Table 1 Comparison of simulation shock speeds with experiment[9]

Distance from diaphragm, m	Simulation, km/s	Experiment, km/s
9.35	3.00	2.92
11.64	2.89	2.85
14.26	2.83	2.75

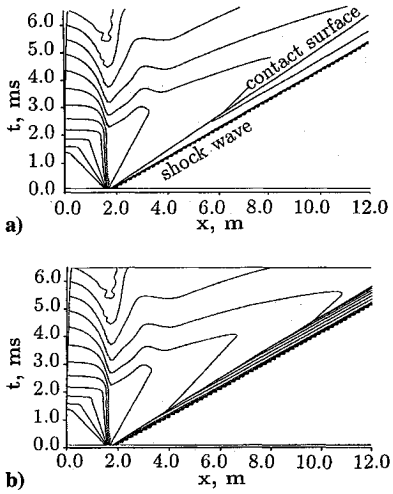

Fig. 2 Space–time diagrams comparing two different boundary-layer models; contours are equally spaced values of log ρ. In each case the driver gas was H_2, the shock tube initial pressure 206.7 kPa, and the diaphragm pressure ratio 445: a) the model without mass loss[1,7] and b) the present mass-loss model.

the gas was hydrogen. The shock tube was initially at 68.9 kPa and 296 K, and the gas was air in thermochemical equilibrium. Shock speed measurements were made along the length of the shock tube, and a comparison between the experimental data and the simulations was made in Table 1. Simulation and experiment differ in the range of 1–3%.

Concluding Remarks

By using fundamental boundary-layer theory, a model has been derived that can be used to provide shock tube test time and shock speed predictions. This model can be incorporated easily into a quasi-one-dimensional simulation code to provide a useful tool to the designer of shock and expansion tubes and gives accurate test time predictions for all turbulent flow shock tube cases tested.

Although they do not provide the depth of quality of information available in two-dimensional viscous codes, quasi-one-dimensional formulations can give adequate design information using a computationally less expensive approach. An adaptive one-dimensional mesh would aid the computational efficiency as, at this stage, the mass loss traps many cells with very little mass near the contact surface at long distances from the diaphragm ($L/D \approx 100$).

References

[1]Jacobs, P. A., "Quasi-One-Dimensional Modeling of a Free-Piston Shock Tunnel," *AIAA Journal*, Vol. 32, No. 1, 1994, pp. 137–145.

[2]Tani, K., Itoh, K., Takahashi, M., Tanno, H., Komuro, T., and Miyajima, H., "Numerical Study of Free-Piston Shock Tunnel Performance," *Shock Waves Journal*, Vol. 3, No. 4, 1994, pp. 313–319.

[3]Sharma, S. P., and Wilson, G. J., "Test-Times in Hypersonic Shock Tubes," AIAA Paper 95-0713, Jan. 1995.

[4]Mirels, H., "Shock Tube Test Time Limitation Due to Turbulent-Wall Boundary Layer," *AIAA Journal*, Vol. 2, No. 1, 1964, pp. 84–93.

[5]Watson, R. D., "Generalized Velocities in the Outer Region of Hypersonic Turbulent Boundary Layers," *AIAA Journal*, Vol. 17, No. 8, 1979, pp. 919–921.

[6]Schetz, J. A., *Boundary Layer Analysis*, Prentice–Hall, Englewood Cliffs, NJ, 1993, pp. 240, 241.

[7]Groth, C. P. T., Gottlieb, J. J., and Sullivan, P. A., "Numerical Investigation of High Temperature Effects in UTIAS-RPI Hypersonic Impulse Tunnel," *Canadian Journal of Physics*, Vol. 69, No. 7, 1991, pp. 897–918.

[8]Lacey, J. J., "Experimental Shock Tube Test Time—Turbulent Regime," *Proceedings of the 7th Shock Tube Symposium*, Univ. of Toronto Press, Toronto, ON, Canada, 1969, pp. 126–142.

[9]Fuehrer, R. G., "Measurements of Incident-Shock Test Time and Reflected Shock Pressure at Fully Turbulent Boundary-Layer Test Conditions," *Proceedings of the 7th Shock Tube Symposium*, Univ. of Toronto Press, Toronto, ON, Canada 1969, pp. 31–59.

