STRUCTURED MULTIBLOCK SIMULATIONS OF GENERAL LAUNCHER FLOWS INCLUDING THE EFFECT OF FINS

Edson Basso[1], João Alves de Oliveira Neto[2], João Luiz F. Azevedo[1]
[1]Instituto de Aeronáutica e Espaço, CTA/IAE/ASE-N, BRAZIL
[2]Universidade Braz Cubas, BRAZIL
basso@iae.cta.br, alves@ita.br, azevedo@iae.cta.br

Keywords: *Structured Multiblock Grids, Launch Vehicles, CFD*

Abstract

Aerodynamic flow simulations over a typical sounding rocket are presented. This work is inserted in the effort of developing the computational tools necessary to simulate aerodynamics flows over the configurations of the interest for Instituto de Aeronáutica e Espaço of Centro Técnico Aeroespacial. Sounding rocket configurations usually require fairly large fins and, quite frequently, have more than one set of fins. In order to be able to handle such configurations, the present paper uses a novel methodology which combines both Chimera and Patched Multiblock grids in the discretization of the computational domain. The flows of interest are modeled using the 3-D Euler equations and the work describes the details of the discretization procedure, which uses a finite difference approach for structured, body-conforming, multiblock grids. The method is used to calculate the aerodynamics of a sounding vehicle currently under development. The results indicate that the present approach can be a powerful aerodynamic analysis and design tool.

1 Introduction

In the present paper, the results obtained for the simulation of aerodynamic flows about a typical sounding rocket, the SONDA-III, are presented. This work is inserted into the effort of developing the computational tools necessary to simulate aerodynamic flow over aerospace geometries, especially those related to the Brazilian Satellite Launcher (VLS). Details of the work developed so far, as well as results that illustrate the advancements that have been accomplished up to now in this long term research effort, can be seen, among other references, in Azevedo *et al* [1], Azevedo *et al* [2], Azevedo *et al* [3] and Strauss and Azevedo [4]. The SONDA-III presents a quite complex geometric configuration with four front fins and four back fins around a central core. The fins are arranged symmetrically around the central body. An illustrative outline of this configuration is presented in Fig. 1 and the region details the front fin region.

The research group has a fair amount of experience with Chimera and patched multiblock flow simulations for launch vehicle aerodynamics. The present application represents, however, the first time that the group uses the two techniques in the same code. This is also the first time that the group simulates a vehicle with fins. The fundamental objective of the present effort is, therefore, to demonstrate that the use of the two techniques combined will enable the generation of better quality grids for the problems at hand. The governing equations are assumed written in conservative form and they are discretized in a finite difference context. Spatial discretization uses second-order accurate, central difference operators. The time march method is based on a 5-stage, Runge-Kutta algorithm described in Jameson [5] , which also has second-order accuracy in time. The artificial dissipa-

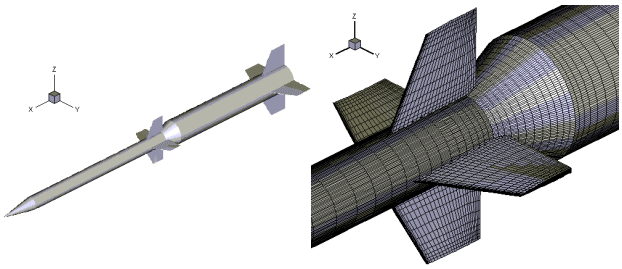

Fig. 1 Perspective view of the SONDA-III (left) and detail of the front fin region (right).

tion terms added are based on the non-isotropic, with described in Turkel [6]. In the present case, Chimera and patched grid techniques are used to simulate flows over the complete SONDA-III rocket. Those techniques together provide the capability to use structured meshes for the discretization of the calculation domain over truly complex configurations. The paper will briefly describe the theoretical formulation used together with a discussion of the numerical implementation aspects, details of the current implementation of the Chimera and Patched grid techniques are also presented and boundary conditions adapted. Results with applications to the SONDA-III are described and some concluding remarks are presented.

