# Elliptic Grid Generation System: Control Functions Revisited—I

Bharat K. Soni

*NSF Engineering Research Center for Computational Field Simulation*
*Mississippi State University*
*Mississippi State, Mississippi 39762*

Transmitted by Joe F. Thompson

ABSTRACT

A methodology for the development of control functions associated with the elliptic grid generation system is presented. This development is based on the optimal blending of the best characteristics of hyperbolic generation system and algebraic techniques. The algorithm is readily applicable to grid adaptation. Computational examples are presented to demonstrate the success of this methodology.

## 1. INTRODUCTION

During this decade, numerical grid generation has evolved as an essential tool in obtaining numerical solutions of the partial differential equations of fluid dynamics. A multitude of techniques and computer codes [1–8] has been developed to support multiblock grid generation requirements associated with general complex configurations. Grid generation methodologies can be grouped into two main categories: (a) direct methods, where algebraic interpolation techniques are used, and (b) Indirect methods, where a set of partial differential equations is solved. Both of these techniques are utilized either separately or in combination to efficiently generate grids in the aforementioned computer codes. In algebraic methods, the widely used technique is transfinite interpolation [1–3] and weighted transfinite interpolation [4–7]. The weighted transfinite interpolation technique has been combined with Bezier and B-spline curve definition to accomplish grid refinement (smoothness and orthogonality) by Soni [6, 11, 12]. The control point




655 Avenue of the Americas, New York, NY 10010 0096-3003/93/$6.00

formulation, assuming the polynomial representation of boundaries is developed by Eiseman [10]. The commonly used indirect methods are elliptic systems with appropriate control functions [1–8] and hyperbolic systems [9].

The development of control functions associated with elliptic grid generation system presented in this paper adapts the best features of algebraic system and hyperbolic generation system. This development is based on the following observations and evaluations: Algebraic systems are fast and economical, precise spacing control (a well-distributed grid) is always achieved with algebraic systems, Grid generation by elliptic systems is always smooth, Algebraic systems may cause grids to overlap, however, elliptic system resist grid line overlapping; weighted transfinite interpolation methods blended with Bezier, B-spline curves/surfaces [6, 11, 12] can produce well-distributed, near orthogonal (at boundaries) and smooth grids (in most cases); the control functions can be formulated to achieve boundary orthogonality and spacing control (near solid boundary surfaces) by elliptic generation systems [1–7, 8]; the control functions can be formulated to accomplish field orthogonality in a given computational direction ($\xi$, $\eta$ or $\zeta$) and spacing control by elliptic generation systems [3] by iteratively updating various terms in the generation system.This is very time consuming, especially in three-dimensional problems. Algebraic systems require a high degree of understanding and visual user interaction. However, elliptic systems can be readily adaptable for generalization. This is extremely useful in grid adaptation. The hyperbolic system preserves the orthogonality at the solid boundary and the point distribution in the field. However, its applicability is restricted to external flows where the accurate geometrical shape of the outer boundaries/surfaces is not important as long as their location is a certain distance away from the body. Also, in three-dimensional applications of hyperbolic systems, the grid quality is directly influenced by the characteristics of the surfaces associated with the computational domain. For general purpose applications involving arbitrary configurations, elliptic generation systems represent the indirect method [1–8] utilized in most grid generation problems.

The development of control functions presented in this paper is an outcome of the aforestated observations. A two-dimensional form of elliptic system can be written as:

$$g_{22}(\underline{r}_{\xi\xi} - \phi\underline{r}_{\xi}) + g_{11}(\underline{r}_{\eta\eta} - \psi\underline{r}_{\eta}) - 2g_{12}\underline{r}_{\xi\eta} = 0 \qquad (1)$$

where

$$\underline{r} = (x, y) \quad \text{physical space}$$

$$(\xi, \eta) \quad \text{computational space}$$

$$g_{11} = \underline{r}_\xi \cdot \underline{r}_\xi = x_\xi^2 + y_\xi^2$$

$$g_{12} = \underline{r}_\xi \cdot \underline{r}_\eta = x_\xi x_\eta + y_\xi y_\eta$$

