

N96-18086

515-34
7337
7-14

AUTO-ADAPTIVE FINITE ELEMENT MESHES

Roland RICHTER * and Pénélope LEYLAND **

* *CRAY Research (Switzerland) S.A., Scientific Parc (PSE)*** *Institut de Machines Hydrauliques et de Mécanique des Fluides
Ecole Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland*

Abstract

Accurate capturing of discontinuities within compressible flow computations is achieved by coupling a suitable solver with an automatic adaptive mesh algorithm for unstructured triangular meshes. The mesh adaptation procedures developed rely on non-hierarchical dynamical local refinement/derefinement techniques, which hence enable structural optimisation as well as geometrical optimisation. The methods described are applied for a number of the ICASE test cases are particularly interesting for unsteady flow simulations.

INTRODUCTION

One of the most important advantages of finite element type unstructured grids is the possibility to refine/derefine locally the mesh during the computation. Successive mesh concentration in "critical" zones may be performed, without knowing them *a priori* during the initial mesh creation, as well as coarsening in regions where the nodes seem superfluous. The overall reduction of the number of nodes gives an optimal relation between precision and calculation cost (CPU time and memory constraint), by tracking the strong physical gradients within the flow field by higher grid point concentrations. This is especially important for unsteady flows.

In this paper, inviscid compressible flow calculations from the ICASE Workshop on adapted grids are performed using dynamically auto-adaptive finite element triangular meshes. The mesh optimisation algorithms as well as the adaptation procedures are completely non-hierarchical, which allows more freedom for imposing optimisation strategies for obtaining admissible, regular grids, which can be geometrical, or structural and physical. The algorithm of the dynamical refinement/derefinement procedure is based on a certain number of basic algorithmic principles taking into consideration the particularities of local mesh refinement for finite element type generated meshes. A new anti-data structure has been adopted, where the successive subdivisions are performed independently of the former operations.

LOCAL MESH REFINEMENT

As the goal of mesh adaptation is to increase the accuracy of the solution process by locally enforcing the *h*-adaptivity by usage of smaller discretisation cells, the first step in a mesh adaptation algorithm is to locally refine the mesh by adding new nodes according to some criteria, thus diminishing local size of the concerned elements. The criteria used should be as close as possible to the error estimations of the underlying discretisation scheme. The principle of adaptive meshing is to uniformly equidistribute the error enhancing the overall convergence.

Error estimates coming from the theory of finite-element simplex type meshes are based upon either *a priori* error estimates coming the governing equations (see e.g. [3]); those based on an *a*

PRECEDING PAGE BLANK NOT FILMED

219

PAGE 28 INTENTIONALLY BLANK

posteriori error estimate where the computed residual of the solution $R(u_h^n)$ is used to define the error [4, 6]; and those which evaluate a combination of the derivatives of the computed variables (e.g. [7]). There is in fact a close relationship between such methods, based on physical gradients, and those of the second kind, [6], at least for hyperbolic transport equations. For the mesh adaptation procedures developed here, the latter strategy has proven to be robust and precise for tracking discontinuities for problems presenting strong and weak shocks, contact discontinuities and so on. For steady state flow resolution, local error criteria are based upon combinations of the L^2 average of $\nabla_h Mach$, $\nabla_h Density$ and $\nabla_h Entropy$, with respect to a certain tolerance; whereas for unsteady flows, these are augmented by combinations of $\Delta_h Mach$, $\Delta_h Density$ and $\Delta_h Entropy$.

We describe here the techniques of mesh adaptation employed in the algorithm, further details can be found in [8, 9].

Symmetrical subdivision of triangles is obtained by adding a node in the middle of each side, rather than in the centre of each triangle, as only this method increases the precision of the numerical method, Figure 1. Indeed, from finite element theory, the interpolation is optimal when the triangle is as equilateral as possible within a certain metric.

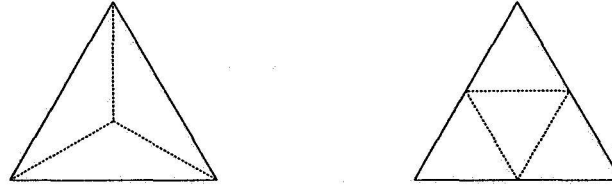


Figure 1: Symmetrical triangle subdivision.

The local mesh refinement algorithm consists of testing the segments of an existing mesh whether they are to be split or not, which means that either 1, 2 or 3 new nodes will be created per corresponding marked triangle. For geometrical considerations of maintaining non-acute angles, the creation of 2 new nodes is replaced by a symmetrical subdivision (Figure 1-right). The procedure performs refinement of triangles by subdividing them into 4 or 2 new elements. Division by two leads to zones of elements which may have a non-optimal distribution of the number of neighbours, leading to irregularities within the mesh. The close link between admissibility, regularity and optimality of an unstructured mesh to inherent properties of node neighbour numbers, has been fully exploited in the algorithms developed here, producing new constraints which are often more rigorous and less ad-hoc than existing ones where the adaptation criterion and its tolerance are the only reference points. For a 2D triangular mesh the optimal number of neighbours is 6. Typically, the two following refinement configurations should be avoided - Figure 2, and next section.

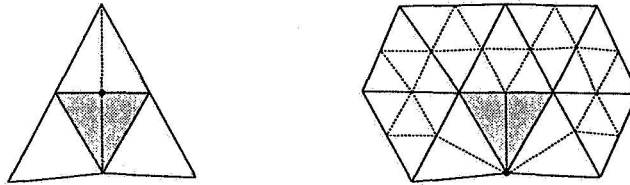


Figure 2: Creation of nodes, with a non optimal node neighbour number.

Their elimination is obtained by requiring that the the number of nodes that can be added within the 3 neighbouring elements should be greater than 2 and smaller than 5. This needs to be completed

by a geometrical criterion on the length relation between the smallest edge and the others in order to avoid creating highly stretched cells, (see next section).

GEOMETRICAL CRITERIA OPTIMISATION

After a refinement sweep, the new grid still “remembers” the old grid structure. To minimise such dependencies it is necessary to smooth globally the mesh as shown in Figure 3, using for instance the standard spring analogy :

$$\tilde{x}_i = \frac{\sum_{j \in k(i)} \alpha_j x_j}{\sum_{j \in k(i)} \alpha_j}$$

$\alpha_j = 1$ and $k(i)$ denotes the nodes neighbouring node i . The resulting mesh shows two specific behaviours : nodes having more than 6 neighbours tend to repulse their neighbours and those having less than 6 tend to attract them.

