JOURNAL OF COMPUTATIONAL PHYSICS 96, 451-469 (1991)

A Numerical Technique for Two-Dimensional Grid Generation with Grid Control at All of the Boundaries

K. HSU AND S. L. LEE

Department of Power Mechanical Engineering, National Tsing-Hua University, Hsinchu Taiwan 30043, Republic of China

Received April 26, 1989; revised April 13, 1990

A numerical technique is developed in the present investigation to generate grids by the use of the Poisson equations. Orthogonal grids are obtained along all of the two boundaries $\eta = 0$ and $\eta = \eta_{max}$. The "stand-off" grid spacing between $\xi = 0$ and $\xi = \Delta\xi$ and between $\xi = \xi_{max} - \Delta\xi$ and $\xi = \xi_{max}$ can be controlled by employing a proper grid point distribution on the boundaries $\eta = 0$ and $\eta = \eta_{max}$. Thanks to the orthogonal boundary grids, the present numerical technique is applicable to complex geometry by patching grids without slope discontinuity across the interface of the patches. This technique also allows the Poisson equations to generate coordinates for O-type grid system and for periodic turbine cascades. In the course of grid generation, the magnitudes of the required control functions might be very large in a region where clustering grids are needed. To guarantee a good numerical stability in spite of the values of the control functions, the weighting function scheme along with the SIS solver is employed. Through the examples illustrated in the present study, the negative Jacobian reported by previous investigators is shown not to arise from the use of the Poisson equations. It, indeed, comes from the truncation error of the central difference scheme used by them.

INTRODUCTION

Solving a partial differential equation defined on an arbitrarily shaped domain is encountered in many physical problems such as fluid mechanics, heat transfer, structures, electromagnetics, and all other areas involving field solutions. To obtain an accurate solution, the irregular physical domain is commonly transformed into a rectangular one by employing a numerical grid generation technique. One of the highly developed techniques for generating a boundary-fitted coordinate system is to let the curvilinear coordinates (ξ, η) be the solution of a system of elliptic differential equations in the physical plane. Although systems of parabolic or hyperbolic differential equations can also be used, only the system of elliptic differential equations allows specified boundary grid distribution along all of the boundaries.

In 1962, Crowley [1] suggested the solution of the Laplace equations $\nabla^2\xi = 0$ and $\nabla^2\eta = 0$ be used as the curvilinear coordinates. However, the earliest successful development of the Laplace grid generation method was formally reported by Winslow [2], since the system had to be reformulated by an interchange of

0021-9991/91 $3.00

dependent and independent variables. The next major step in the development of elliptic methods was given by Thompson *et al.* [3]. In their study, Thompson *et al.* [3] extended the Laplace method to problems having multiply-connected domain. In addition, they proposed the Poisson method by adding a pair of control functions to the right-hand sides of the Laplace equations, i.e., $\nabla^2\xi = P$ and $\nabla^2\eta = Q$. The grid distribution thus can be improved by the assigned values of P and Q. The effect of the control functions on the curvilinear coordinates system has been extensively studied by Thompson *et al.* [4]. Generally speaking, for a boundary with a constant η-value, the control function P changes the intersecting angles of the constant ξ curves at that boundary while Q alters the spacing of the constant η curves. Based on such a finding, Thompson *et al.* [5] developed the TOMCAT code by employing control functions of exponential form. Another form for the control functions was proposed by Middlecoff and Thomas [6].

The major difficulty in the use of Poisson grid generation method is the choice of the control functions. Hodge *et al.* [7] simplified the exponential form of the control functions proposed by Thompson *et al.* [5] and proposed a procedure for automating the choice of the amplitudes of the exponential control functions. In the development of their GRAPE code, Steger and Sorenson [8–10] back-solved the values of the control functions from the governing equations for the two boundaries $\eta = 0$ and $\eta = \eta_{\max}$. The control functions inside the domain then were interpolated with exponential functions as suggested by Thompson *et al.* [5]. Such a procedure gives the advantage that the desired intersecting angles and the "stand-off" spacing at $\eta = 0$ and $\eta = \eta_{\max}$ can be specified.

It should be noted here that, although both grid size and grid skewness can be controlled at the boundaries of $\eta = 0$ and $\eta = \eta_{\max}$, some disadvantages arise in the use of the GRAPE code [9, 10]. First, the control functions inside the domain are defined by the sum of two branches of exponential functions with one arising from $\eta = 0$ and the other from $\eta = \eta_{\max}$. For convenience, the former is called branch A and the latter branch B. For cases having small $\eta_{\max}$ values, branch A does not vanish at $\eta = \eta_{\max}$ such that grid control at the boundary of $\eta = \eta_{\max}$ becomes very difficult. Similarly, the control functions at $\eta = 0$ will pose an interference from the branch B. As a result, the boundary grids cannot be properly controlled unless the value of $\eta_{\max}$ is siffuciently large. For instance, when applied to a periodic turbine cascade, the aspect ratio (pitch to blade length) must be as large as 5 (see Figs. 9 of [9]) in the use of the GRAPE code. However, the aspect ratio in turbine cascade is 0.7, in general. Second, grids cannot be controlled at the other pair of boundaries $\xi = 0$ and $\xi = \xi_{\max}$. The slope continuity thus is not guaranteed at the branch cut in generating O-type grids. Finally, the numerical scheme employed in the GRAPE code poses serious numerical difficulties in solving the Poisson grid generation equations when the control functions have large magnitudes. This point has been remarked by Steger and Sorenson [8] and Sorenson [9].

The purpose of the present investigation is to propose a numerical technique to generate two-dimensional grids with grid control along all of the boundaries. The control functions inside the domain are interpolated from the desired boundary

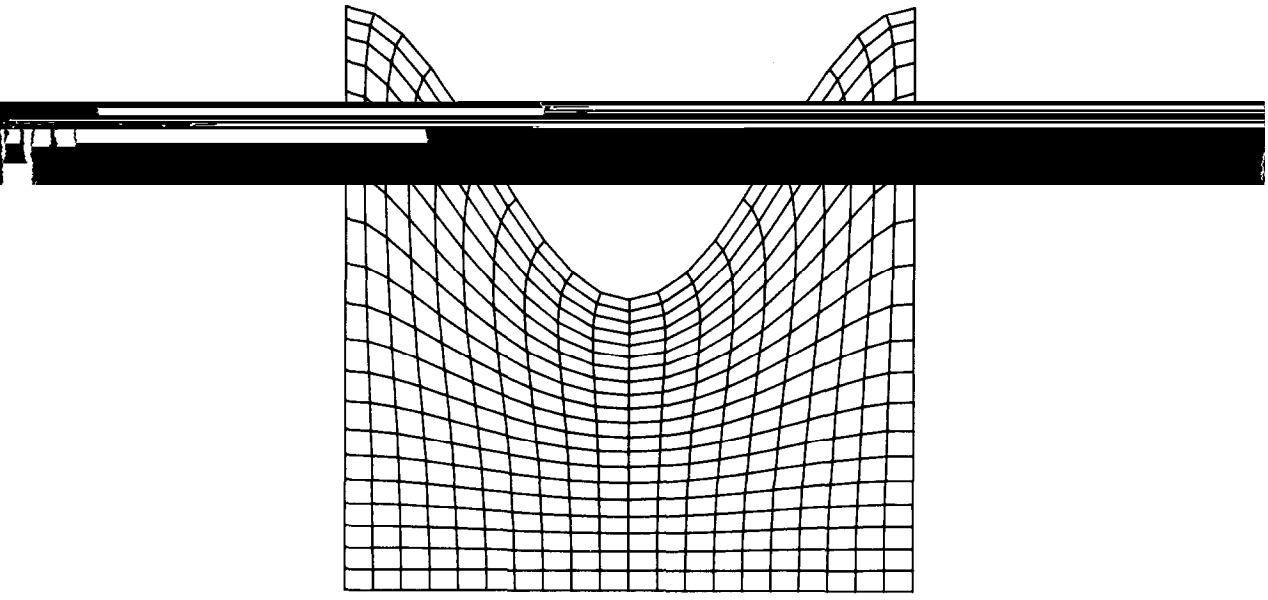

FIG. 3. Grid system generated by the present technique.

