NUMERICAL SIMULATION OF HIGH SPEED COMPRESSIBLE FLOWS WITH UPWIND SCHEMES ON ADAPTED UNSTRUCTURED GRIDS

L.Formaggia and V.Selmin
AERITALIA G.A.D.
Torino, Italy

1. Abstract

The study of high speed flow is receiving attention by aerospace industries in connection with the design of high-supersonic transport aircrafts and reentry vehicles. The development of effective numerical solvers is of particular interest due to the difficulties and high costs associated with experimental work at this flow regime. High speed flow is characterised by the importance of forms of energy which are normally neglected at lower speed, namely the excitation of vibrational internal degrees of freedom, dissociation and ionization. Therefore, the solution algorithm must take into account, to some degree, of all or some of this new mechanisms of energy transfer. In addition, the flow solution normally presents strong shocks and shock interactions, the code must then be able to capture those features neatly and without spurious oscillations.

The presence of localised gradients makes the implementation of adaptive techniques quite attractive. In this work, the geometrical flexibility of triangular unstructured grids has been exploited for the development of a mesh adaptation procedure which allows to efficiently refine the grid where needed. The first part of the paper will illustrate the methodology adopted for the generation and the adaptation of the grid.

Examples of steady 2D inviscid flow solutions for non-reactive and reactive fluid in chemical equilibrium are presented. The flow equations are solved using an hybrid finite volume/ finite element approach which adopts upwind concepts formulated in the context of unstructured grids. The algorithm is capable of quasi-second order accuracy in space by employing a MUSCL type procedure. However, for the chemically reacting flow computations, only first-order solution will be here presented. Work is ongoing for the extension of the methodology to non-equilibrium flows.

2. Mesh generation

The interest in unstructured grids is growing in the CFD world, the main reason being the easy way in which complex geometries can be treated and the efficient control of mesh resolution. Localised refinements may allow to optimise the number of grid points necessary for a given accuracy in the solution, thus compensating the higher computational effort per mesh point normally associated with the use of an unstructured grid based solver. In this work we have coupled an unstructured grid generator with an adaptive procedure to provide an effective tool for high speed flow computations.

The mesh generation algorithm adopted here makes use of the 'front advancing technique' as formulated in [1]. With this procedure mesh elements and points are generated at the same time according to a specified spacing distribution. The algorithm takes its name from the way the triangulation of the domain is carried out, procedure which is extensively described in the previously mentioned reference. The main feature of the method is that it does not require to pre-define the set of points to be triangulated, as it is instead demanded by most unstructured grid generators based on Delauney triangulation[2][3], which usually connect a set of given points. It allows for high variations in shape and size of the elements, it treats naturally non-convex boundaries and it can be extended to 3D problems[4].

Control of mesh resolution

In order to efficiently control the mesh resolution we specify some characteristic dimension parameters at each point of the computational domain, namely the spacings s_1 and s_2, and the associate directions defined by the two orthonormal vectors $\underline{\alpha}^1, \underline{\alpha}^2$. By means of those parameters we can locally define a linear transformation $\mathbf{T}$ which relates a vector in the physical space R with its image in the so called normalised space R^*. The transformation is governed by the relation:

$$v = \mathbf{T} v^* \quad (v \in R, v^* \in R^*) \tag{1}$$

where $\mathbf{T}$ is expressed by

$$\mathbf{T} = s_1 \underline{\alpha}^1 \otimes \underline{\alpha}^1 + s_2 \underline{\alpha}^2 \otimes \underline{\alpha}^2 \tag{2}$$

A unitary circle in the normalised space is transformed by $\mathbf{T}$ into an ellipse whose principal axes, of length s_1 and s_2 respectively, are oriented according to $\underline{\alpha}^1, \underline{\alpha}^2$. We will consider a triangle as 'optimal' if its image in R^* is equilateral with unitary side length (see figure 1). The mesh generator will then try to form elements whose shape is as close as possible to the optimal one.

The values of the dimension parameters, and thus of **T**, can be either given by the user or determined by an automatic adaptive procedure. For an initial grid, **T** will be interpolated from the values input at some user defined points. While for an adapted grid the dimension parameters will be evaluated by analysing a previously computed solution, using the procedure described in the next section.

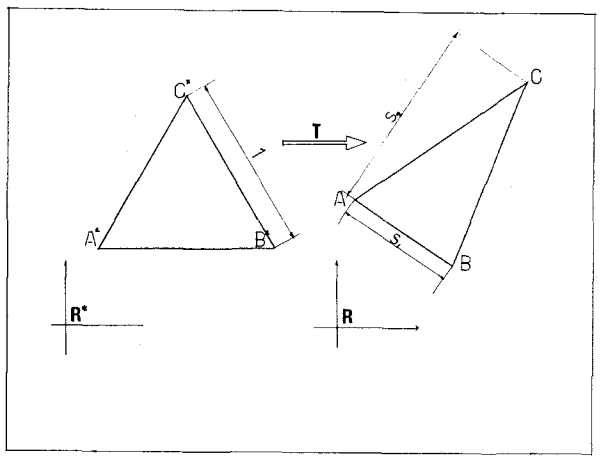


Fig. 1) Optimal element dimension parameters and element image in the normalised space R*.

3. Adaptation

Several strategies may be followed for mesh adaptation on unstructured grids [5]. They are normally based either on node movement, mesh enrichment or remeshing procedures. The methodology adopted here follows the one employed in [1] which consists in successive remeshing, each one made using a previously computed solution for the determination of the adapted element spacing distribution.

The implementation of an adaptive strategy requires first to define what is meant by the 'best' mesh for the problem at hand. A possible criterion is to take as ideal mesh the one which minimises, for a given number of nodes, some norm of the discretization error defined as the difference between the computed and the real solution. Unfortunately, at the time being a general error theory for upwind solutions of system of hyperbolic equations, able to furnish an estimate of the local discretization error seems lacking. We have here employed a strategy which adopts some concepts borrowed from interpolation theory with the addition of 'ad hoc' corrections in order to provide an effective methodology for the class of problems examined.

