IN-02-CR

7623

p 41

FINITE ELEMENT METHODS FOR INTEGRATED AERODYNAMIC HEATING ANALYSIS

Report for the Period 1 October 1989 : 30 September 1990

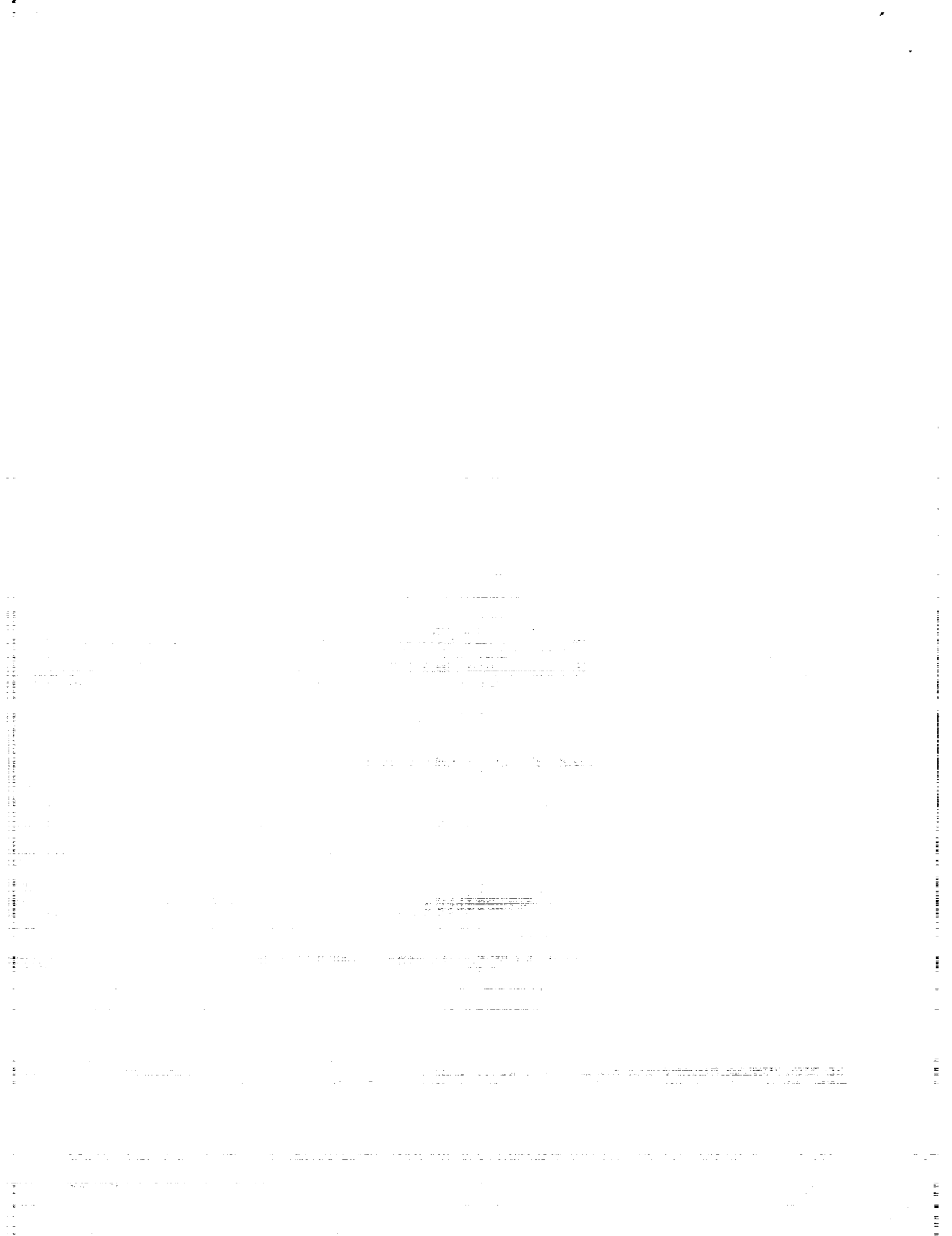
Prepared for the

Aerothermal Loads Branch
NASA Langley Research Center
Hampton, Virginia 23665, U.S.A.

GRANT NAGW-1809 Year 1

Submitted by the

Department of Aeronautics
Imperial College of Science, Technology and Medicine
Prince Consort Road
LONDON SW7 2BY
United Kingdom



INTRODUCTION

Over the past few years we have been developing finite element based procedures for the solution of high speed viscous compressible flows. In this research contract, which extends over a three year period, the objective is to build upon the finite element concepts which have already been demonstrated and to develop these ideas to produce a method which is applicable to the solution of large scale practical problems. The problems of interest range from 3D full vehicle Euler simulations to local analysis of 3D viscous laminar flow. Transient Euler flow simulations involving moving bodies are also to be included. An important feature of the research is to be the coupling of the flow solution methods with thermal/structural modelling techniques to provide an integrated fluid/thermal/structural modelling capability. Here we report upon the progress made towards achieving these goals during the first twelve month period of the research.

WORK ACCOMPLISHED

Over the past twelve months the research activity has concentrated on the two main areas of flow modelling and thermal modelling. Within the area of flow modelling work has been directed towards improving the efficiency, reliability and performance of an unstructured mesh generation technique. An implicit centred scheme for the solution of the Navier-Stokes equations has been investigated and initial work has been undertaken on the construction of an upwind scheme based upon the use of quadratic elements. An investigation into the possibilities of using space marching methods within the unstructured grid context has also been made. Within the area of thermal modelling, an explicit transient adaptive mesh approach for thermal problems has been produced and an implicit thermal solver based upon the use of the conjugate gradient algorithm has been developed. A description of the work accomplishments within each of these areas will now be given.

FLOW MODELLING

Progress in Mesh Generation Reliability, Efficiency and Performance

Over the past twelve months, work has been directed towards the improving the control and the quality of the meshes which are generated. Procedures for removing distorted elements and for mesh smoothing have been incorporated within the generators. These procedures are carried out automatically as the generation is performed. The surface generator has also been modified and made more general. The basic surface generator is currently under modification to enable it to handle open surfaces and to perform adaptive remeshing based upon quantities which are defined only on the surface. This feature was requested by the Aerothermal Loads Branch at

Langley R.C. as it will enable the software to be applied to the analysis of thermal stress development in shells and panelled structures. The process of fully documenting both the flow solvers and the mesh generation software has also begun. The new version of the mesh generation software was fully tested in an inviscid analysis of the flow over a shuttle vehicle. The intention was to reproduce certain computations reported earlier [1,2]. The particular flow case considered consisted of a free stream Mach number of 6 and an angle of attack of 26.6 degrees. A view of the surface mesh employed is given in figure 0a and a view of the computed solution in the symmetry plane is shown in figure 0b. The flow visualisation was accomplished by writing an interface program to enable us to use the software developed for viewing solutions computed on unstructured grids by R. Löhner for the CFD Branch at Langley R.C. Work has also commenced on the modelling of the flow over a shuttle with boosters, using the geometrical definition shown in figure 0c. To date, a mesh of roughly 2 million elements has been generated and a detail of the corresponding surface discretisation is shown in figure 0d. This mesh is to be used for flow computations and will be used later as the starting point in the modelling of the transient flow during a the shuttle separation.

Implicit Centred Scheme for the Navier-Stokes Equations

A fast algorithm has been devised for constructing continuous lines, made up of element sides, which pass through each node of a general unstructured triangular mesh and which are generally aligned in prescribed directions. The lines are used as a basis of an adaptive fully implicit unstructured grid procedure for the solution of two dimensional problems of steady high speed flows, where the equation system is solved by line relaxation using a block tridiagonal equation solver. The flow algorithm is based upon an implicit finite element scheme of the Taylor-Galerkin family. The method has been used with success for the solution of both transonic inviscid and hypersonic laminar viscous problems in two dimensions. An initial approach for the solution of viscous flows in three dimensions has been based upon the use of an implicit/explicit algorithm. To achieve this, a grid exhibiting structure in the normal direction is employed in the vicinity of solid walls while, away from these regions, the grid is totally unstructured. In the structured region lines in the normal direction are readily identified, while lines in the surfaces parallel to the walls are constructed using the proposed two dimensional procedure. The implicit form of the algorithm is used in the structured region, with the equation system solution being achieved by line relaxation. An explicit Taylor-Galerkin method is used on the unstructured portion of the mesh. This method has been used to successfully solve three dimensional hypersonic laminar viscous flow over a double ellipsoid configuration and to produce results which compare well with the available experimental observations of pressure and heating rate distributions.

Quadratic Upwind Algorithm

A vectorised version of a cell centred scheme for the solution of the compressible laminar Navier-Stokes equations has been produced. The method uses Roe's approximate Riemann solver with a higher order extension based upon linear reconstruction with slope limiting. The vectorisation has proved to be effective with a vector/scalar speed-up ratio of around eleven being achieved. However, this viscous code has caused a large number of problems which we are still attempting to resolve. A detailed analysis of a Mach 14 flow over a 24 degree compression corner has been attempted. While the solutions on structured triangular meshes of 26*26 and 51*51 appear to behave as expected, the computed solution on a structured 101*101 mesh exhibits a strange behaviour. On this mesh, there is a tendency for the code to predict a separation length which increases with the number of iterations computed. Investigations are continuing to attempt to understand if this is a problem with highly stretched triangular elements or if there is a problem with the basic algorithmic implementation. It is still our intention to attempt a solution of this problem on a mesh of over 60,000 elements, with over 3,000 points on the wall, when these difficulties have been overcome. The advantage of this mesh is that the cells near the wall will all have an aspect ratio of around unity, so it can be claimed that the computed solution is free from aspect ratio effects. An investigation of the behaviour of the algorithm with quadratic reconstruction has been made within the context of the Euler equations. Initial experiences have indicated that the method is certainly less sensitive to variations in the grid than the algorithm using linear reconstruction. However, the method has associated stability and limiting problems which need to be addressed before it is possible to produce a robust implementation.

Space Marching

As this is a new activity area for us, we will describe our work in some detail. It is well known that in a supersonic flow field the disturbances are carried down stream. This physical property can be employed to devise efficient algorithms for the numerical solution of supersonic steady state problems. The class of schemes based on this line of reasoning are known as space marching schemes [3]. Here, the domain of interest is divided into planes which are nearly normal to the direction of the flow. Since the solution at each of these planes (stations) depends only on the solution in the upstream stations, the steady state equations can be solved completely for each plane and the solution marched downstream. Hence, the solution can be obtained by performing only one sweep over the computational domain. The supersonic flow regime is characterised by strong discontinuities. Upwind methods are used, due to their excellent properties in handling such flow features. For realistic flows, even when the free stream Mach number is much higher than unity, regions of subsonic flow exist near the surface of the body [4]. In these subsonic regions (pockets), disturbances are propagated in all

directions. As a result, the space marching technique can not be applied for the subsonic parts of the flow field. An efficient procedure can be obtained by combining the use of space marching in the prevailing supersonic regions of the flow and time marching in the subsonic regions. This approach has been proposed by Chakravarthy and Szema [5] to solve the Euler equations for an analytical forebody configuration. For space marching schemes, mesh generation requires some consideration. Unstructured grids provide great flexibility in handling complex realistic geometries, however, it can be seen from the above discussion that space marching techniques require a structured mesh. Here, a compromise is arrived at by using grids which exhibit a structure in the direction of the flow but are otherwise unstructured [6]. An upwind space marching algorithm, based on the upwind cell centred scheme already implemented at the Aerothermal Loads Branch, NASA Langley Research Center, is presented for the Euler equations. Application of this approach is first described for the case of a fully supersonic flow. A discussion on the types of grids suitable for this method is given. A synopsis of the mesh enrichment procedure for improving the quality of the solution, whilst retaining the basic structure of the grid is presented. Attention is then drawn to more realistic situations where the supersonic flow contains pockets of subsonic flow. In this case the solution domain is divided into space marching and time marching regions. The formulation of the numerical scheme is derived from the Euler equations in transient form. Hence, a uniform approach can be taken for both space marching and time marching zones. The extension to higher order accuracy is then considered. Calculation of the gradients requires the use of values from downstream planes, which is in conflict with the space marching philosophy. This problem is addressed and a remedy provided. Numerical examples are presented to evaluate the performance of the new algorithm.

Space marching for the Euler equations, fully supersonic flow

For many practical high speed flight situations, where the body is slender, the direction of the flow around the body does not vary drastically from the free stream direction. For these cases, the domain of interest can be subdivided into a set of planes which are roughly normal to the direction of the flow. The mesh generation within each of these planes can be completely unstructured. The equations to be solved are the steady state Euler equations which can be written in the following compact form

$$\frac{\partial \boldsymbol{F}_i}{\partial x_i} = \boldsymbol{0} \qquad i = 1, 2 \qquad (1)$$

However, in order to have a unified formulation for both supersonic and subsonic regions, the equations are considered in their transient form. The numerical discretisation is basically achieved by a cell centred formulation and the numerical flux of Roe [7] is used when the

solution of a Riemann problem is required. For fully supersonic flows, starting from the inflow, the time dependent equations can be solved for each marching plane in turn. The finite volume formulation can be written as

$$\frac{\Delta U_e}{\Delta t_e} = - \frac{1}{\Omega_e} \sum_{s_e} F_n \, \delta_s \qquad (2)$$

where Ω_e is the area of the element e and δ_{se} is the length of the side s_e. If a side has both its neighbouring elements in the marching plane then the numerical fluxes, denoted by F_n, are obtained by use of the approximate Riemann solver. For the sides that lie on the boundaries of the marching plane the numerical flux is simply calculated using the upwind element values, and application of an approximate Riemann solver is unnecessary (see figure 1). As long as the component of the flow velocity normal to the marching plane remains supersonic, this procedure is correct. It is clear that this determination of the numerical fluxes implies that there is no contribution from the downstream elements. By choosing a large time step (e.g. of the order of 10^6) the transient term in equation (2) is negligible and the formulation leads to an iterative procedure for the steady state equations. The system of equations resulting from a discretisation of (2) is solved implicitly, in the marching plane, according to a point implicit iteration. It should be noted that the sweeping is now only performed over the number of elements in the marching plane, and not over the total number of elements, thereby achieving fast convergence for fully supersonic flows. Once the solution is obtained for a marching plane, the same procedure is repeated for the next plane downstream. In this manner one sweep over the domain will result in the steady state solution being obtained.