Hyperbolic Equation Method of Grid Generation for Enclosed Regions

Yih Nen Jeng*
National Cheng Kung University, Tainan 70101, Taiwan, Republic of China
and
Yuan-Chang Liou†
Kung Shan Institute of Technology and Commerce, Tainan 70111, Taiwan, Republic of China

Introduction

In the recent past, two popular types of method for grid system generation have been developed.[1] The first type employs algebraic formulas, and the second type solves elliptical, parabolic, or hyperbolic partial differential equations.[1–8] Among the second type, the hyperbolic equation technique is very fast and has been extensively studied.[5–10]

Because of the open boundary constraint of the hyperbolic system, the classical hyperbolic equation method is not available for grid generation within enclosed regions. This limitation is partially removed by the combination methods of Refs. 9 and 10, which sum up grids generated from each boundary separately. However, combination methods require at least twice the CPU time as the original method.

In a recent study,[11] the authors combined the grid front marching method[12] and the parabolic equation method. In this study, a similar idea is applied to extend the Cordova and Barth hyperbolic equation method[13] to enclosed regions.

Formulation

Cordova and Barth[13] considered the following two equations to define the mapping between an irregular region on the x–y plane and a regular region on the ξ–η plane:

$$x_\xi x_\eta + y_\eta y_\xi = s_\xi s_\eta \cos\theta, \qquad x_\xi y_\eta - x_\eta y_\xi = s_\xi s_\eta \sin\theta \quad (1)$$

where $x_\xi = \partial x/\partial\xi$ and θ denotes the angle between two intersecting grid lines. Assuming that x^o and y^o along the grid level are known and x and y along an adjacent grid level are to be found, the grid equations can be linearized[5] into the following forms:

$$\boldsymbol{A}\boldsymbol{r}_\xi + \boldsymbol{B}\boldsymbol{r}_\eta = \boldsymbol{S}$$

$$\boldsymbol{A} = \begin{bmatrix} x^o_\eta & y^o_\eta \\ y^o_\eta & -x^o_\eta \end{bmatrix}, \qquad \boldsymbol{B} = \begin{bmatrix} x^o_\xi & y^o_\xi \\ -y^o_\xi & x^o_\xi \end{bmatrix} \quad (2)$$

$$\boldsymbol{S} = \begin{bmatrix} s_\xi s_\eta \cos\theta + s^o_\xi s^o_\eta \cos\theta^o \\ s_\xi s_\eta \sin\theta + s^o_\xi s^o_\eta \sin\theta^o \end{bmatrix}$$

Received Sept. 18, 1995; revision received Feb. 12, 1996; accepted for publication Feb. 12, 1996. Copyright © 1996 by Yih Nen Jeng and Yuan-Chang Liou. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission.

*Professor, Institute of Aeronautics and Astronautics. Member AIAA.
†Associate Professor, Department of Mechanical Engineering.

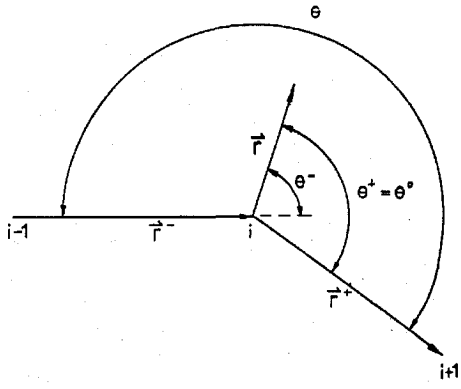

Fig. 1 Schematic diagram of θ^+ and θ^-.

where s_ξ, s_η, and θ on the right-hand side are estimated via the algebraic grid generation method. These equations, which apply the upwind splitting method of Tai et al.,[8] are properly approximated by the upwind scheme.

In the author's experience, if the terms on the right-hand side of Eq. (2) are not smooth enough, even the modified methods of Refs. 8–10 cannot successfully eliminate grid oscillation. In this study the algebraic grid generation method provides smooth distribution of s_ξ, s_η, and θ. Consequently, the enhancement techniques of the hyperbolic equation method[5–10] are not necessary.

If the grid front is started from all of the boundaries, an intuitive method can solve the hyperbolic grid equations associated with the enclosed grid front next to all of the boundaries. This results in a set of 2×2 block-tridiagonal system of simultaneous equations with periodic boundary conditions. Another method is to separately apply Eq. (2) to each boundary. The multiple values of x and y at the four corners are made unique by applying the direct average. Because all of the test cases show that the difference between these two methods is insignificant, only the results of the second method are discussed in this work.