shows the resulting grid system. As expected, the weighting function scheme does not produce grids with negative Jacobian and thus does not reproduce the grids shown in Fig. 1. This seems to substantiate the point that the negative Jacobian as can be seen from Fig. 1 comes from the truncation error of the central difference scheme. To examine the performance of the present grid generation technique, Coleman's problem [16] is solved by the algorithm presented in the previous section. The resulting grid system is shown in Fig. 3. From Fig. 3, one sees that the grid system has a smoothly varying grid size over the entire domain with orthogonal grids at the four boundaries. No negative Jacobian is found even though the cavity in Fig. 3 is deeper than that in Fig. 1.

Large control functions occur also in an adaptive grid system due to the need of clustering grids at some particular locations. In testing the performance of their adaptive method, Matsuno and Dwyer [15] employed a physical quantity simulated by $f(x; \mathbf{a})$ that has a sharp but continuous increase at $x = 0.5$. The variation of the function $f(x; \mathbf{a})$ in a narrow region containing the point $x = 0.5$ is specified by the the parameter $\mathbf{a}$. As $\mathbf{a}$ assumes the value of infinity, the function $f(x; \infty)$ becomes a unit step function at $x = 0.5$. For a given value of $\mathbf{a}$, Matsuno and Dwyer [15] generated a grid system by using their adaptive method for the particular f-function. Both first and second derivatives of the function (i.e., f_x and f_{xx}) were then evaluated numerically from the exact value of the f-function defined at the grid points. The numerical results of f_x and f_{xx} can be expected to have poor accuracies if an improper grid system has been provided by the adaptive

particularly true when the heat transfer coefficient in heat transfer problems (or friction factor in fluid mechanics problems) is evaluated. Hence, orthogonal grids are usually desired along the boundaries. Based on such a requirement, the values of the control functions $P(\xi, 0)$ and $Q(\xi, 0)$ at the boundary $\eta = 0$ are back-solved from Eqs. (2) and (3) as

$$\begin{aligned} P(\xi, 0) &= (Bx_\eta - Ay_\eta)/J^3 \\ Q(\xi, 0) &= (Ay_\xi - Bx_\xi)/J^3 \\ A &= \alpha x_{\xi\xi} + \gamma x_{\eta\eta} \\ B &= \alpha y_{\xi\xi} + \gamma y_{\eta\eta}, \end{aligned} \tag{5}$$

where the β-value has been assigned zero. In Eqs. (5), the values of x_ξ, y_ξ, $x_{\xi\xi}$, and $y_{\xi\xi}$ can be evaluated from $x(\xi, 0)$ and $y(\xi, 0)$, whereas the values of x_η and y_η are determined by

$$\begin{aligned} x_\eta &= -s_\eta y_\xi (x_\xi^2 + y_\xi^2)^{-1/2} \\ y_\eta &= s_\eta x_\xi (x_\xi^2 + y_\xi^2)^{-1/2} \\ s_\eta &= \Delta s(\xi)/\Delta\eta, \end{aligned} \tag{6}$$

where $\Delta s(\xi)$ is the desired "stand-off" spacing between the two curves $\eta = 0$ and $\eta = \Delta\eta$. The values of $x_{\eta\eta}$ and $y_{\eta\eta}$ will be guessed or estimated from the previous solution of $x(\xi, \eta)$ and $y(\xi, \eta)$ by the second-order fintie difference [9]

$$\begin{aligned} x_{\eta\eta}(\xi, 0) &= (-3.5x_1 + 4x_2 - 0.5x_3)/\Delta\eta^2 - 3(x_\eta)_1/\Delta\eta \\ y_{\eta\eta}(\xi, 0) &= (-3.5y_1 + 4y_2 - 0.5y_3)/\Delta\eta^2 - 3(y_\eta)_1/\Delta\eta, \end{aligned} \tag{7}$$

where the subscripts 1, 2, and 3 stand for, respectively, the positions $\eta = 0$, $\Delta\eta$, and $2\Delta\eta$. The values of the control functions $P(\xi, \eta_{\max})$ and $Q(\xi, \eta_{\max})$ at the boundary $\eta = \eta_{\max}$ can be determined in a similar manner.

Once the control functions at the two boundaries $\eta = 0$ and $\eta = \eta_{\max}$ are available, the distribution of the control functions inside the computational domain is interpolated by using power-law functions as

$$\begin{aligned} P(\xi, \eta) &= P(\xi, 0)[1 - (\eta/\eta_{\max})]^a \\ &\quad + P(\xi, \eta_{\max})(\eta/\eta_{\max})^b \end{aligned} \tag{8}$$

$$\begin{aligned} Q(\xi, \eta) &= Q(\xi, 0)[1 - (\eta/\eta_{\max})]^c \\ &\quad + Q(\xi, \eta_{\max})(\eta/\eta_{\max})^d, \end{aligned} \tag{9}$$

where the powers a, b, c, and d are used to specify the variations of the control functions inside the domain. The values of a, b, c, and d will be assigned by the user

in the range of 2 to 5, in general. However, they do not necessarily have the same values. As mentioned earlier, the present control functions (8) and (9) reduce, respectively, to $P(\xi, 0)$ and $Q(\xi, 0)$ as $\eta = 0$ and to $P(\xi, \eta_{\max})$ and $Q(\xi, \eta_{\max})$ as $\eta = \eta_{\max}$. Thus, unlike the exponential functions employed by Steger and Sorenson [8–10], the present power-law functions (8) and (9) control both grid size and grid skewness at $\eta = 0$ and $\eta = \eta_{\max}$ very well, despite the decaying powers and the $\eta_{\max}$ value.

It is noted that the control functions along the boundary $\xi = 0$, i.e., $P(0, \eta)$ and $Q(0, \eta)$, have been defined by Eqs. (8) and (9). Thus, no orthogonal grids are guaranteed at the boundary $\xi = 0$, if fixed grid point distribution is specified there. In the present investigation, the grid points are allowed to move along the curve of $\xi = 0$ until orthogonal grids are obtained. To achieve this, the slope of the curve $\xi = 0$ is expressed by

$$e = y_\eta / x_\eta = -x_\xi / y_\xi. \tag{10}$$

For convenience of computation, the boundary $\xi = 0$ is divided into sections that have either $|e| \leqslant 1$ or $|e| > 1$. The boundary conditions at $\xi = 0$ thus can be defined by

$$\begin{aligned} x_\xi &= -e y_\xi, \qquad y = y^*(t) \qquad \text{if} \quad |e| \leqslant 1 \\ x &= x^*(t), \qquad y_\xi = -x_\xi / e \qquad \text{if} \quad |e| > 1, \end{aligned} \tag{11}$$

where the curve of $\xi = 0$ is represented by $x^*(t)$ and $y^*(t)$, $0 \leqslant t \leqslant 1$, with $x^*(0) = x(0, 0)$, $y^*(0) = y(0, 0)$, $x^*(1) = x(0, \eta_{\max})$, and $y^*(1) = y(0, \eta_{\max})$. For the sections having $|e| \leqslant 1$, the values of t and y_ξ are evaluated, respectively, from $x(0, \eta)$ and $y(\xi, \eta)$ obtained in the previous iteration. A similar procedure can be applied on the sections of $|e| > 1$.