Mesh enrichment

The advancing front technique for mesh generation [8] starts from a coarse background grid. Within this grid, starting from the boundaries of the domain, new mesh points are introduced according to some geometric parameters. These parameters are the node spacing, δ ; the stretching parameter, s ; and a direction of stretching denoted by the vector s. New elements are formed by connecting the generated points in a manner that prevents the generation of highly distorted elements, and the process is continued until the whole domain is covered. This generator also has the capability of generating grids within pre-specified regions. For the grids employed here, each marching plane is defined as an independent region for the mesh generator. One of the main advantages of using unstructured grids is the ease in which mesh refinement can be carried out. The mesh refinement procedure is more restricted for space marching calculations, since the structure of the marching planes must be preserved. So the adaptive remeshing technique, used in the previous chapters to improve the quality of the grids and hence the solution, is not directly applicable here. Mesh enrichment [9], however, has the

property that the underlying structure of the mesh does not change. The structure of the devised mesh enrichment algorithm is now discussed. An important factor in mesh refinement is the evaluation of an error indicator which is an indication of the level of error in the numerical solution on the present mesh. For problems where the analytical solution is not available, and indeed these are the problems which we are interested in, it is not possible to determine the exact level of error introduced by a numerical scheme. For the inviscid problems, considered here, the mesh is refined in order that more elements are introduced where the gradients in the solution are large. Therefore, the gradient of a chosen variable can be a basis for the error indicator. This variable, termed as the key variable, is usually taken to be either the density, the pressure, or the local Mach number. The gradients of the key variable are determined for each element from the point values of the variables and the derivatives of the standard linear shape functions [10] as

$$\frac{\partial V}{\partial x_i} = \sum_{m=1,3} V_{P_m} \frac{\partial N_m}{\partial x_i} \tag{3}$$

where m is the local vertex number and V_P represents the value of the key variable at vertex P. The summation, in the above equation, takes place over the vertices of each triangular element. In equation (3), N_m is the linear triangular shape function for vertex m. Expressions for other vertices follow from a cyclic permutation of subscripts in the order i, j, k. The point values in turn, are obtained from the cell-averaged values via a consistent finite element procedure which can be written as

$$V_P = \frac{\sum \Omega_e V_e^a}{\sum \Omega_e} \tag{4}$$

where V_e^a is the cell-averaged value of the key variable at element e. In the above equation the summation extends over all the elements surrounding point P. The maximum gradient of V over each element is obtained from the expression

$$\nabla V_e = \left(\left(\frac{\partial V}{\partial x} \right)_e^2 + \left(\frac{\partial V}{\partial y} \right)_e^2 \right)^{\frac{1}{2}} \tag{5}$$

The maximum value of the error indicator denoted here by β is determined by sweeping over all the elements as

$$\beta = max\ (\ \Omega_e \nabla V_e) \tag{6}$$

The criteria for mesh enrichment is that all the elements where the error indicator is larger than a certain proportion of β , i.e. those elements for which

$$\Omega_e \nabla U_e > \alpha \beta \tag{7}$$

need to be refined. In the above inequality α is the scaling factor. In this way the region which needs to be refined can be determined. To avoid the creation of points which are not consistently connected to the other points, care must be taken for the elements that lie on the boundary of the refinement region. If any element has all its three vertices lying on this boundary then this element is also added to the refinement zone. This is illustrated in figure 2. Indicating the refinement region by Ω_R, all the elements in Ω_R are subdivided into four triangular elements as shown in figure 3. Application of this procedure will lead to introducing some points, which are not properly connected to their surrounding elements, around the boundary of Ω_R. This problem is solved by subdividing the elements that have two of their vertices on the boundary of Ω_R into two. The final refined mesh is as shown in figure 4. It should be emphasized here, that the mesh enrichment procedure does not lead to any change in the number of marching planes. Therefore, only the number of elements in those marching planes where the refinement has taken place will change.

Supersonic flow with subsonic pockets

For supersonic flow with subsonic pockets, the problem is to track down and identify the subsonic regions and solve the equations in those regions by using a time marching scheme. The algorithm devised to accomplish this goal can be described through the following stages:

I. Starting from the inflow each plane is solved in turn without taking contributions from downstream. For the first sweep across the plane, which is defined as first sweeping over the elements according to their numbering and then sweeping in the reverse order, a small CFL number is used. The reason for using a small Courant number is that if the flow at steady state in a particular plane is subsonic, a relaxation procedure must be used and this calls for a sensible choice of the time step size and a consideration of downstream conditions.

II. The local Mach number is calculated for all the elements in the marching plane. If this is less than unity (subsonic) for any of these elements, the current marching plane is recorded as pertaining to the relaxation region.

III. The plane recorded as being subsonic in stage II and its two adjacent planes are taken to be the relaxation region (see figure 5) which has to be solved by standard time marching techniques. Since the flow in marching planes *mp-1* and *mp+1* is still supersonic, there is no need to consider contributions from the next neighbouring planes. It should be noted here that application of the time marching relaxation method to a supersonic plane is

allowed, but the contrary, i.e. using a space marching method for a subsonic plane, is not permitted. Therefore, the numerical relaxation region must contain the physical subsonic region for the whole scheme to be consistent and stable.

IV. After one sweep over the subsonic region the two planes which form the boundaries of the relaxation region (in this case planes *mp-1* and *mp+1*) are checked to see if any subsonic velocity has formed. There are two possibilities:

IV.a. If the flow is subsonic in one or both of these planes , for the next sweep the next neighbouring plane will also be included in the relaxation region. For example if after stage III the flow has become subsonic in some positions in plane *mp-1*, the relaxation region is extended to include plane *mp-2*. Then the solution of the unsteady equations in the relaxation region continues (stage III).

IV.b. If the flow is still supersonic, stage III is repeated until convergence in the relaxation region is achieved.

By repetition of stages III and IV the subsonic region gradually forms until it reaches its full extension in the steady state (See figure 6). It should be mentioned here that this procedure is compatible with the physical behaviour of the flow around a body which has suddenly "materialised" in a supersonic regime. Once the subsonic region is fully formed, step IV will not add any new plane to the relaxation region and the time marching procedure will be continued in the relaxation region until convergence is obtained.

V. When the steady state is achieved in the relaxation region, the space marching procedure is continued downstream.

The numerical examples presented here have indicated that the above procedure is capable of producing results which are consistent with the results of time marching calculations.

Higher order spatial accuracy

The extension to higher order accuracy follows from the use of linear reconstruction with limiting. There are, however, some modifications necessary to comply with the requirements of the space marching technique. The standard procedure makes use of the values of the variables in all the elements which have a common vertex with the element being considered. But the space marching methodology is based on the assumption that information does not propagate upstream. This calls for a reformulation of the gradient recovery procedure as follows. Consider the marching plane *mp* where the flow is fully supersonic, and its neighbouring planes. From the cell-averaged values of the variables the corresponding values

are calculated at the vertices. This is done in a consistent finite element manner as described above. Hence for a point K and its surrounding elements one can write

$$U_P = \frac{\sum_{e \in mp \text{ or } mp\text{-}1} \Omega_e U_e^a}{\sum_{e \in mp \text{ or } mp\text{-}1} \Omega_e} \tag{8}$$

It should be noted that the summation extends over the elements in the marching plane and its neighbouring upstream plane (see figure 7). If the plane is in the subsonic region the summation takes place over all the surrounding elements and the formulation will be identical to (4). The limiting procedure is accomplished by employing the standard method used in the upwind cell centred code. The evaluation of the maximum and minimum allowable values, however, must be done differently. If the marching plane is in the space marching zone (supersonic flow every where across the plane) then only those neighbouring elements which are in the marching plane itself and the upstream planes are considered. This is due to the fact that the downstream values are not updated. This method may not be always monotonicity preserving. The alternative is to make use of all the available values, i.e. upstream values, values at the marching plane considered, and downstream values. This means that the solution procedure, even in the fully supersonic regions, will depend on the downstream values. In this case a much smaller CFL number is allowed for the supersonic planes in the space marching zone and the whole space marching procedure as was outlined above has to be repeated iteratively until convergence is achieved throughout the domain. Consequently, the computation will take a longer time which is comparable to the computing time for a time marching scheme. Numerical examples performed during the course of this research work have produced good results using the former approach.

Numerical Results

To evaluate the behaviour of the proposed scheme two examples are provided. The first example is for the case where the flow is supersonic every where. The second example corresponds to the case of a supersonic flow with embedded subsonic regions.

Supersonic flow past a wedge

This example consists of a Mach 2 inviscid flow past a symmetrical wedge. The half angle of the wedge is 10°. Due to the symmetry of the problem, only half of the domain is considered. The analytical solution to this problem can be obtained from elementary gas dynamics [11]. The solution consists of two regions of constant states which are separated by an oblique shock wave as is sketched in figure 8. From the information given in figure 8, it can be seen that the

flow remains supersonic behind the shock wave. Therefore, the flow field is supersonic throughout the domain, and the space marching technique can be employed. A relatively regular unstructured grid is employed initially for this problem. The mesh consists of 158 triangular elements and 99 points, and is shown in figure 9. The mesh is subdivided into 13 marching planes. Solution contours, by using the higher order scheme, for the pressure coefficient are also depicted in figure 9. The solution is virtually identical to the solution obtained by the higher order upwind scheme in its standard form. The convergence history for this problem is shown in figure 13a. The residual is defined as the L_2 norm of the term $\Delta\rho/\Delta t$. It should be noted that each iteration represents a sweep over the number of elements in the marching plane, and not the total number of elements. It should also be mentioned that the jumps in the convergence curve correspond to the beginning of iterations in the new marching plane. To illustrate the application of the above mesh enrichment procedure. Using the solution on the initial mesh, a series of successive refinements are performed. The number of elements in the refined meshes are 487, 1203, and 2713, respectively. The computations, in each case, has started from the converged solution on the previous mesh. The adaptively refined meshes together with the contours of C_P are presented in figures 10, 11, and 12. These figures clearly demonstrate the enhanced resolution obtained by using the mesh enrichment procedure. The corresponding convergence curves are given in figures 13 and 14.

Supersonic flow past a blunt body

This example consists of a supersonic flow past a symmetrical blunt body. The free stream Mach number of the flow is 6.57, and $\gamma = 1.38$. The solution to this problem exhibits a subsonic region behind the bow shock. Due to symmetry, the computational grid covers only half of the domain. The initial mesh used for this problem is shown in figure 15. The mesh consists of 419 triangular elements and 243 points. The number of marching planes is 15. The results obtained with the Streamwise Upwind Petrov-Galerkin (SUPG) method of Hughes are also available for this problem. The mesh employed for the SUPG calculations is also given in figure 15. Calculations for this example are performed by using the higher order scheme. In figure 16 the solution contours of pressure coefficient, and Mach number for the space marching calculations are given. Using the values of density, and Mach number along the axis and over the surface of the blunt body, a comparison is made between the space marching and the SUPG results. This is shown in figure 17. The corresponding comparisons for the velocity components at the exit are presented in figure 18. As can be seen from these figures, although the mesh used for the SUPG computations is finer than the one for space marching, good resolution is obtained with the present algorithm. The convergence history for the space marching calculations is shown in figure 19. The nearly flat region in the convergence curve corresponds to the formation of the subsonic region. Two levels of mesh enrichment are used to enhance the accuracy of the solution. There are 1121, and 2043 elements in the first and

second refined mesh respectively. These are presented in figure 20. In each case, the converged solution is used as the initial condition for the next refined grid. Solution contours of C_P and M for the first refined mesh are given in figure 21. Figure 22 illustrates the variation of density and Mach number along the axis and over the surface of the blunt body, for the space marching and the SUPG computations. The values of velocity components along the outflow boundary are shown in figure 23. The corresponding results for the second refined mesh are presented in figures 24, 25, and 26. Computed boundary of the subsonic region on the final mesh is shown in figure 27. Figure 28 presents the convergence curves for these two cases. As can be seen from these figures the solution obtained with the space marching scheme compares well with the SUPG results. The resolution of the shock wave, even on a coarser mesh, is sharper in the space marching results. The converged steady state solution, in each case, has been obtained by one pass over the solution domain. No monotonicity problem was observed for the higher order solution in this example.

THERMAL MODELLING

Transient Adaptive Remeshing Solution Procedure

An unstructured adaptive remeshing technique has been devised for the solution of transient thermal problems. The method relies heavily upon upon the software written originally for the solution of strongly transient compressible flows. The method advances the solution explicitly in time for a prescribed number of steps and then subjects the computed solution to an error estimation analysis. A new distribution of the mesh parameters is computed from the error variation and the mesh is reconstructed locally in regions where the original mesh is regarded as being either too fine or too coarse. In this way the available degrees of freedom are efficiently utilised. The performance of the method has been illustrated by application to the solution of a problem involving heat conduction with a square region with moving surface heat sources. This code is currently being used at the Aerothermal Loads Branch, NASA Langley R.C. for transient thermal structural analysis. The present version of the code can only handle problems involving straight boundary segments but this restriction will be removed in the new version of the code which will be released during the next twelve month period.

Implicit Thermal Analysis Solver

As simple explicit methods are generally regarded as inefficient for the solution of non-linear heat conduction problems, an implicit formulation has been introduced. To maintain the overall efficiency of the thermal analysis module, the solution of the implicit equation system is achieved via a conjugate gradient procedure. A full analysis of the transient adaptive remeshing process when used in conjunction with this solver is currently underway.

CONCLUSIONS

Work has progressed satisfactorily, in a number of different areas, in the directions needed for successful completion of the project. However, a major concern are the problems which have been encountered with the cell-centred upwind code, when applied to the analysis of high Mach number viscous flows using meshes of triangular elements, as these problems have hampered the investigations into the quadratic extension. It is to be hoped that these difficulties will be overcome during the next phase of the research.

REFERENCES

[1] R. P. F. Bradley, P. M. Siemens and K. J. Weilmunster, "Comparison of shuttle flight pressure data to computational and wind tunnel results", AIAA Paper 81-2477, 1981.

[2] K. J. Weilmunster, "High angle of attack inviscid flow calculations over a shuttle-like vehicle with comparison to the flight data", AIAA Paper 83-1798, 1983.

[3] D.A. Anderson, J.C. Tannehill, R.H. Pletcher, "Computational fluid mechanics and heat transfer", McGraw-Hill, 1984.

[4] K.Y. Szema, S.R. Chakravarthy, V. Shauban, J. Byerly, "Comparison of Euler and full potential marching techniques for flows over complex configurations", AIAA Paper 86-0244, 1986.

[5] S.R. Chakravarthy, K.Y. Szema, "An Euler solver for three-dimensional supersonic flows with subsonic pockets", AIAA Paper 85-1703, 1985.

[6] W.D. McGrory, R.W. Walters, R. Löhner, "A three-dimensional space marching algorithm for the solution of the Euler equations on unstructured grids", AIAA Paper 90-0014, 1990.

[7] P.L. Roe, "Approximate Riemann solvers, parameter vectors and difference schemes", J. Comp. Phys. 43, 357-372, 1981.

