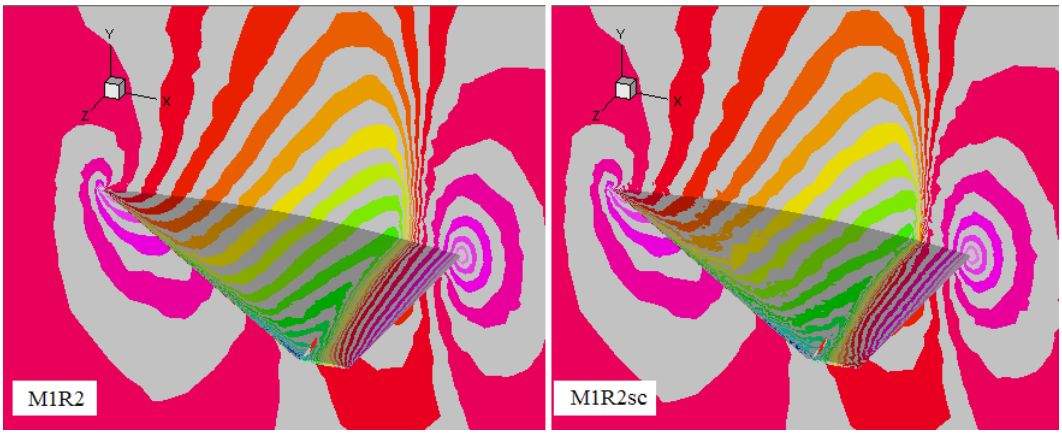

Figure 4. Pressure coefficient distribution on the mesh M1R2 for test case 9E15

In Table 1, the percentage of elements belonging to each element group, the theoretical speed-up and speed-up obtained are presented for the clipped delta wing problem. These results show that the sub-cycles technique in meshes M1, M1R1 and M1R2 are approximately 3.68, 5.77 and 8.22 times faster than cases where a unique global time step was adopted.

Table 1. Percentage of elements belonging to each element group for the CDW.

Group	M1	M1R1	M1R2
1 Δt	0.21	0.09	0.06
2 Δt	0.50	0.20	0.04
4 Δt	19.91	8.45	2.13
8 Δt	56.25	49.29	21.71
16 Δt	11.94	20.03	30.91
32 Δt	5.56	10.03	17.76
64 Δt	1.10	4.50	12.40
128 Δt	0.37	1.10	4.45
256 Δt	4.17	6.31	10.53
Theoretical Speed-Up	***7.45***	***9.86***	***16.4***
Obtained Speed-Up	***3.68***	***5.77***	***8.22***

The time steps distributions over the clipped delta wing on the final mesh, applying twice the refinement process, is shown in Figure 5. It should be noted that smaller time steps ($1\Delta t$ and $2\Delta t$) are localized in the leading and trailing edges. Larger time steps ($4\Delta t$ to $16\Delta t$) are located around the wing and more larger time steps ($32\Delta t$ to $256\Delta t$) are located relatively far from the wing.

Finally, in Figure 6 the pressure coefficient distributions obtained with the time adaptive technique are compared with the distributions obtained without sub-cycles. Both cases agree well. However, in the tip of the wing, small discrepancies are evident when results obtained with sub-cycles are compared with those obtained without sub-cycles due to an abrupt transition from a fine to a coarse mesh.

Differences between theoretical and real speed-up are due the fact that not all the computational work of the code is associated to elements loops, and mainly due to the loss of precision when using sub-cycles, demanding more iterations per time step than the solution with a global unique time step.

