ON THE DEVELOPMENT OF A 3-D MULTIGRID UNSTRUCTURED CODE FOR COMPLEX AEROSPACE APPLICATIONS

Enda Dimitri V. Bigarella
Instituto Tecnológico de Aeronáutica, CTA/ITA, São José dos Campos, BRAZIL
enda.bigarella@ig.com.br

Edson Basso and João Luiz F. Azevedo
Instituto de Aeronáutica e Espaço, CTA/IAE, São José dos Campos, BRAZIL
basso@iae.cta.br, azevedo@iae.cta.br

Abstract. *The paper discusses the development and the results obtained using a finite volume code for 3-D unstructured meshes to simulate high-Reynolds-number turbulent compressible flows. This work is inserted in the context of the development of a numerical tool for simulation of typical aerospace flows of interest of the Brazilian aerospace community. An explicit Runge-Kutta scheme is used to perform time marching. Spatial discretization is performed with second-order centered or upwind schemes. The implementation uses a cell-centered, face-based data structure. A full agglomeration multigrid scheme is available to accelerate convergence to steady state. Turbulence models are available in order to include the turbulent effects into the numerical formulation. The paper presents simulation results for typical aerospace applications, such as high-Reynolds-number transonic and supersonic turbulent flows. In general, good agreement with theoretical or experimental results is obtained with the present numerical tool.*

Keywords: *CFD, aerospace applications, unstructured grids*

1. Introduction

The CFD group at Instituto de Aeronáutica e Espaço (IAE) already achieved good results using finite volume methods on 3-D unstructured meshes to simulate turbulent viscous flows over typical aerospace configurations. The computational code solves the Reynolds-averaged Navier-Stokes (RANS) equations. Advanced eddy-viscosity turbulence models are available in order to include the turbulence effects into the RANS equations. Viscous simulations at high Reynolds numbers are typical for aerospace applications, such as the ones of interest to IAE. Numerical simulations of such flight condition which do not consider turbulence effects do not have any physical meaning and, therefore, have limited practical application. The chosen turbulence closures are suitable for external aerodynamics applications and they can predict flow separation with acceptable levels of accuracy.

A fully explicit, 2nd-order accurate, 5-stage, Runge-Kutta time stepping scheme is used to perform the time march of the flow equations. For flux calculations on the volume faces, Jameson's centered scheme (Jameson et al., 1981) plus explicitly added artificial dissipation terms, or Roe's flux-difference splitting method (Roe, 1981) can be used. Boundary conditions are set through the use of ghost cells attached to the boundary faces. The implementation uses a cell-centered, face-based data structure and the code can use meshes with any combination of tetrahedra, hexahedra, triangular-base prisms and pyramids. A full multigrid (FMG) scheme is also available in order to achieve better convergence rates for the simulations. To build the mesh sequence, an agglomeration scheme based on cell or node seeds is used. The CFD group of IAE has experience with such technique (Strauss and Azevedo, 2002) in other 2-D numerical codes. A robust and consistent method for 3-D turbulent flow simulations has been derived and included into the present numerical formulation. This methodology allowed for successful simulations of high-Reynolds number turbulent flows at very acceptable costs. Extensive validation of this method had already been initiated and one is referred to Scalabrin, 2002, and Bigarella et al., 2004, for a careful analysis of the successful initial validation results.

The simulation results obtained for the VLS configuration as well as other typical aerospace test cases using the present code are discussed in this paper. Turbulent transonic and supersonic flows are simulated. The numerical results obtained show good agreement with the experimental data and they represent all the relevant aerodynamic features observed in experimental tests.

2. Theoretical Formulation

The flows of interest in the present context are modeled by the 3-D compressible Reynolds-averaged Navier-Stokes (RANS) equations. These equations can be written in dimensionless form, assuming a perfect gas, as

$$\frac{\partial Q}{\partial t} + \frac{\partial E_e}{\partial x} + \frac{\partial F_e}{\partial y} + \frac{\partial G_e}{\partial z} = \frac{\partial E_v}{\partial x} + \frac{\partial F_v}{\partial y} + \frac{\partial G_v}{\partial z} , \quad Q = \left[\begin{array}{ccccc} \rho & \rho u & \rho v & \rho w & e \end{array} \right]^{\mathrm{T}} . \tag{1}$$

where Q is the dimensionless vector of conserved variables. Here, ρ is the fluid density, u, v and w are the Cartesian velocity components and e is the fluid total energy per unit of volume.

In this work, all properties are made dimensionless according to a set of dimensional reference variables provided by the user. The necessary dimensional reference variables are composed of a reference length, D^*_{ref}, a reference speed, V^*_{ref}, a reference dynamic viscosity coefficient, $\mu_{\ell}{}^*_{ref}$, a reference temperature, T^*_{ref} and a reference density, ρ^*_{ref}. The $*$ superscript denotes dimensional properties. The user must also provide the gas properties, namely the constant of the gas, R^*; the specific heat at constant volume, Cv^*; the specific heat at constant pressure, Cp^*; the Prandtl number, Pr; and the turbulent Prandtl number, Pr_t. The gas R^*, Cp^* and Cv^* properties are also made dimensionless with the provided reference variables. Furthermore, the Reynolds number is defined as $Re = \rho^*_{ref}\, V^*_{ref}\, D^*_{ref}/\mu_{\ell}{}^*_{ref}$.