[8] J. Peraire, K. Morgan, M. Vahdati, O.C. Zienkiewicz, "Adaptive remeshing for compressible flow computation", J. Comp. Phys. 72, 449-466, 1987.

[9] K. Morgan, J. Peraire, "Finite element methods for compressible flows", von Karman Institute for Fluid Dynamics Lecture Series 1987-04, 1987.

[10] O.C. Zienkiewicz, K. Morgan, " Finite elements and approximation", John Wiley & Sons, 1983.

[11] J.D. Anderson, Jr., "Fundamentals of aerodynamics", McGraw-Hill, 1985.

PUBLICATIONS

[a] R. R. Thareja, R. K. Prabhu, K. Morgan, J. Peraire, J. Peiro and S. Soltani, "Applications of an adaptive unstructured solution algorithm to the analysis of high speed flows", AIAA Paper 90-0395, 1990.

[b] O. Hassan, M. Vahdati, S. Soltani, J. Peraire and K. Morgan, "Contribution to problem 3", Proceedings of the Workshop on Hypersonic Flows for Reentry Problems, Antibes, 1990.

[c] O. Hassan, J. Peiro, J. Peraire and K. Morgan, "Contribution to problem 6", Proceedings of the Workshop on Hypersonic Flows for Reentry Problems, Antibes, 1990.

[d] S. Soltani, K. Morgan and J. Peraire, "An upwind unstructured grid cell-centred scheme for compressible flow", Proceedings of the NUMETA Conference, Swansea, 1990.

[e] J. Peiro, M. Vahdati, J. Peraire and K. Morgan, "A comparison of three unstructured mesh algorithms for solving the Euler equations", Proceedings of the Langley Workshop on Unstructured Flow Solvers, 1990.

[f] O. Hassan, K. Morgan and J. Peraire, "An implicit finite element method for high speed flows", AIAA Paper 90-0402, 1990.

[g] J. Peraire, K. Morgan and J. Peiro, "Unstructured mesh methods for CFD", von Karman Institute for Fluid Mechanics Lecture Series, 1990.

