CODE VALIDATION FOR HIGH SPEED FLOW SIMULATION OVER THE VLS LAUNCHER FAIRING

J.L.F. Azevedo†, P. Moraes Jr.‡
Instituto de Aeronáutica e Espaço
CTA/IAE/ASE-N
12228-904 São José dos Campos – SP – BRAZIL

C.R. Maliska*, C.H. Marchi‡‡, A.F.C. Silva**
Dept. Eng. Mecânica – Universidade Federal de Santa Catarina
Caixa Postal 476
88040-900 Florianópolis – SC – BRAZIL

Abstract

The details of an experimental wind tunnel investigation for pressure distribution and force measurements for the Brazilian VLS are described. The development of an all speed Euler and/or Navier-Stokes flow simulation code which uses a segregated finite volume algorithm for 3-D body conforming curvilinear coordinates with a colocated variable arrangement is also described. The efforts towards the physical validation of this code are the major contribution of the present work. The results obtained for subsonic and supersonic flow conditions are in very good agreement with the experimental data. Transonic calculations, however, show a much poorer agreement with the available data for equivalent and even more refined meshes. The computational results produced by the codes here described are already supplying aerodynamic information which is currently being used in the VLS design process.

Introduction

The first Brazilian Satellite Launch Vehicle (VLS) is of the cluster type with four strap-on boosters around the central core, as shown in Fig. 1. The hammerhead type fairing is of non-conventional shape, but its use is rather common on satellite launchers in order to accommodate spacecraft with a diameter larger than the last boosting stage. The prediction of the local aerodynamic characteristics of such type of fairing, for shape optimization and design purposes, is typically made primarily through wind tunnel tests. These tests, however, require high costs and usually a long turnaround time for the complete cycle, which encompasses test specification, model production, the test itself and data reduction. Therefore, there is great interest and a lot of effort has been put into developing the capability of simulating such aerodynamic flowfields. We are particularly interested in accurately computing the high speed flow about the hammerhead fairing, and so far only steady state cases have been considered.

† Head, CFD Group, Member AIAA
‡ Head, Aerodynamics Subdivision
* Professor, Member AIAA
‡‡ Graduate Student
** Associate Professor

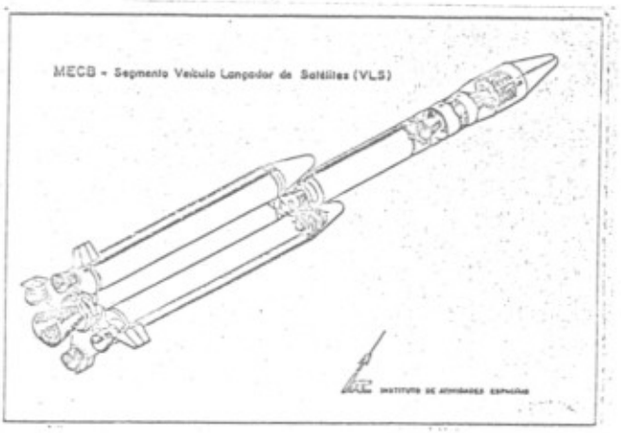

Figure 1: General configuration of the VLS vehicle.

Both for actual design and for code validation purposes, a wind tunnel test program was established and performed. This has covered a wide range of test parameters such as Mach number, Reynolds number and angle of incidence. The test program has attempted to cover the relevant aerodynamic regimes for the complete expected flight trajectory of the vehicle. Moreover, the test program was organized in such a way that the experimental data so obtained could be readily used for Computational Fluid Dynamics (CFD) code development and validation studies. In particular, although the configuration tested was inherently complex, care was exercised in order to avoid booster attachment details and body external protuberances, which would be extremely difficult to numerically simulate without severely penalizing the size of the computational grids.

The institutions here represented have been actively involved in the development of high speed flow simulation codes for several years now. It is hoped that these codes will help to ease the burden presently put onto the experimental investigations for the VLS aerodynamic design. However, before these codes can be used in a design environment, they must be thoroughly validated. Moreover, although of a more subjective value, project engineers have to develop confidence in the use of the codes for their analysis and design work. Two quite distinc-

tive lines of work have been pursued. The first one was based on the methods typically used in aerodynamics in which the governing equations are solved simultaneously with local linearization based on Jacobian matrices. The other was based on segregated solution methods which were originally developed for low speed flows and heat transfer problems. The emphasis here will be on the second class of methods, since a novel methodology for flow simulation at all speeds has been recently developed by some of the present authors[1,2,3] in this context.

The present work will briefly describe the experimental investigation conducted and the code development, and it will concentrate on the physical validation of the results produced by the new code against the experimental data. Code-to-code comparisons are also performed, using independent computational results generated by a central difference, implicit, approximate factorization algorithm which has been previously tested for similar configurations[4,5]. Further details of the experimental test program are reported by Moraes and Neto[6]. More details of the numerical formulation in the new segregated, finite volume code are described by Maliska and co-workers[1,2,3,7].

Experimental Test Program

The general wind tunnel test program comprised three different test series, although we will only be concerned with the first two in the present work. These two considered the acquisition of global and local vehicle characteristics through the measurement of forces, moments and local pressures along the vehicle. The third test series considered the simulation of lift-off conditions in a low speed wind tunnel. The computational simulation of lift-off conditions has not been attempted yet and, therefore, the results of this third test series will be of no concern for the present work. Moreover, our major interest in the present case will be the local pressure measurements reported in Ref. [8].