The boundary conditions (11) can be greatly simplified when $\xi = 0$ is a straight line ($e = \text{const}$). For an inclined straight line, the boundary conditions become

$$\begin{aligned} x_\xi &= -e y_\xi, \qquad y = y_0 + e(x - x_0) \qquad \text{if} \quad |e| \leqslant 1 \\ x &= x_0 + (y - y_0)/e, \qquad y_\xi = -x_\xi / e \qquad \text{if} \quad |e| > 1, \end{aligned} \tag{12}$$

where $x_0 = x(0, 0)$ and $y_0 = y(0, 0)$. In generating O-type grids, it is convenient to define a branch cut in the horizontal position ($e = 0$). The boundary conditions thus reduce to

$$x_\xi = 0 \qquad \text{and} \qquad y = y_0. \tag{13}$$

For the case of a vertical boundary line ($|e| = \infty$), as encountered in many C-type grids, the boundary conditions are

$$x = x_0 \qquad \text{and} \qquad y_\xi = 0. \tag{14}$$

The boundary conditions for $\xi = \xi_{\max}$ can be defined similarly. It should be noted that, although the spacings between $\xi = 0$ and $\xi = \Delta\xi$ and between $\xi = \xi_{\max} - \Delta\xi$ and $\xi = \xi_{\max}$ are not specified in the present method, they can be controlled by the grid distributions at the boundaries $\eta = 0$ and $\eta = \eta_{\max}$.

In this section, a mesh with orthogonal boundary grids has been emphasized. However, with a minor modification, the present technique is equally applicable to cases that require a particular grid skewness on the boundaries. This can be easily accomplished by employing the variables

$$\begin{aligned} A &= \alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} \\ B &= \alpha y_{\xi\xi} + 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} \end{aligned} \tag{15}$$

$$\begin{aligned} x_\eta &= s_\eta (x_\xi \cos\theta - y_\xi \sin\theta)(x_\xi^2 + y_\xi^2)^{-1/2} \\ y_\eta &= s_\eta (y_\xi \cos\theta + x_\xi \sin\theta)(x_\xi^2 + y_\xi^2)^{-1/2}, \end{aligned} \tag{16}$$

instead of their counterparts in Eqs. (5) and (6), where θ is the desired intersection angle between the curves of $\eta = 0$ and $\xi = \text{const}$. Similarly, a desired grid skewness can be imposed on the boundaries $\xi = 0$ and $\xi_{\max}$ by using a Neumann boundary condition. It is noted that Eqs. (16) are derived from

$$\nabla\xi \cdot \nabla\eta = |\nabla\xi|\,|\nabla\eta| \cos(\pi - \theta). \tag{17}$$

This same geometric property, which is presented in Refs. [8, 9], is incorrect.

Numerical Solution Methods

Equations (2) and (3), the boundary conditions (11), and the given functions $x(\xi, 0)$, $y(\xi, 0)$, $x(\xi, \eta_{\max})$, and $y(\xi, \eta_{\max})$ constitute a system of elliptic differential equations in the rectangular region $0 \leqslant \xi \leqslant \xi_{\max}$ and $0 \leqslant \eta \leqslant \eta_{\max}$. To obtain an accurate result with a good numerical stability in spite of the values of J^2P/α and J^2Q/γ, the weighting function scheme [11] is employed to discretize Eqs. (2) and (3). After applying the weighting function scheme [11] to Eq. (2) on a uniform grid system $\Delta\xi = \Delta\eta = 1$, one obtains

$$\begin{aligned} &a_{\mathrm{SW}} x_{i-1,j-1} + a_{\mathrm{W}} x_{i-1,j} + a_{\mathrm{NW}} x_{i-1,j+1} \\ &\quad + a_{\mathrm{S}} x_{i,j-1} + a_{\mathrm{P}} x_{i,j} + a_{\mathrm{N}} x_{i,j+1} \\ &\quad + a_{\mathrm{SE}} x_{i+1,j-1} + a_{\mathrm{E}} x_{i+1,j} + a_{\mathrm{NE}} x_{i+1,j+1} = 0 \end{aligned} \tag{18}$$

$$\begin{aligned} a_{\mathrm{W}} &= \alpha w_f(-Z_1), \qquad a_{\mathrm{E}} = \alpha w_f(Z_1) \\ a_{\mathrm{S}} &= \gamma w_f(-Z_2), \qquad a_{\mathrm{N}} = \gamma w_f(Z_2) \\ a_{\mathrm{SW}} &= -2\beta w_f^*(Z_1)\, w_f^*(Z_2) \end{aligned} \tag{19}$$

$$
\begin{aligned}
&a_{NE} = a_{SW}, \qquad a_{NW} = -a_{SW}, \qquad a_{SE} = a_{NW} \\
&a_P = -a_W - a_E - a_S - a_N \\
&Z_1 = J^2 P/\alpha, \qquad Z_2 = J^2 Q/\gamma \\
&w_f(Z) = Z/(1-e^{-Z}), \qquad w_f^*(Z) = Z/(e^Z - e^{-Z}),
\end{aligned} \tag{20}
$$

where the subscripts i, j denote a quantity at the point (ξ_i, η_j) with ξ_i and η_j being $\xi_i = i-1$ and $\eta_j = j-1$. Note that the cross derivative term $x_{\xi\eta}$ in Eq. (2) has been approximated by

$$
x_{\xi\eta} = (x_{i-1,j-1} + x_{i+1,j+1} - x_{i-1,j+1} - x_{i+1,j-1})\, w_f^*(Z_1)\, w_f^*(Z_2), \tag{21}
$$

based on an interpolation theory described in Appendix A. A similar algebraic equation can be obtained for Eq. (3).

In Eqs. (19), all of the variable quantities, α, β, γ, J, P, Q are evaluated at the location (ξ_i, η_j). To conserve the CPU time, the weighting functions $w_f(Z)$ and $w_f^*(Z)$ are evaluated, respectively, from the approximations

$$
w_f(Z) = [0, (1 - 0.1\,|Z|)^5] + [0, Z] \tag{22}
$$

$$
w_f^*(Z) = (2 + 0.3332Z^2 + 0.0172Z^4)^{-1}, \tag{23}
$$

where $[a, b]$ stands for the greater of a and b. The accuracy of the approximation (22) for the weighting function $w_f(Z)$ has been discussed by Lee *et al.* [13]. The exact value of the weighting function $w_f^*(Z)$ along with its approximation (23) is presented in Table I for comparison. From Table I, it can be seen that the weighting function $w_f^*(Z)$ reaches it maximum value $w_f^*(0) = 0.5$ as $Z = 0$ and becomes zero when $|Z| = \infty$. In conventional studies, the central difference scheme is used to formulate the cross derivatives $x_{\xi\eta}$ and $y_{\xi\eta}$. This implies that $w_f^*(Z)$ is treated as a constant function having the value 0.5. Therefore, the contribution of the cross derivative terms are overestimated in conventional studies.