figure 0a
surface discretisation of a shuttle vehicle

figure 0b
computed solution in the symmetry plane of the flow over a shuttle

figure 0c
geometrical definition for shuttle/booster configuration

figure 0d
surface discretisation of shuttle/booster configuration

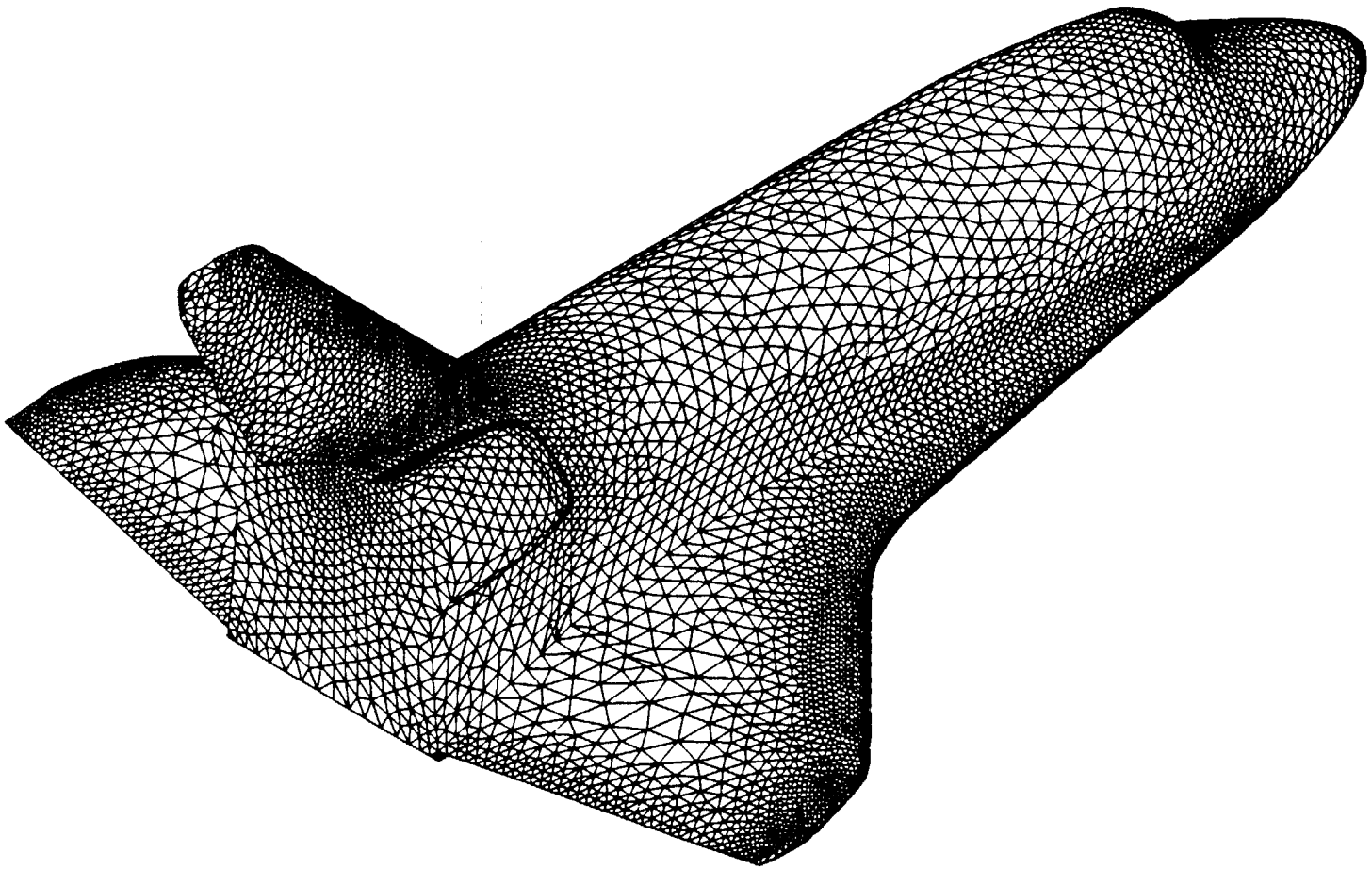


figure 0a
surface discretisation of a shuttle vehicle

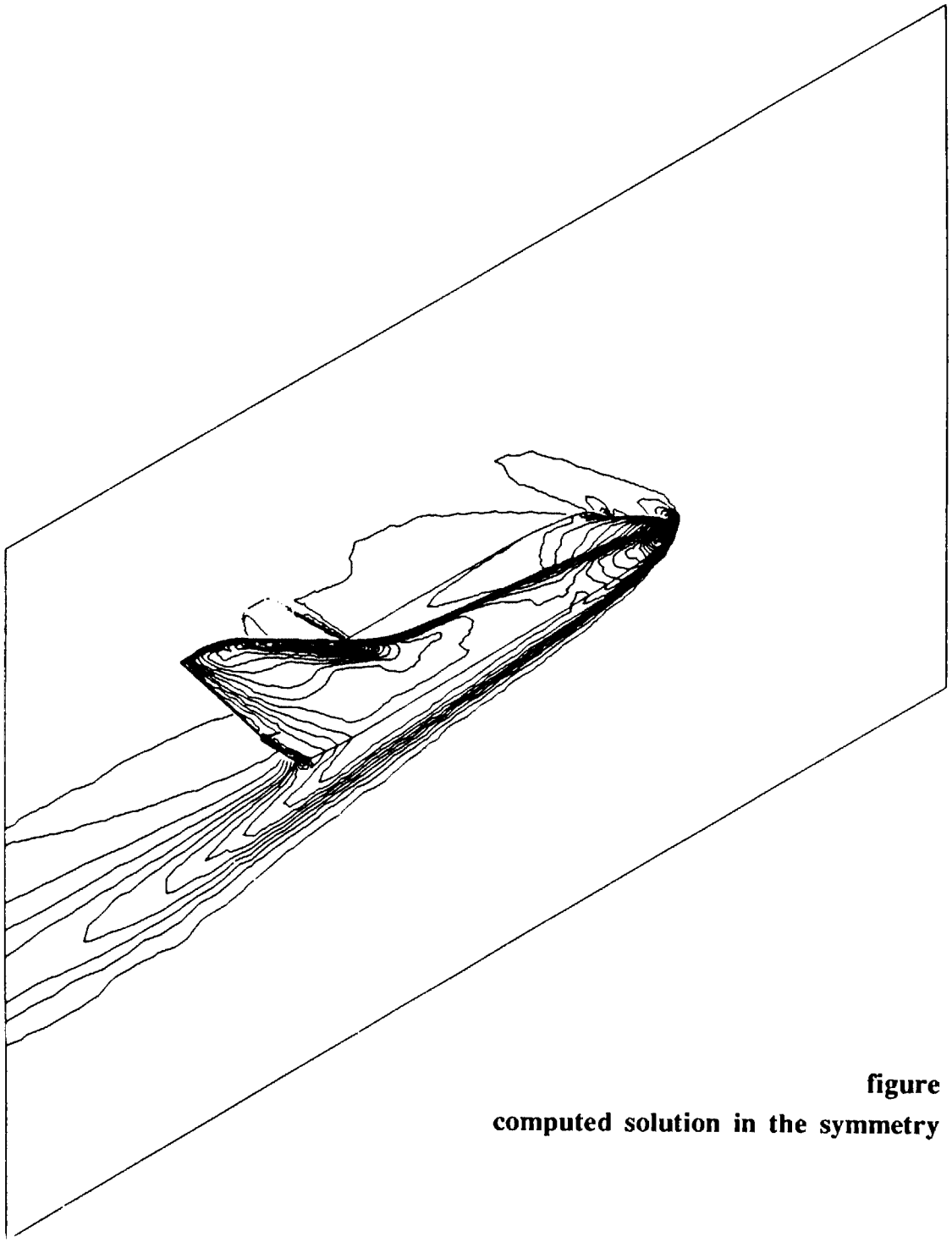


figure 0b
computed solution in the symmetry plane of the flow over a shuttle

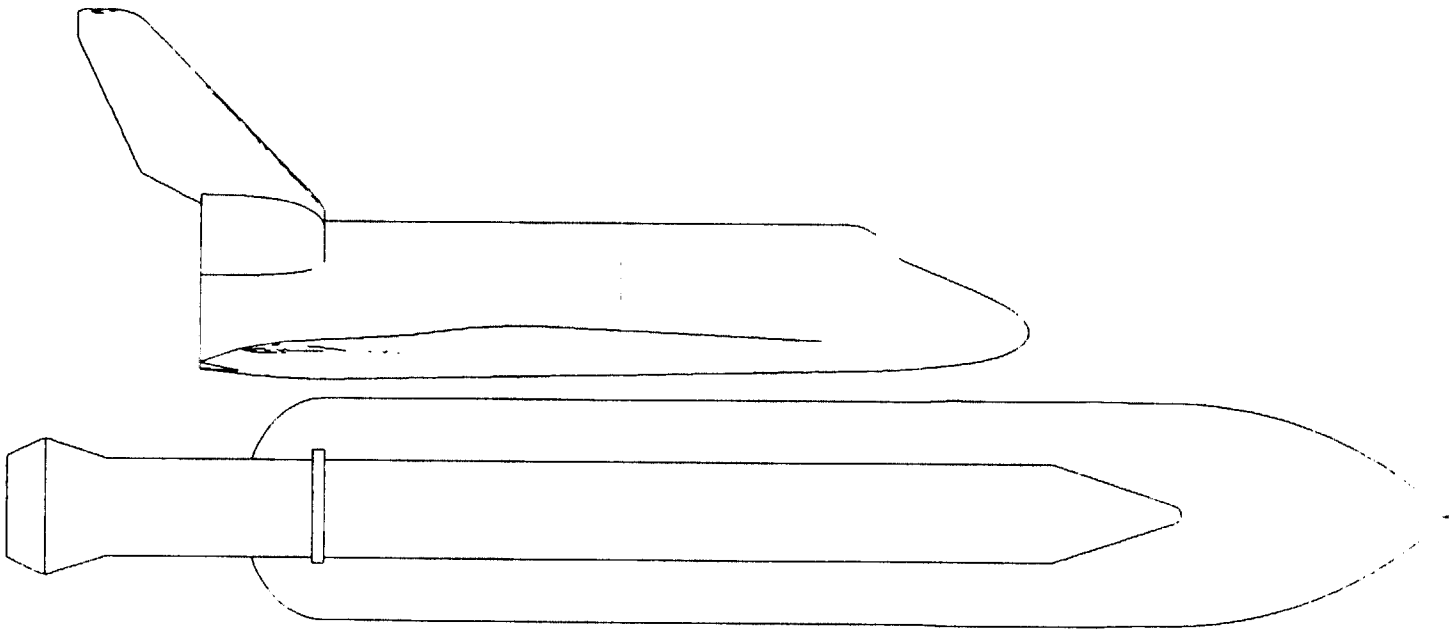


figure 0c
geometrical definition for shuttle/booster configuration

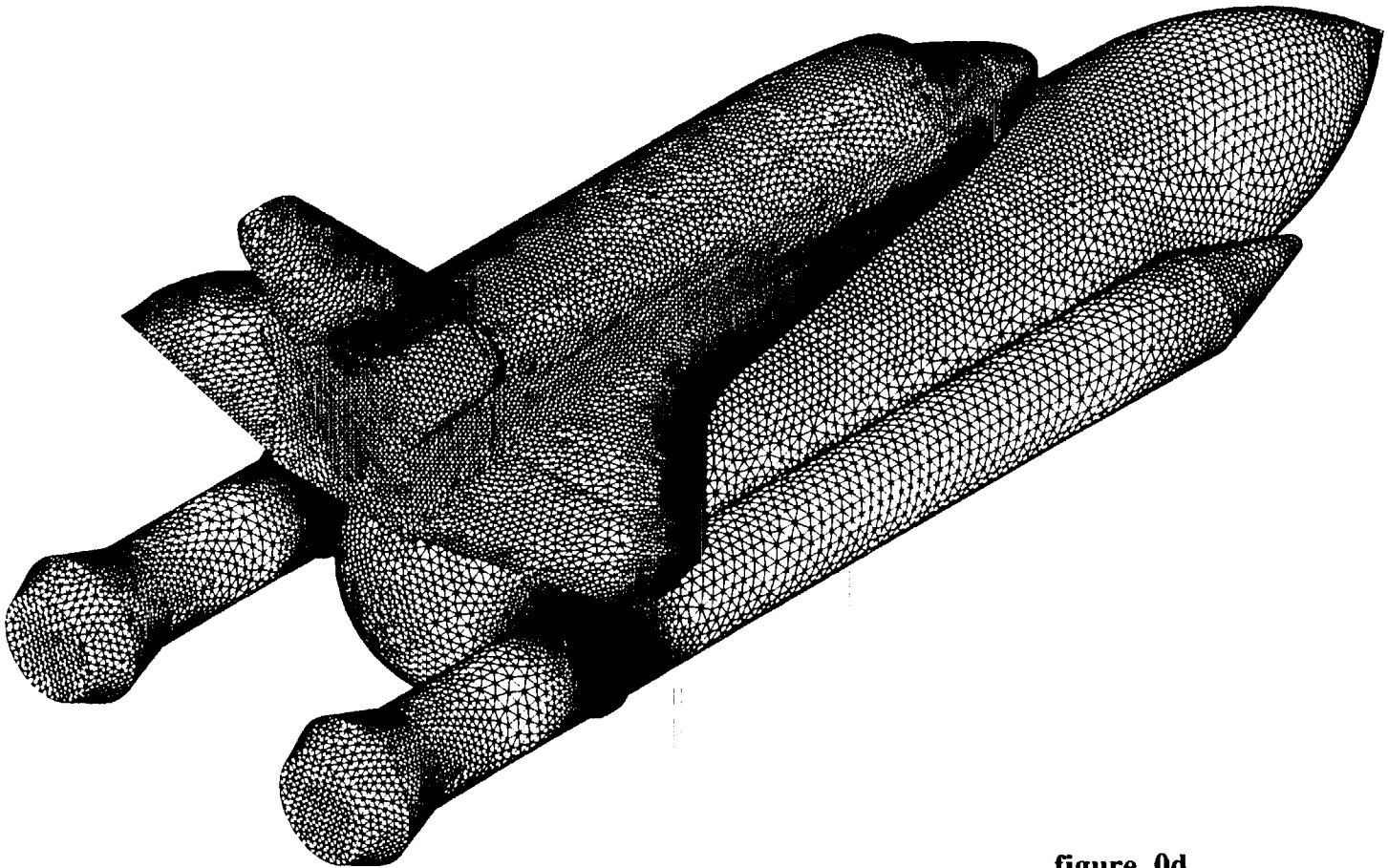


figure 0d
surface discretisation of shuttle/booster configuration

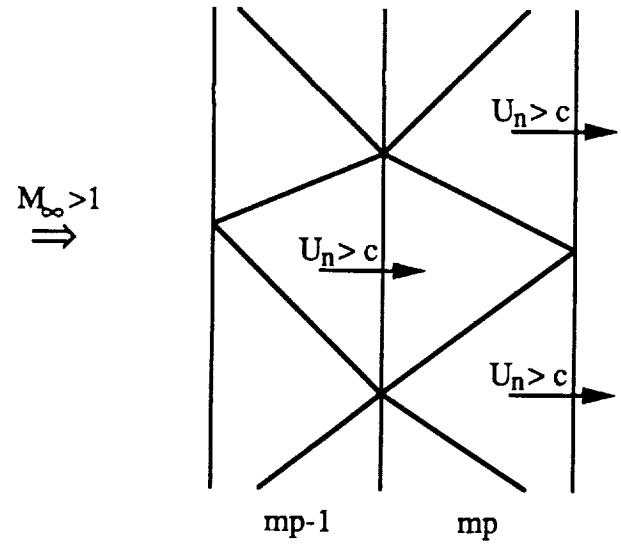


figure 1

marching planes for supersonic problem

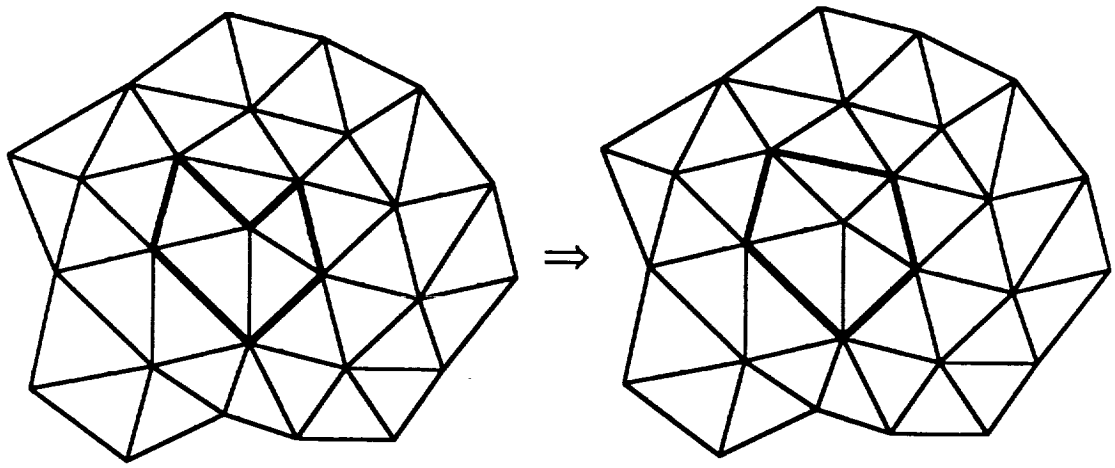


figure 2

formation of the mesh refinement region Ω_R

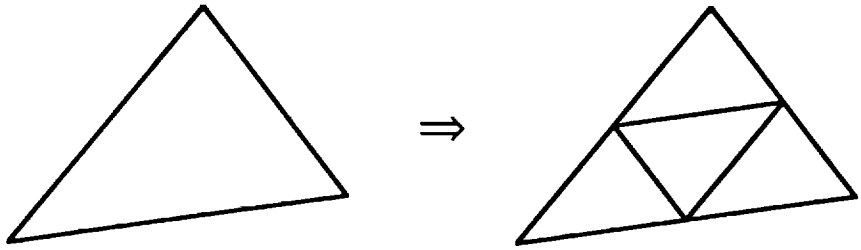


figure 3

subdivision of a triangular element for mesh enrichement

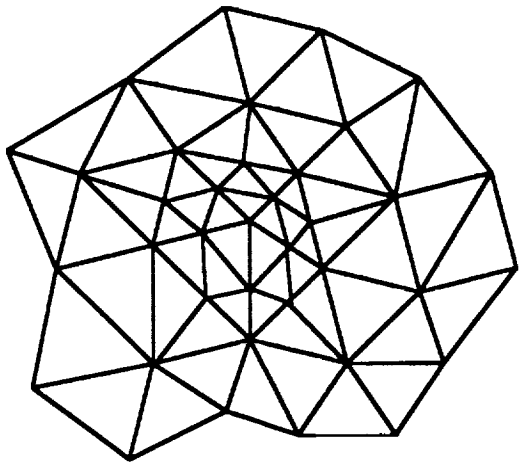


figure 4

completed refined mesh

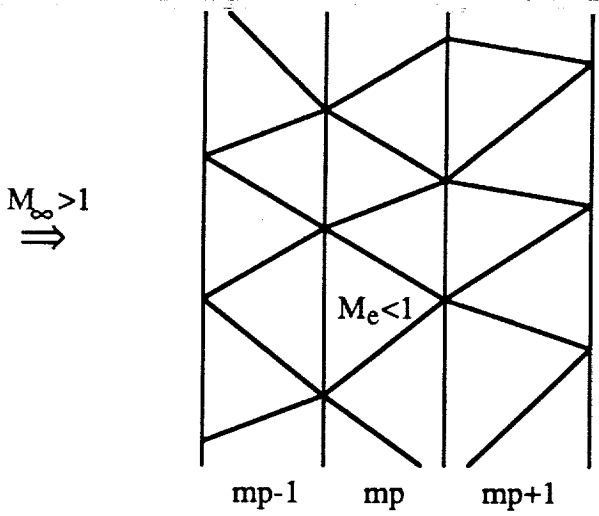


figure 5

marching planes for supersonic flow containing subsonic regions

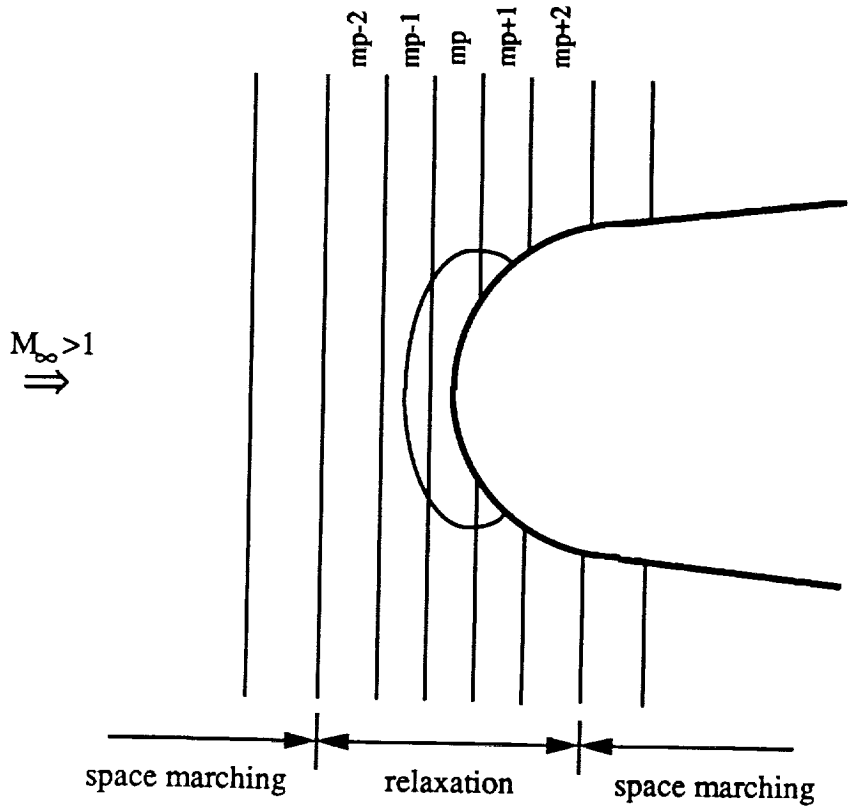


figure 6

formation of the subsonic region near the body

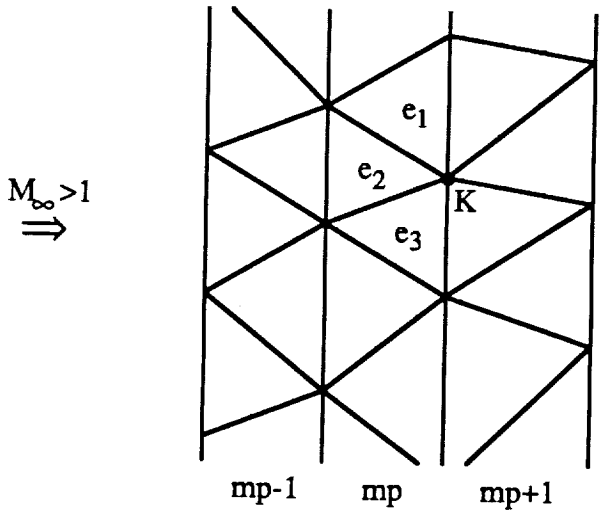
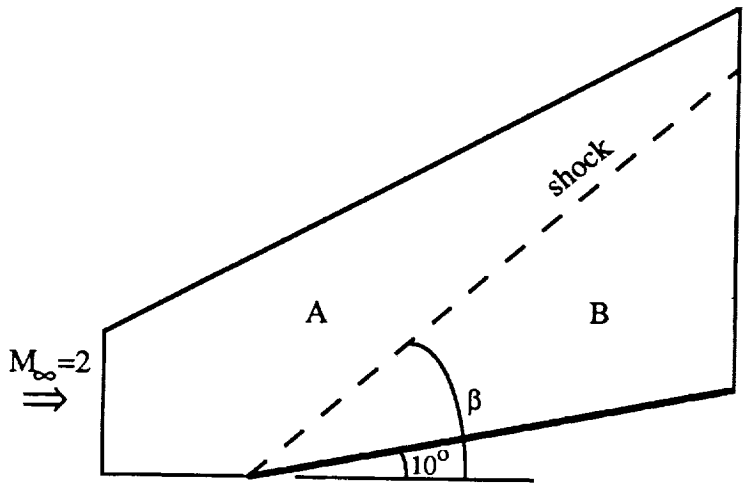


figure 7

patch of elements used to calculate the values at vertex K

for fully supersonic flow

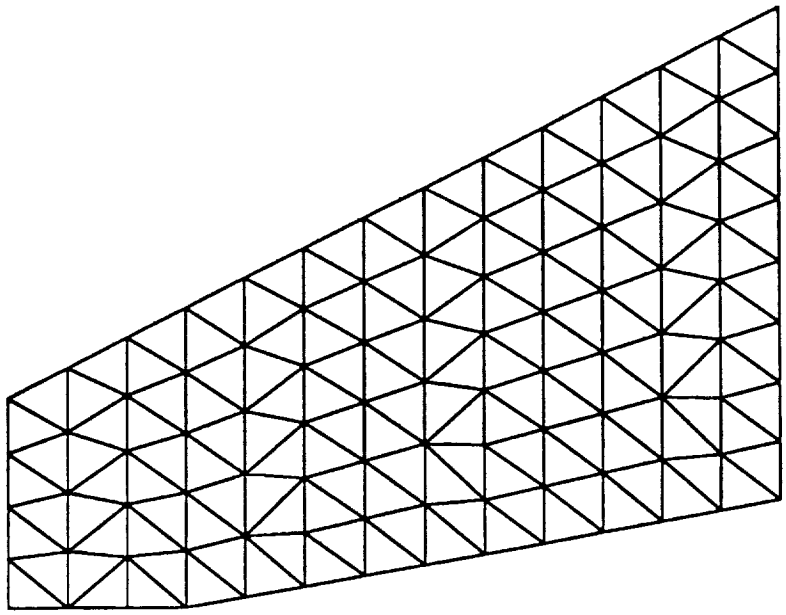


$\beta = 39.32^\circ$

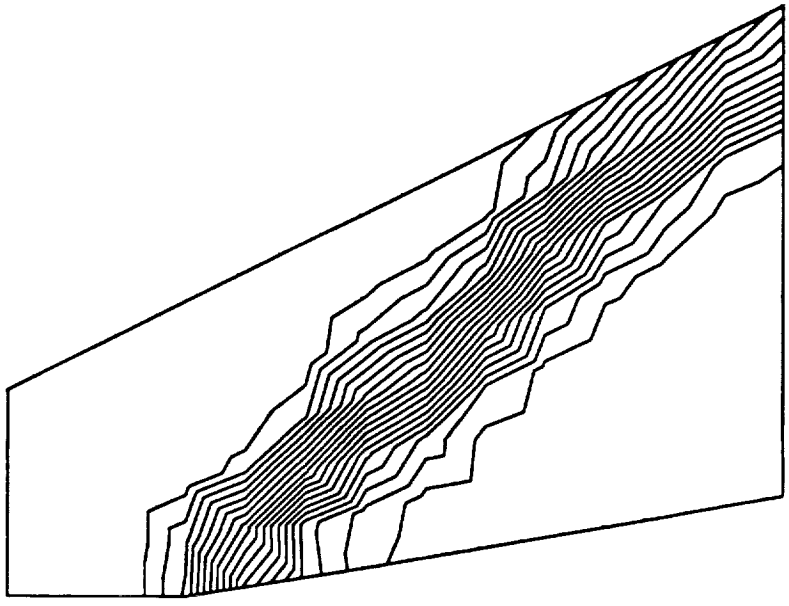
Region A	Region B
$\rho = 1.0$	$\rho = 1.4587$
$P = 1.0$	$P = 1.7071$
$M = 2.0$	$M = 1.6400$

figure 8

definition of the supersonic flow past a wedge problem



(a)

