JOURNAL OF COMPUTATIONAL PHYSICS 118, 92–108 (1995)

Unified Zonal Method Based on the Fortified Solution Algorithm

KOZO FUJII

Institute of Space and Astronautical Science, Yoshinodai 3-1-1, Sagamihara, Kanagawa, 229, Japan

Received December 23, 1991; revised September 13, 1994

Computational fluid dynamics, with the aid of the supercomputer development are becoming in the matured stage both for the physical research problems and the practical engineering problems. Currently, a method handling complex body configurations and increasing the accuracy of simulations are two of the important topics in the CFD research. In the present paper, a new zonal method is proposed to solve both these problems. The method is an extension of the FSA (fortified solution algorithm for generality, originally called FNS, fortified Navier–Stokes) zonal method that the present author proposed in the past to improve the local accuracy of the flow field to be simulated. The FSA concept is used as an interface strategy and both overset and slightly overlapped zonal methods can be treated in a similar manner. The multi-block method can also be incorporated. Implementation of this idea into the existing explicit and implicit codes are easy and the treatment of complex body configurations and the improvement of local grid resolutions become enabled with a slight modification of existing conventional program codes. The idea may be considered to fall into the category of the Chimera method, but it has some advantages. The test case and some of the applications are given. They indicate that the present unified zonal method can be an effective CFD analytical tool for complex flow physics and complex body configurations. © 1995 Academic Press, Inc.

INTRODUCTION

Computational fluid dynamics (CFD) is now used for a wide variety of problems. However, there are still many obstacles to overcome before the CFD truly becomes a feasible tool for engineering design problems as well as physical findings. One of the obstacles is how to handle complex flow configurations. As the flow field to be simulated becomes more and more complicated, even the sophisticated grid generation computer codes cannot supply reasonable grids that are adequate for the flow simulations. One strategy is to adopt an unstructured grid concept [1, 2]. Discretization of the flow field is much easier and the geometrical adaptability can be improved. Under the concept of the structured grid, so-called zonal methods may be help for this. The Chimera method developed by Steger and his group [3, 4] is a good candidate. This type of approach using the overset grid concept allows us to create a computational grid for each body component and thus alleviate the burden of creating the computational grid over a complex body configuration.

One of the other obstacles is the accuracy enhancement. So far as discretization methods such as finite difference or finite volume methods are concerned, computed solutions become more accurate as the number of the grid points are increased. Quantitative data is necessary when CFD is used for practical problems, and a large number of grid points are required to assure the reliability of the simulation data. Since the computer memory is critically restricted, it is necessary to improve the grid resolution without increasing the number of grid points. Grid adaptation may be a solution. Local grid resolution can be improved by adopting the solution-adaptive gridding technique [5, 6] without increasing the total number of grid points. Another option may be local grid enrichment [7, 8]. In addition to the original grid, a locally defined, finer grid is created and overlapped or overlaid on the region of interest of the original grid.

To tackle both these two obstacles, we propose a unified zonal method based on the fortified solution algorithm (FSA zonal approach). The concept is similar to the Chimera method but has some advantages as will be shown later. To illustrate how the present FSA zonal method is organized, one of the problems to be used later as an application example is picked up and its schematic picture is shown in Fig. 1. The problem is a simulation of the flow field induced by a train moving into a tunnel. The flow field consists of (1) the region in front of the tunnel entrance, (2) the region inside the tunnel, (3) the region outside of the tunnel exit. Besides, the additional zonal grid that surrounds the train is prepared. This grid moves with the train. The train grid is overlaid onto the other three zonal grids. The region outside of the tunnel and inside the tunnel are connected with several grid points overlapped at the interface. As will be shown later, high-order accuracy can be maintained since the grid points in the overlapped region coincide with each other. The aerodynamic forces that act on the train are important to be analyzed, but also important is the propagation of strong pressure waves induced by the train entrance. To capture these relatively weak compression waves, another zone having finer grid distribution is created and this zone moves with the speed of the compression waves. Here again, the overset moving grid concept is used.

The FSA zonal method can treat several zonal strategies that appear in this example in the same manner. To some problems,

0021-9991/95 $6.00

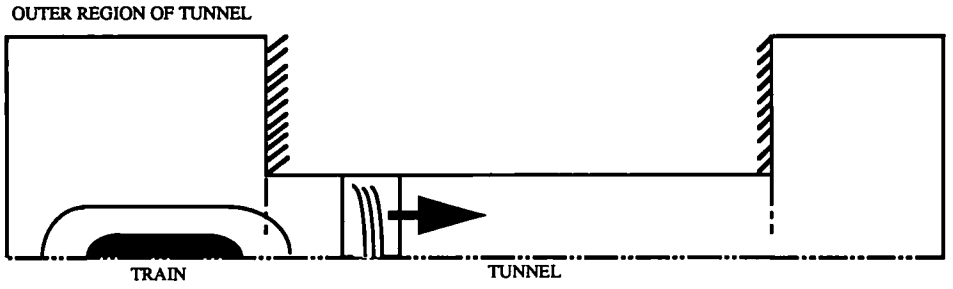


FIG. 1. Schematic picture of the flow field induced by a train moving into a tunnel.

a slightly overlapped grid concept is easy to apply. To some problems, an overset grid concept may be easier. The multiblock method [9], where the body geometry is mapped onto the rectangular-shaped blocks in the computational domain, may be convenient for some problems. The FSA zonal method can treat all these approaches under the same concept. In the next section, the formulation of the FSA zonal method is reviewed before showing the application examples.

FORTIFIED SOLUTION ALGORITHM AND THE DISCRETIZATION METHOD

The governing equations under consideration are the unsteady Navier-Stokes or Euler equations written under the generalized coordinate system (ξ, η).

$$\partial_\tau \hat{Q} + \partial_\xi(\hat{E} - \hat{E}_v) + \partial_\eta(\hat{F} - \hat{F}_v) = 0, \tag{1}$$

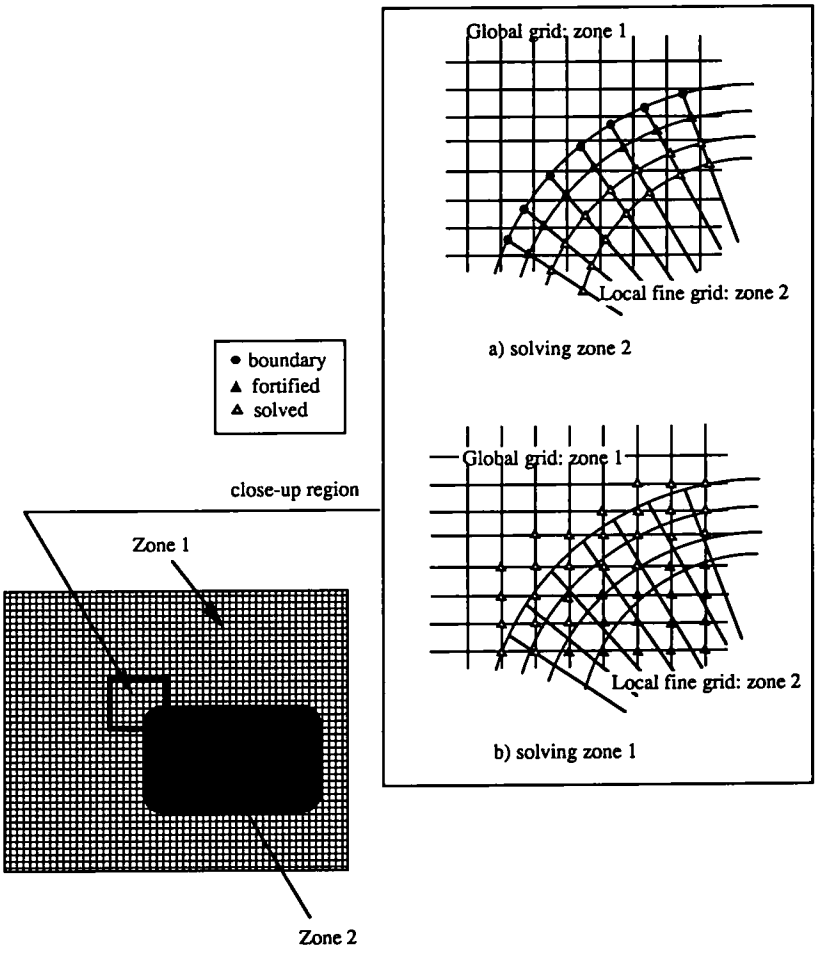


FIG. 2. FSA zonal procedure for the overlaid grid.

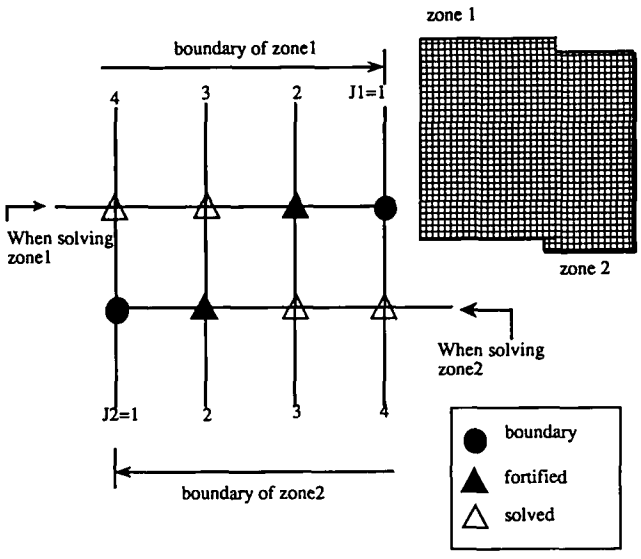


FIG. 3. FSA zonal procedure for the slightly overlapped grid.

