ANNEXE A

DEVELOPPEMENT D'UN CODE EULER MULTIDOMAINE 3D

Par Ph. GUILLEN et M. DORMIEUX

DESIGN OF A 3D MULTIDOMAIN EULER CODE

Ph. Guillen
ONERA, Aerodynamics Department
29 Avenue de la Division Leclerc - 92320 CHATILLON (France)
and
M. Dormieux
AEROSPATIALE, Division Engins Tactiques
2 à 18 Rue Béranger - 92320 CHATILLON (France)

ABSTRACT

A Multidomain Euler Code has been developed to numerically simulate flows of gases of different nature around complex configurations with an emphasis on supersonic and hypersonic flows. The main choices concerning computational and numerical aspects are described. The code has been written in order to allow easy implementation of new boundary treatments or further extensions to more complex sets of equations. Some typical results concerning classical shapes, supersonic and hypersonic Hermes shuttle and transverse hot jets are shown.

INTRODUCTION

The development of industrial numerical codes to solve the equations of fluid mechanics represents an important investment with some uncertain prospective choices to be made. These last years quite an important number of attractive methods have been proposed for the simulation of perfect fluid flows around 3D configurations: finite volume or finite element methods, using structured or unstructured grids, having a centered or a non-centered numerical scheme, etc... The selection of the most promising method seems to be an hazardous and difficult choice, a matter of compromise between different considerations which are generally contradictory. Moreover some important elements of choice, such as the available computer technology are difficult to be estimated and it is clear that the evaluation of the algorithm efficiency is quite different according to the kind of processing: scalar, vector or parallel. From an industrial point of view once the desired level of accuracy is reached by the method, the most important quality of the code is robustness which can lead to select algorithms not very computationally efficient. An other important point to be considered is the extension of the code to a more complex set of equations of the same family. For instance numerical codes solving the Euler set of equations for a perfect gas will probably have to be extended to more complex state equations or to multi-species gas sooner or later. The easy implementation of new boundary treatments by means of a modular coding, as well as ergonomic considerations are also very important matters for future development and use of the code. The aim of this paper is to present a 3D Euler code developed for the numerical simulation of flows of gases of different nature keeping in mind all the points stated before. The different gases considered are perfect gas, real gas at equilibrium and non-reactive two-species mixture. In a first part, the choices leading to the architecture of the code are described. The second part deals with the numerical scheme and its implementation. Lastly the third part exhibits some first results obtained.

1. ARCHITECTURE OF THE CODE: GENERAL CONSIDERATIONS

The architecture of the code has been dictated by constraints concerning geometrical considerations, computational aspects and the specific nature of the flow.

Geometrical considerations

The treatment of complex geometries has led us to adopt a multiblock grid made of several structured, possibly overlapping or patched domains. This choice considerably simplifies the mesh construction and allows the same generality as unstructured grids. Such multiblock grid strategies are currently being developed at ONERA [1] and AEROSPATIALE [2] and will be implemented with the code. Another interesting possibility has been introduced to enable different kinds of boundary conditions on a given domain face.

Computational aspects

The constraints concerning the computational aspects are clearly linked to the available computer technology. From this point of view, vector and parallel processing have been considered. A natural idea in multidomain codes is to distribute each domain on a processor, but this could be not so interesting practically since the domains have very different numbers of points in usual situations and most of the processors will have to wait for those dealing with the greatest domains. From this synchronization waiting-time point of view a code based on the computation of separated planes seems more interesting. Another favorable computational aspect of a plane structure is that the working arrays for the numerical scheme are addressed by two indices and hence not very expensive in core for the present time computers. This can be very interesting especially in the case of implicit algorithms. Of course this way of limiting the arrays indexed by three indices restricts the choice of 3D algorithms. For instance ADI in the 3 directions is no more possible. With this choice and for meshes of a current industrial size that is to say about 200000 nodes it is possible to work without any massive I/O on a computer with a few megawords of memory core.

Flow pattern

For supersonic flows it is very important to take advantage of the hyperbolic property of the steady Euler equations (see [3] for instance) in the main direction by using space marching techniques whenever possible. Doing so the CPU time required decreases by an order of magnitude. The fact that a plane of a domain can be computed separately allows to make multiblock space-marching computations plane by plane without any additional effort. Another consequence of this supersonic nature is that for steady problems the nature of the computation can be different depending on the domain. A block with a subsonic pocket such as the blunt body part will have to be calculated in an unsteady way, the downstream part of the flow with a space-marching strategy. Possibly some real gas effects will have to be considered in some parts of the flow and not in others. The order in which the different blocks have to be computed is not obvious, and it is no more possible to iterate similarly in all the domains as it can be done for transonic computations. The computation has to be managed by a command interpreter to allow a certain flexibility.

Code organisation

It results in a code organisation built around 3 key units: a command interpreter which assumes the user interface, a plane monitoring unit which decides of the type of the computation, and a plane processor including the numerical scheme. The plane processor is described thoroughly in the second part. We now detail a little more the two first units.

Command interpreter

The command interpreter is a language, this means that the input file is interpreted dynamically. As a language it has to own classical control instructions such as DO loops or IF statements.

Its main functions are the followings.

Monitoring the core : The memory allocation is made so that the arrays indexed by the whole 3D data are reduced to the minimum, namely for unsteady applications the three coordinates plus the necessary conservative variables (in number equal to that of the equations of the system). This kind of data is stored sequentially domain after domain in the core by the mean of pointers. Monitoring the core consists in attributing pointers when a new domain is created or reorganizing it when a domain has been computed and is no more useful. The same kind of monitoring is made for the data necessary to coupling boundary conditions where 4 trilinear interpolating data have to be kept for each node involved [4].

Defining thermodynamic states : According to the aerodynamic conditions different kinds of systems of units can be appropriate. For low supersonic computations it might be interesting to define the variables relatively to critical conditions that is to say values defined for a sonic flow. For hypersonic flow the use of Mollier tables leads to SI units. From an ergonomic point of view it has also seemed interesting to give the opportunity to the user to use a large set of possible ways of defining thermodynamic states.

Defining scheme parameters and boundary conditions : For instance the Courant number or the nature of the boundary condition.

Initializing domain variables : Many options can be used. Initialization can be made by means of formated or unformatted files, with uniform states previously defined. If necessary it can be done on a part of a domain only.

Extracting data : This includes not only result files to be interpreted but also intermediate printings and creation of PHIGS metafiles containing graphical data.

Calling the plane monitor : And so doing calling the numerical scheme.

Plane monitoring

As the calculation is based on the computation of separated planes, different kinds of calculations can be made according to the way these planes are computed :

Unsteady flows : The time step is the same for every cell of every domain.

Pseudo unsteady flows : The solution is advanced in an unsteady way but without taking care of the physical meaning of the flow before convergence.

Space marching computation : The solution on a physical plane is advanced till convergence before computing the next physical plane.

Since there are basically two kinds of calculation, space marching and unsteady computations, there are also two kinds of multidomain strategies. A plane multidomain strategy where all the planes of different domains corresponding to a physical plane are calculated together, and a more classical way where the solution is advanced on a whole domain before considering another one.

2. THE NUMERICAL SCHEME

Formulation

The unsteady 3D Euler equations are written in conservation form:

$$W_t + F_x + G_y + H_z = 0$$

where, for instance, for a one species perfect gas

$$W = {}^t(\rho, \rho u, \rho v, \rho w, e)$$

$F=^t(\rho u,\ p+\rho u^2,\ \rho uv,\ \rho uw,\ (e+p)u)$
$G=^t(\rho v,\ \rho uv,\ p+\rho v^2,\ \rho vw,\ (e+p)v)$
$H=^t(\rho w,\ \rho uw,\ \rho vw,\ p+\rho w^2,\ (e+p)w)$
with $p=(\gamma-1)(e-1/2.\rho.(u^2+v^2+w^2))$

To solve them we use an implicit upwind TVD finite volume scheme of Van Leer MUSCL type. To obtain the maximum benefit from the structured organisation of the grid and the organisation by plane of the code, the implicit part consists in an ADI like inversion in each plane coupled with a Gauss-Seidel like relaxation in the third direction. Basically the scheme comprises the 3 following steps:

1) *Introduction of a linear distribution for each direction in each cell to compute the cell interfaces*

$$U_{i+\sigma,\ j+\lambda,\ k+\mu} = U_{i,j,k} + \sigma\ g^i_{ijk} + \lambda\ g^j_{ijk} + \mu\ g^k_{ijk}$$

where $U=P^{-1}(W)$ is a set of variables. To preserve stability near discontinuities it is necessary to introduce limiters in each direction:

if $\alpha_{i+1/2}= U_{i+1,j,k}-U_{i,j,k}$
$g^i_{ijk}=\text{limiter}(\alpha_{i+1/2},\ \alpha_{i-1/2})$

Many limiters have been implemented among them the "minmod" [5],Van Leer [6],Van Albada [7], and "superbee" [5] formulations. The set of variables can be chosen among the conservative, the primitive (ρ,u,v,w,p) or the characteristic one. Then the scheme fulfills a monodimensional TVD property.

2) *Computation of the explicit part*

$$\Delta W_{exp} = -\Delta t/Vol(F_{i+1/2}-F_{i-1/2}+G_{j+1/2}-G_{j-1/2}+H_{k+1/2}-H_{k-1/2})$$

where $F_{i+1/2}$ (resp. G, H) is an evaluation of the fluxes at an interface of the cell control volume by means of an approximate Riemann Solver between the two states on each side of the interface calculated in the first step.
Many approximate Riemann Solvers have been tested:
For perfect gas: The Van Leer [8], Roe [9] and Osher [10] formulations
For a mixture of non reactive two species gas: Abgrall [11] extension of Roe fluxes and Abgrall-Montagné [12] extension of Osher fluxes
For real gas with an equilibrium assumption: Vinokur-Montagné [13] extension of Van Leer and Roe scheme, and Abgrall-Montagné [12] extension of Osher scheme.

3) *Computation of the implicit part in each plane*

$$A.\Delta W=\Delta W_{exp}$$

where A can be seen as an approximation of

$$(I + \Delta t\ \partial F/\partial w)$$

where F is an approximation of $F_x+G_y+H_z$

Two approximations have been tested: the linearized conservative implicit formulation of Steger-Warming and the linearized non conservative formulation of Harten-Yee [14] or Chakravarthy [15].

For boundary conditions many treatments have been considered, from Viviand-Veuillot [16] compatibility relations to more classical flux treatments and their implicitations.

As it can be seen, Van Leer's MUSCL approach presents several advantages :
The scheme is TVD and so well suited for flows with strong discontinuities.
The extension to more complicated state equations or multispecies convective flows is straightforward by implementing the corresponding new approximate Riemann solver and new jacobian of the fluxes. The modifications are made very locally in the computer code and the overall architecture is fully kept. Moreover all the modifications can be added so that we have a unique source for all the different kinds of flows.
For most of the variants of the scheme, no parameter have to be fixed by the user.

Programming notes
To implement this numerical scheme the following steps are coded in the plane processor which advances the solution on a plane k of a domain.

1. *Research of intersecting boundary conditions.*
This step determines among the boundary conditions that have been declared those which are concerned with this plane.
2. *Treament of geometric singularities.*
In those routines boundary nodes including fictitious points might be moved in order to treat some geometrical singularities like axes or some special boundary treatments like those dealing with half control volume cell (for instance wall like conditions) or symmetry.
3. *Computation of metrics.*
From the data of either the cell centers or the cell vertex, the attributes of the control volume such as volume, diameter of the included sphere or outward normal vectors are computed.
4. *Computation of time step.*

And then repeat the following steps 5 to 11 for the 3 directions i,j,k.

5. *Treatment of boundary nodes.*
According to the boundary treatment involved, conservative values of frontier nodes (possibly fictitious nodes) might have to be assigned before the computation of the slopes. It concerns nearly all the boundary conditions since the fictitious points have at least to be extrapolated. It is crucial for matching boundary conditions where the nodes might be interpolated into another domain.
6. *Computation of slopes.*
This is the part 1) of the numerical scheme described above.
7. *Treatment of boundary interfaces.*
It concerns boundary treatments for which the values are treated on the interfaces rather than at the nodes.
8. *Computation of fluxe*
This is the part 2) of the numerical scheme.
9. *Treatment of boundary fluxes.*
Here, boundary fluxes are possibly modified. For instance for a wall treatment the flux computed by the scheme is replaced by a pressure flux.
10. *Computation of implicit coefficients.*
This is part 3 of the numerical scheme.
11. *Treatment of boundary implicit coefficients*
According to the boundary conditions the implicit matrices are modified.

12. *Explicit result.*
Obtained by summing the fluxes it constitutes the right hand side of the implicit part.
13. *Resolution of the implicit system.*
It consists of an ADI like inversion but several other options such as a Gauss-Seidel, Jacobi

algorithms can easily be implemented.
14. *Compatibility relation treatment of a wall boundary.*
It consists in a modification of the values at the frontiers according to compatibility relations.

This decomposition of the algorithm coding allows the implementation of a large variety of treatments. To be more concrete, let us take the case of the implementation of an explicit wall treatment condition and a matching condition and see some of their possible implementations.

For a first order like cell mirror condition the fictitious node can be imposed symmetrically to the boundary node and so step 5 can be used.
For a second order like cell mirror condition the outward interface value can be imposed symetrically to the inner interface value so that step 7 is used.
For a classical flux treatment, first a half control volume can be defined next to the wall and then the wall flux is explicitly computed as a pressure flux. Step 2 and 9 are used in this case.
For a compatibility relation treatment of a slip boundary (see [16]) only step 14 is used.