Error estimation

A good error estimator for an adaptive procedure should give an indication on how the error varies with the mesh size. We will here employ a methodology based on the formulas for the interpolation error which is defined as the difference between a function u(x) and its interpolant.

If we consider the approximation $u_h(x)$ of a one dimensional function u(x) on a linear finite element one-dimensional grid, we can give a local estimate of the bound of the approximation error E_k on an element $k=[x_k, x_{k+1}]$ of length $h_k=|x_{k+1}-x_k|$ as[6]:

$$\|E_k\|_2 = \|u-u_h\|_2 \le C h_k^2 \left\| \frac{d^2u}{dx^2} \right\|_2 \qquad (3)$$

where $\|\ \|_2$ denotes the L_2 norm.

This formula may be used for the construction of an adaptive procedure. After having computed a solution on a given mesh, an approximation of the second derivative may be evaluated, for example using a variational recovery procedure[7]. The imposition of an even distribution of the approximation error may allow to determine an optimal mesh spacing distribution. For example, we can obtain a distribution for the spacing s_P at each point P of the current mesh by requiring that

$$s_P^2 \left\| \frac{d^2u_h}{dx^2} \right\|_2 = c \qquad (4)$$

at each point P.

The value of the constant c may be determined, for example by imposing a value for the minimum spacing and a new mesh may then be generated whose element size varies according to the spacing distribution computed in (4).

The concepts here illustrated for a 1D problem can be extended to multidimensional situations, and in particular to the 2D case. If we consider a solution $u_h(x)$ computed on a given triangulation, we can estimate the second derivative tensor **D**

$$D_{ij}|_P = \left. \frac{\partial^2 u_h}{\partial x^i \partial x^j} \right|_P \qquad (5)$$

at each node P of the current mesh. Two local principal directions, $\underline{\alpha}_1$ $\underline{\alpha}_2$, can be evaluated from the eigenvectors of **D**, and they locally define a coordinate system X_1, X_2. The eigenvalues λ_1, λ_2 of **D** will represent the second derivatives with respect to this coordinate frame.

$$\lambda_{1P} = \left. \frac{\partial^2 u_h}{\partial X_1^2} \right|_P \qquad \lambda_{2P} = \left. \frac{\partial^2 u_h}{\partial X_2^2} \right|_P \qquad (6)$$

$$(|\lambda_1| \ge |\lambda_2|)$$

The one-dimensional analysis can be now applied on each principal direction separately. The imposition of a constant value for the interpolation error norm leads to two expressions analogous to (4). It has been found, however, that better results are obtained by using the expression for the relative error e_k, instead of the absolute error.

$$e_k = \frac{|u - u_h|}{|u|} \tag{7}$$

This has been accomplished in practice by scaling the second derivatives. The formulation for the optimal spacings s_1 and s_2 in the two principal direction at every point P of the given mesh may then be found by requiring that

$$s_{1_P}^2 \left\| \bar{\lambda}_{1_P} \right\|_2 = s_{2_P}^2 \left\| \bar{\lambda}_{2_P} \right\|_2 = c \tag{8}$$

where

$$\bar{\lambda} = \frac{\lambda}{|u|}$$

and the subscript P denotes quantities evaluated at point P. The L_2 norm is here defined as

$$|\bar{\lambda}|_2 = \left(\int_k |\bar{\lambda}|^2 d\Omega \right)^{\left(\frac{1}{2}\right)} \tag{9}$$

and the estimation of the integral over element k has been performed using a rectangular rule and assuming for the element an area proportional to $s_1 s_2$. The final expression is

$$\left| \bar{\lambda}_P \right|_2 \cong (s_{1_P} s_{2_P})^{\left(\frac{1}{2}\right)} |\bar{\lambda}_P| \tag{10}$$

The constant c in (8) is computed by imposing a value for the minimum spacing allowed.

Finally, the spacing distribution, evaluated using relations (8) and (10), is input, together with the corresponding principal directions, into the mesh generator in order to provide a completely new, adapted mesh.

Extension to systems of equations

The extension of the procedure to systems of equations has required to select one or more 'key variables' to be employed for the error analysis. The procedure has been carried out separately on each variable and the smallest value of the resulting spacings has been adopted. In the present investigation, it has been found that the use of density and absolute value of the velocity as key variables gives satisfactory results for the non-reactive flow problems. For flows in chemical equilibrium, however, the mass fractions of atomic oxygen and nitrogen have been added for a better control of the mesh.

Special treatment for discontinuities and for stagnation points

The analysis previously illustrated assumes that the variable u is smooth and in particular that it possesses finite second derivatives. This is generally not the case for the solution of inviscid flow at high speed, where discontinuities in the flow field are usually present. In these situations the second derivative is locally unbounded and we have to provide some corrections to the general methodology. We feel that the use of the L_2 norm in the error analysis already permits to reduce the effect of a peak in the second derivative compared to methodologies which make use of the RMS value as in [1]. It has been found, however, that it is better to also limit the maximum value for the estimate of the second derivative. This is accomplished in practice by considering the distribution of the derivative and cutting its value for the highest 0.5 per cent of mesh points.

Another special technique has been employed in order to ensure a good refinement at the stagnation region. This feature can be of particular importance for hypersonic flows when a detached shock is very close to the body and it may be desirable to have some extra refinement in the subsonic pocket. The Mach number has been chosen as the non-dimensional variable which allows to best localise stagnation regions. The treatment is then performed by specifying the spacing desired at the stagnation point s_{stag} as a fraction of the minimum value s_{min} in the mesh, and a limit value M_{lim} for the Mach number. The Mach M_P at each point P is then computed and the spacings are corrected according to

$$(s_{1_P})_{new} = r\, s_{stag} + (1 - r) s_{1_P} \tag{11}$$

$$\left(\frac{s_{2_P}}{s_{1_P}} \right)_{new} = r + (1 - r) \left(\frac{s_{2_P}}{s_{1_P}} \right) \tag{12}$$

where

$$r = \frac{\mathrm{Max}(0., M_{\lim} - M_P)}{M_{\lim}} \tag{13}$$

4. The Solution Algorithm

The solver employs some finite element/volume concepts in order to formulate a discretization procedure able to operate on completely unstructured grids.