2 Theoretical Formulation

In the present work it is assumed that the flows of interest can be represented by the Euler equations in three dimensions. These equations can be written in conservative law form for a curvilinear coordinate system as,

$$\frac{\partial \overline{Q}}{\partial \tau}+\frac{\partial \overline{E}}{\partial \xi}+\frac{\partial \overline{F}}{\partial \eta}+\frac{\partial \overline{G}}{\partial \zeta}=0, \qquad (1)$$

where $\overline{E}$, $\overline{F}$, and $\overline{G}$ are the inviscid flux vectors can be in more details in Vieira [7] and $\overline{Q}$ is the vector of conserved variables, defined as,

$$\overline{Q}=J^{-1}[\rho,\rho u,\rho v,\rho w,e]^T. \qquad (2)$$

In the Eq.(2), ρ is the density, u,v,w are the cartesian velocity components, e is the total energy per unit volume and J is Jacobian of the transformation, represented as,

$$J=(x_\xi y_\eta z_\zeta + x_\eta y_\zeta z_\xi + x_\zeta y_\xi z_\eta \\ -x_\xi y_\zeta z_\eta - x_\eta y_\xi z_\eta - x_\zeta y_\eta z_\xi)^{-1}. \qquad (3)$$

The pressure can be obtained from the equation of state for a perfect gas as,

$$p=(\gamma-1)\left[e-\frac{1}{2}\rho\left(u^2+v^2+w^2\right)\right]. \qquad (4)$$

A suitable nondimensionalization of the governing equations has been assumed in order to write Eq.(1). In particular, the values of flow properties are made dimensionless with respect to freestream quantities, as described in Pulliam [8]. The governing equations were discretized in a finite difference context in structured hexahedral meshes which would conform to the bodies in the computational domain. Since a central difference spatial discretization method is being used, artificial dissipation terms must de added to the formulation in order to control nonlinear instabilities. The artificial dissipation terms used here are based in Turkel [6]. This model is nonlinear and nonisotropic, with the scaling of the artificial dissipation operator in each coordinate direction weighted by its own spectral radius of the corresponding flux Jacobian matrix. In the present implementation, the residue operator is defined as,

$$RHS^n=-\Delta t(\delta_\xi E^n+\delta_\eta F^n+\delta_\zeta G^n). \qquad (5)$$

where, the $\delta_\xi,\delta_\eta,\delta_\zeta$ terms represent mid-point central difference operators in the ξ,η,ζ directions, respectively. The numerical flux vectors and artificial dissipation operators are defined as,

$$E_{i\pm\frac{1}{2},j,k}=\frac{1}{2}\left(E_{i,j,k}+E_{i+1,j,k}\right)-d_{i\pm\frac{1}{2},j,k}, \\ F_{i,j\pm\frac{1}{2},k}=\frac{1}{2}\left(F_{i,j,k}+F_{i,j+1,k}\right)-d_{i,j\pm\frac{1}{2},k}, \qquad (6) \\ G_{i,j,k\pm\frac{1}{2}}=\frac{1}{2}\left(G_{i,j,k}+E_{i,j,k+1}\right)-d_{i,j,k\pm\frac{1}{2}}.$$

The artificial dissipation operators, $d_{i\pm\frac{1}{2},j,k}$,$d_{i,j\pm\frac{1}{2},k}$ and $d_{i,j,k\pm\frac{1}{2}}$, are defined

precisely as described in Turkel [6]. Since steady state solutions were the major interest in present study, a variable time step convergence acceleration procedure has been implemented. The time march is performed based on a 5-stage, 2nd-order accurate, hybrid Runge-Kutta time-stepping scheme, which can be written as,

$$\begin{aligned} Q_i^{(0)} &= Q_i^n, \\ Q_i^{(1)} &= Q_i^{(0)} - \alpha_1 RHS^{(0)}, \\ Q_i^{(2)} &= Q_i^{(0)} - \alpha_2 RHS^{(1)}, \\ Q_i^{(3)} &= Q_i^{(0)} - \alpha_3 RHS^{(2)}, \\ Q_i^{(4)} &= Q_i^{(0)} - \alpha_4 RHS^{(3)}, \\ Q_i^{(5)} &= Q_i^{(0)} - \alpha_5 RHS^{(4)}, \\ Q_i^{n+1} &= Q_i^{(5)}, \end{aligned} \tag{7}$$

where $\alpha_1 = \frac{1}{4}$, $\alpha_2 = \frac{1}{6}$, $\alpha_3 = \frac{3}{8}$, $\alpha_4 = \frac{1}{2}$ and $\alpha_5 = 1$. Since steady state solutions are the major interest in the present study, a variable time step convergence acceleration procedure has been implemented. The time step is defined as,