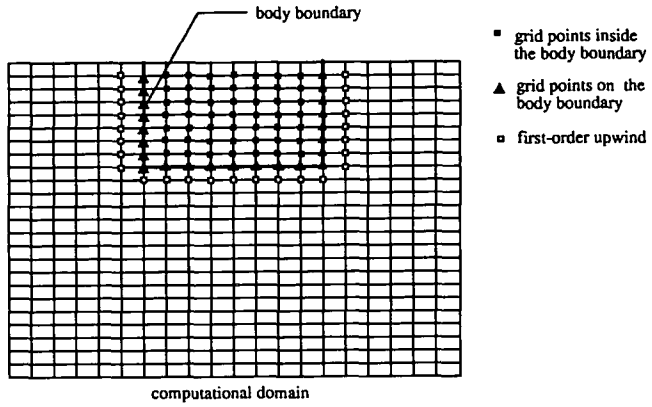


FIG. 4. FSA zonal procedure for the multi-block grid.

where τ is non-dimensional time, $\hat{Q}$ is a vector for conservative variables, $\hat{E}$ and $\hat{F}$ are the flux vectors, and $\hat{E}_v$ and $\hat{F}_v$ are the viscous flux vectors defined as

$$\hat{Q} = \frac{1}{J}\begin{bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{bmatrix}, \quad \hat{E} = \frac{1}{J}\begin{bmatrix} \rho U \\ \rho u U + \xi_x P \\ \rho v U + \xi_y P \\ (e+p)U \end{bmatrix},$$

$$\hat{F} = \frac{1}{J}\begin{bmatrix} \rho V \\ \rho u V + \eta_x P \\ \rho v V + \eta_y P \\ (e+p)V \end{bmatrix},$$

$$\hat{E}_v = \frac{1}{J}\begin{bmatrix} 0 \\ \xi_x \tau_{xx} + \xi_y \tau_{xy} \\ \xi_x \tau_{yx} + \xi_y \tau_{yy} \\ \xi_x \beta_x + \xi_y \beta_y \end{bmatrix}, \quad \hat{F}_v = \frac{1}{J}\begin{bmatrix} 0 \\ \eta_x \tau_{xx} + \eta_y \tau_{xy} \\ \eta_x \tau_{yx} + \eta_y \tau_{yy} \\ \eta_x \beta_x + \eta_y \beta_y \end{bmatrix};$$

U and V are contravariant velocity vectors and

$$\tau_{xx} = \frac{2}{3}\mu(2u_x - v_y), \; \tau_{yy} = \frac{2}{3}\mu(2v_y - u_x)$$

$$\tau_{xy} = \tau_{yx} = \mu(u_y + v_x)$$

$$\beta_x = \tau_{xx} u + \tau_{xy} v + \kappa T_x, \; \beta_y = \tau_{yx} u + \tau_{yy} v + \kappa T_y;$$

ξ_x, ξ_y, η_x, and η_y are metric terms that appear due to the coordinate transformation. The pressure, density, and the velocity components are related to the energy for an ideal gas by

$$p = (\gamma - 1)[e - \tfrac{1}{2}(u^2 + v^2)]. \tag{2}$$

Here, only two-dimensional equations are considered to explain the implementation of the FSA concept. Extension to the three-dimensional equations is straightforward and some of the application examples will be shown.

The FSA took the basic concept from the idea of the fortified Navier–Stokes (FNS) approach and, hence, the present method was originally called the FNS zonal method. The FNS concept itself (not the FNS zonal method) was originally developed by Van Dalsem and Steger [10] to improve the performance of Navier–Stokes algorithms by using fast auxiliary algorithms that solve subsets of the Navier–Stokes equations. In the FNS approach, solutions to the subset equations are used to add forcing terms to the Navier–Stokes algorithm in the appropriate flow regions. Van Dalsem and Steger used a boundary layer algorithm as subset equations, and the drag was accurately predicted with about an order-of-magnitude savings in computer time. The potential payoff of this approach is an improvement of both the efficiency and the accuracy of a given Navier–Stokes algorithm. The following is the concept of fortified Navier–Stokes approach.

In the FNS approach, the Navier–Stokes equations are modified to include the forcing term as

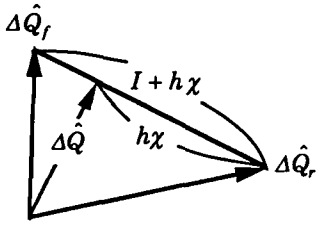


FIG. 5. Updated solution for the moderate χ.

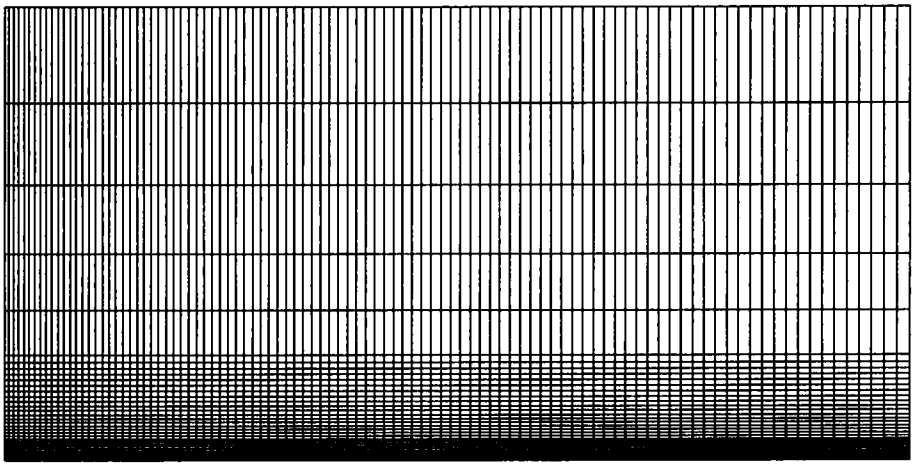


FIG. 6. Computational grid for the simulation of Blasius solutions.

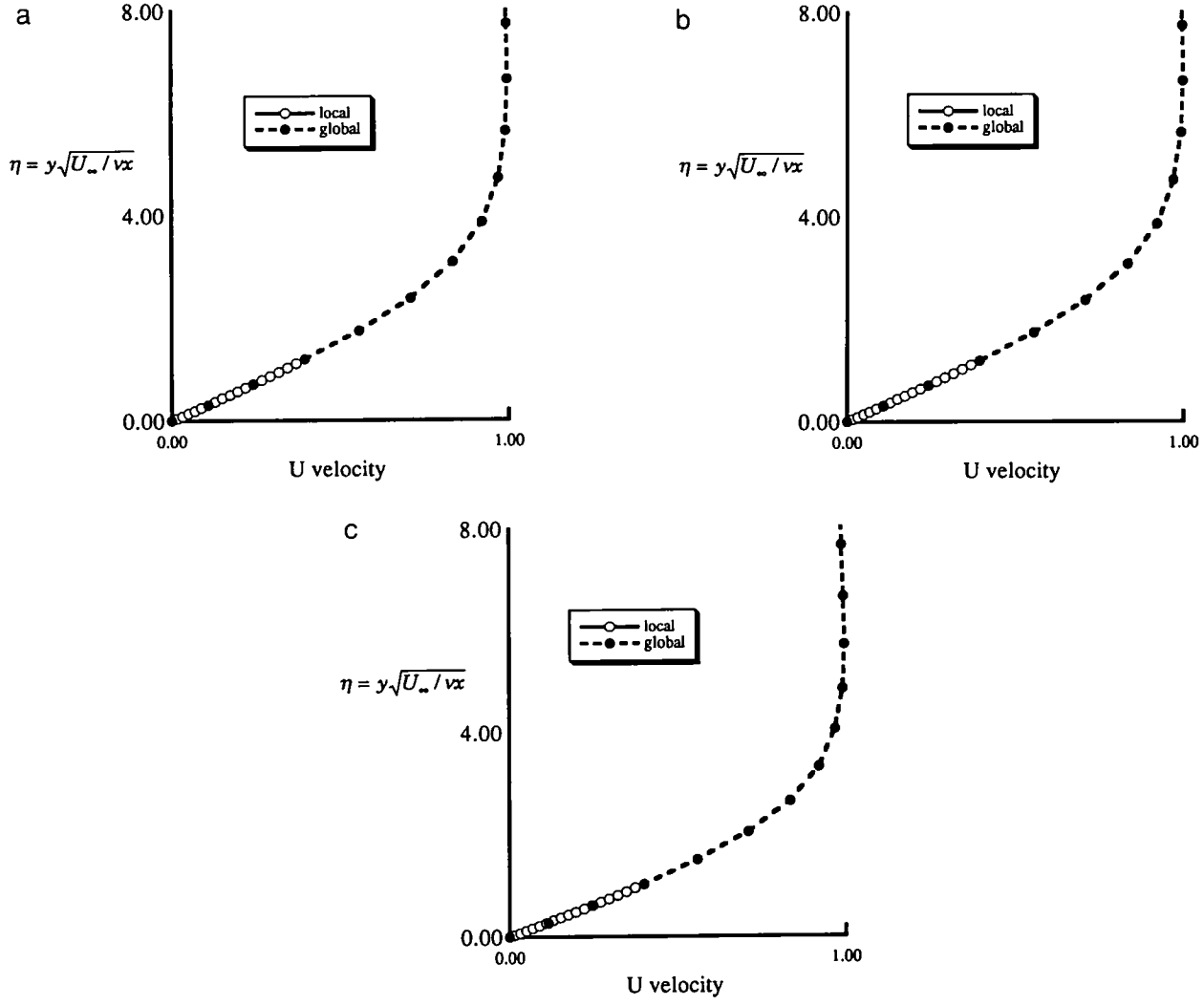


FIG. 7. Transformed U-velocity profile for different χ distributions. (101 × 21: global zone, 101 × 16: local zone): (a) $\chi = 10^9$; (b) $\chi = 10^2$;) $\chi = 0$.