Basic Euler Solver

We start from the basic Euler equations written in conservation form as

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = 0 \tag{14}$$

where $U = \left[\rho, \rho u, \rho v, \rho E \right]$ are the conservative variables and F = F(U) and G = G(U) are the inviscid fluxes. Here ρ denotes the fluid density,

u and v are the components of the velocity with respect to a cartesian coordinate system (x,y) and E represents the total specific energy. In particular, we will consider weak solutions[8] of (14), which results in employing the conservation equation written in integral form.

Spatial Approximation

The spatial discretisation is obtained by considering polygonal control volumes C_i surrounding each vertex i of the triangular mesh. These control volumes are constructed by cutting each triangle along its median lines to form six sub-elements[9].

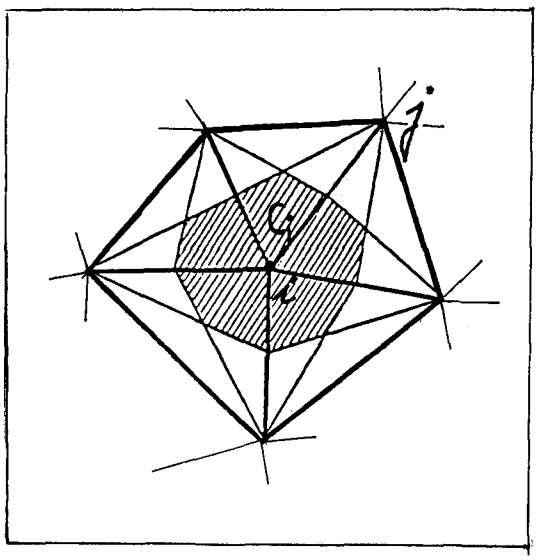


Fig.2) Construction of the control volume.

The control volume C_i is formed by the union of those sub-elements which have i as vertex, as it is sketched in figure 2. We can now associate to each side (i,j) of the triangular mesh the bi-segment B_{ij}, which forms the boundary between the control volumes C_i and C_j, as illustrated in figure 3.

The integral equations of conservation laws over each C_i may be written as

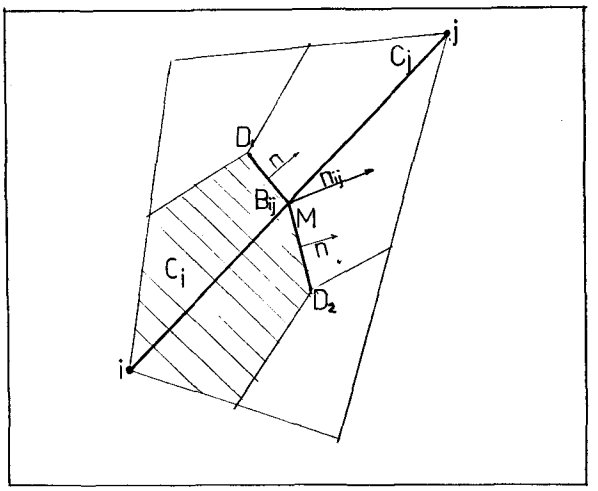


Fig.3) To each side (i,j) of the triangular mesh it corresponds a bi-segment line $B_{ij}=D_1$-M-D_2, which will be employed for the numerical discretization of the flux integral. Here M is the mid-point of side (i,j) and D_1,D_2 are the baricenters of the two adjacent triangles.

$$\int_{C_i} \frac{\partial U}{\partial t} d\Omega = -\int_{\partial C_i} \mathbf{F}\cdot\mathbf{n}\, d\Gamma \qquad (15)$$

where $\mathbf{n}=(n_x,n_y)$ denotes the outward normal to the boundary ∂C_i of the control volume and

$$\mathbf{F}\cdot\mathbf{n} = F(U)n_x + G(U)n_y \qquad (16)$$

is the projection of the flux vector on the normal direction. The flux integral is estimated by summing the contribution coming from of each bi-segment associated to the mesh sides departing from node i , while the time derivative integral is evaluated by considering a constant value for U over the control volume. The resulting expression may be written as

$$Ae_i \frac{\partial U_i}{\partial t} = -\sum_{(i,j)\in S_{ij}} \mathbf{H}(U_i,U_j,\mathbf{n}_{ij})\cdot\mathbf{n}_{ij} \qquad (17)$$

where Ae_i is the area of control volume C_i, S_{ij} is the set of sides (i,j) converging to node i and H represents the numerical flux. $\mathbf{n}_{ij}$ is the normal integrated over B_{ij}, outward oriented with respect to control volume C_i (ref. figure 3), and it is computed as

$$\mathbf{n}_{ij} = \int_{B_{ij}} \mathbf{n}\, d\Gamma \qquad (18)$$

The numerical flux **H** will in general depend on the value of the variables at the end points i and j and on the direction defined by $\mathbf{n}_{ij}$. A centered scheme may be obtained by taking **H** equal to the average value of the fluxes at the two end points.

Upwinding

The scheme adopted in this work makes use of upwinded numerical fluxes by employing a flux vector splitting procedure. The numerical fluxes **H** in equation (17) are splitted into a 'positive' and a 'negative' contribution.

$$H(U_i, U_j, \mathbf{n}_{ij}) =$$

$$H^+(U_i, \mathbf{n}_{ij}) + H^-(U_j, \mathbf{n}_{ij}) \qquad (19)$$

The formulation here used for the flux splitting adopts a modification of the technique proposed by Haenel et Al. [10] and it expresses $\mathbf{H}^+$ and $\mathbf{H}^-$ as functions of the density ρ, the fluid velocity **u**, the pressure p, the sound speed c and the total specific enthalpy h, with no explicit dependence on the specific heat ratio γ. It is not restricted to any particular formulation for the equation of state of the fluid under investigation. In addition, the basic first order scheme obtained by the substitution of the upwinded fluxes (19) into (17) satisfies the condition of constant total enthalpy at steady state. This property is considered important for the evaluation of accurate stagnation temperature.