$$g_{22} = \underline{r}_\eta \cdot \underline{r}_\eta = x_\eta^2 + y_\eta^2$$

$\phi, \psi$ control functions

The usual practice is to assume orthogonality [13], i.e.,

$$g_{12} = \underline{r}_\xi \cdot \underline{r}_\eta = 0$$

and by taking the dot product of [1] by $\underline{r}_\xi$ we obtain

$$g_{22}(\underline{r}_{\xi\xi} \cdot \underline{r}_\xi - \phi \underline{r}_\xi \cdot \underline{r}_\xi) + g_{11}(\underline{r}_{\eta\eta} \cdot \underline{r}_\xi - \psi \underline{r}_\xi \cdot \underline{r}_\eta) = 0$$

and hence

$$\phi = \frac{\underline{r}_{\xi\xi} \cdot \underline{r}_\xi}{\underline{r}_\xi \cdot \underline{r}_\xi} + \frac{\underline{r}_{\eta\eta} \cdot \underline{r}_\xi}{\underline{r}_\eta \cdot \underline{r}_\eta} \tag{2}$$

and similarly

$$\psi = \frac{\underline{r}_{\eta\eta} \cdot \underline{r}_\eta}{\underline{r}_\eta \cdot \underline{r}_\eta} + \frac{\underline{r}_{\xi\xi} \cdot \underline{r}_\eta}{\underline{r}_\xi \cdot \underline{r}_\xi} \tag{3}$$

In most of the techniques developed to date, the evaluation of $\phi$ is accomplished in $\xi$ directions (on the $\eta$ min and $\eta$ max boundaries) by evaluating $\underline{r}_\xi$ and $\underline{r}_{\xi\xi}$ using finite difference expressions, by evaluating $\underline{r}_\eta$ and $\underline{r}_{\eta\eta}$ using the current grid (updated every iteration), and then by solving [3]:

$$\underline{r}_\xi \cdot \underline{r}_\eta = 0, \quad \text{and}$$

$$|\underline{r}_\eta| = \text{distance off the } \xi = \xi \text{ min or } \xi = \xi \text{ max boundary}$$

Similarly, forcing function $\psi$ is evaluated on $\xi$ min and $\xi$ max boundaries. The control functions in the interior field are interpolated from the boundaries. In the present effort a similar philosophy is utilized to formulate two algorithms blending the best characteristics of algebraic grids and equations

involved in hyperbolic generation systems for the evaluation of control functions.

## 2. ALGORITHM I: CELL AREA APPROACH

Evaluation of $\phi$ is accomplished using the entire grid as follows:

$$\underline{r}_\xi \cdot \underline{r}_\eta = x_\xi x_\eta + y_\xi y_\eta = 0 \tag{4}$$

and

$$\|\underline{r}_\xi \underline{x} \underline{r}_\eta\| = x_\xi y_\eta - x_\eta y_\xi = v \tag{5}$$

i.e.

$$\begin{bmatrix} x_\xi & y_\xi \\ -y_\xi & x_\xi \end{bmatrix} \begin{bmatrix} x_\eta \\ y_\eta \end{bmatrix} = \begin{bmatrix} 0 \\ v \end{bmatrix}$$

The matrix

$$\begin{bmatrix} x_\xi & y_\xi \\ -y_\xi & x_\xi \end{bmatrix}$$

is nonsingular because $g_{11} = x_\xi^2 + y_\xi^2 \neq 0$. (It is assured that the grid points where points/lines lie on top of each other in physical space resulting in $g_{11} = 0$ will be treated as special cases.)

Here $v$ is obtained (and kept fixed) from the initial algebraic grid. Notice that $v$ represents the areas covered by the grid cells in the field and from the properties discussed earlier the algebraic grid has precise control on the grid distribution and $\underline{r}_\xi = (x_\xi, y_\xi)$ is known.

Compute $\underline{r}_{\xi\eta}$ by solving

$$(\underline{r}_\xi \cdot \underline{r}_\eta)_\xi = (x_\xi x_\eta + y_\xi y_\eta)_\xi = 0 \tag{6}$$

and

$$\|\underline{r}_\xi \underline{x} \underline{r}_\eta\|_\xi = (x_\xi y_\eta - x_\eta y_\xi)_\xi = v_\xi \tag{7}$$

Note that this system results in

$$\begin{bmatrix} x_\xi & y_\xi \\ -y_\xi & x_\xi \end{bmatrix} \begin{bmatrix} x_{\xi\eta} \\ y_{\xi\eta} \end{bmatrix} = \begin{bmatrix} -x_{\xi\xi} x_\eta - y_{\xi\xi} y_\eta \\ v_\xi - x_{\xi\xi} y_\eta + x_\eta y_{\xi\xi} \end{bmatrix}$$

Since $x_\eta$ and $y_\eta$ were evaluated earlier, each term is known except $x_{\xi\eta}$, $y_{\xi\eta}$. Also, $v_\xi$ represents the change in area in the $\xi$ direction ($v_\xi$ are kept fixed throughout the refinement process).