$$\Delta t_{i,j,k} = \frac{CFL}{c_{i,j,k}}. \tag{8}$$

The characteristic velocity $c_{i,j,k}$ is defined as,

$$\begin{aligned} c_{i,j,k} = max\Big(& |U| + a\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2}, \\ & |V| + a\sqrt{\eta_x^2 + \eta_y^2 + \eta_z^2}, \\ & |W| + a\sqrt{\zeta_x^2 + \zeta_y^2 + \zeta_z^2}\Big), \end{aligned} \tag{9}$$

where a is the speed of sound and U, V and W are the contravariant velocity components. It should be emphatized that only the convective operator inside RHS term indicated in Eq.(7) is actually evaluated at every time step. The artificial dissipation term is only evaluated in the first and second stages of the time march procedure. It can be shown that this provides enough damping to mantain nonlinear stability as defined in Jameson [5] whereas it yields a more efficient numerical scheme.

3 Computational Grid Topology

The SONDA-III rocket possesses a central body where four fins and four back fins are mounted. In order to save computational resources, $\frac{1}{8}$ of complete configuration in the azimutal direction was simulated. This simplification is valid in the present work because only simulations with null attack angle were considered. This way, taking advantage the symmetry of the problem, the configuration is reduced to $\frac{1}{8}$ of the central body in the azimutal direction, $\frac{1}{2}$ front fin and $\frac{1}{2}$ back fin. In total, 13 meshes with relatively simple geometry to model the rocket and the the fins are used. These meshes are distributed in the following way:
- 7 meshes for the front fin;
- 3 meshes for the back fin;
- 1 mesh for central body, as seen in Fig. 2;
- 1 (background) mesh for front fin, as seen in Fig. 2;
- 1 (background) mesh for back fin, as seen in Fig. 2.

The computational meshes used in the present

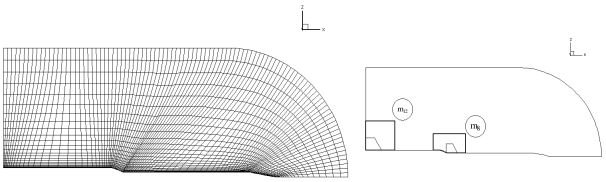

Fig. 2 Central body mesh (left) and the background meshes m_8 and m_{12} (right).

work were all generated by algebric methods with in each block. In particular, the multisurface algebric grid generation technique described by Fletcher [9] has been implemented in a fairly general code for the present configurations. The code allows grid clustering at various regions and a fair amount of control on the grid point distribution along the normal direction. Both hyperbolic tangent and exponential grid stretching functions are used to obtain the desired grid clustering and coarsening over the body. The meshes generated by that method are 2-D. The mesh that discretizes

the central body is rotated around the longitudinal axis, obtaining a 3-D mesh. Initially, for the fins, 2-D meshes are generated for the superior and inferior surface. Those meshes can be seen in details in Fig. 3. The inferior surface is deformed

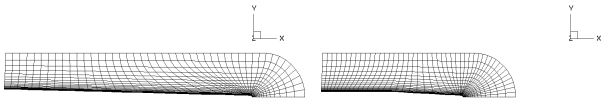

Fig. 3 2-D surface on the root (left) and on the top (right) of the front fin generated by an algebraic method.

through a coordenate transformation to conform to the cylindrical and conical sections of the central body. Finally, intermediate surfaces are obtained through an interpolation on the top and bottom surfaces previously calculated, as shown in Fig. 4.

Previous work on tridimensional configuration

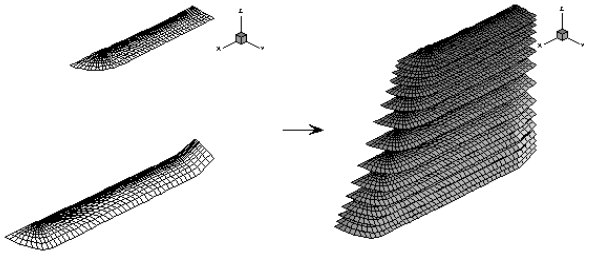

Fig. 4 Intermediate surfaces obtained by interpolation of the tip and root surfaces of the front fin.