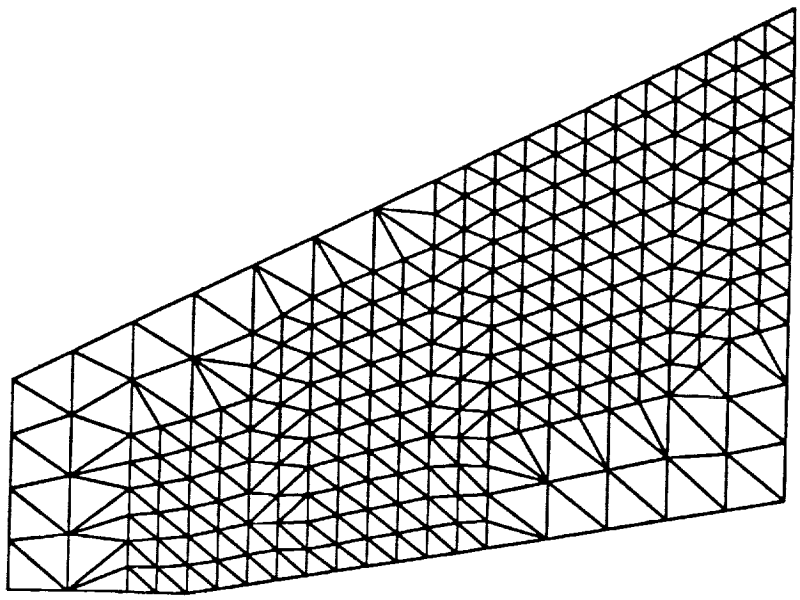


(b)

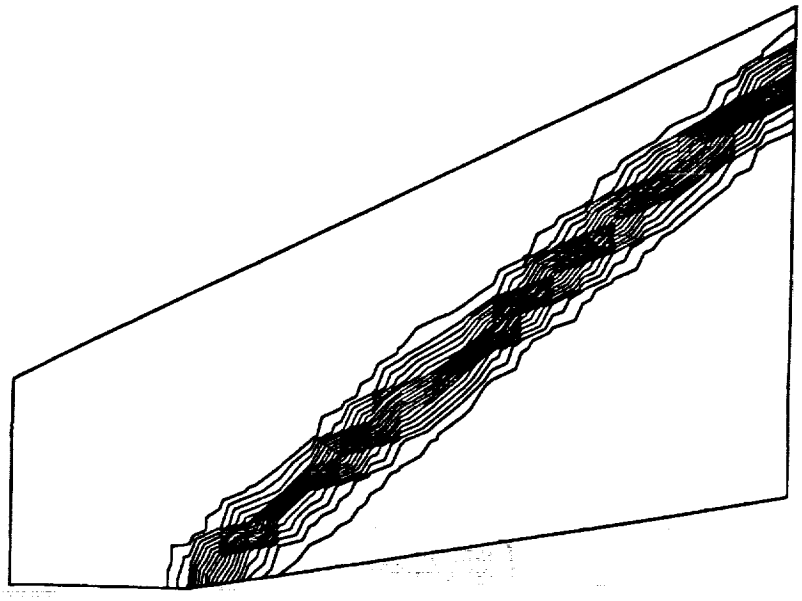
figure 9

supersonic flow past a wedge; first mesh

(a) mesh; (b) pressure coefficient contours



(a)

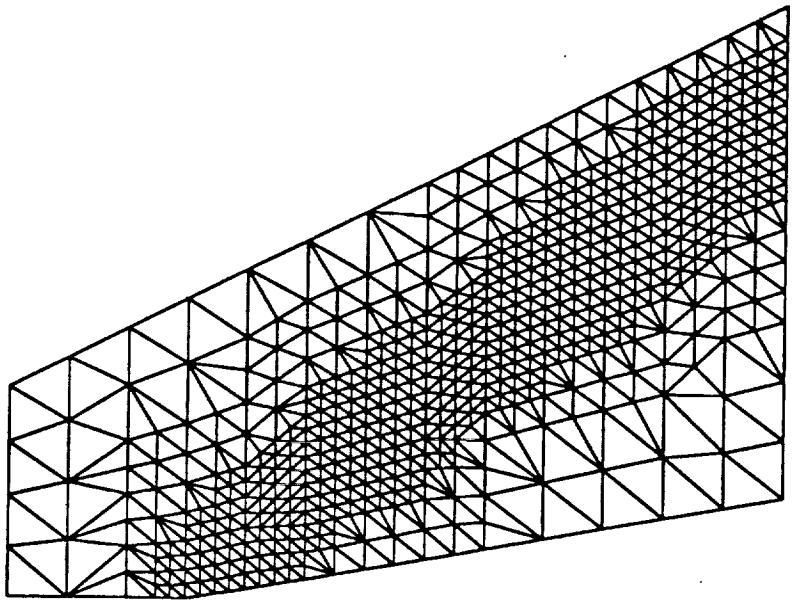


(b)

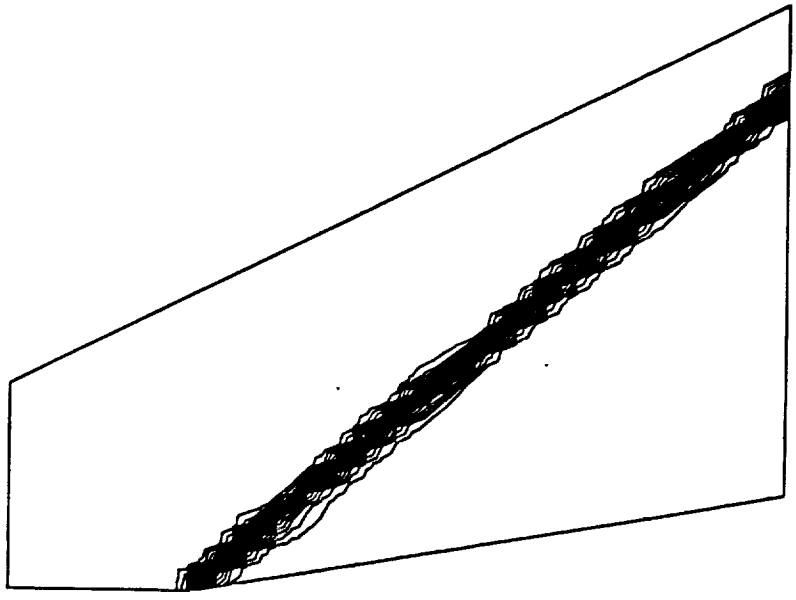
figure 10

supersonic flow past a wedge; second mesh

(a) mesh; (b) pressure coefficient contours



(a)

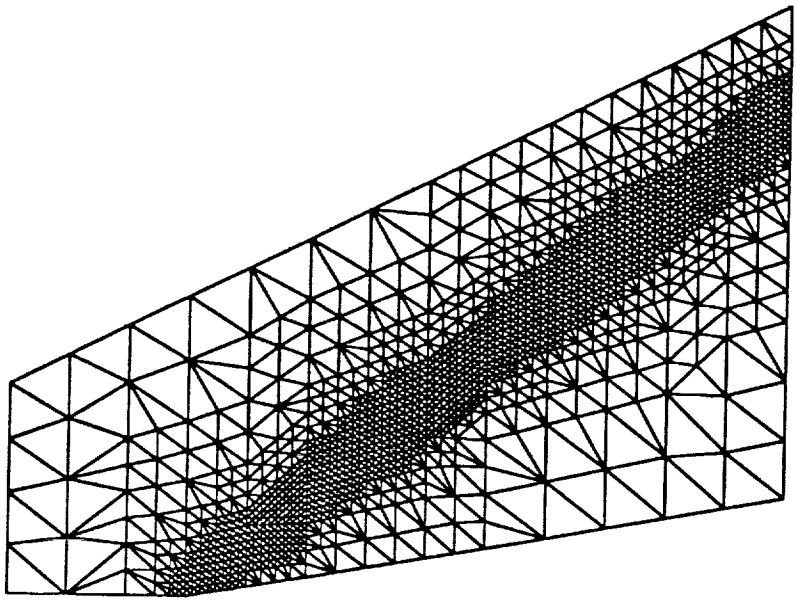


(b)

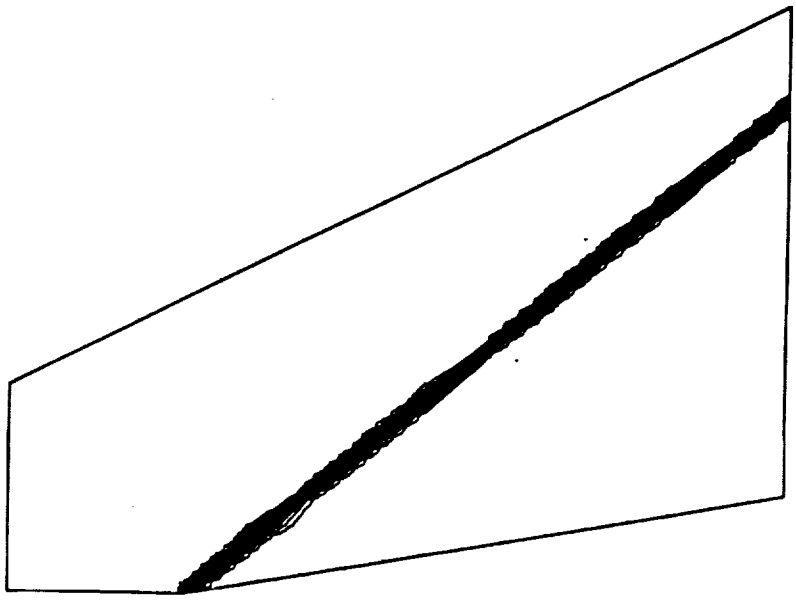
figure 11

supersonic flow past a wedge; third mesh

(a) mesh; (b) pressure coefficient contours



(a)

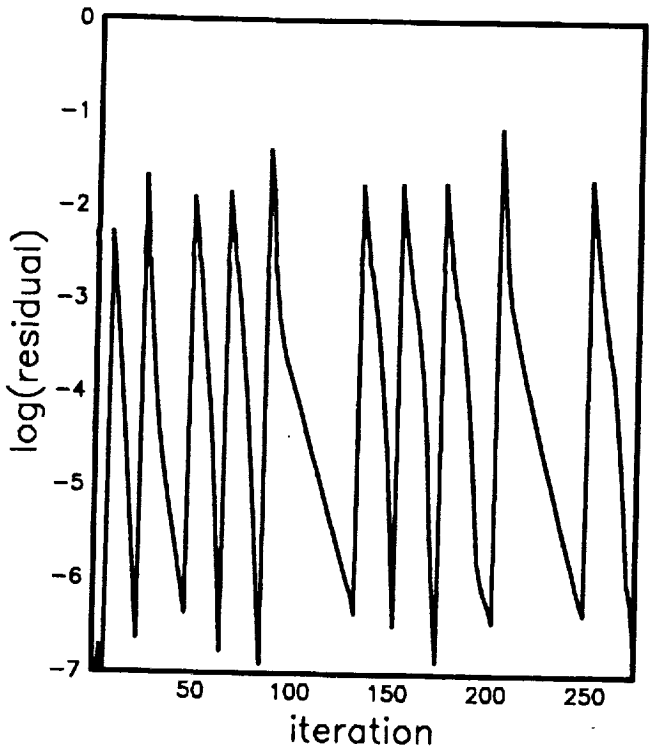


(b)

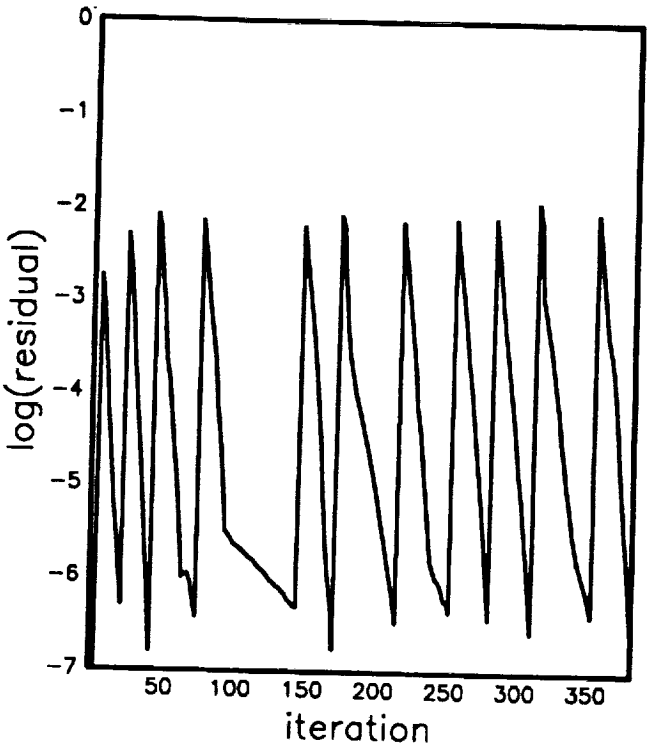
figure 12

supersonic flow past a wedge; forth mesh

(a) mesh; (b) pressure coefficient contours



(a)

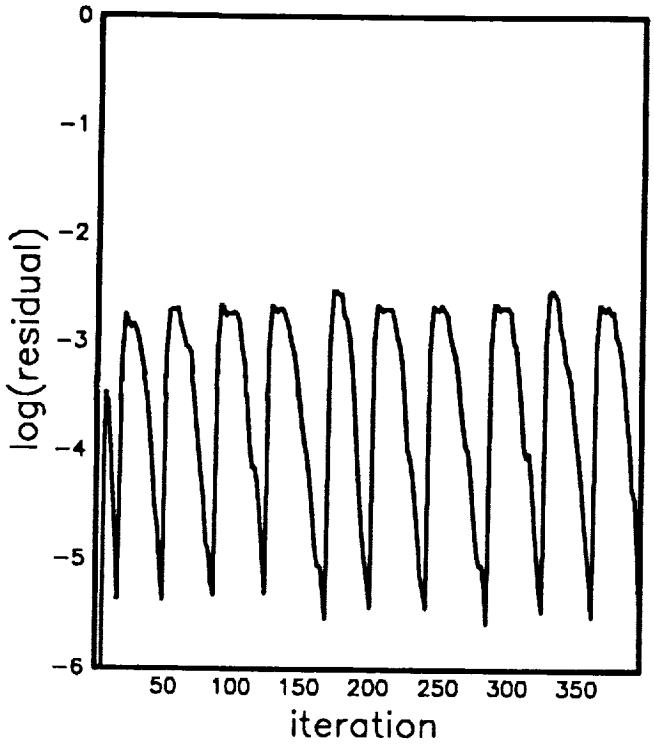


(b)

