JOURNAL OF COMPUTATIONAL PHYSICS 98, 129–144 (1992)

Numerical Treatment of Grid Interfaces for Viscous Flows

YANNIS KALLINDERIS

Department of Aerospace Engineering and Engineering Mechanics, University of Texas at Austin, Austin, Texas 78712

Received February 23, 1990; revised January 27, 1991

Numerical treatment of grid interfaces is one of the most important considerations for algorithms that employ different grids within the computational domain. The issue of numerical treatment of quadrilateral grid interfaces with a representative finite-volume Navier–Stokes integration scheme is addressed. Interfaces are created by local embedding of quadrilateral grids and are the borders between different grids. Grid embedding is one of the basic functions of adaptive algorithms that have been developed in order to increase both accuracy and efficiency of computations. The present work both develops and investigates interface treatment schemes that have certain properties such as accuracy and conservation. It is a novel study of interfaces for the case of viscous flow computations. Various treatments are proposed and evaluated with the emphasis being on a comparison between accurate and conservative treatments. Two methodologies have been followed in order to study interface treatments. The first is analytical and yields orders of possible numerical errors, while the second approach employs model test cases, which are especially designed to evaluate certain aspects of the described interface treatments. Also, a transonic airfoil flow case is included as an example of accuracy and robustness of a particular interface treatment scheme. All numerical treatment schemes that are discussed have been coded and evaluated.

1. INTRODUCTION

Grid embedding consists of dividing grid cells locally so that the truncation error is relatively small and equally distributed throughout the solution domain. This is accomplished by increasing grid solution locally in regions in which flow features exist. In this way, computations with grid embedding require fewer computer resources than when a globally refined grid is employed [1–6].

Embedding of quadrilateral cells introduces internal boundaries between cells with either different refinement levels or with different types of subdivisions. Grid interfaces in turn can be categorized into two main groups depending upon the type of cell division. The first are interfaces which are characterized by an abrupt change in cell size only. The grid is continuous across the interface, but cell metrics change as shown in Fig. 1a (*metric discontinuous grid*). The second type includes grid lines which actually are interrupted by interfaces as illustrated in Fig. 1b (*discontinuous grid*). In this case cells on the coarser side of the interface may contain additional nodes at the face midpoint.

Numerical treatments for interfaces have been examined in [7, 8] for the potential equation and in [9–13] for the Euler equations. The present paper presents a novel study of grid interfaces for the case of the Navier–Stokes equations. There are a number of problems which are imposed on the integration scheme due to the presence of interfaces. The sudden change in grid size introduces a significant stretching error, which may result in a reduction of order of accuracy for the scheme. Existing schemes have been developed for cells with nodes at only the four cell corners, and they require some modification in order to take into account the extra face nodes. Another important issue is maintaining conservation across interfaces. The fluxes across the boundaries surrounding an interface cell should cancel one another in order for the scheme to be conservative in a steady state situation. Also, the form of the artificial dissipation operator at an interface (convective, dissipative) deserves special consideration. Other important issues are coding complexity and the ease with which an interface treatment scheme can be extended to three dimensions.

It is clear that these considerations impose serious limitations on construction of an interface scheme and that in many cases the above requirements contradict one another. In fact, simultaneous achievement of both conservation and accuracy is very difficult and even impossible in most cases. However, not all of the above requirements are important for a specific interface. For regions in which solution variations are relatively small, reduction in order of accuracy has a negligible effect on numerical results. Conservation proves to be an important property mostly in cases of moving shocks for accurate prediction of their location and speed. Conversely, accuracy is more of an issue in a boundary layer, since the second-order derivatives (viscous terms) are important and are more "sensitive" to grid stretching error

0021-9991/92 $3.00

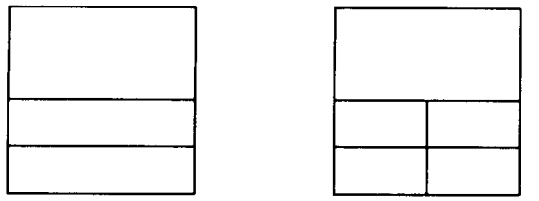


FIG. 1. Types of grid interfaces: (a) metric discontinuous; (b) discontinuous.

than first-order derivatives (inviscid terms). Ni's scheme is a typical central-space differencing scheme and faces the same interface problems that other schemes, which discretize spatial derivatives with central differences, face [14–16].

Two methodologies have been followed in order to study interface treatments. The first is analytical and yields orders of possible numerical errors, while the second approach employs model test cases, which are especially designed to evaluate certain aspects of the described interface treatments. Also, the effect on solution of interfaces that are located within regions of significant flow gradients is monitored. All treatments that are discussed have been coded and evaluated.

In the following, the Navier–Stokes algorithm that was employed is described briefly. Then, an accurate but nonconservative treatment is presented and investigated. The nonconservation errors which are associated with inviscid, viscous, and smoothing terms are evaluated, followed by a numerical test case. Then a way of deriving conservative treatments is presented. Each one of the operators that constitute the scheme, namely, the inviscid, viscous, and smoothing terms, are examined separately. The problems of stretching error and of the convective form, which the smoothing operators take for this treatment, are investigated. Then, the two interface treatments are compared, together with a hybrid third treatment, which is a combination of the two. Finally, a transonic airfoil flow case is presented in order to investigate both accuracy and robustness of a specific interface treatment.

2. NAVIER–STOKES ALGORITHM

An explicit, finite volume scheme is employed in order to integrate the full Navier–Stokes equations. The state variables are stored at grid nodes, and both second as well as fourth order artificial dissipation is employed in order to capture shocks and to suppress odd–even decoupling of the solution. The scheme is typical of a class of finite-volume, explicit Navier–Stokes solvers [17, 18], and therefore, both the present approach of grid interface study and the conclusions can be applicable to those other schemes, as well.

The purpose for describing the algorithm is to present the formulas to be used in the interface study. The overall scheme is described briefly, since the emphasis is on treatment of grid interfaces. For more details on the algorithm the reader is referred to [1, 2].

The system of the two-dimensional Navier–Stokes equations may be written in cartesian two-dimensional conservation form as

$$\frac{\partial U}{\partial t}+\frac{\partial F}{\partial x}+\frac{\partial G}{\partial y}=\frac{\partial R}{\partial x}+\frac{\partial S}{\partial y}, \tag{1}$$

where

$$U=\begin{pmatrix}\rho\\ \rho u\\ \rho v\\ E\end{pmatrix},\qquad F=\begin{pmatrix}\rho u\\ \rho u^2+p\\ \rho uv\\ (E+p)u\end{pmatrix},\qquad G=\begin{pmatrix}\rho v\\ \rho uv\\ \rho v^2+p\\ (E+p)v\end{pmatrix}$$

are state and convective flux vectors in the x- and y-directions, respectively. The viscous flux vectors are

$$R=\begin{pmatrix}0\\ \tau_{xx}\\ \tau_{xy}\\ u\tau_{xx}+v\tau_{xy}-q_x\end{pmatrix},\qquad S=\begin{pmatrix}0\\ \tau_{xy}\\ \tau_{yy}\\ u\tau_{xy}+v\tau_{yy}-q_y\end{pmatrix},$$

where τ_{xx}, τ_{yy}, τ_{xy} are viscous stresses, and q_x, q_y are heat conduction terms. In the above relations, ρ is density, u and v are velocity components, E is total internal energy per unit volume, p is pressure, and T is the temperature.

2.1. *Inviscid Terms*

A one-step Lax–Wendroff-type integration scheme due to Ni [19] has been employed for discretization of the convective terms of the Navier–Stokes equations. Omitting the viscous terms, the Euler approximation is

$$\frac{\partial \mathbf{U}}{\partial t}+\frac{\partial \mathbf{F}}{\partial x}+\frac{\partial \mathbf{G}}{\partial y}=0, \tag{2}$$

where $\mathbf{U}$ is the state-variable vector, and $\mathbf{F}$, $\mathbf{G}$ are the convective terms flux vectors as defined in (1). Integration of the above relation over the area of a cell, leads to the following equivalent integral relation:

$$\frac{d}{dt}\iint_{\text{cell area}}\mathbf{U}\,dx\,dy+\oint_{\text{cell faces}}(\mathbf{F}\,dy-\mathbf{G}\,dx)=0. \tag{3}$$

The first term in the above relation represents the change in time of the state vector over the cell area S, and is discretized as $(\Delta\mathbf{U}/\Delta t)S$, where $\mathbf{U}$ is the state-vector value at the center of a cell, and $\Delta\mathbf{U}=\mathbf{U}^{n+1}-\mathbf{U}^n$ is its change over

one time step. The second term in (3), represents the convective fluxes across the cell faces and is computed via the trapezoidal integration rule, which is second-order accurate.

The state-vector change in time $\Delta\mathbf{U}$ at the center of the cell has to be distributed to the corners. Consider the change $\delta\mathbf{U}$ at a grid node and a second-order Taylor series expansion in time:

$$\delta\mathbf{U} = \mathbf{U}^{n+1} - \mathbf{U}^n \cong \Delta t \left(\frac{\partial \mathbf{U}}{\partial t}\right)^n + \frac{(\Delta t)^2}{2}\left(\frac{\partial^2 \mathbf{U}}{\partial t^2}\right)^n. \quad (4)$$

The second-order time term in (4) is calculated following the Lax–Wendroff [20] approach which replaces time derivatives with spatial derivatives from the governing equation.

