Noniterative Parabolic Grid Generation for Parabolized Equations

J. K. Hodge,* S. A. Leone,† and R. L. McCarty‡
Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio

Three-dimensional generalized curvilinear grids are generated for Cartesian, cylindrical, and spherical shaped bodies using parabolic partial differential equations. Elliptic grid equations are parabolized in an axial direction consistent with the Parabolized Navier Stokes (PNS) equations or the boundary-layer equations. The parabolized grid equations are solved in a noniterative fashion by marching in two directions. Any limitations of parabolic grid generation are investigated on convex and concave corners. Even though body surfaces are not smooth, the grid between axial stations is smoothed. Highly stretched and optimized grids for high Reynold's number are accurately and efficiently generated for the first time using parabolic grid equations with a difference approximation based on exponentials. The grid can also be adapted easily based on a flow solution at the previous axial station using this approach.

Introduction

NUMERICAL generation of boundary-fitted curvilinear grid systems has contributed significantly to the development of generalized finite-difference algorithms for the fluid flow about arbitrary bodies.[1] Many different grid generation techniques are available, and are presented in the proceedings in Refs. 1 and 2. Most techniques are based on the numerical solution of partial differential equations in a transformed region which is conveniently selected to be a square or cubic Cartesian grid. The numerical solution provides a one-to-one mapping with a general curvilinear grid in a physical region. The numerical solution of elliptic partial differential equations[3] has been developed for grid generation and used more than other techniques. The numerical solution of hyperbolic grid equations has also been developed.[4] Recently, parabolic equations which are approximations to the elliptic equations have been introduced by Nakamura,[5] and applied by Noack[6] and Edwards.[7]

Advantages and disadvantages of these elliptic, hyperbolic, and parabolic grid equations are discussed, and then an overview of new extensions to the parabolic grid technique presented in this paper will be given.

The elliptic grid generation equations are based on Laplace's equation or Poisson's equation, which normally guarantees that a maximum principle is satisfied, as long as Dirchlet boundary values are prescribed and appropriate forms of the grid control functions are selected. The major advantages of elliptic grid generation can be attributed to the solution of Laplace's equation. As a result of the maximum principle, a one-to-one mapping is obtained, the Jacobian of the general transformation is not zero in the interior of the domain, and a generalized transformation exists. Prescribing the boundary points for Dirchlet boundary values is also considered an advantage in many applications, although these boundary values are given up in some applications to obtain orthogonality. The domain of influence of an elliptic equation and Laplace's equation could be considered an advantage since all points in the domain influence each other. It is a disadvantage when the cost of numerically solving the equation is considered, especially since the equation to be solved in the transformed space becomes quasilinear and requires iteration. Finally, the solution to Laplace's equation is analytic and therefore provides a smooth grid distribution with no discontinuities in the interior of the domain, which may occur with hyperbolic equations.

The hyperbolic grid generation equations are usually chosen from orthogonal relations and by prescribing the Jacobian. Therefore, an orthogonal grid is generated which may be an advantage in some applications. The major advantage of the hyperbolic equations is that a marching solution can be obtained requiring no iteration, and thus the cost of numerically solving the hyperbolic equations is low compared to elliptic grid generation. However, in the numerical solution artificial damping terms for stability are required, outer boundary points can not be prescribed, and discontinuities are allowed. Discontinuities at boundaries propagate throughout the interior of the domain. Grid generation equations which smooth out such discontinuities but would allow numerical solution by marching would avoid the major disadvantages of both the hyperbolic equations and the elliptic equations.