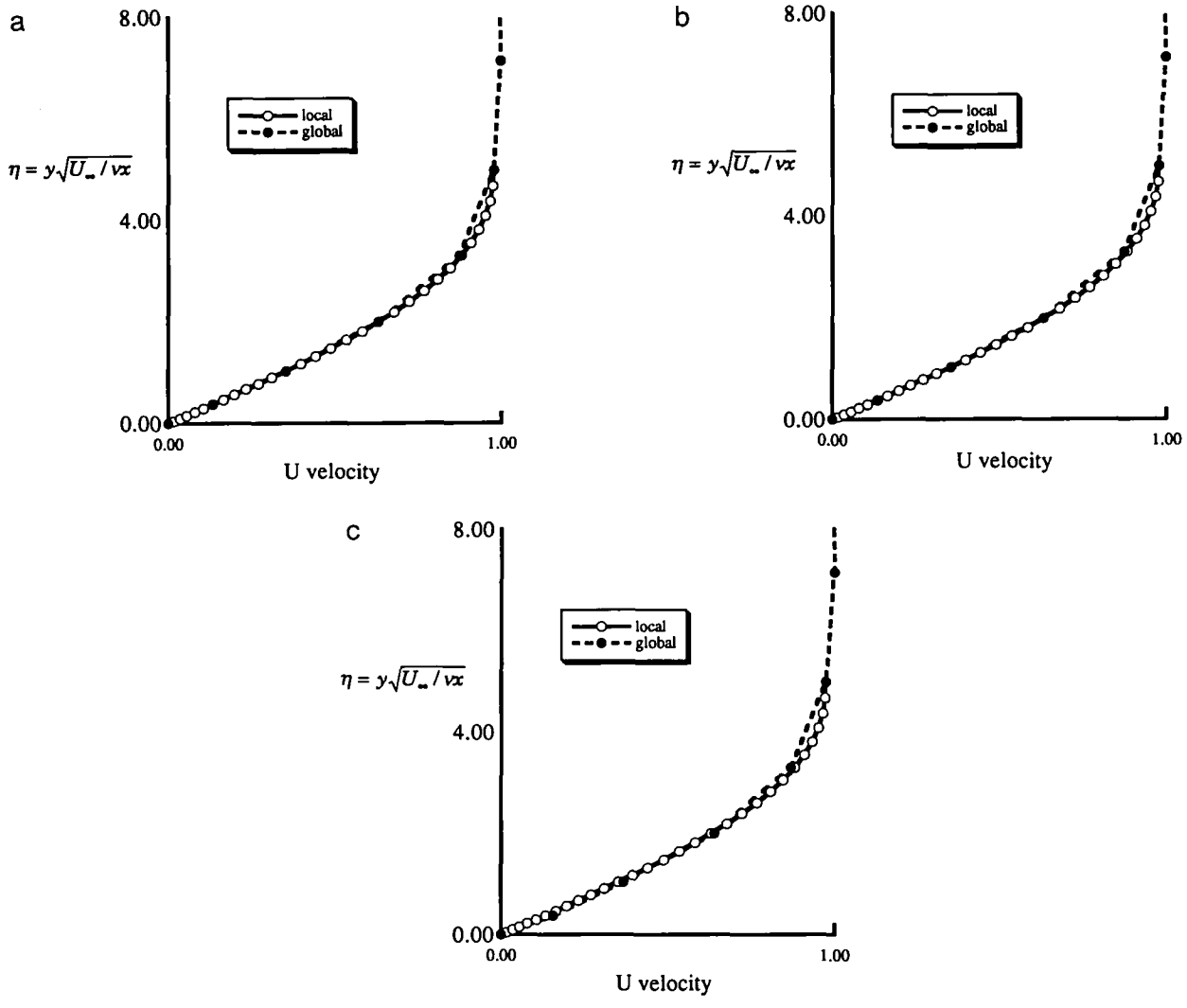


FIG. 8. Transformed U-velocity profile for different χ distributions. (101 × 11: global zone, 101 × 31: local zone): (a) $\chi = 10^9$; (b) $\chi = 10^2$; (c) $\chi = 0$.

$$\partial_\tau \hat{Q} + \partial_\xi(\hat{E} - \hat{E}_v) + \partial_\eta(\hat{F} - \hat{F}_v) = \chi(\hat{Q}_f - \hat{Q}). \tag{3}$$

The switching parameter χ is set to be sufficiently large, compared to all the other terms in the region, where the solution $\hat{Q}_f$ is available by the subset equations, and zero outside the region. For $\chi \gg 1$, the added source term simply forces $\hat{Q} = \hat{Q}_f$; otherwise it blends $\hat{Q}$ with $\hat{Q}_f$. When $\chi = 0$, the equations go back to the ordinary Navier–Stokes equations.

The solution algorithm for the Navier–Stokes equations should be modified so that the forcing terms are treated properly in the solution process. In the case of simple explicit time integrations, a finite difference expression of the Eq. (3) on a delta form can be written as

$$\begin{aligned} &I(1 + h\chi)(\hat{Q}^{n+1} - \hat{Q}^n) \\ &\quad = -h[\partial_\xi(\hat{E} - \hat{E}_v) + \partial_\eta(\hat{F} - \hat{F}_v) + \chi(\hat{Q}_f - \hat{Q}^n)]. \end{aligned} \tag{4}$$

Here, h is an integration time step size. The superscripts n indicate the level of the time step. The source term is easily treated implicitly. When $\chi = 0$, this reduces to the standard algorithm. If χ is sufficiently large, the algorithm reduces simply to $\hat{Q}^{n+1} = \hat{Q}_f$. Thus, in the region that χ is set to be large, the Navier–Stokes (or Euler) algorithm is turned off and the solution $\hat{Q}_f$ is "fortified" there. Note that positive diagonal terms are implicitly added when χ is large. Obviously, this has an effect of improving the stability of the algorithm.

Modification is almost the same for implicit time integration schemes. It is simply to change the I matrix to the $I(1 + h\chi)$ matrix with the consideration that χ can become large without concern for the large factorization errors. In the case of the LU-ADI factorization time integration algorithm that is used in many of the following examples, the modification can be written as follows. The LU-ADI algorithm uses ADI-type fac-

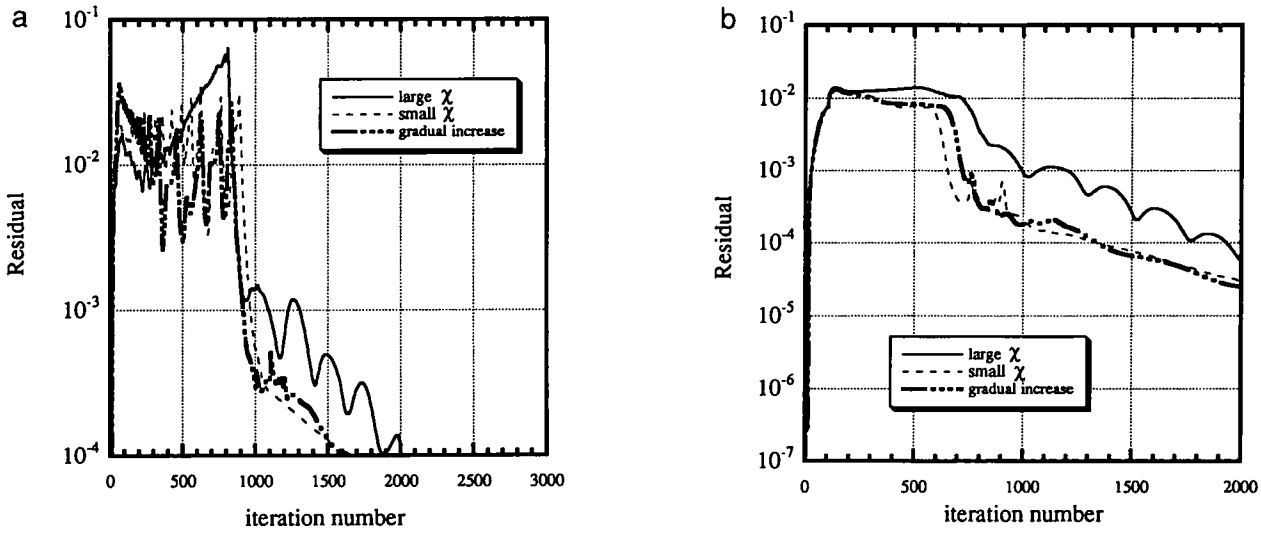


FIG. 9. Convergence history of the residual for supersonic blunt-body problem: (a) global zone; (b) local zone.