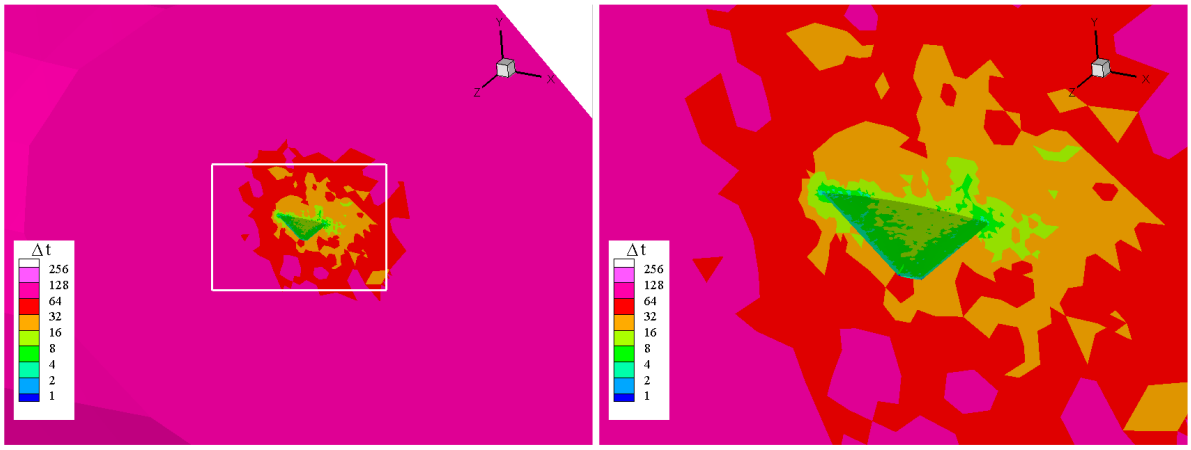

Figure 5. Time steps distributions on the mesh M1R2 for test case 9E15

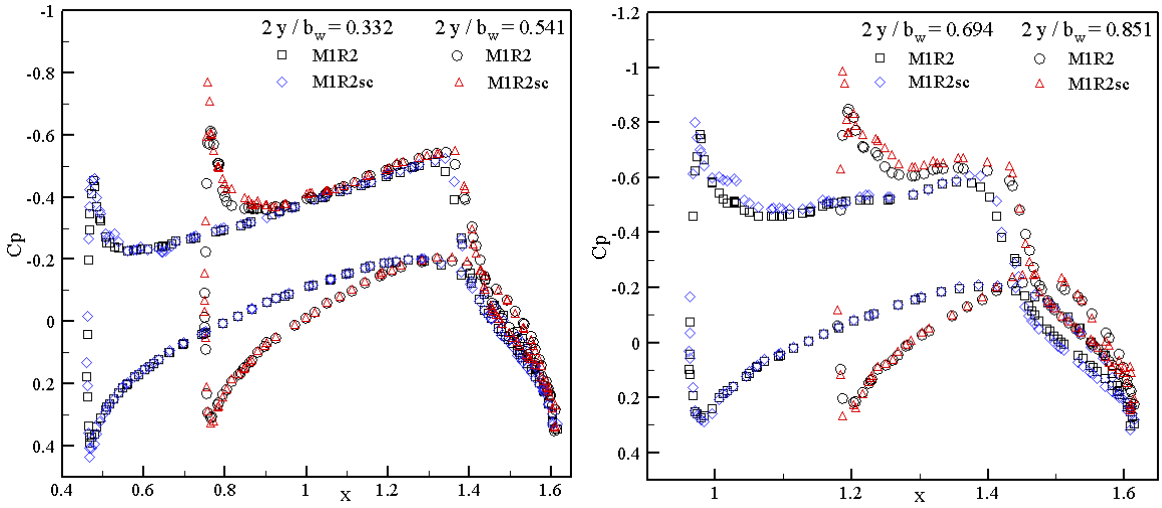

Figure 6. Graphical representation of the sub-cycle method with three groups of elements

In order to estimate the loss of precision using sub-cycles with respect to results obtained with a global unique time step, the mesh M1 was analyzed with both alternatives. Time-steps of the different groups of nodes and elements could spread from the minimum value (1Δt) up to a specified limit value 2^k, which is a multiple of Δt and assuming that k is equal to 1,2,3,4 and 8. The specific mass residue of the solution at each time step, defined as

$$residue = \sqrt{\sum_{i=1}^{nno}\left(\rho_i^{t+\Delta t} - \rho_i^{t}\right)^2} \tag{6}$$

was calculated along the solution time, where ρ_i^t and $\rho_i^{t+\Delta t}$ are the i^{th} values of variable at time t and $t+\Delta t$, respectively, and *nno* is total number of nodes in the mesh. As far the times steps spread from 1Δt, bigger is the value of convergence of the residue, indicating that less accurate results are obtained.