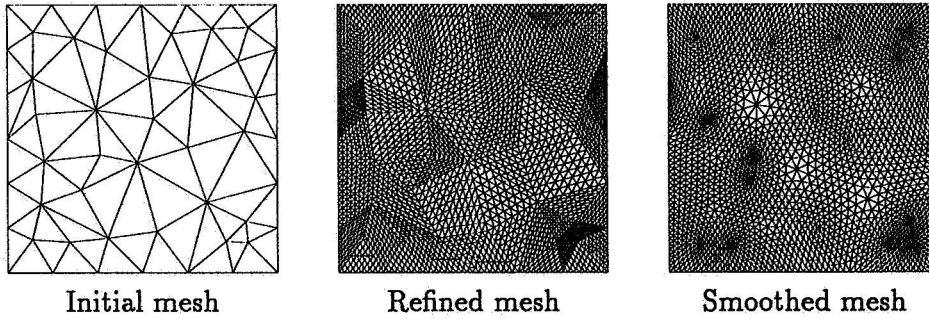


Figure 3: Geometrical smoothing procedure.

In order to reduce the above magnetic effect, a weight function related to the number of neighbours is introduced :

$$\alpha_j = \max[1, 6 + \beta (\mathcal{N}_j - 6)]$$

with

$$0 < \beta < 4$$

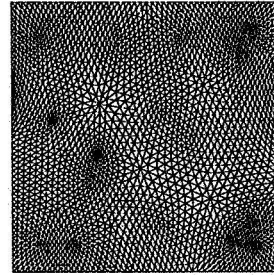


Figure 4: Smoothing with a weight function.

where \mathcal{N}_j denotes the number of neighbours of node j . The result is illustrated in Figure 4.

This optimisation could be more effective if nodes having more than 7 or less than 5 neighbours could be eliminated from the mesh. Such a requirement can be performed, by using the procedure of diagonal swapping with a neighbour node number minimising criterion. Let us denote by $\mathcal{N}_1, \mathcal{N}_2$ the number of neighbours for the vertices of a segment and $\mathcal{N}_3, \mathcal{N}_4$ these numbers after swapping the segment as shown in Figure 5. Diagonal swapping is accepted if either :

$$\mathcal{N}_3 + \mathcal{N}_4 < \mathcal{N}_1 + \mathcal{N}_2$$

or

$$\begin{cases} \mathcal{N}_3 + \mathcal{N}_4 = \mathcal{N}_1 + \mathcal{N}_2 \\ \max(\mathcal{N}_3, \mathcal{N}_4) < \max(\mathcal{N}_1, \mathcal{N}_2) \end{cases}$$

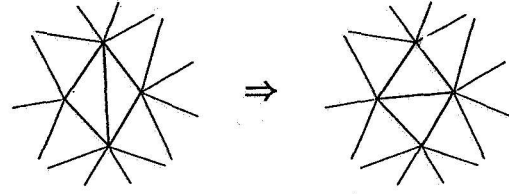


Figure 5: Diagonal swapping.

Then the procedure allows to limit, in most cases, the maximum number of neighbours to 7 and thus increase the number of nodes having an optimal number of neighbours.

To set the minimum number of neighbouring nodes, a technique of suppressing undesirable triangles can be used, which acts as shown in Figure 6.

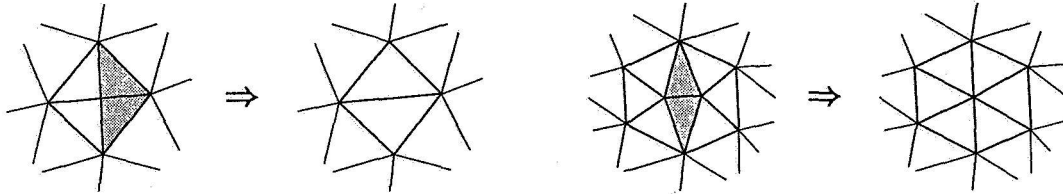


Figure 6: Cancellation of degenerate configurations.

The algorithm used consists in marking segments having a vertex with less than 5 neighbours or with exactly 5 on each side. In both cases, the specific segment and their associate triangles, the shaded elements of Figure 6, can be suppressed.

For the above square mesh, these techniques allow to regulate the optimal node neighbour number, and, combined with the weighted smoothing function, a considerable improvement in mesh regularity is achieved.

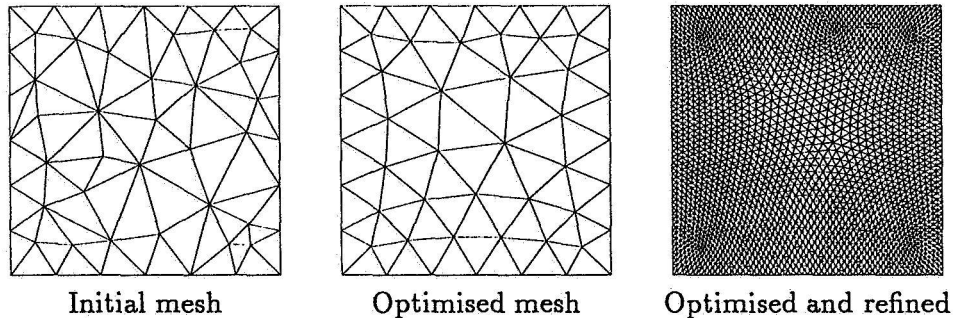


Figure 7: Geometrical optimisation procedure.

PHYSICAL CRITERIA OPTIMISATION

The techniques of structural changes via moving nodes and diagonal swapping can also be applied depending on physical quantities. An improvement of the accuracy of the capture of a discontinuity can be obtained by aligning the edges. In Figure 8 the orientation of the edges were originally normal to a shock, which produces, on the left hand side, a relative thick shock. On the right hand side, a procedure based on diagonal swapping has been used, minimising the angles between the discontinuity and edge.

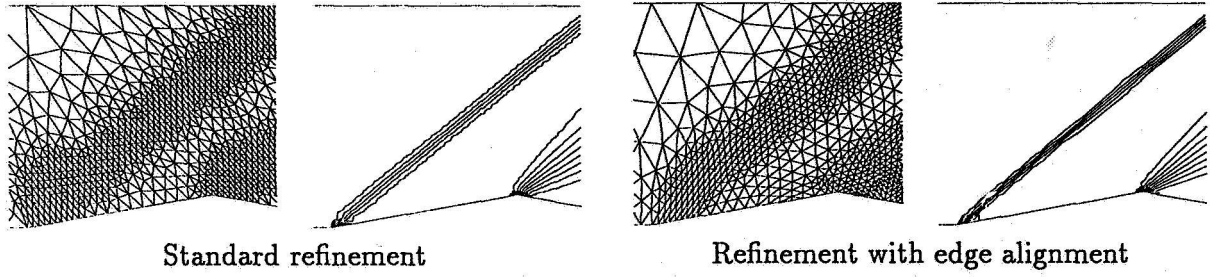
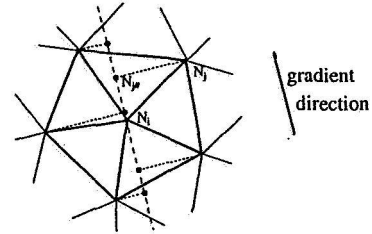


Figure 8: Improvement by aligning edges in the direction of the discontinuity.

