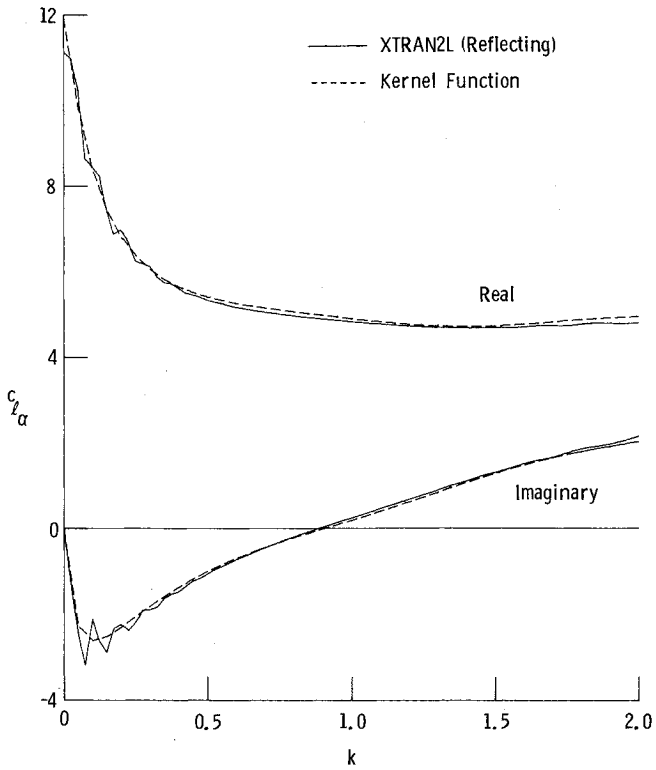

Fig. 2a Force response for flat plate ($M_\infty = 0.85$, reflecting boundary conditions used in XTRAN2L).

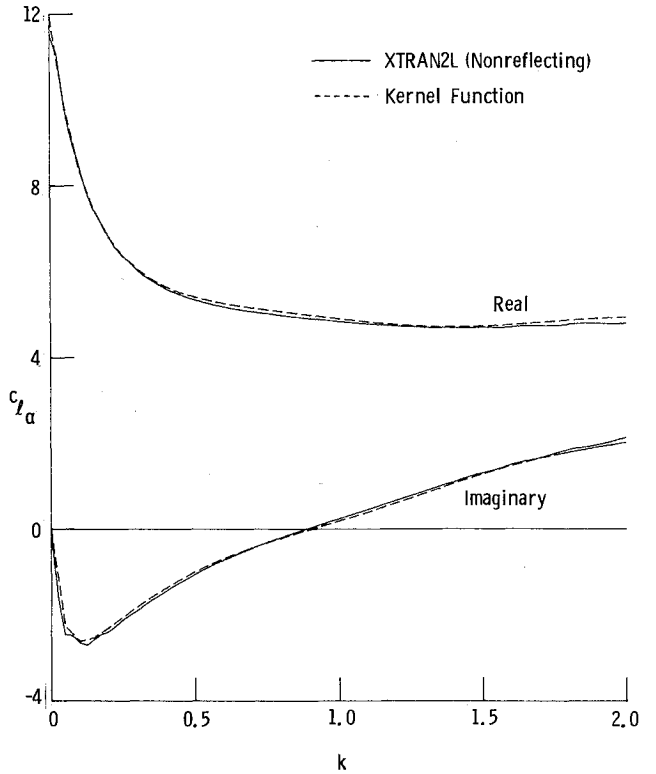

Fig. 2b Force response for flat plate ($M_\infty = 0.85$, nonreflecting boundary conditions used in XTRAN2L).

spurious oscillations due to reflected disturbances from the boundaries. When the nonreflecting boundary conditions were used (Fig. 2b), the reflected disturbances were small and good agreement with the exact solution was obtained.

Conclusions

Nonreflecting far-field boundary conditions that are consistent with the complete transonic small-disturbance (TSD) equation have been derived. They were implemented in a new code for solving the complete TSD equation and were tested for a harmonically oscillating NACA 64A010 airfoil in transonic flow and for a flat-plate airfoil with a pulse in the angle of attack. Using the new boundary conditions on a relatively small grid, solutions for an NACA 64A010 airfoil that agreed with the large-grid calculations were obtained with a 44% savings in computer time. Frequency responses for the flat plate showed that most of the disturbances incident on the computational boundaries were absorbed by the boundary conditions.

