Adaptive Finite Element Technique for high-speed Compressible flows

P. Dechaumphai, P. Janphaisaeng

Chulalongkorn University
Bangkok 10330, Thailand

Abstract

An upwind cell-centered finite element formulation is combined with an adaptive meshing technique for solving high-speed compressible flow problems. The finite element formulation and its computational procedure are described. A corresponding computer program that can be executed on standard personal computers has been developed. An adaptive meshing technique is applied to increase the analysis solution accuracy, and at the same time, to minimize the computer time and memory. The efficiency of the combined method is evaluated by examples of high-speed compressible flows past a wedge and a cylinder.

1. Introduction

High-speed compressible flows past a typical geometry normally include complex flow characters, such as shock waves, flow expansions, thin boundary layers, shock-shock interactions, and shock-boundary layer interactions [1,2]. In addition, resolution of these flow characters is required to accurately predict aerodynamic pressure and skin friction distributions as well as aerodynamic heating rates on the geometry surfaces. These quantities are important and critical for the design of high-speed structural configurations. Most of these flow features are characterized by steep gradients that need robust analysis computational techniques with dense mesh refinement for high flow-behavior resolution. These requirements mean that the high-speed compressible flow analysis always involves large-scale computations with vast amounts of computer time and data storage.

The analysis computational techniques that have been developed and used in the past few decades are based on the finite difference method [3]. During the past decade, several finite element algorithms have been under development to alleviate the difficulty of the analysis computation due to complex geometry of the flow field. The algorithms include the Taylor-Galerkin algorithm [4], the Petrov-Galerkin algorithm [5], the least-squares algorithm [6], and the upwind cell-centered algorithm [7].

In this paper, the upwind cell-centered algorithm is combined with the adaptive meshing technique [8,9] to improve the finite element analysis solution accuracy and reduce the computational time as well as the required computer memory. The upwind cell-centered algorithm is selected because of its robustness in capturing shock with less formulation complexity compared to other algorithms. The adaptive meshing technique is applied to place small elements in the region of large change in the solution gradients to increase solution accuracy and, at the same time, to place larger elements in the other regions to reduce the computational time and memory. The paper starts by explaining the theoretical formulation for high-speed compressible flow analysis and the solution procedure that leads to the development of a computer program. The basic idea behind the adaptive meshing technique is then described. Finally, the combined technique is evaluated by analyzing the problems of high-speed compressible flow past a wedge and a cylinder. Results are compared with those

obtained from the conventional technique that uses nonadaptive meshes.

2. Theoretical Formulation and Solution Procedure

2.1 Governing Differential Equations

The equations for high-speed inviscid laminar compressible flow are governed by the conservation for mass, momentum, and energy. These equations in two dimensions are written in the conservation form [10] as,

$$\frac{\partial}{\partial t}\{U\}+\frac{\partial}{\partial x}\{E\}+\frac{\partial}{\partial y}\{F\} = 0 \qquad (1)$$

The vector $\{U\}$ contains the conservation variables defined by,

$$\{U\} = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ \rho \varepsilon \end{Bmatrix} \qquad (2)$$

where ρ is the fluid density, u and v are the velocity components, and ε is the total energy, The vectors $\{E\}$ and $\{F\}$ consist of the inviscid fluxes in the x and y directions, respectively. These inviscid flux vectors are given by,

$$\{E\} = \begin{Bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho u\varepsilon + pu \end{Bmatrix}; \{F\} = \begin{Bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho v\varepsilon + pv \end{Bmatrix} \qquad (3)$$

where p is the pressure. The total energy consists of the internal energy and the kinetic energy defined by,

$$\varepsilon = e + \frac{1}{2}(u^2+v^2) \qquad (4)$$

where e is the internal energy that can be written in the form,

$$e = c_v T \qquad (5)$$

where c_v is the specific heat at constant volume, and T is the temperature.