A modified spring analogy using a weight function related to a physical quantity can also be applied to obtain a squeezing property in regions of strong gradients. The weighted spring analogy is increased by the projection of neighbouring nodes onto the direction of the local gradient, and by weighting this projection by the local physical difference. This part of the algorithm corresponds to:

$$\tilde{x}_i = \frac{\sum_{j \in k(i)} \alpha_j x_j + \beta_{ij} \left(x_i + \frac{\bar{x}_i x_j \cdot \nabla \rho_i}{\|\nabla \rho_i\|} \right)}{\sum_{j \in k(i)} \alpha_j + \beta_{ij}}$$



Projection of node neighbours onto a gradient's direction.

where α_j denotes the node number weight function, β_{ij} the local physical difference and $\nabla \rho_i / \|\nabla \rho_i\|$ the gradient direction of the physical quantity. Here the density has been chosen as the specific physical quantity.

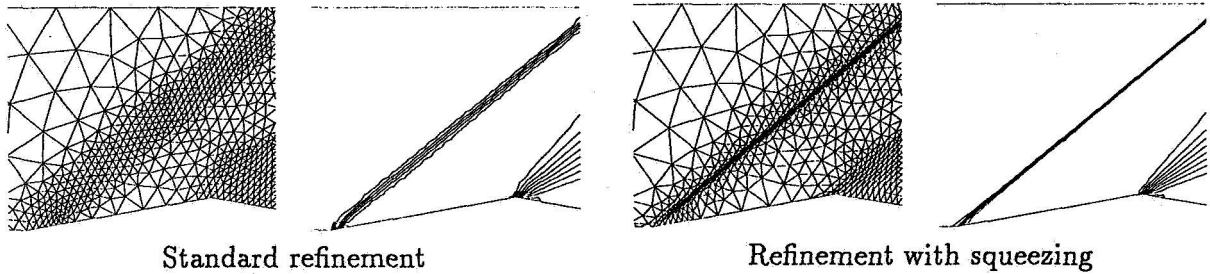


Figure 9: Improvement by local squeezing around discontinuities.

MESH DEREFINEMENT OR COARSENING

In order to optimise the ratio precision and accuracy versus minimal number of discretisation nodes, dynamically auto-adaptive remeshing via local refinement *and* local derefinement (or coarsening) is performed. Most local mesh derefinement techniques are based on a hierarchical data structure, which allows forward and backward scanning through the mesh, and maintains a history of previously added nodes. We have developed a new anti-hierarchical data structure; this algorithm enables nodes of an initial mesh to be removed, and also allows the use of the above structural optimisation techniques, which are incompatible with conventional hierarchical data structures.

This new algorithm is constructed as follows. The graph of points to be derefined is not straightforward, so the inverse problem is solved. A list of nodes, called fixed points, which are never to be derefined is identified. These are either the singular nodes defining, for instance, the corners of the domain, symmetry points..etc., or the nodes belonging to segments to be refined, or nodes having more than 6 neighbours. Then there are a number of possible fixed nodes allowing a maximal coarsening which come from either boundary or internal configurations. To find these, first the border of the domain is scanned so that each second node becomes a fixed point, and finally the configuration of Figure 10 is searched through the mesh. Thus all triangles containing no fixed point have at least three fixed neighbouring nodes, as shown in Figure 10.

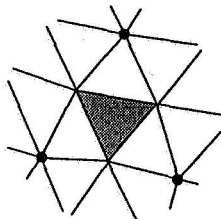


Figure 10: An element defined with 3 removable nodes should be surrounded by 3 fixed nodes.

Thus the grid becomes coloured according to the nodes to be removed and those to be kept in the mesh. The derefinition can then be done by performing successively the operation presented in Figure 11. It is necessary to treat firstly the element with 3 nodes to be invalidated, then the cases with 2 nodes to be removed, and finally the case having only one node which disappear. The dark shaded elements of Figure 11 vanish during the procedure and the light shaded triangles change their shape.

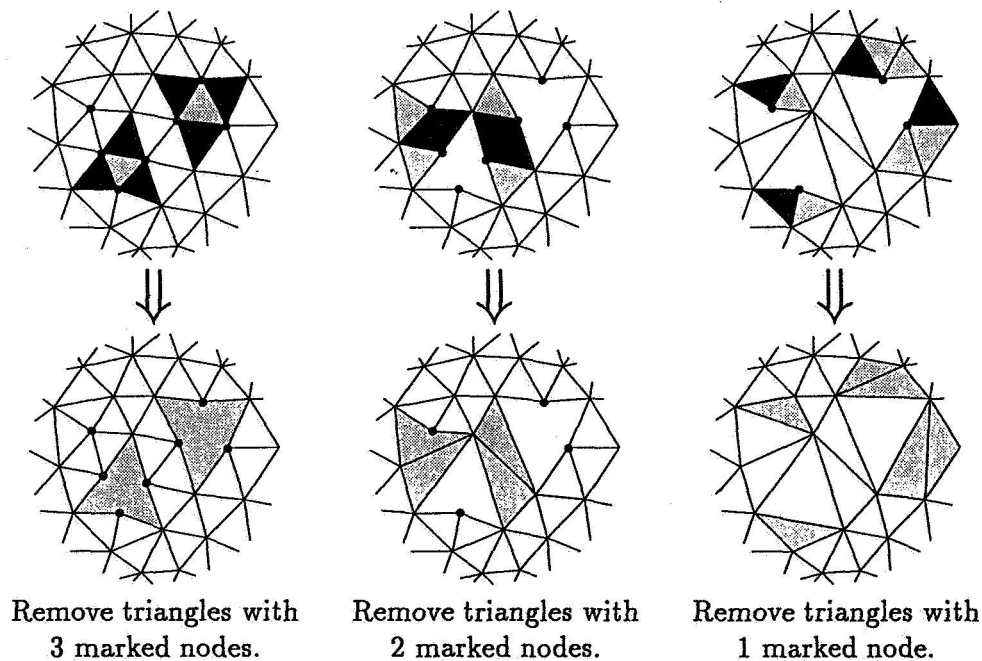


Figure 11: Derefinition stages.