In Eq. (1), the E_e, F_e and G_e terms are the dimensionless inviscid flux vectors and E_v, F_v and G_v are the dimensionless viscous flux vectors, which can be written as $A_e = \left[\begin{array}{ccccc} \rho C & \rho uC + \delta_{kx} p & \rho vC + \delta_{ky} p & \rho wC + \delta_{kz} p & (e+p)\, C \end{array}\right]^{\mathrm{T}}$, and $A_v = (1/Re) \left[\begin{array}{ccccc} 0 & \tau_{kx} & \tau_{ky} & \tau_{kz} & \beta_k \end{array}\right]^{\mathrm{T}}$. In these expressions, $A = E$, F or G, $k = x$, y or z, and $C = u$, v or w, respectively; δ_{ij} is the Kronecker delta, and the other relations can be given as

$$\tau_{ij} = (\mu_\ell + \mu_t)\left[\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{2}{3}\frac{\partial u_m}{\partial x_m}\delta_{ij}\right] , \qquad \beta_i = \tau_{ij} u_j - q_i \, , \qquad q_j = -\gamma\left(\frac{\mu_\ell}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial e_i}{\partial x_j} \, , \tag{2}$$

where $u_i = (u, v, w)$ are the Cartesian velocity components, $x_i = (x, y, z)$ are the Cartesian coordinates and μ_t is the eddy viscosity coefficient, computed by the chosen turbulence model. Furthermore, e_i is the internal energy and γ is the ratio of specific heats. The dimensionless pressure, p, can be calculated from the perfect gas equation of state as $p = (\gamma - 1)\left[e - \frac{1}{2}\rho\left(u^2 + v^2 + w^2\right)\right]$.

The present work is mainly interested in high-Reynolds number simulations of flows over complex aerodynamic configurations. Two turbulence closures have been chosen in the present context. The Spalart and Allmaras (SA) model (Spalart and Allmaras, 1992) is assembled using, basically, empiricism. It solves a transport equation for a modified turbulent viscosity, simplified for an entirely turbulent flow, *i.e.*, neglecting transitional terms. The shear-stress-transport (SST) model, as proposed by Menter, 1993, is derived from both the famous $k - \omega$ (Wilcox, 1993) and the standard $k - \epsilon$ models. It solves some reported problems of the $k - \omega$ closure regarding freestream value dependency (Menter, 1994) while keeping the better numerical behavior of this model at the wall when compared to $k - \epsilon$ closures. Both SA and SST closures are particularly suited for aerodynamic flow simulations and separation prediction. Furthermore, they are also less restrictive in relation to the grid refinement near the wall than other two-equation models such as the $k - \epsilon$ family of models.

3. Numerical Formulation

The finite volume method is used to obtain the solution of the RANS equations. The formulation of the method is obtained by an integration of the flow equations in a finite volume. The application of Gauss' theorem for each finite volume yields

$$\int_{V_i} \frac{\partial Q}{\partial t}\, dV + \int_{S_i} \left[\vec{P}_e - \vec{P}_v\right] \cdot d\vec{S} = 0 \, , \qquad \vec{P} = E\,\hat{\imath}_x + F\,\hat{\imath}_y + G\,\hat{\imath}_z \, , \tag{3}$$

where $d\vec{S}$ is the outward oriented normal area vector for the i-th control volume and $\hat{\imath}_x$, $\hat{\imath}_y$ and $\hat{\imath}_z$ are the Cartesian unit vectors. The discrete value of the vector of conserved variables for the i-th control volume is defined as the mean value of the conserved variables in the volume, as shown in Eq. (4). Hence, the final form of the finite volume formulation for the RANS equations can be written for an elementary volume and assuming a stationary mesh as

$$Q_i = \frac{1}{V_i}\int_{V_i} Q\, dV_i \, , \qquad \frac{\partial Q_i}{\partial t} = -\frac{1}{V_i}\sum_{k=1}^{nf}\left[\vec{P}_{e_k} - \vec{P}_{v_k}\right] \cdot \vec{S}_k \, , \tag{4}$$

where nf is the number of faces which form the control volume and $\vec{S}_k$ is the outward oriented normal area vector of the k-th face. The code developed is able to simulate flows on grids comprised of tetrahedra, hexahedra, triangular-base prisms, pyramids or a mix of these types of elements. The previous equation also indicates that the integral was discretized assuming the fluxes to be constant on the faces.

