

**EULER CALCULATIONS BY
UPWIND FINITE ELEMENT METHODS AND
ADAPTIVE MESH ALGORITHMS**

**A. Dervieux, J.A. Désidéri and L. Fesoui, H. Stève
INRIA, Avenue Emile Hughes, Sophia Antipolis, 06565 VALBONNE,**

**B. Palmerio,
Université de Nice, Parc Valrose, 06034 NICE,**

**J. Périaux, J.P. Rosenblum, B. Stoufflet,
AMD-BA, 78 Quai Carnot, 92214 SAINT-CLOUD, FRANCE**

ABSTRACT

This paper presents Euler calculations employing several recent developments :

- a family of upwind finite-element schemes based on approximate Riemann solvers,
- mesh adaption procedures.

The approximation is a second-order accurate MUSCL-like scheme, that includes monotonicity-preserving limiters. Efficiency is achieved by an implicit formulation. The solution procedure is combined with a mesh enrichment algorithm by local element division, and a node movement algorithm more specifically adapted to accurately capture 1-D structures (shocks, layers,...).

INTRODUCTION

The 2-D algorithms employed in this contribution were designed in preparation of the calculation of 3-D flows around complete aircraft configurations. One way to reach this goal is to develop finite-element methods (FEM) capable of handling unstructured meshes. This approach proved its high reliability in structural mechanics, but it is still fairly new in the field of compressible CFD.

In our point of view, the construction of the mesh by the engineer should avoid excessive efforts, and hence the solution method should be robust enough to perform satisfactorily when fastly constructed meshes are employed. The method should apply to unstructured, irregular, even distorted (strongly non-orthogonal) meshes, and should be self-adapted to the computation. Consequently, our effort concentrates on "unstructured" FEM and mesh adaption.

Two research groups at INRIA and AMD-BA cooperated in the study of such schemes:

-Originally, a "Richtmyer-Galerkin" scheme was constructed and applied to 2-D and 3-D flow problems [1,2]. This scheme is presented in details in an AMD-BA publication [18].

-In the present contribution is presented a more recent method that combines an upwind FEM [4,5,6] with new mesh-adaption devices. The purpose of this paper is also to present an evaluation of the performance of a newly developed vectorized version [3] of the code in calculations where unstructured meshes are used.

1. UPWIND SPATIAL APPROXIMATION

Many known schemes are derived by writing at each node of the mesh a consistent spatial approximation of the Euler equations,

$$F(w)_x + G(w)_y = 0$$

that are a set of first-order PDE's. Such approximations are basically equivalent to a central-differencing approximation + additional terms, that are, essentially, diffusion terms. It is well known that this diffusion is necessary since the central difference cannot be used alone, and one important concern that occupied numerous CFD scientists and engineers has been to:

- introduce enough diffusion to realize a robust method and smooth results,
- avoid excessive amounts of diffusion that would result in severe accuracy degradation, and
- design the automatic control of the level of diffusion.

The solution to this problem is even more difficult in the context of:
-complex solutions,
-unstructured meshes,
-local mesh refinements.

The proposed upwind schemes have the following properties:

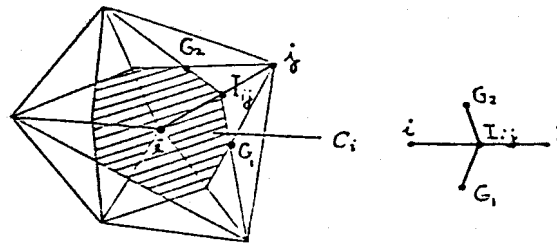
-they are fully conservative,
-they require the adjustment of no parameter,
-they perform well (smooth results) when applied to unstructured, even locally strongly refined meshes.

2. BASIC SCHEME

The construction of the basic scheme employs some of the ideas of B. van Leer [7] and was derived by F. Fezoui [4].

We consider a finite-element discretization using triangular elements; the nodes (supporting the values of all the unknowns) are the vertices of the triangles.

The steady Euler equations are written in conservative form, that is, $F(w)_x + G(w)_y = 0$, in which the vector w contains the usual variables $w = (\rho, \rho u, \rho v, E)$. An essentially equivalent finite-volume scheme is then constructed : cells are defined via the medians of the triangles; the discrete system is then obtained by summation of flux integrals over each bi-segment $G_1 I_{ij} G_2$ with one evaluation, at I_{ij} of the integrand (see sketch below). At each integration point I_{ij} we define two values w_{ij} and w_{ji} by interpolating the dependent variable w in each neighboring cell i and j .



2.1. Flux splitting and original first-order scheme

In the first-order accurate scheme, a constant-by-cell interpolation is employed :

$$w_{ij} = w_i \quad , \quad \text{and} \quad w_{ji} = w_j$$

Then flux-splitting is performed at I_{ij} . In the present calculations, we considered the following flux-splittings:

-the **STEGER** and **WARMING** splitting [11] :

$$\Phi_{ij} = P^+(w_{ij})w_{ij} + P^-(w_{ji})w_{ji}$$

-**VIJAYASUNDARAM's** simplified splitting [14] :

$$\Phi_{ij} = P^+ \left(\frac{w_{ij} + w_{ji}}{2} \right) w_{ij} + P^- \left(\frac{w_{ij} + w_{ji}}{2} \right) w_{ji}$$

where P is the derivative of an appropriate linear combination H of F and G ; P is diagonalizable, $P = T\Lambda T^{-1}$, $\Lambda = (\lambda_k)$, and $P^\pm = T(\lambda_k^\pm)T^{-1}$.

-the **Q-splitting** [7] ($|P| = T(|\lambda_k|)T^{-1}$) :

$$\Phi_{ij} = \frac{1}{2} (H(w_{ij}) + H(w_{ji})) + \frac{1}{2} \left| P \left(\frac{w_{ij} + w_{ji}}{2} \right) \right| (w_{ij} - w_{ji})$$

-**OSHER's** splitting [8] shortly defined by :

$$\Phi_{ij} = \frac{1}{2} (H(w_{ij}) + H(w_{ji})) + \frac{1}{2} \int_{w_{ji}}^{w_{ij}} |P(w)| dw$$

-**van LEER's** flux-vector splitting:

$$\Phi_{ij} = h^+(w_i) + h^-(w_j)$$

in which by considering a rotation of the axes, the vector $h(w) = \eta_x F(w) + \eta_y G(w)$ is split as in the one-dimensional case as detailed in [19].

