

Euler solution using adaptive Cartesian grid with a gridless boundary treatment

Liang Xiang · Guowei Yang

Received: 30 June 2008 / Revised: 25 September 2008 / Accepted: 13 October 2008 / Published online: 16 December 2008
© The Chinese Society of Theoretical and Applied Mechanics and Springer-Verlag GmbH 2008

Abstract A quadtree-based adaptive Cartesian grid generator and flow solver were developed. The grid adaptation based on pressure or density gradient was performed and a gridless method based on the least-square fashion was used to treat the wall surface boundary condition, which is generally difficult to be handled for the common Cartesian grid. First, to validate the technique of grid adaptation, the benchmarks over a forward-facing step and double Mach reflection were computed. Second, the flows over the NACA 0012 airfoil and a two-element airfoil were calculated to validate the developed gridless method. The computational results indicate the developed method is reasonable for complex flows.

Keywords Cartesian grid · Gridless method · Adaptation technique · Euler solution

1 Introduction

The technique of grid generation for complex geometry has become a big obstacle to popularize CFD techniques, and is also a key factor affecting the computation of CFD. Existing types of grid for CFD techniques include unstructured grids, body-fitted curvilinear grids, and Cartesian grids. In fact, structured Cartesian grid was widely used at the initial stage of CFD due to its ease and rapidity of grid generation, however, which was rapidly replaced of the curvilinear grid due to its complexity and lower accuracy for the treatment

of the wall surface boundary. Recently, for the unstructured Cartesian grid, because the simplicity of data structure and the ease in implementing adaptive refinement based on quadtree or octree topology tend to be mature, a renewed interest was excited for Cartesian grids [1, 2]. However, the main challenge in using a Cartesian method is in dealing with arbitrary boundaries.

Because the Cartesian grids are not body aligned, Cartesian cells near the body can extend through the surfaces of solid objects. So an accurate means of representations for surface boundary conditions is essential for the success of Cartesian method. Before the cut-cells method was used to resolve the boundary conditions. A cut-cells method generates Cartesian grids over the whole flowfield in the first step, and clears the cells in the solid object in the second step, and then cut the cells which extend through the surfaces in the last step. The shapes of cells generated by a cut-cells method at the boundary are irregular, and this method can sometimes lead to the creation of very small cells at the boundary. These irregularly shaped cut cells bring considerable complexity to the computation of the volume and fluxes for them and, moreover, these tiny cells pose the problem of numerical stability.

An alternative to resolving the boundary conditions is the gridless method [3, 4]. Unlike the cut-cells method, this method uses a direct least-squares fitting method to get the flux variables and has a great flexibility in handling the complex arbitrary distribution of grid points near solid boundaries. In the present paper, we attempt to combine the Cartesian grid method and the gridless method. We use the Cartesian grids method for the grid point in the interior of the computational domain, and use the gridless method to implement the boundary conditions. A finite volume formulation with Roe formulation for the Euler equations and use gridless approach to discretize the wall surface was

The project was supported by the National Natural Science Foundation of China (10672168).

L. Xiang · G. Yang (✉)
LHD of Institute of Mechanics,
Chinese Academy of Sciences, 100190 Beijing, China
e-mail: gwyang@imech.ac.cn

developed. The time-marching was implemented with the modified four-stage Runge–Kutta scheme. The developed method has been used to solve the flow field of shock reflection over a forward-facing step, double Mach reflection, NACA 0012 airfoil and NLR 7301 two-element airfoil. Results are compared with those of body-fitted curvilinear grids or experimental data.

2 Numerical method

2.1 Two-dimensional Euler equations

The two-dimensional Euler equations consisting of the mass, momentum, and energy conservation laws that govern the motion of an inviscid fluid can be written in integral form as

$$\frac{\partial}{\partial t} \iint_S Q dS + \oint_{\partial S} F \cdot n dl = 0, \quad (1)$$

where S denotes the area; F is the flux vector; dl is a surface element, and n is the outward normal; and

$$Q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix}, \quad F = \begin{bmatrix} \rho u & \rho v \\ (\rho u^2 + p) & \rho uv \\ \rho uv & (\rho v^2 + p) \\ (\rho uE + up) & (\rho vE + vp) \end{bmatrix}. \quad (2)$$

Here, p , ρ , u , v , and E denote the pressure, density, Cartesian velocity components, and total energy. For a perfect gas,

$$E = \frac{p}{(\gamma - 1)\rho} + \frac{u^2 + v^2}{2}, \quad (3)$$

where γ is the ratio of specific heats.

Spatial discretization of the Euler equations for an interior node in the Cartesian grid is performed using Roe's scheme

$$\tilde{F}_{i+1/2} = \frac{1}{2}[F(U_l) + F(U_r)] - \frac{1}{2}L^{-1}|\tilde{\Lambda}|L(U_r - U_l). \quad (4)$$

The Runge–Kutta time marching method is applied in time discretization. The scheme is explicit with a Courant–Friedrichs–Lewy (CFL) condition of $CFL \leq 2\sqrt{2}$ and has second-order accuracy in time for a nonlinear equation. To increase the allowable stability bound, the local time stepping is used to accelerate convergence to steady state.

2.2 Cartesian grid

2.2.1 Data structure

Data structure plays an important role in grid generation and flow field calculation. The quadtree data structure is employed in the adaptive Cartesian grid. The generation of

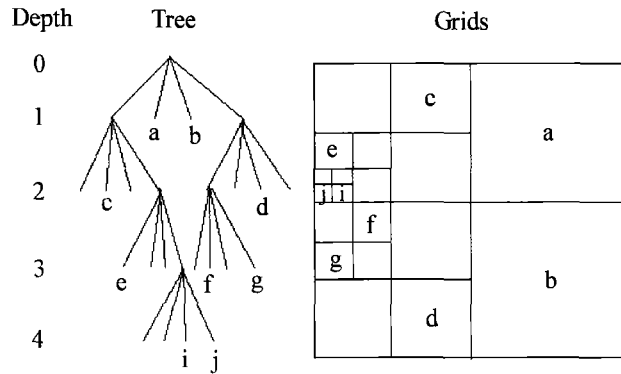


Fig. 1 Generation of Cartesian grid based on quadtree

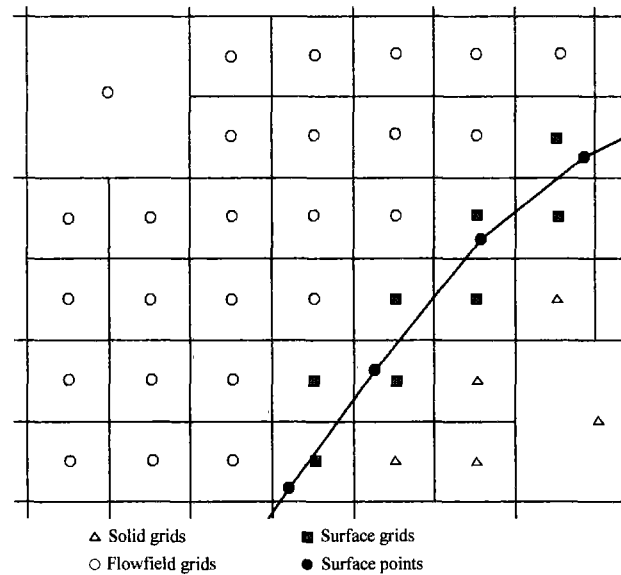


Fig. 2 The types of grid points

Cartesian grid based on quadtree data structure is illustrated in Fig. 1.

