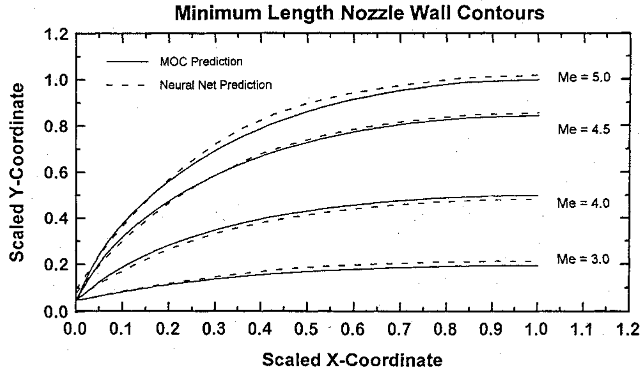

Fig. 3 Comparison of nozzle wall shapes.

Since the relation between exit Mach number and minimum length is a simple one-dimensional map, a polynomial approximation was used as the first network. The second network had a single neuron input layer, two hidden layers with four neurons each, and an output layer with four neurons.

The traditional MLN problem can be solved using method of characteristics (MOC) procedures and a program to that end was developed by the authors and used to generate training data. This phase involved randomly generating 200 exit Mach numbers in the range [1.05, 5]. The MOC program was then run for each Mach number and a fourth-order polynomial of the presented form was used to fit the shape output from the MOC program. The polynomial coefficients along with the corresponding exit Mach number and minimum length formed the training sample set. Network training was accomplished through repeated presentation of the training samples and backpropagation of error. Training both networks took 6 h on an IBM RS/6000 workstation.

The networks were then verified by randomly generating 40 Mach numbers in the range [1.05, 5] and running both the MOC code and the neural networks for each exit Mach number and obtaining the corresponding shapes. Errors were computed using

$$e_{\text{shape}} = \sum_{i=1}^{50} [s_{nn}(x_i) - s_{\text{moc}}(x_i)]^2 \tag{4}$$

where $\{x_i\}_{i=1}^{50}$ are points along the length of each nozzle. Typical results can be seen in Fig. 3. Good agreement was obtained between the neural network predicted shapes and the MOC predictions over the range of Mach numbers considered. The average error in shape was 3.9% and the average error in exit Mach number was never more than 3.1%. It must be noted that the 40 nozzle shapes were predicted by the neural networks in only 2.4 s of CPU time on an IBM RS/6000 workstation.

Conclusions

Experiments with two design problems have demonstrated the ability of neural networks to accurately and rapidly solve inverse aerodynamic problems. Once trained, a neural network is far superior computationally than any standard technique. In situations where inverse problems must be solved in real time, neural networks are certainly beneficial. Practical difficulties with this approach are associated with training data generation, sizing of the network, and choice of training algorithms. Problems more complex than those discussed here may require large amounts of computer time to solve the forward problem and, thus, it would be difficult to generate sufficient training data. Some complex problems may not have an analytic forward solution available and would need to resort to expensive experimentally generated training data. Robust neural networks designed with minimal training are needed and is an area suggested for further research.

Acknowledgments

We are grateful to Rob Markin, Karen Harwell, and Dedra Caddell, undergraduate students in the Department of Aerospace Engineering, for providing programming help on numerous occasions.

References

[1]Widrow, B., and Angell, J. B., "Reliable, Trainable Networks for Computing and Control," *Aerospace Engineering*, Vol. 21, 1962, pp. 78–123.

[2]Hecht-Nielsen, R., *Neurocomputing*, Addison–Wesley, Reading, MA, 1990.

[3]Rumelhart, D. E., and McClelland, J. L., *Parallel Distributed Processing*, MIT Press, Cambridge, MA, 1986.

[4]Hornik, M., Stinchcombe, M., and White, H., "Multilayer Feed-Forward Networks are Universal Approximators," *Neural Networks*, Vol. 2, 1989, pp. 359–366.