of launch vehicles, using the VLS configuration, as can be seen in Basso [10], used only Chimera grids to discretized the computational domain. However, during the initial phase of planning of the meshes for the SONDA-III, the research group noticed that, due to the geometric characteristics of the new problem, using only Chimera meshes would not be viable. The adapted solution was to use Chimera in conjunction with patched grids since this procedure allowed the generation of meshes in a much simpler way, in comparison with other proposals that just used

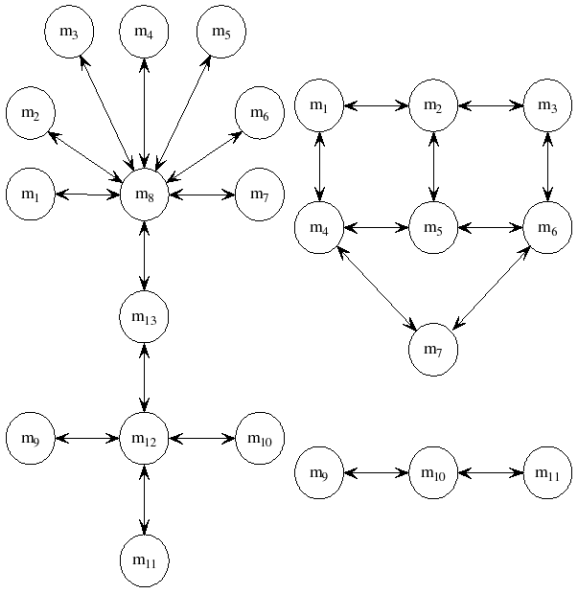

Fig. 5 Information flow of the Chimera meshes (left) and the patched meshes (right).

one technique or another. The Chimera subroutines of the original solver for the VLS were adapted and it was implemented additional subroutines for the use of patched grids. It was also implemented routines for the control of the flow of information among the meshes, and all the particularities of the original code for the configuration of the VLS were eliminated. With that, the research group developed a somewhat general code, that can work with Chimera and patched grids, in complex configurations. The number of meshes that the code can manage is just limited by the amount of memory of the machine.

Basically, the meshes m_1 to m_7, that involve the front fin, exchange information amongst themselves using the technique of patched meshes. These 7 meshes exchange information, through of the Chimera interfaces with the m_8 mesh (*background*), and finally, the m_8 mesh (*background*) exchanges information with the central body mesh m_{13}. For the back fins, the process is similar. The *background* mesh (m_8 and m_{12} meshes) has the function of serving as transition among the fin meshes, that possess a great number of points, and the central body mesh, that possesses few points. Besides, the *background*

mesh hides the complexity of the configuration, since the central body mesh does not see the fin meshes. In case the *background* meshes were not used, the central body mesh would have many more points in order to communicate in an efficient way with the fin meshes. The flow of information among the meshes can be seen in Fig. 5.

4 Boundary Conditions

For the configurations of interest here, the types of boundary conditions that should be considered include upstream (entrance), solid-wall, far-field, symmetry, upstream centerline and downstream (exit) conditions. The rocket upstream centerline is a singularity of the coordinate transformation and, hence, an adequate treatment for this boundary must be provided. The approach consists in extrapolating the property values from the adjacent longitudinal plane and averaging the extrapolated values in the azimuthal direction in order to define the updated properties at the upstream centerline.

The vast majority of the previous experience of the research group in the use of Chimera or patched multiblock grids considered the VLS configuration, without including afterbody or plume effects. For such a configuration, the number of grid blocks required is rather small. Therefore, it is possible to have the boundary conditions hard-coded for each specific case. However, in the present case, the procedure of writing 13 separate subroutines to work with the 13 meshes would be extremely difficult and prone to mistakes. Furthermore, the objective should always be to try to come up with a code as general as possible and, certainly, that does not depend on the particularities of the configuration under consideration. Again, the approach is to eliminate the particularities of the original code and to create a more powerful subroutine, that could work with the diversity of boundary conditions that the SONDA-III meshes present.

The mesh blocks are considered as hexahedra in computational space and each one of the 6 faces is numered as indicated in Fig. 6. The code, that represents a certain boundary condition is associated to each face. With this method, the solver implements the 78 boundary conditions in a simple format for the user.