The depth, which denotes the times of grid subdivision, is in the left of Fig. 1. The root grid has a depth of zero. The quadtree data structure diagram is in the middle of Fig. 1, and it shows the generation of grid based on quadtree. The graph of grid generation is in the right of Fig. 1, and it shows the process of grid subdivision from a root grid. The flow field calculation is performed on the leaf grids, the grids that are not subdivided.

2.2.2 Classification of grids

Three types of grids and one type of points are identified as shown in Fig. 2.

Solid grids are grids that are inside the solid body and they are deleted in the computation. Surface grids are grids cut by solid body surfaces. Flowfield grids are grids that are

outside the solid body, and surface points are discrete points that represent the solid body surfaces.

2.2.3 Grid adaptation

A reasonable distribution of grids is very important for computational efficiency and accuracy. In present work, the grid adaptation is based on the gradient of density or pressure. For any flowfield grid, when its adaptation parameter is greater than the threshold, it is flagged for refinement, and contrarily when its adaptation parameter is less than the threshold and its depth is less than or equal to the depth of its adjacent grid, it is flagged for coarsening. The threshold is defined by

$$V_s = \frac{CR^{(n1-n0)}}{N}, \quad (5)$$

where V_s is the threshold, $n1$ is the depth of present grid, $n0$ is the depth of initial grids, and CR and N are control parameters.

2.3 Implementation of boundary conditions

Far-field boundary conditions are easily implemented in the same way as in any structured grid method. In the present work, the usual characteristic analysis based on one-dimensional Riemann invariants is used to determine the values of the inviscid flow variables. No special treatment is needed. The main challenge in using Cartesian grids is in the surface boundary condition implementation as the grids are not body aligned. A gridless method with least-squares technique is used in the present work to implement the surface boundary conditions. The boundary conditions are implemented such that the boundary conditions are satisfied in the process of solving the surface grid values.

2.4 Gridless treatment for surface grids

To avoid the complexity of cells-cutting method, a gridless method is used to treat the surface grids in present work. This is a very simple method for boundary treatment as it uses only clouds of grids and does not require the grids to have any prespecified connectivity among one another. The surface grids are treated separately. This makes the method easy to be implemented into any scheme.

The process of gridless approach treatment for the surface grids is presented as follows:

- (1) For each surface grid, the point P that is the nearest point on the solid wall surface to the center of the surface grid can be found as shown in Fig. 3.
- (2) A cloud of grids for each surface grid is set up for boundary condition implementation. For each surface grid, a cloud of eight flowfield grids are selected from its vic-

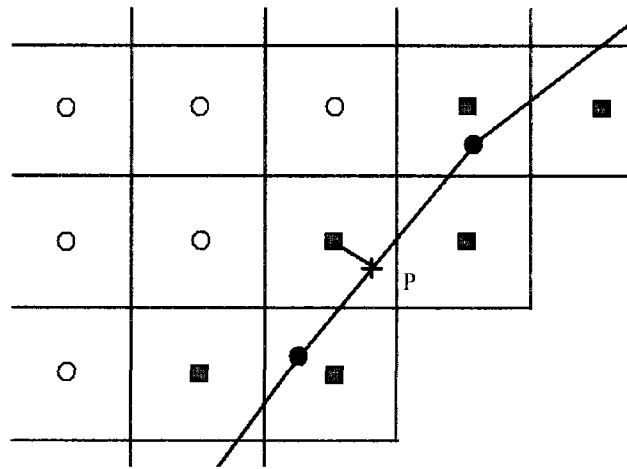


Fig. 3 The nearest point P to the center of surface grid

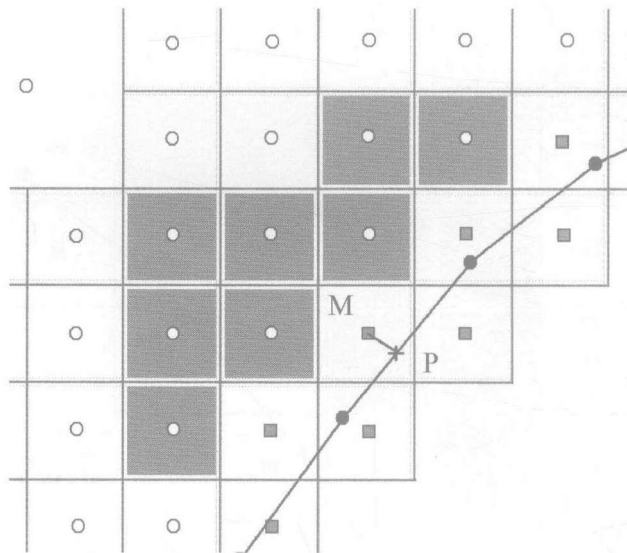


Fig. 4 Cloud grids (gray grids) for the surface grid M

nity. This selection is based on nearest distance to the point P of the grid as shown in Fig. 4. The eight shaded flowfield grids are included in the cloud for the surface grid M . To avoid choosing grids from the wrong side of the profile at thin surfaces, the selected cloud grids must fall on the same side as the outward normal direction of the surface as illustrated in Fig. 5.