Considering as a reference the numerical solution with a global unique time step, error of the solutions using sub-cycles could be estimated. In Table 2(a), the average error of the specific mass for solutions using sub-cycles with maximum time step varying from 2Δt to 256Δt, as well as the speed-up obtained in each case, are shown. The average error for specific mass of each node was estimated as

$$average\ error = \frac{\sum_{i=1}^{nno}\left|\rho_i^{SC} - \rho_i^{1\Delta t}\right|}{\sum_{i=1}^{nno}\left|\rho_i^{1\Delta t}\right|} \tag{7}$$

where ρ_i^{SC} is the i^{th} value of variable for the sub-cycle solution, $\rho_i^{1\Delta t}$ is for the solution with a global unique time step. The average errors obtained can be considered low, and a significant reduction in the CPU time could be reached. Greater values of the relative errors may occur mainly where the nodal variables have very small values, and could be considered as a measure of the solution accuracy only in these specific regions of the domain, with little physical importance; as a global parameter, average error must be considered.

In problems where the variable *time* has its real physical meaning (e.g. transient problems), the use of multi-time integration technique is useful to obtain values of the variables of problem at certain simulation time with considerable reduction in CPU time and acceptable errors from Engineering point of view. However, when considering steady state problems, an approximate solution with the same error levels of sub-cycles solutions may be obtained stopping the solution with global unique time step before its convergence is reached. Assuming the solution at 210000 Δt as a reference solution, Table 2(b) shows the average errors obtained with the global unique time step solution stopped at a limited number of time steps, and the resultant speed-up in CPU time (calculated as a direct relation between then reference number of time steps and the number of time steps when the approximate solution was stopped).

From the comparison of Tables 2(a) and 2(b), a solution using sub-cycles with time steps spread up to 256 Δt can be considered equivalent in accuracy to the global unique time step solution stopped at 28500 Δt, since both have similar average errors. Comparing the corresponding speed-ups, the last solution is approximately twice faster in CPU time, decreasing gradually for the other cases.

Table 2. Error and speed-up from the use of (a) sub-cycles (SC) solution with time steps spreading up to a limited value, (b) global unique time step (GUTS) solution, stopping the analysis before the convergence is reached. CDW with mesh M1.

(a) Sub-cycles solution (SC)			(b) Global Unique Time Step solution (GUTS)		
Spread	Average Error	Speed-Up obtained	Solution stopped at	Average Error	Speed-Up obtained
up to1 Δt *(= GUTS)*	0.0%	1.00	210000 Δt	0.0%	1.00
up to 2 Δt	0.8%	1.36	98000 Δt	0.5%	2.14
up to 4 Δt	1.6%	2.59	49000 Δt	1.5%	4.29
up to 8 Δt	2.3%	3.28	28500 Δt	2.4%	7.37
up to 16 Δt	2.5%	3.53	20500 Δt	3.0%	10.24
up to 256 Δt	2.5%	3.68	16500 Δt	3.4%	12.72

5.2. Canard-Wing-Body Configuration

Finally, a numerical investigation is carried out over a generic Canard-Wing-Body (CWB) configuration in supersonic flow. The geometry of this configuration is based on the experimental model employed by Jernell (1971). The CWB configuration could be compared with some supersonic transport vehicles, aerospace vehicles or a generic Unmanned Combat Air Vehicle (UCAV). In this study, the flow fields are computed with angles of attack equal to 0.0 deg and 6.2 deg, with Mach and Reynolds numbers at the free stream equal to 4.63 and 1.0×10^5, respectively.

As in the CDW problem, only an half of the CWB configuration is modeled and the domain is the same shown in Figure 2, substituting the CDW by the CWB, being the dimensions R = 20, R_1 = 30, L = 55 and L_1 = 15. The free stream conditions are: velocity $V_\infty = \left(M_\infty \cos\alpha; M_\infty \operatorname{sen}\alpha; 0\right)$, specific mass $\rho_\infty = 1.0$, total energy $e_\infty = 12.5041$ and pressure $p_\infty = 0.71428$. The inflow boundary conditions were applied in the plane defined by the edges AB and BDEF. In ACEF symmetric boundary conditions were applied and the non slip boundary condition was imposed over the CWB configuration. Finally, in CD no boundary conditions were prescribed. The values of the free stream conditions were taken as initial conditions and implemented in all the nodes in the domain, except in the nodes over the CWB configuration.