For a matching condition among many possibilities the two following ones have been implemented:
for an area with a smooth flow, a second order matching condition can be applied. The boundary node and its fictitious counterpart are interpolated in the coupling domain. Then only step 5 is used.
for an area with strong gradients a first order matching condition can be used. A constant distribution is introduced in the boundary cell for the outward direction and the external interface value is interpolated in the coupling domain. This can be done with steps 5 and 7

3. CODE VALIDATION

In this part some preliminary results showing some possibilities of the code are presented. The first case permits the validation of the multidomain space-marching strategy. The second case exhibits some results obtained on a more complex shape, the Hermes shuttle, in supersonic and hypersonic aerodynamic configurations. The last case illustrates a multiblock computation of a non reactive two species gas flow with an adequate refinement.

Ogive-cylinder-flare configuration

Experimental results on this configuration are available in [17]. In the selected test case the Mach number is 2.96 and the incidence 4 degrees. Three space marching computations have been made. The first one with a monodomain grid, the second one with a continuous two domain grid, the last one with a discontinuous two domain grid (figure 1). Figure 2 shows the Mach number contours on the body and in the plane of symmetry. It can be seen that results are very similar except a small wave issued from the intersection of the ogive attached shock with the coupling frontier. Figure 3 puts into evidence the very weak influence of the different grids on body pressures and shows a rather good comparison of the computed results with experimental ones. The small differences between the curves come from the intersection of the previous wave with the body.

Hermes shuttle configuration

Two supersonic computations have been made for a Mach number of 2.5 and incidences of 0 and 10 degrees. The Mach number contours on the body and in the symmetry plane are displayed on figure 4, comparison of the lift with experimental data has been found to be in very good agreement. Figure 5 shows density contours obtained on Hermes forebody for an hypersonic flow of Mach number 12 without incidence. The fluxes used in this case are the Van Leer-Vinokur-Montagné ones.

Hot transverse jet calculation

This simulation deals with the interactions occuring between a jet emerging straight up from a flat plate into a flow parallel to this one [18] (see figure 6 for the initial acrodynamic conditions). In a practical situation the hot jet has a specific heat ratio different from the outflow. This can be simulated numerically by considering non reactive mixtures of two species flows. An important experimental feature of these flows is the existence of a subsonic area that could not be observed on the initial coarse grid. So in order to capture this subsonic pocket a refined domain has been added on its supposed location (see the frame figure 7 and the computational grids figure 8). This domain has been initialized with the solution obtained in the coarse grid and then the calculation has been achieved only on the refined domain (weak coupling). Figure 9 shows the concentration distribution of the hot jet in the refined area for the two meshes. One can see the strong influence of the grid refinement on the jet shape. Figure 10 shows the Mach number distribution in the flowfield. Clearly the subsonic pocket is nicely captured on the refined grid (minimum Mach number of 0.67).

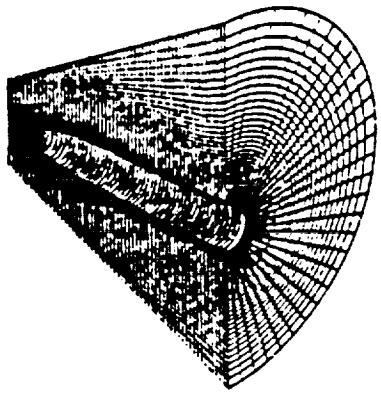
CONCLUSION

A 3D multidomain Euler code has been developed, its very modular coding allows the implementation of various numerical variants of Van Leer MUSCL scheme and a large variety of boundary conditions. The extension to different state equations or multispecies flows has been shown to be straightforward. Its ability to handle unpatched grids eases considerably the multiblock gridding and permits to make refinements wherever wanted. The code is monitored by a command interpreter which possesses the necessary flexibility to handle supersonic and hypersonic computations around complex shapes.

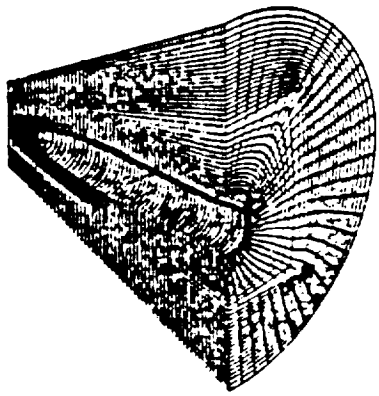
REFERENCES

1. Jacquotte, O.P. Generation, Optimization, and Adaptation of Multiblock Grids around Complex Configurations, Proceedings of AGARD 64th FDP, Loen, 1989.
2. Ranoux, G., Lordon, J., Diet, J. Géométrie et Maillage de Configurations Complexes pour les Calculs Aérodynamiques, Proceedings of AGARD 64th FDP, Loen, 1989.
3. Borrel, M., Montagné, J.L., Diet J., Guillen, Ph., Lordon, J. Upwind Scheme for Supersonic Flows around Tactical Missile, La Recherche Aérospatiale, 1988-2.
4. Vuillot, A.M. A Multi-Domain 3D Euler Solver for Flows in Turbomachines, Proceedings of the 9th ISABE Symposium, Athens, 1989. (to be published).
5. Roe, P.L. Some Contributions to the Modelling of Discontinuous flows, Lecture in Applied Mathematics, vol. 22, 1985.
6. Van Leer, B. A. Second Order Sequel to Godunov Method, Journal of Computational Physics, vol. 23 pp. 276-299, 1977.
7. Van Leer, B. II Monotony and Conservation Combined in a Second Order Scheme, Journal of Computational Physics, vol. 14 , pp. 361-370,1974.
8. Van Leer, B. Flux Vector Splitting of the Euler Equations, Icase Report 82-30, 1982.
9. Roe, P.L. Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes, Journal of Computational Physics, Vol. 43 pp. 357-372, 1982.
10. Osher, S., Solomon F., Upwind Difference Schemes for Hyperbolic Systems of Conservation Laws, Mathematics of Computation, Vol. 38158 pp. 339-374, 1982.
11. Abgrall, R. Généralisation du Schéma de Roe pour le Calcul d'Ecoulements de Mélanges de Gaz à Concentrations Variables, La Recherche Aérospatiale, 1988-6.

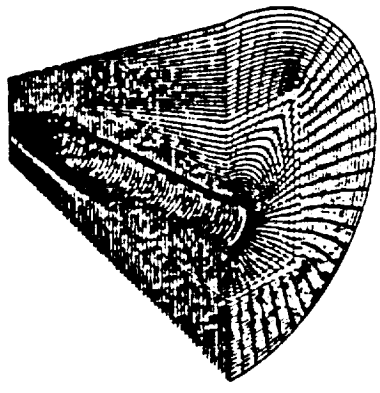
12. Abgrall, R., Montagné, J.L. Généralisation du Schéma d'Osher pour le Calcul d'Ecoulements de Mélanges de Gaz à Concentrations Variables et de Gaz Réels, To be published in La Recherche Aérospatiale, 1989.
13. Montagné, J.L., Yee, H.C., Vinokur, M. Comparative Study of High Resolution Shock Capturing Schemes for a Real Gas, Proceedings of th 7th Gamm Conference on Numerical Methods in Fluid Mechanics, 1987.
14. Yee, H.C., Harten, A. Implicit TVD Schemes for Hyperbolic Conservation Laws in Curvilinear Coordinates, AIAA 85-153.
15. Chakravarthy, S.R., High Resolution Upwind Formulations for the Navier-Stokes Equations, VKI Lecture 1988-05.
16. Viviand, H., Veuillot, J.P. Méthodes Pseudo Instationnaires pour le Calcul d'Ecoulements Transsoniques, Publication ONERA 1978-4 (English Translation ESA TT 561).
17. Landrum, E.J. Wind Tunnel Pressure Data at Mach Numbers from 1.63 to 4.63 for a Series of Bodies of Revolution at Angles of Attack from -4 to 60 Degrees, Nasa Technical Memorandum, Langley Research Center, 1977.
18. Dormieux, M., Mahé, C. Calculs Tridimensionnels de l'Interaction d'un Jet Latéral avec un Ecoulement Supersonique Externe, AGARD-CP n°437, Lisbonne.(1988).



a) Monodomain grid.

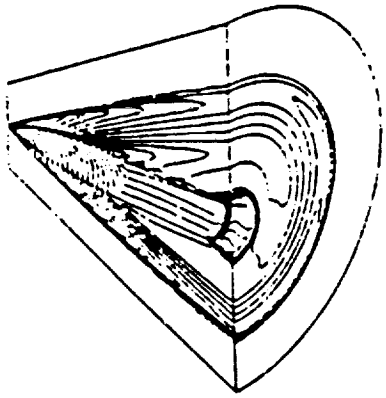


b) Continuous two-domain grid.

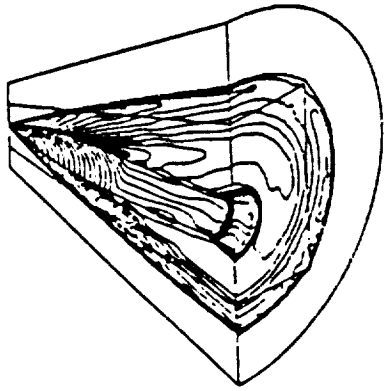


c) Discontinuous two-domain grid.

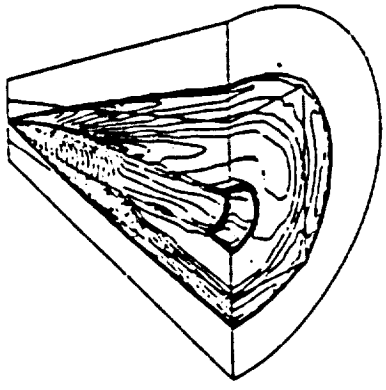
Figure 1. Computation grids of the ogive-cylinder-flare configuration.



a) Monodomain computation.



b) Two-domain computation using a continuous grid.



c) Two-domain computation using a discontinuous grid.

Figure 2. Mach number contours on the body and in the plane of symmetry for the ogive-cylinder-flare configuration.

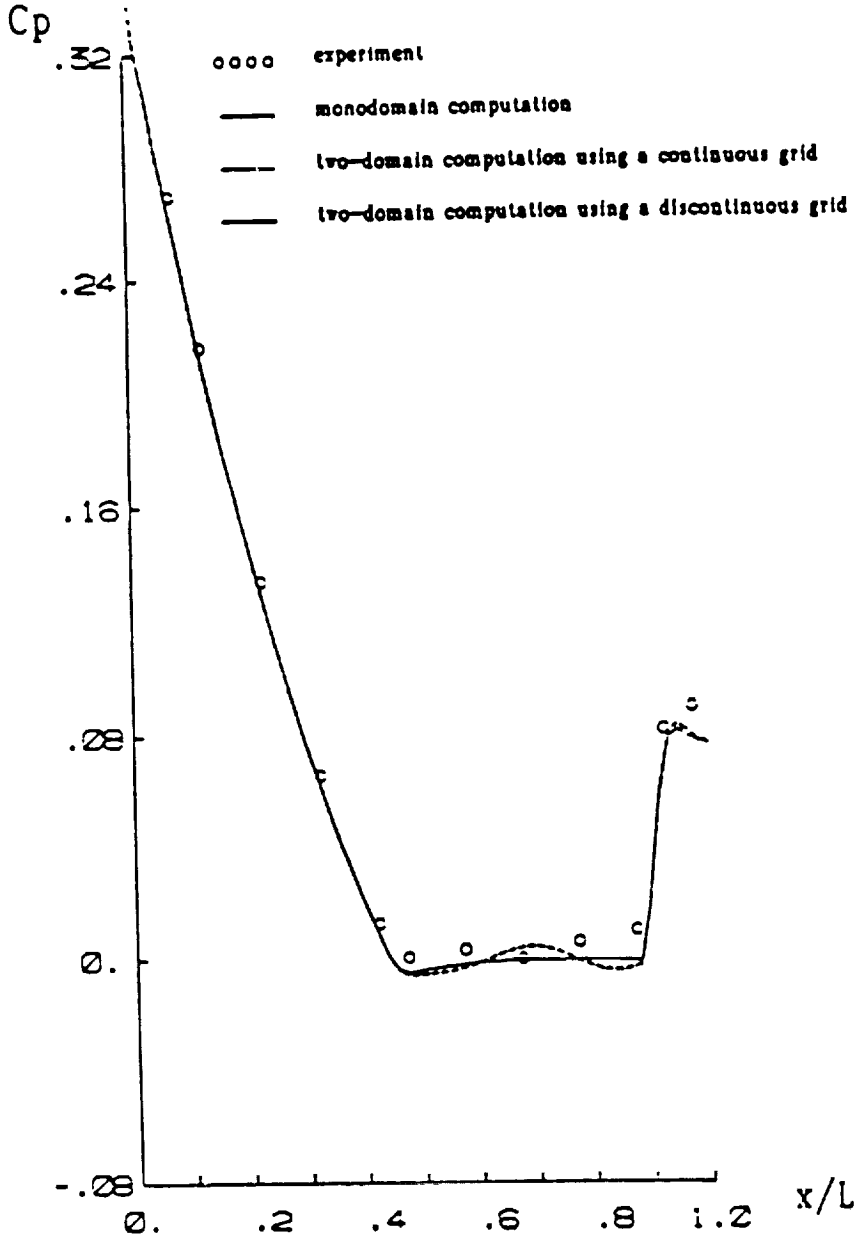
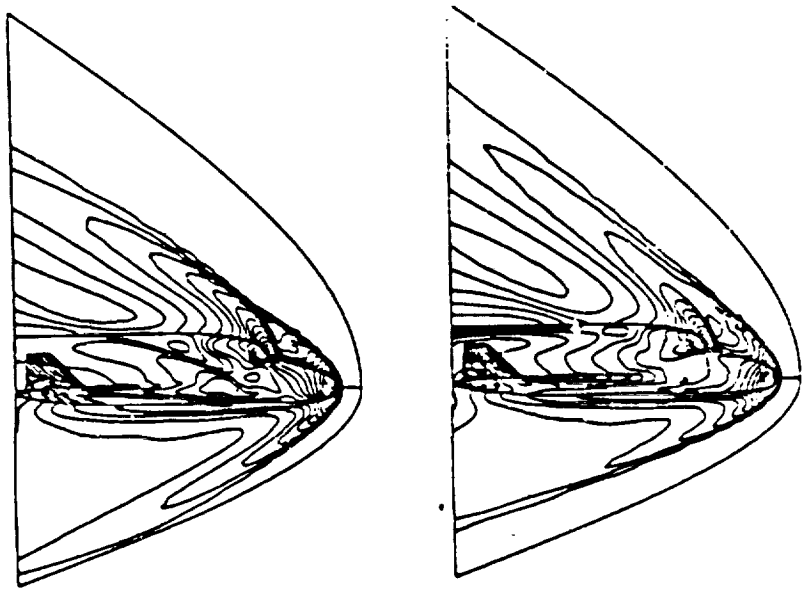


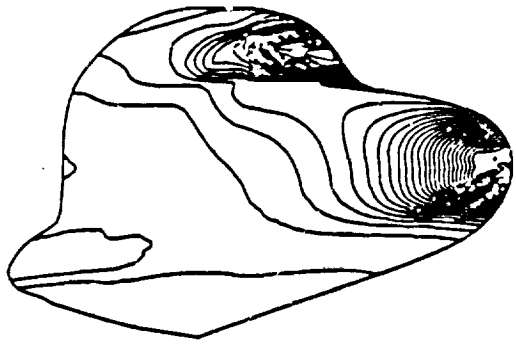
Figure 3. Computations-experiment comparison.
Evolution of the coefficient of pressure along the winward side.



a) M_∞=2.5 - α=0° - Roe solver

b) M_∞=2.5 - α=10° - Van Leer solver

Figure 4. Supersonic HERMES computations - Mach number contours on the body and in the plane of symmetry.