Compute $\underline{r}_{\eta\eta}$ by solving

$$(\underline{r}_\xi \cdot \underline{r}_\eta)_\eta = (x_\xi x_\eta + y_\xi y_\eta)_\eta = 0 \tag{8}$$

$$\|\underline{r}_\xi \underline{x} \underline{r}_\eta\|_\eta = (x_\xi y_\eta - x_\eta y_\xi)_\eta = v_\eta \tag{9}$$

Note that the resulting system associated with unknown $x_{\eta\eta}$, $y_{\eta\eta}$ is

$$\begin{bmatrix} x_\xi & y_\xi \\ -y_\xi & x_\xi \end{bmatrix} \begin{bmatrix} x_{\eta\eta} \\ y_{\eta\eta} \end{bmatrix} = \begin{bmatrix} -x_{\xi\eta} x_\eta - y_{\xi\eta} y_\eta \\ v_\eta - x_{\xi\eta} y_\eta + x_\eta y_{\xi\eta} \end{bmatrix}$$

$v_\eta$ represents the change in area in the $\eta$ direction, ($v_\eta$ are kept fixed throughout the refinement process).

$\phi$ is then computed by using formula (2). Similarly, $\psi$ is computed by using formula (3) and by evaluating: (i) $\underline{r}_\eta$ and $\underline{r}_{\eta\eta}$ using appropriate finite difference at every grid point, (ii) $\underline{r}_\xi$ by solving equations (4) and (5) for $x_\xi$ and $y_\xi$, (iii) $\underline{r}_{\xi\eta}$ by solving equations (8) and (9) for $x_{\xi\eta}$, $y_{\xi\eta}$, and (v) $\underline{r}_{\xi\xi}$ by solving equations (6) and (7) for $x_{\xi\xi}$, $y_{\xi\xi}$.

## 3. ALGORITHM II: GRID SPACING APPROACH

Evaluation of $\phi$ is accomplished using the entire grid as follows:

a. $\underline{r}_\xi$ and $\underline{r}_{\xi\xi}$ are computed using appropriate finite difference approximations at every grid point

b. $\underline{r}_\eta$ is computed by solving

$$\underline{r}_\xi \cdot \underline{r}_\eta = x_\xi x_\eta + y_\xi y_\eta = 0$$

and

$$x_\eta^2 + y_\eta^2 = g_{22} \tag{10}$$

where $g_{22}$ is obtained (and kept fixed) from the initial algebraic grid. Notice that ($g_{22}$) at $i = \eta_{\min}$ represents the distance off of the $\eta = \eta_{\min}$ at the $i$th grid point as it is used in the analysis presented by Thompson [3].

c. $\underline{r}_{\xi\eta}$ is computed by solving

$$(\underline{r}_\xi \cdot \underline{r}_\eta)_\xi = (x_\xi x_\eta + y_\xi y_\eta)_\xi = 0 \tag{11}$$

and

$$(\underline{r}_\eta \cdot \underline{r}_\eta)_\xi = \left(x_\eta^2 + y_\eta^2\right)_\xi = (g_{22})_\xi$$

where $(g_{22})_\xi$ represents the change in $g_{22}$ in $\xi$ direction [$(g_{22})_\xi$ are computed by numerically differentiating $g_{22}$ from an algebraic grid and are kept fixed throughout the refinement process].

The resulting system is:

$$\begin{bmatrix} x_\xi & y_\xi \\ x_\eta & y_\eta \end{bmatrix} \begin{bmatrix} x_{\xi\eta} \\ y_{\xi\eta} \end{bmatrix} = \begin{bmatrix} -x_{\xi\xi} x_\eta - y_{\xi\xi} y_\eta \\ \dfrac{(g_{22})_\xi}{2} \end{bmatrix}$$

Note that $(x_\xi y_\eta - x_\eta y_\xi)$ is the jacobian of transformation and hence is nonzero. $\underline{r}_{\eta\eta}$ is computed by solving

$$(\underline{r}_\xi \cdot \underline{r}_\eta)_\eta = (x_\xi x_\eta + y_\xi y_\eta)_\eta = 0 \tag{12}$$

$$(\underline{r}_\eta \cdot \underline{r}_\eta)_\eta = \left(x_\eta^2 + y_\eta^2\right)_\eta = (g_{22})_\eta$$

where $(g_{22})_\eta$ represents the change in $g_{22}$ in $\eta$ direction ($(g_{22})_\eta$ are computed by numerically differentiating $g_{22}$ from an algebraic grid and are kept fixed throughout the refinement process).

d. $\phi$ is then computed by using formula (2).