References

[1] Ballhaus, W. F. and Goorjian, P. M., "Implicit Finite-Difference Computations of Unsteady Transonic Flows about Airfoils," *AIAA Journal,* Vol. 15, Dec. 1977, pp. 1728-1735.

[2] Houwink, R. and van der Vooren, J., "Improved Version of LTRAN2 for Unsteady Transonic Flow Computations," *AIAA Journal,* Vol. 18, Aug. 1980, pp. 1008-1010.

[3] Hessenius, K. A. and Goorjian, P. M., "Validation of LTRAN2-HI by Comparison with Unsteady Transonic Experiment," *AIAA Journal,* Vol. 20, May 1982, pp. 731-732.

[4] Kwak, D., "Nonreflecting Far-Field Boundary Conditions for Unsteady Transonic Flow Computation," *AIAA Journal,* Vol. 19, Nov. 1981, pp. 1401-1407.

[5] Engquist, B. and Majda, A., "Numerical Radiation Boundary Conditions for Unsteady Transonic Flow," *Journal of Computational Physics,* Vol. 40, March 1981, pp. 91-103.

[6] Whitlow, W. Jr., "XTRAN2L: A Program for Solving the General-Frequency Unsteady Transonic Small Disturbance Equation," NASA TM 85723, Nov. 1983.

[7] Rizzetta, D. P. and Chin, W. C., "Effect of Frequency in Unsteady Transonic Flow," *AIAA Journal,* Vol. 17, July 1979, pp. 779-781.

[8] Bayliss, A. and Turkel, E., "Far Field Boundary Conditions for Compressible Flows," NASA CP-2201, 1982, pp. 1-19.

[9] Seidel, D. A., Bennett, R. M., and Whitlow, W. Jr., "An Exploratory Study of Finite Difference Grids for Transonic Unsteady Aerodynamics," AIAA Paper 83-0503, Jan. 1983.

[10] Bland, S. R., "Development of Low-Frequency Kernel-Function Aerodynamics for Comparison with Time-Dependent Finite-Difference Methods," NASA TM 83283, May 1982.

A New Self-Adaptive Grid Method

J.B. Greenberg*
Technion—Israel Institute of Technology, Haifa, Israel

Introduction

THE numerical solution of fluid flow and/or heat and mass transfer problems often entails the necessity to place a high concentration of finite difference points in regions of large gradients, in order to obtain adequate resolution of relevant parameters. In particular, there are many transient situations in which it is desirable to permit the difference grid

Presented as Paper 83-1934 at the AIAA 6th Computational Fluid Dynamics Conference, Danvers, Mass., July 13-15, 1983; received Aug. 5, 1983; revision submitted Dec. 27, 1983.

*Senior Lecturer, Department of Aeronautical Engineering, currently on sabbatical leave at Division of Applied Sciences, Harvard University, Cambridge, Mass.

to alter in accordance with the *current state* of the solution of the governing equations. To this aim, several adaptive grid methods have been suggested.[1-5] They have been demonstrated to be effective in shifting the mesh distribution dynamically so as to capture regions of large gradients and, in general, gave more accurate solutions than use of a uniform mesh would have furnished. In this Note, the basic concepts underlying a new alternative method for self-adaptive gridding is presented. The method is versatile and easy to apply, requiring that the user specify only a minimal amount of information.

Construction of the New Self-Adaptive Grid Method

For illustrative purposes, attention will be focused on the one-dimensional case. Consider a differential equation E to be solved in the interval $0 \le x \le 1$ in the physical domain. The finite difference mesh imposed on this region will be allowed to stray from uniformity in spacing according to some predefined criterion. Actually, the numerical solution of E will be carried out in a *uniform* computational domain, $0 \le \xi \le 1$, such that

$$\xi = \xi(x,t) \tag{1}$$

If both domains are subdivided into N elementary segments, then

$$\sum_{i=1}^{N} x_i - x_{i-1} = \sum_{i=1}^{N} \Delta x_i = \sum_{i=1}^{N} \Delta \xi_i = 1 \tag{2}$$