[5]Stinchcombe, M., and White, H., "Universal Approximation Using Feed-Forward Networks with Non-Sigmoidal Hidden Layer Activation Functions," *Proceedings of the IEEE—INS Conference on Neural Networks*, Vol. 1, 1989, pp. 185–191.

[6]Werbos, P., "Beyond Regression: New tools for Prediction and Analysis in the Behavioral Sciences," Ph.D. Dissertation, Committee on Applied Mathematics, Harvard Univ., Boston, MA, Nov. 1974.

[7]Rummelhart, D. E., Hinton, G. E., and Williams, R. J., "Learning Internal Representations by Error Propagation," *Nature*, Vol. 323, 1986, pp. 318–362.

Grid Combination Method for Hyperbolic Grid Solver in Regions with Enclosed Boundaries

Yih Nen Jeng* and Yi-Liang Shu†
National Cheng Kung University,
Tainan 70101, Taiwan, Republic of China

Introduction

THE hyperbolic equation method of grid generation[1–6] developed by Steger and Chaussee[2] is widely applied to external flow problems because it is very fast and can provide approximate grid orthogonality. However, grid oscillations frequently occur whenever the distribution of the Jacobian is not smooth[7], particularly in regions around a sharp convex or concave corner. In 1992, Chan and Steger[5] introduced several enhanced methods to eliminate these oscillations.

In Ref. 7, Tai et al. pointed out that some grid oscillations are due to the insufficient or excess damping of numerical equations proposed in Refs. 2 and 5. Subsequently, they employed the first-order upwind scheme[8] to approximate the grid equations. However, oscillation along the marching direction may still be present. Recently, a predictor-corrector method was proposed in Refs. 9 and 10, where the numerical procedure of Ref. 7 is considered as the predictor to recalculate the Jacobian. In Ref. 10, the restriction that the opposite boundary can not be specified in hyperbolic grid systems was partially alleviated by introducing a one-dimensional combination technique for the one-through type internal flow problems. This study employs the smoothing effect of the modified hyperbolic grid solver and modifies the combination technique developed in Refs. 10–12. Consequently, a hyperbolic grid solver for regions with prescribed grid points along all of the boundaries is developed.

Formulation

Steger and Chaussee[2] considered the following two equations

Received Aug. 12, 1994; revision received Dec. 6, 1994; accepted for publication Dec. 10, 1994; presented as Paper 95-0857 at the AIAA 33rd Aerospace Sciences Meeting, Reno, NV, Jan. 9–12, 1995.

*Associate Professor, Institute of Aeronautics and Astronautics. Member AIAA.

†Ph.D. Candidate, Institute of Aeronautics and Astronautics.

which define the mapping between an irregular region on the (x, y) plane and a regular region on the (ξ, η) plane:

$$x_\xi x_\eta + y_\xi y_\eta = 0$$
$$x_\xi y_\eta - x_\eta y_\xi = J^{-1} \tag{1}$$

Without loss of generality, the marching from $\eta = 0$ to $\eta = \eta_{\max}$ is considered. Steger and Chaussee approximated Eq. (1) in terms of the equations

$$A\begin{bmatrix} x_\eta \\ y_\eta \end{bmatrix} + B\begin{bmatrix} x_\xi \\ y_\xi \end{bmatrix} = \begin{bmatrix} 0 \\ J^{-1} + (J^{-1})^n \end{bmatrix}$$
$$A = \begin{bmatrix} x_\xi^n & y_\xi^n \\ -y_\xi^n & x_\xi^n \end{bmatrix}, \quad B = \begin{bmatrix} x_\eta^n & y_\eta^n \\ y_\eta^n & -x_\eta^n \end{bmatrix} \tag{2}$$

where the variables without the superscript are at an unknown level $\eta = (n+1)\Delta\eta$, and the cell area J^{-1} is unknown and should be previously specified.