M_∞=12 - α=0° - Van Leer-Vinokur-Montagné solver.

Figure 5. Hypersonic HERMES forebody computations - Density contours on the body.

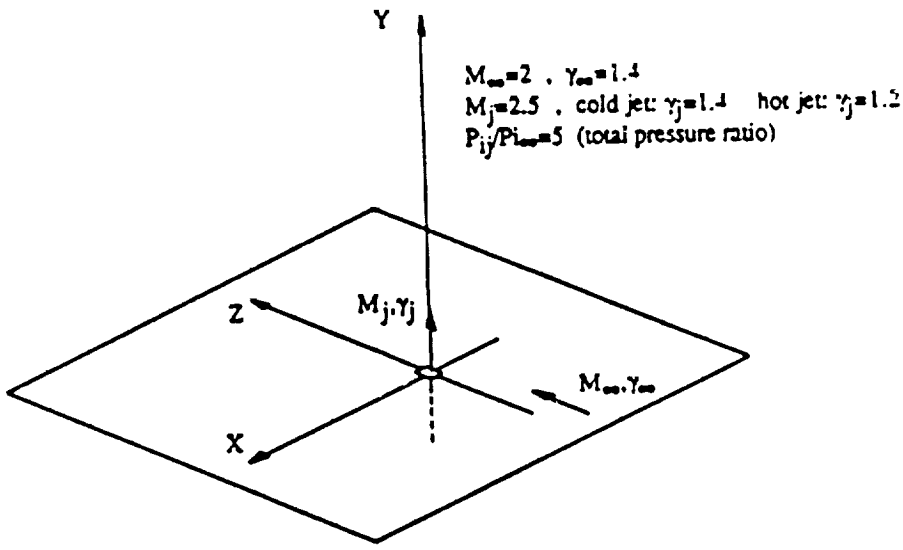


Figure 6. Aerodynamic conditions for the flat plate configuration.

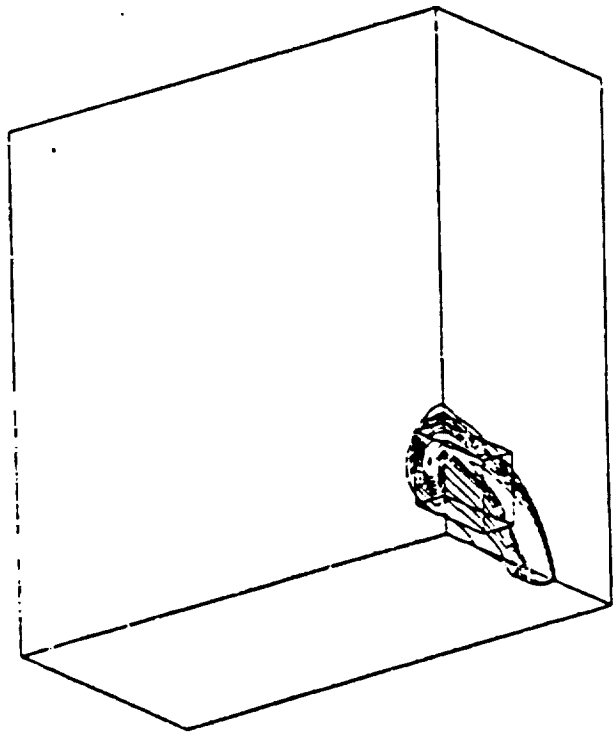
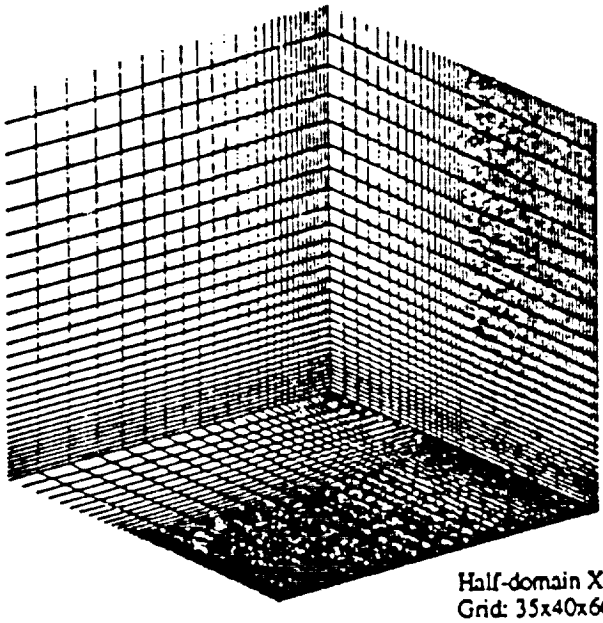
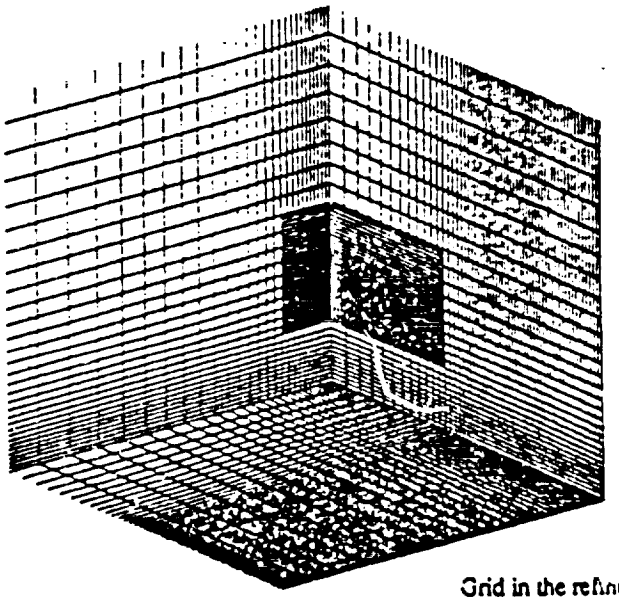


Figure 7. The frame.



Half-domain X>0
Grid: 35x40x60
66 gridpoints in the nozzle exit area

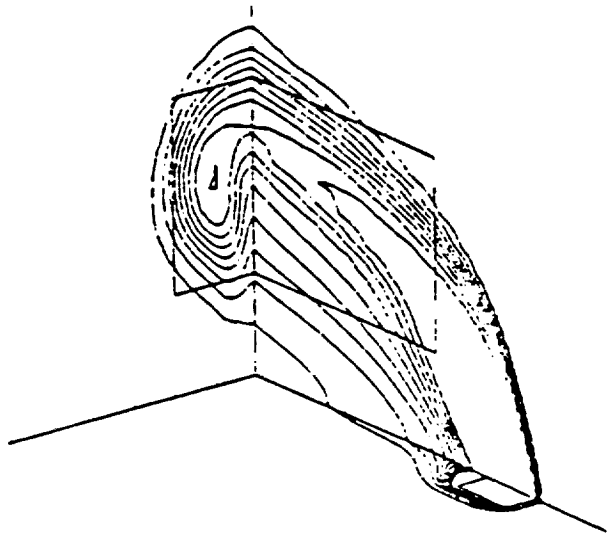
a) Coarse grid.



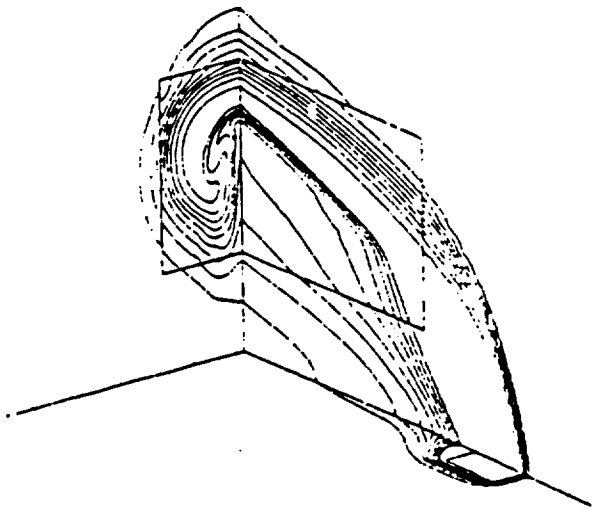
Grid in the refined area: 45x41x49

b) Refined grid.

Figure 8. Computational grids.

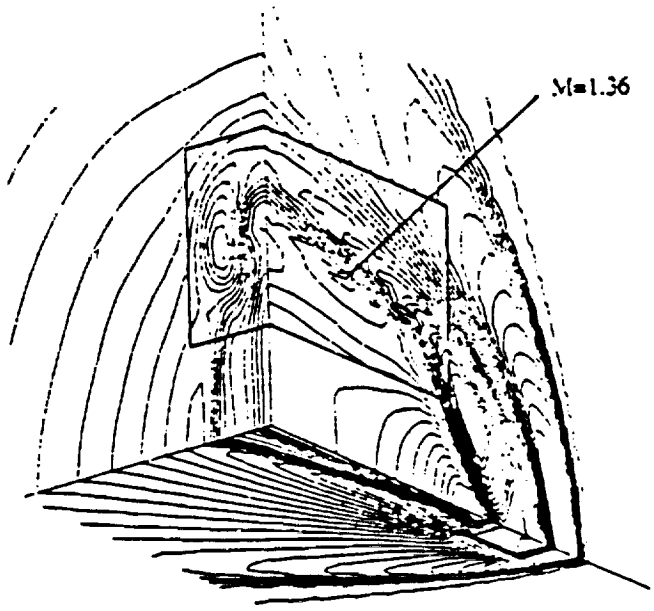


a) Coarse grid case.

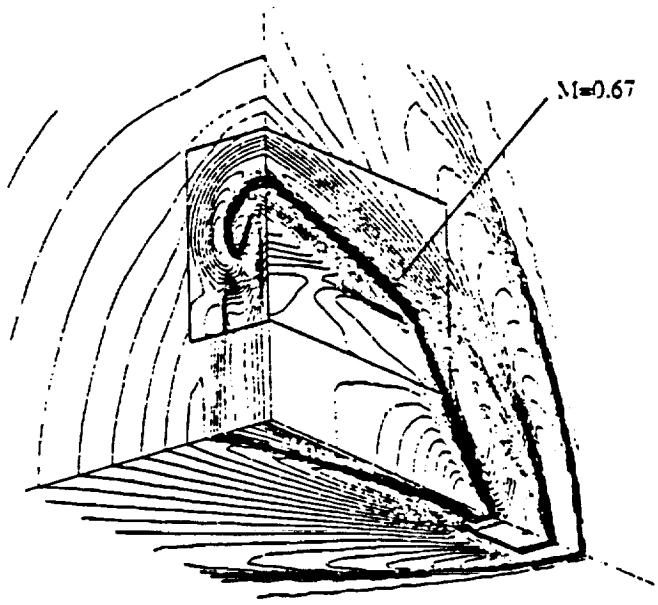


b) Refined grid case.

Figure 9. Concentration contours.



a) Coarse grid case.



b) Refined grid case.

Figure 10. Mach number contours.

ANNEXE B

SIMULATION NUMERIQUE D'ECOULEMENTS DE FLUIDE PARFAIT AUTOUR DE CONFIGURATIONS COMPLEXES 3D PAR UN SOLVEUR MULTIDOMAINE UTILISANT L'APPROCHE MUSCL

Par Ph. GUILLEN, M. BORREL et M. DORMIEUX

Numerical Simulation of perfect fluid flows around complex 3D Configurations by a multidomain solver using the MUSCL approach

Ph. Guillen, M. Borrel
ONERA
Aerodynamics Department
29 Av. de la Division Leclerc
92320, Chatillon sous Bagneux

and
M. Dormieux
Aerospatiale, DET
2,18 rue Béranger
92320, Chatillon sous Bagneux

General considerations

This poster presents a 3D Euler code developed for the numerical simulation of flows of gases of different natures around complex configurations with an emphasis on supersonic and hypersonic flows. The numerical scheme of the MUSCL type uses approximate Riemann solvers of the Van Leer, Roe, and Osher types which have been developed for perfect gas flows and recently extended to non reactive mixtures of two species and real gas flows by Abgrall, Montagne and Vinokur. The architecture of the code has been dictated by constraints concerning geometrical considerations, computational aspects, the specific nature of the flow, and ergonomy.