- (3) For an inviscid condition the flow slips over the solid surface, and hence it is tangent to the surface. The slip condition is thus imposed on the boundary, and the flow variables on the solid surface are approximated as follows:

$$\frac{\partial \rho}{\partial \eta} = 0, \quad \frac{\partial V_{\xi}}{\partial \eta} = 0, \quad V_{\eta} = 0, \quad \frac{\partial p}{\partial \eta} = 0, \quad (6)$$

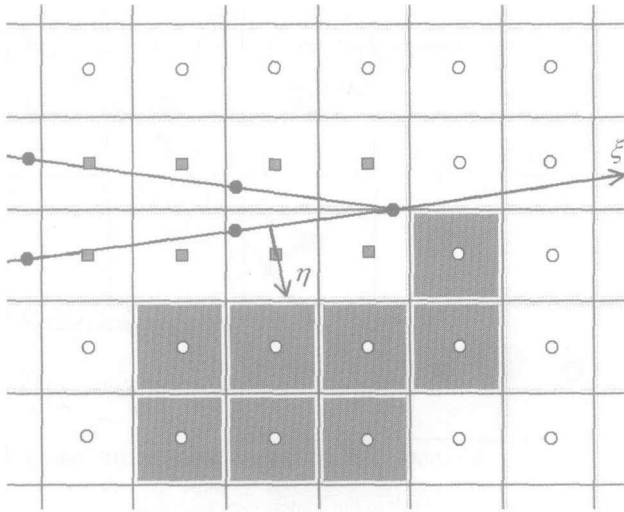


Fig. 5 Cloud grids (gray grids) in the thin surfaces

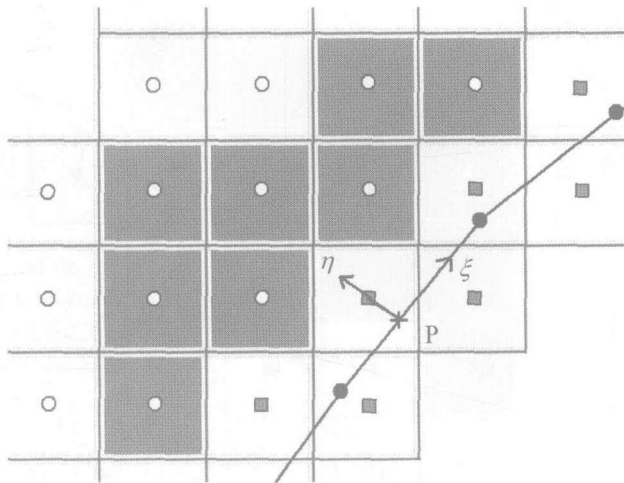


Fig. 6 ξ-η coordinate system

where the subscripts η and ξ stand for the normal and tangential directions, respectively, along the surface.

As shown in Fig. 6, for each surface grid, the values of flow variables at its cloud grids are resolved to the tangential ξ and normal η directions of the local ξ - η coordinate system with its origin at the point P. This is for easy implementation of the boundary conditions because they are given in the ξ and η directions.

In each cloud, the flow variables p, ρ, u and v are assumed to vary according to the function:

$$f(\xi, \eta) = b_1 + b_2\xi + b_3\eta + b_4\xi\eta. \tag{7}$$

Taking p as an example and applying the boundary condition in Eq. (6) on the surface grid will result in the following:

$$p(\xi, \eta) = b_1 + b_2\xi + b_3\eta + b_4\xi\eta, \tag{8}$$

$$\left. \frac{\partial p}{\partial \eta} \right|_{(0,0)} = b_3 + b_4\xi \Big|_{(0,0)} = b_3 = 0. \tag{9}$$

Substituting Eq. (9) into Eq. (8), a new function is obtained with the number of unknowns reduced by one

$$p(\xi, \eta) = b_1 + b_2\xi + b_4\xi\eta. \tag{10}$$

Substituting the values of pressure and coordinate at the center of each cloud grid into the Eq. (10) will result a system of eight equations with three unknowns:

$$\begin{aligned} p_1 &= b_1 + b_2\xi_1 + b_4\xi_1\eta_1, \\ p_2 &= b_1 + b_2\xi_2 + b_4\xi_2\eta_2, \\ p_3 &= b_1 + b_2\xi_3 + b_4\xi_3\eta_3, \\ p_4 &= b_1 + b_2\xi_4 + b_4\xi_4\eta_4, \\ p_5 &= b_1 + b_2\xi_5 + b_4\xi_5\eta_5, \\ p_6 &= b_1 + b_2\xi_6 + b_4\xi_6\eta_6, \\ p_7 &= b_1 + b_2\xi_7 + b_4\xi_7\eta_7, \\ p_8 &= b_1 + b_2\xi_8 + b_4\xi_8\eta_8. \end{aligned} \tag{11}$$

This is an overdetermined system, and can be solved by a least-squares method [5]. By determining the values of b_1, b_2, b_3 and b_4 , the functions (8) is found. Then the pressure values at the boundaries of surface grids, which are needed for the Roe's scheme, are evaluated according to the function. Because the boundary condition is incorporated into the system of equations to be solved, the values computed automatically satisfy the boundary conditions.

The ρ and V_ξ are computed in a similar fashion, because the boundary conditions of ρ and V_ξ are the same as that of p .

The situation is different for V_η , because the boundary condition about it is different from the others. Applying the boundary condition in Eq. (6) for V_η will result in the following:

$$V_\eta(\xi, \eta) = b_1 + b_2\xi + b_3\eta + b_4\xi\eta, \tag{12}$$

$$V_\eta \Big|_{(0,0)} = b_1 = 0. \tag{13}$$

Substituting Eq. (13) into Eq. (12), a new function is obtained with the number of unknowns reduced by one

$$V_\eta(\xi, \eta) = b_2\xi + b_3\eta + b_4\xi\eta. \tag{14}$$

Substituting the values of normal velocity and coordinate at the center of each cloud grid into the Eq. (14) gives a system

of eight equations with three unknowns:

$$\begin{aligned}
 V_{\eta 1} &= b_2 \xi_1 + b_3 \eta_1 + b_4 \xi_1 \eta_1, \\
 V_{\eta 2} &= b_2 \xi_2 + b_3 \eta_2 + b_4 \xi_2 \eta_2, \\
 V_{\eta 3} &= b_2 \xi_3 + b_3 \eta_3 + b_4 \xi_3 \eta_3, \\
 V_{\eta 4} &= b_2 \xi_4 + b_3 \eta_4 + b_4 \xi_4 \eta_4, \\
 V_{\eta 5} &= b_2 \xi_5 + b_3 \eta_5 + b_4 \xi_5 \eta_5, \\
 V_{\eta 6} &= b_2 \xi_6 + b_3 \eta_6 + b_4 \xi_6 \eta_6, \\
 V_{\eta 7} &= b_2 \xi_7 + b_3 \eta_7 + b_4 \xi_7 \eta_7, \\
 V_{\eta 8} &= b_2 \xi_8 + b_3 \eta_8 + b_4 \xi_8 \eta_8.
 \end{aligned}
 \tag{15}$$

Then this system is solved by using the same method as described earlier.