Tai et al.[7] defined $W = [x, y]^T$, $C = A^{-1}B$, and $S = A^{-1} \cdot [0, J^{-1} + (J^n)^{-1}]^T$, and approximated the flux terms in terms of the first-order upwind difference scheme.[7–10] Then, the hyperbolic grid generation equations take the form

$$\tfrac{1}{2}\left[C_i^n W^{n+1} + 2W_i^{n+1} - C_i^n W_{i-1}^{n+1}\right]$$
$$= S_i + W_i^n + \lambda_i^n/2\left[W_{i+1}^{n+1} - 2W_i^{n+1} + W_{i-1}^{n+1}\right] \tag{3}$$

$$\lambda_i^n = \left[\frac{\left\{\left(x_\eta^n\right)^2 + \left(y_\eta^n\right)^2\right\}}{\left\{\left(x_\xi^n\right)^2 + \left(y_\xi^n\right)^2\right\}}\right]^{\frac{1}{2}} \tag{4}$$

Although this system works very well for many problems, there are still two additional drawbacks of two-dimensional grid generation: one is grid oscillation along the marching direction, and the other is the occurrence of shock-like grid distribution. To remedy these problems, Eq. (3) is employed as a predictor to obtain smoothly distributed J and other coefficients. The corrector[9,10] is

$$\tfrac{1}{2}\left[C_i^{n+\frac{1}{2}} W_{i+1}^{n+1} - 2W_i^{n+1} + C_i^{n+\frac{1}{2}} W_{i-1}^{n+1}\right] = +W_i^n + \left(A_i^{n+\frac{1}{2}}\right)^{-1}$$
$$\times \begin{bmatrix} 0 \\ (J^{-1})_i^{n+1} + (J^{-1})_i^n \end{bmatrix} + \frac{\lambda_i^{n+\frac{1}{2}}}{2}$$
$$\times \left[W_{i+1}^{n+1} - 2W_i^{n+1} + W_{i-1}^{n+1}\right] \tag{5}$$

To smoothly specify the Jacobian, the step-by-step strategy of Steger and Chaussee[2] is employed to estimate the Jacobian for external flow problems. For internal flow problems, associating to each point of (x_i^n, y_i^n) defines a targeting point on the opposite solid boundary and connects these points by straight parallel lines. Then a prescribed grid stretching function along the line is used to determine J.

Equations (3–5) are first-order schemes which flatten every grid curvature rapidly but make grid lines run out of the boundary around sharp convex corners. A simple and effective procedure useful here is to average those coefficients of Eqs. (3) and (5) from neighboring points at the first grid line next to the boundary. For example, the eigenvalue is calculated as

$$\bar{\lambda}_i^n = \frac{\lambda_{i-1}^n/\Delta s^- + \lambda_{i+1}^n/\Delta s^+}{1/\Delta s^- + 1/\Delta s^+}$$
$$\Delta s^- = \sqrt{\left(x_i^n - x_{i-1}^n\right)^2 + \left(y_i^n - y_{i-1}^n\right)^2}$$
$$\Delta s^+ = \sqrt{\left(x_{i+1}^n - x_i^n\right)^2 + \left(y_{i+1}^n - y_i^n\right)^2} \tag{6}$$
$$\lambda_i^n = 2^{-n}\bar{\lambda}_i^n + (1 - 2^{-n})\lambda_i^n, \qquad n = 0, 1, 2, \ldots,$$

The coefficients C_i^n, A_i^n, and J_i^n are evaluated by the same form. This method is in some sense similar to the average method of Ref. 5, but the present method does not require modification of the governing equations.