torization first, and then it uses approximate LDU decomposition for each ADI operator. In the original LU-ADI algorithm, each ADI operator, for instance, the operator for the ξ-direction, can be written as,

$$I + h\partial_\xi \hat{A} \doteq T_\xi (I - h\Lambda_\xi^- + h\delta_\xi^b \Lambda_\xi^+) \\ (I + h|\Lambda_\xi|_j)^{-1}(I + h\Lambda_\xi^+ + h\delta_\xi^f \Lambda_\xi^-) T_\xi^{-1}, \tag{5}$$

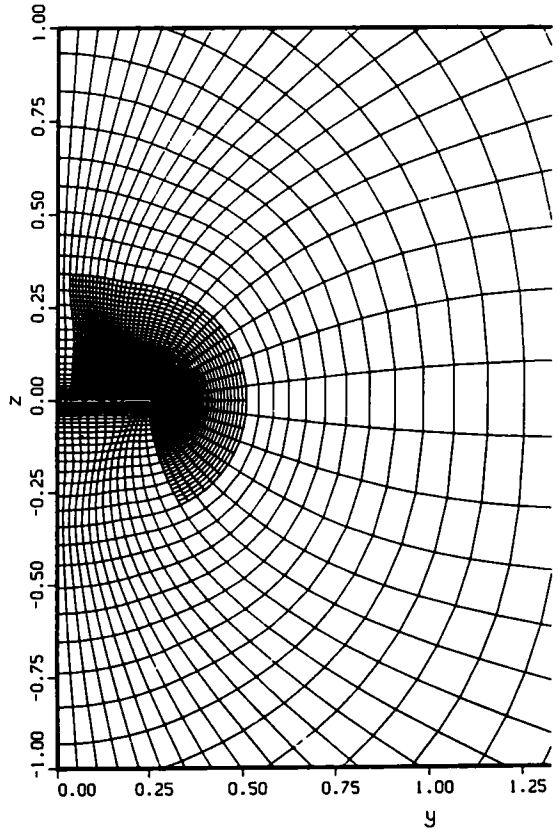


FIG. 10. Zonal grid distribution for the delta wing simulation.

where T_ξ and T_ξ^{-1} are the right and left eigenmatrices, respectively. Λ_ξ is the diagonal matrix of the eigenvalues, and its positive and negative parts are defined as $\Lambda_\xi^\pm = (\Lambda_\xi \pm |\Lambda_\xi|)/2$. δ_ξ^f and δ_ξ^b are forward and backward finite difference operators. Usually, first-order differencing is used. The operator defined in Eq. (5) is modified to add the forcing term as

$$T_\xi (I(1 + h\chi) - h\Lambda_\xi^- + h\delta_\xi^b \Lambda_\xi^+)(I(1 + h\chi) + h|\Lambda_\xi|_j)^{-1} \\ (I(1 + h\chi) + h\Lambda_\xi^+ + h\delta_\xi^f \Lambda_\xi^-) T_\xi^{-1}. \tag{6}$$

When χ is zero, Eq. (6) becomes Eq. (5). By putting the inversion of $I(1 + h\chi)$ between each ADI operator to avoid the large factorization errors, $\hat{Q}^{n+1} = \hat{Q}_f$ is realized when χ is large. The modification of the right-hand side is just to add source terms like Eq. (4). The fortified feature can be similarly implemented into many of the implicit time integration algorithms straightforwardly.

Using this FNS concept, the present author proposed a FNS zonal method to locally increase the number of grid points to enhance the accuracy of the vortical flow simulations [7]. Although the FNS was developed for the Navier–Stokes equations, it can be applied to any governing equations as can be imagined. Therefore, it is now called the fortified solution algorithm for generality and is considered to be a tool not only to increase the local grid points but also to handle complex flow configurations even though the concept itself does not change. The FSA zonal method utilizes the FNS concept above for the transformation of the information of one zone to another.

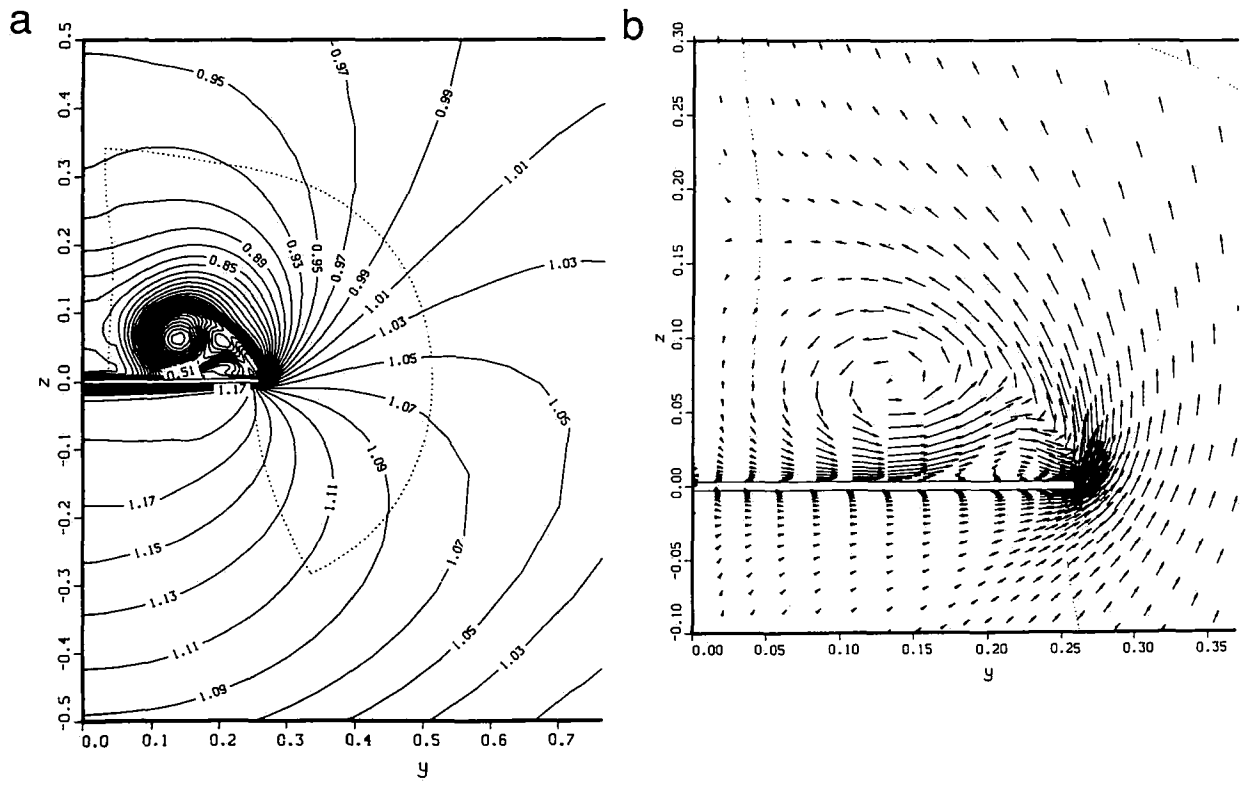


FIG. 11. Computed result for the FSA zonal method: (a) density contour plots; (b) velocity vector plots.

The basic equations are discretized with the finite difference fashion. The LU-ADI time integration algorithm [11] is used for the steady state problems, and the two step explicit time integration algorithm is used for the unsteady wave-propagation problems. In the right-hand side, convective terms are evaluated using the flux difference splitting by Roe [12]. The MUSCL interpolation is used for the higher-order extension. All the details about the solution algorithm except the treatment of the fortified source terms shown above can be found in Ref. [11].

It should be noted that the forcing solution $\hat{Q}_f$ can be a solution of any formula. It can be an analytical or experimental solution if available. New boundary conditions can be introduced in the computational region using the FSA when choosing the forcing region not to be an area but to be a line (in the three-dimensional case, not a volume but a surface). For instance, Van Dalsem and Steger solved a jet impingement problem [13].

UNIFIED ZONAL METHOD BASED ON THE FSA CONCEPT

Using the modified equations shown above, the FSA zonal method is defined. Figure 2 shows the schematic picture of the overset zonal grid. Zone 1 is prepared to solve the global region. Zone 2, a small region having finer grid distributions is prepared, for instance, to enhance the local grid resolution, or it may be prepared to improve the geometry adaptability. In the case of a train moving into a tunnel, zone 2 corresponds to a local moving grid surrounding the train. Close-up views of the zone 2 boundary are attached in Fig. 2 for the following description of the FSA treatment. Let us assume that the boundary conditions are given explicitly and the solution process starts at the next points from the boundary.