The Q-splitting and Osher's splitting are employed in the (explicit) spatial approximation; Vijayasundaram's and the Steger-Warming splittings are used

in the construction of linearized implicit operators. The combination of the Q-splitting explicitly with Vijayasundaram's splitting implicitly is less computer-time consuming and suitable to transonic calculations, while Osher's splitting explicitly with the Steger-Warming splitting implicitly result in a more robust scheme that is employed to compute the supersonic cases.

The van Leer flux-vector splitting was employed to compute the blunt-body problem. This splitting is comparable to Osher's splitting since it continuously differentiable and very robust [20].

2.2. Second-order accurate extension [4]

Following [7], an elegant way to reach second-order accuracy is to derive the values w_{ij} and w_{ji} from linear interpolations. The dependent variable w is linearly interpolated on each cell C_i around a vertex i ; this interpolation is computed from the node value w_i and from an approximate gradient (\bar{w}_x, \bar{w}_y) obtained from the Lagrange interpolant w_L as follows :

$$\text{area}(C_i) \bar{w}_x(i) = \int_{C_i} \frac{\partial w_L}{\partial x} dx dy$$

Then the two values w_{ij} , w_{ji} are defined at the mid-point I_{ij} as follows :

$$w_{ij} = w_i + \frac{1}{2} \begin{vmatrix} \bar{w}_x(i) \\ \bar{w}_y(i) \end{vmatrix} \cdot \vec{i j}$$

$$w_{ji} = w_j + \frac{1}{2} \begin{vmatrix} \bar{w}_x(j) \\ \bar{w}_y(j) \end{vmatrix} \cdot \vec{j i}$$

Actually, in the present work, this interpolation has not been applied to the usual conservative variables but instead to the so-called physical/primitive variables (density, velocity components, pressure), following van Leer [7]. However, the (conservative) corrector step has been expressed in terms of the conservative variables.

Note that this construction results in a FROMM-like half-upwind scheme. For the blunt-body calculation, we also considered a half-upwind scheme, but the construction differed slightly. We averaged the central-differencing scheme obtained by letting,

$$w_{ij} = w_i + \frac{1}{2}(w_j - w_i)$$

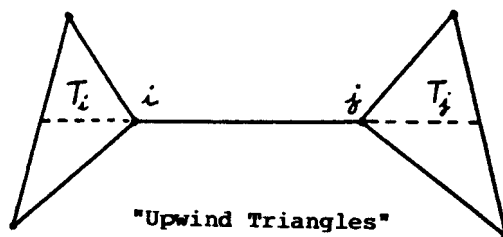
$$w_{ji} = w_j - \frac{1}{2}(w_j - w_i)$$

with the fully upwind scheme defined by,

$$w_{ij} = w_i + \frac{1}{2} \overrightarrow{\nabla w}(T_i) \cdot \overrightarrow{ij}$$

$$w_{ji} = w_j - \frac{1}{2} \overrightarrow{\nabla w}(T_j) \cdot \overrightarrow{ij}$$

in which the gradients are evaluated by the P_1 -Lagrange interpolation in the "upwind triangles" T_i and T_j . (Of all triangles whose vertices include node i , T_i is the one whose intersection with the line ij is a segment not contained by the segment ij itself. The triangle T_j is defined in a similar way with respect to node j . See sketch below.)



3. TIME INTEGRATION

Pseudo-unsteady methods are used :

3.1. *Explicit scheme*

A cheap explicit scheme is obtained [4], following an idea by Hancock [7], by combining in a two-step formulation a predictor step that applies a linearized central differencing scheme with a corrector step based on the spatial discretization defined previously.

3.2. *Implicit scheme*

The finite-element implicit scheme was originally derived to be combined

with a Galerkin approximation [12] and subsequently extended to upwind schemes [7,8].

Basically, the scheme is derived from two spatial approximations of the stationary Euler system:

-a second-order accurate approximation : $w \rightarrow E_2(w)$ constructed as described in Section 2.2,

-a first-order accurate linearization : $(w, \delta w) \rightarrow E'_1(w) \cdot \delta w$ constructed via the linearization involved in the flux splitting : for example, when the STEGER-WARMING splitting is chosen, the operator E'_1 is obtained by summation of fluxes computed from :

$$P^+(w_i)\delta w_i + P^-(w_j)\delta w_j$$

In this case, note that constant-by-cell values of w are used in the flux splitting (first-order accuracy).

Then the global scheme for one time-step can be written :

A. Explicit phase:

$$\delta \hat{w} = \Delta t E_2(w^n) \quad (1)$$

B. Implicit phase: Solve

$$(Id - \Delta t E'_1(w^n)) \delta w^{n+1} = \delta \hat{w} \quad (2a)$$

and update

$$w^{n+1} = w^n + \delta w^{n+1} \quad (2b)$$

To solve (2a), a 4×4 block Jacobi nodewise iteration is applied ; 10-20 sweeps revealed to be a good strategy from both stability and efficiency stand-points. Furthermore the matrix $E'_1(w_n)$ is not recomputed at each time step, but is frozen over several time-steps, usually 10. A last (significant) improvement is obtained by using local time-stepping. Typically, the CFL number in the final phase of the convergence is of the order of a few hundreds. Finally, different flux splittings are used in the two phases ; thus the implicit phase only acts as a preconditioner of the problem ; it does not require the solution of the exactly linearized system.

4. MONOTONICITY DEVICES

Oscillation-free solutions are obtained by introducing a "slope-limiter" in the explicit phase.

For Case 1.2 and Case 1.3 van Leer-type slope limiters were used.

For Case 1.4 a first-order algorithm was applied.

For Case 1.5, a newly derived upwind TVD scheme [13] was employed.

For Case 2.3, Case 1.2, and for the blunt-body calculations, van Leer-type slope limiters were used.

5. ADAPTIVE MESH ENRICHMENT [9]

Since the solution method is capable of handling unstructured meshes, a coarse mesh used in a preliminary calculation can be enriched a posteriori and the new mesh directly utilized to perform a new (and more accurate) calculation with the *same solver*, with no modifications to the computer program.