Suppose that we want to generate a grid system in an enclosed region. First, four grid systems starting from each boundary are obtained by solving Eqs. (3–6). These grid systems are named (x_l, y_l), (x_r, y_r), (x_b, y_b), and (x_t, y_t), which corresponds to grid systems starting from the left, right, bottom, and top boundaries, respectively. The final grid system is a combination of these grid systems. Using the idea of combination formulas for the control functions of the adaptive grid scheme solving elliptic equations,[11,12] the grid combination formula takes the form

$$x_i^n = s_l(x_l)_i^n + s_r(x_r)_i^n + s_b(x_b)_i^n + s_t(x_t)_i^n$$
$$y_i^n = s_l(y_l)_i^n + s_r(y_r)_i^n + s_b(y_b)_i^n + s_t(y_t)_i^n$$
$$s_l = \frac{1}{T}\frac{1}{|i-k|^a}, \qquad s_r = \frac{1}{T}\frac{1}{|i_{\max} - i - k|^a}$$
$$s_b = \frac{1}{T}\frac{1}{|n-k|^a}, \qquad s_t = \frac{1}{T}\frac{1}{|n_{\max} - n - k|^a} \tag{7}$$
$$T = \frac{1}{|i-k|^a} + \frac{1}{|i_{\max} - i - k|^a} + \frac{1}{|n-k|^a} + \frac{1}{|n_{\max} - n - k|^a}$$
$$i = 0, 1, 2, \ldots, i_{\max}, \qquad n = 0, 1, 2, \ldots, n_{\max}$$

where a and k are positive constants determining the grid quality around a boundary. For example, consider the grid distribution near the left boundary. A larger value of a, for example $a \geq 4$, expresses that a larger region of the grid distribution is dominated by (x_l, y_l).

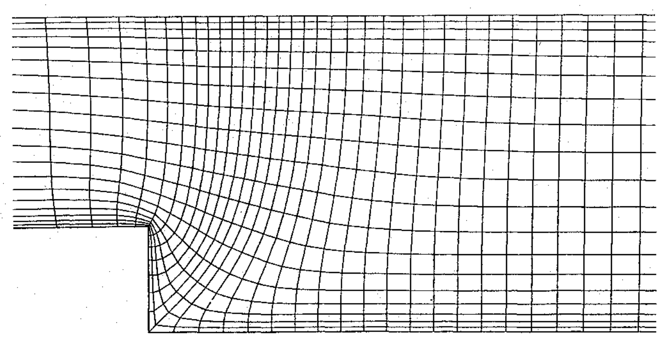

Fig. 1 Detailed grid distribution of a backward facing step flow region, generated by the present method, Eqs. (3–6).

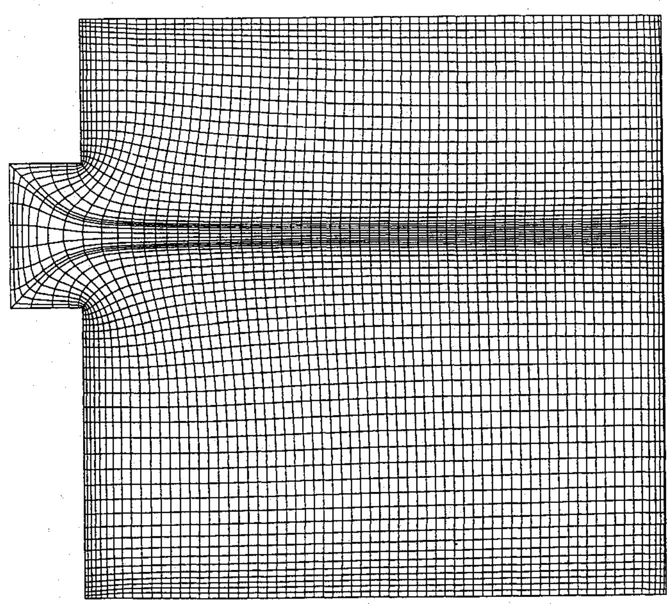

Fig. 2 Grid distribution generated by the present method, starting from the left boundary.