Finally, the following formulas for the distribution of the time change $\Delta\mathbf{U}$ to the cell corners, are derived,

$$(\delta U)_{sw} = \tfrac{1}{4}\{\Delta U - \Delta f - \Delta g\} \quad (5a)$$

$$(\delta U)_{nw} = \tfrac{1}{4}\{\Delta U - \Delta f + \Delta g\} \quad (5b)$$

$$(\delta U)_{ne} = \tfrac{1}{4}\{\Delta U + \Delta f + \Delta g\} \quad (5c)$$

$$(\delta U)_{se} = \tfrac{1}{4}\{\Delta U + \Delta f - \Delta g\}, \quad (5d)$$

where

$$\Delta f \equiv \frac{\Delta t}{S}(\Delta\mathbf{F}\,\Delta y^l - \Delta\mathbf{G}\,\Delta x^l) \quad (6a)$$

$$\Delta g \equiv \frac{\Delta t}{S}(\Delta\mathbf{G}\,\Delta x^m - \Delta\mathbf{F}\,\Delta y^m) \quad (6b)$$

$$\Delta f \equiv \left(\frac{\partial \mathbf{F}}{\partial \mathbf{U}}\right)\Delta\mathbf{U}, \qquad \Delta g \equiv \left(\frac{\partial \mathbf{G}}{\partial \mathbf{U}}\right)\Delta\mathbf{U}.$$

The subscripts *sw*, *nw*, *ne*, *se* denote the four corners of a grid cell. The cell-metric terms Δx^l, Δy^l, Δx^m, Δy^m are the x, y-dimensions along the two directions of a cell that are denoted by superscripts l, m.

2.2. *Viscous Terms*

The above described Lax–Wendroff-type scheme for the discretization of inviscid terms can be extended to include viscous terms as well. The change in time of the state variables, which is contributed to by the viscous terms, is also given by the Taylor series expansion (4). Only the first-order temporal term in the Taylor expansion (4) is kept, which yields a scheme that is first-order accurate in time and which eliminates the need to compute viscous term Jacobians. The development of the scheme starts with a consideration of the viscous part of the Navier–Stokes equations in integral form over a grid cell:

$$\iint \frac{\partial \mathbf{U}}{\partial t}\,dx\,dy = \iint \left(\frac{\partial \mathbf{R}}{\partial x} + \frac{\partial \mathbf{S}}{\partial y}\right) = \oint (\mathbf{R}\,dy - \mathbf{S}\,dx). \quad (7)$$

The terms represented by $\mathbf{R}$, $\mathbf{S}$ include stress and heat conduction terms.

The unsteady term in (7) is discretized as $(\Delta\mathbf{U}_{\text{vis}}/\Delta t)S$. The term $\Delta\mathbf{U}_{\text{vis}}$ represents the change in time of the state-vector at a grid node due to the viscous terms only. After some manipulations it can be found that the cell contributions to the change in time $\Delta\mathbf{U}_{\text{vis}}$ of state-vector $\mathbf{U}$, due to the viscous terms at the four cell corners, are given by the formulas

$$(\Delta U_{\text{vis}})_{sw} = \frac{\Delta t}{S}\{(+\mathbf{R}^S\,\Delta y^l - \mathbf{R}^W\,\Delta y^m) - (+\mathbf{S}^S\,\Delta x^l - \mathbf{S}^W\,\Delta x^m)\} \quad (8a)$$

$$(\Delta U_{\text{vis}})_{nw} = \frac{\Delta t}{S}\{(+\mathbf{R}^N\,\Delta y^l + \mathbf{R}^W\,\Delta y^m) - (+\mathbf{S}^N\,\Delta x^l + \mathbf{S}^W\,\Delta x^m)\} \quad (8b)$$

$$(\Delta U_{\text{vis}})_{ne} = \frac{\Delta t}{S}\{(-\mathbf{R}^N\,\Delta y^l + \mathbf{R}^E\,\Delta y^m) - (-\mathbf{S}^N\,\Delta x^l + \mathbf{S}^E\,\Delta x^m)\} \quad (8c)$$

$$(\Delta U_{\text{vis}})_{se} = \frac{\Delta t}{S}\{(-\mathbf{R}^S\,\Delta y^l - \mathbf{R}^E\,\Delta y^m) - (-\mathbf{S}^S\,\Delta x^l - \mathbf{S}^E\,\Delta x^m)\}, \quad (8d)$$

where the subscripts *sw*, *nw*, *ne*, *se* denote the four cell corners, and the superscripts *S*, *W*, *N*, *E* represent the four cell faces. For example, the viscous term $\mathbf{R}^S$ denotes that $\mathbf{R}$ is evaluated at the south face. The above spatial discretization is second-order accurate for a uniform mesh [1].

2.3. *Artificial Dissipation*

Smoothing that is accomplished by explicitly adding dissipation is employed by the vast majority of existing schemes, especially those concerned with compressible flows. There are two main types of such artificial smoothing: one is used to capture shocks; the second is designed to damp spurious oscillations throughout the field and to suppress odd–even decoupling of the solution.

Second-order (shock) smoothing. Second-order smoothing provides damping necessary to smear a shock, which ideally has zero thickness, in such a way that oscillation are avoided. Since such smoothing is required only in shock regions, a switch is employed to turn it off elsewhere.

In two dimensions, the damping term may have the form:

$$\left|\frac{\partial P}{\partial x}\right| \cdot \frac{\partial u}{\partial x} + \left|\frac{\partial P}{\partial y}\right| \cdot \frac{\partial u}{\partial y}. \quad (9)$$

Consider the specific discretization for a cell-vertex scheme

at node 0 (Fig. 2). The node receives contributions from each of the four surrounding cells; that from cell A is

$$S_{0A}^{(2)} = +\left|\frac{(P_3+P_5)-(P_2+P_0)}{P_3+P_5+P_2+P_0}\right| \times\{(u_3+u_5)-(u_2+u_0)\} + \left|\frac{(P_2+P_3)-(P_0+P_5)}{P_2+P_3+P_0+P_5}\right| \times\{(u_2+u_3)-(u_0+u_5)\}. \quad (10)$$

Similarly, from cell D it is

$$S_{0D}^{(2)} = +\left|\frac{(P_1+P_4)-(P_2+P_0)}{P_1+P_4+P_2+P_0}\right| \times\{(u_1+u_4)-(u_2+u_0)\} + \left|\frac{(P_1+P_2)-(P_4+P_0)}{P_1+P_2+P_4+P_0}\right| \times\{(u_1+u_2)-(u_4+u_0)\}. \quad (11)$$

Similar expressions furnish contributions from cells C and B. Pressure differences in the switch are normalized by the sum of the pressures at the four corners of each cell. The shock smoothing operator introduces a first-order error to the solution at the shock region.

Fourth-order smoothing. Fourth-order smoothing is used in order to suppress odd–even modes and to damp spurious oscillations. The operator is formed in two steps. The second-order difference operator is formed in the first step (Fig. 2):

$$\begin{aligned} D_{0A}^2 &= u_0+u_2+u_3+u_5-4u_0 \\ D_{0B}^2 &= u_7+u_0+u_5+u_8-4u_0 \\ D_{0C}^2 &= u_6+u_4+u_0+u_7-4u_0 \\ D_{0D}^2 &= u_4+u_1+u_2+u_0-4u_0. \end{aligned} \quad (12)$$

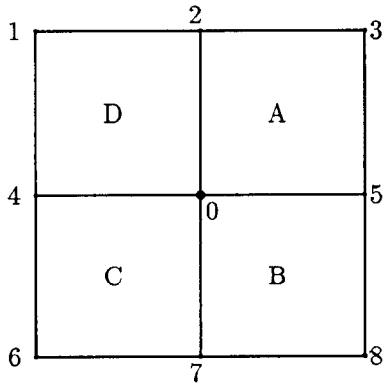


FIG. 2. Node and cell designations for smoothing operators.

The second step duplicates the first, replacing state variables by second-order differences from the first step:

$$\begin{aligned} -D_{0A}^4 &= D_0^2+D_2^2+D_3^2+D_5^2-4D_0^2 \\ -D_{0B}^4 &= D_7^2+D_0^2+D_5^2+D_8^2-4D_0^2 \\ -D_{0C}^4 &= D_6^2+D_4^2+D_0^2+D_7^2-4D_0^2 \\ -D_{0D}^4 &= D_4^2+D_1^2+D_2^2+D_0^2-4D_0^2. \end{aligned} \quad (13)$$

The operator introduces a third-order error into the solution.

2.4. *Conservation*

The Navier–Stokes equations express conservation of mass, momentum, and energy. The flow domain is divided into smaller volumes, and the numerical scheme then is applied on each of the cells. For global accuracy of the scheme, it is important to conserve the above flow quantities over each individual cell and therefore over the entire domain. Schemes that accomplish this are said to have the *conservation property.*

A definition of conservation can be derived by considering the Navier–Stokes equations in integral form over the computation domain:

$$\frac{d}{dt}\iint_{\text{domain}} \mathbf{U}\,dx\,dy = -\oint_{\text{domain boundary}} \mathbf{F}\,dy - \mathbf{G}\,dx + \oint_{\text{domain boundary}} \mathbf{R}\,dy - \mathbf{S}\,dx. \quad (14)$$

The domain is divided into smaller control surfaces, the cells, and the above expression leads to the following general definition of conservation:

$$\Sigma_{\text{nodes}} \frac{1}{\Delta t_{\text{node}}} \cdot \delta \mathbf{U}_{\text{node}} \cdot A_{\text{node}} = \text{boundary terms.} \quad (15)$$

In the above, A_{node} and Δt_{node} are suitable area and time-step which surround the grid nodes, and they satisfy the property

$$\Sigma_{\text{nodes}} A_{\text{node}} = A_{\text{domain}}.$$

For the present scheme, the change $\delta \mathbf{U}_{\text{node}}$ at each node is the sum of the contributed changes from the adjoining cells, as given by formulae (5) and (8). Therefore, the definition of conservation can be written as

$$\Sigma_{\text{cells}} \frac{1}{\Delta t_{\text{node}}} (\Sigma_1^4 \delta \mathbf{U}_{\text{corner}} \cdot A_{\text{cell}}) = \text{boundary terms.} \quad (16)$$

Using distribution formulae (5) and (8), the above equation yields: $\Sigma \Delta \mathbf{U}$ = boundary terms, that is simply the sum of the

flux balances over all the cells, which consists of boundary terms only.