ALGORITHM

The whole algorithm becomes thus : first, a list of nodes to be invalidated is defined, followed by their destruction, even if they belong to the initial macro-mesh. Then the phase of local enrichment (adding new nodes) is performed by adding them in the middle of segments. This is followed by both structural optimisation procedures, (first the physical diagonal swapping and then the geometrical ones), and finally the mesh is smoothed with the weighted spring analogy to obtain a regular, conformal and admissible mesh. For steady state simulations the squeezing procedure is performed after a specified convergence on the last two meshes. The total number of different meshes created depends upon the physical solution, for steady state calculations, between 4 to 8 different stages of remeshing are performed. The whole procedure is directly integrated into the flow solver giving a fully dynamic auto-adaptive mesh algorithm. For unsteady calculations, the remeshing depends greatly upon the speed of the transient states, and for a calculation cycle, there can be as many as 400 or more new meshes created. The increase of CPU time introduced by these procedures is the order of 2 to 5 % for steady state calculations, and is inferior to 30 % for unsteady ones. The internal accounting of memory requirements are managed by a dynamical memory manager and a archival data base structure; an updated mesh replaces the former one which requires only a minimal extra memory allocation.

APPLICATIONS FOR EXTERNAL FLOWS OVER PROFILES

The flow solver used for all these applications is based upon an equivalent Galerkin finite volume approximation on the dual control volumes of the P1 triangular simplex. A Jameson type centred scheme is employed with artificial dissipation, [9, 8].

Transonic flow over a NACA0012 at $M_\infty = 0.80$, $\alpha = 1.25^\circ$

This is the AGARD 01 test case [1], concerning transonic flow over a NACA profile. Its particularities are the presence of a weak shock on the windward side which is a good challenge for testing shock capturing criteria for local refinement and squeezing in this region, where the gradients are not as strong as those that can be observed, for instance, with another very standard test case, $M_\infty = 0.85$, $\alpha = 1^\circ$, (AGARD 02). The shock on the leeward side is a much sharper discontinuity, and thus is easier to localise. Since the test case concerns a non-zero angle of attack flow over a profile in a bounded domain, a non-zero circulation is induced on the profile, it is necessary to correct the outer boundary condition in order to simulate as well as possible an infinite domain [10]. The sensibility of the values of the lift coefficient to this induced circulation effect is high. The outer boundary should be chosen as far as possible, and be of a suitable dimension, as well as a "vorticity" correction which is applied to the infinite boundary conditions. For transonic flows, the infinite boundary values correspond to three entering characteristics and one outgoing one. The outgoing component can allow the generated circulation to influence the infinite boundary values of the solution.

Several different geometrical shapes for the domain boundaries were tested, as well as varying profile to infinite boundary lengths, as a function of the chord length. For a fixed outer boundary distance, the best results were obtained by a circular outer boundary. The aerodynamical coefficients are tabulated below, where a significant variation is found even by taking the precaution of applying the vorticity correction. The values stabilize after a distance of 1000 chord lengths, (up to 10000 chord lengths were tested). They correspond well to the references values established by AGARD.

The solution adaptive mesh procedures presented in this paper were applied, and despite the

Table 1: Flight coefficients for NACA0012 at $M_\infty = 0.80$, $\alpha = 1.25^\circ$

Outer boundary	C_l	C_d	C_m	X_{upper}	X_{lower}
10 chords	.3375	.0224	-.0351	.6274	.3441
30 chords	.3495	.0232	-.0371	.6327	.3422
100 chords	.3549	.0235	-.0382	.6353	.3382
1000 chords	.3569	.0237	-.0387	.6363	.3357

difficulties of capturing the weak windward side shock, a highly satisfactory adaptive mesh was obtained. For this test case and the next one, where shocks and expansion fans are present within the flow field, a combination of gradients and differences of physical variables were chosen to provide the adaptation criteria as explained in the previous sections. All six distinct meshes were generated during the process. The initial mesh was of 1704 nodes and 3332 elements, the final one of 8136 nodes and 16078 elements, Figures 12.

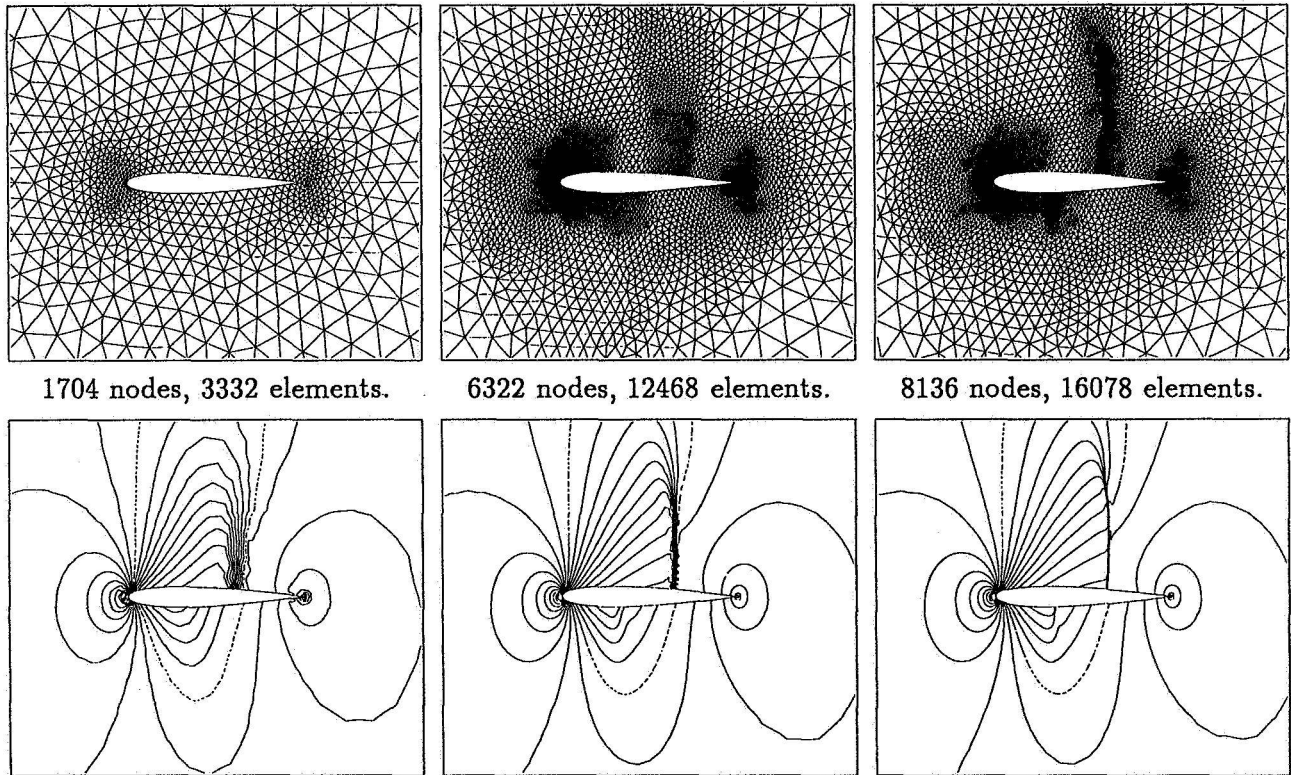


Figure 12: Evolution of the meshes and Iso C_p lines for a NACA0012 at $M_\infty = 0.80$, and $\alpha = 1.25^\circ$.