The first simulations for both cases are performed using the same initial mesh (CWBns) consisting of 242979 linear tetrahedral elements and 45823 nodes. The adaptive mesh technique is employed in both cases with the following errors indicators: low velocity, changes in the velocity direction, velocity, pressure and specific mass gradients. The refinement process was applied only once and it was identified as R1. The meshes CWBns1R1 e CWBns2R1 consisting of 1172187 and 1445062 elements, respectively.

The Mach number distribution over the CWB configuration for different stations (from x_b/L=0.1 to 1.2, with steps equal to 0.1) and for the adapted mesh in the symmetric plane are shown in Figure 7 for both angles of attack. The flow field surrounding the CWB configuration is extremely complex, and some features such as multiple vortices influencing the CWB Mach number distribution are observed for an angle of attack equal to 6.2 deg. Flow over the delta wing at

low and moderate angles of attacks is dominated by two large, counter-rotating leading-edge vortices that are formed by the roll-up of vortex sheets and are carried downstream by the longitudinal component of the free stream velocity.

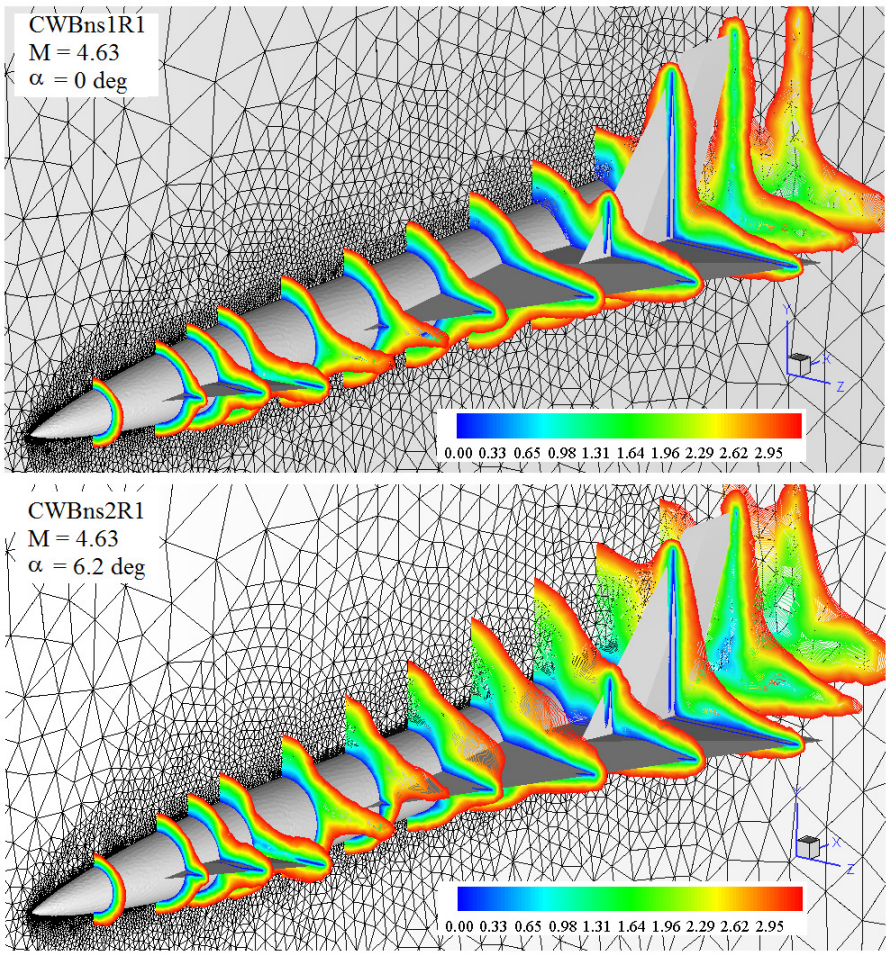

Figure 7. Mach number distribution over a CWB configuration

It should be observed that elements are concentrated in regions where the physical phenomena present high gradients and low velocity. In the fore-body region the results does not present a good resolution because the mesh does not provide a good transition between the different element dimensions.

For CWBns2R1 case, it is observed, two small vortices on the upper wing (the first one between the vertical stabilizer / body and the second on the outside of the wing). The cross-flow separation and roll-up of the shear layer into a strong primary vortex on the leeward-side of the body is clearly evident.