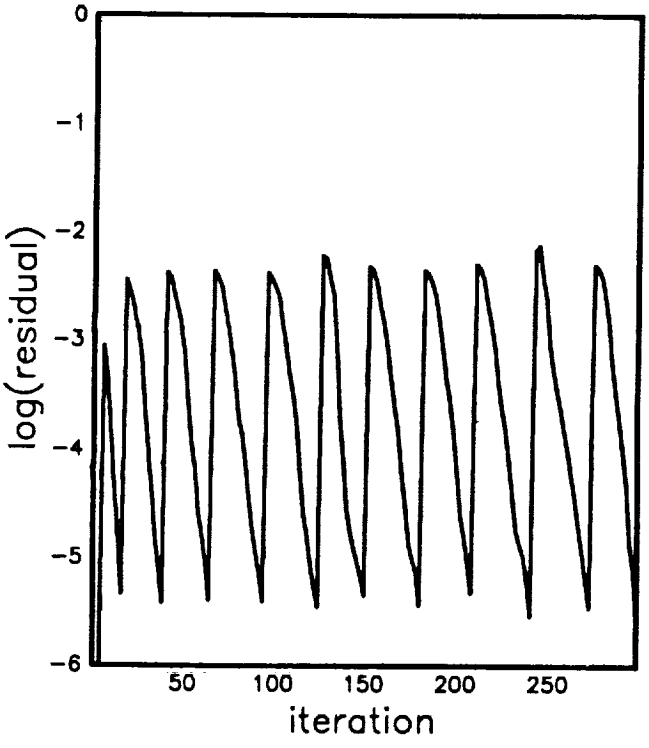
figure 13

supersonic flow past a wedge; convergence curves

(a) first mesh; (b) second mesh



(a)

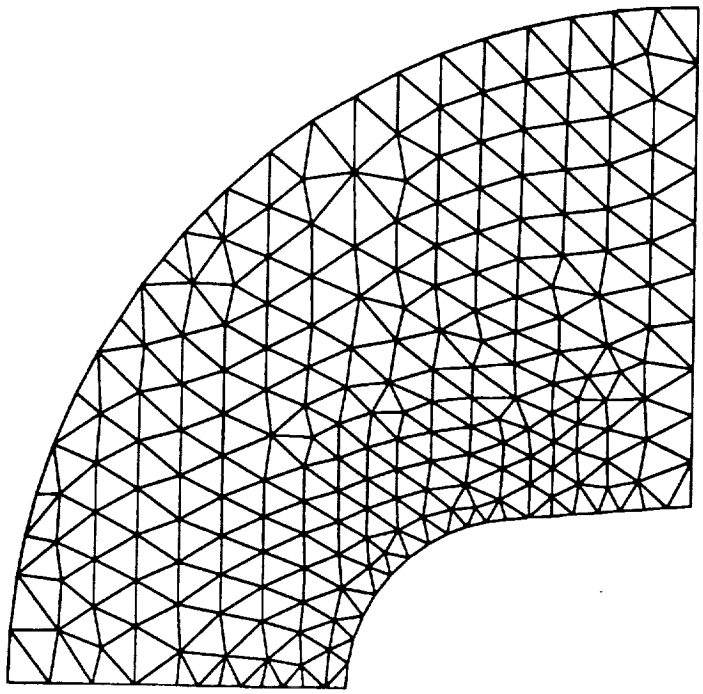


(b)

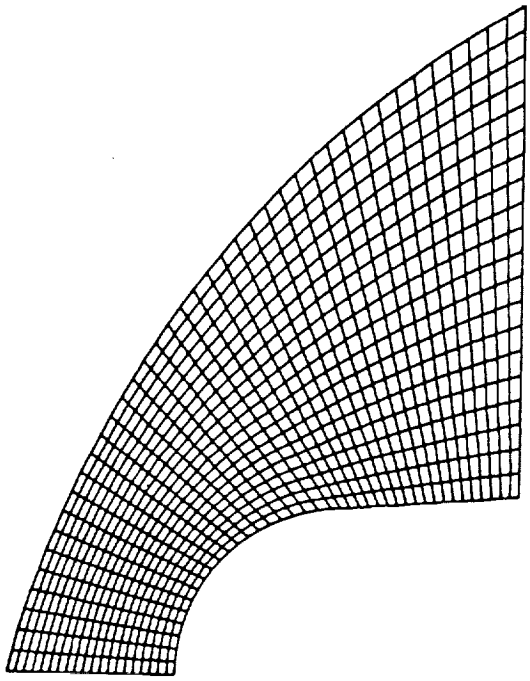
figure 14

supersonic flow past a wedge; convergence curves

(a) third mesh; (b) forth mesh



(a)

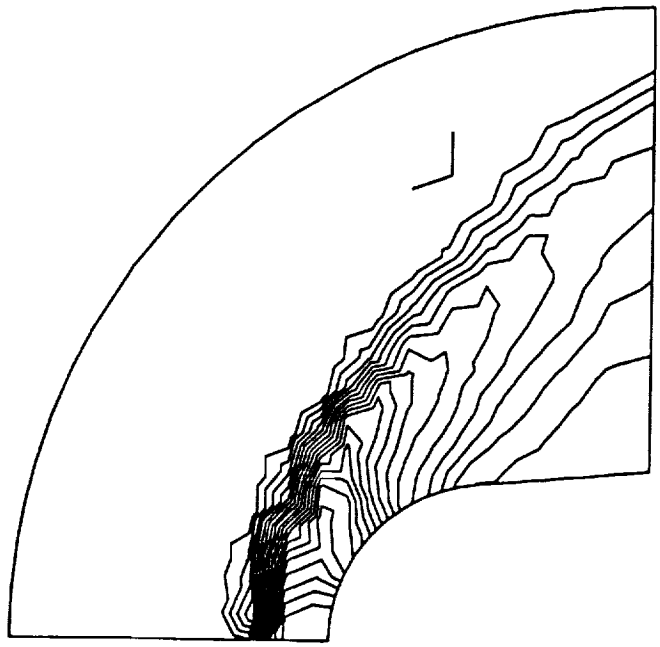


(b)

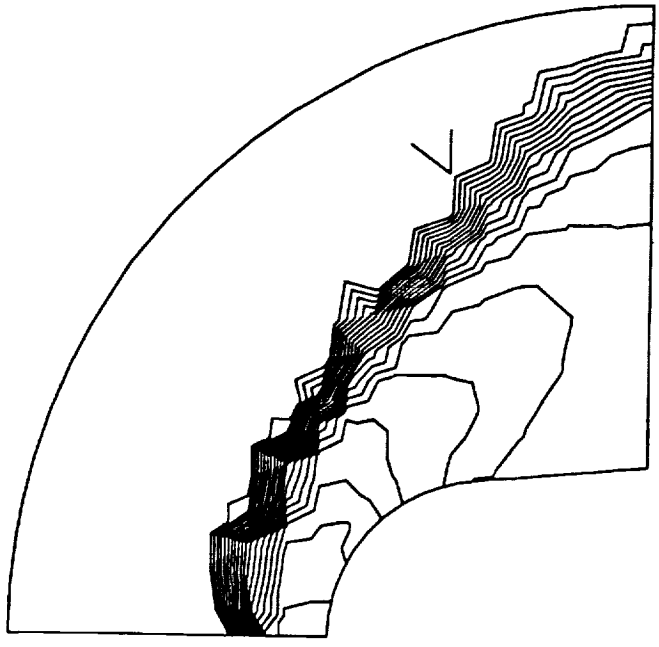
figure 15

supersonic flow past a blunt body

(a) first mesh for space marching; (b) SUPG mesh



(a)

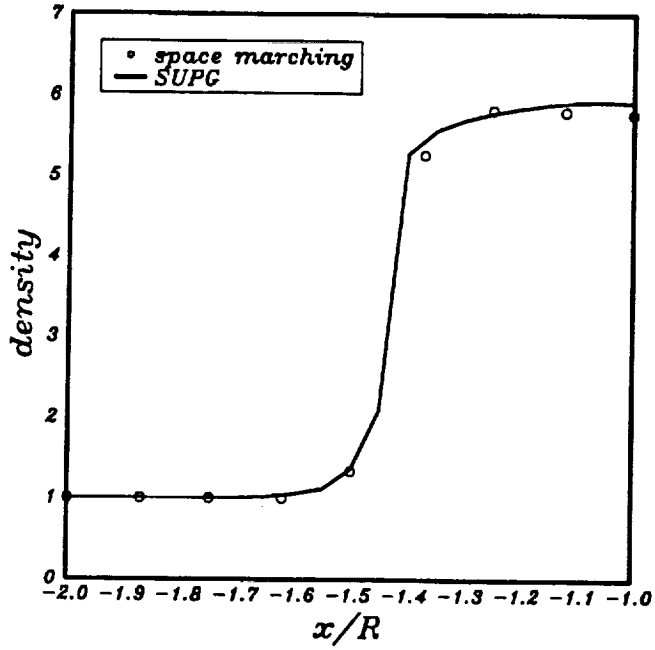


(b)

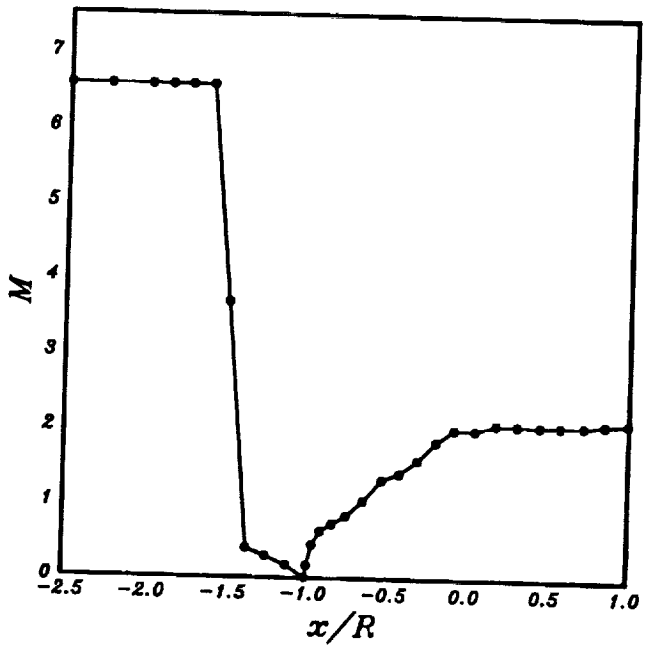
figure 16

supersonic flow past a blunt body; solution contours for the first mesh

(a) pressure coefficient; (b) Mach number



(a)



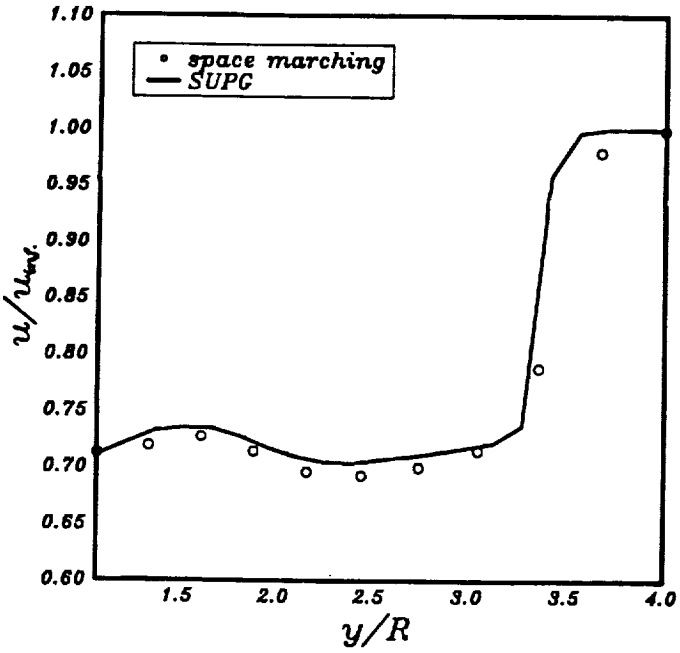
(b)

figure 17

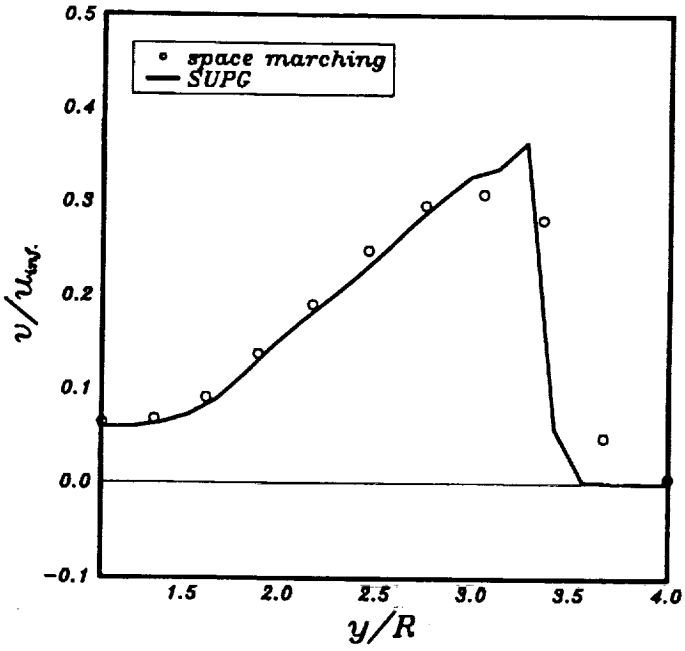
supersonic flow past a blunt body

values along the axis and over the surface; first mesh

(a) density; (b) Mach number



(a)