A partial view is given below, Figure 13, showing clearly the various types of adaptation performed, local refinement, regularisation, alignment, squeezing, and the corresponding pressure coefficient body profiles are presented to demonstrate the precision of solver on this adapted mesh.

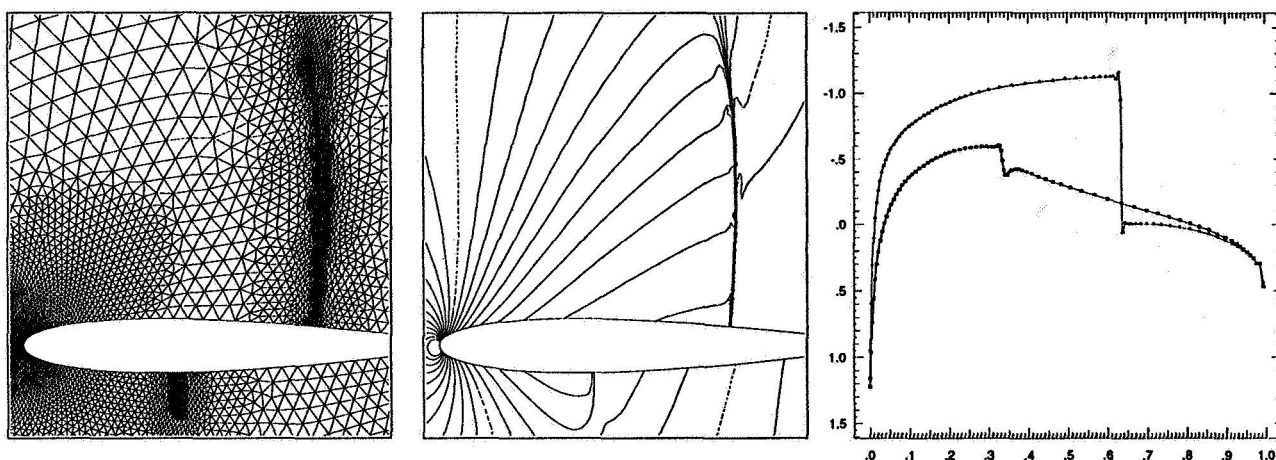


Figure 13: Mesh Details, Iso C_p lines ($\Delta C_p = 0.1$) and C_p body coefficients.

Transonic flow over a NACA0012 at $M_\infty = 0.95$, $\alpha = 0^\circ$

The second test case over the NACA 0012 presented here corresponds to the AGARD 03 test case. Despite the zero angle of attack for this high transonic flow, there is again a great sensibility of the solution to the position of the outer boundary. Indeed, the test case gives rise to a fish-tail shock structure, with an oblique shock attached to the trailing edge and a normal shock emanating from the intersection of the sonic line with the trailing edge oblique shock (see Figures 14). The distance

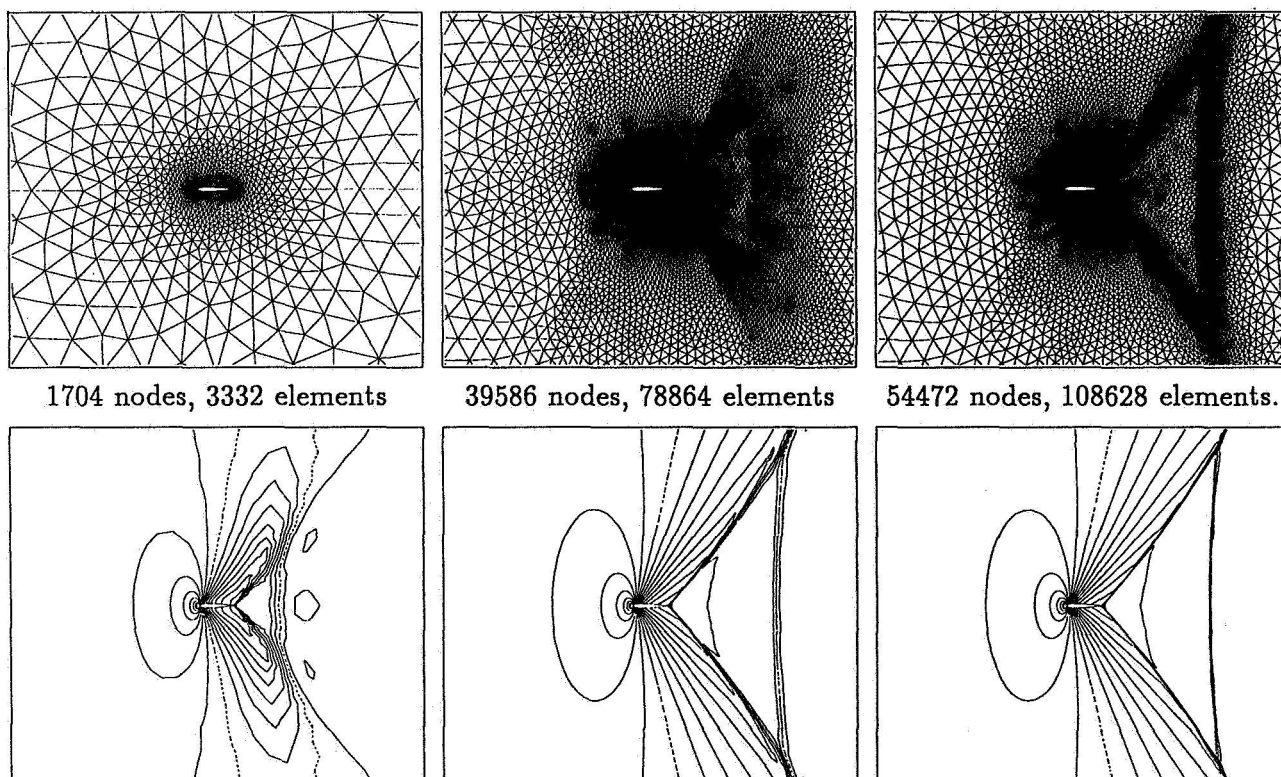


Figure 14: Evolution of meshes and Iso C_p lines, for a NACA0012 at $M_\infty = 0.95$, $\alpha = 0.0^\circ$