Mesh enrichment is performed by dividing selected triangles in 4; are divided those triangles where the measure of an appropriate criterion (evaluated using the preliminary solution) exceeds a given tolerance. The tolerance can be adjusted to control the total number of nodes of the new mesh. The criterion is an approximation of the quantity $\vec{U} \cdot \vec{\nabla} S$ where \vec{U} is the material velocity and S the entropy. With this definition, the criterion will be small in regions where the solution is regular provided the scheme is accurate enough.

For Cases 1.2-1.3, the initial mesh is radial and contains 64×17 (=1088) nodes. This mesh which extends to 20 diameters away from the center is rather coarse.

For Cases 1.4-1.5, a topologically similar mesh extending to only 2.5 diameters was used.

The enriched mesh that resulted contained:

Case 1.2: 1992 nodes

Case 1.3: 1992 nodes

Case 1.4: 2448 nodes

Case 1.5: 2431 nodes

Case 2.3: 2514 nodes

We stress that in general, the complete calculation (solution on the coarse mesh + solution on the fine mesh when the preliminary solution is initially interpolated to the fine mesh) requires less computing-time than a single calculation on the fine mesh when a trivial initialization is made (uniform flow).

6. MESH MOVEMENT

In a final phase, adaptive mesh movement [10] can be applied and allowed to interact with the Euler solver (Cases 1.4-1.5). This results in a better capture of the shock. To realize this, 2000 explicit additional time iterations were made and 10 mesh movements were computed. Overall, the computing time was doubled.

7. VECTORIZATION

Most of the calculations presented here were performed on a CII HB68 with a non-optimized "research" computer program, and sometimes many successive one-hour long runs were necessary. Whence, to obtain a pertinent information on the efficiency of our numerical methods, one case (Case 1.1) was recomputed using a still partially vectorized version of the implicit scheme newly derived by F. Angrand and J. Erhel [3]. The vectorization of codes is a more complex task in the FEM context than it is in the FDM or FVM context, and to date, the vectorization of our programs is not yet complete. It is realized by first separating the computations in two parts:

- (1) gather and scatter operations,
- (2) the actual calculation (arithmetic operations).

Part (1) is easily vectorized. Part (2) is partially vectorized for runs on the new generation of supercomputers (CRAY XMP, CRAY 2) using coloring methods. Nevertheless, the computations presented here were made with a CRAY 1-S, and average rates of about 25-30 Mflops were attained. About 10 sec of CPU time were necessary in a calculation using a mesh of 1088 nodes, the residual being reduced by a factor of 10^4 .

8. CONCLUSION

This work resulted in the development of a robust finite-element upwind scheme. The method proved to combine well with mesh adaption procedures and in particular to be very powerful for the accurate resolution of strong shocks. Efficiency was achieved by the vectorization of the (linearized) implicit scheme.

Many further improvements were made since in the area of efficiency of implicit schemes [15] and multigrid schemes [16]. The upwind formulation has also served to initiate a research in the field of hypersonic reactive flow simulation [17].

ACKNOWLEDGEMENT

The access to CRAY-1S was provided by the C.C.V.R. Scientific Committee.

REFERENCES

- [1] F. ANGRAND, V. BILLEY, A. DERVIEUX, J. PERIAUX, B. STOUFFLET, 2-D and 3-D Euler flow calculations with a second-order Galerkin Finite Element method, AIAA Paper 85-1706 (1985).
- [2] F. ANGRAND, A. DERVIEUX, Some explicit triangular finite element schemes for the Euler equations, Int. J. For Num. Methods In Fluids, Vol. 4, 749-764 (1984).
- [3] F. ANGRAND, J. ERHEL, Vectorized Finite Element codes for compressible flows, Sixth Int. Symp. on Finite Element in Flow Problems, Antibes (France) June 16-20, 1986 (to be published by Wiley and Sons).
- [4] F. FEZOUI, Résolution des équations d'Euler par un schéma de van Leer en éléments finis, INRIA Report 358 (1985).
- [5] F. FEZOUI, B. STOUFFLET, A class of implicit upwind schemes for Euler simulations with unstructured meshes, INRIA Report 517, (1986).
- [6] F. FEZOUI, B. STOUFFLET, J. PERIAUX, A. DERVIEUX, Implicit high-order upwind Finite-Element schemes for the Euler equations, Fourth Int. Symposium on Numerical Methods in Engineering, Atlanta (GA, USA), March 24-28, 1986, to be published by Computational Mechanics Pub., Southampton (UK).
- [7] B. van LEER, Computational methods for ideal compressible flow, von Karman Institute for Fluid Dynamics, Lecture Series 1983-04.
- [8] S. OSHER, S. CHAKRAVARTHY, Upwind schemes and boundary conditions with applications to Euler equations in general geometries, J. of Comp. Physics, Vol. 50, 3, 447-81 (1983).
- [9] B. PALMERIO, V. BILLEY, A. DERVIEUX, J. PERIAUX, Self-adaptive refinements and F.E.M. for solving the Euler equations, Int. Conf. on Num. Meth. for Fluid Dyn. April 1-4, 1985, Reading (UK), to appear.
- [10] B. PALMERIO, A. DERVIEUX, Application of a FEM moving node adaptive method to accurate shock capturing, The First Int. Conf. on Numerical grid generation in Comp. Fluid Dyn., Landshut (Germany), July 14-17, 1986.