2.2 Analysis Solution Algorithm

The explicit upwind cell-centered finite element algorithm is applied to solve the Euler Eqs. (1-3). The basic concept behind the algorithm is to determine the flux across element interfaces using Roe's averaging procedure [7]. The average inviscid flux $\overline{G}$ (The "overbar" denotes average quantity normal to the element interface between the left element L and the right element R) is given by,

$$\overline{G} = \frac{1}{2}\left[G_L + G_R + \left|A^*\right|(U_L - U_R)\right] \qquad (6)$$

The last term in Eq. (6) may be viewed as artificial diffusion needed for solution stability. This diffusion is represented by the product of the Jacobian matrix $\left[A^*\right]$ and the difference between the left and right element conservation variables U_L and U_R. The Jacobian matrix $\left[A^*\right]$ is defined by,

$$\left[A^*\right] = [R]^{-1}\,[|\Lambda|]\,[R] \qquad (7)$$

where

$$[R]^{-1} = \begin{bmatrix} -\frac{1}{c^2} & 0 & \frac{1}{2c^2} & \frac{1}{2c^2} \\ -\frac{u}{c^2} & -m & \frac{u+c_x}{2c^2} & \frac{u-c_x}{2c^2} \\ -\frac{v}{c^2} & \ell & \frac{v+c_y}{2c^2} & \frac{v-c_y}{2c^2} \\ -\frac{\alpha}{c^2} & V & \frac{\alpha+Uc}{2c^2}+\frac{1}{2\beta} & \frac{\alpha-Uc}{2c^2}+\frac{1}{2\beta} \end{bmatrix} \qquad (8)$$

$$[|\Lambda|] = \begin{bmatrix} |U| & 0 & 0 & 0 \\ 0 & |U| & 0 & 0 \\ 0 & 0 & |U+c| & 0 \\ 0 & 0 & 0 & |U-c| \end{bmatrix} \qquad (9)$$

$$[R] = \begin{bmatrix} \alpha\beta - c^2 & -\beta u & -\beta v & \beta \\ -V & -m & \ell & 0 \\ \alpha\beta - Uc & c_x - \beta u & c_y - \beta v & \beta \\ \alpha\beta + Uc & -c_x - \beta u & -c_y - \beta v & \beta \end{bmatrix} \qquad (10)$$

In these matrices,

$U = u\ell + vm \quad ; \quad V = -um + v\ell$

$c_x = c\ell \quad ; \quad c_y = cm \quad ; \quad c^2 = \gamma p/\rho$

$\alpha = (u^2 + v^2)/2 \; ; \quad \beta = \gamma - 1$

where ℓ and m are the direction cosines of the unit vector normal to that side, and γ is the specific heat ratio.

Increments of the conservation variables, $\Delta U = U^{n+1} - U^n$, where n is the n^{th} time step, are determined from,

$$\left[[I] + \frac{\Delta t}{A}\sum\left[\left|A^*\right|\right]\frac{S}{2}\right]\{\Delta U\} = \frac{-\Delta t}{A}\sum S\{\overline{G}\} \tag{11}$$

where [I] is the identity matrix, Δt is the time step, A is the element area, S is the length of element side where summation is performed for all sides.

2.3 Computational Procedure and Computer Program

The element Eq. (11) consists of four equations for solutions of the conservation variables. These equations are solved iteratively for each element by the Gauss-Seidel technique [11]. Thus the algorithm requires relatively low computer memory even for a finite element model with a large number of elements. This Eq. (11) is used in the development of a computer program for analysis of high-speed inviscid compressible flows. The program is written in FORTRAN and has been developed for running on standard personal computers. The program employs explicit time integration technique by requiring initial conditions of the conservation variables for the entire flow field at the beginning of the computation. With proper time steps, converged solution is obtained when the change of the conservation variables for all elements is less than the specified tolerance or the specified total number of time steps. This program is used together with the adaptive meshing technique which is described in the next section. The efficiency of the combined procedure is evaluated by the analysis of high-speed compressible flows past a wedge and a cylinder which are presented in the example section.

3. Adaptive Meshing Technique

Adaptive mesh-generation techniques may be classified into two major categories: 1) refinement/derefinement, and 2) remeshing. The first category, the adaptive refinement/ derefinement technique, can be further classified into three subcategories: a) the h method, b) the p method, and c) the r method. In the h method, the elements in the initial mesh are refined into smaller elements or derefined into larger elements [12]. The p method maintains the geometry of the elements of the initial mesh but increases (or decreases) the order of the polynomials used for the element interpolation functions [13]. The r method keeps the number of elements and their connectivities the same but relocates the nodes [14].