Similarly, $\psi$ is computed by using formula (3) and by evaluating: (i) $\underline{r}_\eta$ and $\underline{r}_{\eta\eta}$ using appropriate finite difference at every grid point, (ii) $\underline{r}_\xi$ by solving equation (4) and

$$\underline{r}_\xi \cdot \underline{r}_\xi = x_\xi^2 + y_\xi^2 = g_{11} \tag{13}$$

$$\text{for } x_\xi \text{ and } y_\xi.$$

where $g_{11}$ is evaluated (and kept fixed) from the initial algebraic grid, (iii) $\underline{r}_{\xi\eta}$

by solving equation (8) and

$$(\underline{r}_\xi \cdot \underline{r}_\xi)_\eta = \left(x_\xi^2 + y_\xi^2\right)_\eta = (g_{11})_\eta \tag{14}$$

for $x_{\xi\eta}$ and $y_{\xi\eta}$

and (iv) $\underline{r}_{\xi\xi}$ by solving equation (6) and

$$(\underline{r}_\xi \cdot \underline{r}_\xi)_\xi = \left(x_\xi^2 + y_\xi^2\right)_\xi = (g_{11})_\xi \tag{15}$$

for $x_{\xi\xi}$ and $y_{\xi\xi}$.

Both these algorithms were coded in the general purpose computer code GENIE, developed by the author [4–6]. The options to evaluate these forcing functions either from boundaries (and then interpolated for field values) or from entire grids are allowed. Computational examples follow to illustrate the success of these methodologies.

Note that one other attractive feature of this algorithm is its application to adaptive gridding. In this case, applying the equidistribution law [14] with proper weight functions $v, v_\xi, v_\eta$ (desired cell areas and their changes in $\xi$ and $\eta$ directions) and/or $g_{11}, (g_{11})_\xi, g_{22}, (g_{22})_\xi, (g_{22})_\eta$ (desired cell grid spacings in both directions and their respective changes in both directions) can be evaluated. These quantities are then utilized in the aforementioned algorithms to adapt grid points. Naturally, desired cell area approach should be suitable for finite volume techniques while desired grid spacing approach seems appropriate for finite difference techniques.

## 4. COMPUTATIONAL EXAMPLES

Three distinct examples demonstrating the application of these control functions are presented. The first example, involving an H-type grid in a circular region, is presented in Figures 1 and 2. An algebraic grid is demonstrated in Figure 1a. Figures 1b and 1c display the application of algorithm 1 (cell area approach) and algorithm 2 (grid spacing approach). Even though Figures 1b and 1c are seemingly similar, it can be observed in Figures 2a to 2c that the cell area approach results in a better (in view of aspect ratio) grid near the special point region. Also Figures 2a to 2c demonstrate that the elliptic grid has well-defined areas and the skewness problem that appeared in one algebraic grid has been washed out. In both