3.1 Time Integration

The integration in time of Eq. (4) can be written using a 5-stage Runge-Kutta type scheme as proposed by Jameson et al., 1981

$$Q_i^{(0)} = Q_i^n \, , \qquad Q_i^{(\ell)} = Q_i^{(0)} - \alpha_\ell \frac{\Delta t_i}{V_i} RHS_i^{(l-1)} \, , \qquad Q_i^{n+1} = Q_i^{(5)} \, , \tag{5}$$

where $\ell = 1 \cdots 5$, and the residue, RHS, is defined as $RHS_i = CO_i - VI_i - DI_i$, where CO_i, VI_i and DI_i are, respectively, the convective operator, the viscous operator and the artificial dissipation operator calculated for the i-th control volume. These operators are calculated according to the spatial discretization scheme and they are detailed in the forthcoming sections. The α_ℓ coefficients are $1/4$, $1/6$, $3/8$, $1/2$ and 1 for $\ell = 1, \cdots, 5$, respectively. The viscous operator is calculated only on the first stage of the Runge-Kutta scheme in order to save computational resources.

The time step for each volume, Δt_i, is calculated assuming a constant CFL number throughout the computational domain. Hence, $\Delta t_i = \text{CFL}\ \ell_i / \left(|\vec{q_i}| + a_i\right)$, where a_i is the speed of sound, $|\vec{q_i}|$ is the magnitude of the flow velocity and ℓ_i is the characteristic length, in the i-th cell. The characteristic length is set as the smallest distance between the control volume centroid and the centroids of each face that forms the cell.

3.2 Spatial Discretization

3.2.1 Centered Scheme

The centered scheme used in this work for spatial discretization of the convective fluxes was proposed by Jameson et al., 1981. For this scheme, the convective operator is calculated as the sum of the inviscid fluxes on the faces of the i-th volume, i.e.,

$$CO_i = \sum_{k=1}^{nf} \vec{P}_e\left(Q_k\right) \cdot \vec{S}_k\,, \qquad Q_k = \frac{1}{2}\left(Q_i + Q_m\right). \tag{6}$$

In this expression, Q_i and Q_m are the conserved properties in the volumes at each side of the k-th face and m indicates the neighbor of the i-th control volume.

The artificial dissipation operator is built by the undivided Laplacian and bi-harmonic operators. In regions of high property gradients, the bi-harmonic operator is turned off in order to avoid oscillations. In smooth regions, the undivided Laplacian operator is turned off in order to maintain 2nd order accuracy. A numerical pressure sensor is responsible for this switching between the operators. The expression for the artificial dissipation operator is given by

$$DI_i = \sum_{m=1}^{nb} \left\{ \left(\frac{A_m + A_i}{2}\right) \left[\, \epsilon_2\left(Q_m - Q_i\right) - \epsilon_4\left(\nabla^2 Q_m - \nabla^2 Q_i\right) \,\right] \right\}, \tag{7}$$

where m represents the neighbors of the i-th element and nb is the total number of neighbors of the i-th control volume. The other terms in this equation can be written as

$$\nabla^2 Q_i = \sum_{m=1}^{nb} \left[Q_m - Q_i\right]\ ,\quad \epsilon_2 = K_2 \max\left(\nu_i, \nu_m\right)\ ,\quad \epsilon_4 = \max\left(0, K_4 - \epsilon_2\right)\ ,\quad \nu_i = \frac{\sum_{m=1}^{nb} |p_m - p_i|}{\sum_{m=1}^{nb} \left[p_m + p_i\right]}\,. \tag{8}$$

In this work, K_2 and K_4 are assumed equal to $1/2$ and $3/256$, respectively. The A_i coefficient calculation is described by Mavriplis, 1990. The expression for the coefficients is $A_i = \sum_{k=1}^{nf}\left[\left|\vec{q}_k \cdot \vec{S}_k\right| + a_k \left|\vec{S}_k\right|\right]$.

In the multistage Runge-Kutta time integration previously described, the artificial dissipation operator is updated only in the first, third and fifth stages for viscous simulations. For inviscid calculations, the artificial dissipation operator is calculated in the first and in the second stages. This approach guarantees the accuracy for the numerical solution while reducing computational costs per iteration (Jameson et al., 1981).

3.2.2 Upwind Scheme

The upwind discretization in the present context is performed by the Roe flux-difference splitting method (Roe, 1981). For this scheme, the numerical flux in the k-th face can be written as

$$\vec{P}_{e_k} = \frac{1}{2}\left(\vec{P}_{e_L} + \vec{P}_{e_R}\right) - \frac{1}{2}\left|\widetilde{A}_k\right|\left(Q_R - Q_L\right)\ , \tag{9}$$

where $\left|\widetilde{A}_k\right|$ is the Roe matrix associated with the k-th face normal direction, defined as $\left|\widetilde{A}\right|\left(Q_R - Q_L\right) = |\lambda_i|\, \delta_i\, \mathbf{r}_i$. In this formulation, $|\lambda_i|$ represents the absolute values of the eigenvalues associated with the Euler equations, given as $|\mathbf{\Lambda}| = \left[\begin{array}{ccccc} |v_n| & |v_n| & |v_n| & |v_n + a| & |v_n - a| \end{array}\right]^{\mathrm{T}}$, where v_n is the normal velocity. Similarly, $\mathbf{r}_i$ represents the associated eigenvectors to the right. The δ_i term represents the element of projection of the property jump at the interface over the system eigenvectors, which is defined as $\mathbf{\Delta} = \mathbf{L}\left[\begin{array}{ccccc} \Delta\rho & \Delta(\rho u) & \Delta(\rho v) & \Delta(\rho w) & \Delta e \end{array}\right]^{\mathrm{T}}$, where the rows of $\mathbf{L}$ represents the left eigenvectors. Properties in the volume faces are computed using the Roe average procedure. More details on the Roe scheme can be found in Bigarella and Azevedo, 2005.