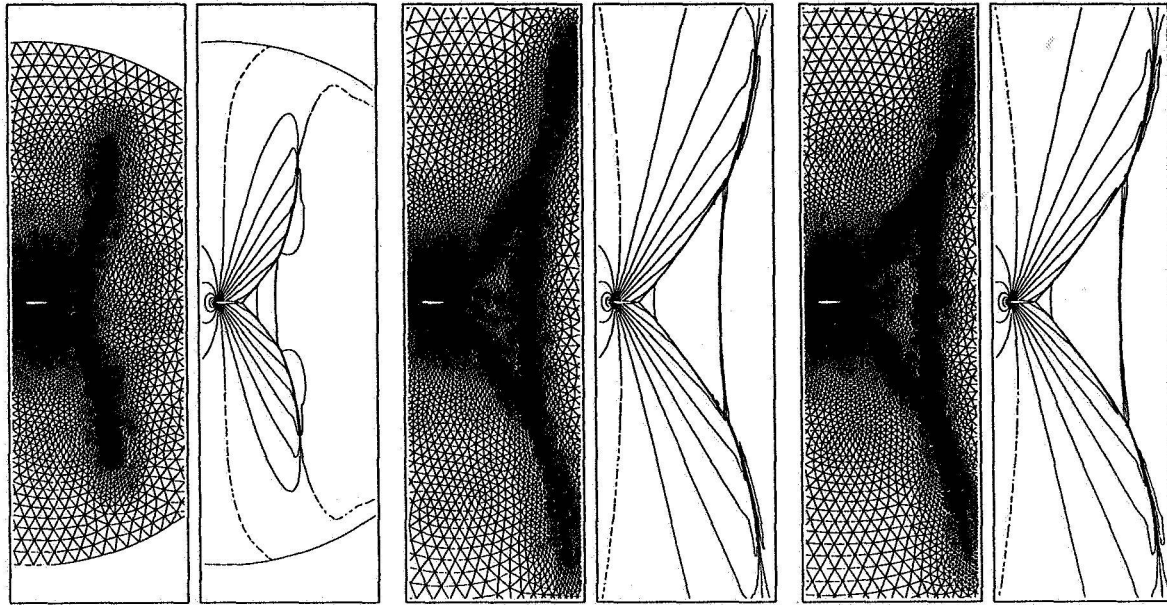


Figure 15: Meshes and C_p solutions for different outer boundary distances (10, 100, 10000 chords).

of this normal shock to the trailing edge is extremely sensitive not only to the position of the outer boundary, but also to the degree of adaptation around the body itself due to the strong expansion waves which are created along the profile.

The outer boundary was placed successfully at 10, 30, 100, 1000 and 10000 chord lengths from the body. Whereas the flight coefficients converged after 100 chords, the normal shock position continues to change throughout the series, increasing with increasing outer boundary distance. The degree of adaptation for these comparisons (Figures 15) was kept constant at a level equivalent to those of the final mesh shown in the Figures 14. The evolution of the refinements for the case with 100 chords are given in the Figures 14.

The convergence of the flight coefficients and the distance X_S of the normal shock to the trailing edge (normalised to the chord length) are tabulated below. As mentioned above, the normal shock distance evolves with the outer boundary distance, as has also been observed in the literature, [1]. The orders of magnitude of the drag coefficient are in agreement with the references, however the

Table 2: Flight coefficients for NACA0012 at $M_\infty = 0.95$, $\alpha = 0.0^\circ$

Outer boundary	C_l	C_d	C_m	X_S
10 chords	.0007	.1090	-.0002	1.280
30 chords	-.0003	.1091	.0002	2.515
100 chords	.0004	.1092	-.0001	3.125
300 chords	-.0009	.1091	.0001	3.179
1000 chords	.0003	.1091	.0000	3.195
10000 chords	-.0004	.1091	.0000	3.231

normal shock stand-off distance is found to be closer than the references for a low number of chord distances for the outer boundary. The calculations presented within this monograph however were often on non-adaptive grids, and for a relatively low number of chord lengths for the outer boundary distance.

Subsonic flow over a multi element airfoil at $M_\infty = 0.2$, $\alpha = 0.0^\circ$

This test case, corresponds to a four-element airfoil of Suddhoo and Hall, obtained by applying the Karman-Trefftz mapping function, [2]. The flow conditions are subsonic with zero angle of attack. The adaptation criteria here were taken to be based upon simply the gradient of the local Mach number, as there are no shocks present within the flow field. Again, this test case is a steady state calculation. Despite the absence of discontinuities the mesh adaptation algorithms provided very regular final meshes, starting from an initial coarse and non-optimal mesh, Figures 16.