- [11] J. STEGER, R.F. WARMING, Flux vector splitting for the inviscid gas dynamic equations with application to finite difference methods, *J. of Comp. Physics*, Vol.40, 2, 263-293 (1981).
- [12] B. STOUFFLET, Implicit Finite Element Methods for the Euler Equations, in "Numerical Methods for the Euler Equations of Fluid Dynamics", F. Angrand et al. Eds., SIAM, Philadelphia (1985).
- [13] B. STOUFFLET, J. PERIAUX, F. FEZOU, A. DERVIEUX, Numerical Simulation of 3-D hypersonic Euler flows around space vehicles using adapted finite element, AIAA Paper 87-0560.
- [14] G. VIJAYASUNDARAM, Transonic flow simulations using an upstream-centered scheme of Godunov in Finite Elements, *J. of Comp. Physics*, Vol. 63, 416-433 (1986).
- [15] H. STEVE, Méthodes efficaces pour la résolution des équations d'Euler en éléments finis, INRIA report, to appear,
- [16] M.-H. LALLEMAND, A. DERVIEUX, A Multigrid Finite Element Method For Solving the Two-Dimensional Euler Equations, INRIA report, to appear.
- [17] J.-A. DESIDERI, E. HETTENA, Numerical Simulation of Hypersonic Equilibrium-Air Reactive Flow, Euromech 225 The Aerodynamics of Spacecraft, 12th-16th July, 1987, Cranfield, UK, also INRIA Report, to appear.
- [18] F. ANGRAND, V. BILLEY, J. PERIAUX, J. P. ROSENBLUM, B. STOUFFLET, Euler Calculations by a Galerkin Finite Element Approximation using Implicit Methods, GAMM Workshop on the "Numerical Simulation of Compressible Euler Flows", Rocquencourt, June 10-13, 1986, to be published by Vieweg.
- [19] B. van LEER, Flux-Vector Splitting for the Euler Equations, Lecture Notes in Physics, vol. 170, Aachen 1982, pp. 505-512.
- [20] L. FEZOU, H. STEVE, Décomposition de flux de van Leer en Eléments Finis, INRIA Report, to appear.

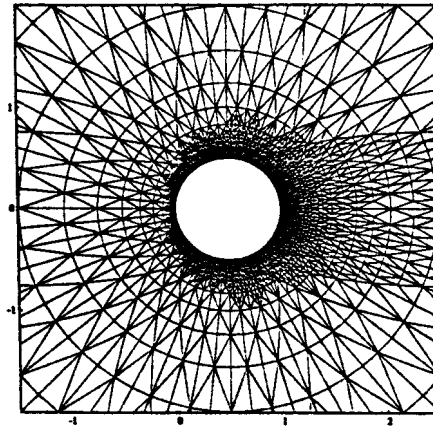
2-D EXTERNAL FLOW AROUND A
CIRCULAR CYLINDER
- INRIA -

Test Case 1.2 - MACH = 0.50

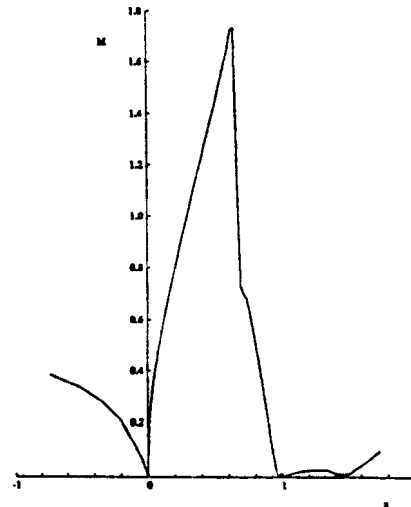
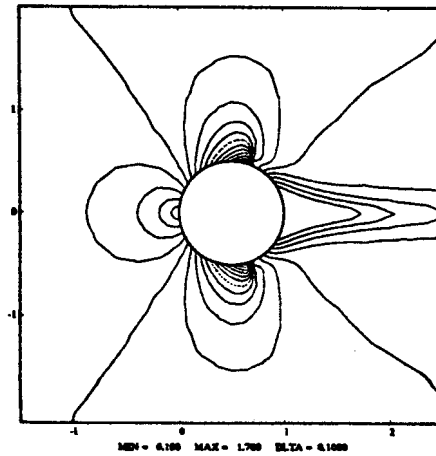
2-Step Explicit 2nd-order Upwind Scheme
($\beta = 1/2$)

Slope limiters - Local timestep
Enriched mesh (1992 nodes).

ENLARGEMENT OF THE MESH NEAR THE CYLINDER



MACH-LINES

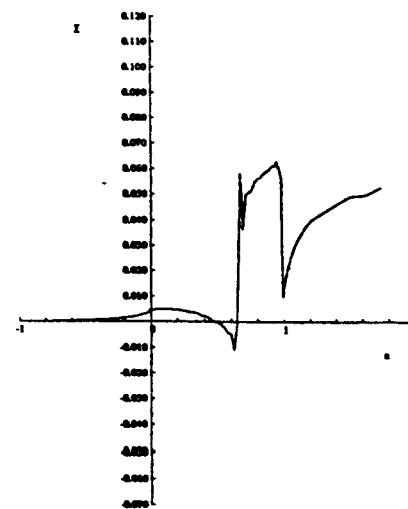
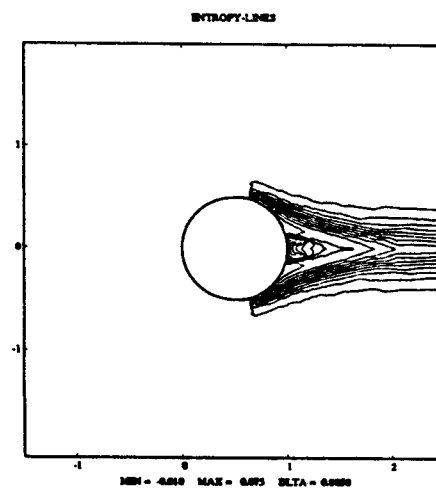
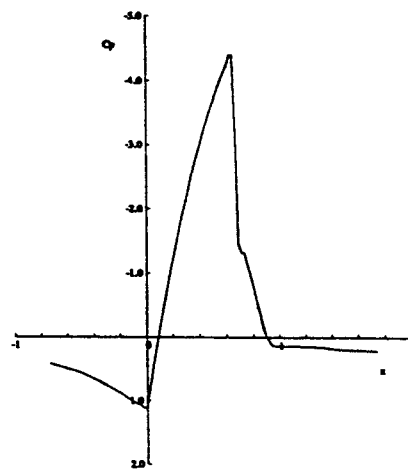
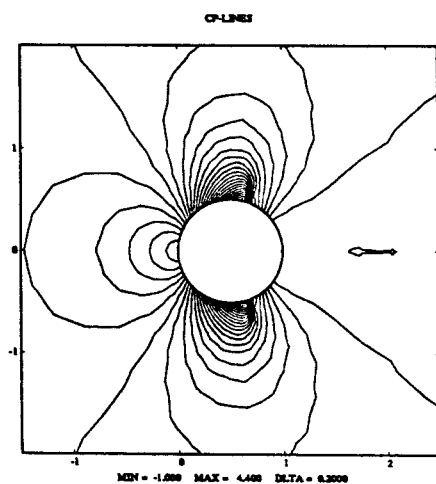