At each time step, zone 2 is solved first. χ values are set to be sufficiently large for the outermost two grid points (denoted as filled circles and triangles) as is shown in Fig. 2. $\hat{Q}_f$ variables are specified there by $\hat{Q}$ of zone 1 in the previous time step. Namely, the variables for the filled marks in Fig. 2 are given by the interpolation of the solution at the surrounding grid points in zone 1 obtained at previous time step. Filled circles can be treated as a boundary condition or just fortified without giving any boundary conditions. χ values are set to zero inside of zone 2 (open triangles in Fig. 2). Then Fortified equations, Eq. (3) are solved. All the grid points denoted by the open triangles are actually solved even though the solution process starts from the points next to the boundary just as in a regular program. Next, zone 1 is solved using the fortified Navier–Stokes equations. χ values are now set to be sufficiently large inside the subset zone 2 (denoted as filled triangles) and $\hat{Q}_f$ in

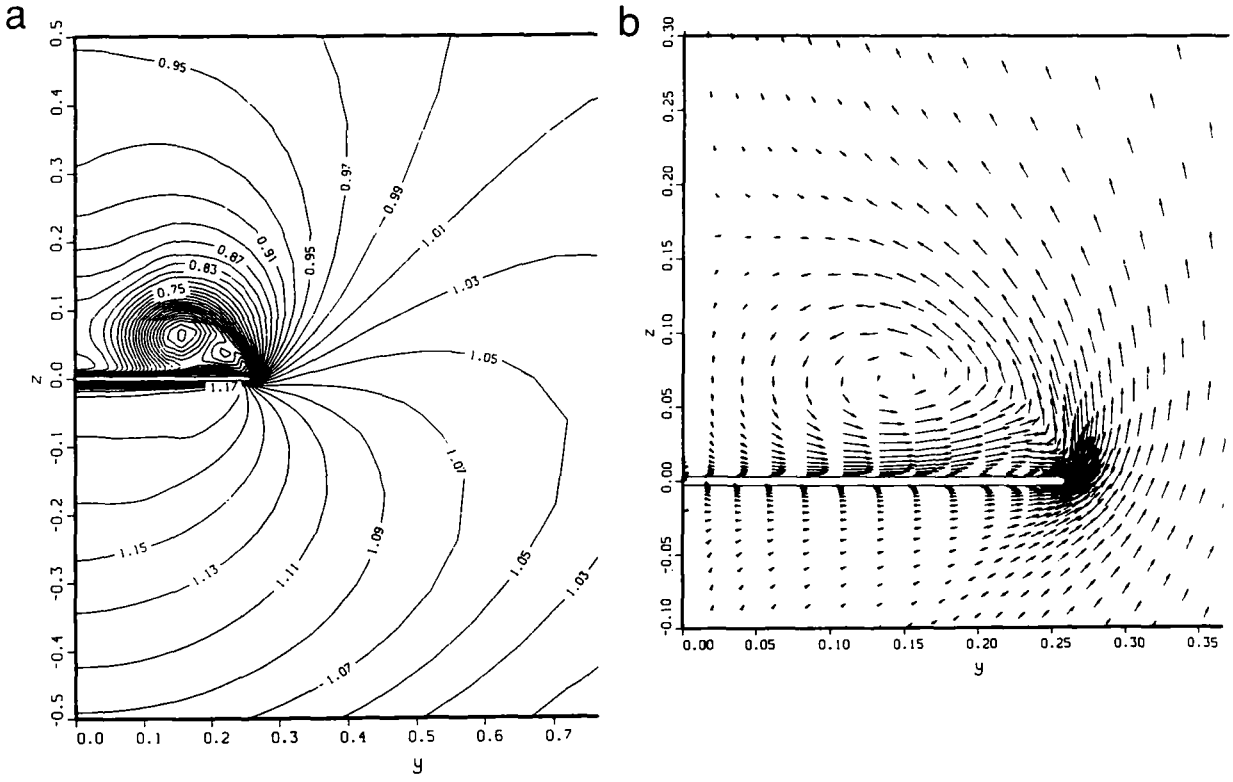


FIG. 12. Computed result obtained only with the global base grid: (a) density contour plots; (b) velocity vector plots.

this zone (inside the frame region) is specified by the solution that has just been obtained inside zone 2. Outside zone 2, χ values are zero. Solving the fortified equations, Eq.(3) the global region is solved with the zone 2 solution overlaid onto it in the region having large χ values. The solution process does not change but only open triangles are solved in reality. The process is repeated at each time step until the convergence is obtained for both zones for the steady-state problems. In case of unsteady flow problems, there exists time lag and the effect should be studied further, but the research so far indicated that the solution is reliable so far as the time step is taken to be small.

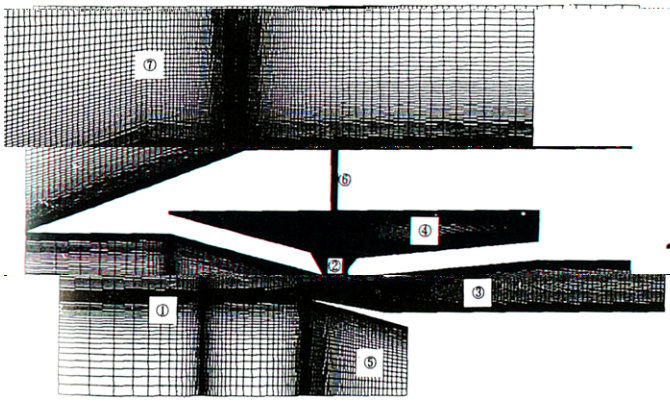


FIG. 13. Computational grid (seven slightly overlapped zones).

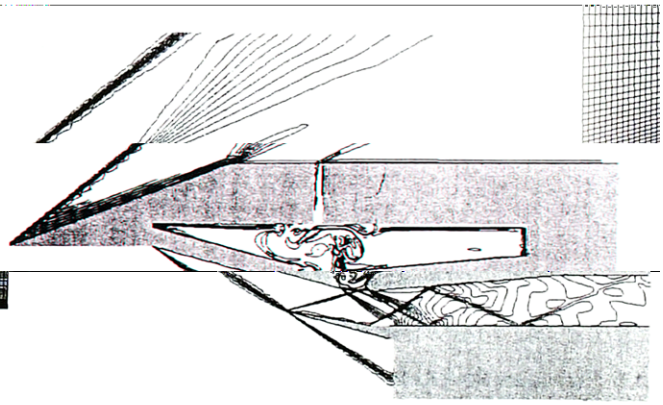


FIG. 14. Computed Mach number contour plots.

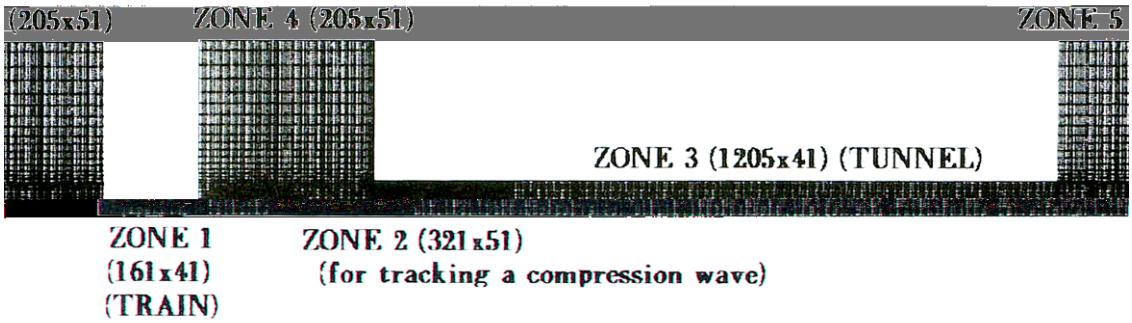


FIG. 15. Computational grid for the two-dimensional train moving into a tunnel.

FIG. 16. Time sequence of the computed pressure contour plots (M = 0.42; Rl = 0.20).

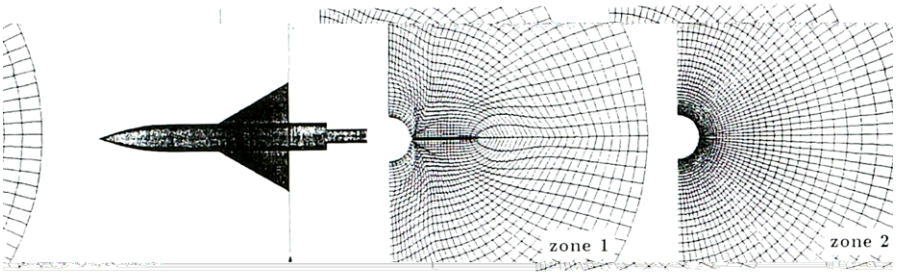


FIG. 21. Cross sectional grid surface at two streamwise stations.

The computational grids go through the body boundary and several grid points are located in the body geometry. It is necessary to let the solution process know which points correspond to the body boundary and inside the body. In the FSA zonal approach, we define χ values to be sufficiently large at the body boundary (denoted as filled triangles) so that the $\hat{Q}$ values are not updated there. As the $\hat{Q}$ values inside the body region (denoted as filled squares) should not be used for the computation, the χ values are specified to be negatively large inside the body region. Flags are defined and inserted in the MUSCL interpolation process so that the following condition is maintained. If the grid points having negative χ values are included in the grid points to be used for the MUSCL interpolation, the interpolation automatically goes down to first order without using those grid points. Thus, the MUSCL interpolation reduces to first order at the grid points just outside of the body boundary (denoted as open squares).