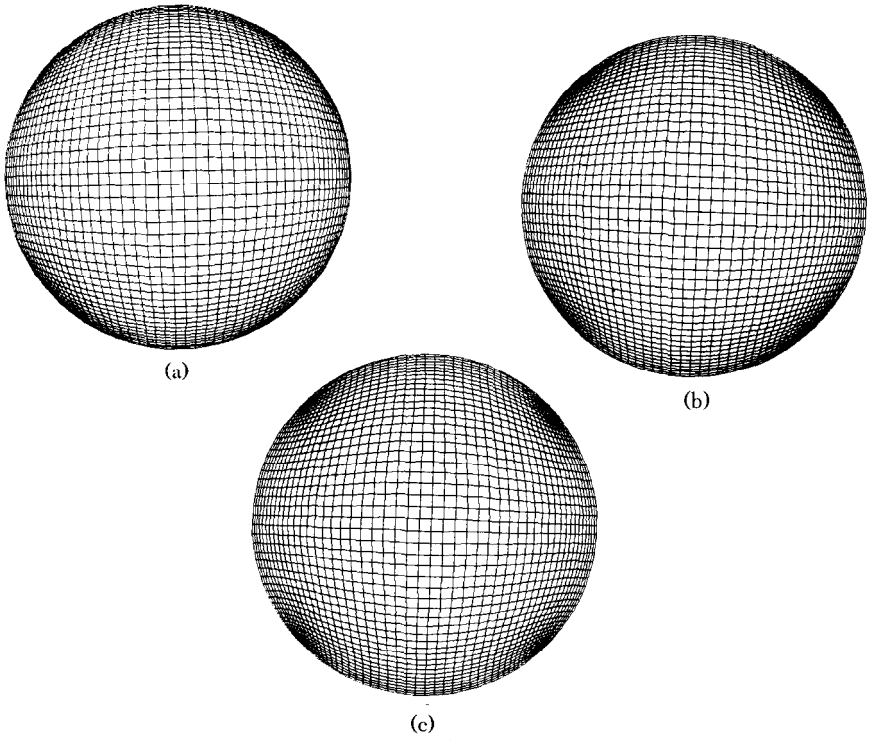



FIG. 1. H-type grid in a circular region. (a) Algebraic grid. (b) Elliptic grid; grid spacing approach. (c) Elliptic grid; cell area approach.

cases, the elliptic solver was applied for seven iterations. Our goal is not to drive the elliptic system to convergence for perfect orthogonality but to obtain a balanced (orthogonality–smoothness–distribution) grid quickly and efficiently.

The second example demonstrating the influence of the control functions in a configuration involving smooth concave and convex regions is presented in Figure 3. Figures 3a and 3b demonstrate an algebraic and elliptic grids. The behavior of grid lines in concave–convex regions and nose regions is demonstrated in Figures 3c to 3f for both algebraic and elliptic grids. The third example, involving sharp concave–convex regions, is presented in the Figure 4a to 4d. The elliptic grids show improvements in smoothness and orthogonality by keeping the distribution of grid points very close to that of the algebraic grids.

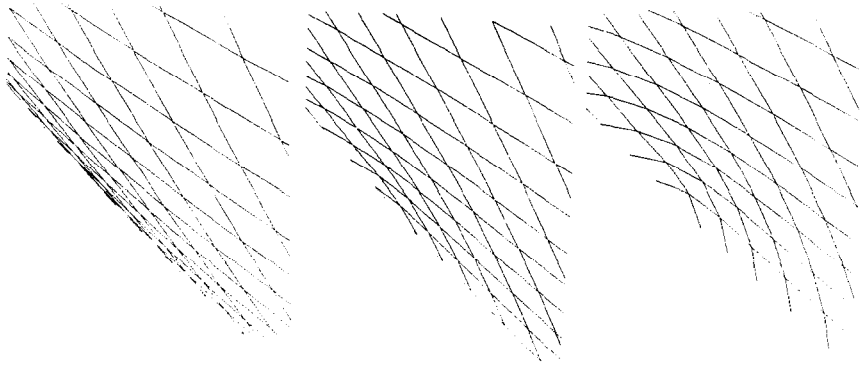



FIG. 2. Close up of grid lines; special point region. (a) Algebraic grid. (b) Elliptic grid; grid spacing approach. (c) Elliptic grid; cell area approach.

## 5. FUTURE WORK

Computational examples demonstrate the success of the presented algorithm. Grids resulting from the cell area approach on the entire algebraic grid show very good control of distribution compared to the boundary-interpolated method. The extension of this methodology to three-dimensional configurations is straight forward. For example, in case of the evaluation of the forcing function $\phi$ utilizing the cell volume approach, the following sets of equations should be solved:

$$r_\xi \cdot r_\eta = 0$$

$$r_\xi \cdot r_\zeta = 0$$

$$r_\xi \cdot (r_\eta x r_\zeta) = v \tag{16}$$

$$(r_\xi \cdot r_\eta)_\eta = 0$$

$$(r_\xi \cdot r_\zeta)_\eta = 0$$

$$(r_\xi \cdot (r_\eta x r_\zeta))_\eta = v_\eta \tag{17}$$

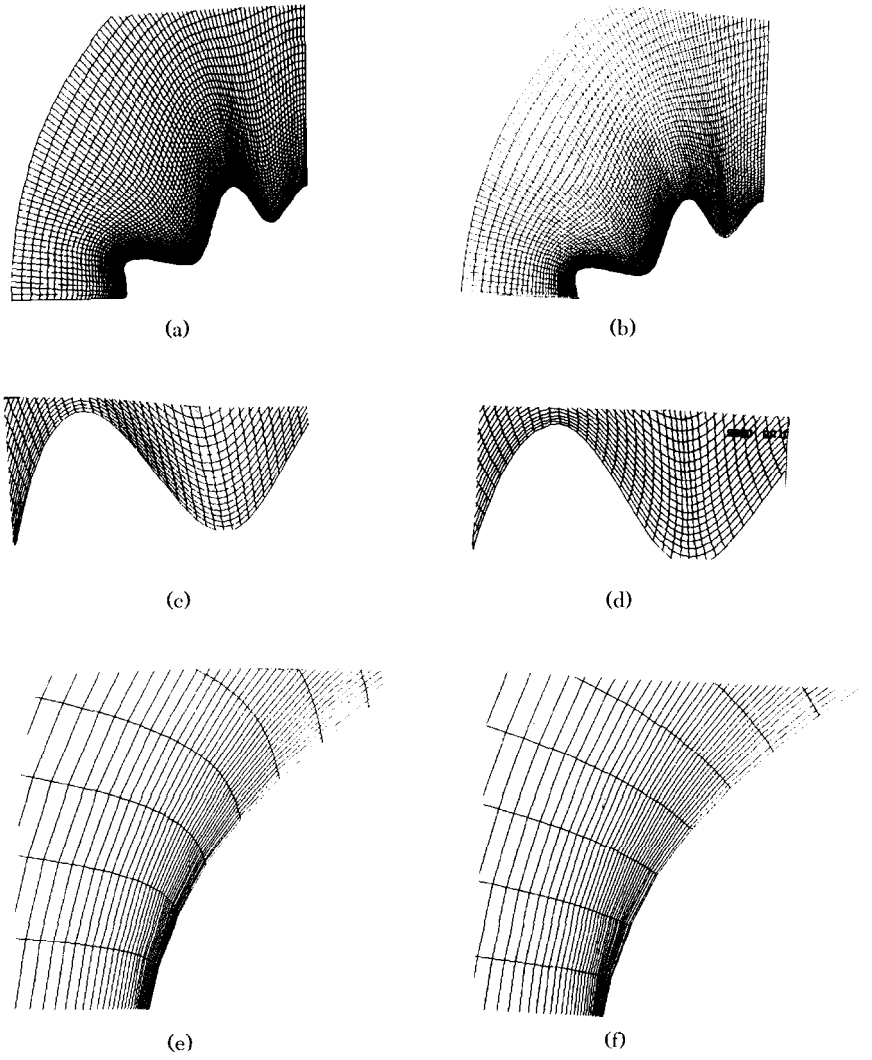
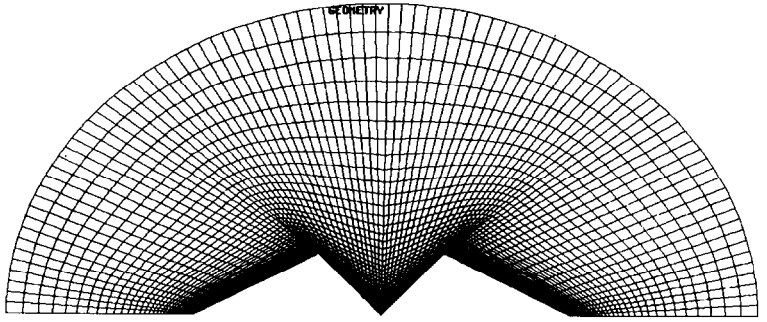
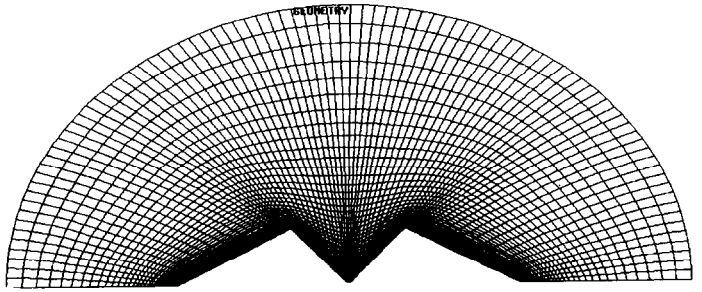


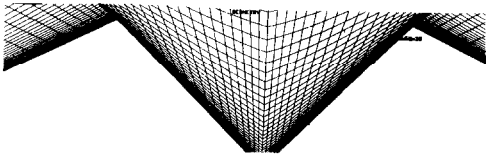

FIG. 3. Arbitrary configuration, smooth concave–convex regions. (a) Algebraic grid. (b) Elliptic grid. (c) Algebraic grid, region A. (d) Elliptic grid, region A. (e) Algebraic grid, nose region. (f) Elliptic grid, nose region.