2-D EXTERNAL FLOW AROUND A
CIRCULAR CYLINDER
- INRIA -

Test Case 1.2 - MACH = 0.50

2-Step Explicit 2nd-Order Upwind Scheme
(Beta = 1/2)

Slope Limiters - Local Timestep
Enriched Mesh (1992 nodes)

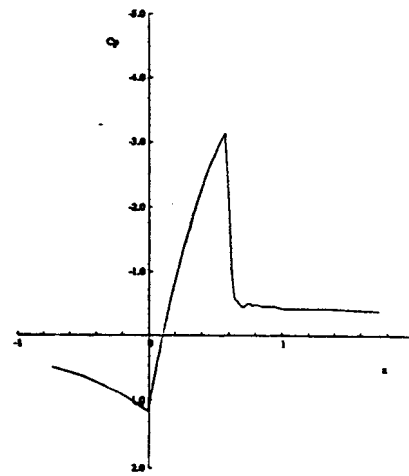
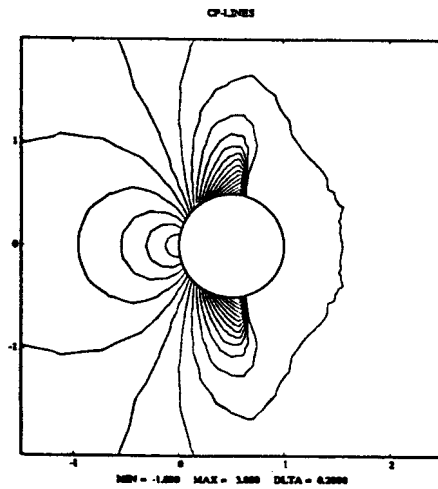
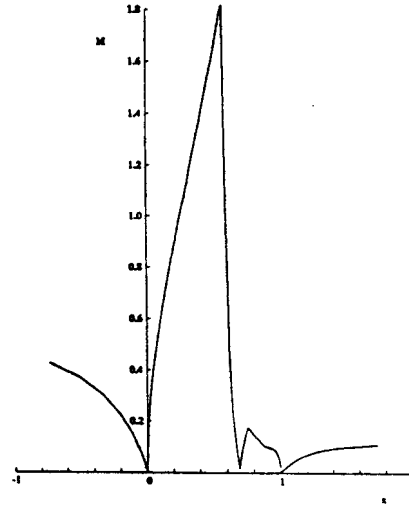
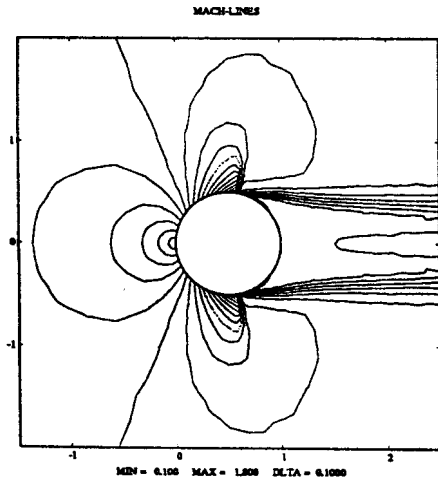


2-D EXTERNAL FLOW AROUND A
CIRCULAR CYLINDER
- INRIA -

Test Case 1.3 - MACH = 0.60

2-Step Explicit 2nd-Order Upwind Scheme
(Beta = 1/2)

Slope Limiters - Constant Timestep
Enriched Mesh (1992 nodes)

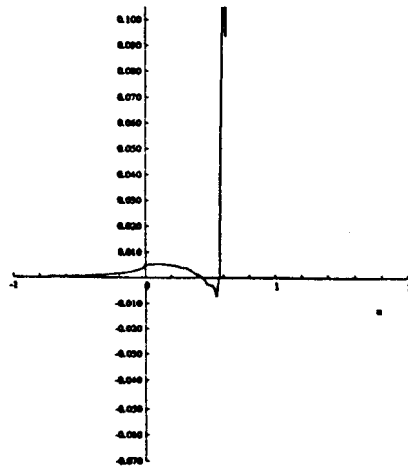
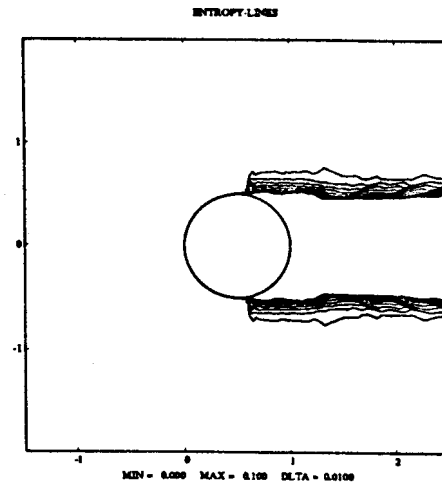
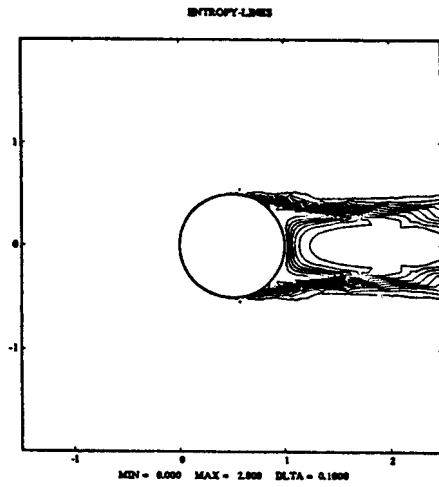


2-D EXTERNAL FLOW AROUND A
CIRCULAR CYLINDER
- INRIA -

Test Case 1.3 - MACH = 0.60

2-Step Explicit 2nd-Order Upwind Scheme
(Beta = 1/2)

Slope Limiters - Constant Timestep
Enriched Mesh (1992 nodes)