Geometrical considerations

The treatment of complex geometries has led us to adopt a multiblock grid made of several structured, possibly overlapping or patched, domains. This choice considerably simplifies the mesh construction. Another interesting possibility has been introduced to enable different kinds of boundary conditions on a given domain surface.

Computational aspects

The numerical scheme is based on the computation of isolated planes. This structure presents several advantages. Working arrays for the numerical scheme are plane addressed and hence not very expensive in core for present time computers. This can be very interesting especially in case of implicit algorithms. For future adaptation to parallel machines, synchronization waiting times are considerably decreased since every plane requires about the same amount of CPU, which is not the case for domains in practical cases.

Flow pattern

For supersonic flows it is very important to take advantage of the hyperbolic property of the steady Euler equations in the main direction by using space marching techniques whenever possible. Doing so the CPU time decreases by an order of magnitude(1). The fact that a plane of a domain can be computed separately allows to make multiblock space marching computations physical plane by physical plane without any additional effort. The presence of strong shocks in this kind of flow leads us to select upwind TVD schemes which have proved to be well suited for accurate capture of discontinuities.

Ergonomy

For this kind of application the order in which the domains are computed is not always straightforward. Some domains have to be converged before others, and the type of computation in each domain can be different (pseudo unsteady, space marching, time accurate...). Thus, in order to add flexibility and user-friendliness a command interpreter has been implemented.

It results in a code organisation built around 3 key units: a command interpreter which assumes the user interface, a plane monitoring unit which decides of the type of the computation, and a plane processor including the numerical scheme.

The numerical method

The unsteady 3D Euler equations are written in conservation law:

$$W_t+F_x+G_y+H_z=0$$

where, for instance, for a one specie perfect gas :

$$W={}^t(\rho,\rho u,\rho v,\rho w,e)$$
$$F={}^t(\rho u,p+\rho u^2,\rho uv,\rho uw,(e+p)u)$$
$$G={}^t(\rho v,\rho uv,p+\rho v^2,\rho vw,(e+p)v)$$
$$H={}^t(\rho w,\rho uw,\rho vw,p+\rho w^2,(e+p)w)$$

with $p=(\gamma-1)(e-1/2\rho(u^2+v^2+w^2))$

To solve them we use an implicit upwind TVD-finite-volume scheme of Van Leer MUSCL type. To use the maximum potential of the structured organisation of the grid and the plane organisation of the code, the implicit part is constitued by an ADI like inversion in each plane coupled with a Gauss-Seidel like relaxation in the third direction. Basically the scheme is constitued by the 3 following steps:

1) Introduction of a linear distribution for each direction in each cell to compute the cell interfaces:

$$U_{i+\tau\,j+\lambda\,k+\mu}=U_{i,j,k}+\tau\,g^i_{ijk}+\lambda g^j_{ijk}+\mu g^k_{ijk}$$

where $U=P^{-1}(W)$ is a set of variables, to preserve stability near discontinuities it is necessary to introduce limiters in each direction, if

$$\alpha_{i+1/2}=U_{i+1,j,k}-U_{i,j,k}$$
$$g^i_{ijk}=\text{limiter}(\alpha_{i+1/2},\alpha_{i-1/2})$$

Many limiters have been implemented among them the "Minmod"(2),"Van Leer("3),"Van Albada"(4), and "Superbee"(2). The set of variables can be chosen among the conservative, the primitive (ρ,u,v,w,p) or the characteristic one. Then the scheme assumes monodimensional TVD property.

2) Computation of the explicit part

$$\Delta W_{exp}=-\Delta t/Vol(F_{i+1/2}-F_{i-1/2}+G_{j+1/2}-G_{j-1/2}+H_{k+1/2}-H_{k-1/2})$$

where $F_{i+1/2}$ is an evaluation of the fluxes at an interface of the cell control volume.

This is made with the help of an approximate Riemann Solver between the two states on each side of the interface calculated in the first step.

Many approximate Riemann Solvers have been tested:

-For perfect gas: The Van Leer(5), Roe(6) and Osher(7) formulations

-For a mixture of non reactive two species gas: Abgrall(8) extension of Roe fluxes and Abgrall-Montagné(9) extension of Osher fluxes

-For real gas with an equilibrium assumption: Vinokur-Montagné(10) extension of Van Leer and Roe scheme, and Abgrall-Montagné(9) extension of Osher scheme.

3) Computation of the implicit part in each plane

$$A.\Delta W=\Delta W_{exp}$$

where A can be seen as an approximation of

$$(I+\Delta t\, dE/dw)$$

where E is an approximation of $F_x+G_y+H_z$

Two approximations have been tested, the linearized conservative implicitation of Steger-Warming or the linearized non conservative formulation of Harten-Yee(11) or Chakravarthy(12). The inversion is made by a DDADI algorithm (Diagonal Dominant Alternate Direction Implicit).

For Boundary Conditions many treatments have been considered, from Viviand-Veuillot(13) compatibility relations to more classical fluxes treatments and their implicitation(14).

After a description of the organization of the code, some typical test cases are presented. First, some results obtained by the implicit multidomain 3D algorithm on a classical test case consisting of a cylinder in a supersonic flow. Then some 3D results obtained with the explicit algorithm alone. The interaction of a transverse jet with a supersonic flow which enables to see differences obtained when using hot or cold jets (14) (γ=1.2 or γ=1.4) or when using different fluxes. The spreading of the iso-concentration lines which should be gathered on a thin surface theoretically, shows the diffusive aspect of the schemes. A computation to demonstrate the possibility to refine locally the mesh by the multidomain capability is shown. Finally some flowfields around Hermes at supersonic speeds or its forebody alone at hypersonic regime show the ability of the method of dealing with realistic geometries.

References

(1)M.Borrel, J.L.Montagné, J.Diet, Ph.Guillen, J.Lordon."Upwind Scheme for Supersonic Flows around Tactical Missiles".La Recherche Aerospatiale.1988-2.

(2)P.L.Roe."Some Contributions to the Modelling of Discontinuous Flows". Lectures in Applied Mathematics,vol. 22. (1985)

(3)B.Van Leer."A Second Order Sequel to Godunov Method". Journal of Computational Physics, vol. 23 pp. 276-299 (1977)

(4)B.Van Leer."II. Monotony and Conservation Combined in a Second Order Scheme". Journal of Computational Physics, vol.14 pp. 361-370. (1974)

(5)B.Van Leer."Flux Vector Splitting of The Euler Equations". Icase report 82-30 (1982)

(6)P.L.Roe."Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes". Journal of Computational Physics, vol. 43 pp. 357-372. (1982)

(7)S.Osher and F.Solomon."Upwind Difference Schemes for Hyperbolic Systems of Conservation Laws". Mathematics of Computation, vol. 38158 pp. 339-374. (1982)

(8)R. Abgrall. "Généralisation du Schéma de Roe pour le Calcul d'Ecoulements de Mélanges de Gaz à Concentrations Variables". La Recherche Aérospatiale n°6. (1988)

(9)R. Abgrall, J.L. Montagné. "Généralisation du Schéma d'Osher pour le Calcul d'Ecoulements de Mélanges de Gaz à Concentrations Variables et de gaz réels" . To be published.

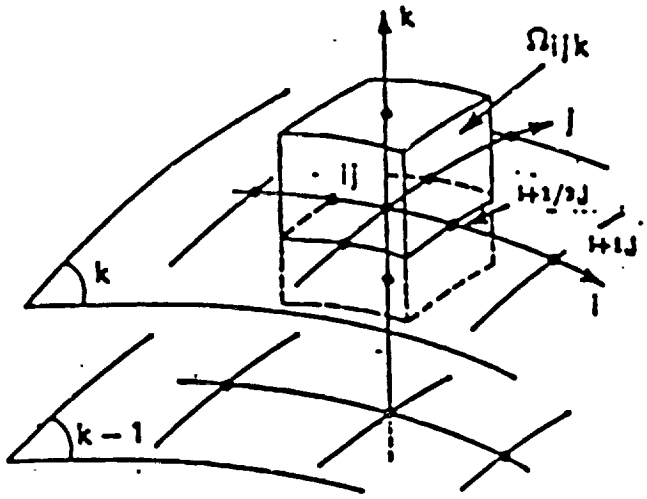
(10)J.L. Montagné, H.C. Yee, M. Vinokur. "Comparative Study of High-Resolution Shock-Capturing Schemes for a Real Gas". Proceedings of the 7th GAMM Conference on Numerical Methods in Fluid Mechanics, Sept. 9-11, 1987.

(11)H.C. Yee, A. Harten. "Implicit TVD Schemes for Hyperbolic Conservation Laws in Curvilinear Coordinates". AIAA 85-1513.

(12)S.R. Chakravarthy. "High Resolution Upwind Formulations for the Navier-Stokes Equations". VKI lecture 1988-05.

(13)H. Viviand, J.P. Veuillot. "Méthodes Pseudo-Instationnaires pour le Calcul d'Ecoulements Transsoniques". Publication Onera n°1978-4.

(14)M. Borrel.Characteristic Boundary Conditions. To be published.



Control-Volume for the Scheme

command system
-Initialization
-States definition
-Output
-Core monitoring
-Boundary condition definitions
-Scheme parameter definitions
-Control instructions
-Plane monitoring calls

Magnetic Disks

Definitions
N^3: number of nodes including fictitious points
N^2: number of nodes in a plane
N_f^2: number of points to be coupled
neq: number of equations of the system to be solved

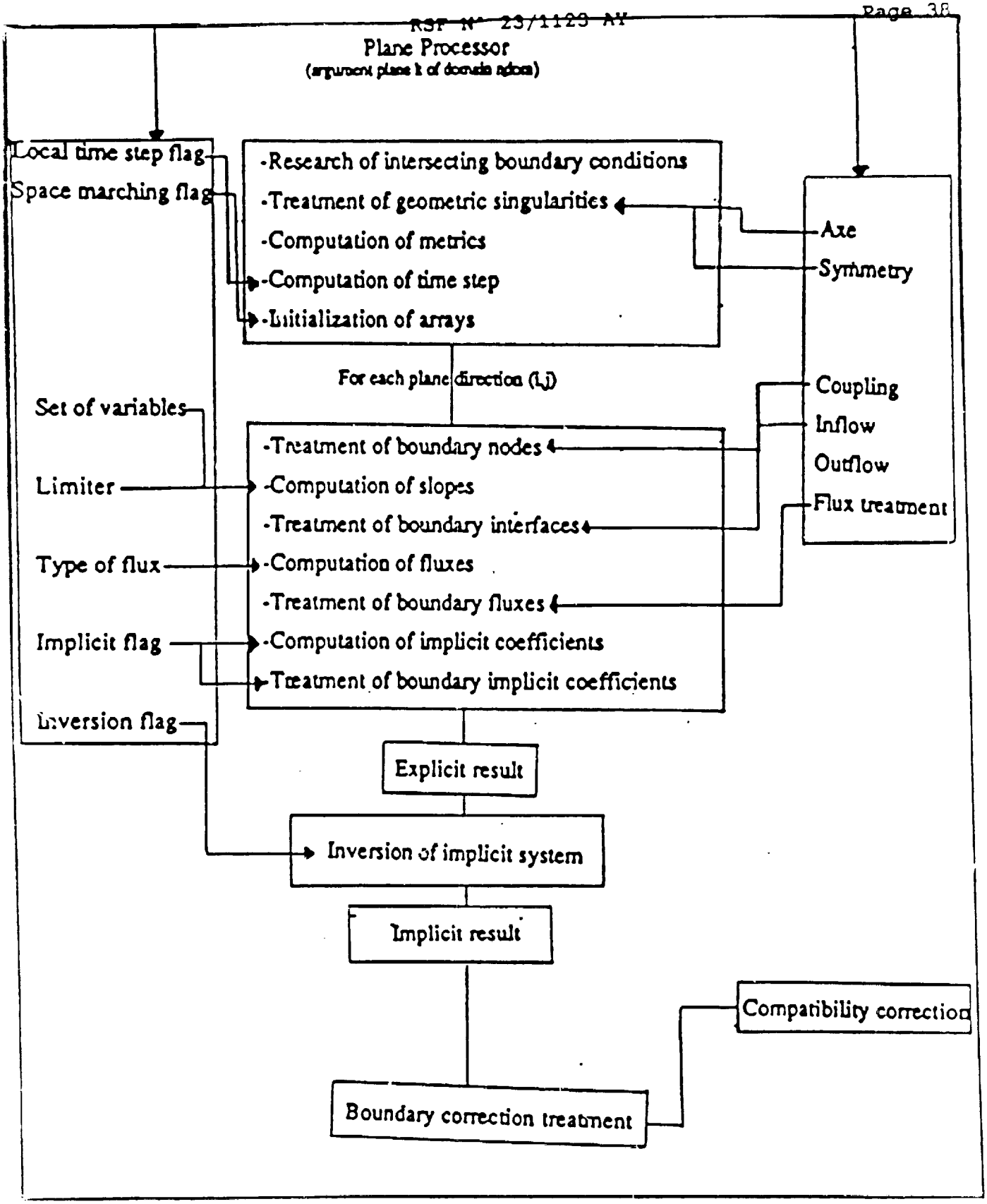
3D Data

Conservatives variables
neq N^3
Geometric coordinates
$3N^3$

Boundaries Data
1 node address + 3 trilinear interpolation coefficients for each node involved in the coupling ($N_f 2$)