(b)

figure 18

supersonic flow past a blunt body

velocity components along the outflow; first mesh

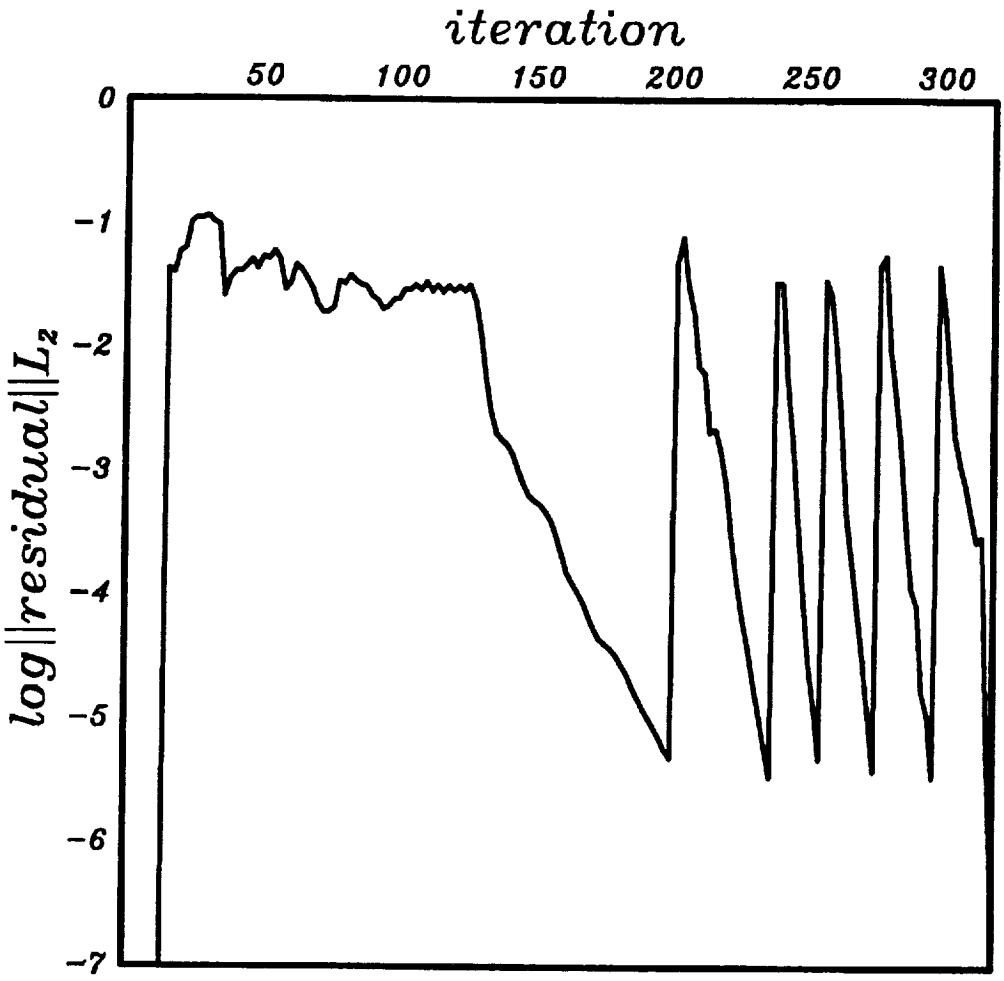
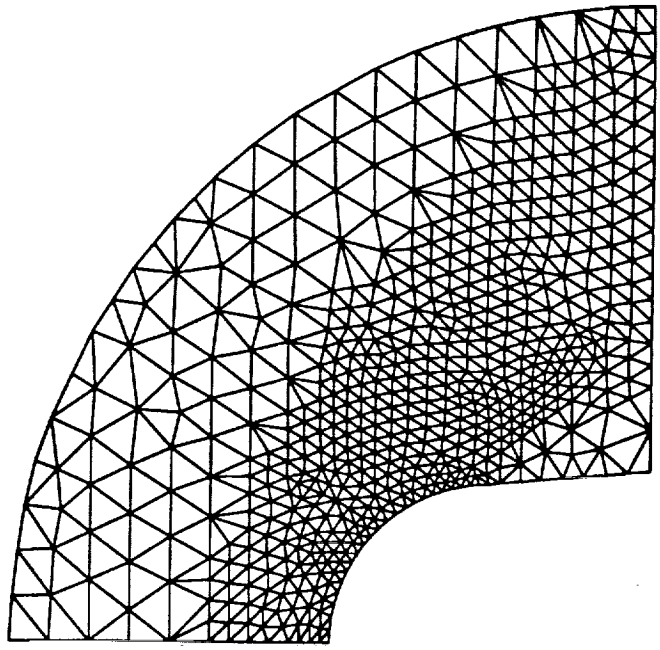


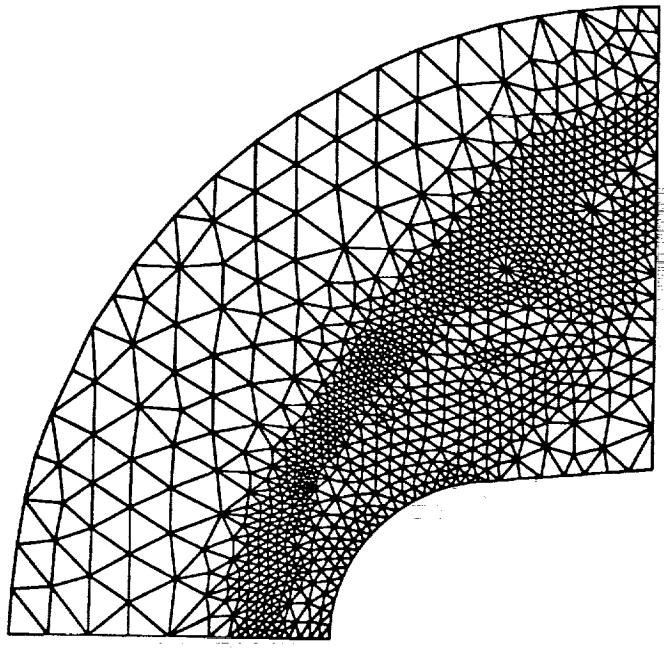
figure 19

supersonic flow past a blunt body

convergence history for the first mesh



(a)

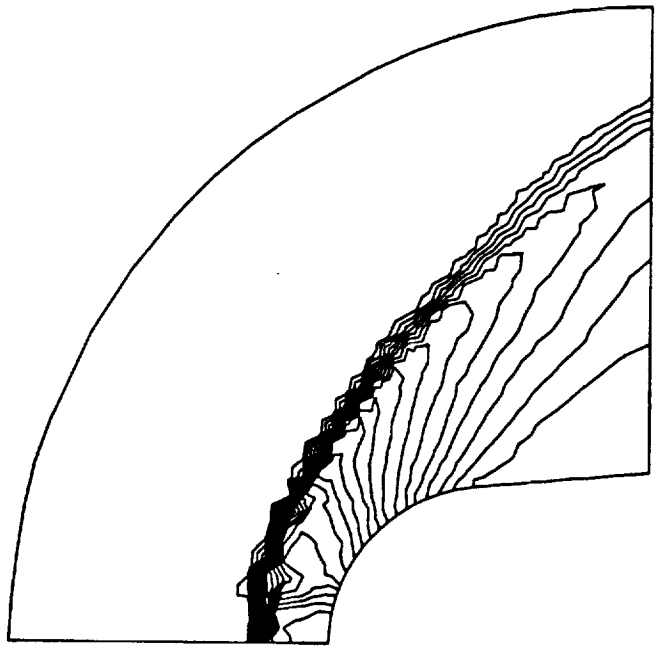


(b)

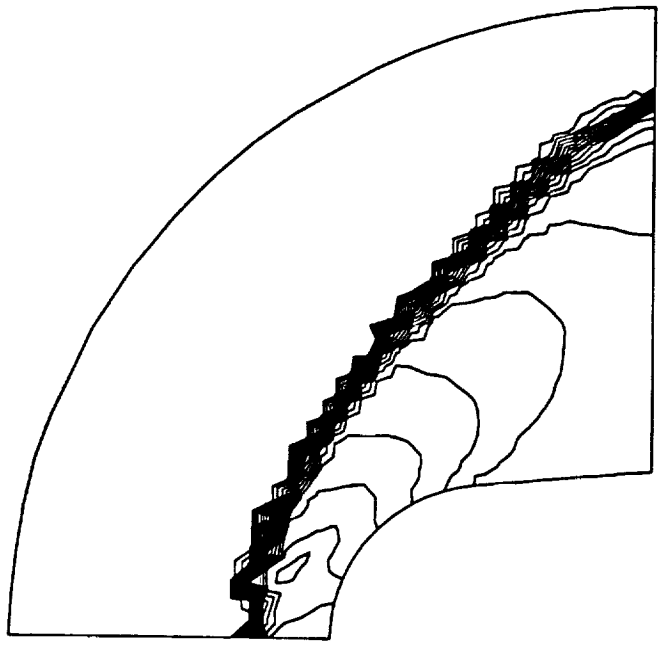
figure 20

supersonic flow past a blunt body

(a) second mesh; (b) third mesh



(a)



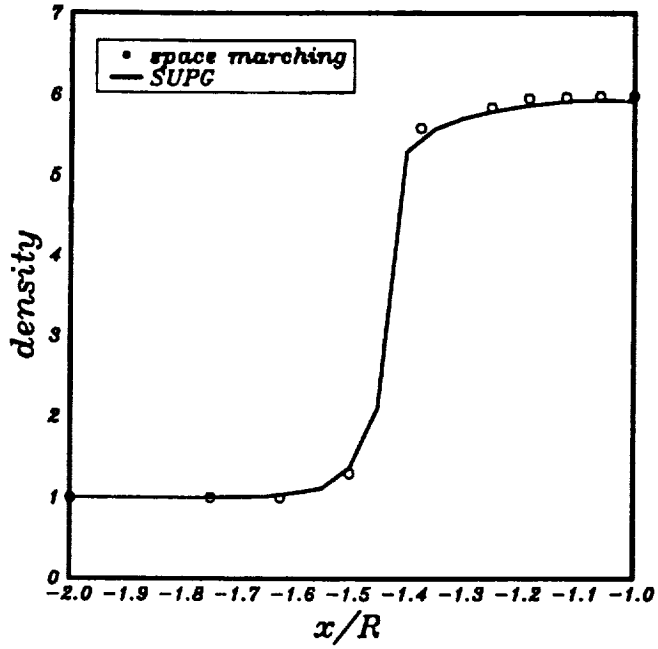
(b)

figure 21

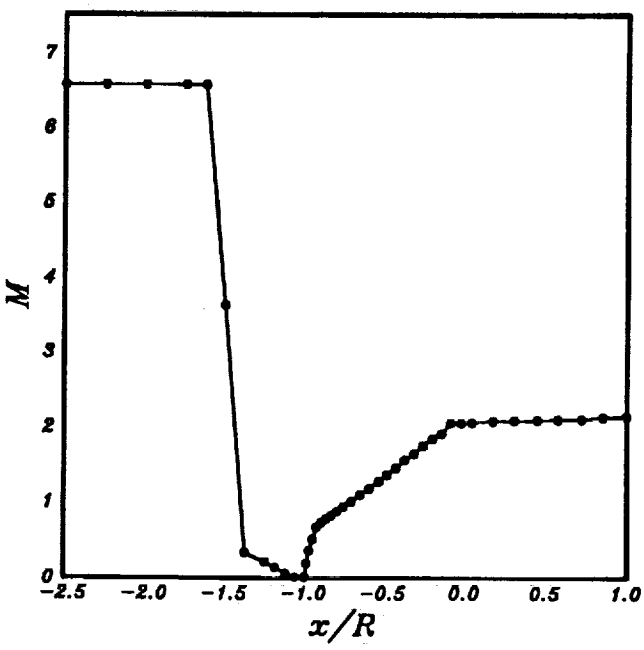
supersonic flow past a blunt body

solution contours for the second mesh

(a) pressure coefficient; (b) Mach number



(a)



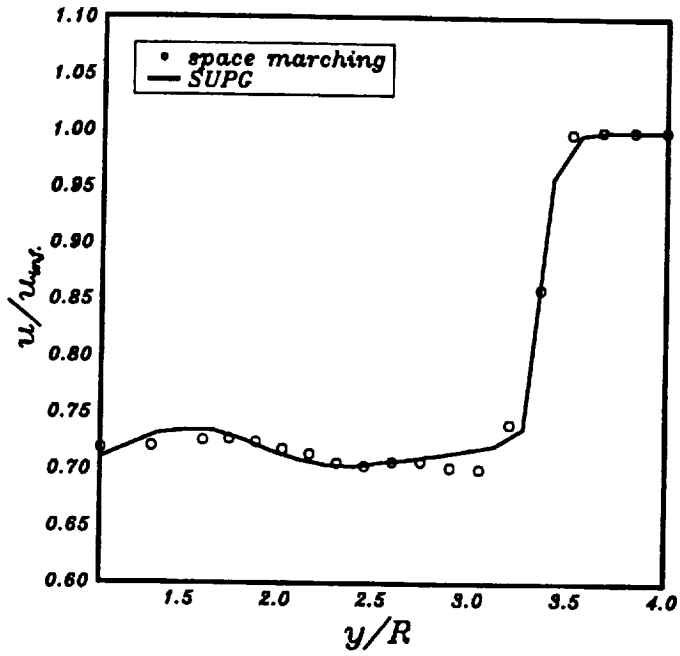
(b)

figure 22

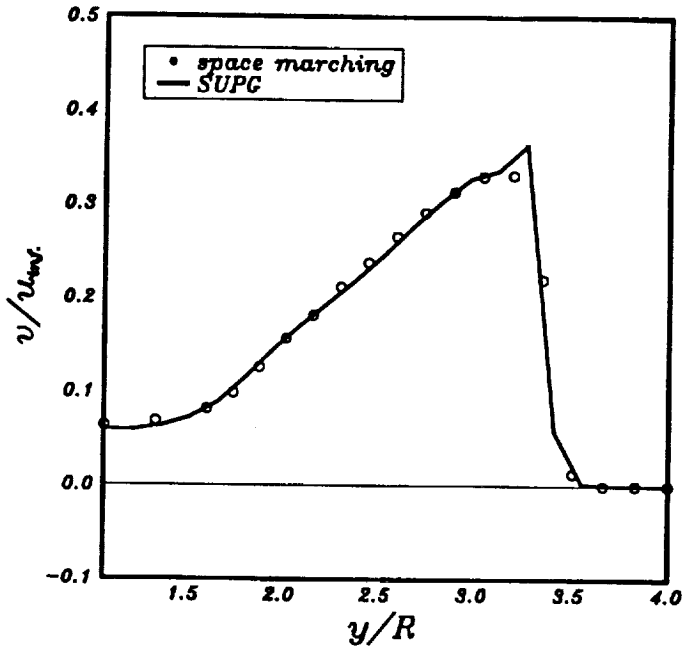
supersonic flow past a blunt body

values along the axis and over the surface; second mesh

(a) density; (b) Mach number



(a)