In the end, the velocity V_ξ and V_η are transformed back into u and v in the x - y coordinate system.

Although simple to implement, the gridless method is not a conservative method [6]. By applying the gridless method only on surface grids, potential nonconservation is thus restricted to a layer of grids surrounding the solid body while the rest of the computational domain is computed using a fully conservative finite volume formulation. The size of this layer goes to zero, and thus the effect of the potential nonconservative effect also goes to zero as the grid size is reduced. The lack of conservativeness can be minimized with the use of sufficient resolution near the surface.

3 Results and discussion

Results are presented for shock reflection over a forward-facing step, double Mach reflection, NACA 0012 airfoil and NLR 7301 two-element airfoil are computed.

3.1 Shock reflection over a forward-facing step

The computing model is a Mach 3.0 flow over a forward step. The height of step is one-fifth of the width of tunnel. The initial condition of the inflow is $Ma = 3$, $\rho = 1$, $p = 1/1.4$, $u = 3$. The initial grid is presented in Fig. 7 and the adaptive grid and unsteady pressure contour at typical nondimensional time are presented in Figs. 8, 9 and 10. The results

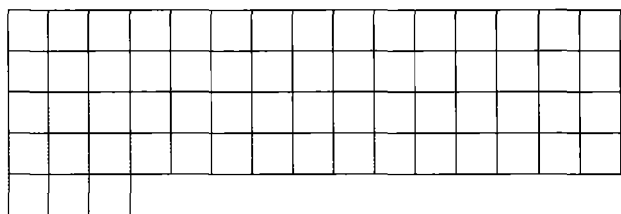


Fig. 7 Initial grid for forward-facing step case

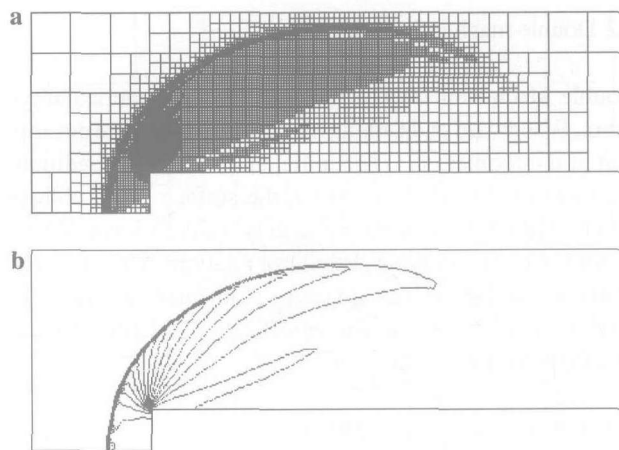


Fig. 8 Result for forward-facing step case after 5 nondimensional time units. a Adaptive Cartesian grid; b Pressure contours

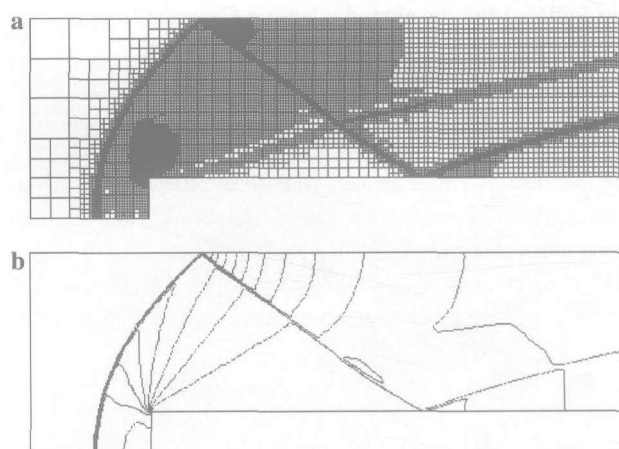


Fig. 9 Result for forward-facing step case after 15 nondimensional time units. a Adaptive Cartesian grid; b Pressure contours

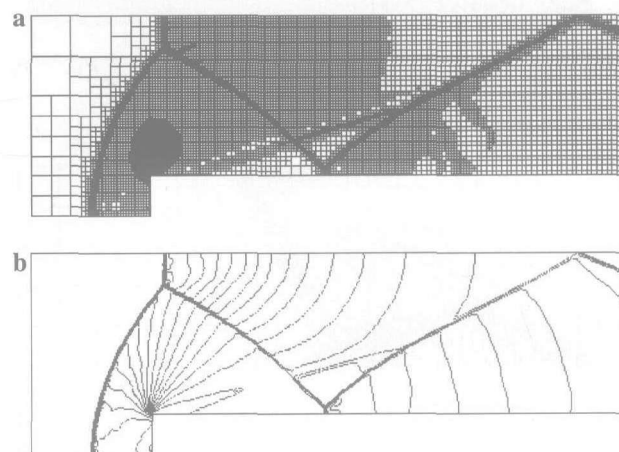


Fig. 10 Result for forward-facing step case after 30 nondimensional time units. a Adaptive Cartesian grid; b Pressure contours

indicated the developed adaptive Cartesian method can simulate the complex unsteady shock generation and interaction very well.

3.2 Double mach reflection

Double Mach Reflection is a classic test of numerical algorithm. Woodward and Colella have made a complete description of the problem [7]. The initial and boundary conditions are illustrated in Fig. 11. AD is the surface of an oblique shock at Mach 10, and the flows in two sides of gap AD are identified by the Rankine-Hugoniot relation. AB is the wall boundary, and the others are outlet boundaries. Because the shockwave moves, the length of DE varies with time and can be calculated by the function:

$$DE(t^n) = \frac{1}{6} + \frac{\sqrt{3}}{3}(1 + 2M_s ct^n), \quad (16)$$

where M_s is the Mach number, t^n is the number of time step. The results are presented in Figs. 12 and 13. The results also indicated the developed adaptive Cartesian method can simulate the shock/vortex interaction very well.