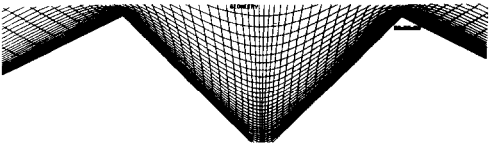
(a)



(b)



(c)



(d)

FIG. 4. Configuration with sharp corners. (a) Algebraic grid. (b) Elliptic grid. (c) Grid line behavior near sharp corners; algebraic grid. (d) Grid line behavior near sharp corners; elliptic grid.

$$(r_\xi \cdot r_\eta)_\zeta = 0$$

$$(r_\xi \cdot r_\zeta)_\zeta = 0$$

$$\left(r_\xi \cdot (r_\eta x r_\zeta)\right)_\zeta = v_\zeta \tag{18}$$

The work on three-dimensional application of this methodology with computational examples will be reported in the near future. This methodology is being applied to grid adaptation in two dimensions. The preliminary results are presented in [15].

## REFERENCES


1 J. F. Thompson, *Program EAGLE: Numerical Grid Generation System User's Manual, Vols. II and III*, AFATL-TR-87-15, 1987.

2 J. F. Thompson and B. Gatlin, Easy EAGLE: An Introduction to the EAGLE Grid Code, Mississippi State University, 1988.

3 J. F. Thompson, A general three-dimensional elliptic grid generation system on a composite block structure, *Comput. Methods Appl. Mech. Engrg.* 377 (1987).

4 B. K. Soni, GENIE: GENeration of computational geometry-grids for internal-external flow configurations, in *Proceedings of the Numerical Grid Generation in Computational Fluid Mechanics '88*, Miami, 1988.

5 B. K. Soni, M. D. McClure, and C. W. Mastin, Geometry and Generation in N + 1 Easy Steps, presented at the First International Conference on Numerical Grid Generation in Computational Fluid Dynamics, Landshut, Germany, 1986.

6 B. K. Soni, *Two and Three Dimensional Grid Generation for Internal Flow Applications of Computational Fluid Dynamics*, AIAA-1526-85.

7 J. P. Steinbrenner, J. R. Chawner, and C. L. Fouts, *The GRIDGEN 3D Multiple Block Grid Generation System, Vols. I and II*, WRDC-TR-90-3022, United States Air Force, Wright Research & Development Center, Dayton, OH, July 1990.

8 R. L. Sorenson, Three-dimensional zonal grids about arbitrary shapes by Poisson's equation, in *Numerical Grid Generation for Computational Fluid Mechanics '88* Pineridge press, 1988.

9 P. R. Eiseman, Grid generation for fluid mechanics computations, *Annual Rev. Fluid Mechanics* (1985).

10 J. L. Steger and D. S. Chaussee, Generation of body-fitted coordinates using hyperbolic partial differential equations, *SIAM J. Sci. Statist. Comput.* 431 (1980).

11 B. K. Soni and Ming-Hsin Shih, TIGER: turbomachinery interactive grid GENeration, in *Proceedings of the Third International Conference of Numerical Grid Generation in CFD*, Barcelona, June 1991.

12 B. K. Soni, Grid generation for internal flow configurations, *J. Comput. Appl.*, to appear.

13 A. Middlecoff and P. D. Thomas, *Direct Control of the Grid Point Distribution in Meshes Generated by Elliptic Equations for Solution of Navier-Stokes Nozzle Flow*, AIAA-79-1462.

14 D. A. Anderson and J. P. Steinbrenner, *Generating Adaptive Grids with a Conventional Grid Scheme*, AIAA-86-0427.

15. N. P. Weatherill and B. K. Soni, *Grid Adaptation and Refinement in Structured and Unstructured Algorithms*, in *Proceedings of the Third International Conference of Numerical Grid Generation in CFD*, Barcelona, 1991.