Pressure measurements tests were performed both in a continuous type transonic wind tunnel and in a blow-down supersonic tunnel. The continuous transonic tunnel has a test section size of $1.75 \times 1.77 m^2$ and it uses perforated walls for tests in the transonic speed regime. Tests were conducted in this tunnel for the Mach number range from 0.5 up to 2.5. Tests in the Mach number range $2.5 \leq M_\infty \leq 3.75$ were performed in the blow-down tunnel. For the measurements, a 1 : 15 scale smooth model with non-attached boosters has been used. It was configured with approximately 320 pressure taps distributed along the surface of the vehicle's central core and boosters. The model was held in the test section using a five-sting support system, which would allowed even the simulation of the strap-ons separation in a static manner. A sketch of the five-sting support system is indicated in Fig. 2. Moreover, with the objective of increasing the amount of azimuthal pressure information without exceeding the available internal space of the model, pressure taps were distributed along some azimuthal positions on the core and on the boosters, and the model was tested at two

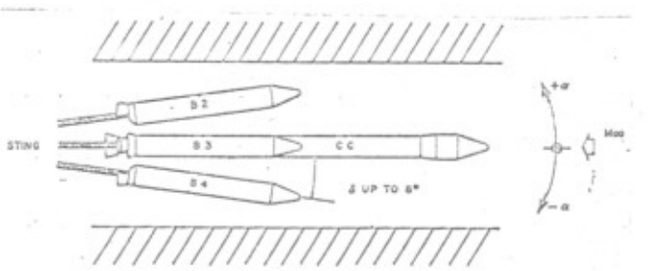

Figure 2: Sketch of the wind tunnel model with the five-sting support system.

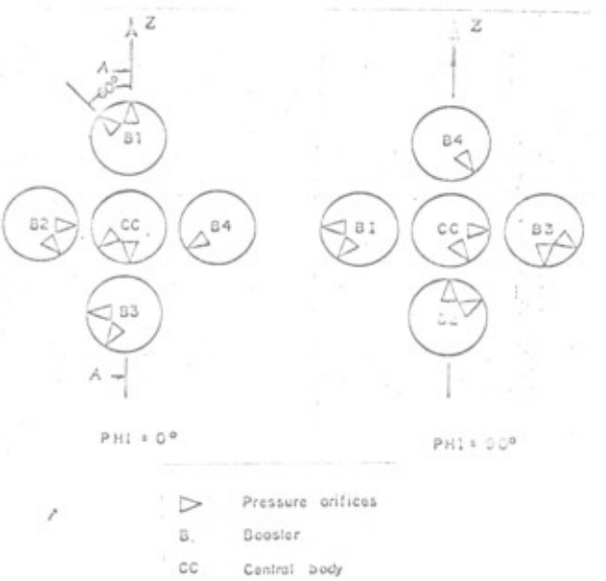

Figure 3: Front view of the pressure taps distribution in the azimuthal direction and indication of the two model positions used for testing.

different positions rotated by 90° with respect to each other. We call these as corresponding to $\varphi = 0°$ and $\varphi = 90°$. This is indicated in Fig. 3 where a front view of the model is sketched. Since all tests were performed for positive and negative angles of attack, this distribution of pressure tap lines allowed measurements every 30° in the azimuthal direction, from lee- to windside, for all bodies involved. It is also important to mention that, for every Mach number, tests were performed from $-6°$ to $+6°$ in angle of attack at 2° intervals.

One of the major thrusts for the specification of the five-sting support was precisely the need for good experimental data for code validation. Since it is very difficult to computationally model all the details of actual booster attachments, we have decided to perform experimental tests without these attachments in order to provide the computational specialists with clean data with which to compare their results. Moreover, it is fairly clear that in a code validation effort one should not unnecessarily complicate the geometry because the grid, and the corresponding computational time, can become very large.

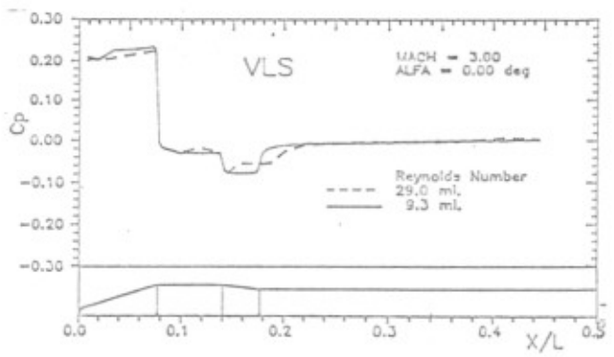

Figure 4: Reynolds number influence on the vehicle's longitudinal pressure coefficient distribution.

The solution found was, then, to support each individual body, i.e., central core and four boosters, with its own sting. Certainly, the same system can be used to test the vehicle simulating the condition of strap-on separation, as previously mentioned.

Finally, it is worth observing that the wind tunnel tests have also contemplated the problem of Reynolds number influence in the flow topology, i.e., the effect of separation and reattachment points in the flowfield. In particular, experimental results have been able to clearly indicate the effect of the Reynolds number in the flow separated region in the boattail. This is shown in Fig. 4 for a $M_\infty = 3$ and zero angle of attack case, and considering Reynolds numbers (based on the model length) of 9.3×10^6 and 29.0×10^6. This is a very important effect because flow separation in the boattail can adversely modify the buffeting behavior of the vehicle. Therefore, it is obviously desirable that flow simulation codes be able to capture such Reynolds number influence. As we will discuss later, the present viscous flow simulation capability implemented is still not able to capture such effects, and further work is certainly necessary in this area.

The Central Difference Algorithm

The compressible Euler equations[9] can be written in strong conservation-law form for general three dimensional, body-conforming, curvilinear coordinates[10] 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 the vector of conserved quantities, $\overline{Q}$, is defined as

$$\overline{Q} = J^{-1} \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ e \end{Bmatrix} \qquad (2)$$

The flux vectors $\overline{E}$, $\overline{F}$ and $\overline{G}$ can be written as

$$\overline{E} = J^{-1} \begin{Bmatrix} \rho U \\ \rho u U + p\xi_x \\ \rho v U + p\xi_y \\ \rho w U + p\xi_z \\ (e+p)U - p\xi_t \end{Bmatrix} \qquad (3)$$

$$\overline{F} = J^{-1} \begin{Bmatrix} \rho V \\ \rho u V + p\eta_x \\ \rho v V + p\eta_y \\ \rho w V + p\eta_z \\ (e+p)V - p\eta_t \end{Bmatrix} \qquad (4)$$

$$\overline{G} = J^{-1} \begin{Bmatrix} \rho W \\ \rho u W + p\zeta_x \\ \rho v W + p\zeta_y \\ \rho w W + p\zeta_z \\ (e+p)W - p\zeta_t \end{Bmatrix} \qquad (5)$$

In the above equations, the usual nomenclature is being used. Therefore, ρ is the density, u, v and w are the cartesian components of velocity, and e is the total energy per unit of volume. The equations have been nondimensionalized following the work of Pulliam and Steger[10,11].