The second- and fourth-order smoothing operators are not scaled with the term $\Delta t/S$, and therefore the appropriate definition of conservation is

$$\Sigma_{\text{nodes}}\, \delta \mathbf{U}_{\text{node}} = \text{boundary terms.} \tag{17}$$

The distribution formulae (10), (11), (13) satisfy the above equation, since $\Sigma_1^4\, \delta \mathbf{U}_{\text{corner}}$ is zero, and therefore the smoothing operator is conservative, as well.

3. ACCURATE INTERFACE TREATMENT

Consider the two types of discontinuous and metric discontinuous grids of Fig. 3 and the interface node a. For both types, cells A_1, A_2, A_3, A_4 would have normally been employed in order to evaluate the inviscid, viscous, and smoothing contributions to node a. Cells A_3, A_4 are embedded fine cells, while cells A_1, A_2 are unembedded coarse cells with vertical dimensions twice those of cells A_3, A_4. It is clear that an evaluation, for example, of the viscous derivative u_{yy} at node a,

$$(u_{yy})_a = \frac{u_{a1} - 2u_a + u_{a2}}{(y_{a1} - y_{a2})^2},$$

suffers from a severe stretching error. The order of accuracy reduces to first. In order to alleviate these problems, use is made of the "parent" cells A_1', A_2', with roughly the same size as the coarse cells A_1, A_2. The expression for u_{yy} now becomes

$$(u_{yy})_a = \frac{u_{a1} - 2u_a + u_{a1'}}{(y_{a1} - y_{a1'})^2}$$

(Fig. 3), which is second-order accurate. In the case of a discontinuous interface, the treatment ignores the interface face node c, and the state-variable values at the node are obtained through interpolation from corresponding values at interface nodes b, a. The above also "ignores" the type of division (directional or two-directional) of the interface cells, and both discontinuous and metric discontinuous types of interfaces are treated identically, which simplifies the interface algorithm significantly. This is very important, if extension of the algorithm to three dimensions is of interest.

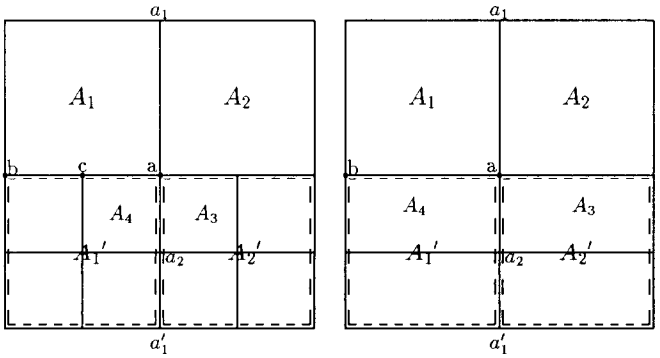


FIG. 3. Interface treatment without stretching error.

3.1. *Nonconservation Error*

However, the above does suffer from a nonconservation error. Use of "parent" cells A_1', A_2' instead of fine cells A_3, A_4, as well as the averaging that is used for the middle interface node c, introduces a nonconservation error. This error will now be evaluated by separately considering the inviscid, viscous, and smoothing parts of the scheme. The integration would be conservative if the sum $E \equiv \Sigma_{\text{nodes}}(S/\Delta t)\, \delta U$ consisted only of contributions from the boundary nodes (see Eq. (15)). Consider the interface configuration of Fig. 4. Using distribution formulas for the inviscid terms (5) and calculating the above sum over nodes 1 through 11, one obtains

$$\begin{aligned} E = &+ \frac{S_A}{\Delta t_A}\left(\frac{5}{4}\Delta U_A - \frac{1}{4}\Delta g_A\right) + \frac{S_B}{\Delta t_B}\frac{1}{2}(\Delta U_B - \Delta g_B) \\ &+ \frac{S_C}{\Delta t_C}\frac{1}{2}(\Delta U_C - \Delta g_C) + \frac{S_D}{\Delta t_D}\Delta U_D + \frac{S_E}{\Delta t_E}\Delta U_E \\ &+ \frac{S_{A'}}{\Delta t_{A'}}\left(\frac{3}{4}\Delta U_{A'} + \frac{3}{4}\Delta g_{A'}\right). \end{aligned} \tag{18}$$

In the above expression S, Δt are the area and time-step of

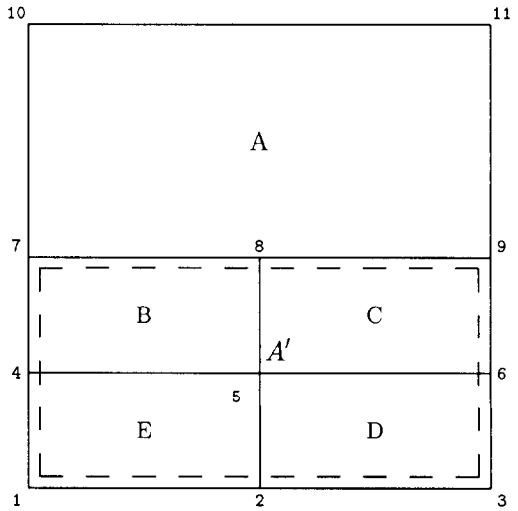


FIG. 4. Interface configuration for evaluation of nonconservation error.

each cell. The terms Δg are the second-order inviscid distribution terms given by formulas (6). For example,

$$\Delta g_A = \frac{\Delta t_A}{S_A}\left\{\left(\frac{\partial G}{\partial U}\right)_A \Delta U_A \, \Delta x_A^m - \left(\frac{\partial F}{\partial U}\right)_A \Delta U_A \, \Delta y_A^m\right\}.$$

These are higher order terms compared to ΔU's, and they will be neglected in the evaluation of the leading term of the nonconservation error. The change with time of the state-vector in cell A is given by the expression

$$\Delta U_A = \frac{\Delta t_A}{S_A}\left\{\frac{F_7+F_9}{2}(Y_7-Y_9) - \frac{G_7+G_9}{2}(X_7-X_9) + bt\right\},$$

where in the following, bt will denote a contribution from boundary nodes 1, 4, 7, 10 and 3, 6, 9, 11. The expressions of ΔU's for the rest of the cells are similar, and after substituting to (18), the leading order term for the nonconservation error is

$$E = +\frac{\Delta Y}{4}\{(2F_8 - F_7 - F_9) - (2F_5 + F_4 + F_6)\} - \frac{\Delta X}{4}\{(2G_8 - G_7 - G_9) - (2G_5 + G_4 + G_6)\} + bt, \tag{19}$$

where

$$\Delta X = X_9 - X_8 = X_8 - X_7 = X_6 - X_5 = X_5 - X_4$$
$$\Delta Y = Y_9 - Y_8 = Y_8 - Y_7 = Y_6 - Y_5 = Y_5 - Y_4. \tag{20}$$

Observe that the nonconservation error due to inviscid terms is first order. It is local error involving flux-vector values at interface nodes 7, 8, 9 and at neighbouring nodes 4, 5, 6.

Similarly for the viscous terms, the sum of contributions ΔU_{vis} from nodes 1 through 11 (formulas (8)) is considered:

$$E = \Sigma_{\text{nodes}} \frac{S}{\Delta t}\Delta U_{\text{vis}} = -R_B^E \cdot \Delta y_B^m - R_C^W \cdot \Delta y_C^m + S_B^E \cdot \Delta x_B^m - S_C^W \cdot \Delta x_C^m + bt. \tag{21}$$

The nonconservation error is first order and depends upon the solution at interface nodes 7, 8, 9 and at neighbouring nodes 4, 5, 6.

The definition of conservation for the smoothing operator is given by Eq. (17) ($\Sigma_{\text{nodes}} \delta U_{\text{node}} = bt.$) Consider the second-order smoothing distribution formulas (9) and define the operators $\delta_x^{(\cdot)}, \delta_y^{(\cdot)}$ to denote differences of state-vector variables along the x, y-directions of cell $(\cdot)$. Summing up the shock smoothing distributions to nodes 1 through 11 of interface formulation of Fig. 4, one obtains

$$E \equiv \Sigma_{\text{nodes}} 4\,\delta U_{\text{node}} = 2\delta_y^B P\,\delta_y^B U - \delta_y^{A'} P\,\delta_y^{A'} U + 2\delta_y^C P\,\delta_y^C U - \delta_y^{A'} P\,\delta_y^{A'} U + \delta_y^A P\,\delta_y^A U - \delta_y^{A'} P\,\delta_y^{A'} U. \tag{22}$$

The nonconservation error is maximum in the case of a shock which is located at and parallel to an interface. In this case, the largest term in (22) is

$$\delta_y^A P\,\delta_y^A U,$$

which is $O(1)$. However, if the shock is located just one cell away from the interface, the $O(1)$ term vanishes. In practice, many schemes capture a shock within approximately four cells, which implies that interfaces should be at least four cells away from shocks which are parallel to them.