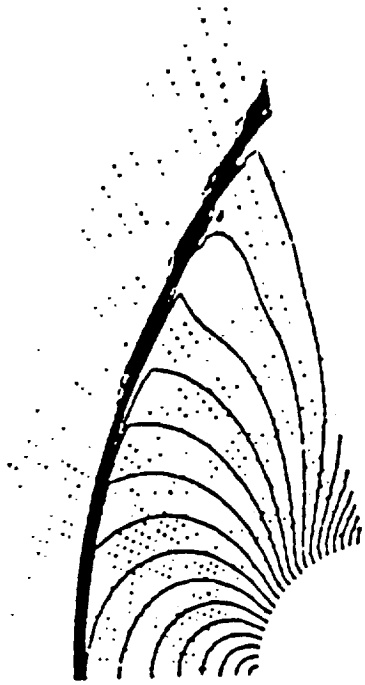
Plane Monitoring

- Multidomain Pseudo-Unsteady computation
- Time-accurate computation
- Multidomain Space-Marching computation

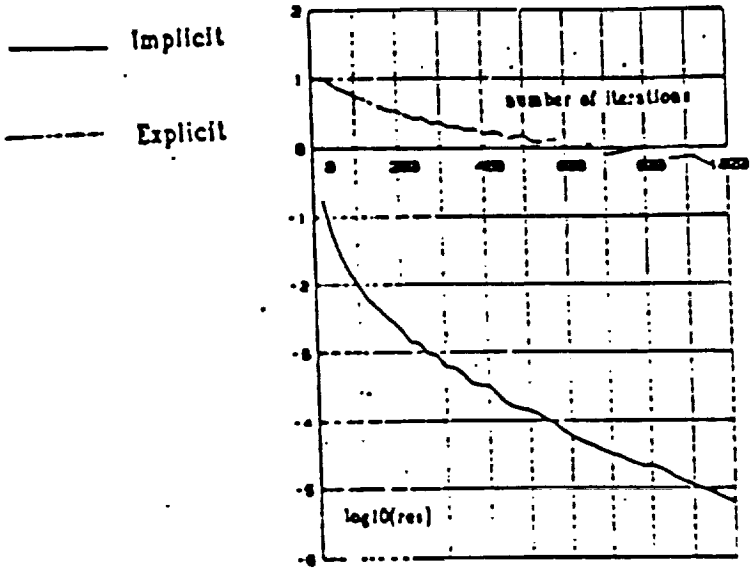

Plane Processor
(argument plane k of domain ndom)
Local time step flag
Space marching flag
-Research of intersecting boundary conditions
-Treatment of geometric singularities
-Computation of metrics
-Computation of time step
-Initialization of arrays
Axe
Symmetry
For each plane direction (i,j)
Coupling
Inflow
Outflow
Flux treatment
Set of variables
Limiter
Type of flux
Implicit flag
Inversion flag
-Treatment of boundary nodes
-Computation of slopes
-Treatment of boundary interfaces
-Computation of fluxes
-Treatment of boundary fluxes
-Computation of implicit coefficients
-Treatment of boundary implicit coefficients
Explicit result
Inversion of implicit system
Implicit result
Compatibility correction
Boundary correction treatment

IMPLICIT MULTIDOMAIN

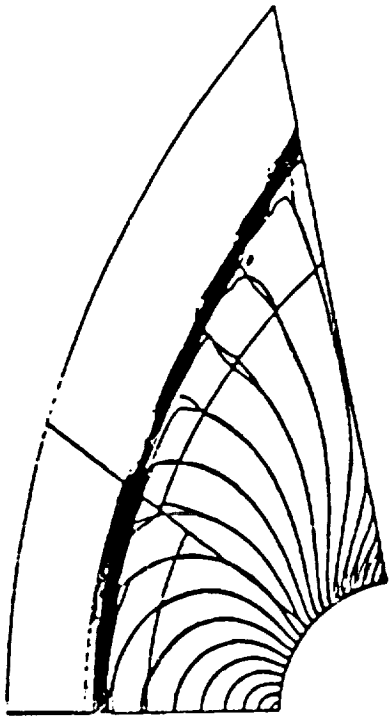
Monodomain Results



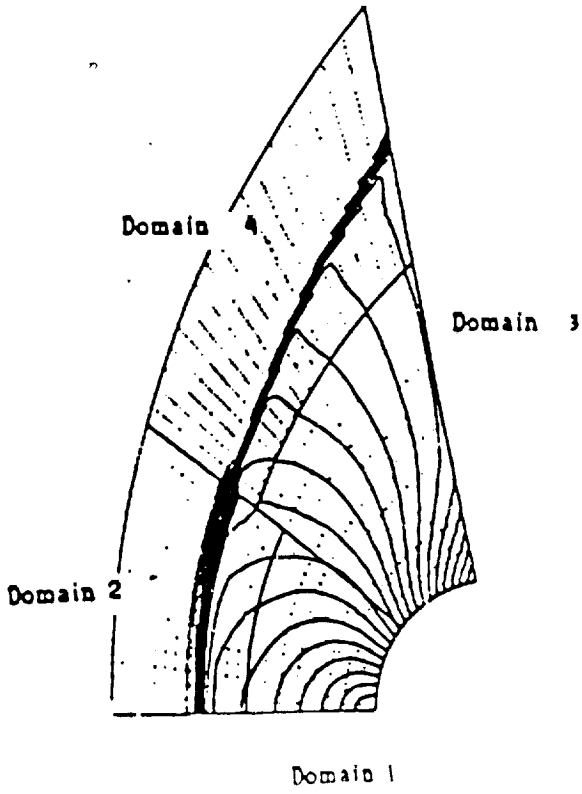
Monodomain explicit and implicit Residuals



Superposition of Four-Domains and Monodomain Results

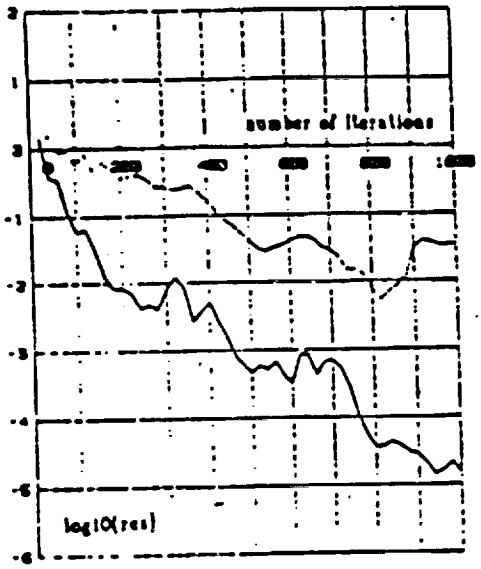


Four-Domains Results

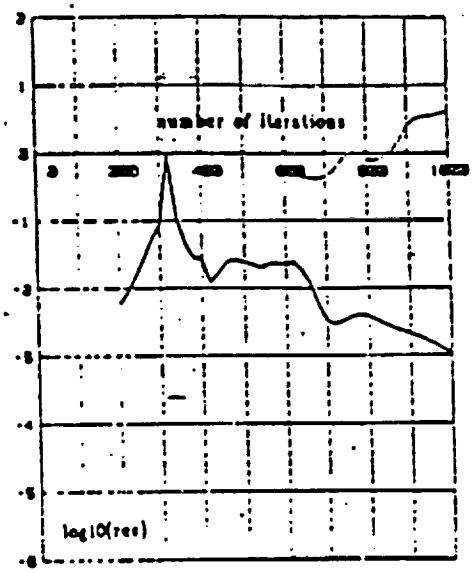


Convergence curves

Domain 1



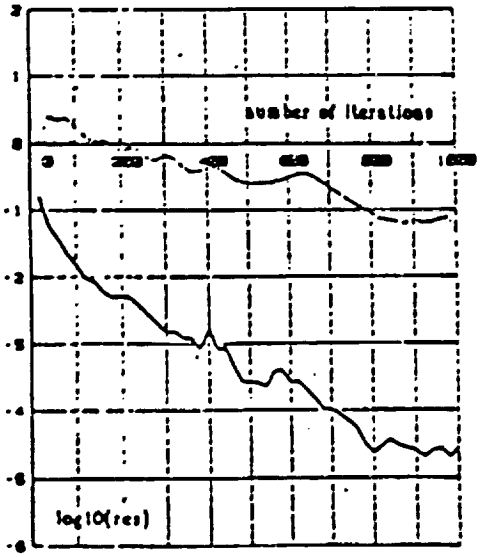
Domain 2



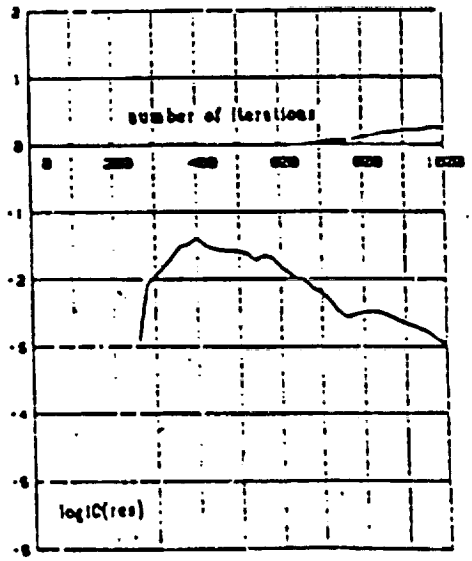
Implicit

Explicit

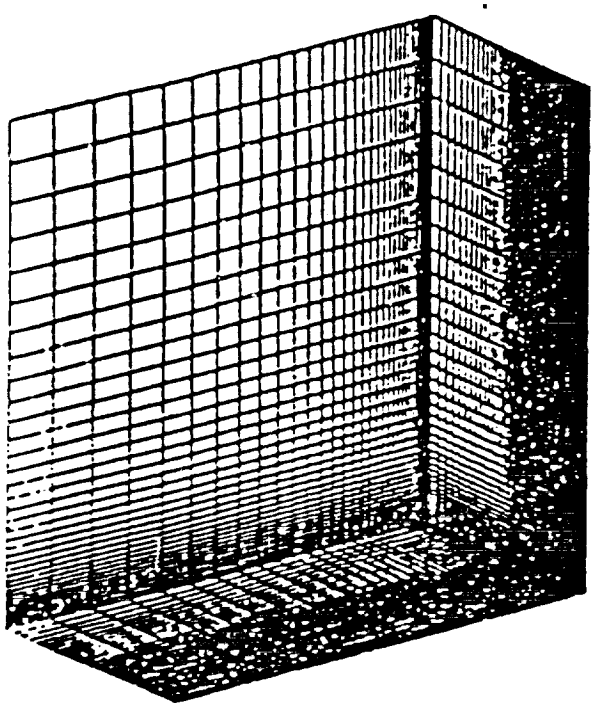
Domain 3



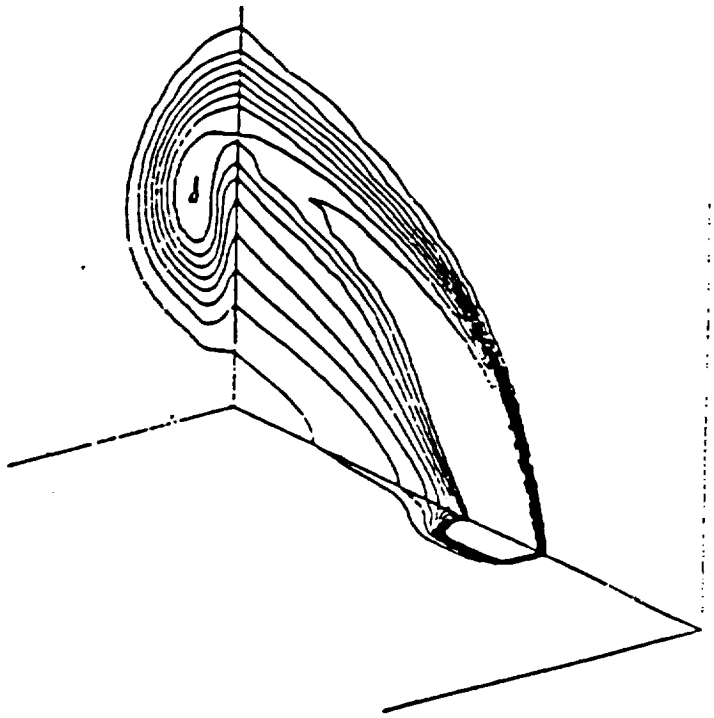
Domain 4



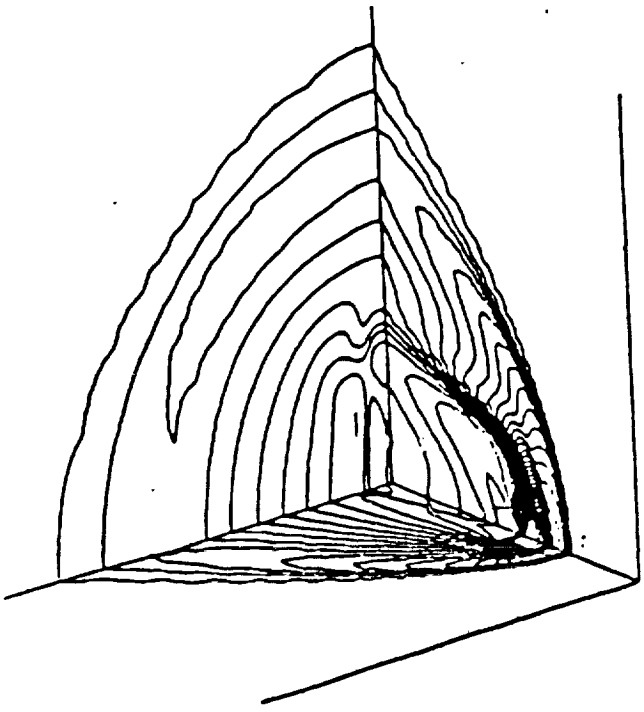
Monodomain Mesh (35x40x60 points)