2-D EXTERNAL FLOW AROUND A
CIRCULAR CYLINDER
- INRIA -

Test Case 1.4 - MACH = 3.0

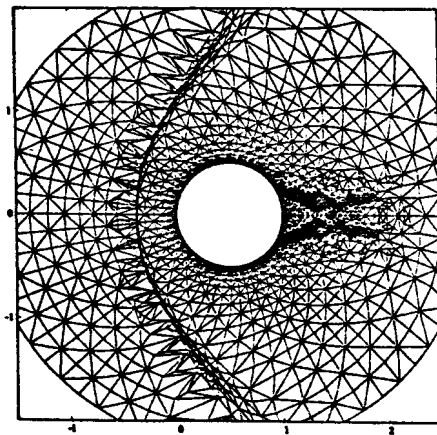
First-Order Upwind Scheme

Osher's Flux Splitting

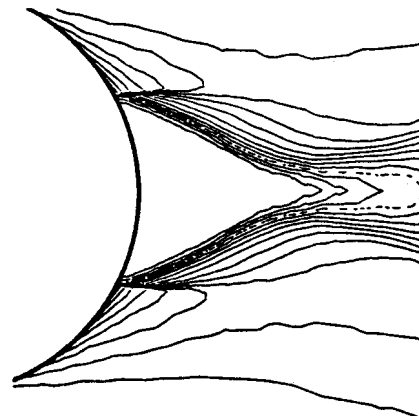
Euler Explicit Method (local timestep)

Mesh Enrichment and Movement (2448 pts)

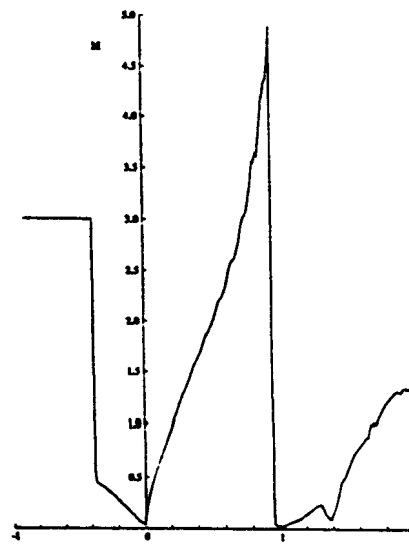
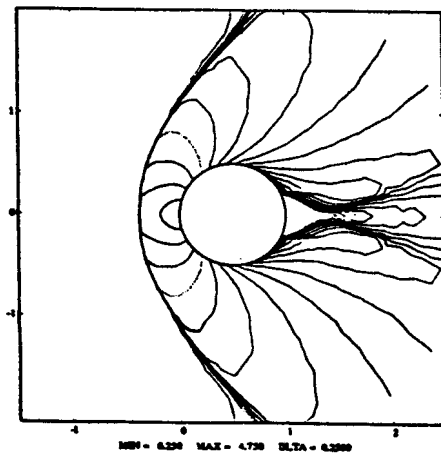
ENLARGEMENT OF THE MESH NEAR THE CYLINDER



MACH LINES AT THE REAR OF THE CYLINDER



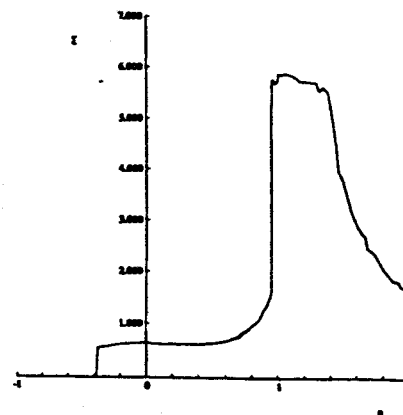
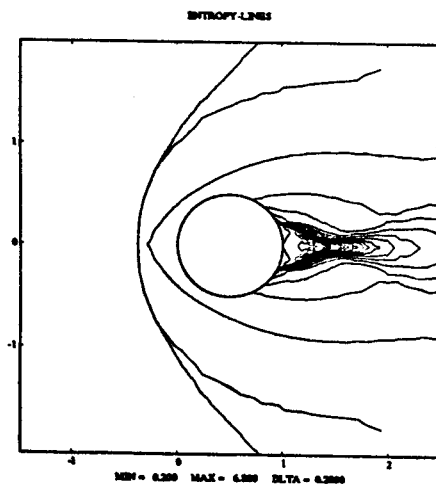
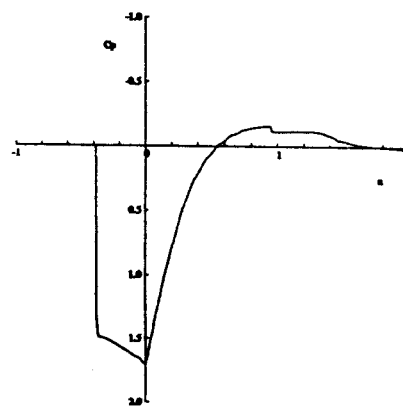
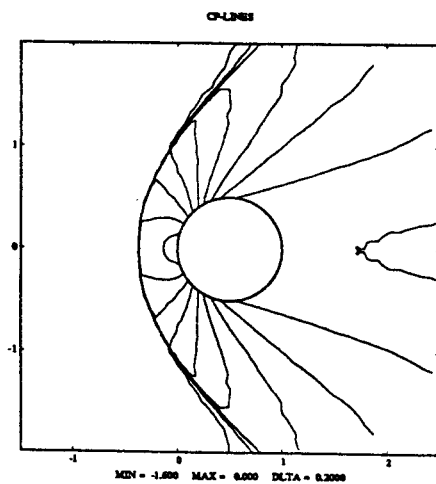
MACH LINES



2-D EXTERNAL FLOW AROUND A
CIRCULAR CYLINDER
- INRIA -

Test Case 1.4 - MACH = 3.0

First-Order Upwind Scheme
Osher's Flux Splitting
Euler Explicit Method (local timestep)
Mesh Enrichment and Movement (2448 pts)