Finally, the fourth-order smoothing nonconservation error is evaluated on defining $S^{(\cdot)}$ to be the pressure switch at cell $(\cdot)$ that is used to turn fourth-order smoothing off near shocks, and $D^{(\cdot)}$ is the sum of the second-order differences at the four cell corners (e.g., $D^E = D_1 + D_4 + D_5 + D_2$).

$$E \equiv \Sigma_{\text{nodes}} 4\,\delta U_{\text{node}} = 2S^B(D^B - 2D_4 - 2D_5) + 2S^C(D^C - 2D_5 - 2D_6) + 3S^{A'}(D^{A'} - 2D_7 - 2D_9) + S^A(D^A - 2D_7 - 2D_9).$$

Away from a shock, the switches S^B, S^C, $S^{A'}$, S are the same and equal to a smoothing coefficient σ_4, while they are zero near shocks. Thus, the above sum resumes the form

$$\Sigma_{\text{nodes}} \frac{4}{\sigma_4}\delta U_{\text{node}} = 3D_1 + 3D_3 + D_{10} + D_{11} - 2D_7 - 2D_9 + 4D_8. \tag{23}$$

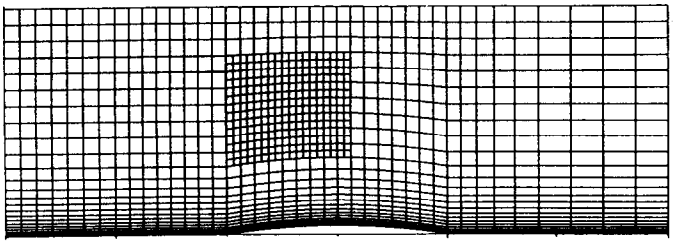
However, $D_8 = \frac{1}{2}(D_7 + D_9)$, which leads to

$$\Sigma_{\text{nodes}} \frac{4}{\sigma_4}\delta U_{\text{node}} = 2D_1 + 2D_3 + bt. \tag{24}$$

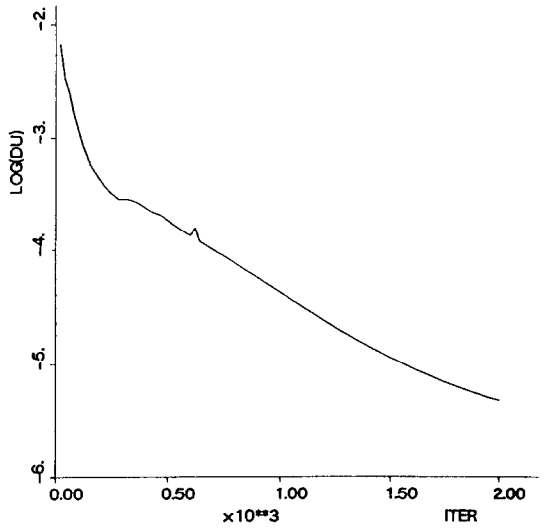
The above D_1, D_3 terms of the nonconservation error represent second-order differences of the solution, which implies that the error is of second order.

3.2. *Example Case*

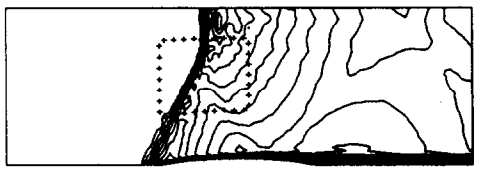
An example serves to illustrate the performance of the above interface treatment. A 4% bump in a channel in supersonic flow ($M_\infty = 1.35$) was computed employing an initial mesh of 25×25 and with embedding placed close to the incoming and reflected shocks at the upper wall of the



(a) embedded mesh



(b) convergence history



(c) Mach contours (+ denotes interface)

FIG. 5. Shock through interfaces for a 4% bump in a channel ($M_\infty = 1.35$, $\mathrm{Re} = 23 \times 10^3$): (a) embedded mesh; (b) convergence history; (c) Mach number contours (+ denotes interface).

channel (Fig. 5a) in order to test robustness of the interface treatment. The solid boundary extends from the leading edge of the bump up to the exit. The case converged despite the nonconservation error, as shown in Fig. 5b and the resulting flow field is depicted in Fig. 5c in terms of Mach number contours. Both shock location and strength prove to be the same as those observed using a globally fine mesh. In Section 5.1 the above treatment will be applied to a transonic flow airfoil case.

4. CONSERVATIVE INTERFACE TREATMENT

In the nonconservative treatment, the equations are not integrated at the middle interface node with solution variables being interpolated to that node. Any conservative treatment should include this face node in the integration procedure. Both the line integration around the cell and the distribution of changes in time to cell nodes should consider the face node as well. A way of constructing conservative treatments will now be described for each one of the contributing terms, i.e., the inviscid, the viscous, and the smoothing terms. The same interface configuration of Fig. 4 is used for the study.

4.1. *Inviscid Terms*

The fluxes across the interface from the surrounding cells A, B, and C (Fig. 6) should cancel in order for the treatment to be conservative at steady state. This is accomplished by performing a special integration around cell A that takes into account face node 8. For a trapezoidal integration around interface cells A, B, and C, the interface fluxes H_A, H_B, H_C are

$$\begin{aligned} H_A = &+\tfrac{1}{2}(F_7+F_8)\cdot(Y_8-Y_7) \\ &-\tfrac{1}{2}(G_7+G_8)\cdot(X_8-X_7) \\ &+\tfrac{1}{2}(F_8+F_9)\cdot(Y_9-Y_8) \\ &-\tfrac{1}{2}(G_8+G_9)\cdot(X_9-X_8) \end{aligned} \tag{25a}$$

$$\begin{aligned} H_B = &+\tfrac{1}{2}(F_8+F_7)\cdot(Y_7-Y_8) \\ &-\tfrac{1}{2}(G_8+G_7)\cdot(X_7-X_8) \end{aligned} \tag{25b}$$

$$\begin{aligned} H_C = &+\tfrac{1}{2}(F_9+F_8)\cdot(Y_8-Y_9) \\ &-\tfrac{1}{2}(G_9+G_8)\cdot(X_8-X_9). \end{aligned} \tag{25c}$$

The above cancellation of fluxes is sufficient for conservation for schemes which solve the steady Navier–Stokes equations. For schemes, which include the unsteady term in the integration procedure, the sufficient condition for conservation is Eq. (15) and the above relation is only a necessary condition.

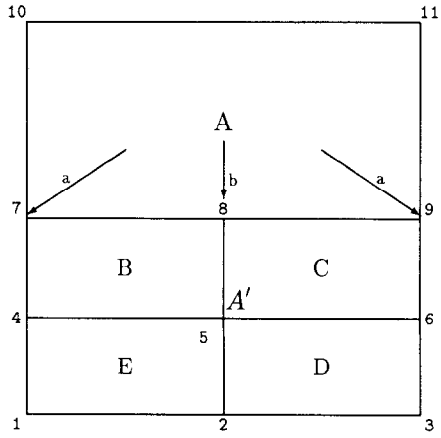


FIG. 6. Conservative interface treatment.

In the normal case of a cell with four nodes, the cell change in time of the state-variables is allocated to the corners using the same distribution coefficient, which is equal to $\frac{1}{4}$ (Eq. (5)). However, in the case in which a fifth face node exists, the distribution coefficients at nodes 7, 8, and 9 should be modified. By symmetry, the distribution coefficients to nodes 7 and 9 must be the same and, say, equal to a, while the distribution coefficient to face node 8 is b. These are unknowns, which are to be evaluated so that the inviscid distributions satisfy the definition of conservation

$$\Sigma_{\text{nodes}} \frac{S}{\Delta t} \delta U_{\text{node}} = bt. \tag{26}$$

In order to be able to use the regular scheme distribution formulas (5) for face node 8 as well, consider cell A to consist of two subcells A_1, A_2 (Fig. 7). Again by symmetry, the distribution coefficients to node 8 from each of subcells A_1, A_2 is $b/2$. The forms of ΔU_{A_1}, ΔU_{A_2}, Δf_{A_1}, Δf_{A_2}, Δg_{A_1}, Δg_{A_2} will be such as to make the treatment conservative, and they remain to be found.

Consider now the sum in the definition of conservation for the interface configuration of Fig. 6,

$$\begin{aligned}\Sigma_{\text{nodes}} \frac{S}{\Delta t} \delta U_{\text{node}} = {} & \frac{S_E}{\Delta t_E} \Delta U_E + \frac{S_D}{\Delta t_D} \Delta U_D \\ & + \frac{S_B}{\Delta t_B} \Delta U_B + \frac{S_C}{\Delta t_C} \Delta U_C \\ & + \frac{S_A}{\Delta t_A} \left(\frac{1}{2} + 2a\right) \Delta U_A \\ & + \frac{S_{A_1}}{\Delta t_{A_1}} \frac{b}{2} \Delta U_{A_1} + \frac{S_{A_2}}{\Delta t_{A_2}} \frac{b}{2} \Delta U_{A_2} \\ & + \frac{S_{A_1}}{\Delta t_{A_1}} \frac{b}{2} (\Delta f_{A_1} - \Delta g_{A_1}) \\ & + \frac{S_{A_2}}{\Delta t_{A_2}} \frac{b}{2} (-\Delta f_{A_2} - \Delta g_{A_2}) \\ & + \frac{S_A}{\Delta t_A} \left(\frac{1}{2} - 2a\right) \Delta g_A. \end{aligned} \tag{27}$$

The above expression should consist of boundary node terms only, in order for the treatment to be conservative for both steady-state and unsteady computations. The terms ΔU, Δf, Δg at subcells A_1, A_2 express the change in time of state-variables within cell A, which is the cell which is actually used in the integration process and, thus,

$$\begin{aligned} \Delta U_{A_1} &= \Delta U_{A_2} = \Delta U_A \\ \Delta f_{A_1} &= \Delta f_{A_2} = \Delta f_A \\ \Delta g_{A_1} &= \Delta g_{A_2} = \Delta g_A. \end{aligned} \tag{28}$$

The sum of (27) therefore becomes

$$\begin{aligned}\Sigma_{\text{nodes}} \frac{S}{\Delta t} \delta U_{\text{node}} = {} & + \frac{S_E}{\Delta t_E} \Delta U_E + \frac{S_D}{\Delta t_D} \Delta U_D \\ & + \frac{S_B}{\Delta t_B} \Delta U_B + \frac{S_C}{\Delta t_C} \Delta U_C \\ & + \frac{S_A}{\Delta t_A} \left(\frac{1}{2} + 2a + b\right) \Delta U_A \\ & + \frac{S_A}{\Delta t_A} \left(\frac{1}{2} - 2a - b\right) \Delta g_A. \end{aligned} \tag{29}$$

The sum involving the ΔU terms yields boundary node contributions only, if

$$\tfrac{1}{2} + 2a + b = 1 \Rightarrow 2a + b = \tfrac{1}{2}. \tag{30}$$

If (30) is satisfied, the last term in (29), which involves Δg vanishes. Clearly, Eq. (30) is the condition that any interface distribution coefficients scheme has to satisfy. There are finite combinations of a and b that satisfy (30). Choosing $b = 2a$, it is obtained:

$$a = \tfrac{1}{8}, \qquad b = \tfrac{1}{4}. \tag{31}$$

The distribution coefficients express the part of the cell area which is allocated to each one of the cell nodes. In the normal case of a cell with four nodes, the cell area is divided into four equal areas, one for each node, and thus a fourth of the total change ΔU is allocated to each node. However, if a fifth face node exists, the cell area is allocated to the five nodes as illustrated in Fig. 8. The ratios of each of the areas to the cell area are the distribution coefficients. Another

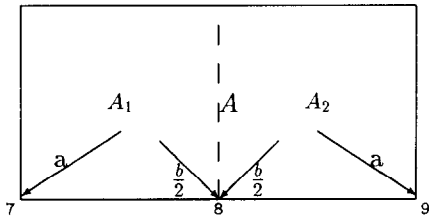


FIG. 7. Subcell distributions to interface nodes.