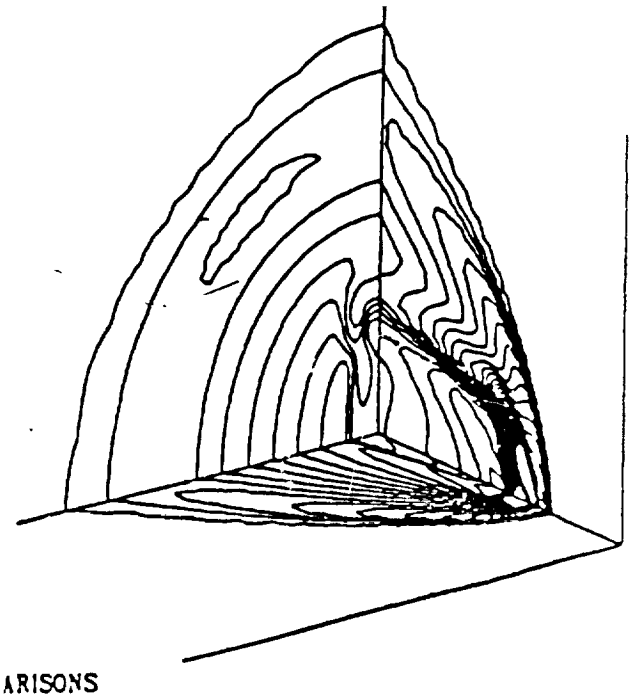
Hot Jet - Iso-Concentration (Roe)



Cold Jet - Iso-Density (Roe)

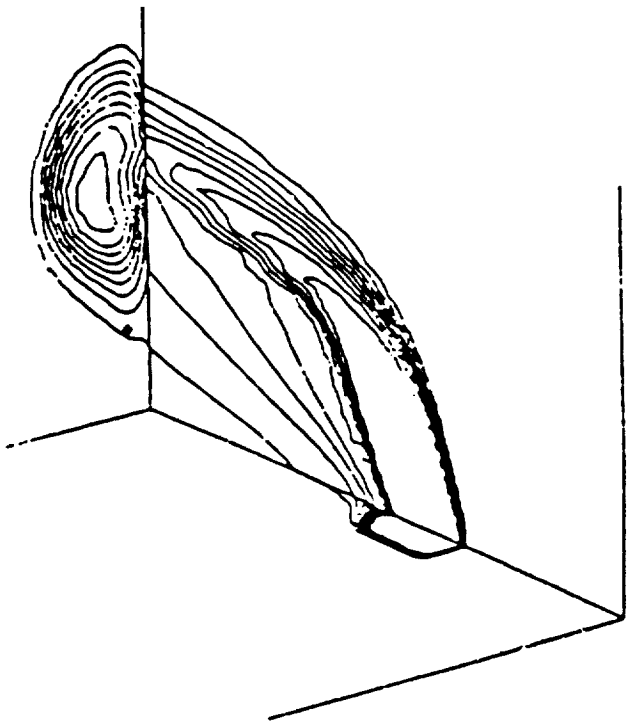


Hot Jet - Iso-Density (Roe)

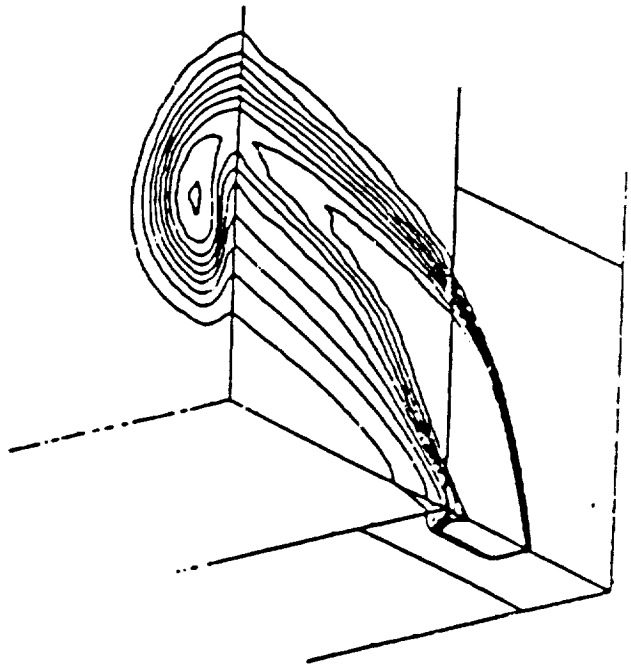


COLD AND HOT JETS COMPARISONS

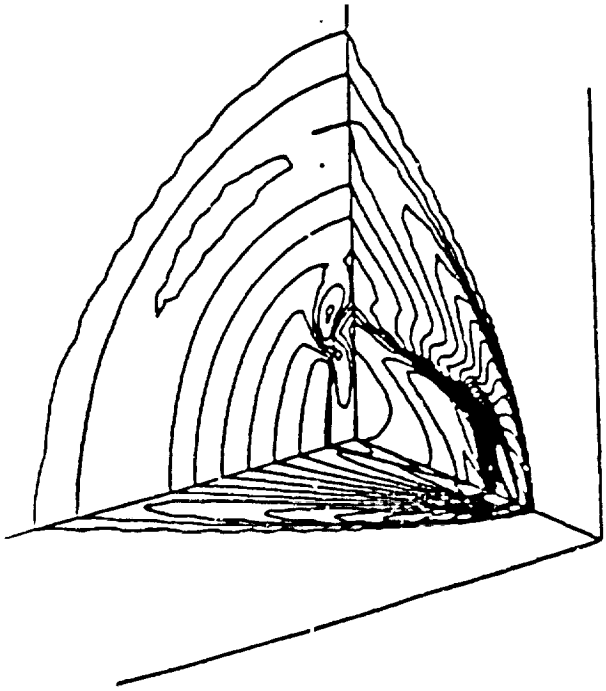
Hot Jet - Iso-Concentration (Osher)



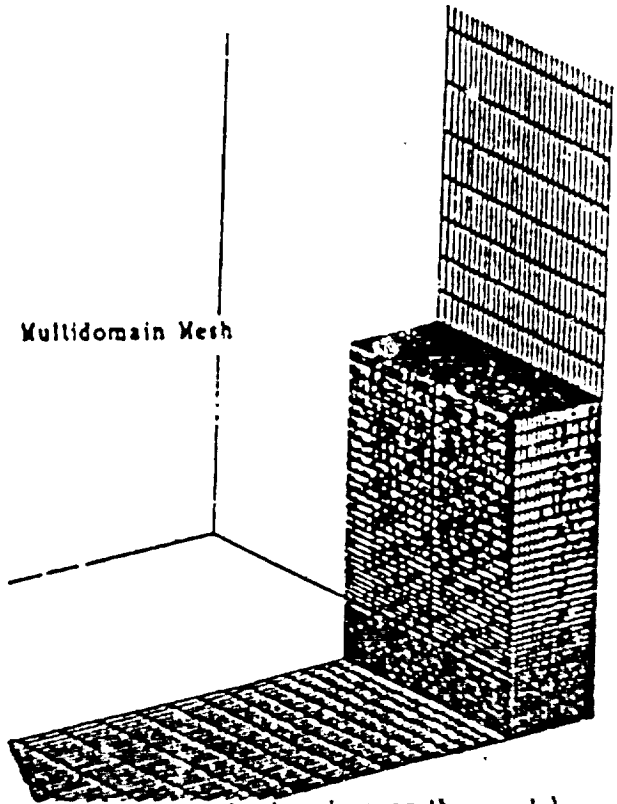
Hot Jet - Iso-Concentration (Roe,Multidomain)



Hot Jet - Iso-Density (Osher)

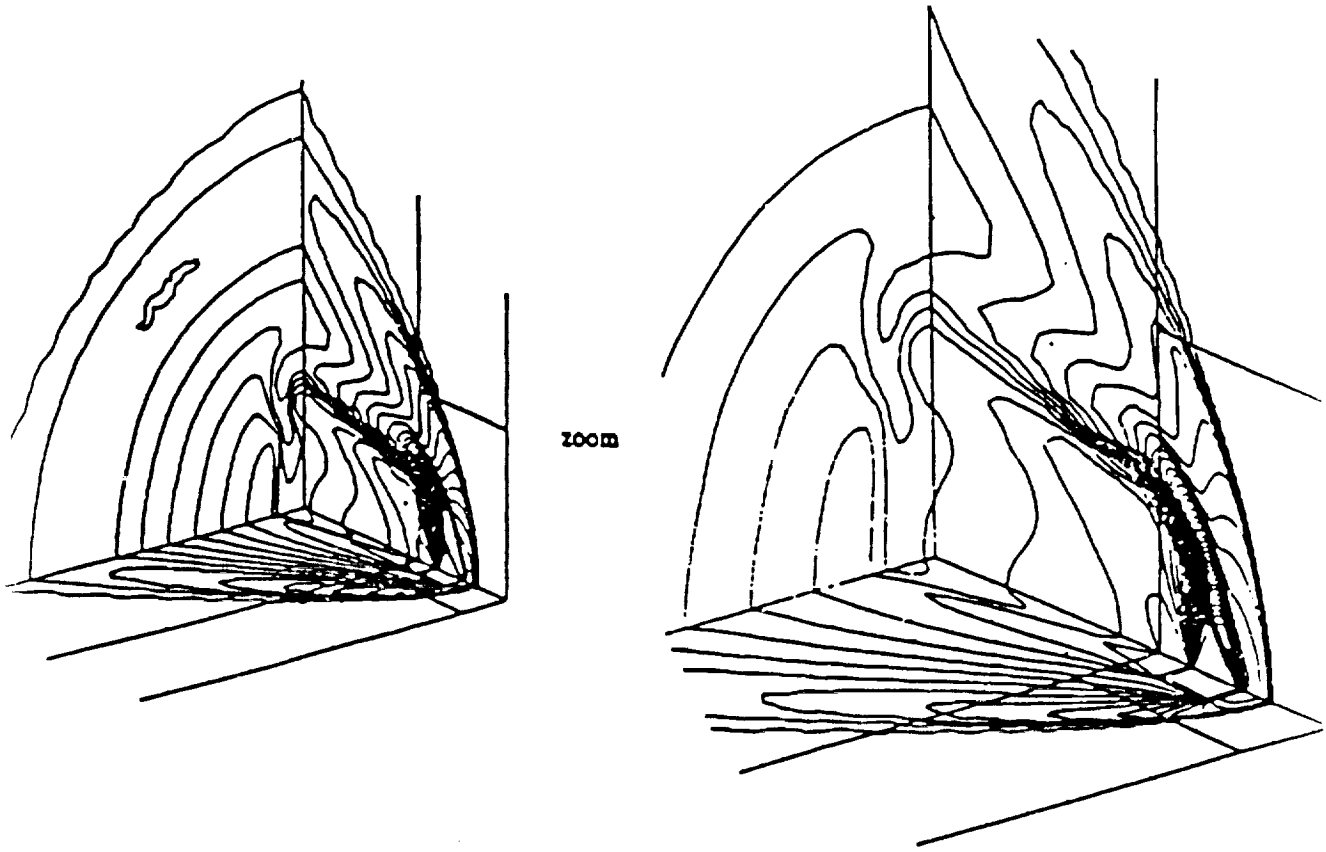


Multidomain Mesh



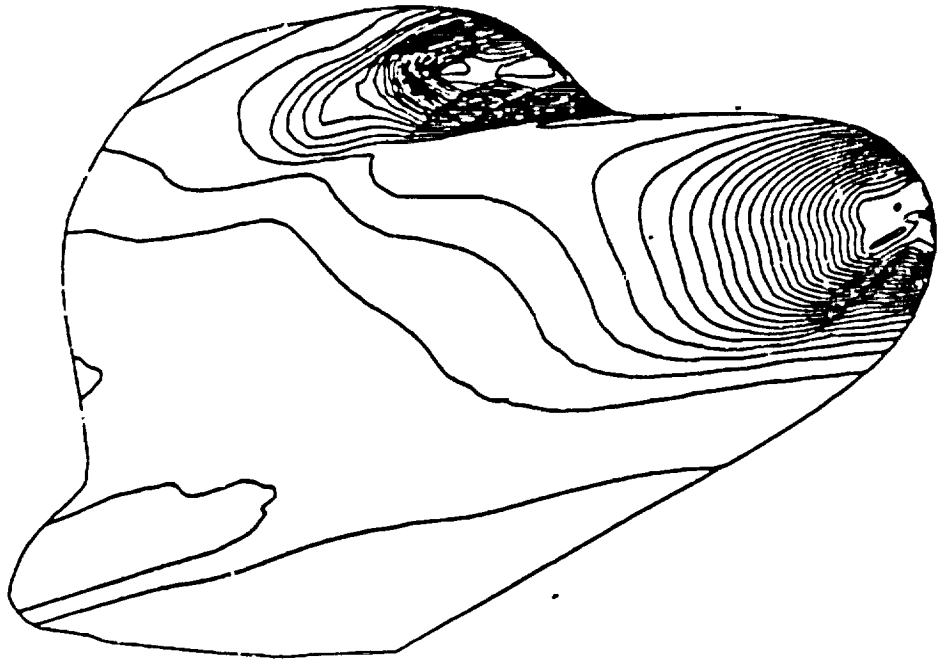
(23x45x63 points for the domain near the nozzle)