Second order extension

Spatial second order accuracy is obtained by the introduction of a linear distribution of the variables on each control volume, as in the MUSCL approach[11]. This effectively results in replacing U_i and U_j in the expression for $\mathbf{H}^+$ and $\mathbf{H}^-$ in (19) with the vectors U_{ij} and U_{ji} obtained from 'left' and 'right' linear extrapolation at the mid-point of side (i,j)[9]. The use of limiters on these distributions is necessary for stability and in order to avoid spurious oscillations in the proximity of discontinuities in the physical solution. The standard procedure is to adopt the variables (ρ,u,p) as 'sensors' for the limiting procedure. However, we have found that this choice in general does not allow to satisfy the condition of constant total enthalpy at steady state. The condition can be instead maintained by using a formulation in which the total enthalpy h is employed in replacement of the pressure.

Time integration

Equations (17) form a system of first order ordinary equations in time, which have been integrated employing a multistage algorithm of the type proposed in [13]. As we are interested only in the steady state solution, local time-stepping has been adopted by letting each mesh node to advance at the maximum allowed local time step.

5. The Chemical Model

Equilibrium flow

The Euler solver has been modified in order to take into account the effect of some chemical reactions. The assumption of thermodynamic equilibrium has been made for the computations presented in this work. The fluid studied here is air and the chemical model adopted considers five species, namely O, N, NO, O_2 and N_2, and 17 chemical reactions [12]. For equilibrium flow 3 independent equation can be extracted from the general model, namely

$$O_2 \Leftrightarrow 2O$$

$$N_2 \Leftrightarrow 2N \qquad (20)$$

$$N_2 + O_2 \Leftrightarrow 2NO$$

The resulting laws of mass action may be written as

$$\frac{Y_2^2}{Y_5} = \frac{m_2^2}{m_5}\frac{K_1(T)}{\rho}$$

$$\frac{Y_1^2}{Y_4} = \frac{m_1^2}{m_4}\frac{K_2(T)}{\rho} \qquad (21)$$

$$\frac{Y_3^2}{Y_4 Y_5} = \frac{m_3^2}{m_5 m_4} K_3(T)$$

where the Y_i's and the m_i's are the mass fractions and the molar masses of O,N,NO,O_2 and N_2 respectively, and T represent the temperature. The expression for the equilibrium constants K_i's may be found in reference [12].

For inviscid flow, species diffusion is normally neglected. As a consequence, we may write the equations of conservation for oxygen and nitrogen nuclei as

$$\frac{2Y_4}{m_4} + \frac{Y_1}{m_1} + \frac{Y_3}{m_3} = 2Y_4^\circ \qquad (22)$$

$$\frac{2Y_5}{m_5} + \frac{Y_2}{m_2} + \frac{Y_3}{m_3} = 2Y_5^\circ \qquad (23)$$

where Y_4°, Y_5° represent the values of Y_4 and Y_5 when no dissociation is present.

Finally, we have the thermodynamic relation for the internal specific energy e, which may in general be written as:

$$e = e(T, Y_1, \ldots, Y_5) \qquad (24)$$

The laws governing chemical equilibrium have been coupled with the explicit Euler solver. At each iteration of the multistage algorithm the

Euler code provides an update for ρ, u, and e.

The corresponding values for temperature and mass fractions are then computed by solving equations (21) to (24) employing a procedure based on Newton's method. The outcome is then used to update the pressure, using the equation of state for the fluid, which for a perfect gas may in general be written in the form

$$p = \rho R T \sum_{i=1}^{s} \frac{Y_i}{m_i} \qquad (23)$$

where R denotes the perfect gas universal

constant. For more general fluids p may be found by searching into appropriate tables. In this work, we have considered only perfect gas behaviour.

Interpolation from a previous solution

When an adaptation technique is utilised, it may be convenient to restart the computation on the new, adapted grid using values interpolated from the previous solution. In this way we may expect to reach the new converged result more rapidly than by restarting the computation from scratch. It has been found, however, that in the case of chemically reacting flows the choice of the set of variables to interpolate is important for the attainment of the best convergence and for the robustness of the procedure. The best result has been obtained by interpolating the density, the velocity vector and the temperature and computing the other thermodynamic variables employing the equilibrium equations.

6. Examples

Several examples will be now presented for both non-reactive and equilibrium flows.

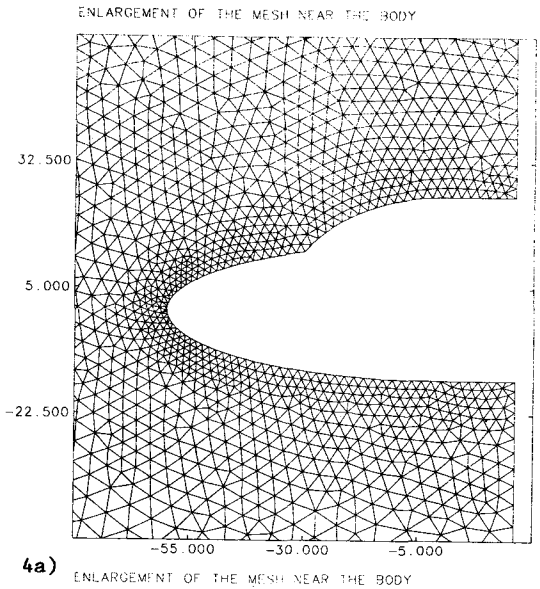
Non-reactive flow over a double ellipse at Mach 25