3.3 NACA 0012 airfoil

For the NACA 0012 airfoil, four flow cases with various combinations of free stream Mach number M_∞ and angle of attack α are computed: (1) $M_\infty = 0.5, \alpha = 3.0^\circ$; (2) $M_\infty = 0.85, \alpha = 0^\circ$; (3) $M_\infty = 0.8, \alpha = 1.25^\circ$; (4) $M_\infty = 2.0, \alpha = 0^\circ$. All four cases were computed using a CFL number

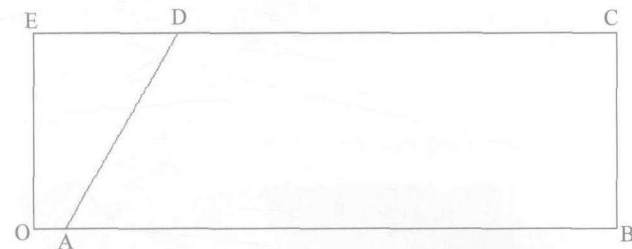


Fig. 11 Initial and boundary conditions

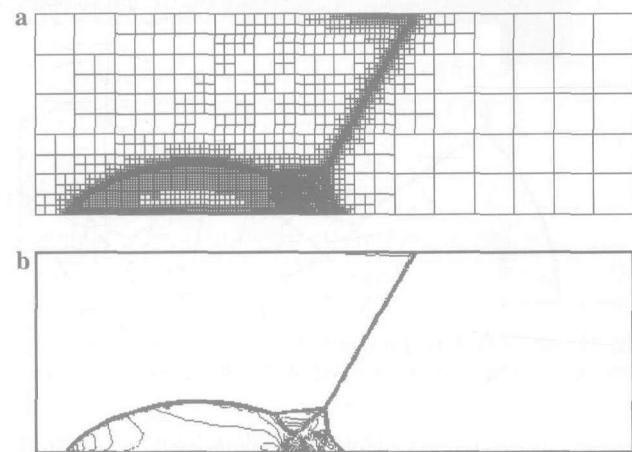


Fig. 12 Result for double Mach reflection after 0.1 nondimensional time unit. a Adaptive Cartesian grid; b Density contours

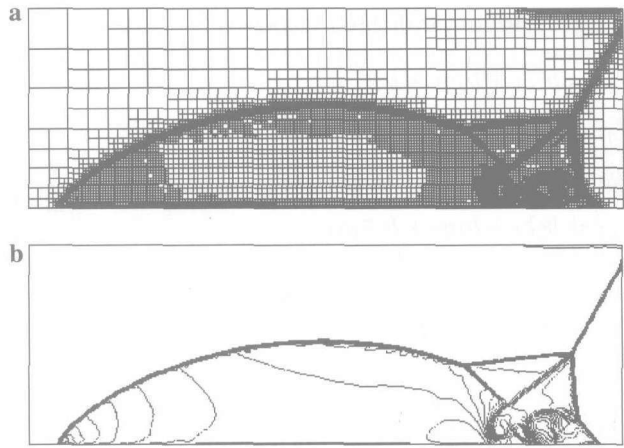


Fig. 13 Result for double Mach reflection after 0.2 nondimensional time unit. a Adaptive Cartesian grid; b Density contours

of 1.0 with local time stepping to accelerate the convergence to steady state. For each of the cases, the distributions of the pressure coefficient

$$C_p = \frac{2(p - p_\infty)}{\rho_\infty V_\infty^2} \quad (17)$$

are plotted. All solutions are compared with those computed using a body-fitted grid. A close-up view of a typical Cartesian grid and body-fitted grid is shown in Fig. 14.

For case 1 with $M_\infty = 0.5, \alpha = 3.0^\circ$, the adaptive Cartesian grid for the subsonic flow is presented in Fig. 15a and the

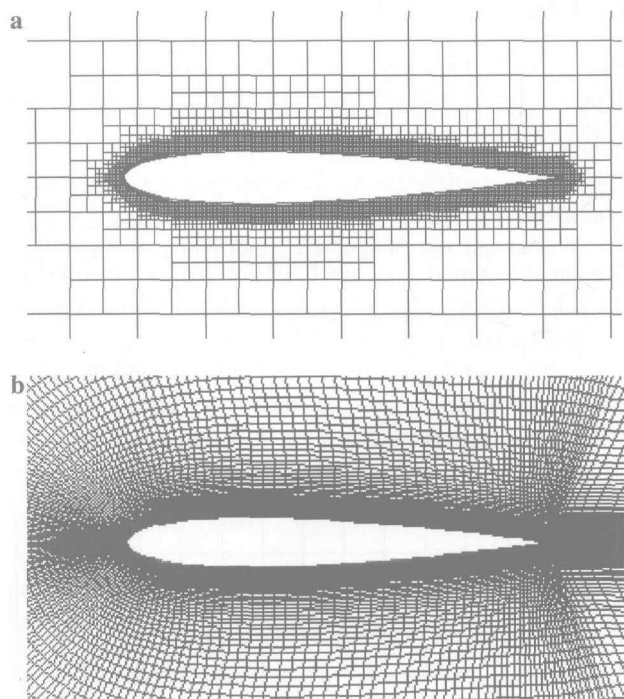


Fig. 14 Close-up view of a initial Cartesian grid; b body-fitted grid for the NACA 0012 airfoil

pressure contour in Fig. 15b. Figure 15c shows the comparison of C_p distribution between the adaptive Cartesian grid and body-fitted grid. They are agreement with each other except the pressure peak at the leading edge is little lower for Cartesian grid.

For case 2 with $M_\infty = 0.85$, $\alpha = 0^\circ$, the adaptive Cartesian grid for the transonic flow is presented in Fig. 16a and the pressure contour in Fig. 16b. Figure 16c shows the C_p comparison for the adaptive Cartesian grid and body-fitted grid. They are also agreement very well.

In case 3, the transonic flow of $M_\infty = 0.8$ with $\alpha = 1.25^\circ$ is computed. The adaptive Cartesian grid, pressure contour and C_p plot are shown in Fig. 17a–c, respectively. Pressure distributions agree very well for the two grid methods, which have a strong shock and a weak shock on the upper and lower surfaces separately. The shock wave in the flow field is well resolved. Because the conventional finite volume method is used in the interior flowfield on a Cartesian grid, the scheme is strictly conservative. The preceding transonic cases demonstrate that this method gives the correct shock speed and strength as expected.

Supersonic flow of $M_\infty = 2.0$ with $\alpha = 0^\circ$ is tested in case 4, and the results are presented in Fig. 18a–c. The C_p distribution of the adaptive Cartesian grid matches also very well with the body-fitted grid.

3.4 Two-element airfoil

To demonstrate the robustness and accuracy of the present method for complex configurations, the NLR 7301 two-element airfoil was taken a validated case [8]. The flow of $M_\infty = 0.185$ with $\alpha = 6^\circ$ and $Gap = 1.3\%$ is computed in current case. The adaptive Cartesian grid is shown in Fig. 19a and the pressure contour in Fig. 19b. Figure 19c compares the distributions of surface pressure distribution over each element with the experiment result. The computed result is in good agreement with the experiment result except in the leading edge, where the computed solution underpredicts the surface-pressure coefficient. Nevertheless, this two-element test case demonstrates the flexibility of the adaptive Cartesian grid method with the present gridless boundary condition implementation.