In the classical form in which the Roe scheme is presented (Eq. 9), the underlining argument is the numerical flux concept. Therefore, each time the numerical flux is built, the inherent numerical dissipation is also evaluated. In an explicit Runge-Kutta-type multistage scheme, this fact means that the Roe matrix defined in Eq. (9) is computed in all stages. The present authors rather understand the Roe scheme as the sum of a centered convective flux and an upwind-biased numerical dissipation contribution, that is given by the subtracted term in the right-hand side of Eq. (9). Therefore, the attractive cheaper alternate dissipation computation in the multistage scheme, as already used for the switched artificial dissipation schemes (Bigarella and Azevedo, 2005), can also be extended for the upwind flux computation.

The linear reconstruction of properties is achieved through a MUSCL scheme (Bigarella and Azevedo, 2005), in which the property at the interface is obtained through a limited extrapolation using the cell properties and their gradients. In order to perform such reconstruction at any point inside the control cell, the following expression is used for a generic element, q, of the conserved variable vector, Q: $q\left(x,y,z\right)=q_i+\nabla q\cdot\vec{r}$, where $\left(x,y,z\right)$ is a generic point in the i-th cell; q_i is the discrete value of the generic property q in the i-th cell, which is attributed to the cell centroid; ∇q is the gradient of property q; and $\vec{r}$ is the distance of the cell centroid to that generic point.

The expressions for the reconstructed properties in the k-th face can be written as $(q_L)_k = q_i + \psi_i \nabla q_i\cdot\vec{r}_{ki}$ and $(q_R)_k = q_m + \psi_m\nabla q_m\cdot\vec{r}_{km}$, where i and m represent the i-th and m-th cells, respectively; ψ_i and ψ_m represent the limiters in these cells; and $\vec{r}_{ki}$ and $\vec{r}_{km}$ are the distance vectors from these cell centroids to the k-th face centroid. The 1st-order Roe scheme can be readily obtained by setting the limiter value to zero.

The limiter computation workload is a very expensive task, amounting to more than half of an iteration computational effort in the present code. Therefore, the idea of freezing the limiter along with the dissipation operator at some stages of the multistage time-stepping scheme is very attractive in terms of large computational resource savings.

The current extension of the 1-D limiters to the multidimensional case is based on the work of Barth and Jespersen, 1989. The Barth and Jespersen limiter is a complete limiter implementation in itself, and it has some advantages as well as disadvantages. The currently proposed extension is aimed at allowing for the user the choice of any desired limiter formulation, and at solving some disadvantages of the original limiter formulation. The limiter options that are available in the present context are the *minmod*, *superbee* and *van Albada* limiters (Hirsch, 1991). One should aknowledge that the *minmod* and *superbee* limiters require the evaluation of maximum and minimum functions, which characterizes these limiters as discontinuous. The *van Albada* limiter, on the other hand, is continuous.

The difficulty in implementing a TVD method in a multidimensional unstructured scheme is related to how to defined the gradient ratio of adjacent cell, Φ. A generalization of Φ to the k-th face of the i-th cell for an unstructured grid is currently proposed (Bigarella and Azevedo, 2005). Assuming a quasi-uniform grid, where $|\vec{r}_{mi}|\approx 2|\vec{r}_{ki}|$, and remebering that in the face, $q_k=(q_m+q_i)/2$, Φ can be given by

$$\Phi=(\Phi_i)_k=\frac{\left(q_i^{\pm}-q_i\right)/|\vec{r}_{mi}|}{\left((q_i)_k-q_i\right)/|\vec{r}_{ki}|}=\frac{q_k^{\pm}-q_i}{(q_i)_k-q_i}\ , \tag{10}$$

where the k-th face is shared by the i-th and m-th cells; and the extrapolated property in the face, $(q_i)_k$, is given by $(q_i)_k = q_i+\nabla q_i\cdot\vec{r}_{ki}$. Furthermore, $q_k^{\pm}$ is the maximum and minimum properties obtained at the centroid of the faces that compose the i-th control volume. The $q_k^{\pm}$ variable can be mathematically defined as $q_k^{\pm}=\max/\min\left(q_i,q_{faces}\right)$, where the property in the faces, q_{faces}, is the arithmetic average of the properties in the neighboring cells. The multidimensional gradient ratio for an unstructured grid face can be finally obtained as

$$\Phi=\begin{cases} num^{+}/den & \text{, if } den>0\ , \\ num^{-}/den & \text{, if } den<0\ , \end{cases} \tag{11}$$

where $den=(q_i)_k-q_i$, $num^{+}=q_k^{+}-q_i$, $num^{-}=q_k^{-}-q_i$. Also, $\Phi=1$ for $den=0$.