The pressure, p, can be obtained by the equation of state for perfect gases

$$p = (\gamma - 1)\rho e_i = (\gamma - 1)\left[e - \frac{1}{2}\rho\left(u^2 + v^2 + w^2\right)\right] \qquad (6)$$

where e_i is the specific internal energy of the fluid, and γ is the ratio of specific heats. The contravariant velocity components are defined as

$$\begin{aligned} U &= \xi_t + \xi_x u + \xi_y v + \xi_z w \\ V &= \eta_t + \eta_x u + \eta_y v + \eta_z w \\ W &= \zeta_t + \zeta_x u + \zeta_y v + \zeta_z w \end{aligned} \qquad (7)$$

Throughout this work, the curvilinear coordinate system is defined such that ξ is the longitudinal direction, η is the normal direction, and ζ is the circumferential direction. This coordinate system is obtained from the transformation of variables

$$\begin{aligned} \tau &= t \\ \xi &= \xi(x, y, z, t) \\ \eta &= \eta(x, y, z, t) \\ \zeta &= \zeta(x, y, z, t) \end{aligned} \qquad (8)$$

The Jacobian of the transformation, J, can be expressed as

$$\begin{aligned} J = \; & (x_\zeta y_\eta z_\zeta + x_\eta y_\zeta z_\xi + x_\zeta y_\xi z_\eta \\ & - x_\xi y_\zeta z_\eta - x_\eta y_\xi z_\zeta - x_\zeta y_\eta z_\xi)^{-1} \end{aligned} \qquad (9)$$

Expressions for the various metric relations can be found, among other references, in Pulliam and Steger[10,11].

In the present case, the above governing equations were implemented through the use of finite difference methods. The implicit Euler method was used for the time-march,

and the spatial derivatives were approximated by three-point, second order central differences. The Beam and Warming implicit approximate factorization scheme[12,13] was used for the solution of the resulting finite difference equations in order to obtain a cost efficient algorithm. The resulting scheme is second order accurate in space, as mentioned, but it is only first order accurate in time due to the use of the implicit Euler method.

The factored finite difference equations can be written in the delta form as

$$L_\eta L_\zeta L_\xi \Delta_t \overline{Q}^n = R_\xi + R_\eta + R_\zeta \qquad (10)$$

The various operators are defined as

$$\begin{aligned}
L_\xi &= \left(I + \Delta t \delta_\xi \hat{A}^n - \epsilon_I \Delta t J^{-1} \nabla_\xi \Delta_\xi J\right) \\
L_\eta &= \left(I + \Delta t \delta_\eta \hat{B}^n - \epsilon_I \Delta t J^{-1} \nabla_\eta \Delta_\eta J\right) \\
L_\zeta &= \left(I + \Delta t \delta_\zeta \hat{C}^n - \epsilon_I \Delta t J^{-1} \nabla_\zeta \Delta_\zeta J\right) \\
R_\xi &= -\Delta t \delta_\xi \overline{E}^n - \epsilon_E \Delta t J^{-1} \left(\nabla_\xi \Delta_\xi\right)^2 J \overline{Q}^n \\
R_\eta &= -\Delta t \delta_\eta \overline{F}^n - \epsilon_E \Delta t J^{-1} \left(\nabla_\eta \Delta_\eta\right)^2 J \overline{Q}^n \\
R_\zeta &= -\Delta t \delta_\zeta \overline{G}^n - \epsilon_E \Delta t J^{-1} \left(\nabla_\zeta \Delta_\zeta\right)^2 J \overline{Q}^n
\end{aligned} \qquad (11)$$

In the above, δ_ξ, δ_η and δ_ζ are central difference operators; ∇_ξ, ∇_η and ∇_ζ are backward difference operators; and Δ_ξ, Δ_η and Δ_ζ are forward difference operators in the ξ- , η- and ζ-directions, respectively. As an example,

$$\begin{aligned}
\delta_\xi \overline{Q}^n_{i,j,k} &= \frac{1}{2}\left[\overline{Q}^n_{i+1,j,k} - \overline{Q}^n_{i-1,j,k}\right] \\
\nabla_\xi \overline{Q}^n_{i,j,k} &= \overline{Q}^n_{i,j,k} - \overline{Q}^n_{i-1,j,k} \\
\Delta_\xi \overline{Q}^n_{i,j,k} &= \overline{Q}^n_{i+1,j,k} - \overline{Q}^n_{i,j,k}
\end{aligned} \qquad (12)$$

The Δ_t is a forward difference operator in time given by

$$\Delta_t \overline{Q}^n = \overline{Q}^{n+1} - \overline{Q}^n \qquad (13)$$

Artificial dissipation terms have been introduced in the operators described by Eq. 11 in order to maintain the stability of the numerical solution process. Fourth order numerical dissipation terms were added to the right-hand side operators, and second order terms were used in the left-hand side operators. From an accuracy standpoint, one would like to also use fourth order artificial dissipation in the implicit operators. However, computational efficiency constraints prevent such use. The Jacobian matrices $\hat{A}^n$, $\hat{B}^n$ and $\hat{C}^n$ are described in detail elsewhere in the literature (see, for instance, Pulliam and Steger[11]).

The Segregated All Speed Scheme

Preliminary Considerations

Most of the available algorithms for the solution of the Euler, or the Navier-Stokes, equations are only suitable for either compressible or incompressible flows. Although there is no universal agreement as to the cause of this behavior, there are several authors who believe that the key point is associated with the form in which density is treated in the mass conservation equation. In the present work, a numerical method for the solution of three-dimensional, viscous or inviscid, all speed flows of a perfect gas is considered. The method is developed in the framework of a boundary-fitted, structured, finite volume spatial discretization, and it uses a fully implicit time march procedure. The system of equations is solved in a segregated manner in which one of the dependent variables is assumed to be the active one in each of the governing equations. All other variables in that equation assume a passive role in the linearization process. Primitive variables are used as dependent variables. Moreover, a colocated variable arrangement is employed which renders compactness to the method and considerable savings in the storage of geometric information concerning the grid system.