The remeshing technique, the second adaptive mesh-generation category, generates an entirely new mesh based on the solution obtained from an earlier mesh [8,9]. The technique is combined with the upwind cell-centered finite element formulation in this paper to solve high-speed compressible flow problems. The idea is to construct a new mesh that consists of small elements in the regions with large change in solution gradients and large elements in the other regions where the change in solution gradients is small. As an example, small elements are needed in the regions of shock waves to capture shock resolution, whereas larger elements can be used in the free-stream regions because the flow behavior is uniform. To determine proper element sizes at different locations in the flow field, the solid-mechanics concept of determining the principal stresses from a given state of stresses at a point is employed. Since the fluid density changes abruptly across the shock waves, the density distribution can be used as an indicator in the determination of proper element sizes.

Because small elements must be placed in the region of the shock wave where large changes in density gradients occur, the second derivatives of the density at a point with respect to global coordinates x and y are needed to compute,

$$\begin{bmatrix} \frac{\partial^2 p}{\partial x^2} & \frac{\partial^2 p}{\partial x \partial y} \\ \frac{\partial^2 p}{\partial x \partial y} & \frac{\partial^2 p}{\partial y^2} \end{bmatrix} \quad (12)$$

Then the principal quantities in the principal directions X and Y where the cross derivatives vanish, are determined,

$$\begin{bmatrix} \frac{\partial^2 \rho}{\partial X^2} & 0 \\ 0 & \frac{\partial^2 \rho}{\partial Y^2} \end{bmatrix} \quad (13)$$

The magnitude of the larger principal quantity is then selected,

$$\lambda = \max \left(\left| \frac{\partial^2 p}{\partial X^2} \right|, \left| \frac{\partial^2 p}{\partial Y^2} \right| \right) \quad (14)$$

This value is used to compute proper element size h at that location from the condition [15],

$$h^2 \lambda = \text{constant} = h_{min}^2 \lambda_{max} \quad (15)$$

where h_{min} is the specified minimum element size, and λ_{max} is the maximum principal quantity for the entire model.

Based on the condition shown in Eq. (15), the element size is generated according to the given minimum element size h_{min}. Specifying too small h_{min} may result in a model with an excessive number of elements. On the other hand, specifying too large h_{min} may result in inadequate solution accuracy or excessive analysis and remeshing cycles. These factors must be considered prior to generating a new mesh. Note that, because the technique generates an entirely new mesh with different nodal locations from the old mesh, interpolation of the solution from the old to the new mesh should be used to increase the analysis solution convergence.

4. Examples

Two examples are presented to evaluate the capability of the upwind cell-centered finite element formulation combining with the adaptive remeshing technique for analysis of high-speed compressible flows. The two examples are a Mach 3 flow over a wedge and a Mach 6 flow over a cylinder. All computations were made on standard personal computer with Pentium 166 MHz processor.

4.1 Mach 3 Flow Past a Wedge

The problem statement of a Mach 3 flow past a wedge is described by Fig. 1. The flow enters through the left boundary of the computational domain and creates an oblique shock wave as highlighted in the figure. Figure 2(a) shows a typical standard finite element model that consists of structured mesh with 400 triangular elements and 231 nodes. With this finite element model, the computer program developed was used to computate the solutions. A typical solution of density distribution in the form of the contour lines is presented in Fig. 2 (b). The figure shows an oblique shock generated from the corner of the inclined surface. The computed oblique shock is not sharp because the elements along the shock line are not small enough. Such a solution suggests the need of small elements clustered along the shock line to provide better shock resolution.

Small elements along the shock line for improving shock resolution can be generated by using the adaptive meshing technique described in the preceding section. The technique starts from generating a relatively uniform mesh such as that shown in Fig. 3(a) with 344 elements and 201 nodes. With this first mesh, the fluid analysis is performed to obtain the corresponding flow solution as illustrated by the density contours in Fig. 3(b). Based on this flow solution and the use of the adaptive meshing technique, the second mesh is created as shown in Fig. 4(a). This second mesh consists of 669 elements and 357 nodes with smaller elements clustered along the shock line and larger elements in the free stream region. The fluid analysis is then repeated to yield a flow solution as illustrated by the density contours in Fig. 4(b). The entire process is repeated again to generate a third adaptive mesh with 970 elements and 506 nodes and the corresponding flow solution of density contours as shown in Figs. 5(a) and 5(b), respectively. The final flow solution as shown in Fig. 5(b) indicates good solution accuracy for oblique shock because small elements are generated