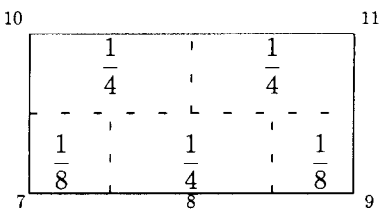


FIG. 8. Cell area allocation to nodes.

derivation of the above distribution coefficients consider time-accuracy of the interface treatment. Shape functions and finite-element theory can be employed to verify the above values for the distribution coefficients to the interface nodes.

4.2. *Viscous Terms*

The subcells into which the coarse interface cell A is divided (Fig. 8), will now be employed in order to include face node 8 into the integration process for the viscous terms. The viscous fluxes, which consist of stress terms should cancel each other in the interior of the domain of Fig. 9. Interface cell A is divided into five cells, and the notation to be used is shown in Fig. 9. The stress flux evaluation in the interior of cell A consists of line integrations along the broken lines. For this purpose, points w, e, n, s_1, s_2, c_1, c_2, c, which lie at the faces and in the interior of cell A, are employed. The values of the stresses at these points are obtained by using interpolations, in such a way that fluxes in the interior of cell A cancel.

More specifically, the viscous contributions from cell A to its five nodes are

$$\Delta U_7 = \left(\frac{\Delta t}{S}\right)_7 \{[R_{s1}(Y_{c1} - Y_{s1}) + R_w(Y_w - Y_{c1})] - [S_{s1}(X_{c1} - X_{s1}) + S_w(X_w - X_{c1})]\} \quad (32a)$$

$$\Delta U_8 = \left(\frac{\Delta t}{S}\right)_8 2\{[R_{s2}(Y_{c2} - Y_{s2}) + R_c(Y_{c1} - Y_{c2}) + R_{s1}(Y_{s1} - Y_{c1})] - [S_{s2}(X_{c2} - X_{s2}) + S_c(X_{c1} - X_{c2}) + S_{s1}(X_{s1} - X_{c1})]\} \quad (32b)$$

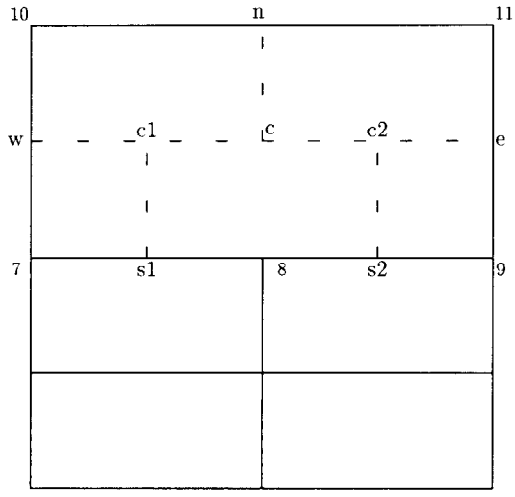


FIG. 9. Conservative treatment of viscous terms.

$$\Delta U_9 = \left(\frac{\Delta t}{S}\right)_9 2\{[R_e(Y_{c2} - Y_e) + R_{s2}(Y_{s2} - Y_{c2})] - [S_e(X_{c2} - X_e) + S_{s2}(X_{s2} - X_{c2})]\} \quad (32c)$$

$$\Delta U_{10} = \left(\frac{\Delta t}{S}\right)_{10} 2\{[R_w(Y_c - Y_w) + R_n(Y_n - Y_c)] - [S_w(X_c - X_w) + S_n(X_n - X_c)]\} \quad (32d)$$

$$\Delta U_{11} = \left(\frac{\Delta t}{S}\right)_{11} 2\{[R_e(Y_e - Y_c) + R_n(Y_c - Y_n)] - [S_e(X_e - X_c) + S_n(X_c - X_n)]\}. \quad (32e)$$

The subscripts of the R and S terms represent the points at which these terms are evaluated. The R, S-values in the interior points c_1, c_2, c are evaluated via the following interpolations:

$$R_c = \frac{R_w + R_e}{2}, \qquad S_c = \frac{S_w + S_e}{2} \quad (33a)$$

$$R_{c1} = \frac{R_w + R_c}{2}, \qquad S_{c1} = \frac{S_w + S_c}{2} \quad (33b)$$

$$R_{c2} = \frac{R_c + R_e}{2}, \qquad S_{c2} = \frac{S_c + S_e}{2}. \quad (33c)$$

Different interpolations to obtain R_c, S_c (e.g., $R_c = \frac{1}{2}(R_n + (R_{s1} + R_{s2})/2)$) result in a nonconservative treatment.

For steady-state evaluations, the unsteady term $\Delta t/S$ for each node can be the same and equal to $(\Delta t/S)_A$. This simplifies coding of the interface algorithm, and required information is strictly within each cell, which is important for unstructured grid algorithms. However, if time accuracy is required, then different ratios $\Delta t/S$ should be used for each node.

Stretching error. The above treatment suffers from stretching error. Consider face node 8 and the *secondary* cell, which is employed in order to evaluate the viscous derivatives such as u_{yy} (Fig. 10). Distance $c8$ is about twice

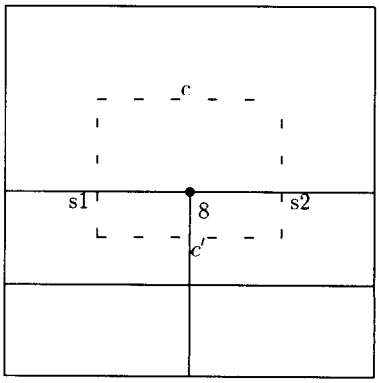


FIG. 10. Abrupt change in grid size at an interface introduces accuracy error to the viscous terms evaluation.

distance $8c'$, which introduces a serious error. The evaluation could be made accurate if special area weighting factors are used when performing the line integration around cell $s1cs2c'$. This will result in noncancellation of the fluxes through the faces and, therefore, in nonconservation.

4.3. *Shock Smoothing*

The distributions to cell nodes due to second-order smoothing (10), (11) yield a conservative operator for a normal cell with four nodes. These distributions should be modified in order to maintain conservation when face nodes exist. The same approach that was employed in order to derive conservative interface distributions for the inviscid terms is followed. The two subcells A_1, A_2 of cell A are considered and the unknown distribution coefficients to the interface nodes 7, 8, 9 are denoted with a and b (Fig. 7). Consider the sum of δU's and use the same notation that was used for the evaluation of the nonconservation error earlier; it is obtained:

$$\begin{aligned}\Sigma_{\text{nodes}} 4\,\delta U_{\text{node}} = {} & (4a-2b)\,\delta_x^{A_1} P\,\delta_x^{A_1} U \\ & + (4a+2b)\,\delta_y^{A_1} P\,\delta_y^{A_1} U \\ & - (4a-2b)\,\delta_x^{A_2} P\,\delta_x^{A_2} U \\ & + (4a+2b)\,\delta_y^{A_2} P\,\delta_y^{A_2} U - 2\delta_y^{A} P\,\delta_y^{A} U. \quad (34)\end{aligned}$$

The terms $\delta_x^{A_1} P\,\delta_x^{A_1} U$, $\delta_x^{A_2} P\,\delta_x^{A_2} U$, $\delta_y^{A_1} P\,\delta_y^{A_1} U$, $\delta_y^{A_2} P\,\delta_y^{A_2} U$ should be defined so that the above sum reduces to zero. This is achieved by defining the subcell terms in a suitable way as follows:

$$\delta_x^{A_1} P\,\delta_x^{A_1} U = \delta_x^{A_2} P\,\delta_x^{A_2} U = \delta_x^{A} P\,\delta_x^{A} U \quad (35a)$$

$$\delta_y^{A_1} P\,\delta_y^{A_1} U = \delta_y^{A_2} P\,\delta_y^{A_2} U = \delta_y^{A} P\,\delta_y^{A} U. \quad (35b)$$

Substituting (35) into (34),

$$\Sigma_{\text{nodes}} 4\,\delta U_{\text{node}} = (8a+4b-2)\,\delta_y^{A} P\,\delta_y^{A} U,$$

is obtained which implies that

$$2a+b=\tfrac{1}{2}. \quad (36)$$

Again there are infinite combinations of a and b that satisfy (36). The choice of a and b is not as clear as it was for the inviscid terms, in which time accuracy dictated the choice of coefficients. Again making the same consideration that $b=2a$,

$$a=\tfrac{1}{8}, \qquad b=\tfrac{1}{4} \quad (37)$$

as in (31).