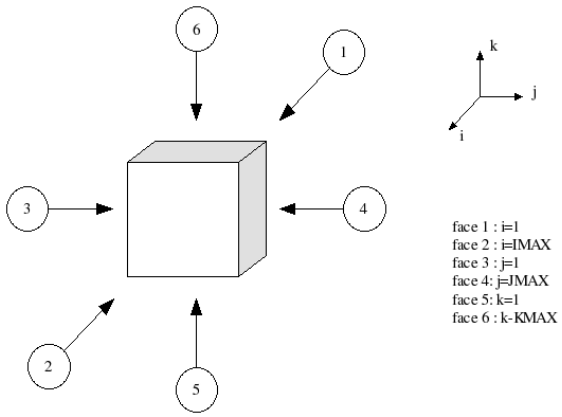

Fig. 6 Definition of the meshes faces for the boundary conditions.

5 Treatment of Patched Grid Interfaces

In this present work, a patched grid block always shares a common faces of points with other patched grid block, as indicated in Fig. 7. In order to illustrate this procedure, it is assumed that there are two meshes, denominated *A* and *B*, as presented in Fig. 7. Those meshes should be expanded, in order to allow the implementation of a code with the capacity of transferring information through the common faces. It is desirable to maintain the order of the artificial dissipation operators at all points. Therefore taking into account that the artificial dissipation operators use 5 points, a possible solution is to expand the meshes so that there is an area of 5 rows of points in common, as indicated in the right side of Fig. 7. It can be observe that 2 rows of points were added to each mesh, which caused the displacement of the first columm of points. The following steps are executed:

1. Initially, the properties of all interior points located in the expanded mesh *A* are calculated,advancing one step in time.
2. The points located in the first column of the mesh *B* receive the values of the properties of the

points of the fifth column of the mesh *A*.
3. The points located in the second column of the mesh *B* receive the values of the properties of the points of the fourth column of the mesh *A*.
4. All the interior points of the mesh *B* are calculated, advancing one step in time for that mesh.
5. The values of the points located in the fifth column of the mesh *B* are transferred for the first column of the mesh *A*, and values of the fourth column of the mesh *B* for the second column of the mesh *A*.
6. The interior points of the mesh *A* are calculated again and the process repeats.
The third column of the two meshes is left "free" and its value is determined by the calculation of the interior points, without any imposition of value, as it happened with the first and the second columns. Attempts of imposing some value in the third column, as for example an average among the two meshes, resulted in a significant decrease of the convergence rate. In 3-D, instead of lines or columns, the meshes have planes in common. In the present paper, the meshes are built with a single face in common, and an additional code takes charge of reading a connection matrix to decide which faces of each mesh should be expanded.

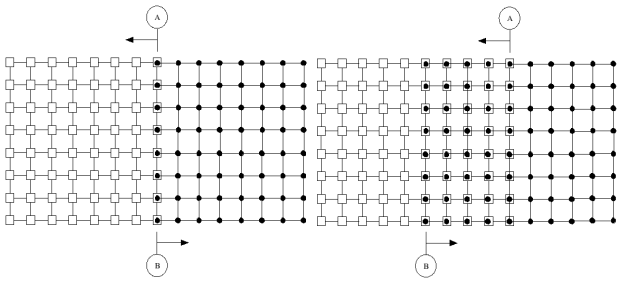

Fig. 7 Meshes *A* and *B* before (left) and after (right) the expansion process.

6 The Chimera Holecutting Process

The Chimera grid possesses a sobreposition area but, unlike the patched grid, there is no need for the points to coincide. Again, that area is responsible for the change of information among the meshes. However, as not all of the points are necessary for the communication among the meshes, we can logically eliminate some points. Actually, all of the points continue to exist in the computer memory. We create an auxiliary matrix that associates to each point of the mesh an on value or an off value. The points are eliminated by two reasons. The first one concerns the fact that points of a certain mesh are located inside an area without physical meaning of another mesh as, for example, inside a body of some other component of the configuration. An example would be the points of the m_8 mesh that are located inside the front fin. The left side of Fig. 8 exhibits the m_8 mesh with the points eliminated and the outline of the fin. In practice, a virtual volume larger than the solid volume is created, and all points of the mesh that are inside the virtual volume are eliminated. The creation of the virtual volume makes possible the control of the amount of points to be eliminated. The second reason to eliminate points is to reduce the sobreposition area. An example would be the central body mesh m_{13}, that contains the two *background* meshes m_8 and m_{12}. A virtual volume completely contained in a *background* mesh is created and all points of the mesh that are inside this volume are eliminated. The right hand side in Fig. 8 displays the result of this process.