2-D EXTERNAL FLOW AROUND A
CIRCULAR CYLINDER

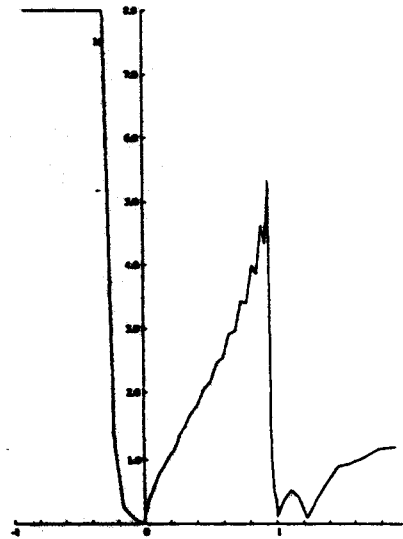
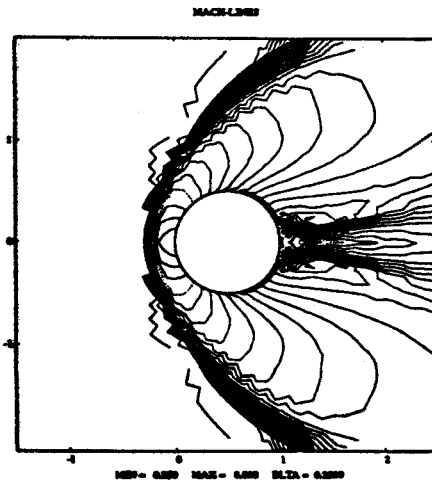
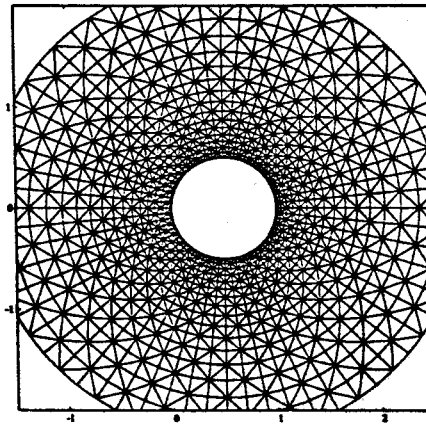
- INRIA -

Test Case 1.5 - MACH = 8.0

2-Step Explicit 2nd-Order Scheme
(Beta = 1/2)

Slope Limiters - Local Timestep
Radial Mesh (1088 nodes)

ENLARGEMENT OF THE MESH NEAR THE CYLINDER

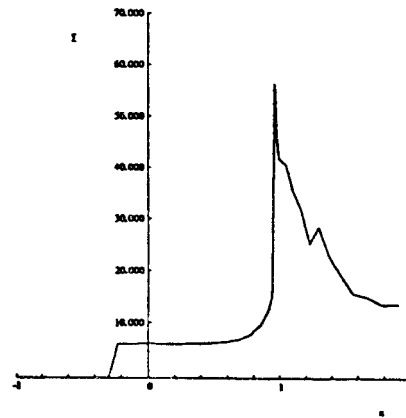
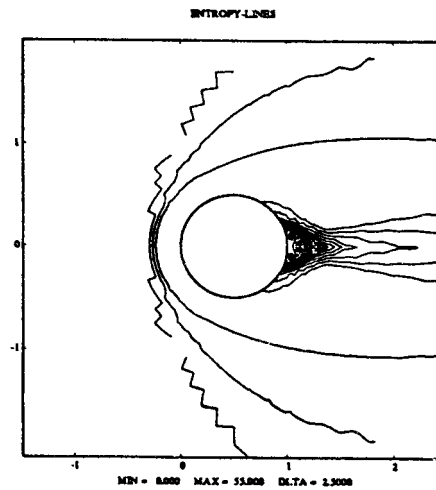
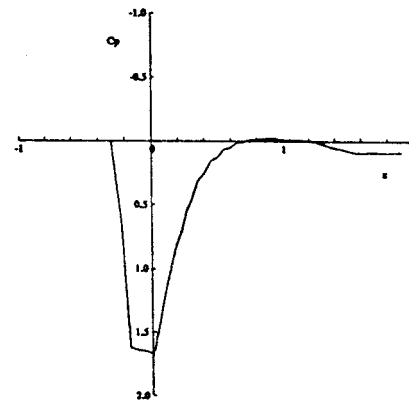
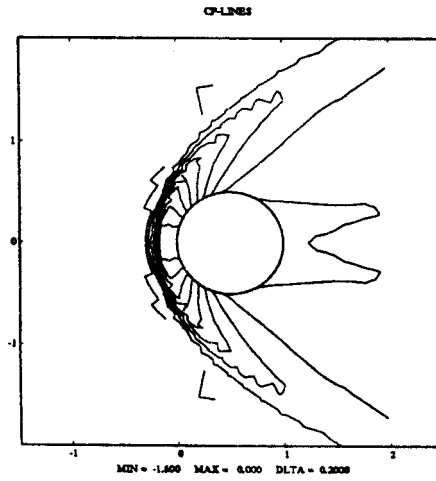


2-D EXTERNAL FLOW AROUND A
CIRCULAR CYLINDER
- INRIA -

Test Case 1.5 - MACH = 8.0

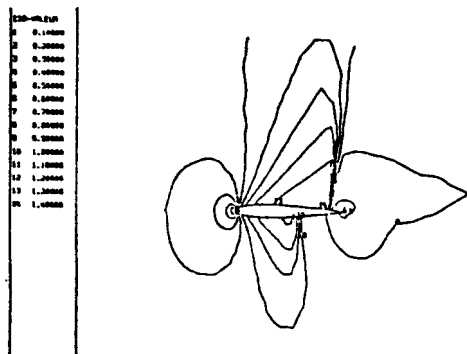
2-Step Explicit 2nd-Order Scheme
(Beta = 1/2)

Slope Limiters - Local Timestep
Radial Mesh (1088 nodes)

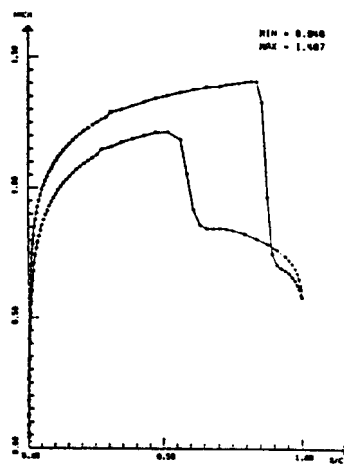


Case 2.3 NACA012 LIFTING AIRFOIL
($M_\infty = 0.85$ - $\alpha = 1^\circ$)