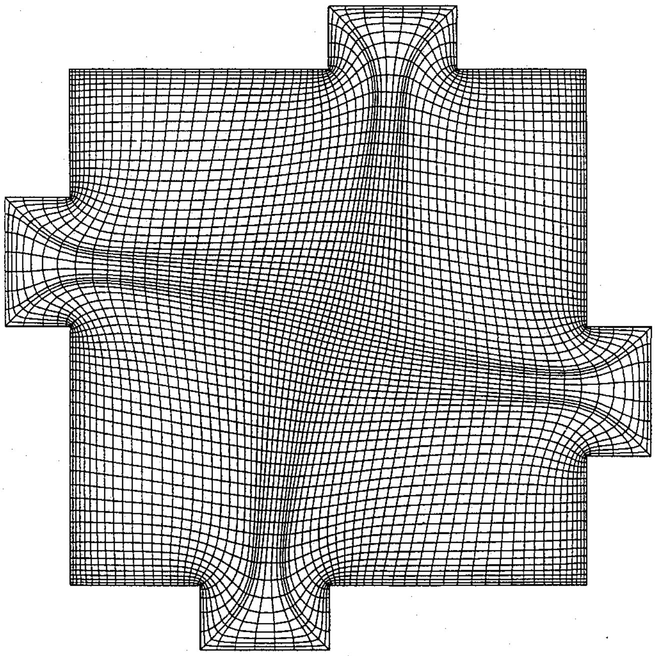

Fig. 3 Final grid system combining Fig. 8a and those starting from the right, bottom, and upper boundaries, $a = 2$ and $k = 1.4$.

The range of $1 < k < 1.5$ is employed to guarantee grid orthogonality at points next to a boundary and remote from a corner.

Results and Discussion

Figure 1 is a typical example of the application of Eqs. (3–6), which march from the lower boundary toward the upper boundary with the Hoffmann floating boundary conditions[3] on the left and right boundaries. The required CPU time is 0.2 s on a SUN Sparc X workstation and the grid size is 51×20. The targeting grid points along the upper boundary are vertical projections of the grid points on the $\eta = n\Delta\eta$ line. The Jacobian is then determined by specifying grid spacing along the vertical connecting line, which obviously has the effect of adjusting grid lines to fit the upper boundary. Grid smoothness around the lower and upper corners of the backward facing step is satisfactory. If only Eqs. (3) and (4) are employed, three drawbacks (not shown here) occur: the grid line will run out of the boundary in the upper corner, the grid line stemming from the upper corner will have a minor oscillation, and grid lines will cluster around the grid line stemming from the lower corner.

Figure 2 is the (x_l, y_l) grid system generated by the present grid solver whose grid size is 58×58. The targeting boundary is assumed to be a straight line. The grid systems generated from the bottom, right, and upper boundaries are rotational forms of Fig. 2. The result (see Fig. 3), which combines these figures and uses $a = 2$, $k = 1.4$, spent 3.0 CPU s. The grid orthogonality around all of the boundaries is preserved very well. In the central region, although grid orthogonality can not be preserved, grid points are smoothly distributed as shown.

Conclusions

A predictor-corrector version of the hyperbolic grid solver of Tai et al. is employed to construct a grid combination method for the internal flow problems. The resulting grid system preserves grid orthogonality around all of the boundaries and preserves grid smoothness in the interior region. The extension of the present study to three dimensions is simple and straightforward.

Acknowledgment

This research was supported by National Science Council of Taiwan Grant 83-0401-E006-006.

References

[1]Thompson, J. F., Thames, F. C., and Mastin, C. W., "Boundary-Fitted Coordinate Systems for Numerical Solution of Partial Differential Equations—A Review," *Journal of Computational Physics*, Vol. 47, No. 1, 1982, pp. 1–108.

[2]Steger, J. L., and Chaussee, D. S., "Generation of Body-Fitted Coordinates Using Hyperbolic Partial Differential Equations," *SIAM Journal on Scientific and Statistical Computing*, Vol. 1, No. 4, 1980, pp. 431–437.