4 Conclusions

A quadtree-based adaptive Cartesian grid generator and flow solver have been developed successfully in this study. Features of the developed approach are summarized below:

- (1) Quadtree data structure was used for Cartesian grid and grid adaptation technique was performed.

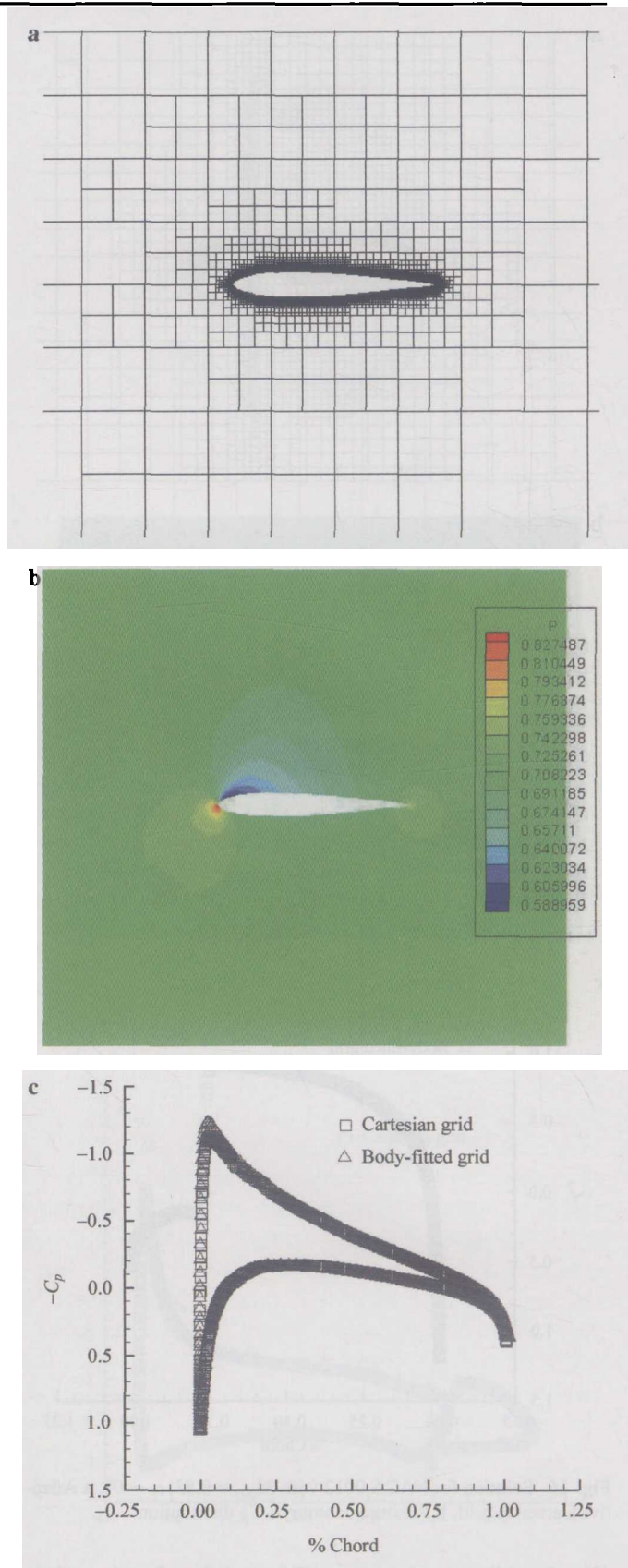


Fig. 15 Solution for NACA 0012 with $M_\infty = 0.5$, $\alpha = 3.0^\circ$. **a** Adaptive Cartesian grid; **b** Pressure contour; **c** C_p distribution