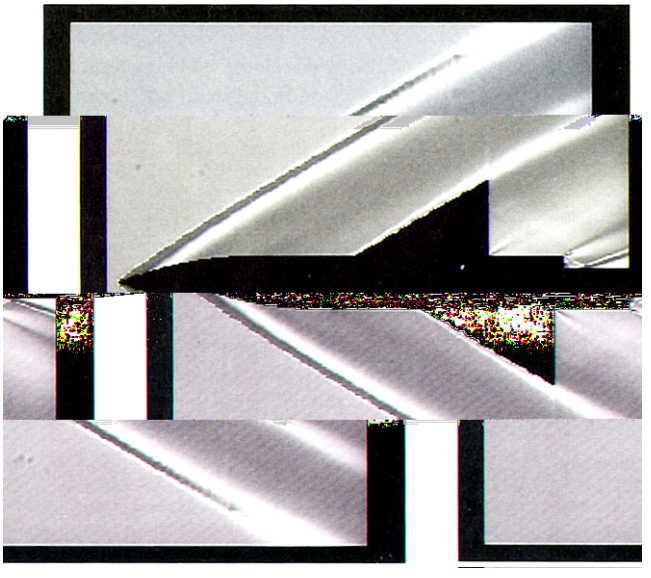


FIG. 22. Computed schlieren photograph based on the simulated result.

TEST PROBLEMS

The presence of the forcing terms basically blends different solution sets together when moderate χ values are used. Since both solution sets are reasonable solutions, they never contaminate the solution. Let us go back to Eq. (4). Define $\Delta\hat{Q}_r$ as

$$\Delta\hat{Q}_r = -h(\partial_\xi(\hat{E} - \hat{E}_v) + \partial_\eta(\hat{F} - \hat{F}_v))^n; \tag{7}$$

$\Delta\hat{Q}_r$ is the solution increment that would be obtained by a regular explicit time integration without any forcing terms. Also define $\Delta\hat{Q}_f$ as

$$\Delta\hat{Q}_f = \hat{Q}_f - \hat{Q}^n, \tag{8}$$

where $\Delta\hat{Q}_f$ is the solution increment for the forcing terms that would always give us $\hat{Q}^{n+1} = \hat{Q}_f$.

Using these two variables, Eq. (4) can be rewritten as

$$\Delta\hat{Q} = \frac{\Delta\hat{Q}_r + h\chi\Delta\hat{Q}_f}{I(1 + h\chi)}. \tag{9}$$

As illustrated in Fig. 5, the solution increment $\Delta\hat{Q}$ for the blended equations is a linear combination of the two solutions, $\Delta\hat{Q}_r$ and $\Delta\hat{Q}_f$. $\Delta\hat{Q}$ becomes $\Delta\hat{Q}_r$ when $\chi = 0$, and $\Delta\hat{Q}$ becomes $\Delta\hat{Q}_f$ when $\chi \gg 1$. When χ is moderate, the solution of the FSA equations becomes a linear combination of these two solutions. The same estimation is possible for the implicit time integrations. Therefore, forcing terms would not contaminate the solutions.

To confirm this, a simple example is taken. Blasius solutions for the viscous flow over a flat plate is computed with two zones, one for the global coarse grid and the other for the local fine grid near the flat plate. The outer boundary of the local fine grid is located somewhere in the boundary layer. When solving the local fine grid, the solution of the global zone at the previous time step is given at the outer boundary of the local fine grid. When solving the global zone, the solution inside the local fine grid zone is fortified by the fine-grid solution by specifying large χ values. The χ values are basically set to

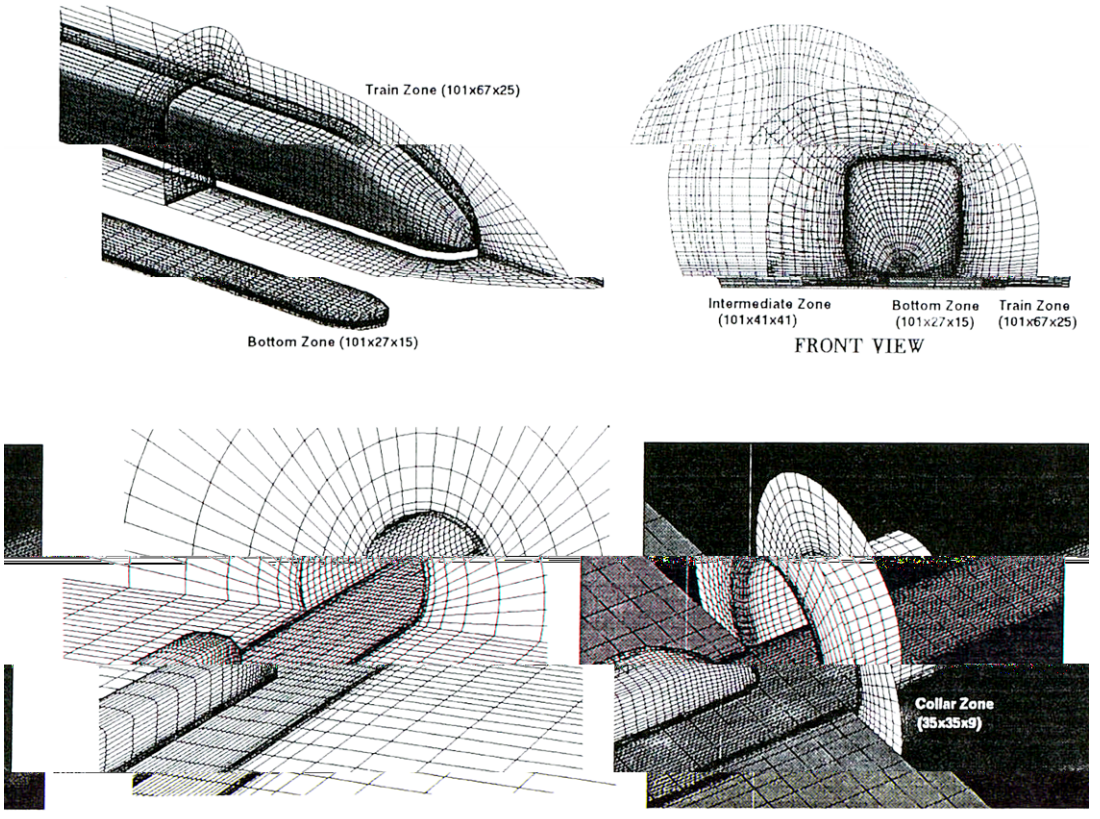


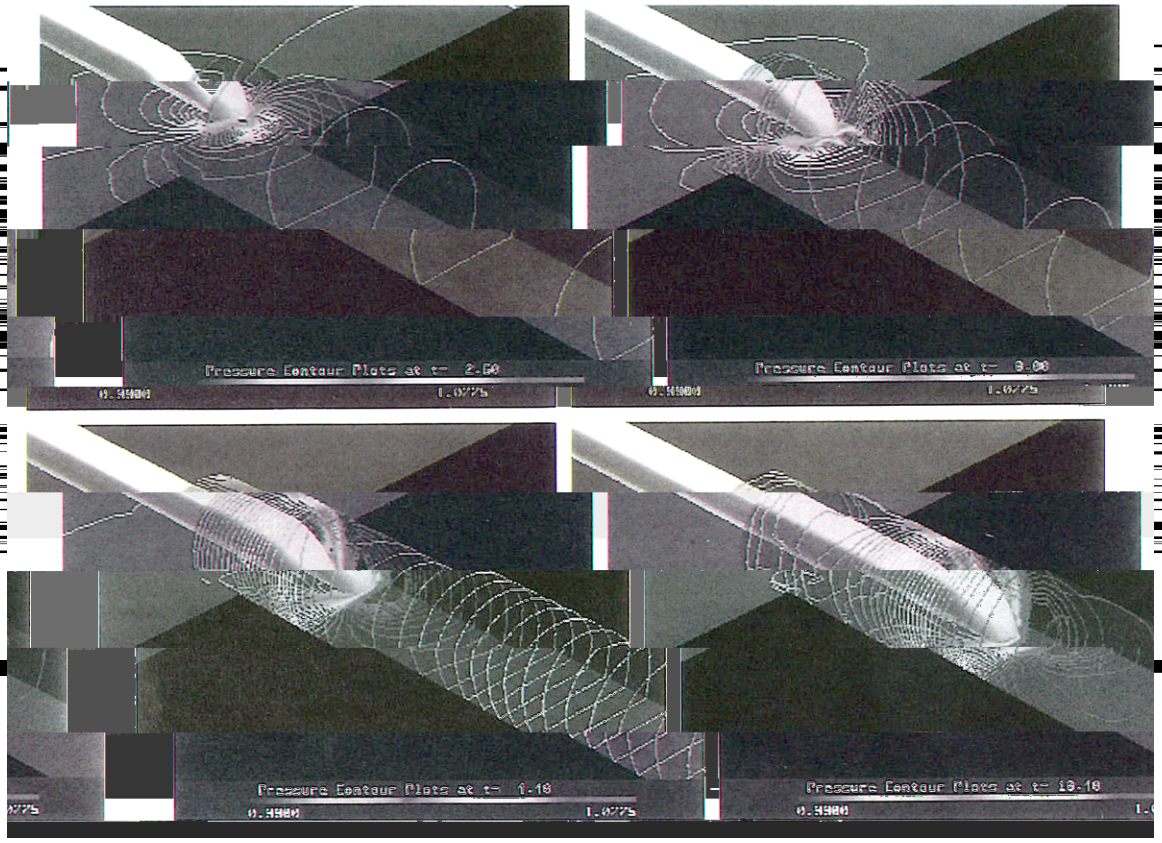
FIG. 23. Computational grid for the three-dimensional train entry problem.