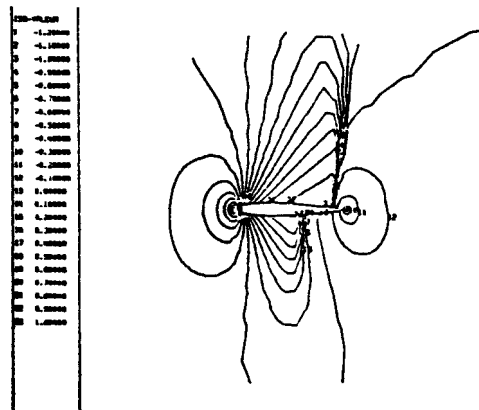
Mach lines



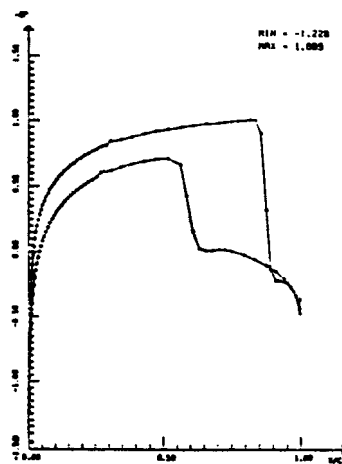
Mach plot



C_p lines

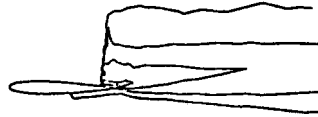


C_p plot

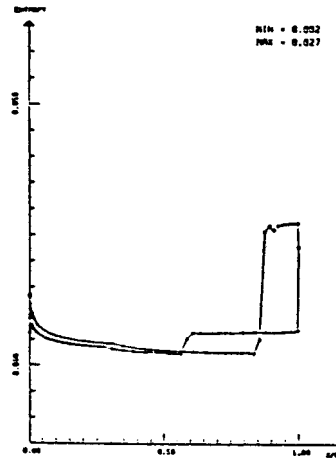


Case 2.3 NACA012 LIFTING AIRFOIL
($M_\infty = 0.85$ - $\alpha = 1^\circ$)

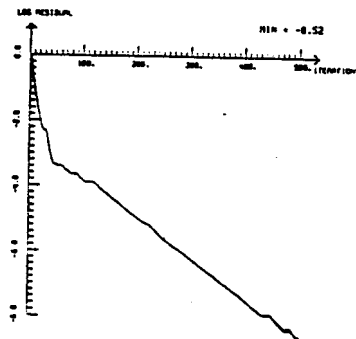
Entropy lines



Entropy plot



Convergence history



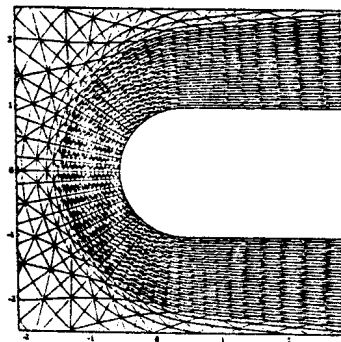
BLUNT-BODY FLOW PROBLEM

$M = 8$

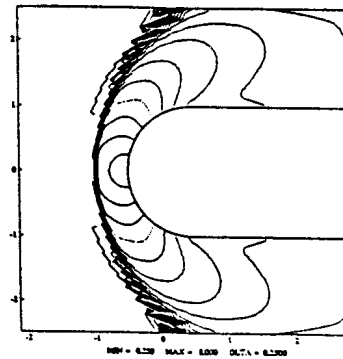
Euler explicit time-integration scheme.
Second-order half-upwind spatial approximation.
Van Leer's flux-vector splitting.
Slope limiters

(a) Zero incidence

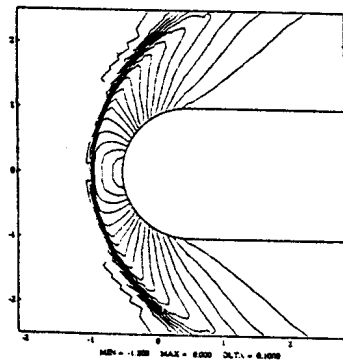
Mesh



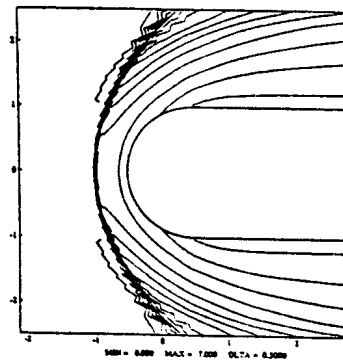
Mach lines



Pressure coefficient

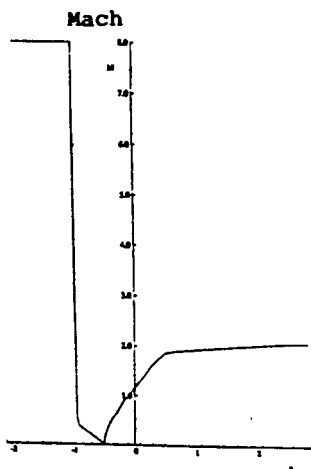


Entropy deviation

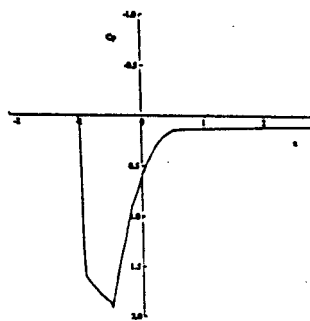


Case (a) continued :

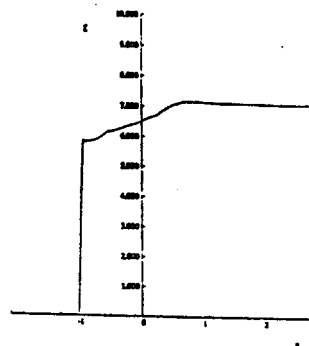
Plots :



Pressure coefficient



Entropy deviation



(b) Incident Case ($\alpha = 30^\circ$)

Mach lines