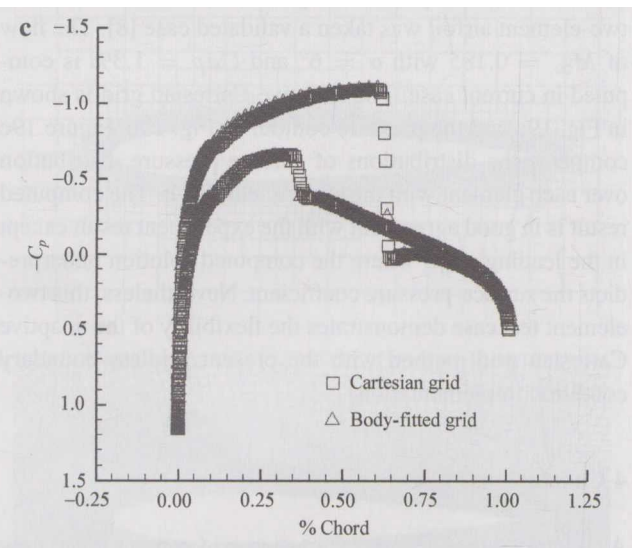
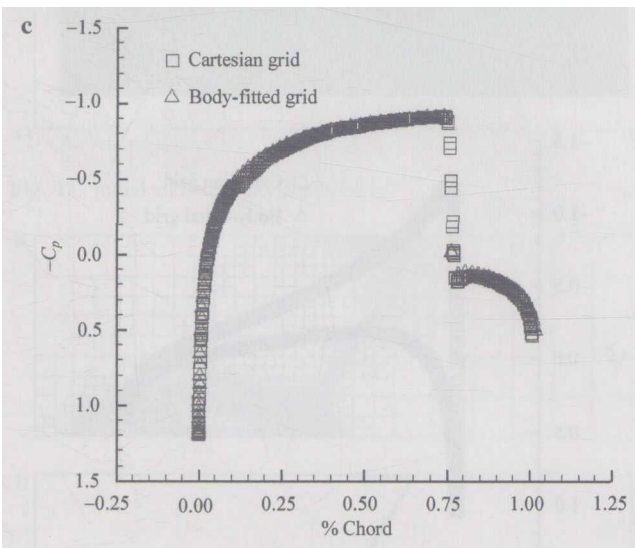
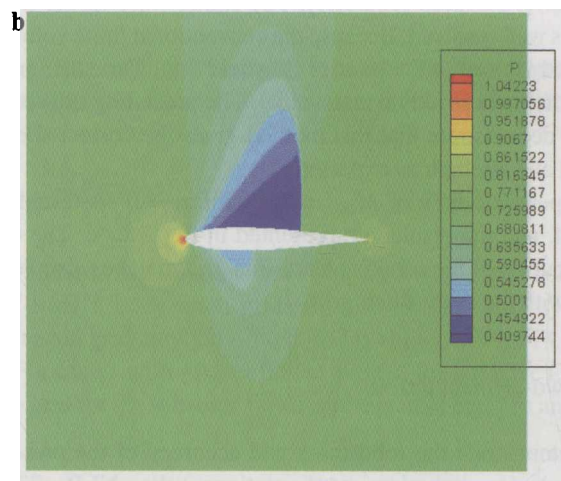
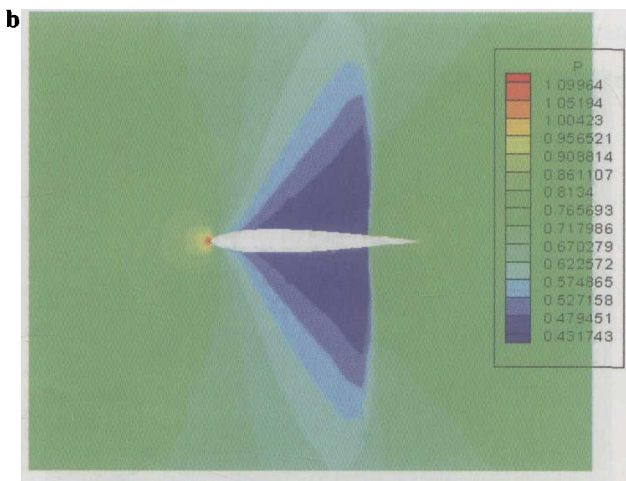
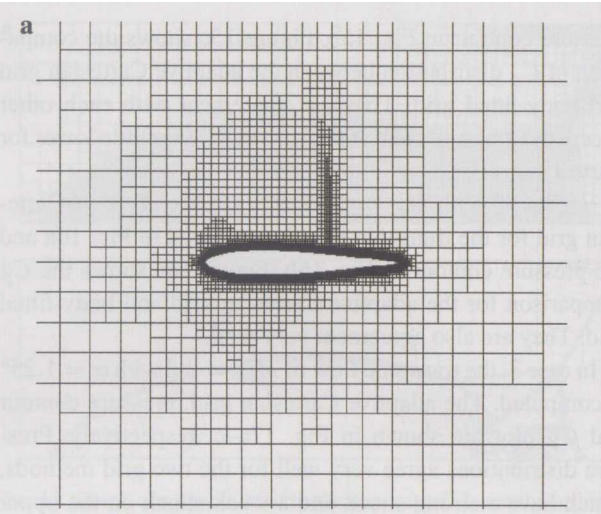
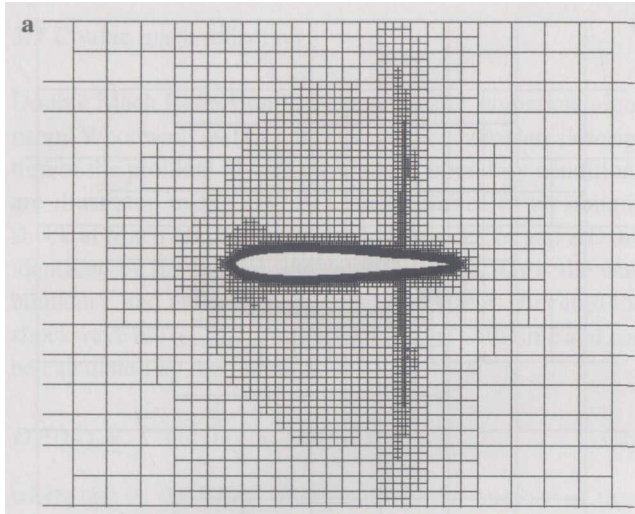


Fig. 16 Solution for NACA 0012 with $M_\infty = 0.85$, $\alpha = 0^\circ$. **a** Adaptive Cartesian grid; **b** Pressure contour; **c** C_p distribution

Fig. 17 Solution for NACA 0012 with $M_\infty = 0.8$, $\alpha = 1.25^\circ$. **a** Adaptive Cartesian grid; **b** Pressure contour; **c** C_p distribution

(2) A gridless method is used for solid surface boundary treatment and this makes the Cartesian grid have good flexibility in handling complex geometry.

(3) The developed method was validated with the many benchmarks. The results indicate that the approach offers accurate solutions and the grid adaptation. The

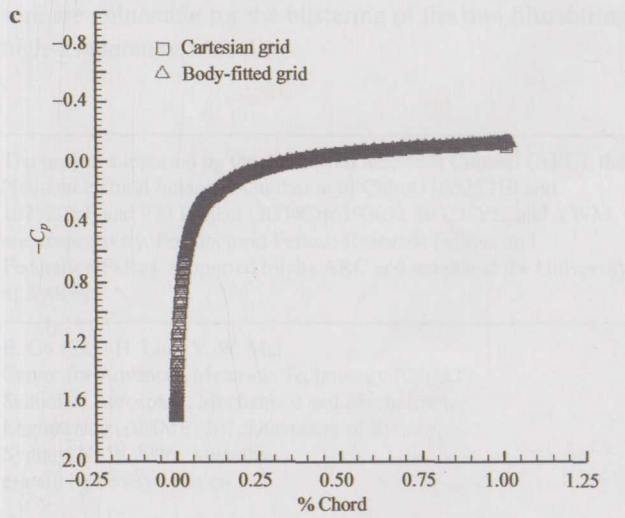
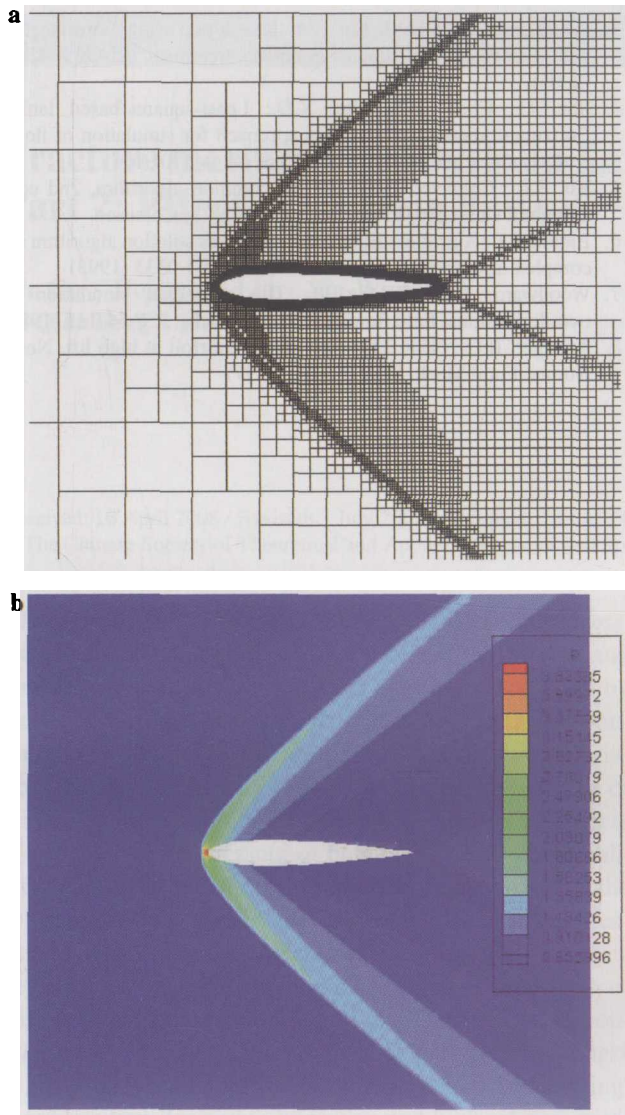