The advantage of the gradient ratio definition in Eq. (11) is that it can be directly used in any other limiter definition. It can also be used to recast the original Barth and Jespersen limiter formulation, as previously discussed, with a slight modification though. As already discussed, the original Barth and Jespersen limiter uses extrapolated properties in the nodes to build the gradient ratio, while extrapolated properties in the faces are preferred in the current implementation. This difference yields in smaller limiter values, which can be interpreted as an undesired increase of diffusivity in the original limiter implementation. This issue can be avoided with the use of extrapolated face properties, as proposed in Eqs. (10).

3.2.3 Viscous Flux Computation

The viscous operator in the i-th control volume is calculated as the sum of the viscous fluxes on the faces which constitute the volume. In this case, both the conserved variable vector and the derivatives of such conserved variables, on the face, are calculated as arithmetic averages between their corresponding values in the two volumes which contain the face. Derivatives of flow variables, for each control volume, are calculated in the standard finite volume approach in which these derivatives are transformed, by the gradient theorem, into surface integrals around the control volume (Swanson and Radespiel, 1991; Azevedo et al., 1999).

3.3 Turbulence Transport Equations

Turbulence models are also solved according to the finite volume approach in the current context. The convective term is discretized using a simplified 1st-order upwind scheme and the diffusion term is discretized using a centered scheme. The time march is performed using the implicit Euler scheme. One should observe that the use of an implicit scheme for the time march, in an unstructured mesh, leads to a sparse linear system. The solution for this system of equations is obtained using the biconjugate gradient method (Press et al., 1992). More implementation details on this formulation can be found in Scalabrin, 2002.

3.4 Boundary Conditions

"Ghost" volumes are used in order to enforce the boundary conditions. The boundary conditions for external flow implemented in the three dimensional finite volume code are wall conditions for viscous and inviscid flows, farfield, symmetry and supersonic exit conditions. A detailed discussion of the boundary conditions for the RANS equations can be found in Scalabrin, 2002.

3.5 Multigrid Technique

A technique that may allow an excellent convergence acceleration for a numerical method is the multigrid procedure. The multigrid algorithm chosen for the present work is denominated full approximation storage (FAS), which is the recommended method for non-linear problems (Fletcher, 1988). This method is based on exchanging both solution and residue values between different grid levels. It also relies on a good time marching procedure to be effective. In order to improve the multigrid algorithm as well as the computational method, the simulations start at the coarsest grid level. Some iterations with the Runge-Kutta time stepping are performed at this grid and a high-order interpolation is performed to the next finer grid. Some multigrid cycles are, then, performed to improve the solution at this grid. This procedure is repeated successively until the finest grid is reached, with a good initial guess to the solution. Multigrid cycles are, then, performed on the finest mesh until convergence is reached. This technique is usually denominated as a full multigrid (FMG) method.

Exchange operators are used for the connection between two consecutive grid levels. The restriction operator exchanges information from a grid level to the next coarser mesh. The prolongation operator is applied only to conserved property transfers. The prolongation operator interpolates from a grid level to the next finer one. For the present finite volume framework, both operators can be easily constructed by volume-based averaging functions.

The coarse mesh levels used by the multigrid scheme are generated with an agglomeration technique. In this agglomeration algorithm, a seed volume is chosen in the fine mesh and, then, all the volumes that have at least one node in common with this seed volume are grouped and they form the coarse mesh volume. Another seed volume is selected and the agglomeration procedure continues grouping all the fine mesh volumes. In order to avoid large mesh growth between successive mesh levels, there is also an option to use seed nodes instead of seed volumes for agglomeration. For mesh composed of tetrahedra this proved to be an effective approach to avoid numerical problems in the simulations.

Better coarse mesh quality can be obtained if the selection of the seed volumes is not random. Therefore, a list containing all the fine mesh volumes is generated prior to the agglomeration procedure. For this work, the list may be formed such that the first volumes are the volumes next to a solid surface, other boundaries and, after them, the interior volumes. Although it does not necessarily provide the best agglomeration of the interior volumes, it results in good quality coarse mesh volumes close to the boundaries. Further details on the multigrid scheme can be found in Bigarella et al., 2004.

4. Results and Discussions

4.1 Flat Plate Turbulent Flow

Zero-pressure gradient flat plate low speed flows are considered for Reynolds number $Re = 1$ million and freestream Mach number $M_\infty = 0.3$. This is a very important test case since a theoretical solution is provided for the turbulent boundary layer that builds up over the flat plate surface, known as the log-law solution. Simulations with both the SA and the SST turbulence models are included in order to provide some comparison of the numerical results for both closures. The mesh about the flat plate is clustered near the flat plate surface in order to guarantee the condition of $y+ \approx 1$ near the wall, which is very important for a turbulent flow solution. The mesh is also clustered near the flat plate leading edge in order to account for the larger velocity gradients that are expected in this region.