As shown in Ref. 11, the grid front marching method modifies grid angles around sharp convex and concave corners. After employing a simple grid smoother preserving original grid spacings to smooth all of the boundaries, the application of the Soni–Hermite transfinite interpolation method generates one level of grids next to all of the boundaries. By considering the new grid level as the new boundaries, the procedure is repeatedly applied. In this study, the same procedure is equipped with a different algebraic method. First, a grid point is determined through the proper specification of grid spacing and grid angle. Then a combination formula similar to that proposed in Ref. 3 is employed to include the linear interpolation between opposite boundaries, say,

$$x_\eta = (x_\xi \cos\theta - y_\xi \sin\theta) s_\eta^\ell / s_\xi$$
$$y_\eta = (x_\xi \sin\theta + y_\xi \cos\theta) s_\eta^\ell / s_\xi \tag{3a}$$

$$\theta = \omega_{ij}\theta^o + (1 - \omega_{ij})\theta^\ell$$
$$\theta^\ell = \cos^{-1} \frac{x_\xi x_\eta^\ell + y_\xi y_\eta^\ell}{s_\xi s_\eta^\ell} \tag{3b}$$

$$s_\eta^\ell = \sqrt{\left(x_\eta^\ell\right)^2 + \left(y_\eta^\ell\right)^2}\Big|_{i,j}, \qquad s_\xi = \sqrt{x_\xi^2 + y_\xi^2}\Big|_{i,j}$$
$$x_\xi = x_{i+1,j} - x_{i,j}, \qquad y_\xi = y_{i+1,j} - y_{i,j}$$
$$\theta^o = \text{desired angle measured from } \boldsymbol{r}^+ \text{ to } \boldsymbol{r}$$
$$\boldsymbol{r} = (x_{i,j+1} - x_{i,j})\boldsymbol{i} + (y_{i,j+1} - y_{i,j})\boldsymbol{j} \tag{3c}$$
$$\boldsymbol{r}^+ = (x_{i+1,j} - x_{i,j})\boldsymbol{i} + (y_{i+1,j} - y_{i,j})\boldsymbol{j}$$
$$x_\eta^\ell = (x_{i,n} - x_{i,j})\Delta s_{i,j}, \qquad y_\eta^\ell = (y_{i,n} - y_{i,j})\Delta s_{i,j}$$

where ω_{ij} is the control weighting function that will be discussed later; $\Delta s_{i,j}$ is the prescribed normalized arclength that is evaluated from the linear interpolation between $\Delta s_{0,j}$ and $\Delta s_{i_{\max},j}$ with i as the independent variable; and $\Delta s_{0,j}$ and $\Delta s_{i_{\max},j}$ are the normalized arclengths (between points j and n) along $i = 0$ and $i = i_{\max}$ boundaries, respectively.

In Eqs. (3a–3c), the desired angle θ^o takes a value of 90 deg. However, at a sharp convex or concave corner along a boundary, a suitable angle must bisect the angle θ between $\boldsymbol{r}^-$ and $\boldsymbol{r}^+$ rather than take a value of 90 deg, where $\boldsymbol{r}^- = (x_{i,j} - x_{i-1,j})\boldsymbol{i} + (y_{i,j} - y_{i-1,j})\boldsymbol{j}$. To provide a smooth angle distribution around a corner, let $\theta^+ = \theta^o$ and define θ^- to be the angle measured from $\boldsymbol{r}^-$ to $\boldsymbol{r}$, which satisfies the conditions that

$$\theta^o = \theta^+ = \theta/2, \qquad \theta^- = 180 \text{ deg} - \theta^o \tag{4}$$

as shown in Fig. 1. The corner angle at the left-hand end of a boundary is defined as θ^+, whereas that at the right-hand end is defined as θ^-. Then the following angle smoother is employed:

$$\theta_{i,j}^o = \left(\theta_{i,j}^+\right)_{\text{new}} = \left(\theta_{i,j}^+\right)_{\text{old}} + \Delta\theta_{i,j}$$
$$\left(\theta_{i,j}^-\right)_{\text{new}} = \left(\theta_{i,j}^-\right)_{\text{old}} + \Delta\theta_{i,j} \tag{5}$$
$$\Delta\theta_{i,j} = \bar{\theta}_{i,j} - \frac{\left(\theta_{i,j}^+\right)_{\text{old}} + \left(\theta_{i,j}^-\right)_{\text{old}}}{2}$$
$$\bar{\theta}_{i,j} = (1 - \omega_q)\frac{\left(\theta_{i,j}^+\right)_{\text{old}} + \left(\theta_{i,j}^-\right)_{\text{old}}}{2} + \omega_q \frac{\left(\theta_{i-1,j}^+\right)_{\text{old}} + \left(\theta_{i+1,j}^-\right)_{\text{old}}}{2}$$

where ω_q takes a value of 0.9. The end boundary effect of an interior grid line can be partly reflected by repeatedly applying Eq. (5), and the remaining effect is approximated by the following modifications:

$$\Delta s_{0,j} = \Delta s_{0,j} \sin\theta_{0,j}^+ \qquad \text{if} \quad \theta_{0,j}^+ < 90 \text{ deg}$$
$$\Delta s_{i_{\max},j} = \Delta s_{i_{\max},j} \sin\theta_{i_{\max},j}^- \qquad \text{if} \quad \theta_{i_m,j}^- > 90 \text{ deg} \tag{6}$$

where situations of $\theta_{0,j}^+ \geq 90$ deg and $\theta_{i_{\max},j}^- \leq 90$ deg need not be modified.