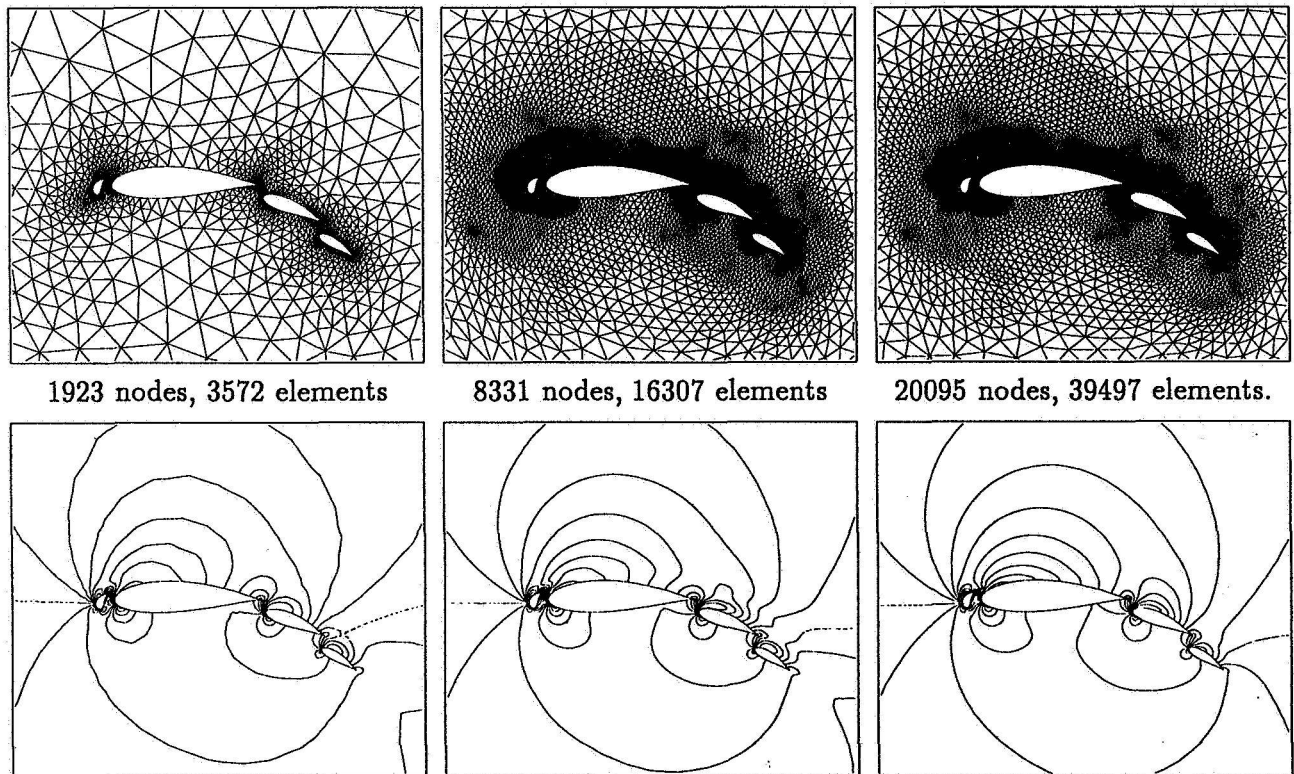


Figure 16: Evolution of the meshes and Iso C_p lines, ($\Delta C_p = 0.3$.) for the Multi-Element airfoil test case, for a subsonic flow of $M_\infty = 0.2$, $\alpha = 0.0^\circ$

Airfoils presenting Non uniqueness of the solutions for the Euler equations

In an AIAA paper concerning airfoils that can occur during an optimum design procedure, where the surface splines are perturbed, A. Jameson found the possibility of obtaining two distinct solution branches of the Euler equations, whereas the meshes used were extremely fine, and convergence was pushed to a maximum, in order to eliminate possibilities of anomalies due to a non-entropy preserving solution [5]. In fact, the Euler equations only admit weak solutions, and the only admissible ones must preserve the mathematical entropy condition. The airfoils studied here show a hysteresis effect in the lift/incidence polar when varying the angle of attack back and forth. The solutions generated

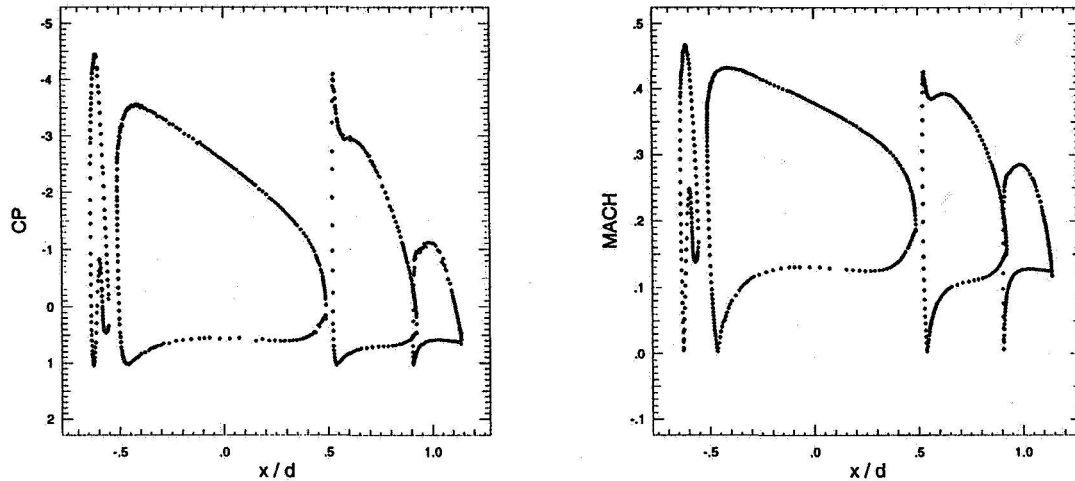


Figure 17: C_p and Mach number body plots, for a multi element airfoil at $M_\infty = 0.2$, $\alpha = 0.0^\circ$.

have two separate supersonic zones and shock waves for a particular angle of attack, and then by increasing or decreasing the lift angle around some critical value, a single supersonic zone is generated with a shift in the lift coefficient. It is extremely important to start this varying α process with an extremely fine and regular grid around the airfoil. The freestream Mach number is fixed at .78, and an initial series of computations is performed upto complete convergence for increasing angle of attack from zero to $-.70$, (up to 30000 time step iterations or more). The different lift coefficients are noted. Then for the solutions around a C_L of .6, angle of attack around $-.43$, a second series are initialised by taking the solution of an angle of attack slightly inferior. As the mesh adaptation is dynamically linked to the solver, the essential characteristics of the changing solution are taken into account, and each calculation is thus made on its own specific adapted mesh. Several different stages of resolution are illustrated in the Figures 19. The hysteresis effect is plotted in Figure 18. The values do not correspond exactly to those of Jameson, as the initial definition of the airfoil was not sufficiently detailed to obtain an equivalent initial shape. The transient stages are well captured by the dynamical mesh method, and the non-uniqueness can be established within a certain margin