Figure 1 shows the turbulent numerical boundary layers obtained for both turbulence models compared to the theoretical solution. One can clearly observe in Fig. 1 a strikingly coherence with the theoretical curve in the case of the SA turbulence model. In the case of the SST two-equation model, the coherence between both results is not as good in the inertial part of the boundary layer. A very similar behavior of the SST closure in this test case has been observed in a

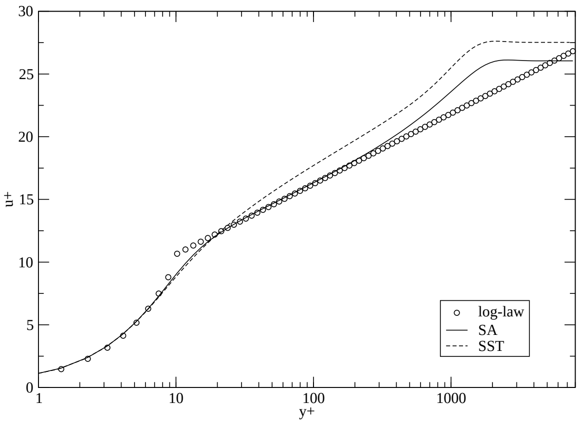
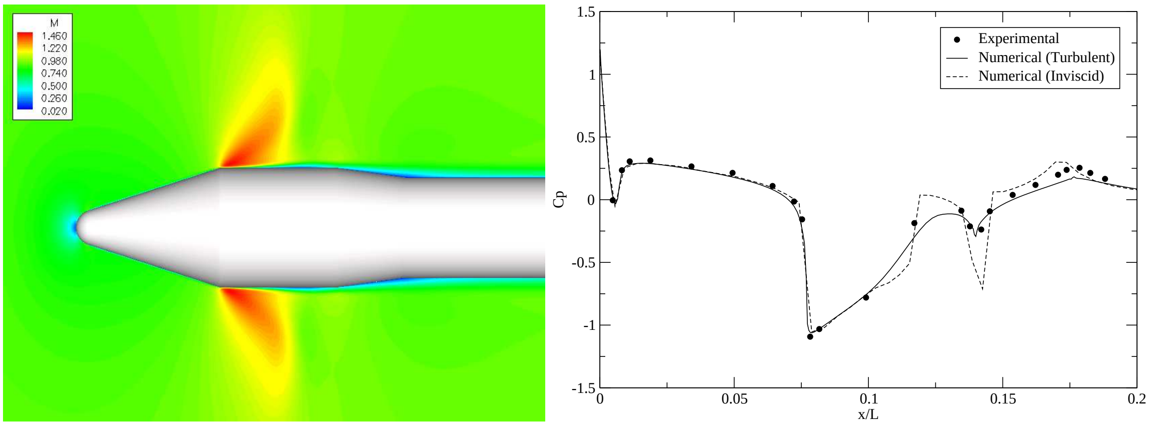


Figure 1. Comparative results between numerical and theoretical boundary layer profiles.



(a) Mach number contours.

(b) Cp distributions.

Figure 2. Results for the VLS at $M_\infty = 0.9$, $Re = 25$ million and zero angle of attack.

finite-difference context (Bigarella, 2002). Due to better numerical results and lower computational cost, the SA model has been chosen for the forthcoming simulations.

4.2 VLS Turbulent Transonic Flows

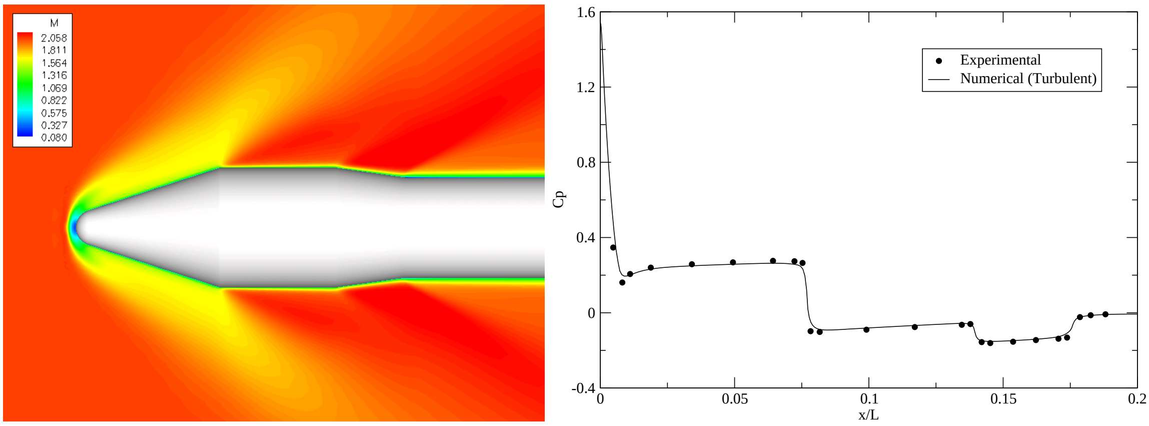
Turbulent viscous flows over the VLS at $M_\infty = 0.9$, $Re = 25$ million and zero angle of attack are considered. The mesh used in this case has $100{,}815$ nodes and $89{,}280$ hexahedra. It is clustered near solid walls to guarantee $y+ \approx 2$ for the interior volumes attached to the solid surface. Figure 2 presents Mach number contours over the vehicle forebody. There is a supersonic expansion over the end of the conical forebody. This causes the formation of a supersonic region on the payload cylinder which is ended by a shock wave. Due to the boundary layer, this shock wave does not reach the body. The region over the end of the boattail presents very small velocities, but the boundary layer does not separate because of the turbulent characteristics of the flow. In fact, laminar simulations of this flow condition indicate boundary layer separation.