After the *holecutting* process, described in Antunes [11], the next step consists in identifying the Chimera boundary points. These points are the ones that were not eliminated by the pre-

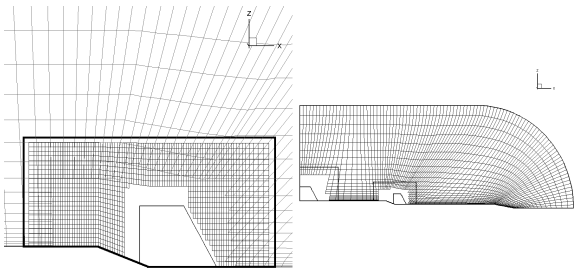

Fig. 8 Detail of the *background* m_8 mesh (left) and central body mesh m_{13} (right) after the holecutting process.

vious process but they have at least one neighbor that was eliminated. The Chimera boundary points are not calculated in the same way that the other interior points. They have the values of their properties interpolated. Each Chimera boundary point is located inside of an hexahedron whose vertices are formed by points of the other Chimera grid. As described in Antunes [12], the distances between a Chimera boundary point of the first mesh and each of the eight vertices of the second mesh are calculated, respectively. It should be emphasized that there is no attempt to satisfy conservation in the present interpolation process. Since shocks may be crossing the interface, it would be interesting to have the enforcement of some conservation statement at grid interfaces. However, this was not implemented in the present case due to the high computational costs associated with such implementation, especially in the 3-D case, and because the present effort should be seen as an evolutionary step towards a more complete simulation capability. Furthermore, the use of a conservative interpolation process would certainly increase the code's memory requirements, which the authors would like to avoid at this time. An interpolation method at the interfaces among Chimera meshes that satisfies conservation was developed by Wang [13]. A detailed discussion of the procedure can be found in more details in Wang [14].

7 Results and Discussion

The results presented refer to simulations of the flow over the SONDA-III rocket during its first stage flight. Preliminary results for this configuration have been presented in Papa [15]. In the cited reference, however, the total number of grid points was of the order of 275,000 points, which did not allow for a more detailed visualization of some critical regions of the flow about the fins. The present work has performed similar simulations, however considering a much finer mesh with approximately 917,000 grid points. Such a level of grid refinement allows for a considerably better visualization of flow details about the configuration. The specific results included here consider only the case with freestream Mach number M_∞=2.0 and zero angle of attack, which is representative of the simulations performed so far for the configuration. Moreover, as the flight time in the lower atmosphere for these rockets is very short and the vehicle is at supersonic speeds during most of this flight, it seems appropriate to select a supersonic flight condition for the present discussion. Within the supersonic speed regime, several interesting aspects of the Chimera and patched grid techniques can be analyzed, such as the communication of information across the internal boundaries among blocks with discountinuities in the flow properties. Mach number contours on the vehicle body in regions around the front and back fins can be observed in Fig. 9. Pressure contours for the sounding rocket, its front and back fins can be observed in Fig. 10. As previously mentioned, the major interest in this work concerns the evaluation of the joint use of the Chimera and patched grid techniques as a tool for flow analysis over geometries of interest for the Instituto de Aeronáutica e Espaço.

The Mach number contours can be observed in Fig. 9. This figure shows in detail the region of interaction between the front and back fins and the conical region of the central body. One can observe that the thickness of the shock in the leading edge of the front fins increases when it approaches the plane of symmetry. Figure 10 shows in detail the pressure contours in the region of front fins. Through this figure one can observe the formation of the shock waves in the leading edges of the fins as well as the expansion waves in the trailing edges of these fins. In the conical region, an increase of pressure due a shock wave that cannot be observed in this visualization. It can also be observed that, after the conical region, the pressure decreases due to an expansion wave in the intersection between the conical and cylindrical regions.