The method is derived from the well known approach used for incompressible flows[14], whereby the mass conservation equation is transformed into an equation to find pressure. In this equation, density and velocity components are replaced by relations involving pressure obtained from the equation of state and from approximate forms of the momentum equations, respectively. This allows both velocity and density to remain active in the continuity equation[1] and, therefore, enables the method to treat both compressible and incompressible flow problems. It should be pointed out that a similar approach has been presented by Karki and Patankar[15]. Here, however, the procedure developed by Maliska and his co-workers[1,2,3] is employed. Moreover, the present version of the code has incorporated the ability of considering multiple block grids. The difficulty in discretizing such complex computational domains such as the flow-field about the VLS vehicle has indicated the need for such procedure.

Formulation of the Method

The segregated all speed finite volume scheme will be described here as applied to the Navier-Stokes equations. For its application to the Euler equations, one simply has to neglect the viscous terms and make obvious changes on the wall boundary conditions. In the nomenclature that is usually used with segregated finite volume schemes, the Navier-Stokes equation can be written for general, body-conforming, curvilinear coordinates as

$$\begin{aligned}
&\frac{1}{J}\frac{\partial}{\partial t}(\rho\phi) + \frac{\partial}{\partial \xi}(\rho U \phi) + \frac{\partial}{\partial \eta}(\rho V \phi) + \frac{\partial}{\partial \zeta}(\rho W \phi) = \\
&\Gamma^\phi \frac{\partial}{\partial \xi}\left[\alpha_{11} J \frac{\partial \phi}{\partial \xi} + \alpha_{12} J \frac{\partial \phi}{\partial \eta} + \alpha_{13} J \frac{\partial \phi}{\partial \zeta}\right] + \\
&\Gamma^\phi \frac{\partial}{\partial \eta}\left[\alpha_{12} J \frac{\partial \phi}{\partial \xi} + \alpha_{22} J \frac{\partial \phi}{\partial \eta} + \alpha_{23} J \frac{\partial \phi}{\partial \zeta}\right] + \\
&\Gamma^\phi \frac{\partial}{\partial \zeta}\left[\alpha_{13} J \frac{\partial \phi}{\partial \xi} + \alpha_{23} J \frac{\partial \phi}{\partial \eta} + \alpha_{33} J \frac{\partial \phi}{\partial \zeta}\right] - \hat{p}^\phi + \hat{S}^\phi
\end{aligned} \qquad (14)$$

The general form given in Eq. 14, with the appropriate various source terms, can recover the continuity equation, the three momentum equations, and the energy equation.

For that, ϕ must be chosen as 1, u, v, w, and T, respectively. J is the Jacobian of the coordinate transformation, and U, V and W are the contravariant velocity components. The expression for the Jacobian is given in Eq. 9, and the contravariant velocity components are defined in the present context as

$$\begin{aligned} U &= \frac{1}{J}\left(\xi_x u + \xi_y v + \xi_z w\right) \\ V &= \frac{1}{J}\left(\eta_x u + \eta_y v + \eta_z w\right) \\ W &= \frac{1}{J}\left(\zeta_x u + \zeta_y v + \zeta_z w\right) \end{aligned} \tag{15}$$

Expressions for the various metric terms can be found in the same references previously given. The α_{11}, α_{12}, ...metric coefficients involve the product of the appropriate metric terms, and they can be found in Maliska et al.[16] among other references. The source terms are defined as

$$\begin{aligned} \hat{p}^u &= \frac{1}{J}\left(\frac{\partial p}{\partial \xi}\xi_x + \frac{\partial p}{\partial \eta}\eta_x + \frac{\partial p}{\partial \zeta}\zeta_x\right) \\ \hat{p}^v &= \frac{1}{J}\left(\frac{\partial p}{\partial \xi}\xi_y + \frac{\partial p}{\partial \eta}\eta_y + \frac{\partial p}{\partial \zeta}\zeta_y\right) \\ \hat{p}^w &= \frac{1}{J}\left(\frac{\partial p}{\partial \xi}\xi_z + \frac{\partial p}{\partial \eta}\eta_z + \frac{\partial p}{\partial \zeta}\zeta_z\right) \\ \hat{S}^u &= \frac{\mu}{3J}\left[\frac{\partial}{\partial \xi}\left(\nabla.\vec{V}\right)\xi_x + \frac{\partial}{\partial \eta}\left(\nabla.\vec{V}\right)\eta_x + \frac{\partial}{\partial \zeta}\left(\nabla.\vec{V}\right)\zeta_x\right] \\ \hat{S}^v &= \frac{\mu}{3J}\left[\frac{\partial}{\partial \xi}\left(\nabla.\vec{V}\right)\xi_y + \frac{\partial}{\partial \eta}\left(\nabla.\vec{V}\right)\eta_y + \frac{\partial}{\partial \zeta}\left(\nabla.\vec{V}\right)\zeta_y\right] \\ \hat{S}^w &= \frac{\mu}{3J}\left[\frac{\partial}{\partial \xi}\left(\nabla.\vec{V}\right)\xi_z + \frac{\partial}{\partial \eta}\left(\nabla.\vec{V}\right)\eta_z + \frac{\partial}{\partial \zeta}\left(\nabla.\vec{V}\right)\zeta_z\right] \\ \hat{S}^T &= \frac{1}{JC_p}\left[\frac{\partial p}{\partial t} + \nabla.\left(p\vec{V}\right) - p\left(\nabla.\vec{V}\right)\right] \end{aligned} \tag{16}$$

where

$$\nabla.\vec{V} = J\left[\frac{\partial U}{\partial \xi} + \frac{\partial V}{\partial \eta} + \frac{\partial W}{\partial \zeta}\right] \tag{17}$$

We emphasize that all viscous simulations performed here were laminar calculations.