A first case for a double ellipse at Mach=25

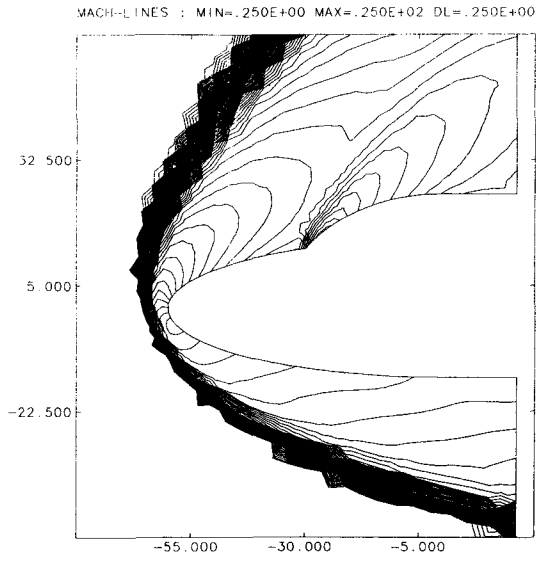
and angle of attach $\alpha = 30°$ for a non-reactive

perfect fluid with heat ratio $\gamma = 1.4$ is illustrated

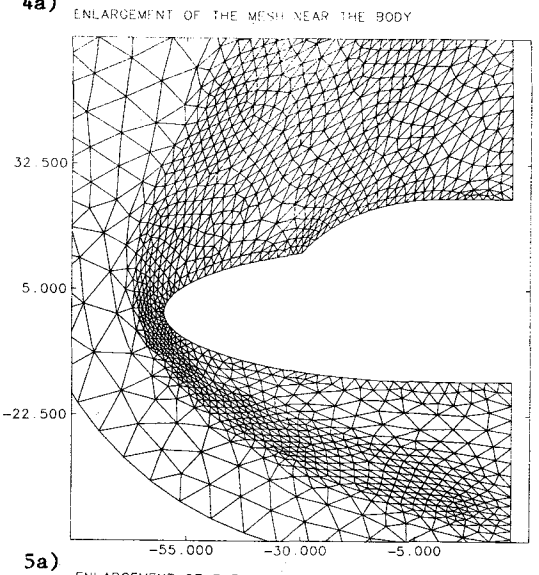
in figures 4 to 6. Figures 4a and 4b show the initial mesh with the corresponding solution obtained using the second-order Euler solver which has been previously discussed. This solution shows a strong detached shock forming in front of the body and a much weaker one in the 'canopy' region. The mesh employed contains approximately 1500 mesh nodes. However, the mesh spacing distribution is clearly far from optimal, and the shocks, particularly the weak one, are badly resolved. This solution has been employed in the adaptive remeshing procedure to obtain a new adapted mesh, illustrated in figure 5a. The determination of the position of the main shock has allowed us to bring the outer boundary closer to the body, thus saving mesh points. The second mesh has been generated maintaining the total number of nodes approximately equal to the previous mesh. It can be noted how the nodes have been concentrated in the areas with high gradients in the solution and the mesh elements have been stretched to follow the one-dimensional features. The solution obtained on the adapted mesh is illustrated in figure 5b. Shocks are better captured. Finally, a third mesh has been generated from this second solution, as it is shown in figures 6a. The number of mesh nodes in the third mesh has been slightly increased to approximately 2000. The canopy shock is now well defined. Some extra refinement has been provided in the subsonic pocket and density plus velocity modulus has been employed for the remeshing procedure. Figure 6b illustrates the final solution, computed using the mesh of figure 6a.



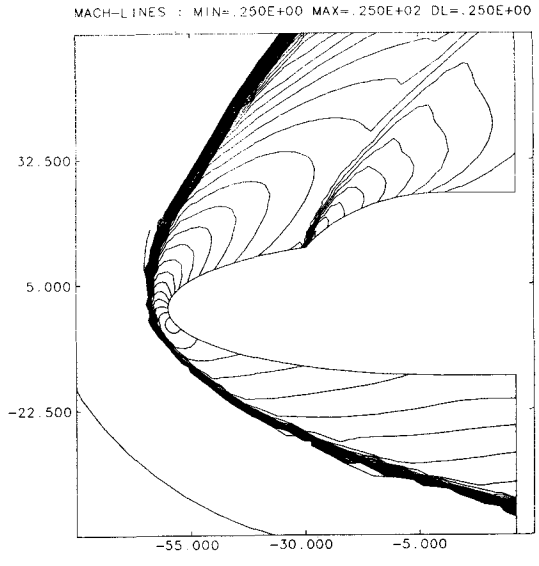
4a)



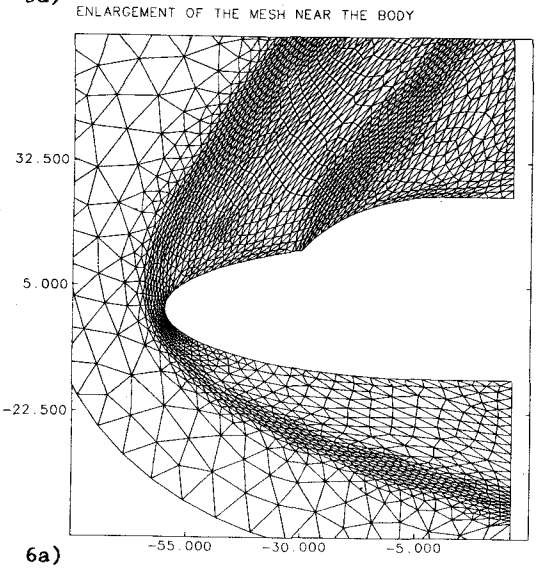
4b)



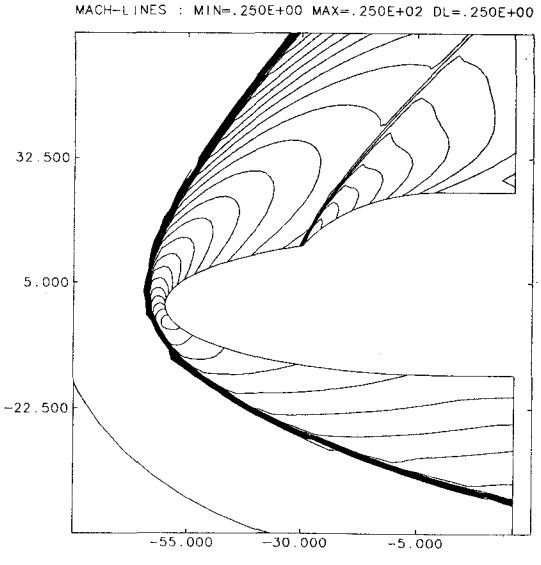
5a)