The weighting function next to the bottom and top boundaries ($j = j_0$ and j_n) is defined by the following equation:

$$\omega_{ij} = \left[1 - \frac{j - 0.99}{(j_{\max}/2) - 0.99}\right]^a, \qquad \text{if} \quad j \leq j_{\max}/2$$
$$= \left[1 - \frac{j_{\max} - j - 0.99}{(j_{\max}/2) - 0.99}\right]^a, \qquad \text{if} \quad j > j_{\max}/2 \tag{7}$$

For the grid level next to the left- and right-hand boundaries ($i = i_0$ and i_m), the weighting function takes a similar form. Then, to avoid multiple definitions of x and y at the four corner points, direct averaging gives a unique definition of the grid point.

When starting to generate grids within an enclosed region, the known level is composed of $i_0 = 0$, $i_m = i_{\max}$, $j_0 = 0$, and $j_n = j_{\max}$ boundaries. As the grid level marches inward, these boundaries move inward by adding 1 to i_0 and j_0, respectively, and subtracting 1 from i_m and j_n, respectively. However, normalized arclengths corresponding to $s_{0,j}$ and $s_{i_{\max},j}$ are evaluated along the original boundaries.

By applying Eqs. (4–7) to estimate the unknown s_ξ, s_η, and θ on the right-hand side of Eq. (2), one grid front marching step can be completed by solving the discretized hyperbolic equations. The marching stops if the remaining region disappears or collapses into a single grid line. For the latter case, we again march the level one more step ahead without estimating the unknown levels. Then we directly average the results. Note that the marching procedure can start either from a single boundary, from a pair of opposite boundaries, or from all of the boundaries, provided that the weighting functions are properly defined. Since all of the procedures of the present study are very simple, the extension of the proposed method to three dimensions is easy and straightforward.

Results and Discussions

Figure 2 (38×20) is the result of starting grid fronts from top and bottom boundaries simultaneously and applying the angle smoother ($\omega_q = 0.9$) twice. Obviously, grid smoothness is satisfactory over

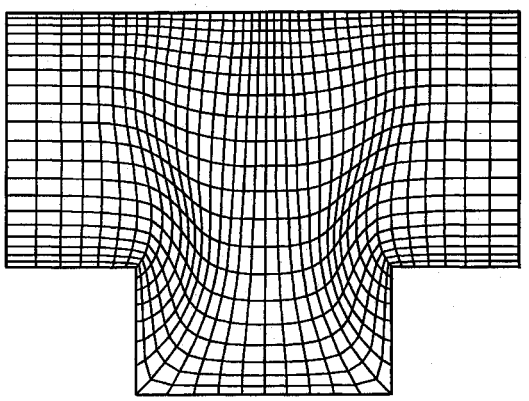

Fig. 2 Grid distribution for a channel with cavity (38 × 20), generated by proposed method starting from the lower and top boundaries, where the angle smoother (ω_q = 0.9) is applied twice.

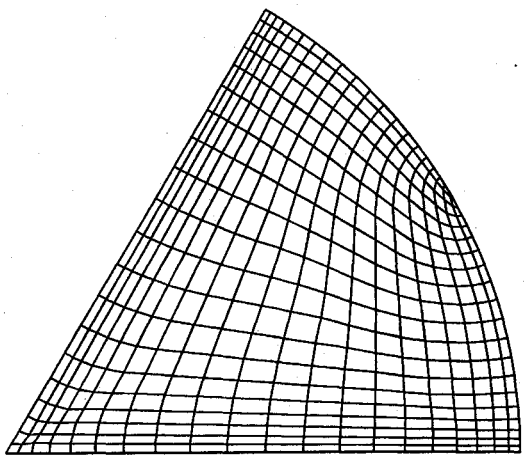

Fig. 3 Grid distribution in a 60-deg sector (20 × 20), generated by the proposed method starting from all of the boundaries, where the angle smoother (ω_q = 0.9) is applied three times.

the entire domain, and the convex and concave corners along the bottom boundary do not cause any trouble.