Fig. 18 Solution for NACA 0012 with $M_\infty = 2.0$, $\alpha = 0^\circ$. **a** Adaptive Cartesian grid; **b** Pressure contour; **c** C_p distribution

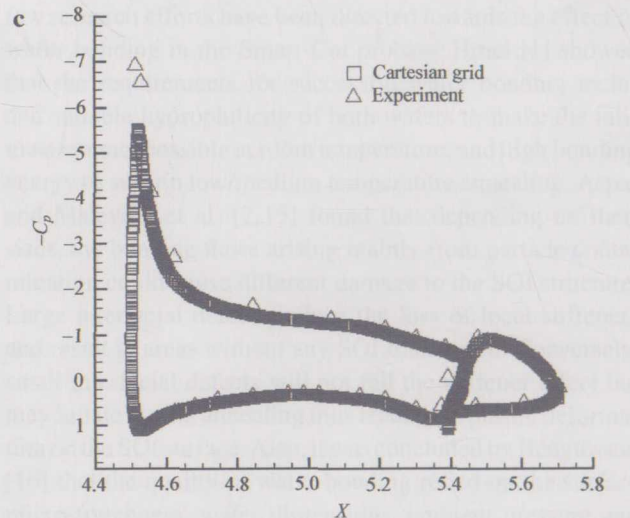
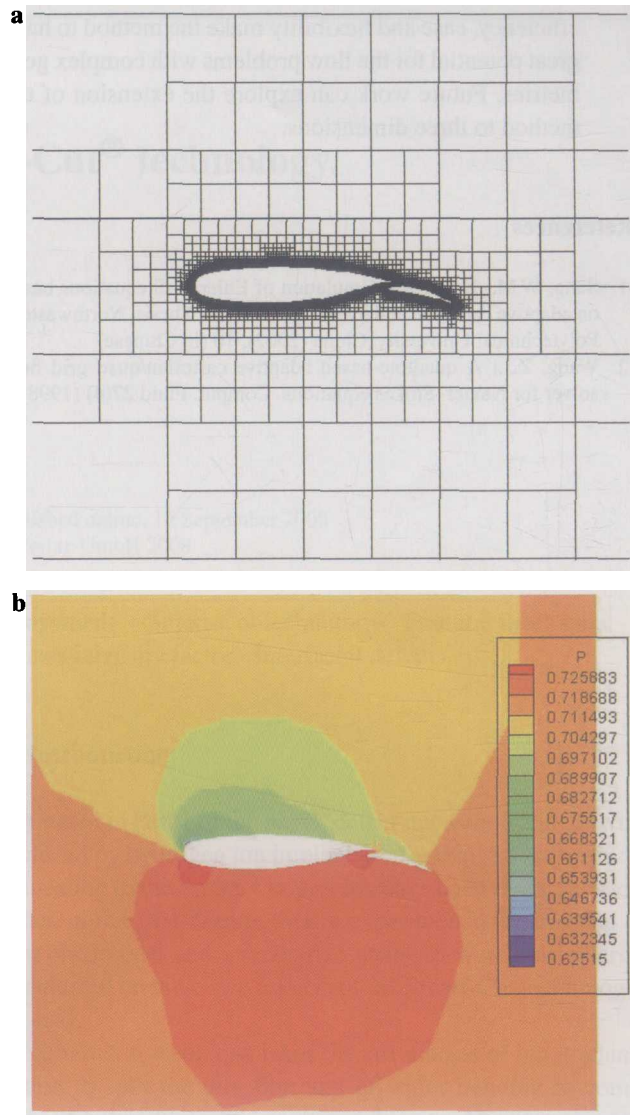


Fig. 19 Solution for NLR 7301 with $M_\infty = 0.185$, $\alpha = 6^\circ$, $Gap = 1.3\%$. **a** Adaptive Cartesian grid; **b** Pressure contour; **c** C_p distribution

efficiency, ease and flexibility make the method to have great potential for the flow problems with complex geometries. Future work can explore the extension of the method to three dimensions.

References

1. Sang, W.M.: Numerical simulation of Euler/N-S equations based on adaptive cartesian and hybrid grids. Ph.D. Thesis, Northwestern Polytechnical University, China (2002), 10 (in Chinese)
2. Wang, Z.J.: A quadtree-based adaptive cartesian/quad grid flow solver for Navier-Stokes equations. *Comput. Fluid* **27**(4) (1998)
3. Koh, E.P.C., Tsau, H.M., Liu, F.: Euler solution using Cartesian grid with a gridless least-squares boundary treatment. *AIAA J.* **43**(2) (2005)
4. Shu, C., Chew Y.T., Niu, X.D.: Least-squares-based lattice Boltzmann method: a meshless approach for simulation of flows with complex geometry. *Phys. Rev. E* **64**, 045701(R)
5. Xu, S.L.: Collected programs for common algorithm, 2nd edn. Tsinghua University Press, Beijing (1992) (in Chinese)
6. Batina, J.T.: A gridless Euler/Navier-Stokes solution algorithm for complex-aircraft application. *AIAA Paper* 93-0333 (1993)
7. Woodward, P.R., Colella, P.: The numerical simulation of two-dimensional fluid flow with strong shocks. *JCP* **54**, 115 (1984)
8. Arlinger, B.G.: NLR 7301 two element airfoil at high lift. *Notes Numer. Fluid Mech.* **58**, 375-396 (1997)