along the shock line. And since larger elements are used in the other regions, thus the total CPU time for the fluid analysis is minimized. Each analysis requires approximately 1,200 time steps for a converged solution. Table 1 compares the computational times, in seconds on a standard personal computer with Pentium 166 MHz processor, needed for the finite models in Figs. 2-5. Figure 6 shows the improvement of the shock resolution (along section A-A in Fig. 1) from the third adaptive mesh by comparing with the exact solution and the nonadaptive mesh solution.

Table 1. Comparative CPU times needed by nonadaptive and adaptive finite element meshes for flow past a wedge.

Mesh	No. of elements	CPU time (sec)
Nonadaptive (Fig. 2)	400	54
1st Adaptive (Fig. 3)	344	48
2nd Adaptive (Fig. 4)	669	86
3rd Adaptive (Fig. 5)	970	122

4.2 Mach 6 Flow Past a Cylinder

To further evaluate the upwind cell-centered finite element formulation combining with the adaptive remeshing technique, the analysis of a Mach 6 flow past a cylinder is performed. The problem statement and the sketch of the flow behavior are shown in Fig. 7. The flow behavior is more complex than that in the previous example. The flow field includes a bow shock detached in front of the cylinder. The flow is subsonic near the centerline between the bow shock and the cylinder. Away from this region, the flow behind the weaker part of the bow shock becomes supersonic with variation of all flow variables.

The analysis is first performed by using a nonadaptive finite element mesh as shown in Fig. 8(a). The mesh consists of 400 elements and 231 nodes. The corresponding predicted density distribution represented by contour lines is presented in Fig. 8(b). The figure shows that the computed bow shock is not sharp because the element sizes are too large along the shock line. However, the predicted flow density behind the bow shock is reasonably good because the flow field changes gradually. The solution from this initial study using nonadaptive mesh thus suggests the need of clustered small elements along the bow shock for better shock resolution. It should be noted that, in general, the shape and location of the bow shock are not known to the analyst a priori. The use of the adaptive remeshing technique that will be explained next can be used to overcome such difficulty.

The combined finite element analysis and adaptive meshing technique starts from generating a relatively uniform mesh such as that shown in Fig. 9(a) with 572 elements and 325 nodes. The fluid analysis is performed to yield the corresponding flow solution shown by the density contours in Fig. 9(b). Based on this flow solution, the adaptive remeshing technique is then used to generate the second adaptive mesh consisting of 548 elements and 299 nodes as shown in Fig. 10(a). The fluid analysis is then performed to yield the corresponding flow solution such as the density contours shown in Fig. 10(b). The process is repeated again to generate the third mesh as shown in Fig. 11(a) with 956 elements and 501 nodes. Small elements are clustered along the bow shock as can be seen clearly in this figure. After the fluid analysis is performed using this third mesh, good quality of flow resolution with sharp bow shock is obtained as shown in Fig. 11(b). Comparison of the computational times required for converged solutions using 6,000 time steps on a standard personal computer with Pentium 166 MHz processor is shown in Table 2 for the nonadaptive and adaptive meshes in Figs. 8-11. Figure 12 also highlights the improved shock resolution along the flow centerline obtained from the third adaptive mesh comparing with the exact solution and the nonadaptive mesh solution. This example demonstrates the capability of the combined finite element method and the adaptive remeshing technique that can provide improved flow solution accuracy for general flow behavior that is not known a priori.

Table 2. Comparative CPU times needed by nonadaptive and adaptive meshes for flow past a cylinder.

Mesh	No. of elements	CPU time (sec)
Nonadaptive (Fig. 8)	400	270
1st Adaptive (Fig. 9)	572	378
2nd Adaptive (Fig. 10)	548	363
3rd Adaptive (Fig. 11)	956	605

5. Concluding Remarks

A finite element method based on an upwind cell-centered algorithm for analysis of high-speed inviscid compressible flow is presented. The method is combined with an adaptive remeshing technique to improve the flow solution accuracy. The technique generates an entirely new mesh based on the solution obtained from a previous mesh. The new mesh consists of clustered elements in the regions with large change in solution gradients to provide higher solution accuracy. At the same time, larger elements are generated in the other regions to reduce the computational time and required computer memory. The finite element formulation was presented and the basic idea behind the adaptive meshing technique was described.