[3]Hoffmann, K. A., Rutledge, W. H., and Rodi, P. E., "Hyperbolic Grid Generation Techniques for Blunt Body Configurations," *Numerical Grid Generation in Computational Fluid Mechanics, 88*, edited by S. Sengupta, J. Häuser, P. R. Eiseman, and J. F. Thompson, 1988, pp. 147–156.

[4]Klopfer, G. H., "Solution Adaptive Meshes with A Hyperbolic Grid Generator," *Numerical Grid Generation in Computational Fluid Mechanics, 88*, edited by S. Sengupta, J. Häuser, P. R. Eiseman, and J. F. Thompson, 1988, pp. 443–453.

[5]Chan, W. M., and Steger, J. L., "Enhancements of a Three-Dimensional Hyperbolic Grid Generation Scheme," *Applied Mathematics and Computations*, Vol. 51, No. 1, 1992, pp. 181–205.

[6]Steger, J. L., and Rizk, Y. M., "Generation of Three-Dimensional Body Fitted Coordinates Using Hyperbolic Partial Differential Equations," NASA TM 86753, June 1985.

[7]Tai, C. H., Yih, S. L., and Soong, C. Y., "A Novel Hyperbolic Grid Generation Procedure with Inherent Adaptive Dissipation," *Journal of Computational Physics* (to be published).

[8]Roe, P. L., "Approximate Riemann Solvers, Parameters, and Difference Schemes," *Journal of Computational Physics*, Vol. 43, No. 2, 1981, pp. 357–372.

[9]Jeng, Y. N., and Shu, Y. L., "On a Smoothing Technique of the Grid Distribution for the Hyperbolic Grid Solver," *Proceedings of the 2nd National Conference on Computational Fluid Dynamics*, AASRC, Taiwan, ROC, 1993, pp. 481–484.

[10]Jeng, Y. N., Shu, Y. L., and Lin, W. W., "On Grid Generation for Internal Flow Problems by Methods Solving Hyperbolic Equations," *Numerical Heat Transfer* (to be published).

[11]Jeng, Y. N., and Liou, Y. C., "Adaptive Grid Generation by Elliptic Equations with Grid Control at all of the Boundaries," *Numerical Heat Transfer*, Pt. B, Vol. 23, No. 1, 1993, pp. 135–151.

[12]Jeng, Y. N., and Liou, Y. C., "A New Adaptive Grid Generation by Elliptic Equations with Orthogonality at all of the Boundaries," *J. Sci. Comput.*, Vol. 7, No. 1, 1992, pp. 63–80.

Semi-Infinite Channel Flows Past a Cylindrical Body: A Numerical Study

J.-H. Chen*
National Taiwan Ocean University,
Keelung 202, Taiwan, Republic of China

Introduction

THE two-dimensional flow past around a circular cylinder has always been an interesting problem due to its simple geometry but complicated physical phenomena. Traditionally, the study is based on flows in an infinite domain with a uniform incoming velocity far upstream. Much experimental data are available in the literature (see, for example, Ref. 1).

Although variations of the flow pattern are easily observed in experiments, few detailed reports on numerical examinations and theoretical studies are available (e.g., Refs. 2 and 3). In fact, it is difficult to simulate these flow phenomena rigorously. This arises, in part, because the decay of wake is not clearly understood and, therefore, no mathematically or physically rigorous downstream boundary conditions are known.

To avoid this difficulty, this study attempts to examine the flow by placing the cylinder in the entrance region of a semi-infinite channel flow. The emergence of the recirculating region is studied and the results are compared with those in an unbounded flow domain.

Received April. 29, 1993; presented as Paper 93-3062 at the AIAA 24th Fluid Dynamics Conference, Orlando, FL, July 6–9, 1993; revision received Sept. 10, 1994; accepted for publication Sept. 20, 1994.

*Associate Professor, Department of Naval Architecture. Member AIAA.