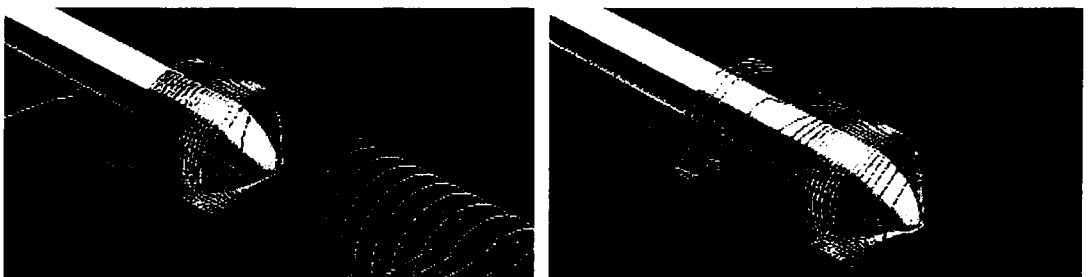
be 10^9, but to check the effect of different χ values, computations with χ to be 0 and 10^2 were tested. Figure 6 shows typical grid distributions used in this study; 101 grid points are distributed streamwisely for all the cases. In the normal direction, the number of the grid points for each zone and the location of the interface(outer boundary of the local fine grid) are parametrically changed. Two results are shown here among them. Figures 7(a)–(c) show the solution for the case with 21 grid points in the global and 16 points in the six-times finer local grid. Each corresponds to each χ values. When χ is 10^9, the computed result of u velocity for each zone shows the same value within the local fine zone because the solution for the global zone are fortified in this region. When χ is 0, the solutions for the global zone and the local zone become independent. The solution for the global zone is just the solutions for the global grid without taking any effect of the solution in the local fine grid. Small difference remains between the computed u velocities for each zone, but it may be difficult to notice in the figure. The same is true for $\chi = 10^2$.

The second case has 11 points in the global and 31 points in the local fine grid. The local fine grid is six times finer also in this case. Same three values of χ are tested and the resultant u velocity distributions are shown in Figs. 8(a)–(c). In this case, the fine grid region covers most of the boundary layer. When χ is 0, the solution of the global zone is independent from the fine grid solution because no points are fortified. There occurs a small discrepancy between the fine grid solution and global grid solution near the curvature of the profile. Since the solution in this region is fortified for the case of $\chi = 10^9$, all the coarse grid solution strictly agrees with the fine grid solutions. The solution for moderate χ values stays between.

In any of the computed results, the solutions were never contaminated even with the moderate χ distributions. For a moderate χ, the solution stayed between the non-fortified result and the fortified result.

With changing the χ values in the solution process dynamically, convergence to the steady state solution can be accelerated. Figure 9(a) shows the convergence history of the computation for the supersonic flow over a blunt body. A small local fine grid is defined near the body surface in addition to the global relatively coarse grid. Although not shown here, the local fine grid is two times finer in the streamwise direction and three times finer in the normal direction to the body. The solid line in Fig. 9(a) is the history of the solution residual for


Pressure Contour Plots at t= 2.50
Pressure Contour Plots at t= 5.00
Pressure Contour Plots at t= 7.50
Pressure Contour Plots at t= 10.00



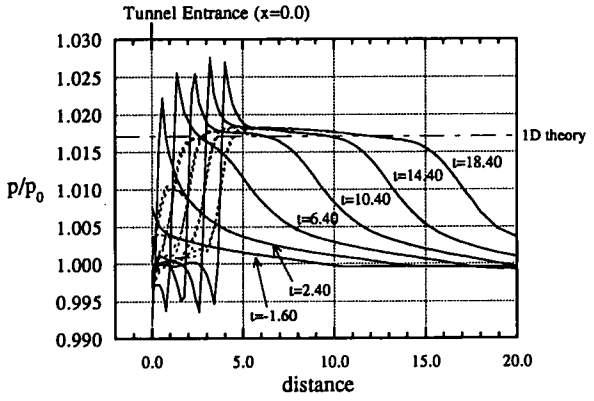


FIG. 25. Pressure distributions along the tunnel surface.

number is 2×10^5. The zonal grid distribution is presented in Fig. 10. A small zone to resolve the flow feature of the leading-edge separation vortex is created in addition to the global zone. In this small zone, the computational grid is twice as dense in the circumferential direction and three times as dense in the radial direction as the grid in the global zone. Figures 11(a) and (b) show the computed density and velocity vector contour plots, respectively. Dashed lines denote the interface of both zones. The contour lines are continuous at the interface, which indicates that the FSA method treated the interface properly. For comparison, the same plots from the solution obtained only using the global zone are shown in Figs. 12 (a) and (b). The same contour levels are used for fair comparison. The effect of the grid resolution enhancement is obviously observed from the clear definition of the rolling-up shear layer and the size of the secondary separation bubble near the leading edge. A bow shock wave exists outside of the plotted region as the freestream is supersonic. In the iteration process, this bow shock wave is created at the body and moves outward until the position is settled. In the present FSA method, the bow shock wave passed through the zonal interface smoothly. Three-dimensional delta wing computation as well as some of the additional two-dimensional computations can be found in Ref. [7].

The second example is a two-dimensional supersonic intake simulation [14]. In this intake model, a bleed chamber is attached in the top of the intake model to decrease the total-pressure loss. The flow field is decomposed into seven slightly overlapped zones (see Fig. 13) and Navier–Stokes equations are used for the simulation. In this example, the overlap grids at the interface in each zone coincide with each other. As no interpolation process is necessary, high-order accuracy is maintained at the interface by simply specifying χ distributions, which is one of the advantages of the FSA zonal interface scheme.

Figure 14 shows the instantaneous Mach number contour plots. The inflow Mach number is 3.0. The intake is designed to have three shock waves to obtain the proper pressure recovery. The first oblique shock wave was generated from the first ramp, the second oblique shock wave from the cowl lip, and the final terminal normal shock wave at the throat. In this simulation, the pressure at the exit is not specified and only oblique shock waves appear and the terminal shock wave is not created in the duct. A complicated flow pattern is observed, especially in the bleed chamber. The flow inside the duct is still unsteady but almost settled and does not change in time. On the other hand, strong unsteadiness remains in the bleed chamber. The computed contours are continuous at all the interfaces and indicate that the interface scheme does not cause any problems.

The next example is a train moving into a tunnel. The flow field analysis is important for the design of high-speed trains under development in the world. The computational grid is shown in Fig. 15. As has already been explained in the Introduction, there are five zones, two of which move. Some of the instantaneous density contour plots from the time evolution of the solution are presented in Figs. 16(a)–(d). Figures 16(a) and (b) are the view of the instantaneous pressure contour plots at the train entrance. Strong compressions and expansions act on the train. In Figs. 16(c) and (d), the pressure contours at later times are plotted to show the pressure wave propagations created in front of the train in the tunnel. The zonal method does not show any problems and is clearly useful for this simulation. More about the moving train simulation is presented in other papers [15, 16].

Another practical application is a blast wave simulation. This type of blast wave is created by a sudden explosion of the nuclear or chemical propellant etc. Since the blast wave energy is supplied only at the initial stage, the blast wave becomes weakened as it propagates. Therefore, there is no danger if the populated area is located far enough from the center of explosion. Such a distance is called a safety distance. For the estimation of the safety distance, the blast wave strength should be accurately evaluated by the numerical simulation. The length scale is of the order of *miles* and the simulation has to capture the weak pressure wave of the order of 10^{-1} psi level. Our experience indicated that the grid resolution had to be tremendously enhanced to resolve such weak pressure wave propagation. However, an important phenomenon occurs within the restricted region which moves with the blast wave but, thus, the region that requires fine grid distribution is small. Here, to enhance the grid resolution near the frontal blast wave, the overlapped zonal grid that moves with the blast wave is used for the simulation. Figure 17 shows the schematic picture of the grid distributions at three time stages. In addition to the moving grid, one more zonal grid is prepared to capture the physical phenomenon near the center of the explosion. The overset moving grid does not necessarily align with the global grid. Figure 18 shows the result of this zonal grid compared with those of the two single-zone grids. Rapid increase of the pressure can be recognized in the zonal solution, whereas only a gradual increase can be observed in the single-zone solutions. More importantly, the pressure increase due to the blast wave