Since we are considering low enough angles of attack, the flowfield is symmetric about the pitch plane. Therefore, the computational solution domain spans only 180° in the circumferential direction around the vehicle, going from the leeward to the windward plane. Flow symmetry conditions are, then, imposed at both lee- and windward planes. At the body surface, no-slip conditions are enforced for the viscous cases and flow tangency conditions are considered for the inviscid cases. Moreover, the wall is assumed to be adiabatic. Freestream conditions are prescribed at the computational entrance surface. At the exit plane, all properties are obtained by extrapolation of interior information. At the upstream stagnation line, it turns out that no numerical boundary conditions are actually required. Freestream conditions are used as initial conditions for the simulations performed here.

Eq. 14 is discretized using a control volume method (see, for instance, Patankar[14]). The mass conservation equation is linearized in such a way that maintains both density and velocity as unknowns[17], therefore allowing the solution of incompressible as well as compressible flows. Through the use of the SIMPLEC method[18] for the pressure-velocity coupling, the continuity equation is used for the calculation of pressure, the equation of state is used to obtain the density, and the three momentum equations plus the energy equation are used in order to obtain the other quantities (u, v, w and T). A co-located variable arrangement is employed in the present work. Further details of the numerical methodology used here can be seen in Marchi et al.[2,3].

The solution procedure that will be described next assumes a single grid block. The necessary modifications in this procedure in order to accommodate a multiblock strategy will be discussed in the next section. Once initial values for the six state variables are known, the solution procedure adopted in the present work takes the following steps. **1.** Estimate of the u, v, w, p, T and ρ fields at instant $t + \Delta t$. **2.** Computation of the coefficients for the three momentum equations. **3.** Computation of the coefficients for the continuity equation. **4.** Computation of the source terms for u, v and w. **5.** Solution of the momentum equations. This step determines new velocity components u^*, v^* and w^* which do not necessarily conserve the mass. **6.** Evaluation of the contravariant velocity components U^*, V^* and W^*. **7.** The error, or the residue, in the continuity equation is computed using the available contravariant velocity components and density field. **8.** A correction to the pressure field is determined using the coefficients evaluated in step (3) and the residues determined in step (7). **9.** Velocity components and densities are corrected by the new pressure field. The resulting fields conserve mass. **10.** Computation of the coefficients and source terms for the energy equation. **11.** Calculation of a new temperature field. **12.** Computation of the density as function of pressure and temperature. **13.** Return to step (1) and iterate until the steady state is reached.

The solution process, as presented above, does not involve any iteration cycle within each time interval. However, due to the type of linearization adopted and due to the coupling scheme implemented, some steps must be executed more than once within each time step. In the present work, the computations associated with steps (3) through (9) were usually executed twice for each time interval. This inner iteration cycle, which has no meaning for incompressible problems, allows the use of larger time steps. Finally, we must emphasize that steps (5), (8) and (11) involve the solution of linear systems.

Multiblock Technique Implementation

The multiblock implementation used in the present work was actually developed in two phases. The initial phase assumed that there would be a perfect match of the boundary control volumes on two adjacent grid blocks. Later, this has been extended to the general case in which the size of the control volumes in two adjacent blocks do not have any specific relation between them.

Moreover, it is important to emphasize that the present approach assumes no overlap of adjacent grid blocks. In other words, adjacent grid blocks simply touch each other at their common interface. We will start our discussion here with the simpler case because it is instrumental to understand the procedure adopted for the general case. We should also emphasize that the important point to be discussed is the transfering of information from one grid block to its neighbor, since, within each grid block, the solution process is precisely as previously described.

The present finite volume approach is using "fictitious", or "slave", control volumes in order to implement boundary conditions in each grid block. These are control volumes which are located outside the computational domain of interest (for each block) and that serve the solely purpose of implementing boundary conditions. Hence, our previous statement that grid blocks do not overlap should be well understood here. We mean that the "actual" control volumes do not overlap, but the "fictitious" volumes of one grid block are overlapping a few "actual" volumes of the adjacent grid block. That is, when solving for a given grid block, the information on its fictitious volumes which correspond to a boundary with another grid block should come precisely from this overlapping. Moreover, this explains the previously made distinction between the two phases of the work here reported. In the initial phase, there is a perfect match between the fictitious volumes of one grid block with the actual boundary volumes of the adjacent one. In the general case, such nice coincidence does not occur, and one must use some form of interpolation in order to obtain the fictitious control volume properties from those of the actual control volumes in the adjacent block that have some overlap with the fictitious volume considered.

The procedure for information transfering in the case of a perfect match between adjacent control volumes (initial phase) is based solely on the direction of the flow at the interface. If the contravariant velocity component at the interface is positive, that is, the flow is leaving the grid block being currently solved at that interface, property values at the interface are set equal to their values at the center of the interior volume. Therefore, in this case, the solution in the current grid block is independent of the solution at the adjacent grid block. However, since the numerical method being used needs equations for the calculation of the fictitious volume, we simply set the properties of the fictitious volume equal to those of the interior volume. The opposite situation occurs when the contravariant velocity component at the interface is negative. In this case, everything works as if the interface was an entrance boundary in which all flow variables are prescribed. The difference between this boundary and, for example, a freestream boundary is that the properties do not receive constant values, but their values are determined from the corresponding current values of the properties in the control volume of the adjacent grid block. Hence, u, v, w, T, ρ and p of the fictitious volume corresponding to the grid block being solved are set equal to their respective values in the control volume of the adjacent block. Moreover, properties at the interface itself are also calculated using the values of the control volume at the adjacent block and, of course, the metric terms of the interface.

When the control volumes on both sides of the interface do not have the same dimensions, the additional difficulty that appears is associated with determining which real control volumes of the adjacent block surround the centroid of the fictitious volume of the grid block currently being solved. Once this has been done, the procedure is similar to what we have described for the case with coincident volumes. The difference, now, is that properties at the centroid of the fictitious volume must be determined by an appropriate interpolation of the values associated with the centroids of the control volumes of the adjacent block which have some overlap with the fictitious volume considered. If the contravariant velocity component at the interface is positive, interface properties are set equal to their corresponding values at the interior volume. If the contravariant velocity component at the interface is negative, one must first perform the interpolation previously discussed, and then determine the interface values following a similar procedure as discussed for the case of coincident volumes.