In the present investigation, the system of algebraic equations is solved by using the strongly-implicit solver (SIS solver) proposed by Lee [12]. Due to its strong implicitness, the SIS solver allows a poor initial guess for the solution such that it shows good performance in solving the algebraic equations (18). For convenience, the algorithm of the present grid generation technique is summarized as follows:

1. Input the grid point distribution $x(\xi, 0)$, $y(\xi, 0)$, $x(\xi, \eta_{max})$, $y(\xi, \eta_{max})$ and the desired "stand-off" spacing $\Delta s(\xi)$ at $\eta = 0$ and $\eta = \eta_{max}$.

2. Determine the values of x_ξ, y_ξ, $x_{\xi\xi}$, and $y_{\xi\xi}$ along the boundaries $\eta = 0$ and $\eta = \eta_{max}$ by using the cubic spline technique [14].

3. Assign an initial guess for the solution $x(\xi, \eta)$ and $y(\xi, \eta)$ simply by using a linear interpolation based on the boundary values defined at $\eta = 0$ and $\eta = \eta_{max}$.

4. Evaluate x_ξ, y_ξ, x_η, and y_η from the guessed x and y by the use of the cubic spline technique for the entire computational domain.

TABLE I

Value of the Weighting Function $w_f^*(Z)$

Z	Exact function[a]	Approximation[b]
−20.0	0.00000	0.00035
−10.0	0.00045	0.00482
−5.0	0.03369	0.04744
−2.0	0.27572	0.27716
−1.0	0.42546	0.42546
−0.8	0.45040	0.45039
−0.6	0.47121	0.47121
−0.5	0.47976	0.47976
−0.4	0.48691	0.48691
−0.2	0.49668	0.49668
−0.1	0.49917	0.49917
0.0	0.50000	0.50000
0.1	0.49917	0.49917
0.2	0.49668	0.49668
0.4	0.48691	0.48691
0.5	0.47976	0.47976
0.6	0.47121	0.47121
0.8	0.45040	0.45039
1.0	0.42546	0.42546
2.0	0.27572	0.27716
5.0	0.03369	0.04744
10.0	0.00045	0.00482
20.0	0.00000	0.00035

[a] Exact function: $w_f(Z) = Z/(e^Z - e^{-Z})$.
[b] Approximation: $w_f^*(Z) = (2 + 0.3332Z^2 + 0.0172Z^4)^{-1}$.

5. Estimate $x_{\eta\eta}$ and $y_{\eta\eta}$ from Eqs. (7) for the boundaries $\eta = 0$ and $\eta = \eta_{\max}$.

6. Evaluate the $P(\xi, 0)$, $Q(\xi, 0)$, $P(\xi, \eta_{\max})$, and $Q(\xi, \eta_{\max})$ values from Eqs. (5), and then determine $P(\xi, \eta)$ and $Q(\xi, \eta)$ from Eqs. (8) and (9).

7. Perform one SIS iteration on Eqs. (18)–(20) to yield $x(\xi, \eta)$ and $y(\xi, \eta)$. If the solution from the previous iteration is still very poor, performing two or more SIS iterations with a SOR value of 1.0 to 1.3 could help convergence significantly.

8. If the renewed x and y values are consistent with the previous values within a prescribed tolerance then stop. Otherwise, modify the solution x and y with a SOR factor and return to step 4. The optimum SOR value needed by the SIS solver [12] for the present grid generation technique is in the range of 0.1 to 0.3.

Performance of the New Technique

In conventional grid generation procedures, the central difference scheme is employed to discretize the governing equations (2) and (3). However, numerical

instability could arise in the use of a central difference scheme when J^2P/α and J^2Q/γ have large absolute values, as reported by Steger and Sorenson [8, 9] and Matsuno and Dwyer [15]. In a workshop sponsored by NASA, Coleman [16] proposed a grid generation technique that generates orthogonal grids. However, his method produced grids with negative Jacobian for a square region with a sine-shaped cavity on the top. For convenience, this particular grid system is presented in Fig. 1. With the $x(\xi, \eta)$ and $y(\xi, \eta)$ values shown in Fig. 1, Coleman back-solved the control functions $P(\xi, \eta)$ and $Q(\xi, \eta)$ from Eqs. (2) and (3). Not surprisingly, the same grid system (Fig. 1) will be regenerated if Eqs. (2) and (3) are solved by using this set of control functions. Based on this, Coleman concluded that the Poisson system could produce a mesh with crossing lines (negative Jacobian). Recently, this same conclusion was once again drawn by Matsuno and Dwyer [15] in a study on adaptive grids. A similar finding was also remarked by Thompson *et al.* [17] in the section of *Poisson system* that states "The extremum principles may be weakened or lost completely with such a system." However, it must be pointed out here that the Poission equations (1) indeed are heat conduction problems with heat generation P and Q, if the dependent variables ξ and η are regarded as temperatures. Thus, no crossing point exists between two isotherms ($\eta = c_1$ and $\eta = c_2$) despite the values of the heat generation P and Q. The conclusion drawn by Matsuno and Dwyer [15], Coleman [16], and Thompson *et al.* [17] seems to violate this natural law.

In the development of the weighting function scheme, Lee [11] examined the

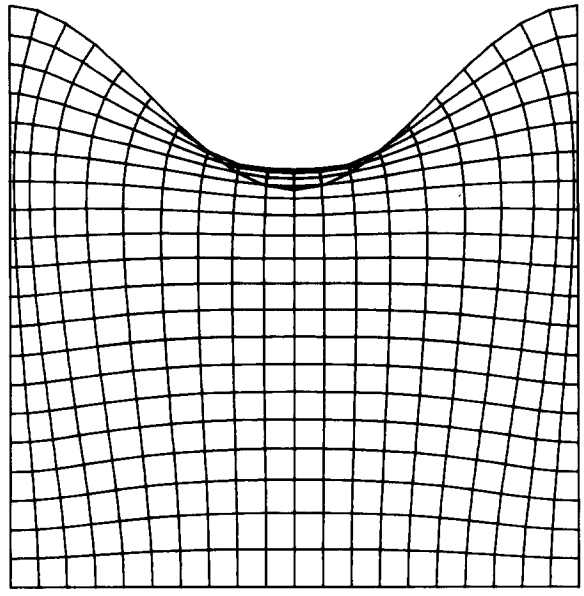

FIG. 1. Orthogonal grid system generated by Coleman's method [16].