The segments Δx_i are to be permitted to change in size. Evidently, from Eq. (2), any alteration in the length Δx_i will result in alterations in the lengths of neighboring segments Δx_j, for $j \neq i$. Suppose that the *extent* to which these changes occur depends on the relative fulfillment of some condition in each segment. This description suggests a sort of interaction between elementary segments that may be formulated as

$$\Delta x_i \underset{k_{ji}}{\overset{k_{ij}}{\rightleftharpoons}} \Delta x_j, \quad i \neq j \tag{3}$$

which is analogous to a set of unimolecular reactions with non-negative rate constants k_{ij}. This sort of chemistry is described by the following ordinary differential equations:

$$\frac{d(\Delta x_i)}{dt} = -\sum_{j=1}^{N} k_{ij} \Delta x_i + \sum_{j=1}^{N} k_{ji} \Delta x_j, \quad i = 1,2,\ldots,N \tag{4}$$

The first and second terms on the right-hand side respectively represent the way in which Δx_i is reduced in size for the sake of Δx_j and the way in which each other segment Δx_j contributes to the growth of Δx_i. Note that the solution of Eq. (4) automatically insures conservation, as expressed in Eq. (2).

The key to actual changes in the segment lengths lies in the rate constants and it is via k_{ij} that complete control over the mesh point movements can be exercised. In order to demonstrate construction of the "constants," suppose a measure of "error" e_i can be associated with each segment. (For example, e_i can be taken as the local gradient of a dependent variable.) Then a typical form for the rate constant will be

$$k_{ij} = k_{ij}^{(1)} \cdot k_{ij}^{(2)} \cdot k_{ij}^{(3)} \cdot c_i^* \tag{5}$$

where

$$k_{ij}^{(1)} = \frac{1}{2}\left[\frac{|e_j - e_i| - (e_j - e_i)}{|e_j - e_i + \delta|}\right] \cdot \left(\frac{e_i + \delta}{e_j + \delta}\right) \tag{6}$$

(Note: δ is a small number introduced to avoid division by zero, e.g., 10^{-30}.)

$$k_{ij}^{(2)} = \exp\left\{-\left[\sum_{n=1}^{N}\left(1 - \left|\frac{e_n}{e_{av}}\right|\right)^2\right]^{-1/2}\right\} \tag{7}$$

$$k_{ij}^{(3)} = \frac{1}{2}\left[\frac{|\Delta x_j - \Delta x_{max}| - (\Delta x_j - \Delta x_{max})}{(\Delta x_{max} - \Delta x_{min})}\right] \tag{8}$$

$$c_i^* = \begin{cases} 1 & \text{if } c_i \le \ell n[\Delta x_i(0)/\Delta x_{min}]/t \\ \ell n[\Delta x_i(0)/\Delta x_{min}]/c_i t & \text{if } c_i > \ell n[\Delta x_i(0)/\Delta x_{min}]/t \end{cases} \tag{9}$$

$$c_i = \sum_{j=1}^{N} k_{ij} \text{ (before normalizing)} \tag{10}$$

Each of these terms serves a specific purpose in controlling the mesh movement. If the "error" in Δx_i is small relative to that in Δx_j, then Δx_j should decrease in size with further depletion of Δx_i unnecessary. On the other hand, k_{ji} should be appropriately large to enable the required change in Δx_j to occur. $k_{ij}^{(1)}$ is one function that achieves this aim. $k_{ij}^{(2)}$ insures that when the segments become distributed so that the standard deviation of errors becomes small, then the constants should accordingly tend to zero. $k_{ij}^{(3)}$ plays the role of restricting the contribution of Δx_i to Δx_j if the latter approaches some prespecified maximum permissible length. This is introduced to avoid the excessive stretching of the segments that can often lead to unnecessary truncation errors which reduce the accuracy of the calculations. Note that this upper bound can be determined and inserted during the actual computation (e.g., when $x_{\xi\xi} \simeq x_\xi^2$, see Ref. 6). Finally, c_i^* is an automatic normalizer that prevents the segments from collapsing beyond the lower bound Δx_{min}. Its derivation stems from assuming that only the depletion terms in Eq. (4) are operative. The analytic solution of these amended equations is

$$\Delta x_i(t) = \Delta x_i(0) e^{-c_i t} \tag{11}$$

whence, upon demanding that $\Delta x_i(t)$ be greater than Δx_{min}, the factor c_i^* follows.