Conservation versus dissipation. Another choice which has to be made, is the form of $\delta_x^A P\,\delta_x^A U$, $\delta_y^A P\,\delta_y^A U$. This choice will not affect conservation, but it will yield different operators. Face node 8 can be ignored and the above terms can be defined in the same way they are defined for a normal four-node cell:

$$\delta_x^A U = (U_{11}+U_9)-(U_{10}+U_7) \quad (38a)$$

$$\delta_y^A U = (U_{10}+U_{11})-(U_7+U_9). \quad (38b)$$

Assuming that the pressure switch is unity, without loss of generality the operator at node 8 is

$$(U_{10}+U_{11})-2(U_7+U_8+U_9)+(U_4+2U_5+U_6),$$

which is *convective*. A possible "cure" may be the use of a form of $\delta_y^A U$, which takes into account face node 8 as well, such as

$$\delta_y^A U = \left(U_{10}+2\,\frac{U_{10}+U_{11}}{2}+U_{11}\right)-(U_7+2U_8+U_9).$$

Now the operator becomes

$$\begin{aligned}&(U_{10}+U_4-2U_7)+(U_{11}+U_6-2U_9) \\ &\quad +2\left(\frac{U_{10}+U_{11}}{2}+U_5-2U_8\right),\end{aligned}$$

which is dissipative. However, the operator becomes convective at nodes 10 and 11. For example, the operator at node 10 is

$$\begin{aligned}&(U_{12}+U_7-2U_{10})+(U_{13}+U_9-2U_{11}) \\ &\quad -2\left(\frac{U_{10}+U_{11}}{2}-U_8\right),\end{aligned}$$

where 12, 13 are the nodes just above nodes 10, 11 and are not included in the figure. Clearly, changing the form of $\delta_x^A U$, $\delta_y^A U$ does not result in a globally dissipative operator. The distribution to the nodes has to be modified in order to make smoothing dissipative everywhere, which sacrifices conservation.

4.4. *Fourth-Order Smoothing*

Similarly the above approach is followed for the fourth-order smoothing operator on the same interface configuration of Figs. (6), (7). In the following, S denotes the switch

which turns the operator off near shocks, and $D^{(\cdot)}$ is the sum of the second-order difference values $D_{(\cdot)}$ at cell nodes:

$$\begin{aligned}\Sigma_{\text{nodes}} 4\,\delta U_{\text{node}} = &+4aS^A(2D^A - 4D_7 - 4D_9)\\ &+ S^A(2D^A - 4D_{10} - 4D_{11})\\ &+ 2bS^{A_1}(D^{A_1} - 4D_8)\\ &+ 2bS^{A_2}(D^{A_2} - 4D_8).\end{aligned} \tag{39}$$

The subcell terms S^{A_1}, S^{A_2}, D^{A_1}, D^{A_2} are defined as

$$S^{A_1} = S^{A_2} = S^A, \qquad D^{A_1} = D^{A_2} = D^A,$$

which leads to

$$\begin{aligned}\Sigma_{\text{nodes}} \frac{\delta U_{\text{node}}}{S^A} = &\left(2a + b + \frac{1}{2}\right) D^A - 4aD_7 - 4aD_9\\ &- 4bD_8 - D_{10} - D_{11}.\end{aligned} \tag{40}$$

D^A is defined as

$$D^A = 4aD_7 + 4aD_9 + 4bD_8 + D_{10} + D_{11}, \tag{41}$$

which yields

$$\Sigma_{\text{nodes}} \frac{\delta U_{\text{node}}}{S^A} = \left(2a + b - \frac{1}{2}\right) D^A. \tag{42}$$

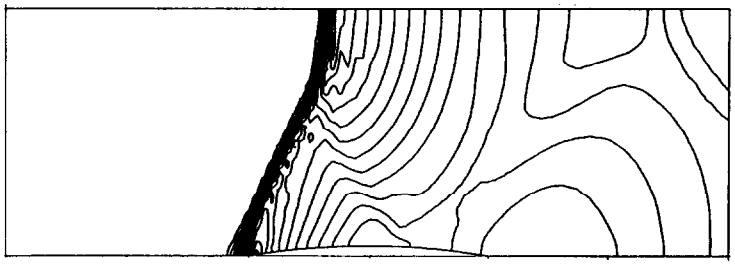
The treatment is conservative if

$$2a + b - \tfrac{1}{2} = 0,$$

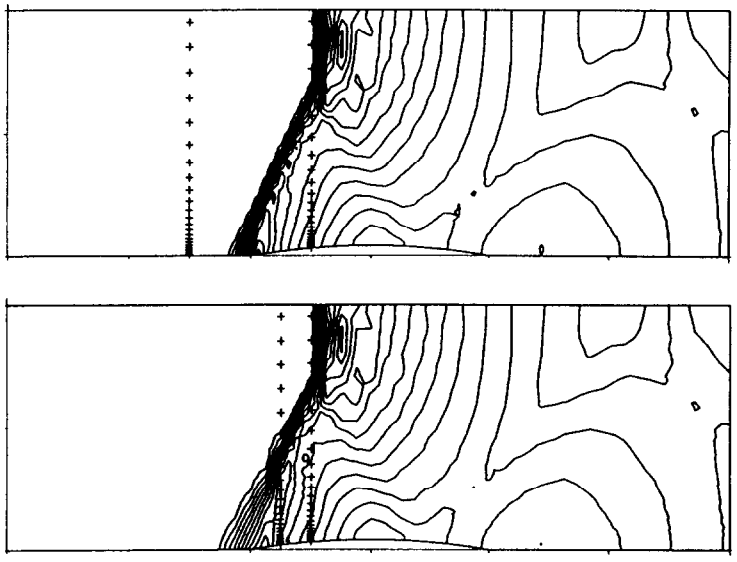
and by choosing $b = 2a$, the distribution coefficients are once more

$$a = \tfrac{1}{8}, \qquad b = \tfrac{1}{4}, \tag{43}$$

as in (31) and (37). The second-order difference operator



Globally fine mesh solution (no interfaces)



+ denotes interface

FIG. 11. Interfaces near a shock—conservative interface treatment for a 4% bump in a channel ($M_\infty = 1.35$, $\text{Re} = 10^4$).

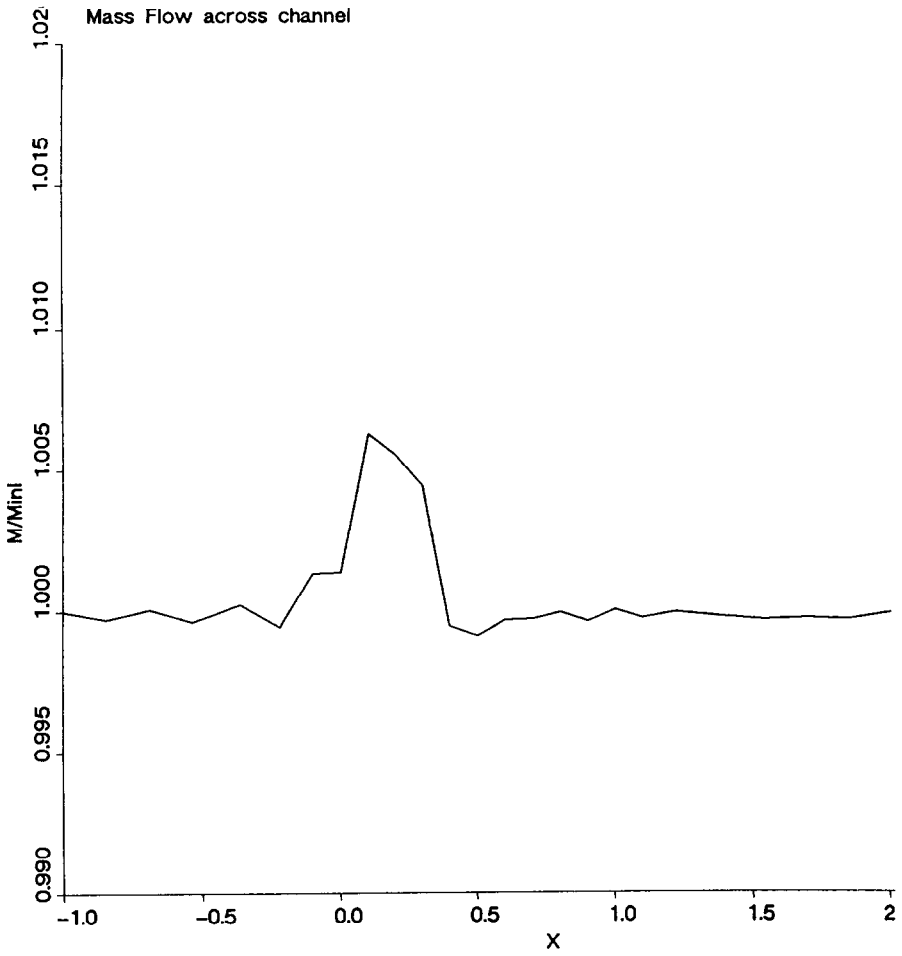


FIG. 12. No mass error with conservative interface treatment.

at face node 8, D_8, can be arbitrary without violating conservation. A choice that has been made is to consider

$$D_8 = \frac{D_7 + D_9}{2}. \tag{44}$$

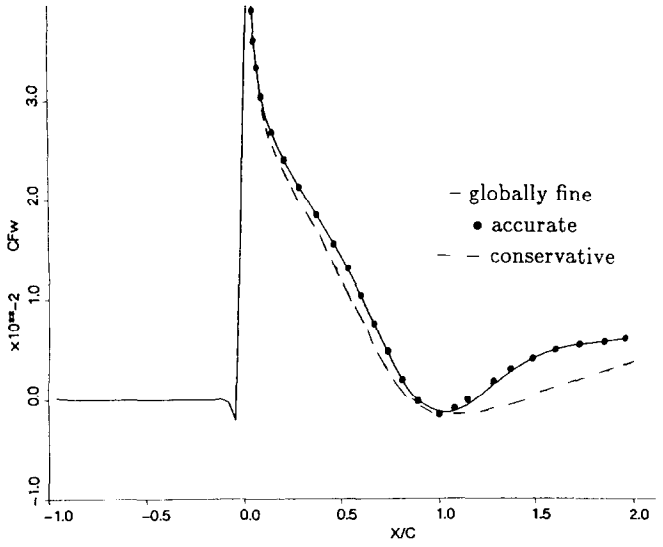
Again, the above conservative operator is *convective* and conservation has to be sacrificed in order to make it dissipative.