8 Concluding Remarks

The paper has presented results for 3-D Euler simulations of the flow over the SONDA-III, a typical sounding rocket. A structured multiblock

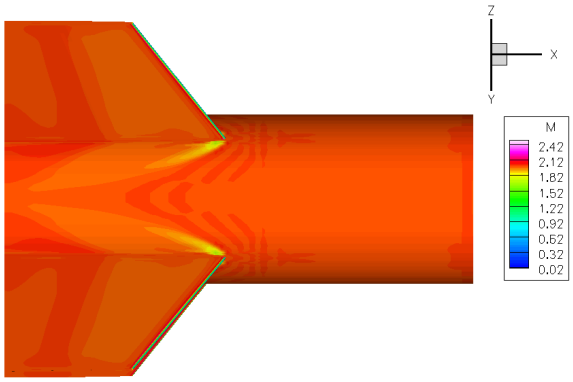

Fig. 9 Mach number contours for the back fins

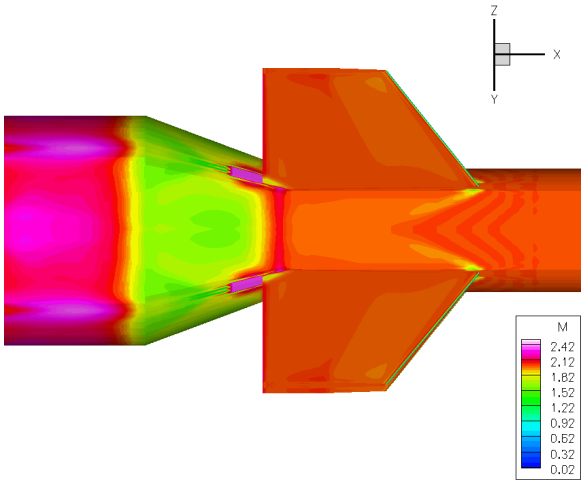

Fig. 10 Mach number contours for the front fins

code has been implemented, using Chimera and patched multiblock grid approaches for handling the geometric complexity of the configuration. All codes used were developed by the research group and represent a powerful aerodynamic analysis and design tool. The present methodology, using a combination of Chimera and patched grid approaches, seems to be quite powerful to work on similar problems with the presence of several fins. The main noticed advantages are:

1. Flexibility: the joint use of the Chimera and patched grids, in the same simulation, allows the generation of meshes in a much easier fashion and much less time.
2. Modularity: the use of background meshes, hiding the complexity of meshes that involve the fins, also seems to be a very interesting approach for these configurations. With such an approach, small grid modifications, or even small configuration modification, are quite simple in the sense that one does not need to generate all the meshes or, even, to reschedule the flow of information.
3. Point concentration: the use of multiblock meshes allows the refinement of localized regions in a way very similar to unstructured meshes.

Finally, the power of this combined Chimera and patched grid simulation capability becomes evident when one considers that it was possible to simulate the flow over a complete sounding rocket and, at the same time, to capture details of phenomena occurring along the trailing edge of the frontal fins. This indicates that the methodology presented allowed grid refinement characteristics similar to those found in unstructured meshes, without the inconvenience of indirect addressing as described in Long [16].

9 Acknowledgements

The authors acknowledge the partial support of Conselho Nacional de Desenvolvimento Científico e Tecnológico, CNPq, through the Integrated Project Research Grant No. 501200/2003-7. The authors are also indebted to CNPq for the direct support of the second author through the Undergraduate Scholarship CNPq Process No. 114698/03-1

10 References

[1] Azevedo, J.L.F., Menezes, J.C.L., and Fico Jr., N.G.C.R., “An Assessment of Boundary Layer Properties for Transonic and Supersonic Flows over the VLS”, AIAA Paper No. 95-1769-CP, Proceedings of the 13th AIAA Applied Aerodynamics Conference, vol. 1, pp. 41-51, San Diego, CA, 1995.