Two examples of high-speed compressible flows were presented to assess the effectiveness of the combined method. These examples are the Mach 3 flow past a wedge and the Mach 6 flow past a cylinder. All computations were performed on standard personal computers. Depending on the number of unknowns on different meshes used in the two examples, typical analysis requires a relatively small computational time on a standard personal computer with Pentium 166 MHz processor. The two examples demonstrate that the combined method can provide high solution accuracy at reduced computational time and memory for analysis of high-speed compressible flow problems.

6. Acknowledgment

The authors are pleased to acknowledge the Thailand Research Fund (TRF) and the National Research Council of Thailand (NRCT), Bangkok, Thailand, for supporting this research work.

7. References

[1] Anderson, J.D.,Jr. (1982), Modern Compressible Flow With Historical Prospective, McGraw-Hill, New York.

[2] Anderson, J.D.,Jr. (1991), Fundamentals of Aerodynamics, 2 ed., McGraw-Hill, New York.

[3] Hirsch, C. (1990), Numerical Computation of Internal and External Flows, John Wiley & Sons, New York.

[4] Dechaumphai, P. and Wieting, A.R. (1989), Coupled Fluid-Thermal-Structural Analysis for Aerodynamically Heated Structures, pp. 165-171, In T.J. Chung and G.R., Karr, Finite Element Analysis in Fluids, Univ. of Alabama in Huntsville Press, Huntsville, Alabama.

[5] Hughes, T.J.R. (1987), Recent Progress in the Development and Understanding of SUPG Methods with Special Reference to the Compressible Euler and Navier-Stokes Equa- tions. Numer, Methods Fluids, Vol.7, pp.1261-1275.

[6] Jiang, B.N. and Carey, G.F. (1988), A Stable Least-Squares Finite Element Method for Nonlinear Hyperbolic Problems, Numer. Methods Fluids, Vol.8, pp.933-942.

[7] Gnoffo, P.A. (1986), Application of Program LAURA to Three-Dimensional AOTV Flowfields, AIAA Paper, Vol. 86-0565.

[8] Dechaumphai, P. (1995), Adaptive Finite Element Technique for Heat Transfer Problems. Energy, Heat & Mass Transfer, Vol.17, pp.87-94.

[9] Peraire, J., Vahjdati, M., Morgan, K., and Zienkiewicz, O.C. (1987), Adaptive Re- meshing for Compressible Flow

Computation, Comp. Phys., Vol.72, pp. 449-466.

[10] Dechaumphai, P. (1994), Finite Element Method in Engineering, Chulalongkorn Univer- sity Press, Bangkok.

[11] Dechaumphai, P. (1995), Numerical Methods in Engineering, Chulalongkorn Uni-versity Press, Bangkok.

[12] Ramakrishnan, R., Bey, K.S. and Thornton, E.A. (1990), Adaptive Quadrilateral and Triangular Finite-Element Scheme for Compressible Flows, AIAA Journal, Vol.28, pp.51-59.

[13] Dechaumphai, P. and Thornton, E.A. (1982), Nodeless Variable Finite Elements for Improved Thermal-Structural Analysis, the International Conference on Finite-Element Methods, pp.139-144.

[14] Lohner, R., Morgan, K. and Zienkiewicz, O.C. (1984), Adaptive Grid Refinement for Compressible Euler and Navier-Stokes Equations, The International Conference on Accuracy Estimates and adaptive Refinements in Finite-Element Computations, pp.189-202.

[15] Oden, J.T. and Carey, G.F. (1981), Finite Elements: Mathematical Aspects, Prentice-Hall, Englewood Cliffs, New Jersey.