5b)



6a)

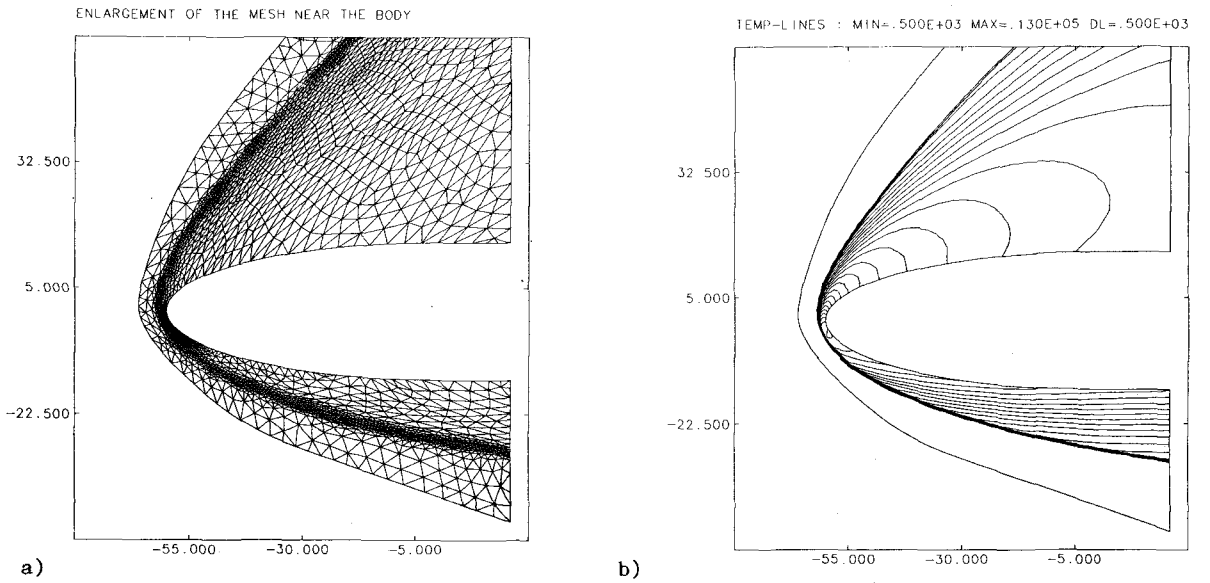


6b)

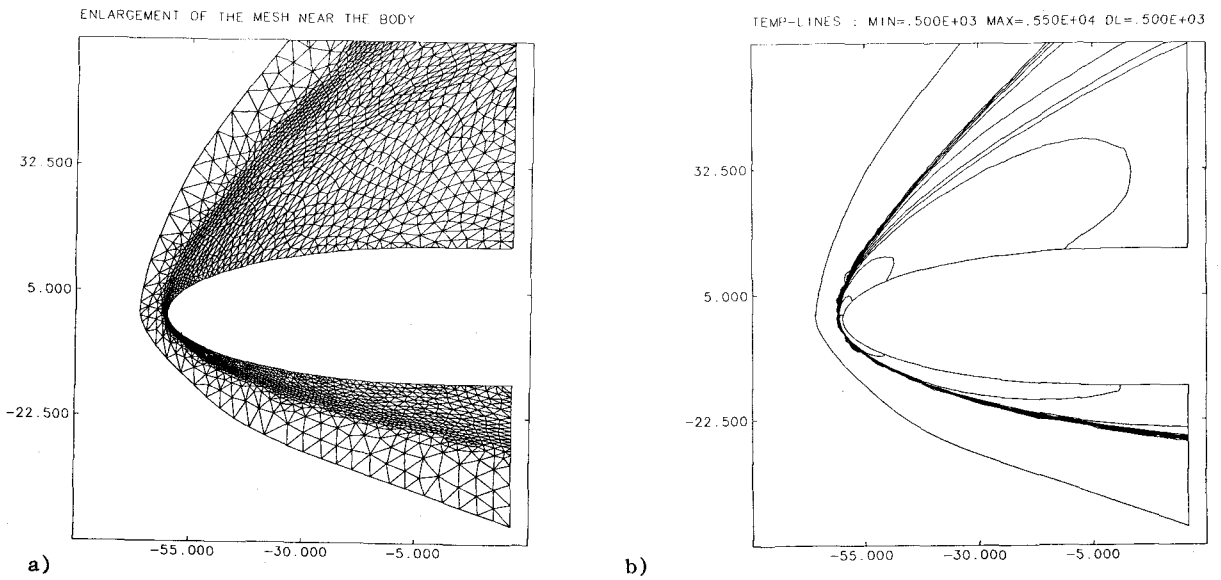
Figs 4-6) Double ellipse at Mach=25, non-reactive flow solution. Succession of meshes (a) and solutions (b) employed in the adaptation procedure.