Burgers' equation without the heat generation term. One of the two boundary temperatures was assigned zero while the other was maintained at unity. For such a problem, the temperature distribution is expected to be in the range of zero to unity due to the extremum principle [17]. However, the central difference scheme was found to produce negative temperature when the magnitude of the Z-parameter is larger than 2. This numerical instability has been proved to arise from the truncation error of the central difference scheme (see page 5 of [13]). The grid generation equations (2) and (3), in fact, are heat convection equations in the computational coordinates (ξ, η) with the control functions being a convective term. Hence, the central difference scheme could produce grids that violate the extremum principle [17] if the Z-parameters ($Z_1 = (J^2P/\alpha)\,\Delta\xi$ and $Z_2 = (J^2Q/\gamma)\,\Delta\eta$) have a magnitude of larger than 2. Such a restriction ($|Z| < 2$) in the use of the central difference scheme was also noted by Matsuno and Dwyer [15]. Fortunately, the weighting function scheme [11, 13] has been successfully developed from a locally analytical solution. This particular scheme produces an exact solution for any homogeneous ordinary differential equation with constant coefficients. It thus possesses no truncation error, although other modes of errors could arise when the coefficients are not constant or when it is applied on a multidimensional problem. The weighting function scheme gives positive weighting factors and thus always satisfies the physical law.

Now, let the weighting function scheme (18)–(20) be applied to Coleman's problem [16] by using the same set of control functions obtained by him. Figure 2

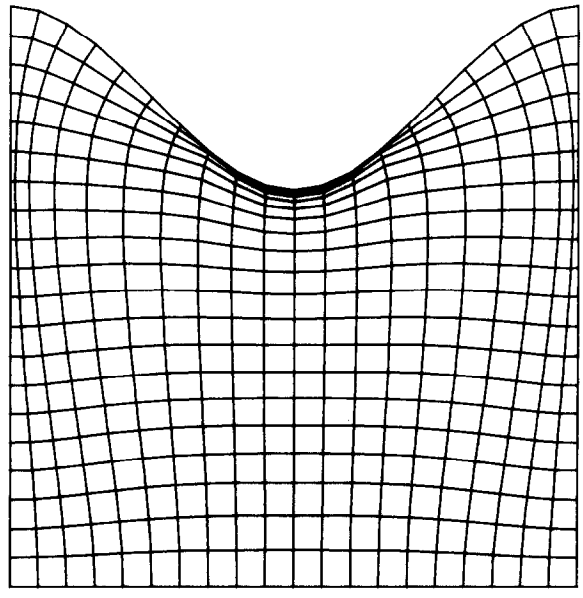

FIG. 2. Grids produced by weighting function scheme based on the control functions obtained from Fig. 1.

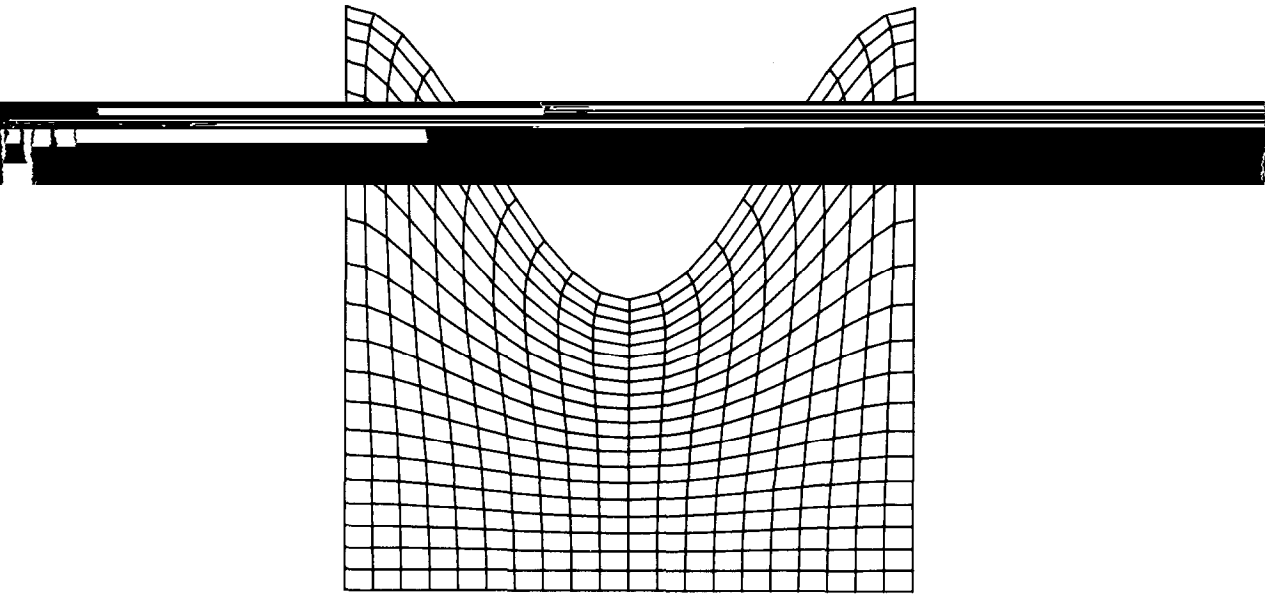

FIG. 3. Grid system generated by the present technique.

shows the resulting grid system. As expected, the weighting function scheme does not produce grids with negative Jacobian and thus does not reproduce the grids shown in Fig. 1. This seems to substantiate the point that the negative Jacobian as can be seen from Fig. 1 comes from the truncation error of the central difference scheme. To examine the performance of the present grid generation technique, Coleman's problem [16] is solved by the algorithm presented in the previous section. The resulting grid system is shown in Fig. 3. From Fig. 3, one sees that the grid system has a smoothly varying grid size over the entire domain with orthogonal grids at the four boundaries. No negative Jacobian is found even though the cavity in Fig. 3 is deeper than that in Fig. 1.

Large control functions occur also in an adaptive grid system due to the need of clustering grids at some particular locations. In testing the performance of their adaptive method, Matsuno and Dwyer [15] employed a physical quantity simulated by $f(x; \mathbf{a})$ that has a sharp but continuous increase at $x = 0.5$. The variation of the function $f(x; \mathbf{a})$ in a narrow region containing the point $x = 0.5$ is specified by the the parameter $\mathbf{a}$. As $\mathbf{a}$ assumes the value of infinity, the function $f(x; \infty)$ becomes a unit step function at $x = 0.5$. For a given value of $\mathbf{a}$, Matsuno and Dwyer [15] generated a grid system by using their adaptive method for the particular f-function. Both first and second derivatives of the function (i.e., f_x and f_{xx}) were then evaluated numerically from the exact value of the f-function defined at the grid points. The numerical results of f_x and f_{xx} can be expected to have poor accuracies if an improper grid system has been provided by the adaptive

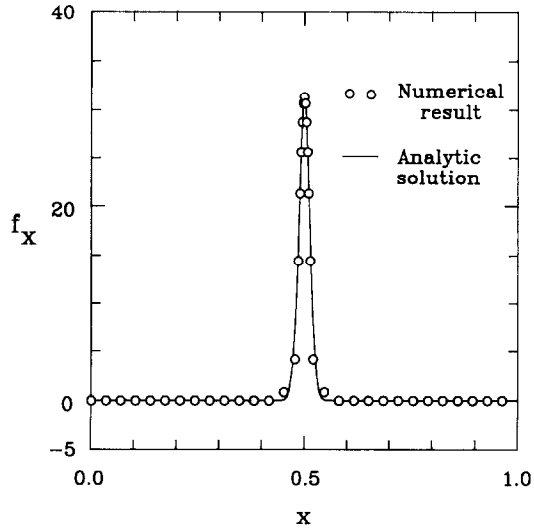

FIG. 4. The $f_x(x; 2000)$ result based on the first derivative adaptive method [15] with $K = 550$.

method. In this study, Matsuno and Dwyer found a serious numerical instability in generating their adaptive grids when the required control function P is very large such that $|J^2P| \geqslant 2$ (i.e., $\alpha = \Delta\xi = 1$). Owing to this, their adaptive method fails for $\mathbf{a} \geqslant 500$. In the present investigation, Matsuno and Dwyer's work [15] is repeated by using the weighting function scheme instead of the central difference scheme employed by them in generating an adaptive grid system. The results of first and second derivatives of the physical quantity $f(x; \mathbf{a})$ for $\mathbf{a} = 2000$ are then obtained by a central difference as done by Matsuno and Dwyer. Figures 4 and 5 show, respectively, the resulting values of f_x and f_{xx}. The exact solution is also plotted in Figs. 4 and 5 for comparison. Unlike the central difference scheme, the weighting function scheme was seen to pose no numerical difficulty for any **a** value. It always performed Matsuno and Dwyer's adaptive method "faithfully" even though the parameter **a** has such a large value ($\mathbf{a} = 2000$).