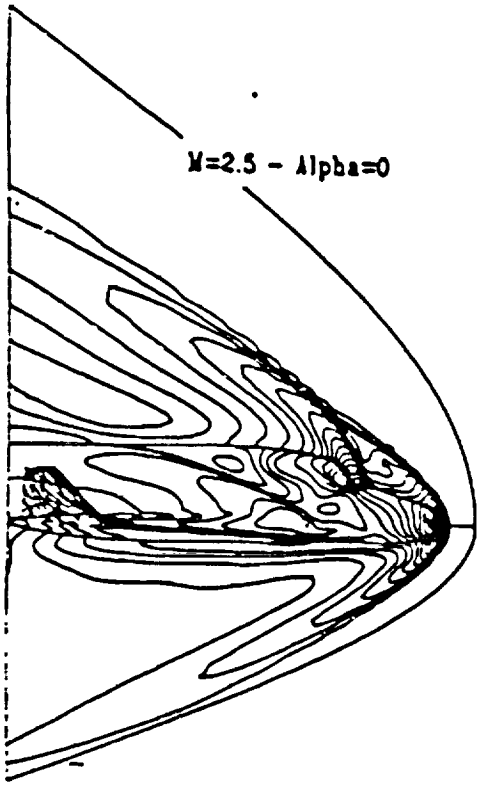
Hot Jet - Iso-Density (Roe,Multidomain)



HYPERSONIC HERMES FOREBODY COMPUTATIONS

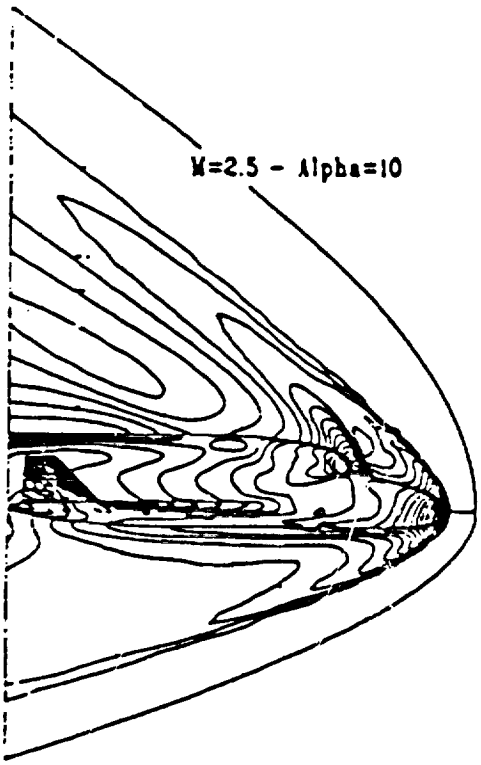
Iso-Density Contours M=12 - Alpha=0 Van Leer-Vinokur-Montagne Fluxes Srinivasan Tables





Roe Fluxes

Iso-Mach Contours



Van Leer Fluxes

Iso-Mach Contours

See also
89A 45187

ANNEXE C

UNE NOUVELLE METHODE VARIATIONNELLE POUR LA GENERATION DE MAILLAGES ADAPTES BI ET TRIDIMENSIONNELS EN MECANIQUE DES FLUIDES

Par O.P. JACQUOTTE et J. CABELLO

A NEW VARIATIONAL METHOD FOR THE GENERATION OF TWO- AND THREE DIMENSIONAL ADAPTED GRIDS IN COMPUTATIONAL FLUID DYNAMICS

Olivier-Pierre JACQUOTTE and Jean CABELLO
Aerodynamics Department
Office National d'Etudes et de Recherches Aérospatiales (ONERA)
B.P. 72 - 92322 CHATILLON FRANCE

SUMMARY

A variational method for the optimization and adaptation of structured grids is presented. This method is applicable to two- or three-dimensional structured mesh as well as to surface mesh. It relies upon the introduction of a proper measure of a cell deformation that is derived from basic principles of continuum mechanics. Several properties are prescribed, which guarantee the well-posedness of the mesh optimization problem and the efficiency of the solution algorithm. The ability of the method for the mesh adaptation is also described. Examples of two- and three-dimensional grids are shown and illustrate the success of the presented method.

1. INTRODUCTION

In recent years, remarkable progress has been observed in Computational Fluid Mechanics; first, they have concerned the complexity of the physical phenomena and enable us to consider the solution of turbulent flow equations including chemical effects. They have also concerned the improvement of numerical algorithms for the computation of approximate solutions so as to obtain more efficiently more accurate solutions. Finally, a new field of research - generation and optimization of grids necessary to the computations - is now being thoroughly investigated, and even becomes a priority for applied CFD.

Numerous strategies have been developed and more and more sophisticated algorithms have been introduced for the optimization of mesh, e.g. [1, 2]. In particular, adaptation has recently become a very important topic of research. This feature allows for instance a redistribution of a fixed number of nodes so as to concentrate the points in regions where large variations of the solution occur, and thus increase the accurracy of the solution with a fixed number of nodes. However this localized refinement cannot be accomplished in any arbitrary way and the quality of the mesh (regularity, non skewness) must be carefully controlled. A method able to meet these requirements would generate the "optimal mesh" that would enable us to obtain the best possible solution for a fixed computational effort [3]. In this paper, we briefly review a method developed in an attempt to more sharply resolve these questions (see also [4-6]) and we show the latest results it has provided. The method realizes a tradeoff between the geometrical quality of the mesh, the ability to handle adaptation with respect to a given criterion, and the robustness and efficiency of the computational algorithm.

2. FORMULATION FOR A GRID GENERATOR

A measure of the grid deformation

The objective of the method is first to obtain a structured and regular mesh as orthogonal as possible in a curvilinearly quadrilateral domain in R^2 or on a surface in R^3, or in a hexahedral domain in R^3; the ability of the method to adapt a mesh is obtained in a simple way as it will be shown. In order to reach the objective, we consider the mesh generation as the finite element discretization of a continuous problem consisting in finding a transformation $x(\xi)$ from the reference unit cube (space ξ) into the domain to be meshed (space x), and we want to exhibit a proper measure σ of this transformation. Four axioms and geometrical properties can be stated [4]; they are basic principles of continuum mechanics [7], and they

allow the definition of functionals that are not defined in terms of regularity, orthogonality or cell skewness, but rather in terms of measure of the deformation of the current cell (with respect to a reference cell for instance); these functionals take into account the whole metric of the grid and thus define new criteria that measure the mesh quality. Indeed, it can be shown that functionals satisfying these axioms must only depend on the invariants I_1, I_2, I_3 of the left Cauchy-Green tensor C of the transformation $\mathbf{x}(\xi)$:

$$\sigma = \sigma (I_1, I_2 , I_3) \tag{1}$$

where

$$I_1 = \text{tr } \mathbf{C} \; ; \; I_2 = \text{tr Cof } \mathbf{C} \; ; \; I_3 = \det \mathbf{C} \tag{2}$$

and

$$\mathbf{F} = \nabla \mathbf{x} \; ; \; \mathbf{C} = \mathbf{F}^t . \mathbf{F} \tag{3}$$

Furthermore, we want the orientation of the cell to be taken into account; for this, it is necessary to make σ sensitive to this orientation, that is to say to the sign of the transformation Jacobian J, and consider functions depending not only on I_3, but on J:

$$\sigma = \sigma (I_1, I_2 , J) \tag{4}$$

with

$$J = \det \mathbf{F} \; ; \; I_3 = J^2 \tag{5}$$

Characterization of the function of the invariants

The next step is to ensure the well-posedness of the minimization of these functionals: in order to do so, we impose new properties relative to the derivatives of σ for the so-called rigid body modes which preserve the shape and the orientation of the cell. We impose a normalization (σ vanishes for any rigid body transformation), an equilibrium condition (the measure is stationary for any rigid body transformation), and most importantly a convexity condition (the measure is convex around any rigid body transformation). These conditions, and in particular the convexity, are the proper mathematical properties that ensures the well-posedness of the minimization of σ as well as the efficient convergence of numerical algorithms towards a unique minimum. They also restrict the possible choice for the function of the invariants [4]; among the functions satisfying the geometrical axioms and these properties, several polynomials can be exhibited, in particular:

$$\sigma = C_1 (I_1 - I_3 - 2) + C_2 (I_2 - 2 I_3 - 1) + C_3 (J - 1)^2 \tag{6}$$

where the constants must verify:

$$3 C_3 > 4 (C_1 + C_2) > 0 \tag{7}$$

Furthermore, it is interesting to point out two classes of polynomials that lead to geometrical interpretations.

- In two dimensions, one considers the function

$$\sigma_{2d} = C (I_1 - I_3 - 2) + K (J - 1)^2 \quad \text{with} \quad K > C > 0 \tag{8}$$

This can be re-written as:

$$\sigma_{2d} = C (I_1 - 2 J) + (K - C) (J - 1)^2 \tag{9}$$

The first term $(I_1 - 2 J)$ can be interpreted when one considers the matrix F and its invariants; we have:

$$\mathbf{F} = \begin{bmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{bmatrix} ; \; I_1 = x_\xi^2 + x_\eta^2 + y_\xi^2 + y_\eta^2 ; \; J = x_\xi y_\eta - x_\eta y_\xi \tag{10}$$

and therefore:

$$I_1 - 2 J = (x_\xi - y_\eta)^2 + (x_\eta + y_\xi)^2 \tag{11}$$

This term represents a least square formulation of the Cauchy-Riemann relations that ensures the conformity of the mesh:

$$x_\xi - y_\eta = 0 \; ; \; y_\xi + x_\eta = 0 \tag{12}$$

and that can also be written:

$$\mathbf{x}_\xi = \mathbf{x}_\eta \times \mathbf{k} \tag{13}$$

Furthermore this functionr can be used to construct a "quasi-conformal" grid: up to this point, the quantity σ was measuring the deformation of a cell with respect to a square reference cell. This can however be generalized and it is possible to consider a rectangular reference cell; this allows the prescription of a ratio "b/a" for each cell; generalized Cauchy-Riemann equations then writes:

$$\mu\, b\, x_\xi - a\, y_\eta = 0 \;;\; \mu\, b\, y_\xi + a\, x_\eta = 0 \tag{14}$$

where μ is a conformity parameter that needs to be introduced in order to have existence of the quasi-conformal solution and that is computed by:

$$\mu^2 = \frac{\int_\Omega (a/b)\,(x_\eta^2 + y_\eta^2)\, d\xi\, d\eta}{\int_\Omega (b/a)\,(x_\xi^2 + y_\xi^2)\, d\xi\, d\eta} \tag{15}$$

This method allows the construction of grids with arbitrary refinements on the sides such as the one presented on Fig.1 which Arina also obtained [8] by directly solving the equations (14, 15).

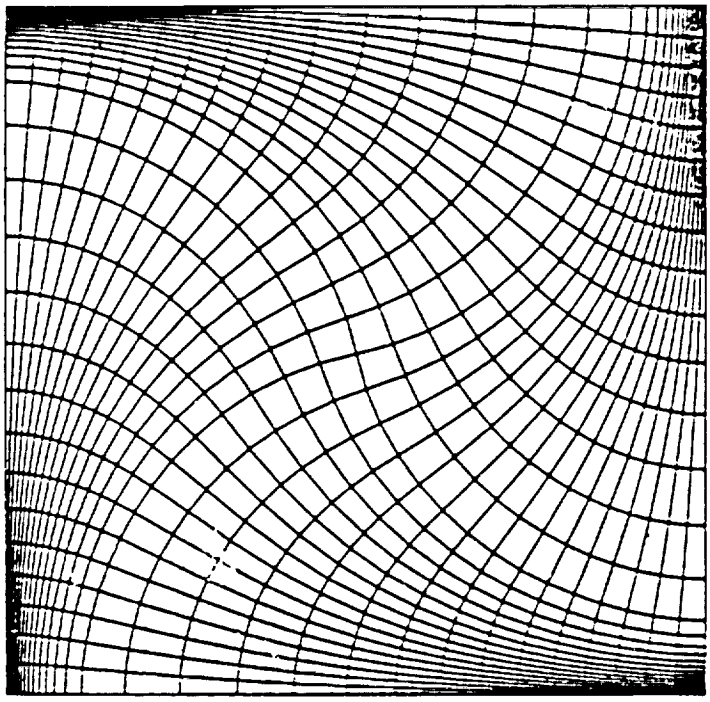


Fig.1: Quasi-Conform Mesh with Arbitrary Refinements on the Boundaries.

The second term in (8), $(J-1)^2$, is interpreted as a least square formulation of a constraint $J = 1$; indeed this penalty term is used as a volume control term that prevents the cell volume V from getting too far from a reference value V_{ref} ($J = V / V_{Ref}$), in particular this term avoids the cell to overlap.

In two dimensions but on a surface in $\mathbf{R}^3$, the functional are still valid, and in particular, the term $I_1 - 2J$ represents a least square formulation of generalized Cauchy-Riemann relations on surfaces [6]:

$$\mathbf{x}_\xi = \mathbf{x}_\eta \times \mathbf{n} \tag{16}$$

where $\mathbf{n}$ is the unit vector normal to the surface and defining its orientation.