Single ellipse at Mach 25

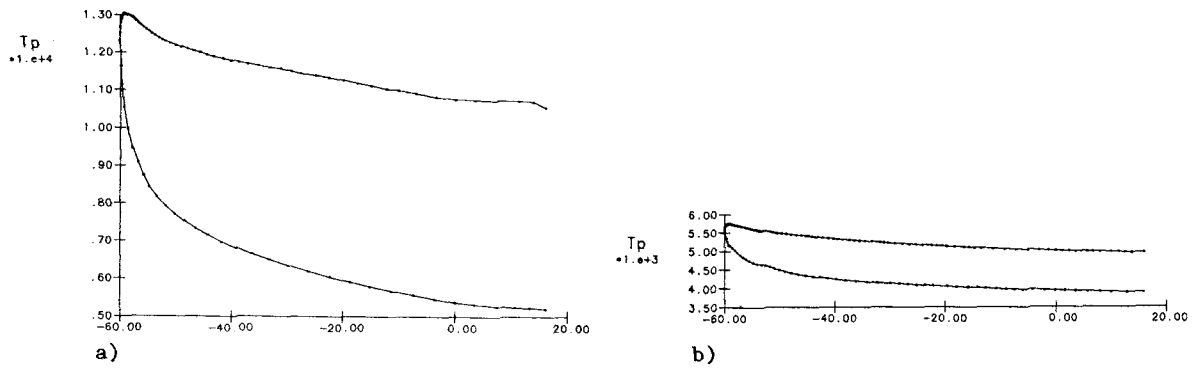
A comparison between non-reacting fluid and reacting flow at chemical equilibrium is now presented for a single ellipse configuration at Mach=25 and $\alpha = 30°$. The non-reactive flow solution has been obtained by assuming a value for the heat ratio equal to $\gamma = 1.2$. The mesh and iso-temperature contours are presented in figures 7a and 7b. The meshes and solutions here shown are in fact the last obtained from a three stage remeshing procedure analogous to the one illustrated in the previous section. The computed stagnation temperature for the non-reactive flow T_{st}=13036K agrees with the theoretical one. Figures 8a and 8b show the result obtained by the first-order Euler + chemical equilibrium solver. Again, the mesh has been adapted. Density, velocity modulus and the mass fraction Y_O and Y_N have been used as indicators in the remeshing algorithm. It can be noted that the shock position is closer to the body in the chemically reacting flow, and the maximum temperature reached, T_{sc}=5760K, is considerably lower than the stagnation temperature for the non-reactive flow case. Figure 9a and 9b show the temperature plots along the body of the ellipse, for the non-reactive flow at $\gamma = 1.2$ and chemical equilibrium respectively.



Figs. 7) Mesh (a) and iso-temperature contours (b) for a non-reactive flow solution at $\gamma = 1.2$ on a single ellipse configuration.



Figs. 8) Mesh and iso-temperature contours for chemical equilibrium flow solution.

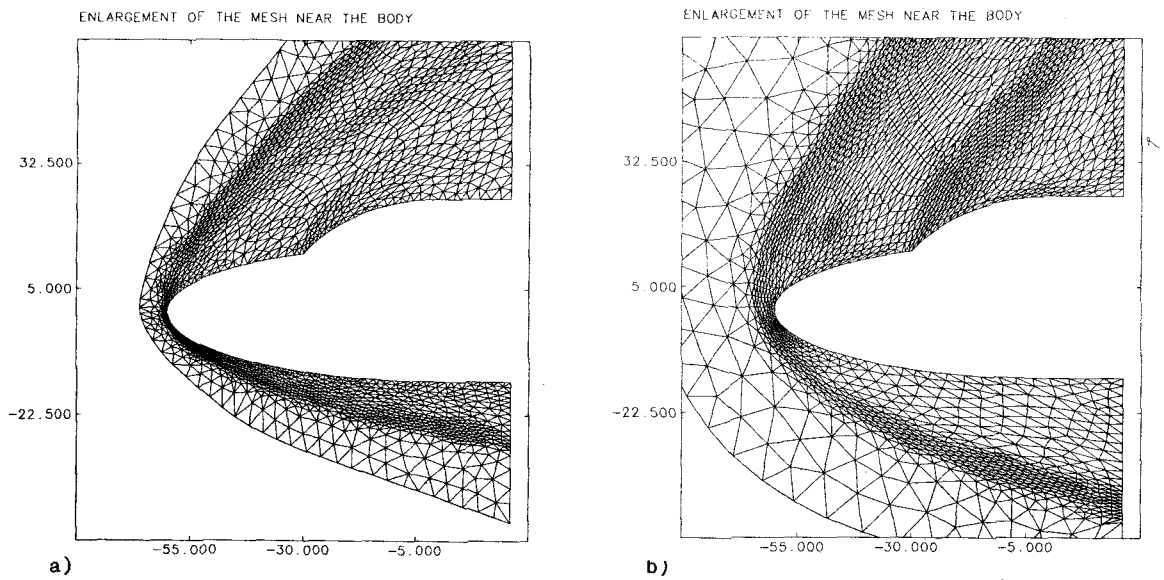


a) b)

Figs. 9) Body temperature distribution for non-reactive (a) and chemical equilibrium (b) computations on a single ellipse.