Figure 6 reveals an S-type grid system generated by the present numerical technique for a simply-connected region. Again, the grid size can be seen to vary smoothly in the entire domain with orthogonal grids at the four boundaries. Figure 7 is an example for an O-type grids. The inner and outer surfaces of the ring are specified, respectively, by $\eta = 0$ and $\eta = \eta_{\max}$. Making a branch cut in the horizontal position and then applying the present numerical technique, one obtains the grid network as shown in Fig. 7. It can be observed from Fig. 7 that the grid size has been controlled on the inner and outer boundaries through the use of a set of control functions P and Q. The horizontal line indicated by arrows is the branch cut. It is noteworthy that the equal-η curves possess continuous slope across the branch cut due to the orthogonal grids on both sides of the branch cut. This feature

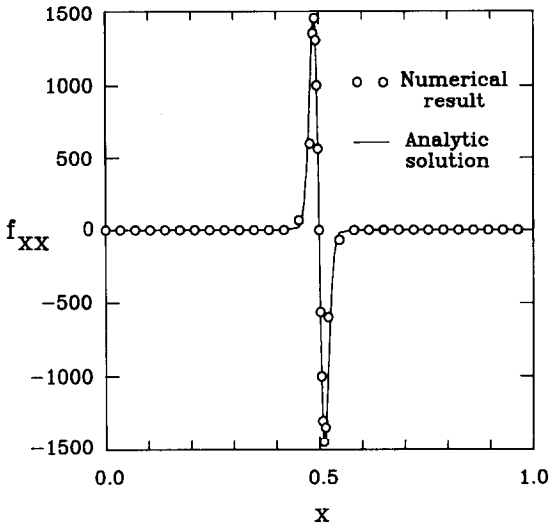

FIG. 5. The $f_{xx}(x; 2000)$ result based on the first derivative adaptive method [15] with $K = 550$.

is very important when the Navier–Stokes equations and the energy equation are applied at the branch cut. Note also that the grids in Fig. 6 is identical to a quarter of the ring shown in Fig 7. This means that the same grid in Fig. 7 can be produced from the result in Fig. 6 by employing the method of images. Therefore, the present grid generation technique has a potential to generate a grid system for a complex region by patching small pieces of patches together.

Good grid system is essential to the numerical solution in computing a flow field in an arbitrarily shaped region. For a turbine cascade, the C-type grids would be

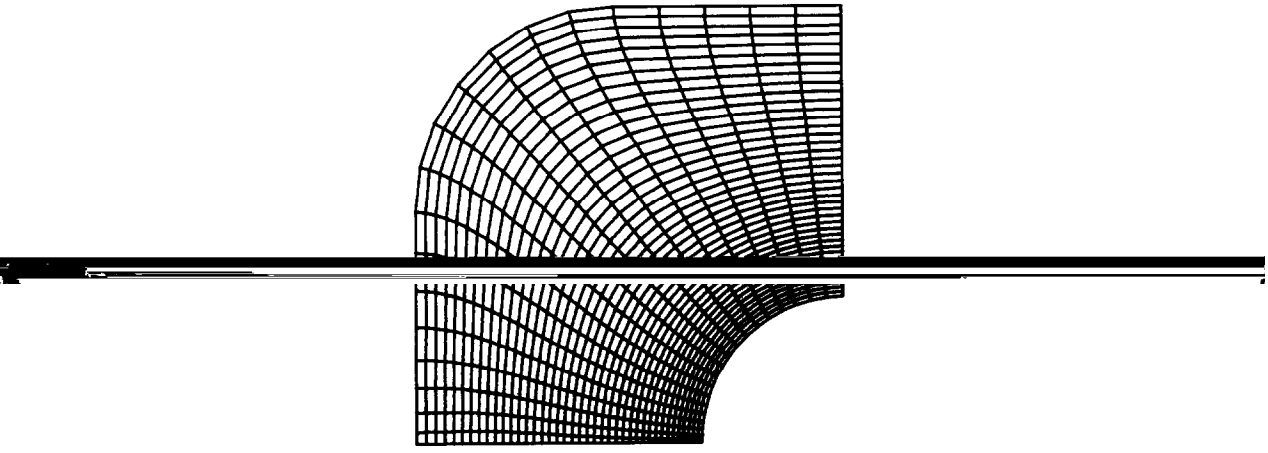

FIG. 6. S-type grid system for a simply-connected region.

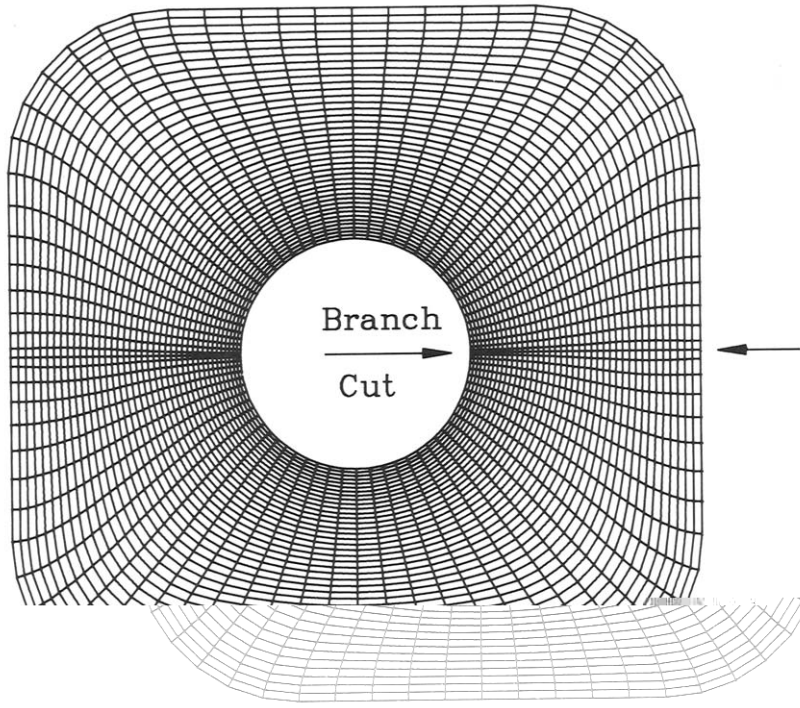

FIG. 7. O-type grid system for a ring.

the best grid system to achieve an accurate result in a region near the stagnation point. Figure 8 is the resulting C-type grid system obtained by using the present numerical technique for a turbine cascade. To impose an incident flow, a vertical surface is employed at the upstream boundary. The C-type grid system shown in Fig. 8 has a smoothly varying grid size over the entire physical domain. Small orthogonal grids are seen to exist around the surface of the turbine blade. The grids are also orthogonal to the outer boundary such that the slope of the ξ-curve is continuous across the interface to the next period of the cascade. Unlike the GRAPE code of Steger and Sorenson [9, 10], the present numerical technique allows an arbitrarily small aspect ratio (pitch to blade length) as mentioned earlier.