4.5. *Test Case*

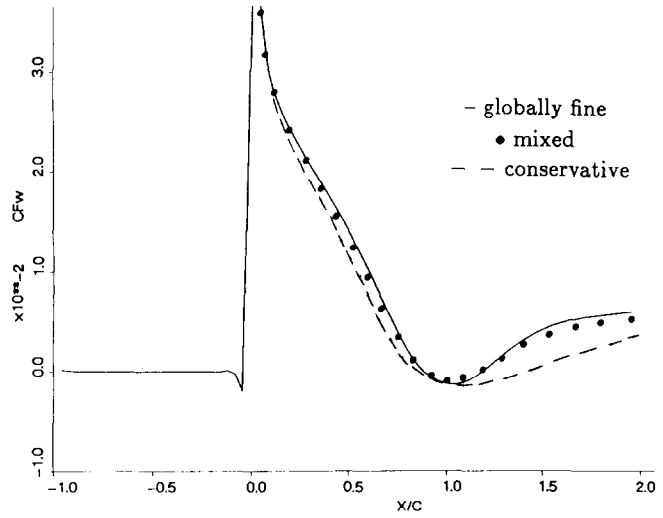
The flow through the 4% bump in a channel ($M_\infty = 1.35$, $\mathrm{Re} = 10^4$) was computed using an initial mesh of 25×25 and one level of embedding, which was placed at various locations within the domain as shown in Fig. 11, which depicts the solution via pressure coefficient contours. The embedded mesh solutions may be compared with the globally fine mesh solution that is shown in the same figure. The extent of the coarse mesh regions is different in Figs. 11b, c, which introduces differences to the solutions. Finally, in Fig. 12 the mass flow across the channel is illustrated for the embedded case of Fig. 11c and shows that there is no mass error between the inlet and the outlet of the channel. The mass increases on the bump region due to addition of shock smoothing. However, there is no net addition or subtraction of mass over that area, since the smoothing operator is conservative.

5. ACCURATE VERSUS CONSERVATIVE INTERFACE TREATMENTS

There are two main advantages for an accurate interface treatment. First is the accuracy which is retained at the interface despite the abrupt change in cell-size. This has proven to be a very important property for cases in which the interface lies within a viscous region. On the other hand, the conservative treatment suffers from a significant stretching error. This is clearly observed in Fig. 13a which shows skin friction distribution for the subsonic ($M_\infty = 0.5$) 5% bump with interfaces within the boundary layer. It is



(a) accurate , conservative treatments



(b) 'mixed', conservative treatments

FIG. 13. Accurate, conservative, and mixed interface treatments for interfaces inside a boundary layer.

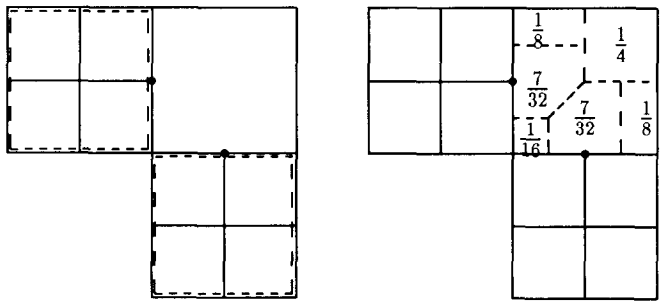
observed that upstream of separation the two treatments give approximately the same result. However, as separation is approached and the boundary layer profile becomes curved at the region where the interfaces are placed, the stretching error in the conservative treatment shows up.

In order to investigate this point further, a mixed interface treatment has been employed. According to this treatment, the viscous terms are treated with the accurate interface treatment, while the inviscid and smoothing parts are treated conservatively. Figure 13b clearly demonstrates that the accuracy problem disappears. Therefore, the stretching error in the evaluation of the viscous interface terms is responsible for the observed inaccuracy of the fully conservative treatment in Fig. 13a.

The second advantage if the accurate but nonconservative treatment is its fairly simple implementation. All kinds of interfaces are treated in the same way by employing the parent cell of the fine interface cells (cell A' in Fig. 4) in order to integrate the interface nodes. The algorithm does not "care" whether there are one or more interface nodes at an interface cell, the treatment being the same as illustrated in Fig. 14a. In the case of the conservative treatment, however, the distribution coefficients are different depending upon the number of face nodes of an interface cell. For example, the distribution coefficients and the splitting of the coarse interface cell for the very common case of a cell with two face nodes are illustrated in Fig. 14b. The numbers denote the distribution coefficients used for the inviscid terms, and the six areas are the ones that are employed for integration of the viscous terms. There are four different such configurations of cells with two face nodes depending upon location of the face nodes.

Another problem associated with the conservative treatment is the locally dispersive smoothing operators and the ambiguity in formulation of that part of the smoothing operator which corresponds to the coarse interface cell. As it was seen, there are numerous forms that the operator can assume depending upon the way in which the face node is taken into account. The determination of distribution coefficients a and b is not unique. A conservation requirement is not sufficient to determine both coefficients, and other considerations such as time accuracy and the form of the smoothing operator are taken into account in order to determine these values. Finally, the same approach, with the "parent" cell now being a cube, can be carried out for 3D interfaces. The coding of the conservative treatment, on the other hand, is fairly complicated.

The main weakness of an accurate treatment is the nonconservation error induced by the shock smoothing operator. In the case of a shock being located at the interface and parallel to it, the nonconservation error is $O(1)$. Figure 15 compares mass flow across the 4% bump for the case of supersonic flow with $M_\infty = 1.35$ with the nonconser-



(a) accurate (b)conservative

FIG. 14. Accurate versus conservative interface treatment for cells with two face nodes.

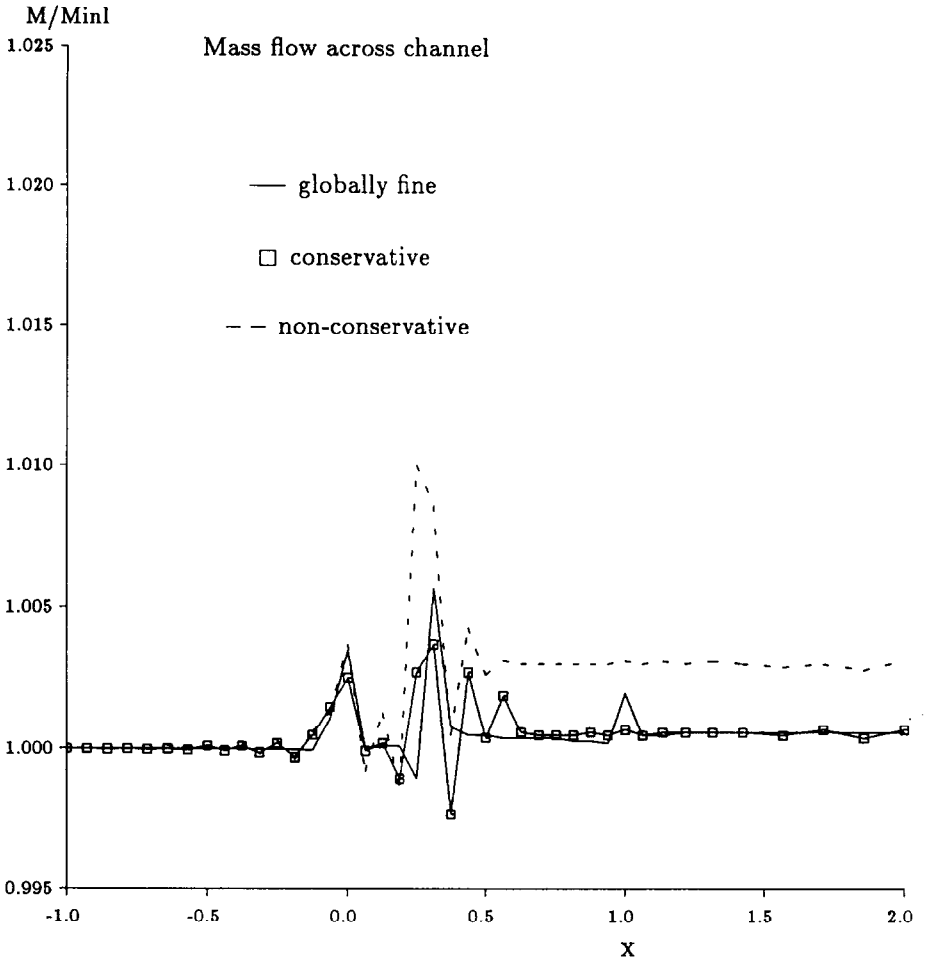


FIG. 15. Comparison of interface treatments for interfaces near shock.

vative and conservative treatments, as well as a globally fine mesh. The nonconservative treatment induces a mass flow error that is approximately five times larger than the error introduced by the conservative treatment. The nonconservation error for both the globally fine (no interfaces) and the conservative case is very small (0.05%).

As far as computing time is concerned, both interface treatments are relatively inexpensive. The number of interfaces for a typical computation within a domain of $N \times N$ points is usually of $O(N)$ and therefore, the CPU time which is consumed is not appreciable. The accurate treatment integrates two cells for each interface (e.g., A and A' in Fig. 4), while the conservative treatment integrates cell A only. However, the special trapezoidal integration, which takes into account possible face nodes and the special distributions to the interface nodes, eliminates this advantage.

5.1. *Airfoil in Transonic Flow*

The flow around a NACA 0012 airfoil under transonic conditions was among the cases considered in [1] in order to evaluate a Navier–Stokes adaptive algorithm. The flow conditions were: $M_\infty = 0.754$, $\mathrm{Re} = 3.76 \times 10^6$, $\alpha = 3.02°$. The interface treatment of Section 3 was employed. A shock forms on the upper side of the airfoil, which interacts with the boundary layer. An initial C-mesh of 65×41 points was applied and three levels of embedded meshes were introduced by the algorithm as illustrated in Fig. 16. Figure 17a shows a view of the shock–boundary layer interaction region. The severe adverse pressure gradient that is induced by the normal shock causes the boundary layer to thicken considerably and eventually to separate at the foot of the normal shock. A separation bubble is formed and it is captured in detail by the adaptive algorithm. The numerical results have been compared with experimental data and the agreement between the two was very good [1].