[2] Azevedo, J.L.F., Menezes, J.C.L., and Fico

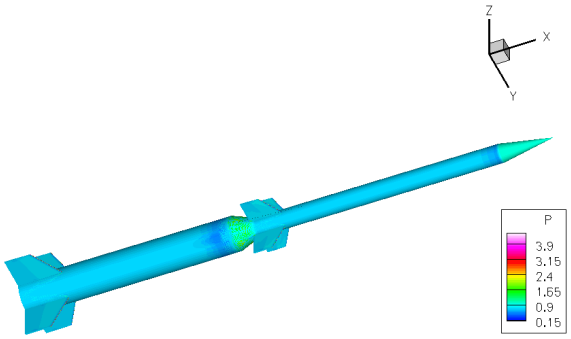

Fig. 11 Pressure contours on the surface sounding rocket

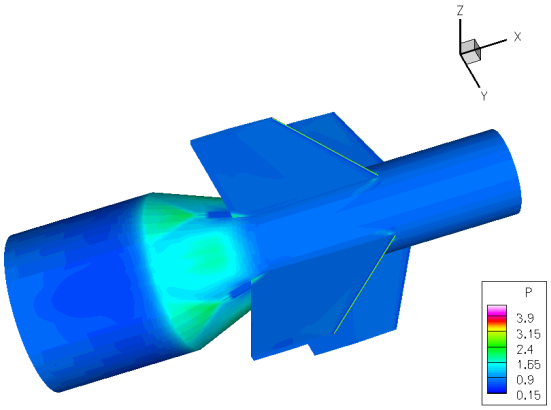

Fig. 12 Pressure contours for the front fins

Jr., N.G.C.R.,"Accurate Turbulent Calculations of Transonic Launch Vehicles Flows",AIAA Paper No. 96-2484-CP, Proceedings of the 14th AIAA Applied Aerodynamics Conference, vol. 3, pp. 841-851, New Orleans, LA, 1996.

[3] Azevedo, J.L.F., Strauss, D., and Ferrari, M.A.S., "Viscous Multiblock Simulations of Axisymetric Launch Vehicle Flows",AIAA Paper No. 97-2300-CP, Proceedings of the 15th AIAA Applied Aerodynamics Conference, vol. 2, pp. 664-674, Atlanta, GA, 1997.

[4] Strauss, D., and Azevedo, J.L.F., "A Numerical Study of Turbulent Afterbody Flows Including a Propulsive Jet", AIAA Paper No. 99-3190-CP, Proceedings of the 17th AIAA Applied Aerodynamics Conference, pp. 654-664, Norfolk, VA, 1999.

[5] Jameson, A., Schmidt, W., and Turkel, E., "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes", AIAA Paper No. 81-1259, 1981.

[6] Turkel, E., and Vatsa, V.N., "Effect of Artificial Viscosity on Three-Dimensional Flow Solutions", AIAA Journal, No. 1, vol. 32, pp 39-45, 1994.

[7] Vieira, R., Azevedo, J.L.F., Fico Jr. N.G.C.R., and Basso, E., "Three Dimensional Flow Simulation in the Test Section of a Slotted Transonic Wind Tunnel", ICAS Paper No. 98-R.3.11, Proceedings of the 21st Congress of the International Council of the Aeronautical Sciences, Melbourne, Australia, 1999.

[8] Pulliam, T.H., and Steger, J.L., "Implicit Finite-Difference Simulations of Three-Dimensional Compressible Flow", AIAA Journal, No. 2, vol. 18, pp. 159-167, 1980.

[9] Fletcher, C.A.J., "Computational Techniques for Fluid Dynamics, Vol II", Spring-Verlag, New York, 1988.

[10] Basso, E., Antunes, A.P., and Azevedo, J.L.F., "Chimera Simulations of Supersonic Flows over a Complex Satellite Launcher Configuration", Journal of Spacecraft and Rockets, No. 3, vol. 40, pp. 345-355, 2003.

[11] Antunes, A.P., Basso, E., and Azevedo, J.L.F., "Holecut-A Program for the Generation of Chimera Grids", Instituto de Aeronáutica e

Espaço, CTA/IAE/ASE-N, São José dos Campos, SP, 2000.

[12] Antunes, A.P., “Simulation of Aerodynamic Flows Using Overset Multiblock Grids”, Masters Thesis, Instituto Tecnológico de Aeronáutica, São José dos Campos, SP, Brasil, 2000.

[13] Wang, Z.J., Buning, P., and Benek, J., “Critical Evaluation of Conservative and Non-Conservative Interface Treatment for Chimera Grids”, AIAA Paper No. 95-0077, 33nd AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, 1995.

[14] Wang, Z.J., and Yang, H.Q., “A Unified Conservative Zonal Interface Treatment for Arbitrarily Patched and Overlapped Grids”, AIAA Paper No. 94-0320, 32nd AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, 1994.

[15] Papa, J.C., and Azevedo, J.L.F., “Three Dimensional Flow Simulations Over a Typical Sounding Rocket”, AIAA Paper No. 2003-3421, Proceedings of the 21st AIAA Applied Aerodynamics Conference, Orlando, FL, 2003.

[16] Long, L.N., Khan, M.S., and Sharp, H.T., “Massively Parallel Three-Dimensional Euler/Navier-Stokes Method”, AIAA Journal, No. 5, vol. 29, pp. 657-666, 1991.