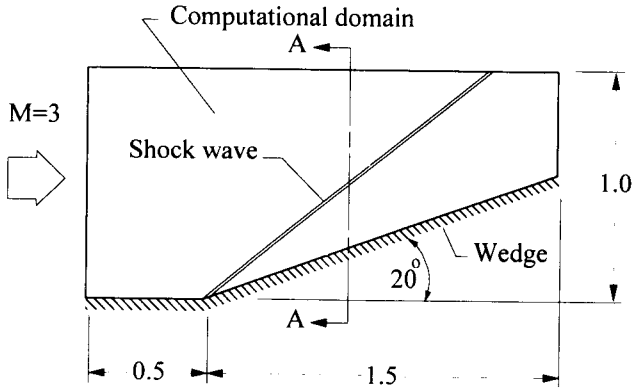
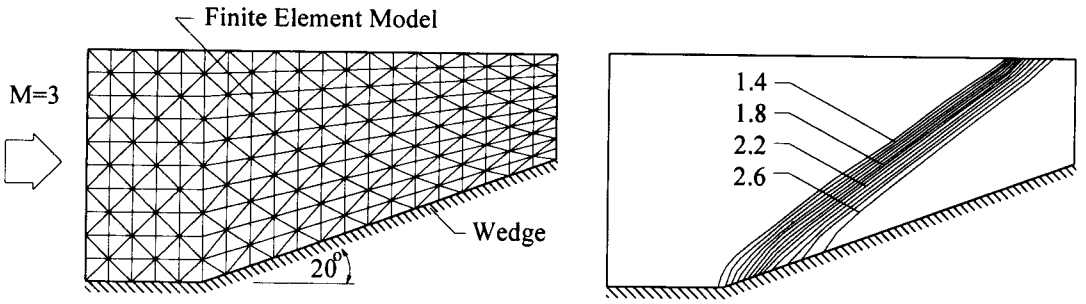
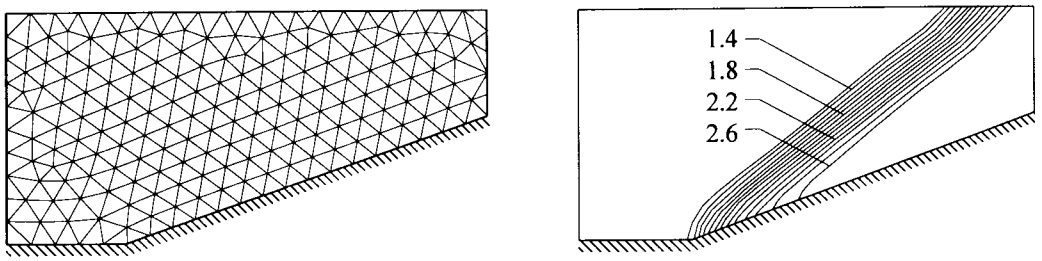


Fig. 1 - Mach 3 flow past a wedge.



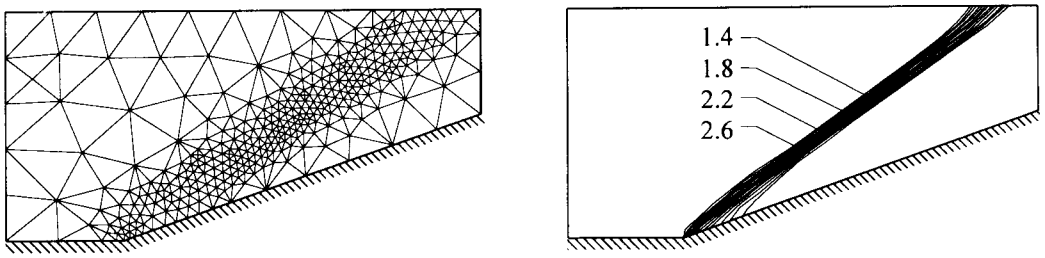
(a) Nonadaptive mesh (b) Density contours

Fig. 2 - Nonadaptive finite element mesh and corresponding density contours for Mach 3 flow past a wedge.