Performance of the interface treatment of Section 3 is examined by showing the grid interface locations within the flow domain. Figure 17a illustrates interface positions at the shock–boundary layer interaction region, while Fig. 17b concentrates on the trailing edge region. A large portion of the interfaces lie parallel to the shear layer with significant

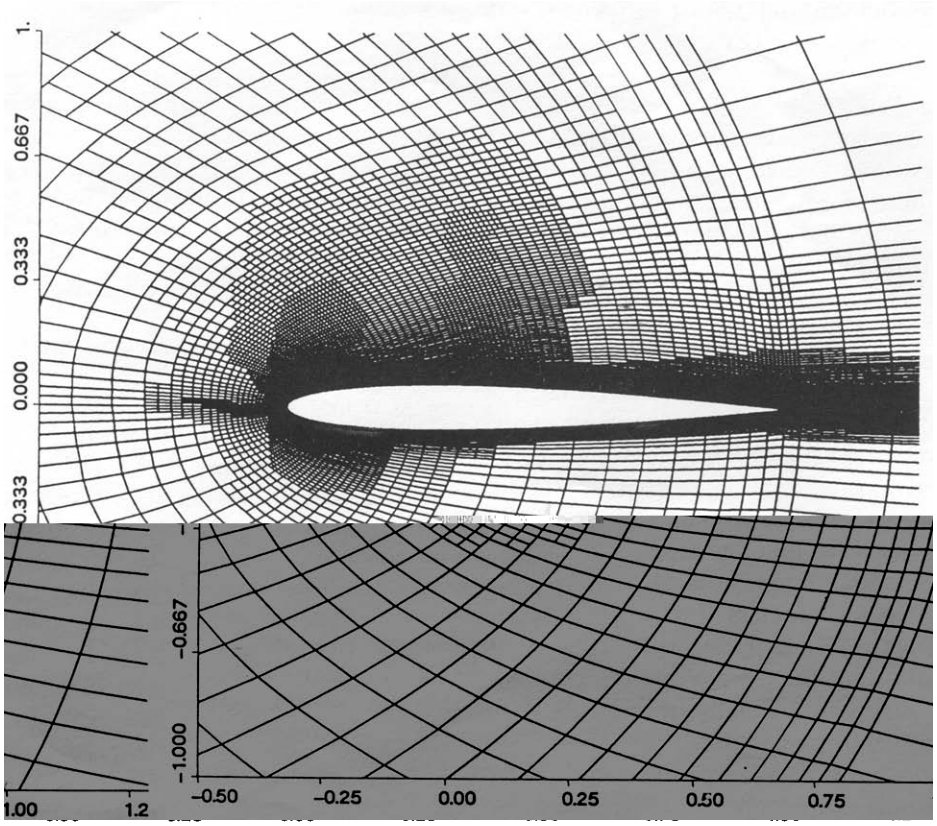


FIG. 16. Three-level embedded grid—transonic NACA 0012 ($M_\infty = 0.754$, $\mathrm{Re} = 3.76 \times 10^6$, $\alpha = 3.02°$).

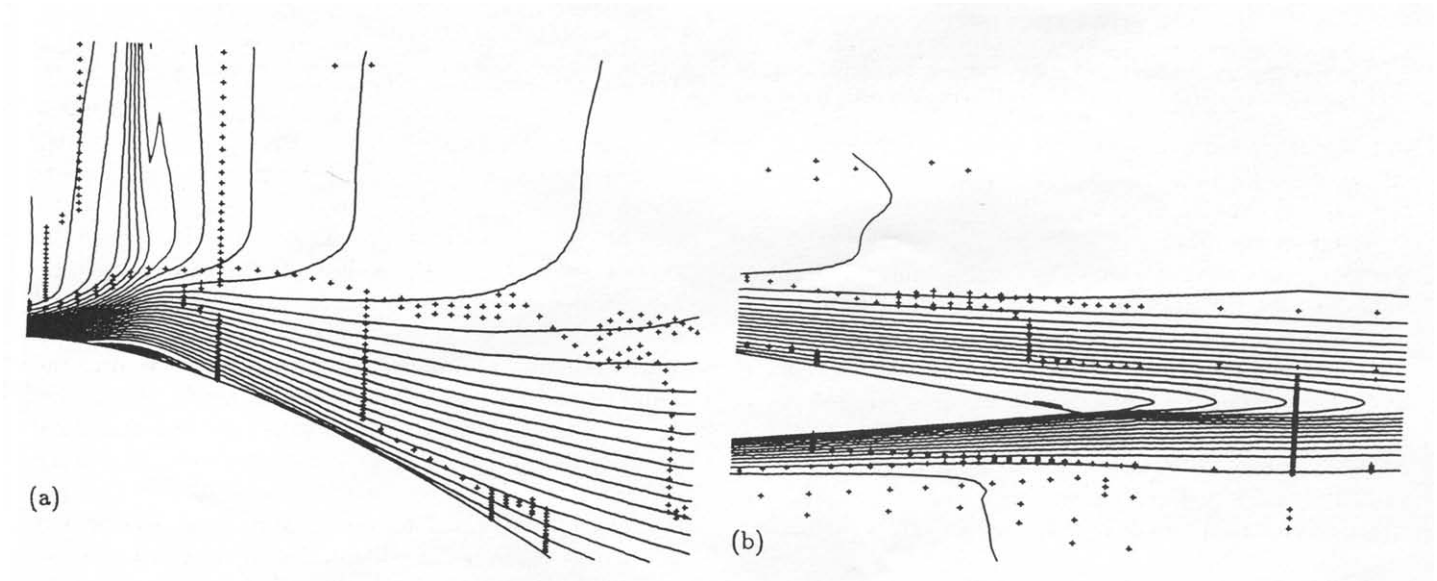


FIG. 17. Mach number contours field with interface locations—transonic NACA 0012 ($M_\infty = 0.754$, $\mathrm{Re} = 3.76 \times 10^6$, $\alpha = 3.02°$): (a) shock/boundary layer interaction region (vertical scale enlarged); (b) trailing edge region (vertical scale enlarged).

flow gradients across them. The presence of interfaces does not disturb the flow solution and almost no "kinks" are observed.

6. SUMMARY

Employment of adaptive quadrilateral grid embedding introduces interfaces into the computation domain. A special scheme is required for numerical treatment of those special areas. Development and evaluation of such a scheme is a main issue for an adaptive solver.

A Navier–Stokes scheme, which is typical of a class of central-space differencing schemes, was described. The viscous terms of the Navier–Stokes equations pose significant accuracy problems, especially if the grid is stretched. A treatment that avoids grid stretching error is presented and evaluated in terms of accuracy and nonconservation error. The most significant nonconservation error is due to the shock-smoothing operator for the case of a shock that is located at an interface and that is parallel to that interface.

A method of constructing conservative interface treatments was presented. It was found that a whole class of conservative treatments can be constructed. These treatments suffer from grid stretching error, and the smoothing operator is convective instead of dissipative. Finally, the accurate and the conservative treatments are compared in terms of accuracy, nonconservation error, simplicity, and possible extension to three dimensions. An interface treatment that avoids interface grid stretching error and that is nonconservative was found to be more accurate over a conservative treatment for viscous flows that do not include moving shocks.

ACKNOWLEDGMENT

The above research was conducted at the department of Aeronautics and Astronautics of the Massachusetts Institute of Technology while the author was a candidate for the Ph.D. degree under the supervision of Professor J. R. Baron.

REFERENCES

1. Y. Kallinderis, Ph.D. thesis, MIT, Dept. of Aeronautics and Astronautics, CFDL Report No. TR-89-5, May 1989 (unpublished).
2. Y. Kallinderis and J. R. Baron, *AIAA J.* **27**, 37 (1989).
3. M. Berger and J. Oliger, *J. Comput. Phys.* **53**, 484 (1984).
4. J. Peraire, M. Vahdati, K. Morgan, and O. C. Zienkiewicz, *J. Comput. Phys.* **72**, 449 (1987).
5. J. T. Oden, T. Strouboulis, and P. Devloo, *Int. J. Numer. Methods Fluids* **7**, 1211 (1987).
6. M. O. Bristeau and J. Periaux, Lecture Series, Von Karman Institute, LS No. 1986-04, 1986 (unpublished).
7. E. H. Atta, AIAA Paper 81-0382, 1981 (unpublished).
8. E. H. Atta and J. Vadyak, AIAA Paper 82-1017, 1982 (unpublished).
9. S. R. Allmaras and J. R. Baron, AIAA Paper 86-0509, Reno, NV, January 1986 (unpublished).
10. M. Berger, NASA Langley Research Center, ICASE Report No. 84-43, 1984 (unpublished).
11. K. A. Hessenius and T. H. Pulliam, AIAA Paper 82-0969, 1982 (unpublished).
12. M. M. Rai, AIAA Paper 84-0164, 1984 (unpublished).
13. M. M. Rai, AIAA Paper 85-0295, 1985 (unpublished).
14. R. W. Beam and R. F. Warming, *J. Comput. Phys.* **23**, 87 (1976).
15. A. Jameson, W. Schmidt, and E. Turkel, AIAA Paper 81-1259, 1981 (unpublished).
16. R. W. MacCormack, AIAA Paper 69-354, 1969 (unpublished).
17. R. L. Davis, R.-H. Ni, and J. E. Carter, AIAA Paper 86-0033, Reno, NV, 1986 (unpublished).
18. R. C. Swanson and E. Turkel, AIAA Paper 85-0035, 1985 (unpublished).
19. R.-H. Ni, *AIAA J.* **20**, 1565 (1982).
20. P. D. Lax and B. Wendroff, *Commun. Pure Applied Math.* **13**, 217 (1960).