The second example for a region with enclosed boundaries is a 60-deg sector region. The result of the proposed method with the angle smoother applied three times ($\omega_q = 0.9$) is shown in Fig. 3 (20 × 20), where approximate grid orthogonality is preserved very well in the interior region. In the region around the flattened corner, the angle smoother makes the grid line incline to the flattened corner so that the resulting grid is smoothly distributed.

Conclusions

Cordova and Barth's grid generation method solving the hyperbolic partial differential equation is successfully extended to problems with enclosed boundaries. The open boundary constraint is partially removed by properly prescribing the unknown grid level via a simple algebraic grid generator equipped with the angle smoother and using corrections from the side boundaries. Several examples show that the proposed method preserves approximate grid orthogonality around all of the desired boundaries and provides smooth grid distribution over the entire domain.

Acknowledgment

This work was supported by the National Science Council of Taiwan under Grant 84-2212-E006-090.

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, 1982, pp. 1–108.

[2]Nakamura, S., "Marching Grid Generation Using Parabolic Partial Differential Equations," *Numerical Grid Generation*, edited by J. F., Thompson, Elsevier, Oxford, England, UK, 1982, pp. 775–786.

[3]Noack, R. W., "Inviscid Flow Field Analysis of Maneuvering Hypersonic Vehicle Using the SCM Formulation and Parabolic Grid Generation," AIAA Paper 85-1682, July 1985.

[4]Hodge, J. K., Leone, S. A., and McCarty, R. L., "Noniterative Parabolic Grid Generation for Parabolized Equations," *AIAA Journal*, Vol. 25, No. 4, 1987, pp. 542–549.

[5]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.

[6]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, Pineridge, Swansea, Wales, UK, 1988, pp. 147–156.

[7]Chan, W. M., and Steger, J. L., "Enhancements of a Three-Dimensional Hyperbolic Grid Generation Scheme," *Applied Mathematics and Computation*, Vol. 51, 1992, pp. 181–205.

[8]Tai, C. H., Yih, S. L., and Soong, C. Y., "A Novel Hyperbolic Grid Generation Procedure with Inherent Adaptive Dissipation," *Journal of Computational Physics*, Vol. 116, 1995, pp. 173–179.

[9]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, Part B*, Vol. 27, 1995, pp. 43–61.

[10]Jeng, Y. N., and Shu, Y. L., "Grid Combination Method for Hyperbolic Grid Solver in Regions with Enclosed Boundaries," *AIAA Journal*, Vol. 33, No. 6, 1995, pp. 1152–1154.

[11]Liou, Y. C., and Jeng, Y. N., "The Algebraic Grid Front Marching Methods," *Numerical Heat Transfer, Part B*, Vol. 28, 1995, pp. 257–276.

[12]Jeng, Y. N., and Liou, Y. C., "The Parabolic Equation Method of Grid Generation for Enclosed Regions," *Numerical Heat Transfer, Part B* (to be published).

[13]Cordova, J. Q., and Barth, T. J., "Grid Generation for General 2-D Regions Using Hyperbolic Equations," AIAA Paper 88-0520, Jan. 1988.

Calculation of Interlaminar Stresses in Laminated Plates Using Walsh Transforms

Martin Crane* and J. T. Boyle†
University of Strathclyde,
Glasgow G1 1XQ, Scotland, United Kingdom

Nomenclature

A, B, D	= extensional/extensional, extensional/bending and bending/bending coupling terms, respectively
$B_{(m)}$	= mth order Walsh operational matrix for differentiation, $H^{-1}_{(m)}$
C_i	= Walsh coefficients
F	= two-dimensional matrix of Walsh coefficients, f_{ij}
$H_{(m)}$	= mth order Walsh operational matrix for integration
h	= plate thickness
l	= plate span in x direction
M^k	= kth laminar moment resultants
N^k	= kth laminar stress resultants
$\bar{Q}^k_{ij}$	= kth laminar transformed reduced stiffnesses
u, v, w	= displacements in x, y, z directions, respectively
$W_{(m)}$	= mth-order Walsh transform matrix
ϵ	= strains
κ	= curvatures
σ	= normal stresses
τ	= shear stresses
$\Phi_{(m)}$	= vector of ϕ_i functions for $i = 1, \ldots, m$
$\phi_i(x)$	= ith one-dimensional Walsh function in x

I. Introduction

SINCE their development in 1923 by Walsh,[1] Walsh functions have found wide applications in the field of engineering compu-

Received Feb. 8, 1995; revision received July 7, 1995; accepted for publication July 20, 1995. Copyright © 1995 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved.

*Research Fellow, Department of Mechanical Engineering.
†Professor, Department of Mechanical Engineering.