- In three dimensions, one considers the case where:

$$C_1 = C_2 = C \quad \text{and} \quad C_3 = 3\,C + K \tag{17}$$

This leads to the function σ_{3d}:

$$\sigma_{3d} = C\,(\,I_1 + I_2 - 6\,J\,) + K\,(\,J-1\,)^2 \qquad (18)$$

that is precisely a least-square formulation of a property that characterizes a rigid body transformation: indeed, a direct orthogonal matrix satisfies:

$$\mathbf{F}^t = \mathrm{Cof}\,\mathbf{F} \quad \text{and} \quad \det \mathbf{F} = +1 \qquad (19)$$

and therefore the least-square formulation holds since:

$$C\,\|\,\mathbf{F}^t - \mathrm{Cof}\,\mathbf{F}\,\|^2 + K\,(\det \mathbf{F}-1)^2 = C\,(\,I_1 + I_2 - 6\,J\,) + K\,(\,J-1\,)^2 \qquad (20)$$

Similarly to the comments previously made, we may notice that the first term in (18) is a least-square formulation of non-linear relations:

$$\mathbf{x}_\xi = \mathbf{x}_\eta \times \mathbf{x}_\zeta \quad + \text{circular permutations of } \xi, \eta, \zeta \qquad (21)$$

that may be considered as generalized Cauchy-Riemann relations in three dimensions. Investigations are under way to look into this idea more carefully.

Mesh adaptation functional

The functionals previously described have in common the volume control term:

$$\sigma_{vol} = (\,J-1\,)^2 \qquad (22)$$

It was first interesting to use this term to control the volume in selected areas and thus adapt the mesh with respect to a given criterion. This is done by minimization of the functionals where the volume control term σ_{vol} has been replaced by an adaptation term σ_{adapt} such as:

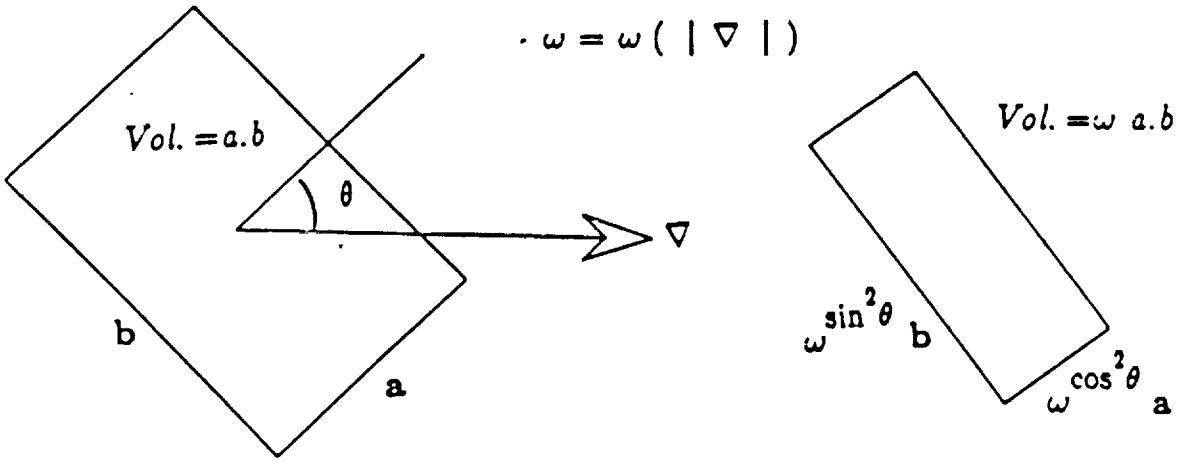
$$\sigma_{adapt} = (\,\omega\,J-1\,)^2 \qquad (23)$$

In this expression, the weight ω can be computed from a posteriori error estimates [1] or, if this kind of data is not available, as a function of a physical quantity gradient describing the solution; in that case, a new computation of the solution on this adapted mesh will improve the accuracy on high gradients and thus the quality of the solution.

Furthermore, it appears natural to take the direction of the gradient into account and to introduce non-isotropic refinements. Let's consider for simplicity an initial 2-dimensional mesh, approximately orthogonal, where a current cell can be described by its side vectors a and b; the non-isotropic refinement consists in measuring the deformation of the current cell in the adapted mesh with respect to a reference cell with sides:

$$\omega^{\sin^2\theta}\,\mathbf{a}\,,\ \omega^{\cos^2\theta}\,\mathbf{b} \qquad (24)$$

where θ is the angle between a and the gradient vector (Fig.2.). For instance if θ vanishes, the refinement is accomplished in the a-direction, as expected.



a) Current Cell in the Initial Mesh b) Reference Cell for the Adapted Mesh

Fig. 2: Non-Isotropic Adaptation.

3. NUMERICAL EXAMPLES

The solution algorithm and several results that this method has produced have been presented by the authors in previous publications. These results showed in particular its robustness (ability to untangle ramdom initializations [4]), its ability to regularize a mesh in the neighborhood of singularities [4] or to adapt a mesh with respect to a criterion defined by several physical variables [5, 6]. Here we present two examples that illustrate the behavior of the method for the adaptation of 2- and 3-dimensional grids.

The first example presented shows the ability of the method to adapt a grid around a physical phenomenon. A bow shock develops in the solution of the Euler equations for a supersonic flow past a cylinder. A first solution can be computed on an initial mesh (Fig. 3a), and a weight ω obtained as a function of the gradient of the pressure; the minimization of the constructed adaptivity functional leads to a mesh that effectively refines in the bow shock area (Fig. 3b). The solution of the equations on this new mesh leads to a much more accurate solution than the one obtained on the initial grid [6].

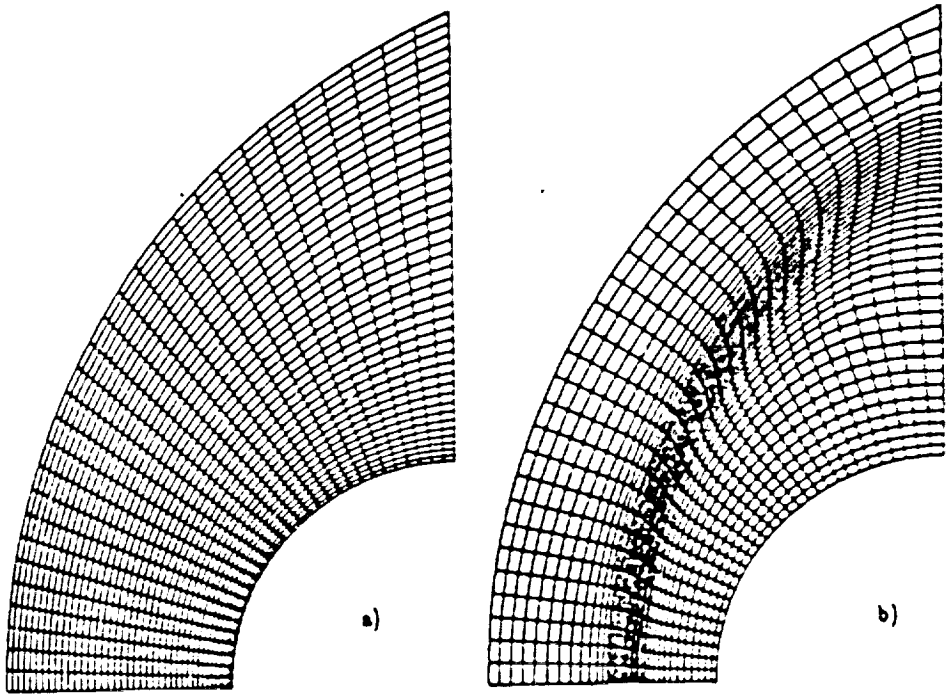


Fig.3: Adaptation around a Bow Shock

A second realistic example is proposed: it regards the three-dimensional adaptation to a bow-shock developing in front of a re-entry vehicle in a flow at Mach 5 [9]. The Fig. 4c shows the solution computed on an initial 31×20×28 point mesh (Fig. 4a), the mesh is adapted with respect to this solution (Fig. 4b). The quality of the solution computed on this adapted mesh has been very much improved (Fig. 4d): in particular both the bow-shock and the cockpit-attached shock are more precisely described.

These examples, and more generally the gratifying results that this method have produced, have shown its robustness, its efficiency and its ability to improve the solution of complicated flow problems: we think that these features are due to a sound mathematical basis which is necessary to achieve progress in the development of grid generation techniques.

REFERENCES

[1] J.U. BRACKBILL and J.S. SALTZMAN, "Adaptive Zoning for Singular Problems in Two Dimensions", J. Comput. Phys., Vol. 46, pp. 342-368, 1982.

[2] R. CARCAILLET, G.S. DULIKRAVITCH and S.R. KENNON, "Generation of Solution Adaptive Computational Grids using Optimization", Comput. Meths. Appl. Mech. Engrg., Vol. 57, pp. 279-295.

1986.
[3] J.T. ODEN, T. STROUBOULIS and P. DEVLOO, " Adaptive Finite Element Methods for the Analysis of Inviscid Compressible Flow: Part I. Fast Refinement/Unrefinement and Moving Mesh Methods for Unstructured Meshes", Comput. Meths. Appl. Mech. Engrg., Vol. 59, pp. 327-362, 1986.
[4] O.-P. JACQUOTTE, "A Mechanical Model for a New Generation Method in Computational Fluid Mechanics", Comput. Meths. Appl. Mech. Engrg., Vol. 66, pp. 323-338, 1988.
[5] O.-P. JACQUOTTE and J. CABELLO, "Optimisation et Adaptation de Maillages Structurés", Calcul des Structures et Intelligence Artificielle, Vol. 2, Ed. J.M. Fouet, P.Ladeveze and R. Ohayon, Editions Pluralis, 1988, also ONERA T.P. No 1988-75.
[6] O.-P. JACQUOTTE and J. CABELLO, "A Variational Method for the Optimization and Adaptation Grids in C.F.D.", Second International Conference on Numerical Grid Generation in C.F.D., Miami-Beach, Florida, U.S.A., December 5-8, 1988, Pineridge Press, also ONERA T.P. No 1988-99.
[7] P.G. CIARLET, Elasticité Tridimensionnelle, Masson, Paris, 1986.
[8] R. ARINA, "Adaptive Orthogonal Curvilinear Coordinates", Conference on Numerical Methods for Fluid Dynamics, Oxford, U.K., March 21-24, 1988, K. Morton and M.J. Baines eds., Oxford Univ. Press.
[9] M. BORREL, J.L. MONTAGNE, J.DIET, Ph. GUILLEN and J. LORDON, "Upwind Scheme for Computing Supersonic Flows around a Tactical Missile", La Recherche Aérospatiale, English Edition, No 1988-2, pp.43-55, 1988.

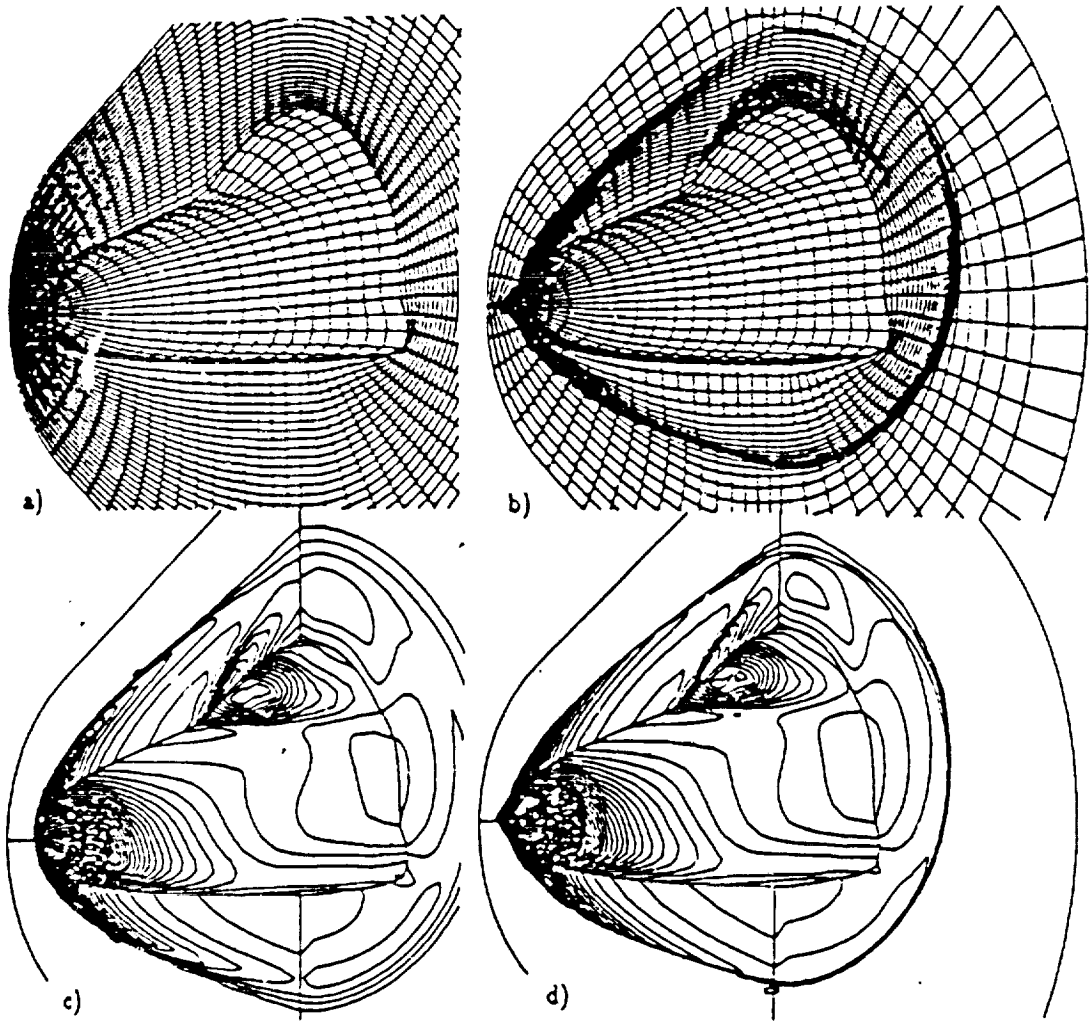


Fig. 4: Adaptation around a Re-Entry Vehicle