As a final discussion, it is mentioned that another branch of elliptic system has been developed for grid generation (see [18] for example). In this branch, the physical coordinates (x, y) is regarded as the temperatures governed by a variable thermal conductivity in the computational domain (ξ, η), i.e.,

$$\begin{aligned} \frac{\partial}{\partial \xi}\left(f \frac{\partial x}{\partial \xi}\right)+\frac{\partial}{\partial \eta}\left(\frac{1}{f} \frac{\partial x}{\partial \eta}\right)&=0 \\ \frac{\partial}{\partial \xi}\left(f \frac{\partial y}{\partial \xi}\right)+\frac{\partial}{\partial \eta}\left(\frac{1}{f} \frac{\partial y}{\partial \eta}\right)&=0, \end{aligned} \tag{24}$$

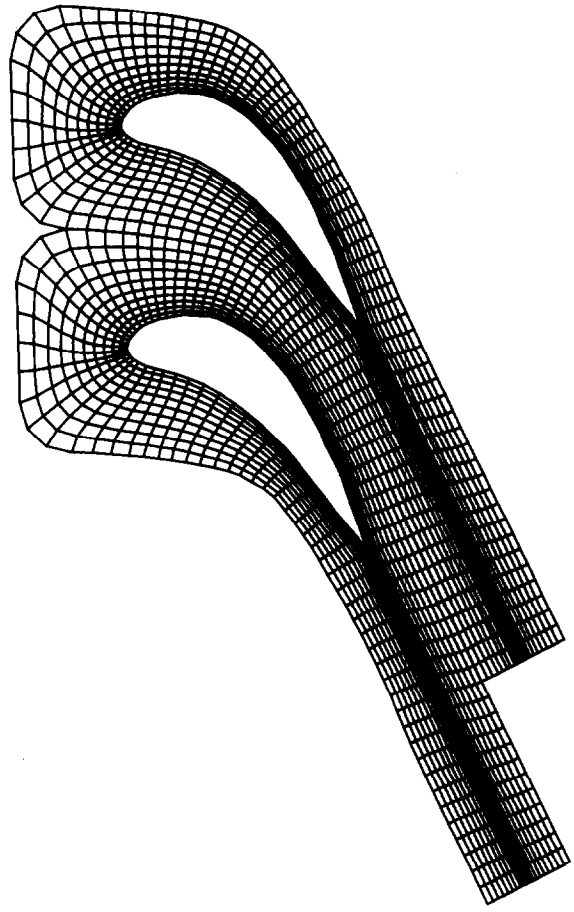

FIG. 8. C-type grid system for a turbine cascade.

where the thermal conductivity is $f(\xi, \eta)$ and $f^{-1}(\xi, \eta)$, respectively, in the ξ- and η-directions. For convenience, this branch of grid generation method will be referred to as the variable conductivity formulation (VCF). A grid system generated by Eqs. (24) can be proved to be orthogonal as long as $f(\xi, \eta)$ has the boundary value

$$f(\xi, \eta) = \left(\frac{x_\eta^2 + y_\eta^2}{x_\xi^2 + y_\xi^2}\right)^{1/2}. \tag{25}$$

In the grid generation procedure, the f-value inside the entire computational domain is interpolated from the boundary value. Equations (24) then are solved and the boundary f-value (25) is renewed. This procedure should be repeated until the solution converges within a prescribed tolerance.

The VCF branch seems quite straightforward in generating orthogonal grids. However, some disadvantages arise in the use of Eqs. (24) and (25). To examine the characteristics of the VCF branch, Eqs. (24) are rewritten in the conventional form (Eqs. (2) and (3)) with the coefficients

$$\alpha = f, \qquad \beta = 0, \qquad \gamma = f^{-1}$$
$$J^2 P = f_\xi, \qquad J^2 Q = -f^{-2} f_\eta. \tag{26}$$

In the conventional formulations (4), α and γ are used to represent the grid size and aspect ratio, while β denotes the grid distortion. This manner would provide good fundamental grids for most irregularly-shaped regions. The control functions P and Q (acting like a fluid flow) then are employed to modify the size and skewness of the fundamental grid. Thanks to this characteristic, a simple uniform grid system (say, $\Delta\xi = \Delta\eta = 1$) can be employed in the computational domain. This would greatly simplify the computations. The VCF branch, in contrast, has only one single variable $f(\xi, \eta)$ for the grid size and aspect ratio control as observable from Eqs. (26). Hence, a nonuniform grid on the computational domain (ξ, η) is generally needed. In addition, the f-value must be positive and the grid is restricted to be orthogonal. Under such a situation, the grid sizes $\Delta\xi$ and $\Delta\eta$ and the interpolation of $f(\xi, \eta)$ could have significant influences on the resulting grids as reported by Theodoropoulos *et al.* [18]. Another difficulty that could occur in the use of the VCF branch is that uncontrollable grids might propagate from a nonorthogonal corner or a nonsmooth boundary point where the boundary value of $f(\xi, \eta)$ is undefined. In the present method, such a numerical difficulty can be properly eliminated because nonorthogonal grids are allowed in the interior region.

Conclusion

A numerical technique has been proposed in the present investigation for grid generation by the use of Poisson equations. The values of the control functions at the two boundaries $\eta = 0$ and $\eta = \eta_{\max}$ are determined such that orthogonal grids with desired grid size can be specified there. The distribution of the control functions inside the computational domain then is interpolated with power-law functions instead of the conventional exponential functions. Thanks to such a strategy, the grids on $\eta = 0$ and $\eta = \eta_{\max}$ can be properly controlled. In the use of the GRAPE code, orthogonal grids are not guaranteed at the boundaries $\xi = 0$ and $\xi = \xi_{\max}$. To remedy this disadvantage, a particular treatment is proposed in the present study to yield orthogonal grids at the two boundaries $\xi = 0$ and $\xi = \xi_{\max}$. This characteristic is important in patching grids together without slope discontinuity across the interface of the patches. It also allows the Poisson equations to generate coordinates for O-type grids and periodic turbine cascade. For a region having boundaries of large curvature or when clustering grids are required at some particular locations, the control functions could have very large magnitudes. To

guarantee a good numerical stability in spite of the values of the control functions, the weighting function scheme along with the SIS solver is used for the solution of the elliptic grid generation equations. Through the examples illustrated in this study, the present numerical technique is seen to have excellent performances in generating S-type, O-type, and C-type grids.