It is important to observe that, in the three-dimensional case and considering the general case discussed above, typically there are eight control volumes involved in this interpolation process. Hence, the computational time involved in finding the control volumes which surround the centroid of the fictitious volume and performing the trilinear interpolation described can be significant. Therefore, an approximate procedure for the transfering of information was also implemented in which the property values at the centroid of the fictitious volume are taken to be equal to their corresponding values at the actual volume of the adjacent block whose centroid is the closest to the centroid of the fictitious volume. This is clearly introducing an approximation which, however, becomes less serious as the mesh is refined. Tests performed comparing the correct treatment of the interface and the approximate one here described have indicated that, in most cases, the results are extremely similar. Obviously, the approximate treatment has a lower computational cost.

The algorithm for the numerical solution of the problem with the multiblock implementation is actually quite similar to the one that has been previously described. The difference is that the advance of the iteration scheme in time is done by performing a few number of iterations in each block, transfering this block information to its neighbors, and then moving on the next block until the complete computational domain is advanced the same number of iterations. Then, we return to the first block and repeat the whole procedure until a convergence criterion is satisfied for all blocks. We have found that something of the order of 3 to 5 iterations in each block, before transfering the information to the adjacent blocks, seems to be a good compromise. It is worth mentioning that this multiblock approach is extremely useful not only

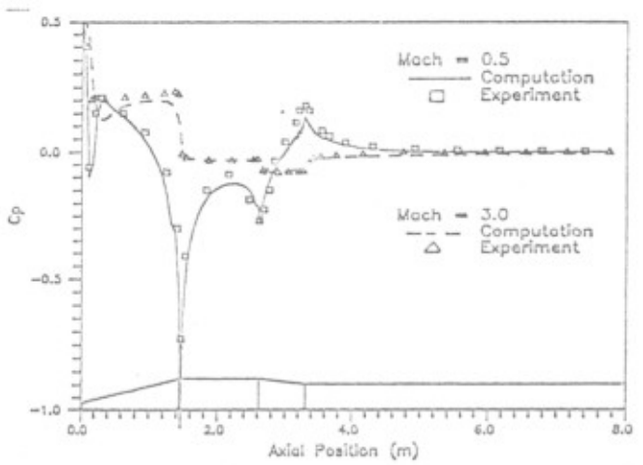

Figure 5: Inviscid finite difference calculations for subsonic and supersonic freestream conditions at zero angle of attack.

for the case of very complex configurations, but also for the case in which the available computational resources do not have enough central memory to accommodate the complete grid (or all the grid blocks) in core. Hence, one can have only the grid block being worked on in central memory. Therefore, from a computational efficiency point of view, it can be extremely interesting to perform several iterations on one grid block before having to write this block's information on disk and reading in the next next block's information. Of course, this can only be done if this procedure does not seriously deteriorate the overall convergence rate of the algorithm. Finally, the authors refer the interested reader to the work of Marchi et al.[7] for further details of the present implementation of the multiblock technique.

Some Validation Results

The calculations performed for the VLS vehicle so far have only considered the central body. Therefore, none of the computational results to be presented here will include the effect of the boosters. When comparing these computations with the experimental data, we have taken precautions to avoid comparisons in regions in which the properties at the central body are affect by the presence of the boosters. Hence, all the computations are mostly concerned with the forebody portion of the vehicle.

Fig. 5 presents pressure coefficient distributions for the VLS obtained with the centered finite difference algorithm previously discussed. These results are presented here mainly for comparison purposes, since our major interest in this work is to discuss results concerning the validation of the segregated finite volume algorithm with the multiblock implementation, using our own experimental data. As indicated in Fig. 5, the freestream conditions considered in these computations were $M_\infty = 0.5$ and 3.0. The two cases assumed a zero angle of attack. The computational meshes used in these calculations were generated algebraically for one axisymmetric longitudinal

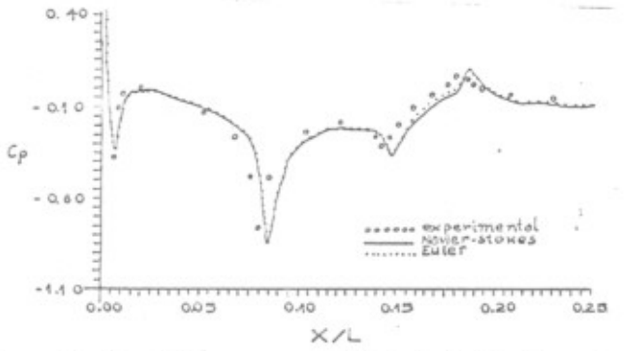

Figure 6: Leeside pressure coefficient distributions for $M_\infty = 0.5$ and $\alpha = 6°$.

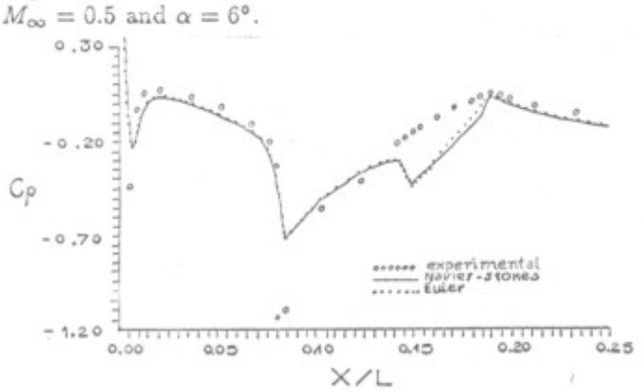

Figure 7: Leeside pressure coefficient distributions for $M_\infty = 0.9$ and $\alpha = 6°$.

plane, and then rotated 360° around the body. Therefore, the finite difference calculations are using periodic boundary conditions in the circumferential direction. This is in contrast with the finite volume calculations, to be presented next, which use symmetry boundary conditions in the circumferential direction. The computational meshes used for these simulations had $63 \times 34 \times 26$ points in the longitudinal, normal and circumferential directions, respectively. It is clear from Fig. 5 that both subsonic and supersonic cases are presenting good agreement with the experimental data. In the subsonic case, we also observe an over-expansion of the flow in the forward cone-cylinder intersection. Our own experience with axisymmetric calculations has indicated that this has a tendency of occuring with Euler simulations with the present method, but that the over-expansion typically disappears when viscous terms are included in the formulation.