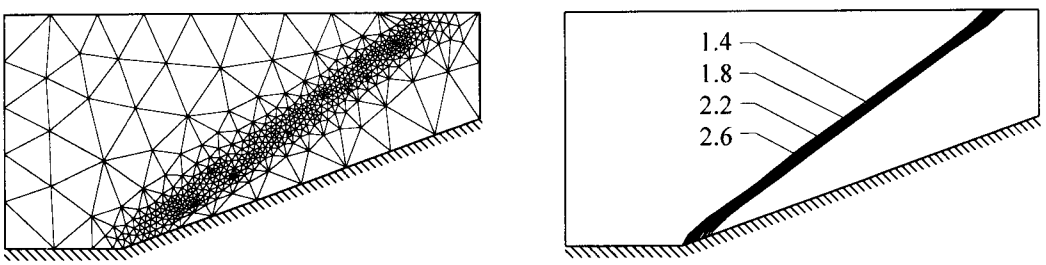
(a) First adaptive mesh (b) Density contours

Fig. 3 - First adaptive finite element mesh and corresponding density contours for Mach 3 flow past a wedge.



(a) Second adaptive mesh (b) Density contours

Fig. 4 - Second adaptive finite element mesh and corresponding density contours for Mach 3 flow past a wedge.



(a) Third adaptive mesh (b) Density contours

Fig. 5 - Third adaptive finite element mesh and corresponding density contours for Mach 3 flow past a wedge.

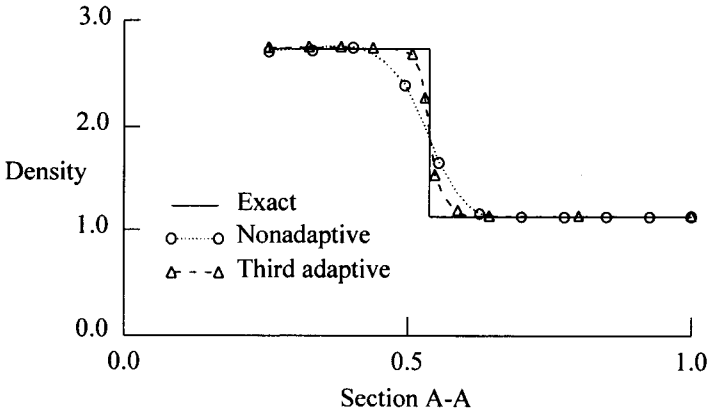


Fig. 6 - Comparative density distributions along section A-A of Fig. 1 between the exact, the nonadaptive mesh, and the third adaptive mesh solutions.

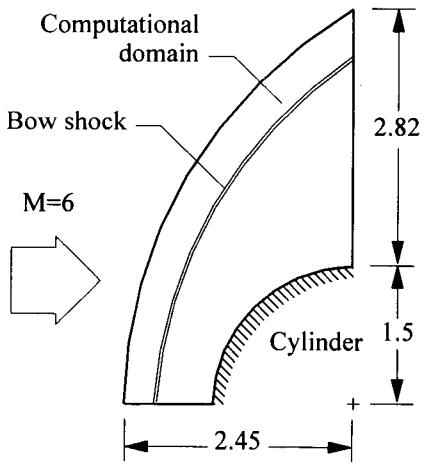
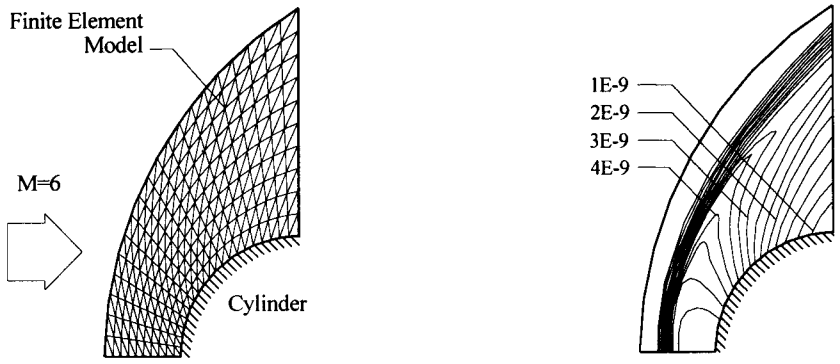
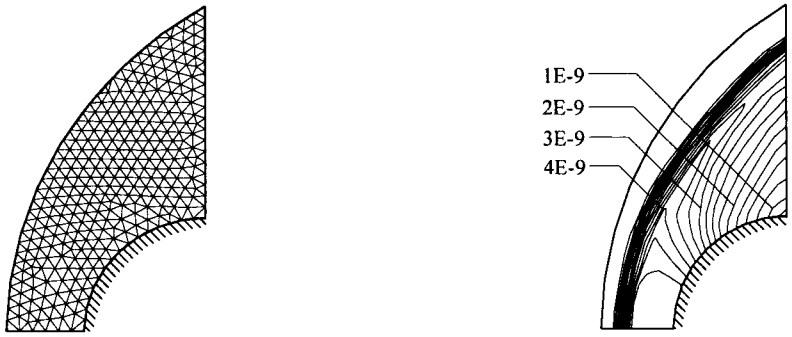


Fig. 7 - Mach 6 flow past a cylinder.