Applied in Eq. (4), these rate constants guarantee that the movement of the mesh points in the physical domain conforms to the aforementioned criteria. Other additional criteria may be readily grafted into the rate constants (e.g., a dependence of segment length changes on the distance be-

Table 1 Data and boundary conditions for one-dimensional test problems

Problem	a	b	Initial conditions	Boundary conditions
1	0	1	$u(x,0) = 0, 0 < x < 1$	$u(0,t) = 1, t > 0$ $u(1,t) = 0, t > 0$
2	0	1	$u(x,0) = 0, 0 < x < 1$	$u(0,t) = u(1,t) = 1, t > 0$
3	$u(x,t)$	1	$u(x,0) = 0, 0 < x < 1$	$u(0,t) = 1, t > 0$ $u(1,t) = 0, t > 0$

tween segments) and it is this adaptability to the demands of the user that is an attractive feature of the method. For the solution of Eq. (4), the system is treated as linear in Δx_i, i.e., the k_{ij} are held constant for the time step t. This necessitates a slightly more stringent demand on the value of Δx_{max} so as not to overstep the upper bound. Although the solution of Eq. (4) can be obtained using standard ordinary differential equation solvers, computational overheads can grow, especially when the equations become stiff. Therefore, the following first-order approximate solution of the full-segment equations has been used, thus cutting computing costs drastically:

$$\Delta x_i(t) = \Delta x_i(0)e^{-C_i t} + \sum_{j=1}^{N} k_{ji}\Delta x_j(0)\,(1-e^{-C_j t})/C_j \qquad (12)$$

This expression satisfies Eq. (2), as required. (Note that solution of Eq. (4) can be generated iteratively using

$$\frac{\mathrm{d}(\Delta x_i^{(n)})}{\mathrm{d}t} = -C_i \Delta x_i^{(n-1)} + \sum_{j=1}^{N} k_{ji}\Delta x_j^{(n-1)}$$

with $\Delta x_i^{(0)} = \Delta x_i(0)e^{-C_i t}$ and n = iteration index.) Once Δx_i is known, the metric terms $\partial\xi/\partial x$ and $\partial\xi/\partial t$ can be computed and E can be solved in the computational domain.

Computed Results

Some preliminary test problems were solved with the sole purpose of examining the features of the proposed method. The error measure adopted here was the local gradient of the dependent variable, computed using central differences. Three problems relating to the differential equation

$$\frac{\partial u}{\partial t} = -a\frac{\partial u}{\partial x} + b\frac{\partial^2 u}{\partial x^2} \qquad (13)$$

were solved. The relevant data are summarized in Table 1. All problems were solved explicitly. The time step was automatically regulated to insure stability. Ten elementary segments were used, with Δx_{min} and Δx_{max} being assigned the (arbitrary) prefixed values 5×10^{-2} and 1.5×10^{-1}, respectively.

The computed solutions and mesh point distribution for problem 1 are shown in Fig. 1 for various times. Initially, the mesh points rush toward $x=0$. As the solution evolves toward the linear steady-state solution, the mesh point redistribute themselves until they return to uniform spacing, as anticipated. In Fig. 2 the (normalized) segment lengths at different times are shown for the second problem. Due to the error measure utilized here, segments in the vicinity of $\xi=0.5$ are always stretched beyond 0.1. As the steady state is approached the outer segments grow, while the central ones shrink toward the $\phi=0.5$ limit. The steady-state solution of problem 3 is

$$u = s\tanh[sb(1-x)/2] \qquad (14)$$

where s is the solution of

$$s = 1 + (s+1)\exp(-s\cdot b) \qquad (15)$$

Various computed solutions and mesh distributions are shown in Fig. 3. The agreement with the steady state is excellent, the final mesh distribution being not quite uniform, in keeping with the solution.

Conclusion

The principles of a new self-adaptive grid method have been presented. The crux of the method involves certain rate constants by means of which the user can regulate the mesh movement with ease according to prespecified demands. Preliminary test cases indicate the successful way the mesh distribution remains faithful to the current state of the solution. Extension of the method to general-shaped multidimensional domains is in progress and results will be reported in the future.