A comparison between pressure coefficient distributions over the VLS surface for inviscid and turbulent numerical solutions, as well as experimental data, is also shown in Fig. 2. The simulation with turbulent effects does a better job if compared with the Euler simulation. A more consistent solution is obtained, which captures the shock wave over the payload fairing and which correctly predicts the position of the compression at the end of the boattail.

4.3 VLS Turbulent Supersonic Flows

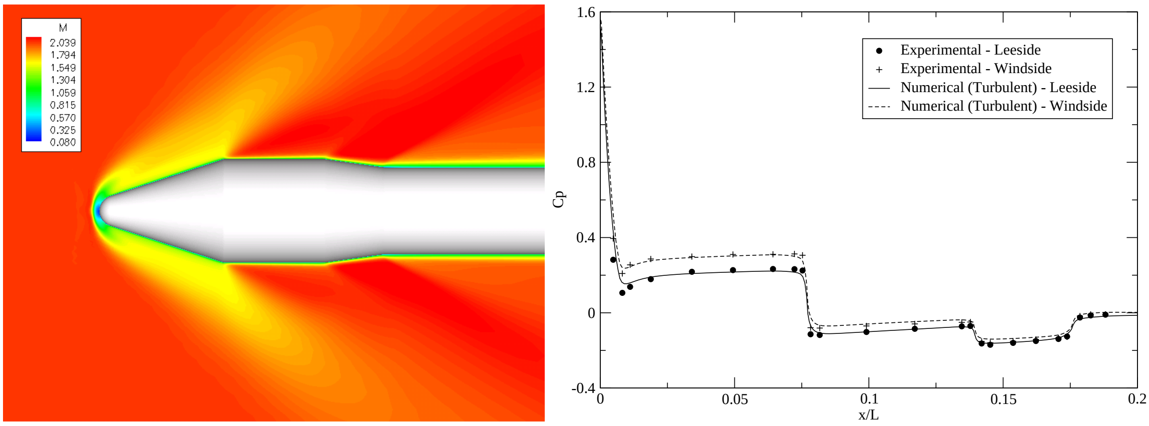
A turbulent flow at $M_\infty = 2.0$, $Re = 30$ million and zero angle of attack over the VLS second stage flight configuration is simulated. The mesh used in this case has $201{,}565$ nodes and $188{,}480$ hexahedra. It is also clustered near the solid surface to guarantee $y+ \approx 2$ for a correct resolution of the turbulent effects in the boundary layer. Figure 3 presents Mach number contours over the vehicle forebody. A comparison between pressure coefficient distributions over the VLS surface for the turbulent numerical solution and the experimental data is also shown in Fig. 3. There are no relevant differences between the solutions for this case, as expected.

The flow over the VLS second stage flight configuration at angle of attack, $M_\infty = 2.0$, $\alpha = 2.0$ deg. and $Re = 30$



(a) Mach number contours. (b) Cp distributions.

Figure 3. Results for the VLS at $M_\infty = 2.0$, $Re = 30$ million and zero angle of attack.



(a) Mach number contours. (b) Cp distributions.

Figure 4. Results for the VLS at $M_\infty = 2.0$, $Re = 30$ million and $\alpha = 2.0$ deg.

million, is also considered. The same computational mesh for the zero angle-of-attack case is used in this computation. Figure 4 presents Mach number contours over the VLS forebody in the pitch plane. A comparison between numerical and experimental pressure coefficient distributions over the vehicle forebody in the pitch plane is also presented in Fig. 4. As one can observe in this figure, the numerical solution once again presents good agreement with experimental data.

5. Concluding Remarks

The paper presents results obtained with a finite volume code developed to solve the RANS equations over aerospace configurations. The code uses a Runge-Kutta type scheme to perform the time march. The fluxes on the volume faces are computed by either a centered scheme plus explicitly added artificial dissipation to control nonlinear instabilities, or a 2nd-order Roe flux-difference-splitting upwind scheme. The code is designed to use unstructured meshes composed by any combination of tetrahedra, hexahedra, prisms and pyramids. The agglomeration multigrid scheme provides a large convergence acceleration of the numerical simulations. In a general manner, numerical solutions of complicated flows such as transonic turbulent flows about a typical aerospace configurations can be obtained in half the previous time used by the single-grid simulation.