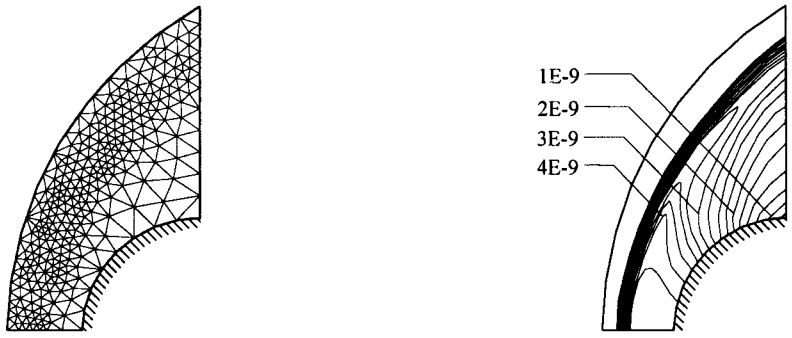
(a) Nonadaptive mesh (b) Density contours

Fig. 8 - Nonadaptive finite element and corresponding density contours for Mach 6 flow past a cylinder.



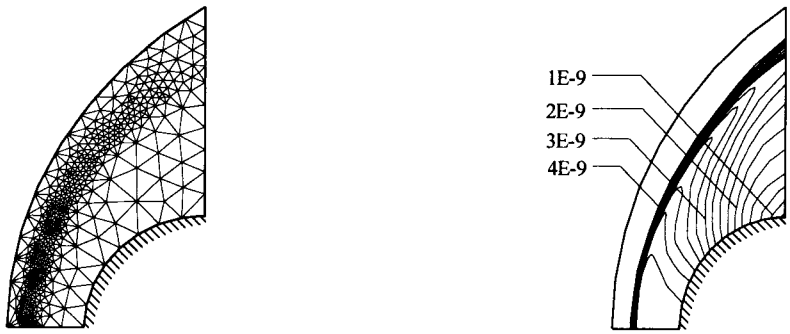
(a) First adaptive mesh (b) Density contours

Fig. 9 - First adaptive finte element mesh and corresponding density contours for Mach 6 flow past a cylinder.



(a) Second adaptive mesh (b) Density contours

Fig.10 - Second adaptive finte element mesh and corresponding density contours for Mach 6 flow past a cylinder.



(a) Third adaptive mesh (b) Density contours

Fig. 11 - Third adaptive finte element mesh and corresponding density contours for Mach 6 flow past a cylinder.

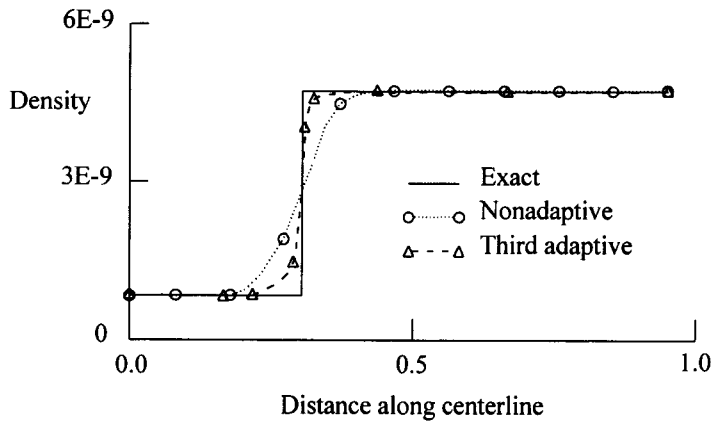


Fig. 12 - Comparative density distributions along the flow centerline between the exact, the nonadaptive mesh, and the third adaptive mesh solutions.