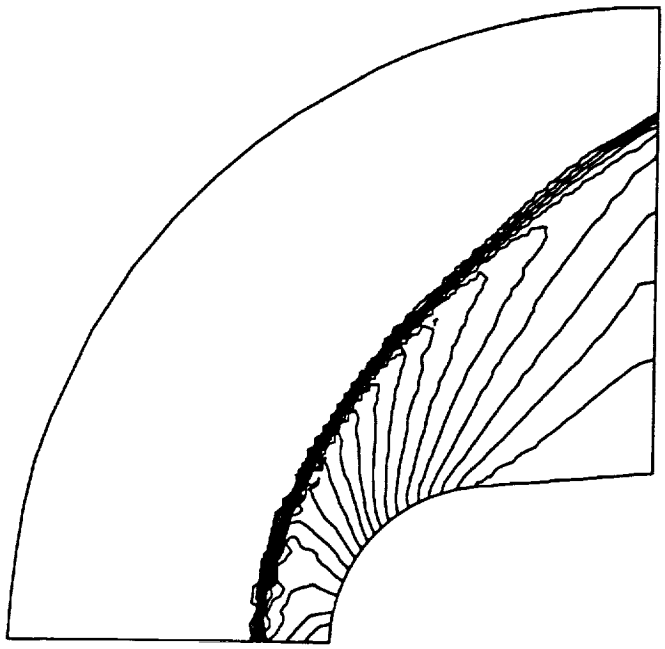


(b)

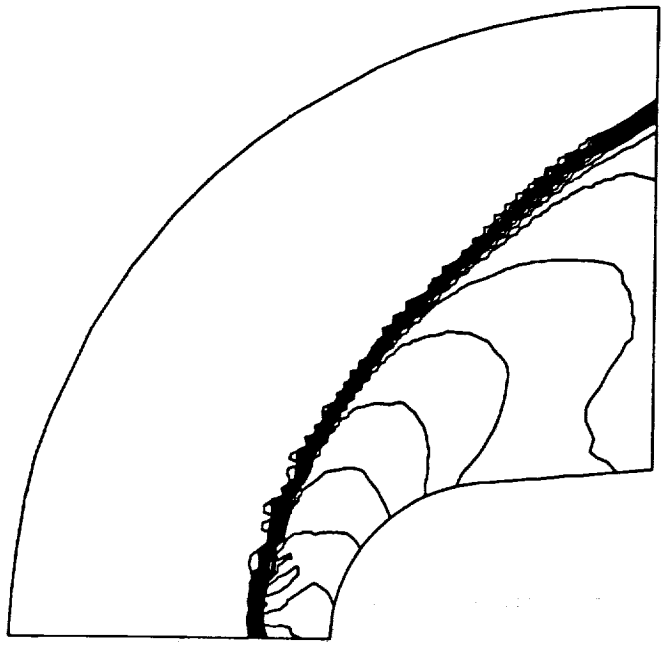
figure 23

supersonic flow past a blunt body

velocity components along the outflow for the second mesh



(a)



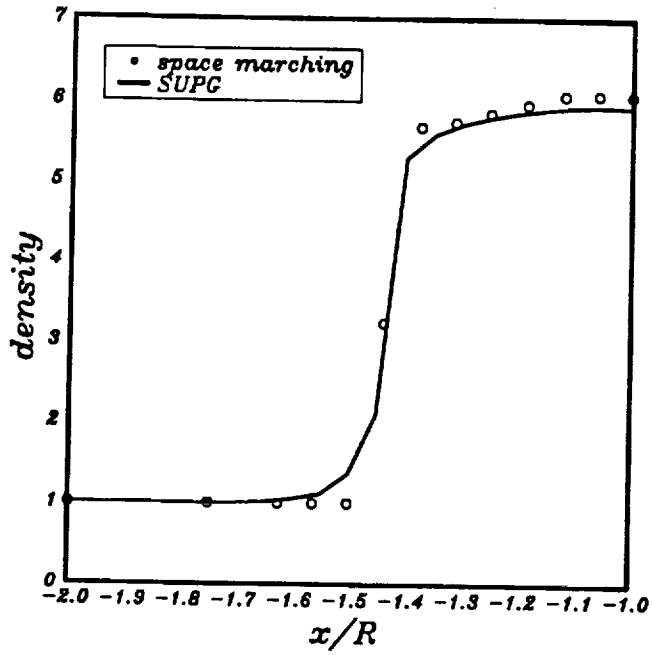
(b)

figure 24

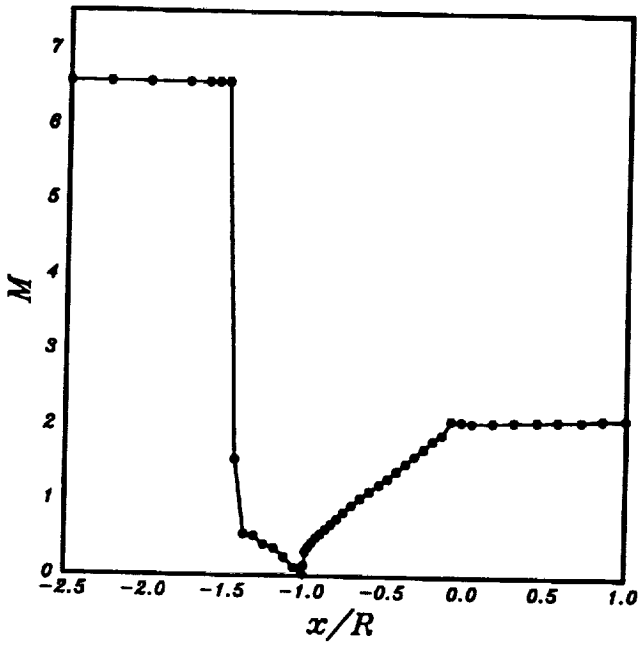
supersonic flow past a blunt body

solution contours for the third mesh

(a) pressure coefficient; (b) Mach number



(a)



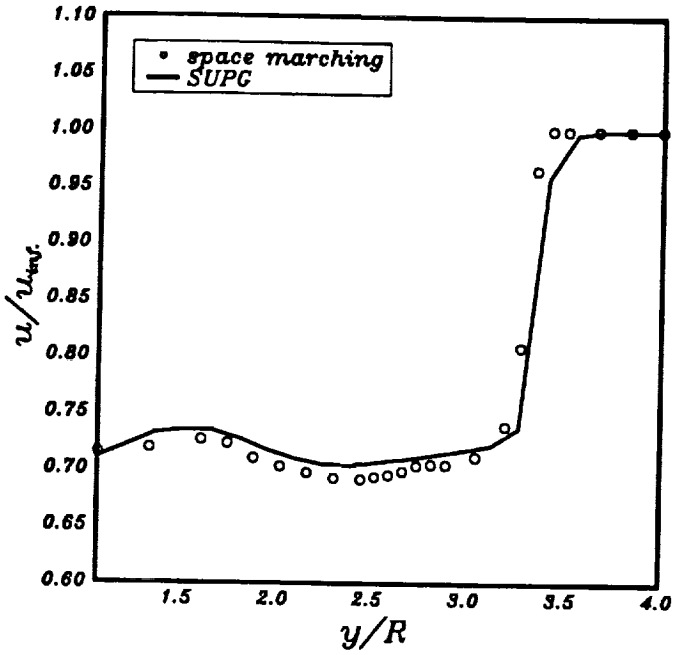
(b)

figure 25

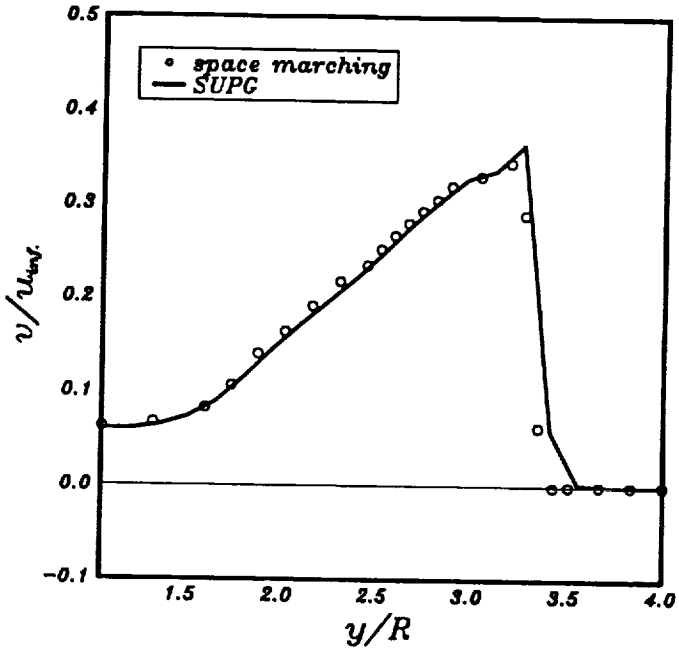
supersonic flow past a blunt body

values along the axis and over the surface; third mesh

(a) density; (b) Mach number



(a)



(b)

figure 26

supersonic flow past a blunt body

velocity components along the outflow for the third mesh

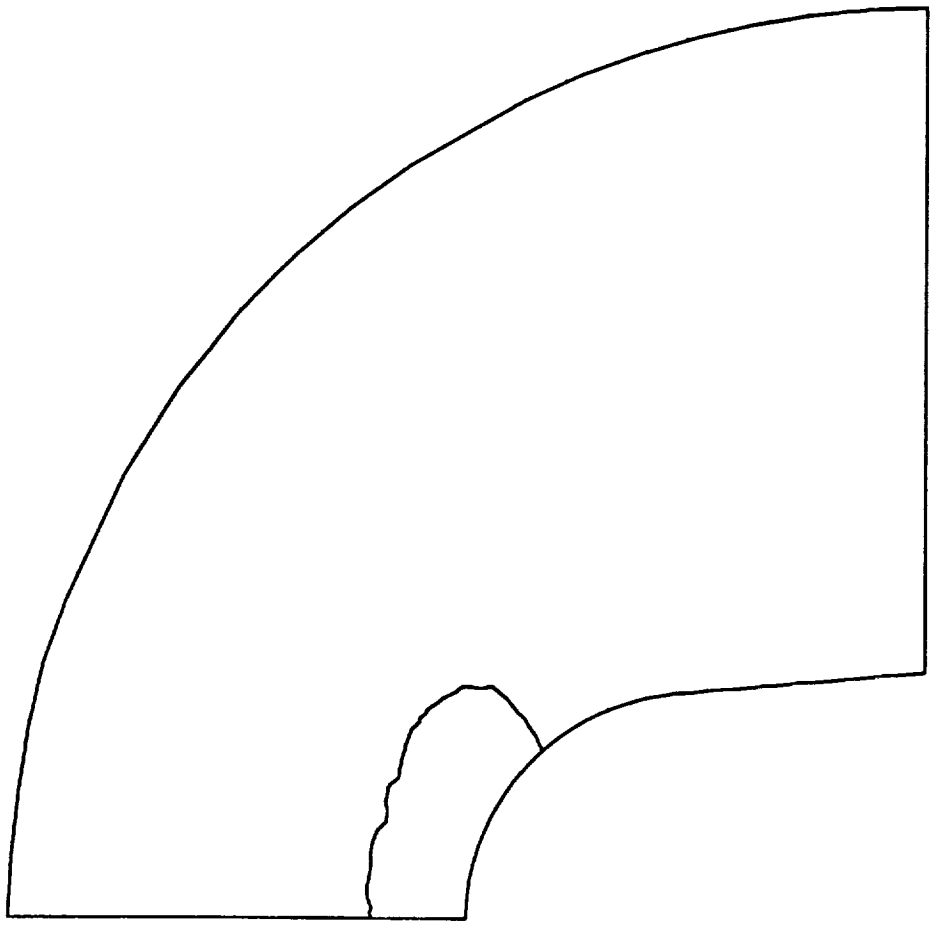
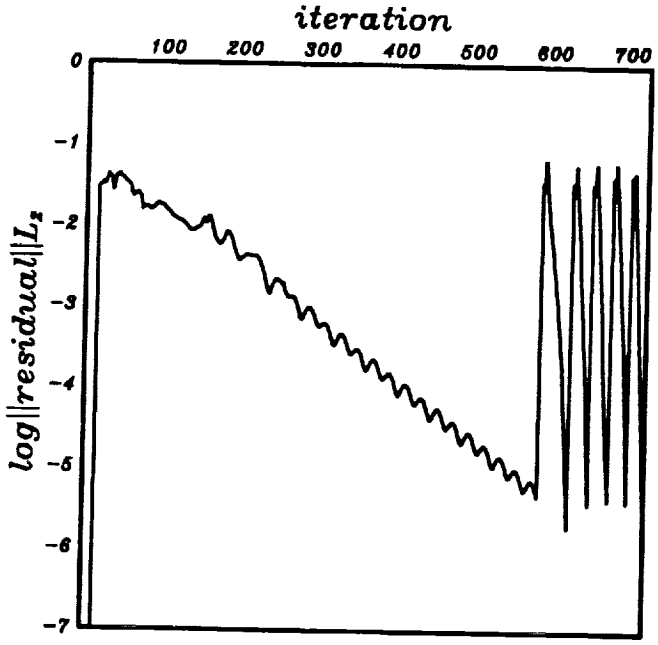


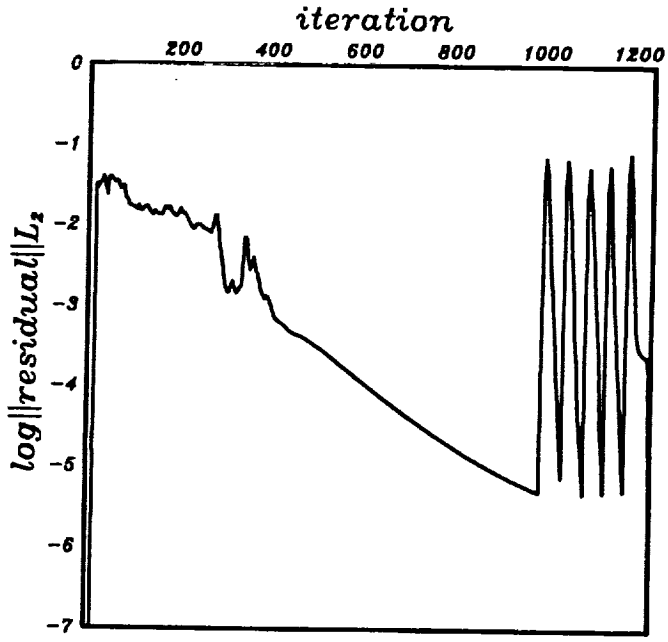
figure 27

supersonic flow past a blunt body

computed subsonic region on the third mesh



(a)



(b)

figure 28

supersonic flow past a blunt body

convergence curves for (a) second mesh; (b) third mesh