APPENDIX A

Let the second-order elliptic differential equation

$$A\theta_{xx} + B\theta_{yy} + C\theta_x + D\theta_y = 0 \tag{A1}$$

be defined in the domain Ω and let the small region Ω_m ($0 \leqslant x \leqslant a$ and $0 \leqslant y \leqslant b$) be a subdomain of Ω. The size of Ω_m is assumed sufficiently small such that inside the small region Ω_m, the coefficients A, B, C, and D can be approximated by constant values. Based on such an approximation, the algebraic equation

$$\begin{aligned}\theta(x, y) = c_0 + c_1 e^{-(C/A)x} + c_2 e^{-(D/B)y} \\ + c_3 e^{-(C/A)x} - (D/B)\, y\end{aligned} \tag{A2}$$

is seen to satisfy the governing equation (A1) inside the subdomain Ω_m. For convenience, the four coefficients c_0, c_1, c_2, and c_3 are determined by employing the four corner values of θ, i.e.,

$$\begin{aligned}\theta_{SW} &= \theta(0, 0), \qquad \theta_{SE} = \theta(0, a) \\ \theta_{NE} &= \theta(a, b), \qquad \theta_{NW} = \theta(0, b).\end{aligned} \tag{A3}$$

After a rearrangement, Eq. (A2) becomes

$$\theta(x, y) = \{N_{SW}, N_{SE}, N_{NE}, N_{NW}\}\{\theta_{SW}, \theta_{SE}, \theta_{NE}, \theta_{NW}\}^T, \tag{A4}$$

where $\{\ \}$ and $\{\ \}^T$ stand for, respectively, a row vector and a column vector. The sape functions (or interpolation functions) are defined by

$$\begin{aligned}N_{SW}(x, y) &= L_1(x)\, L_1(y), \qquad N_{SE}(x, y) = L_2(x)\, L_1(y) \\ N_{NE}(x, y) &= L_2(x)\, L_2(y), \qquad N_{NW}(x, y) = L_1(x)\, L_2(y) \\ L_1(x) &= (e^{-Cx/A} - e^{-Ca/A})/(1 - e^{-Ca/A}) \\ L_2(x) &= (1 - e^{-Cx/A})/(1 - e^{-Ca/A}) \\ L_1(y) &= (e^{-Dy/B} - e^{-Db/B})/(1 - e^{-Db/B}) \\ L_2(y) &= (1 - e^{-Dy/B})/(1 - e^{-Db/B}).\end{aligned} \tag{A5}$$

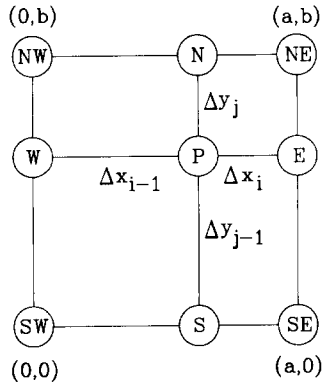

FIG. A1. Location of the point P.

Equations (A4) and (A5) constitute an interpolation for the solution of Eq. (A1) based on the θ values at the four corners of Ω_m. For the limiting case $C=D=0$, the L-functions in Eqs. (A5) will reduce to the first-order Lagrange polynomials that has been widely used in finite element methods. In the present investigation, the value of the cross derivative θ_{xy} at a particular point P inside the subdomain Ω_m is desired. As shown in Fig. A1, the location of point P is denoted with $(\Delta x_{i-1}, \Delta y_{j-1})$, while the values of a and b are, respectively, $a=\Delta x_{i-1}+\Delta x_i$ and $b=\Delta y_{j-1}+\Delta y_j$. Upon differentiating Eq. (A4) with respect to x and y followed by letting $x=\Delta x_{i-1}$ and $y=\Delta y_{j-1}$, one obtains the value of θ_{xy} for the pont P as

$$\begin{aligned} \theta_{xy} &= (\theta_{SW}+\theta_{NE}-\theta_{NW}-\theta_{SE})\,W, \\ W &= (CD/AB)(e^{Z_{1,i-1}}-e^{-Z_{1,i}})^{-1}\,(e^{Z_{2,j-1}}-e^{-Z_{2,j}})^{-1}, \\ Z_{1,i} &= (C/A)\,\Delta x_i, \qquad \text{and} \qquad Z_{2,j}=(D/B)\,\Delta y_j. \end{aligned} \tag{A6}$$

For the case of uniform grid ($\Delta x_{i-1}=\Delta x_i=\Delta x$ and $\Delta y_{j-1}=\Delta y_j=\Delta y$), Eqs. (A6) reduce to

$$\begin{aligned} \theta_{xy} &= (\theta_{SW}+\theta_{NE}-\theta_{NW}-\theta_{SE})\,w_f^*(Z_1)\,w_f^*(Z_2)/(\Delta x\,\Delta y) \\ w_f^*(Z) &= Z/(e^Z-e^{-Z}), \end{aligned} \tag{A7}$$

where $Z_1=(C/A)\,\Delta x$ and $Z_2=(D/B)\,\Delta y$.

ACKNOWLEDGMENTS

The authors wish to express their appreciation to the National Science Council of the Republic of China in Taiwan for the financial support of this work through the project NCS76-0401-E007-11.

REFERENCES

1. W. P. CROWLEY, memorandum, Lawrence Livermore National Laboratory, 1962.
2. A. J. WINSLOW, *J. Comput. Phys.* **1**, 149 (1966).
3. J. F. THOMPSON, F. C. THAMES, AND C. W. MASTIN, *J. Comput. Phys.* **15**, 299 (1974).
4. J. F. THOMPSON, F. C. THAMES, AND C. W. MASTIN, NASA CR-2729, 1977.
5. J. F. THOMPSON, F. C. THAMES, AND C. W. MASTIN, *J. Comput. Phys.* **24**, 274 (1977).
6. J. F. MIDDLECOFF AND P. D. THOMAS, AIAA Paper 79-1462, *AIAA Fourth Computational Fluid Dynamics Conference*, Williamsburg, VA, 1979.
7. J. K. HODGE, A. L. STONE, AND T. E. MILLER, *AIAA J.* **17**, 458 (1979).
8. J. L. STEGER AND R. L. SORENSON, *J. Comput. Phys.* **33**, 405 (1979).
9. R. L. SORENSON, NASA TM-81198, 1980.
10. R. L. SORENSON AND J. L. STEGER, in *Proceedings of Workshop on Numerical Grid Generation*, NASA CP-2166, 1980, p. 449.
11. S. L. LEE, *Int. J. Heat Mass Transfer* **32**, 2065 (1989).
12. S. L. LEE, *Numer. Heat Transfer* **16B**, 161 (1989).
13. S. L. LEE, T. S. CHEN, AND B. F. ARMALY, *Numer. Heat Transfer* **10**, 1 (1986).
14. R. L. BURDEN AND J. D. FAIRS, *Numerical Analysis*, 3rd ed. (PWS Kent, Boston, 1985), p. 117.
15. K. MATSUNO AND H. A. DWYER, *J. Comput. Phys.* **77**, 40 (1988).
16. R. M. COLEMAN, in *Proceedings of Workshop on Numerical Grid Generation*, NASA CP-2166, p. 213.
17. J. F. THOMPSON, Z. U. A. WARSI, AND C. W. MASTIN, *Numerical Grid Generation* (Elsevier Science, New York, 1985), pp. 189, 190, 193.
18. T. THEODOROPOULOS, G. BERGELESS, AND N. ATHANASSIADIS, in *Numerical Methods in Laminar and Turbulent Flow* (Pineridge, New York, 1985), p. 1747.