Pressure distribution comparisons for the segregated finite volume algorithm were performed in the Mach number range $0.5 \leq M_\infty \leq 3.0$, and considering angles of attack of 0°, 2° and 6°. Some representative results of the calculations performed with this code are presented in Figs. 6, 7 and 8. The leeside pressures are being shown in these figures. Similar comparison is obtained for other azimuthal planes. The computational meshes used for these calculations had $48 \times 70 \times 12$ control volumes in the longitudinal, normal and circumferential directions, respectively. We emphasize that the computational domain in the longitudinal direction extends further downstream in the finite difference calculations which justifies

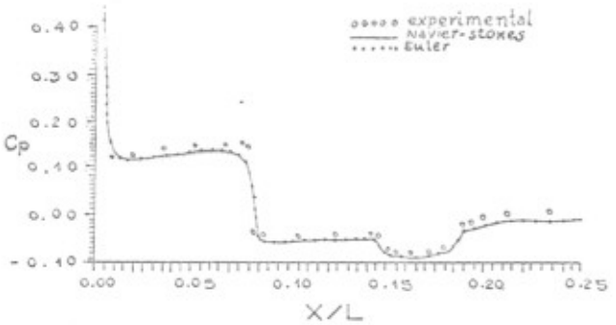

Figure 8: Leeside pressure coefficient distributions for $M_\infty = 3.0$ and $\alpha = 6°$.

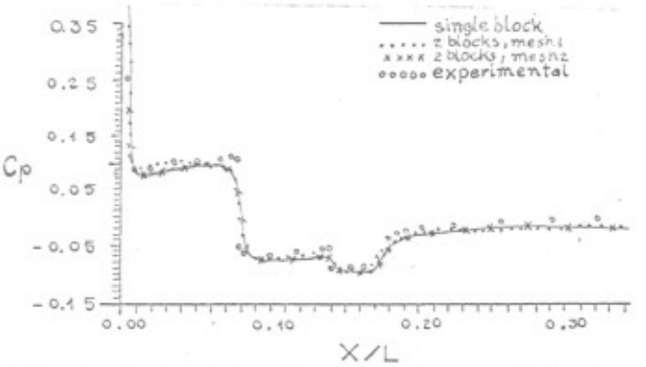

Figure 9: Comparison of leeside pressure coefficient distributions for $M_\infty = 3.0$ and $\alpha = 6°$ with multiblock implementation.

the difference in the number of longitudinal grid points as compared with the calculations presented in Figs. 6–8. On the other hand, the computations presented in these figures show results for viscous simulations which accounts for the marked increase in the number of grid points in the wall normal direction. One can observe that there is good agreement between computational and experimental results for both subsonic ($M_\infty = 0.5$) and supersonic ($M_\infty = 3.0$) flow cases. Moreover, it is evident from Figs. 6 and 8 that the relevant flow features are correctly captured by the simulation.

The agreement in the transonic case is much poorer. Of special concern is the fact that around $x/L = 0.15$, which is the location of the forebody cylinder-boattail intersection, both viscous and inviscid simulations predict a local expansion of the flow whereas the experimental results show no indication of such behavior. Video tapes of the wind tunnel tests indicate that for $M_\infty = 0.9$ there is a rather strong shock wave impinging upon the vehicle's forebody cylindrical section. This is clearly indicated by the experimental pressure coefficient distribution shown in Fig. 7. Therefore, our current explanation for the difference in pressure distribution behavior around $x/L = 0.15$ is that the present implementation of the code is unable to capture the flow separation that occurs at the impingement point of this transonic shock. The reason for such behavior is still being investigated. In a parallel development, the implementation of an eddy viscosity type turbulence model in the present code is currently under way. It is expected that this will be able to improve the correlation of the computational viscous results with the experimental data.

Another important conclusion that one can draw from the results presented in Figs. 6–8 is that the addition of the viscous terms does not seem to significantly improve the correlation of the present computational results with the experimental data. A possible explanation for such behavior is that these simulations are performed for Reynolds numbers of the order 10^7. Therefore, it is clear that the boundary layer should become turbulent in the very nose region of the body. Since we are performing laminar simulations, this could be a reason for the behavior observed. Moreover, as previously discussed, it is expected that the implementation of a turbulence model should improve the present simulation capability.

Inviscid results using the multiblock technique previously described are presented in Fig. 9 for a 3.0 freestream Mach number case with 6° angle of attack. Two different computational meshes were used for these simulations, besides the single block grid. Both of these meshes had two blocks, where the first block is always close to the body surface and the the other one is the outer block. Mesh 1 had $60 \times 10 \times 18$ volumes in the first block and $60 \times 14 \times 18$ volumes in the second block. Mesh 2 had $60 \times 14 \times 18$ in the first block and $60 \times 10 \times 18$ in the second block. The single block grid is simply formed by the addition of these two grid blocks in a single mesh. The leeside pressure distributions shown in Fig. 9 indicate that the multiblock calculations do agree well with the single block solution. The differences observed in mesh 1 in the forebody cone region can be attributed to the fact that, in this case, a block interface was located in a region of very high gradients. As would be expected, this seems to be causing a degradation in the accuracy of the information transfering from one block to the other.

Concluding Remarks

The details of an experimental wind tunnel investigation for pressure distribution and force measurements for the Brazilian VLS are described. The development of an all speed Euler and/or Navier-Stokes flow simulation code which uses a segregated finite volume algorithm for 3-D body conforming curvilinear coordinates with a colocated variable arrangement is also described. The efforts towards the physical validation of this code are the major contribution of the present work. The results obtained for subsonic and supersonic flow conditions are in very good agreement with the experimental data. Transonic calculations, however, show a much poorer agreement with the available data for equivalent and even more refined meshes. Recent calculations by Marchi et al.[19] and by Azevedo et al.[20] for transonic nozzle applications seem to indicate that the flow solution methodology used in the present code does need a severe refinement in the longitudinal direction in order to truly capture the rapid flow gradients present in transonic flow conditions, at least as compared to conventional central difference type

algorithms[20]. The validation results obtained so far also indicate that further work must be done with regard to the capture of Reynolds number effects in viscous computations. Nevertheless, at the present stage of development, the code already gives very useful information which can be used for preliminary design studies, and which is actually being presently used in the VLS design process. Moreover, efforts are also under way in order to use the present multiblock implementation to provide the first three-dimensional computational results of the VLS vehicle together with its 1st stage boosters.