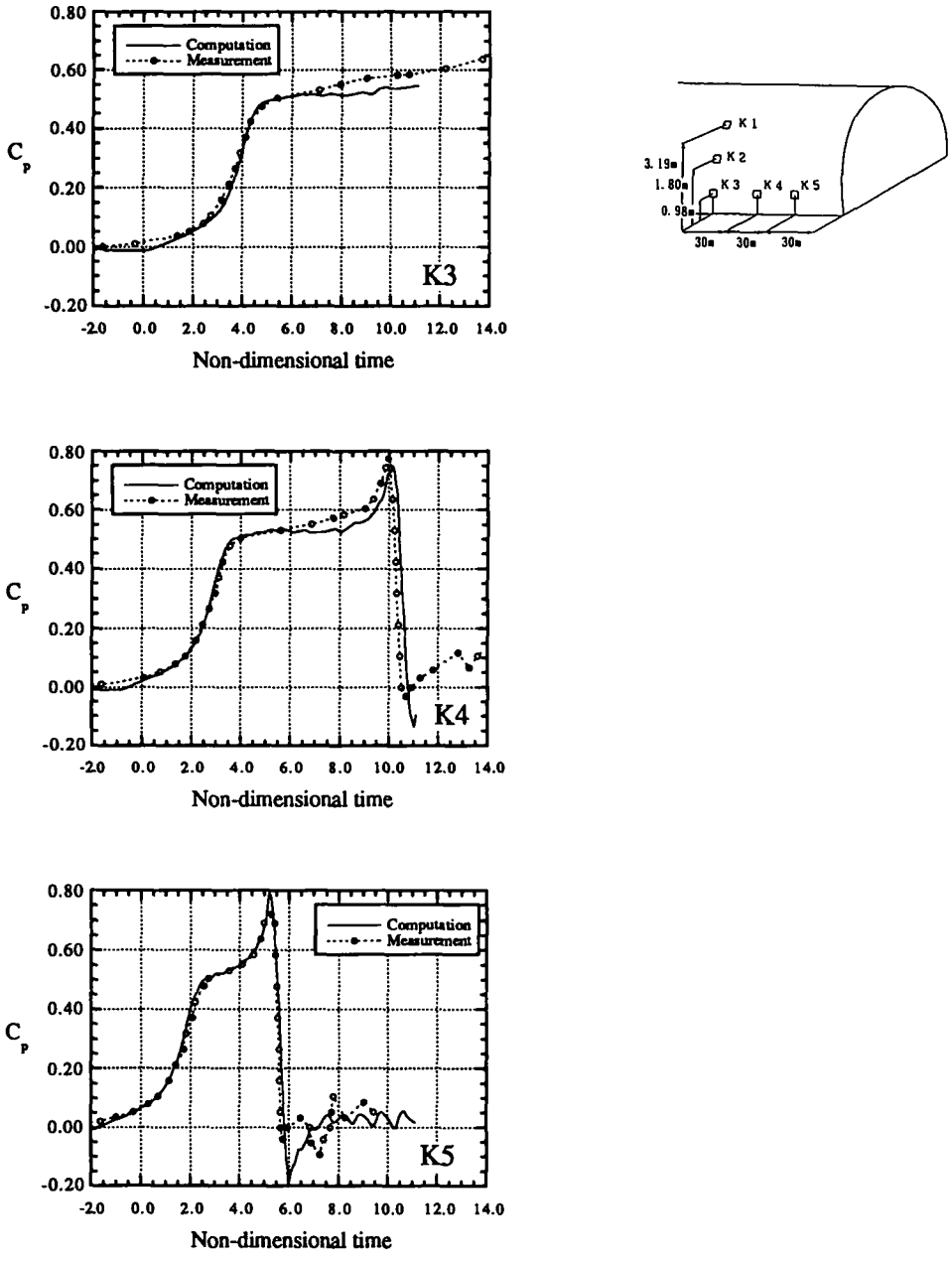


FIG. 26. Time history of the pressure coefficients at the tunnel walls.

is much lower in the single-zone solutions due to the numerical dissipation. The zonal solution captures a frontal shock wave within a few grid points and shows a much sharper shock wave, although the total number of the grid points in this zonal computation is about $\frac{1}{3}$ that of the finer single grid. Suppose that we used the solution by the single-zone computations that did not have sufficient grid resolution; the sudden pressure increase due to the frontal shock wave would be evaluated much smaller. Then, the safety distance would be estimated much longer, which is dangerous. Computational solutions are easily obtained, but obtaining the reliable solution is another story. The zonal method is really useful for obtaining accurate and reliable solutions for practical problems. The details and more practical applications including the effect of the ground surface geometry are found in another paper [17].

The two examples above are unsteady flow simulations. Time derivative terms are included in the evaluation of metrics for the coordinate transformation between the physical space and the computational space. The time derivative terms are evaluated by the first-order approximation. This did not cause any

problems on the two examples above, even when the local grid begins to move, but they should be carefully evaluated when the time step is relatively large and grid movement includes rapid acceleration. The other thing to note is the GCL (geometry conservation law). These terms should be evaluated so that change of the grid geometry does not violate the conservation law. The detail can be found in Refs. [18, 19].

The following two examples are three-dimensional problems. The first one is a three-dimensional supersonic flow over a wing-fuselage combination [20]. The body configuration is an AGARD-B model used for the calibration of the wind tunnel at the ISAS (Institute of Space and Astronautical Science). As is shown in Fig. 19, which is a schlieren photograph in the experiment ($M_\infty = 2.0$), the test model is supported by the sting from the rear portion. The flow simulation was conducted including this sting. The flow field is divided into three portions as is shown in Fig. 20. The first portion is from the front part until the end of the wing, the second portion is the outer region of the rear part, and the third portion is an inner region. The O-type cylindrical grid is used for all the zones, but interpolation is necessary to transfer the information through the interface, as is shown in Fig. 21. The total grid points number more than one million. Figure 22 shows the corresponding numerical result where the computed schlieren photograph is obtained by simulating the schlieren technique on the computed data [21]. The separated region at the sting junction is well simulated.

The final example is a three-dimensional version of the tunnel entry problem. Figure 23 shows several views of the computational grid for this problem. The total number of grid points is about 530,000. There are six zones. The main zones are the region before the tunnel, the region of the tunnel, and the region of the train. In addition, three more zones are created. One is the region between the bottom of the train and the tunnel floor. This region is prepared for the topological view point. The second is the intermediate zone between the tunnel zone and the train zone. This zone moves with the train speed. Since this problem is unsteady, zone-to-zone interpolation is necessary at each time step and it is time consuming. As the computational grid in this zone has an x = const plane, as well as the tunnel grid, only two-dimensional-like interpolation is necessary between this zone and the tunnel zone. As the intermediate zone moves with the train, the interpolation between this zone and the train zone is independent of the train movement and can be prepared in advance. Using this intermediate zone, much computer time necessary for the interpolation at each time step is saved. The third zone is the so-called collar grid at the corner of the tunnel entrance. This zone avoids the singularity of the

Figure 25 shows the growth of the pressure wave in terms of the pressure distribution along the x-axis. As time goes on, gradient of the pressure in front of the train becomes steeper. As the figure says, the pressure peak can be predicted by a simple quasi-one-dimensional simulation that only considers the time change of the cross sectional area. However, our experience also indicated that the dp/dt, which is a key parameter for the booming noise cannot be predicted by a one-dimensional simulation, and multi-dimensional simulations are necessary. The train shape, similar to the current Shinkansen train called "Nozomi," was selected, since the pressure data on the tunnel wall obtained in the field measurement was available. Figure 26 is the time history of the Cp at several points (shown on the sketch) on the tunnel wall. The agreement is pretty good when the many environmental effects in the field measurement are considered.

CONCLUSIONS

A unified zonal method was developed. Overset and overlap zonal methods are treated in the same way by adopting the interface scheme based on the FSA concept. The multi-block approach that is suitable for some problems can also be treated similarly under this concept. It has been shown that the interface scheme is simple and implementation of the present zonal method into existing implicit or explicit single-zone codes is easy. Some of the test cases showed that the FSA zonal method is more flexible than the existing overlaid zonal methods. Based on the application examples, it was shown that the developed unified zonal method can be an effective CFD tool for complex flow physics and complex body configurations.

REFERENCES

1. A. Jameson and T. J. Baker, AIAA Paper 87-0542, Jan. 1987 (unpublished).
2. R. Löhner, AIAA Paper 87-0555, Jan. 1987 (unpublished).
3. J. A. Benek, P. G. Buning, and J. L. Steger, in *Proceedings, AIAA 7th Computational Fluid Dynamics Conference, Cincinatti, Ohio, AIAA Paper* 85-1523-CP, 1985, edited by U. Ghia (AIAA, Washington, DC, 1985), p. 322.
4. P. G. Buning, I. T. Chiu, S. Obayashi, Y. M. Rizk, and J. L. Steger, AIAA Paper 88-4359, Aug. 1988 (unpublished).
5. K. Nakahashi, and G. S. Deiwert, *AIAA J.* **25**(4), 513 1987.
6. A. T. Hsu, AIAA Paper 89-0006, Jan. 1989 (unpublished).
7. K. Fujii, AIAA Paper 88-2562, June 1988 (unpublished).
8. M. Berger, and J. Oliger, *J. Comput. Phys.* **53**, 484 (1987).

12. P. Roe, *Annu. Rev. Fluid Mech.* (1986) 337.
13. W. R. Van Dalsem and J. L. Steger, AIAA Paper 87-2279, Aug. 1987 (unpublished).
14. S. Kuroda, and K. Fujii, AIAA Paper 91-1730, June 1991 (unpublished).
15. T. Ogawa, and K. Fujii, *Comput. Fluid Dynamics J.* **3**(1), 63 (1994).
16. T. Ogawa, and K. Fujii, in *Proceedings, The International Conference on Speedup Technology for Railway and Maglev Vehicles (STECH), Nov.* 1993.
17. F. Shimizu, K. Fujii, and F. Higashino, *Trans. Japan Soc. Aeronaut. Space Sci.* **36**(111), 36 (1993).
18. P. D. Thomas and C. K. Lombard, *AIAA J.* **17**(10), 1030 (1979).
19. S. Obayashi, NASA CR 177572, Jan. 1991 (unpublished).
20. K. Fujii, Y. Tamura, and S. Kuroda, in *Proceedings, AIAA 10th Computational Fluid Dynamics Conference, Honolulu, Hawaii, AIAA Paper 91-1558-CP,1991,* edited by P. Rubbert (AIAA, Washington, DC, 1991), p. 339.
21. Y. Tamura and K. Fujii, AIAA Paper 90-3031, Aug. 1991 (unpublished).