In Table 3, the percentage of elements belonging to each element group, the theoretical speed-up and speed-up obtained are presented for the canard-wing-body configuration. These results show that the sub-cycles technique in meshes CWBns1R1 and CWBns2R1 are approximately 1.77 and 2.09 times faster than cases where a unique global time step was adopted.

6. CONCLUSIONS

The aim of this work is to provide a study of an algorithm which combines an automatic adaptive strategy in space and an explicit time integration in the solution of aerodynamic problems. The adaptation method in space is tested here for steady three-dimensional transonic and supersonic flows. A sub-cycle multi-time integration technique is also applied and the consequent accuracy loss is evaluated.

Results in the previous section show that the adaptive technique in space and time can be used for practical applications. Error indicators, together with a mesh adaptation criterion identify correctly regions where element refinements are necessary to obtain more accurate solutions, producing important improvements.

Multi-time stepping using sub-cycles is a fast and efficient procedure and it is easy to implement. The algorithm is stable and can lead to high values of speed-up when compared to the usual global unique time step approach. For steady state problems, similar speed-ups, with the same accuracy levels may be obtained using sub-cycles or stopping the

problem simulation with a global unique time step before the convergence is reached. In future works the multi-time scheme with sub-cycles will be applied to transient flows, exploring possible important advantages.

Table 3. Percentage of elements belonging to each element group for the CWB.

Group	CWBns1R1	CWBns2R1
1 Δt	2.63	0.50
2 Δt	46.98	20.15
4 Δt	33.59	53.35
8 Δt	8.65	15.36
16 Δt	3.35	5.20
32 Δt	2.25	2.23
64 Δt	0.84	1.52
128 Δt	1.71	1.69
Theoretical Speed-Up	***2.78***	***3.80***
Obtained Speed-Up	***1.77***	***2.09***

7. ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of the CAPES, CNPq and FACEPE.

8. REFERENCES

Bono, G., Awruch, A.M. and Popilolek, T.L., 2008, "A Temporal and Spatial Adaptive Method for Compressible Transonic Flows", Proceedings of the XXIX CILAMCE – Iberian Latin American Congress on Computational Methods in Engineering, Maceió, Brazil, pp. 1-17.

Bono, G., 2008, "Simulação Numérica de Escoamentos em Diferentes Regimes utilizando o Método dos Elementos Finitos" (text in portuguese), Doctoral Thesis, PROMEC, UFRGS, Brazil.

Löhner, R., Morgan K. and Zienkiewicz, O.C., 1984, "The Solution of Non-linear Hyperbolic Equation Systems by the Finite Element Method", International Journal for Numerical Methods in Fluids, Vol.4, pp. 1043-1063.

Chang, H.J., Bass, J.M., Tworzydlo W. and Oden, J.T., 1993, "H-P Adaptive Methods for Finite Element Analysis of Aerothermal Loads in High-speed Flows", CR-189739, NASA.

Teixeira, P.R.F. and Awruch, A.M., 2001, "Three-dimensional Simulation of High Compressible Flows using a Multi-time-step Integration Technique with Sub-cycles", Applied Mathematical Modeling, Vol.25, pp. 613-627.

Argyris, J., Doltsinis, I.S. and Friz, H., 1990, "Study on computational reentry aerodynamics", Computer Methods in Applied Mechanics and Engineering, Vol.81, pp. 257–289.

Donea, J., 1984, "A Taylor-Galerkin for convective transport problems", International Journal for Numerical Methods in Engineering, Vol.20, pp. 101–119.

Popiolek, T.L. and Awruch, A.M. 2006, "Numerical Simulation of Incompressible Flows using Adaptive Unstructured Meshes and the Pseudo-compressibility Hypothesis", Advances in Engineering Software, Vol.37, pp. 260-274.

Bennett, R.M. and Walker, C.E., 1999, "Computational Test Cases for a Clipped Delta Wing With Pitching and Trailing-Edge Control Surface Oscillations", TM-209104, NASA.

Jernell, L.S., 1971, "Comparisons of theoretical and experimental pressure distributions over a wing-body model at high supersonic speeds", TN D-6480, NASA.

9. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.