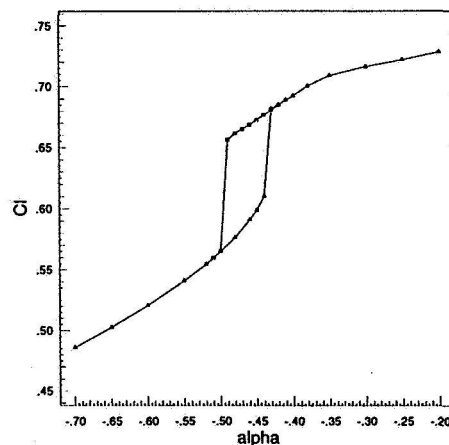


Figure 18: Lift versus angle of attack polar for $Mach_\infty = .78$

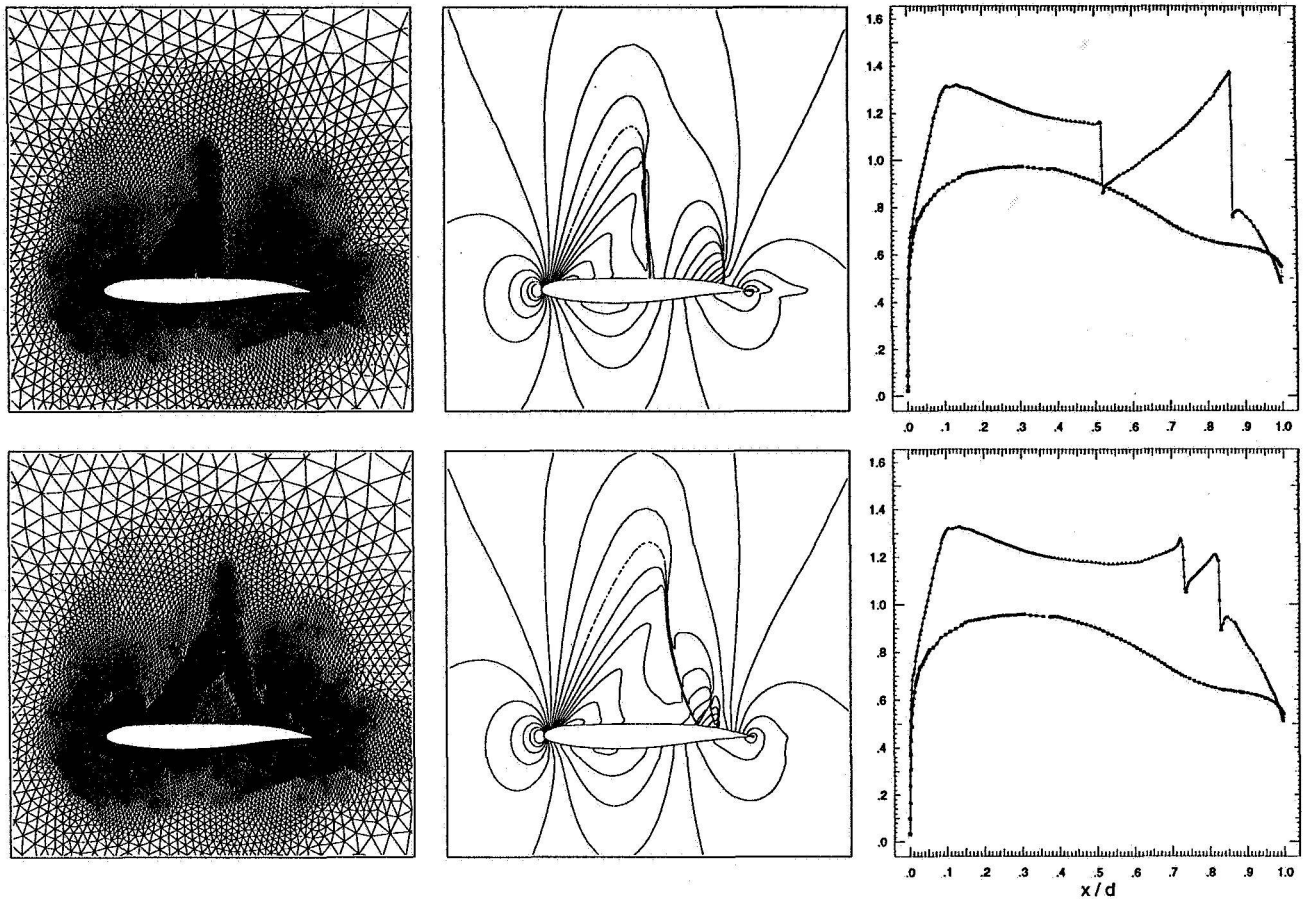


Figure 19: Different Meshes and Mach isolines/body plots obtained for $M_\infty = .78, \alpha = -.47^\circ$, corresponding to hysteresis effect of the lift polar.

of Mach, angle of attack and lift.

CONCLUSIONS

A new “anti”-data structure unstructured mesh adaptation procedure has been presented enabling the use of structural optimisation, together with freedom for derefinement of any level of mesh transformation. The improvement, for steady Euler flows solutions, in number of nodes and CPU time requirement can be up to a factor of 30, and the algorithms developed have also proved to produce very accurate results for unsteady cases. The methods have been applied to a number of test cases, and have proved to be a valuable design tool for aerodynamic investigation. The solver and the dynamical mesh algorithm are intimately coupled, their implementation on a distributed computational platform was undertaken in a Master-Slave environment between a CRAY YMP and a CRAY-T3D. The efficiency of such a technique is extremely promising, and in view of the minimisation of the communication phases by performing the adaptation and the partitioning on the Master, proves to be an interesting alternative to a complete MPP algorithm of adaptation and solution process, which necessitates sorting graphs, redistribution searches and high communication patterns.

References

- [1] Test Cases For Inviscid Flow Field Methods, AGARD - AR - 211, WG 07.
- [2] ICASE/LaRC Workshop on Adaptive Grid Methods, Hampton Virginia, 7-9 November, 1994.
- [3] Babuška, I. and Rheinboldt, W. C. : *Error estimates for adaptive finite element computations*. SIAM J. Numer. Anal., vol. 15, no.4, 1978.
- [4] Eriksson, K.; Johnson, C. and Lennblad, J. : *Error estimates and automatic time and space step control for linear parabolic problems*. SIAM J. Num. Anal. 1990.
- [5] Jameson, A. : *Airfoils Admitting Non-unique Solutions of the Euler equations* AIAA 91-1625.
- [6] Leyland, P.; Benkhaldoun, F.; Maman, N. and Larrouturou, B. : *Dynamical mesh adaptation criteria for accurate capturing of stiff phenomena in combustion*. To appear in Int.Journ. Num. Meth. in Heat and Mass Transfer 1993.
- [7] Löhner, R.; Morgan, K.; Vahdati, M.; Boris, J. P. and Book, D.L. : *FEM-FCT: Combining unstructured grids with high resolution*. Comm. Appl. Num. Meth., vol. 4, 1988, pp. 717-730.
- [8] Richter, R. : *Schémas de capture de discontinuités en maillage non-structuré avec adaptation dynamique*. PhD Thesis, EPFL, 1993.
- [9] Richter, R. ; Leyland, P. *Auto-Adaptive Finite Element Meshes and Distributed CFD* Article submitted for publication, January 1995.
- [10] Thomas, J.L. and Salas, M.D. : *Far-Field Boundary Conditions for Transonic Lifting Solutions to the Euler Equations* AIAA Journal, no.24, pp. 1074-1080, 1986.