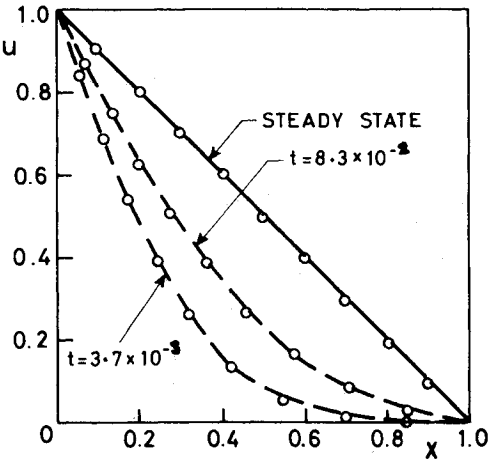

Fig. 1 Computed solution and mesh point distribution for problem 1.

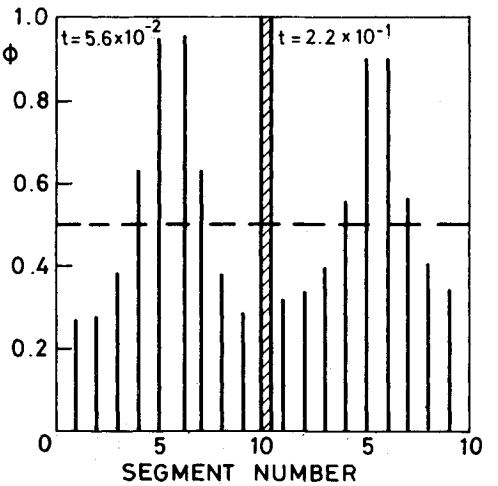

Fig. 2 Segment size distributions at different times for problem 2 [key: $\phi=(\Delta x-\Delta x_{min})/(\Delta x_{max}-\Delta x_{min})$].

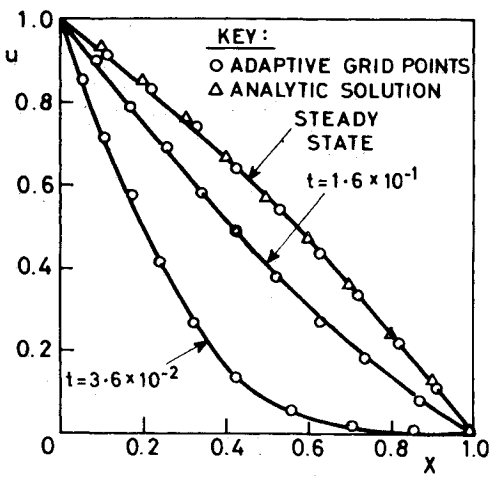

Fig. 3 Computed solution and mesh distribution for problem 3.

Acknowledgments

This research was supported by the Technion V.P.R. Fund—L. Rogow Aeronautical Research Fund.

References

[1]Hindman, R.G., Kutler, P., and Anderson, D.A., "Two-Dimensional Unsteady Euler Equation Solver for Arbitrarily Shaped Flow Regions," *AIAA Journal,* Vol. 19, April 1981, pp. 424-431.

[2]Preison, B.L. and Kutler, P., "Optimal Node Point Distribution for Improved Accuracy in Computational Fluid Dynamics," *AIAA Journal,* Vol. 18, Jan. 1980, pp. 49-54.

[3]Dwyer, H.A., Kee, R.J., and Sanders, B.R., "Adaptive Grid Method for Problems in Fluid Mechanics and Heat Transfer," *AIAA Journal,* Vol. 18, Oct. 1980, pp. 1205-1212.

[4]Rai, M.M. and Anderson, D.A., "Grid Evolution in Time Asymptotic Problems," *Journal of Computational Physics,* Vol. 43, 1981, pp. 327-344.

[5]Rai, M.M. and Anderson, D.A., "The Use of Adaptive Grids in Conjunction with Shock Capturing Methods," *Proceedings of AIAA Computational Fluid Dynamics Conference,* AIAA, New York, 1981, p. 156.

[6]Thompson, J.F., Warsi, Z.U.A., and Mastin, C.W., "Boundary Fitted Coordinate Systems for Numerical Solution of Partial Differential Equations—A Review," *Journal of Computational Physics,* Vol. 47, 1982, pp. 1-108.