The Spalart and Allmaras one-equation and the SST two-equation turbulence closures have been chosen in order to include the turbulence effects into the RANS equations. The code is able to correctly solve for complex flows such as transonic or supersonic turbulent flows about typical aerospace configurations. Good approximation between experimental and numerical results are obtained for such cases. The results presented here are a good indication of the capability of simulating turbulent flows about realistic aerospace configurations that has been developed by the CFD group at IAE.

6. Acknowledgments

The authors acknowledge Conselho Nacional de Desenvolvimento Científico e Tecnológico, CNPq, which partially supported the present work under the Integrated Project Research Grant No. 501200/2003-7. The authors also acknowledge Embraer – Empresa Brasileira de Aeronáutica S.A. for its support on the newer developments of the baseline code. Support for this research was also provided by Fundação de Amparo à Pesquisa do Estado de São Paulo, FAPESP, under the Research Grant No. 2000/13768-4.

7. References

Azevedo, J. L. F., Strauss, D., and Figueira da Silva, L. F., 1999, An Order of Accuracy Analysis for Flux-Vector Splitting Schemes on Unstructured Grids, "Proceedings of the 15th Brazilian Congress of Mechanical Engineering - COBEM 99", Águas de Lindóia, SP, Brazil.

Barth, T. J. and Jespersen, D. C., 1989, The Design and Application of Upwind Schemes on Unstructured Meshes, "27th AIAA Aerospace Sciences Meeting", AIAA Paper No. 89-0366, Reno, NV.

Bigarella, E. D. V., 2002, Three-Dimensional Turbulent Flow Simulations over Aerospace Configurations, Master's thesis, Instituto Tecnológico de Aeronáutica, São José dos Campos, SP, Brazil.

Bigarella, E. D. V. and Azevedo, J. L. F., 2005, A Study of Convective Flux Computation Schemes for Aerodynamic Flows, "43rd AIAA Aerospace Sciences Meeting and Exhibit", AIAA Paper No. 2005-0633, Reno, NV.

Bigarella, E. D. V., Basso, E., and Azevedo, J. L. F., 2004, Centered and Upwind Multigrid Turbulent Flow Simulations with Applications to Launch Vehicles, "22nd AIAA Applied Aerodynamics Conference and Exhibit", AIAA Paper No. 2004-5384, Providence, RI.

Fletcher, C. A. J., 1988, "Computational Techniques for Fluid Dynamics 2. Specific Techniques for Different Flow Categories", chapter 6, pp. 203–209. Springer-Verlag.

Hirsch, C., 1991, "Numerical Computation of Internal and External Flows Volume 2: Computational Methods for Inviscid and Viscous Flows", chapter 21. Wiley, Chichester.

Jameson, A., Schmidt, W., and Turkel, E., 1981, Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes, "14th AIAA Fluid and Plasma Dynamics Conference", AIAA Paper 81-1259, Palo Alto, CA.

Mavriplis, D. J., 1990, Accurate Multigrid Solution of the Euler Equations on Unstructured and Adaptive Meshes, "AIAA Journal", Vol. 28, No. 2, pp. 213–221.

Menter, F. R., 1993, Zonal Two Equation $k - \omega$ Turbulence Models for Aerodynamic Flows, "24th AIAA Fluid Dynamics Conference", AIAA Paper No. 93-2906, Orlando, FL.

Menter, F. R., 1994, Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications, "AIAA Journal", Vol. 32, No. 8, pp. 1598–1605.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P., 1992, "Numerical Recipes in C: The Art of Scientific Computing", chapter 2, pp. 83–89. Cambridge University Press.

Roe, P. L., 1981, Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes, "Journal of Computational Physics", Vol. 43, No. 2, pp. 357–372.

Scalabrin, L. C., 2002, Numerical Simulation of Three-Dimensional Flows over Aerospace Configurations, Master's thesis, Instituto Tecnológico de Aeronáutica, São José dos Campos, SP, Brazil.

Spalart, P. R. and Allmaras, S. R., 1992, A One-Equation Turbulence Model for Aerodynamic Flows, "30th AIAA Aerospace Sciences Meeting and Exhibit", AIAA Paper No. 92-0439, Reno, NV.

Strauss, D. and Azevedo, J. L. F., 2002, Unstructured Multigrid Simulation of Turbulent Launch Vehicle Flows Including a Propulsive Jet, "Proceedings of the 20th AIAA Applied Aerodynamics Conference", AIAA Paper No. 2002-2720, St. Louis, MO.

Swanson, R. C. and Radespiel, R., 1991, Cell Centered and Cell Vertex Multigrid Schemes for the Navier-Stokes Equations, "AIAA Journal", Vol. 29, No. 5, pp. 697–703.

Wilcox, D. C., 1993, Comparison of Two-Equation Turbulence Models for Boundary Layers with Pressure Gradient, "AIAA Journal", Vol. 31, No. 8, pp. 1414–1421.