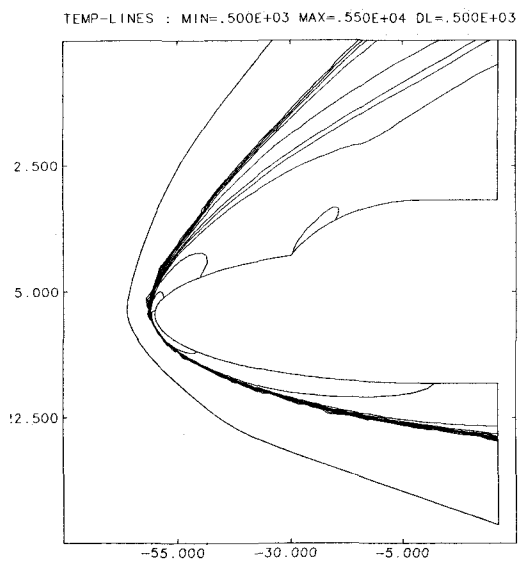
Double ellipse at chemical equilibrium

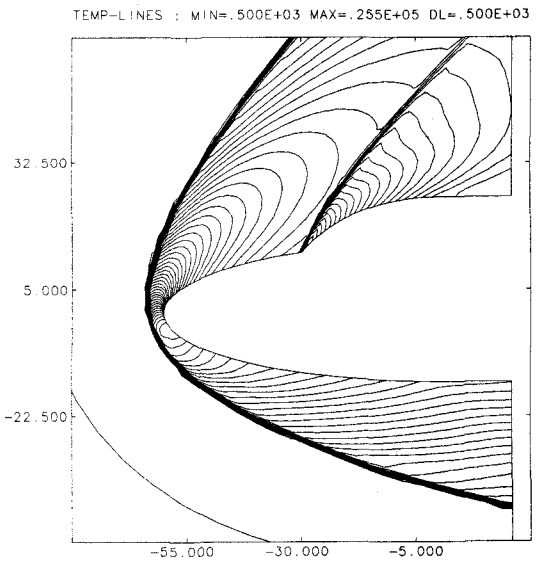
The same double ellipse configuration already shown for the non-reactive flow case has been computed with the assumption of flow in chemical equilibrium. The flow configuration is again Mach=25 and $\alpha = 30$. The adapted mesh is illustrated in figure 10a, where the grid used for the corresponding non-reactive flow computation is also shown (figure 10b) for comparison. The iso-temperature contours for the two cases are presented in figures 11a and 11b respectively. As it is expected, the shock position is now closer to the body and the computed stagnation temperature is considerably lower. The shock structure is also more complex, due to the strong chemical activity occurring in the shock region, as it is revealed by the contours for YO and YN, displayed in figures 12 and 13 respectively. The complexity of the flow features in chemical reacting flows can also be inferred by the distribution of the element size in the adapted meshes for this regime of flow, compared to the ones obtained for analogous non-reactive flow configurations. In the former, the distribution is generally more uniform, due to the larger extension of the areas where gradients in the flow solution are significant.



a) b)

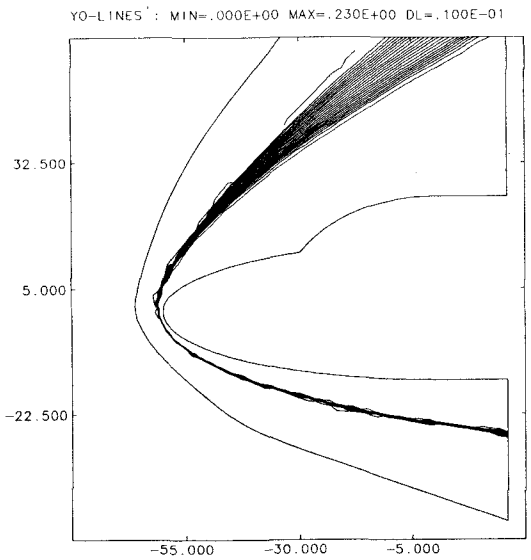
Figs. 10) Comparison of adapted meshes for the reactive (a) and the non-reactive (b) flow computation on a double ellipse.





a) b)

Figs. 11) Comparison of iso-temperature contours for reactive (a) and non-reactive (b) flow computations on a double ellipse.



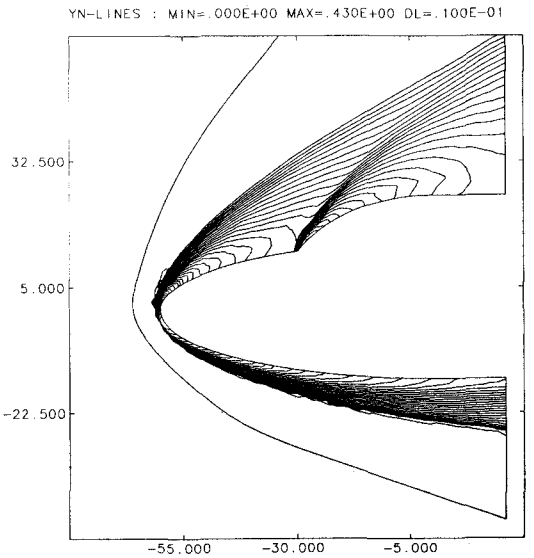


Fig.12) Atomic oxygen mass fraction Y_O contours. Fig(13) Atomic nitrogen mass fraction Y_N contours.

References

(1) Peraire J., Vahadati M., Morgan K. and Zienkiewicz O.C., Adaptive Remeshing for Compressible Flow Computations, Journal of Computational Physics, 72, 449-466, 1987.

(2) Boyer A., Computing Dirichlet tassellations, The Computer Journal, 24, 162-166, 1981.

(3) Baker T.J., in Numerical Generation in CFD, Ed. Sengupta, Hauser, Eiseman and Thompson, Pineridge Press, Swansea, 1988.

(4) Peraire J., Peiro J., Formaggia L., Morgan K. and Zienkiewicz O.C., Finite Element Euler Computations in Three Dimensions, AIAA Paper 88-0032, 1988.

(5) Morgan K. and Peraire J., Finite Element Methods for Compressible Flows, VKI Lecture Series on CFD 1987-04, March 1987.

(6) Johnson C., Numerical Solution of Partial Differential Equations by the Finite Element Method, Cambridge University Press, 1987.

(7) Lohner R., Morgan K. and Zienkiewicz, Lecture Notes in Physics, 218, Springer Verlag Ed., Berlin, 1985.

(8) Lax P.D., Weak Solutions of Nonlinear Hyperbolic Equations and Their Numerical Computation, Comm. on Pure and Applied Maths., 7, 159-193, 1954.

(9) Desideri J.A. and Dervieux A., Compressible Flow Solvers using Unstructured Grids, VKI Lecture Series 1988-05, March 1988.

(10) Haenel D., Schwane R. and Seider G., On the accuracy of Upwind Schemes for the Solution of the Navier-Stokes Equations, AIAA Paper 87-1105, 1987.

(11) Anderson W.K, Thomas J.L. and Van Leer B., A Comparison of Finite Volume Flux Vector Splitting for the Euler Equations, AIAA Paper 85-0122, 1985.

(12) Park C., On the Convergence of Chemically Reacting Flows, AIAA Paper 85-0243, 1985.

(13) Jameson A., Numerical Solution of the Euler Equation for Compressible Inviscid Fluids, MAE Report n. 1643, Princeton University, 1983.