References

1. Maliska, C.R., and Silva, A.F.C., "A Boundary-Fitted Finite Volume Method for the Solution of Compressible and/or Incompressible Fluid Flows Using both Velocity and Density Corrections," in *Finite Elements in Fluids*, T.J. Chung and G. Karr, Eds., Huntsville Press, 1989.
2. Marchi, C.H., Maliska, C.R., and Bortoli, A.L., "The Use of Co-Located Variables in the Solution of Supersonic Flows," *Proceedings of the 10th Brazilian Congress of Mechanical Engineering*, Vol. 1, Rio de Janeiro, Brazil, Dec. 1989, pp. 157-160.
3. Marchi, C.H., Maliska, C.R., and Silva, A.F.C., "A Boundary-Fitted Numerical Method for the Solution of Three Dimensional All Speed Flows Using Co-Located Variables," *Proceedings of the 3rd Brazilian Thermal Sciences Meeting*, Vol. I, Itapema, SC, Brazil, Dec. 1990, pp. 351-356.
4. Azevedo, J.L.F., "Aerodynamic Flow Simulation Using a Finite Difference Method," *Proceedings of the 2nd National Meeting of the Thermal Sciences*, Águas de Lindóia, SP, Brazil, Dec. 1988, pp. 3-6.
5. Azevedo, J.L.F., Zdravistch, F., and Silva, A.F.C., "Implementation and Validation of Euler Solvers for Launch Vehicle Flows," *Proceedings of the 4th International Symposium on Computational Fluid Dynamics*, Vol. I, Davis, CA, Sept. 1991, pp. 42-47.
6. Moraes, P., Jr., and Neto, A.A., "Aerodynamic Experimental Investigation of the Brazilian Satellite Launch Vehicle (VLS)," *Proceedings of the 3rd Brazilian Thermal Sciences Meeting*, Vol. I, Itapema, SC, Brazil, Dec. 1990, pp. 211-215.
7. Marchi, C.H., Maliska, C.R., and Silva, A.F.C., "Numerical Solution of Flows Over Complex Geometries Using the Multiblock Technique," *Proceedings of the 4th Brazilian Thermal Science Meeting*, Rio de Janeiro, Brazil, Dec. 1992, pp. 353-356 (in Portuguese, original title is "Solução Numérica de Escoamentos em Geometrias Complexas Utilizando a Técnica de Multiblocos").
8. Moraes, P., Jr., and Neto, A.A., "High Speed Aerodynamic Wind Tunnel Tests for the VLS," Report, Instituto de Aeronáutica e Espaço, São José dos Campos, SP, Brazil, Mar. 1989 (in Portuguese, original title is "Ensaios Aerodinâmicos em Túnel de Vento de Alta Velocidade para o VLS").
9. Anderson, D.A., Tannehill, J.C., and Pletcher, R.H., *Computational Fluid Mechanics and Heat Transfer*, McGraw-Hill, New York, 1984.
10. Pulliam, T.H., and Steger, J.L., "Implicit Finite-Difference Simulations of Three-Dimensional Compressible Flow," *AIAA Journal*, Vol. 18, No. 2, Feb. 1980, pp. 159-167.
11. Pulliam, T.H., and Steger, J.L., "Recent Improvements in Efficiency, Accuracy and Convergence for Implicit Approximate Factorization Algorithms," AIAA Paper 85-0360, AIAA 23rd Aerospace Sciences Meeting, Reno, Nevada, Jan. 1985.
12. Beam, R.M., and Warming, R.F., "An Implicit Finite-Difference Algorithm for Hyperbolic Systems in Conservation-Law Form," *Journal of Computational Physics*, Vol. 22, 1976, pp. 87-110.
13. Beam, R.M., and Warming, R.F., "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations," *AIAA Journal*, Vol. 16, No. 4, April 1978, pp. 393-402.
14. Patankar, S.V., *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Co., New York, 1980.
15. Karki, K.C., and Patankar, S.V., "Pressure-Based Calculation Procedure for Viscous Flows at All Speeds in Arbitrary Configurations," *AIAA Journal*, Vol. 27, No. 9, Sept. 1989, pp. 1167-1174.
16. Maliska, C.R., Silva, A.F.C., and Marchi, C.H., "Co-Located Variable, Three-Dimensional, Euler and Navier-Stokes Formulation for All Speed Flows," Report prepared for Instituto de Aeronáutica e Espaço, São José dos Campos, SP, Mar. 1991 (in Portuguese, original title is "Formulação Euler e Navier-Stokes Tri-Dimensional em Variáveis Co-Localizadas para Escoamentos a Qualquer Velocidade").
17. Van Doormaal, J.P., "Numerical Methods for the Solution of Incompressible and Compressible Fluid Flows," Ph.D. Thesis, University of Waterloo, Ontario, Canada, 1985.
18. van Doormaal, J.P., and Raithby, G.D., "Enhancements of the Simple Method for Predicting Incompressible Fluid Flows," *Numerical Heat Transfer*, Vol. 7, 1984, pp. 147-163.
19. Marchi, C.H., Silva, A.F.C., and Maliska, C.R., "Numerical Solution of Inviscid Flows in Nozzles with Supersonic Exit Speed," *4th Brazilian Thermal Sciences Meeting*, Rio de Janeiro, Brazil, Dec. 1992, pp. 145-148 (in Portuguese, original title is "Solução Numérica de Escoamentos Invíscidos em Tubeiras com Velocidade Supersônica na Saída").
20. Azevedo, J.L.F., Fico, N.G.C.R., Jr., Ortega, M.A., and Luna, G.C., "Nozzle Flow Computations Using the Euler Equations," ICAS Paper No. 92-4.1.2, *Proceedings of the 18th Congress of the International Council of the Aeronautical Sciences*, Beijing, P.R. of China, Sept. 1992, pp. 97-107.