The parabolic grid equations are chosen to be approximations to the elliptic equations; thus most of the advantages of the elliptic equations can be obtained depending on the approximations. The one-to-one mapping of elliptic equations should be approximated. Since this mapping must be required, this is a major question for parabolic grid generation. Smoothness should also be obtained with parabolic equations. Boundary values can be prescribed even in the direction of marching with appropriate approximations. The domain of influence is controlled by the approximations to the extent desired, but numerically the domain is limited to the parabolic domain behind the marching directions. Numerically, the equations can be solved by marching, and thus the cost is significantly lower than for the elliptic equations. Control of the grid distribution can be accomplished as for elliptic grids, but appears to be easier because of the limited domain of influence, thus allowing easier grid adaption based on flow solutions. This could be a major advantage of parabolic grids, especially when coupled to flow solutions which have the same parabolic character as the grid generation. Therefore, parabolic grid generation has some

Presented as Paper 85-1682 at the AIAA Seventh Computational Fluid Dynamics Conference, Cincinnati, OH, July 15-17, 1985; received Oct. 15, 1985; revision received June 25, 1986. This paper is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

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

†Assistant Professor, Department of Aeronautics and Astronautics. Member AIAA.

‡Aeronautical Engineer, former Graduate Student.

major advantages which could be valuable if the approximations to the elliptic equations are valid. If the limitations of the approximations can be documented, and solutions recommended, then parabolic grid generation can become a very valuable tool for grid generation.

One of the objectives of this paper is to document limitations of various approximations that give parabolic grid generation equations and then recommend solutions. These limitations are related to the complexity of the boundaries. For simple geometries, very simple grid generation techniques will work and will be much more efficient and cost effective. Therefore, one-dimensional grid generation will be covered first.

The one-dimensional grid generation technique in this paper has some unique aspects. The one-dimensional ordinary differential grid equation approximates the elliptic equations along a grid line in each direction in many cases.[8,9] The elliptic equations are chosen such that the one-dimensional grid equation is linear and can be solved by the solution of one tridiagonal matrix equation. Also, if the grid control function is chosen to be a constant, then an analytical solution is of exponential form, and an exact numerical solution is obtained except for round-off error by assuming an exponential form instead of a polynomial.[10]

The one-dimensional technique is then extended to the two-dimensional parabolic grid equations. A simple parabolic equation (which has an analytical solution) is solved first on a flat plate, then on a convex corner, and finally on a concave corner. The elliptic equations are then approximated to avoid limitations that occurred on the concave corner. The parabolic grid solution is also demonstrated for a curved surface for a polar grid with the same program without introducing a singularity.[11] These two-dimensional parabolic grid solutions are uniquely different than those obtained by Nakamura,[5] Noack,[6] or Edwards.[7] The grid equations are parabolized in the axial direction and not in the radial direction, thus allowing coupling with PNS or boundary-layer solutions. The equations also have the elliptic grid control functions to vary grid spacing in both axial and radial directions. In fact, in some cases the spacing in the radial (or normal) direction is optimized based on power law profiles and is a function of the Reynold's number and axial length. The stiffness of the equations in the radial direction, because of the large control functions, does not cause any significant degradation in efficiency or any grid oscillations.

The two-dimensional parabolic grid generation equations are easily extended to three dimensions simply by approximating the elliptic equations and marching in two directions. The planes are dependent, so discontinuities will be smoothed. This is uniquely different than Noack's[6] axis-normal grid, in which the planes are all independent, thus allowing discontinuities to exist. Three-dimensional grids were generated for a flat plate, a cylindrical biconic surface, and a spherical surface with varying radius and offset inner and outer boundaries.

One-Dimensional Grid Generation

One-dimensional grids can be generated by the numerical solution of an ordinary differential equation that is an analog or approximation of the elliptic partial differential equations used in multidimensional grid generation. These grids can be used in one-dimensional model problems or to generate two-dimensional or even three-dimensional grids for configurations that are not very complex and are smoothly varying. The numerical solution of the ordinary differential equation can be accomplished in a fashion similar to the solution of the elliptic partial differential equations.

A one-dimensional grid equation that is an analog of Thompson's original partial differential equation[3] was studied by Ghia and Hodge.[8,9] The resulting equation was quasilinear to third order and stiff when the grid control function was optimized for high Reynold's number flow problems, and was thus very difficult to solve numerically. Ghia solved the equation by using upwind-directed differences, by quasilinearizing the third-order term, and by solving a tridiagonal matrix which was then iterated until convergence. Hodge solved the one- and two-dimensional equations by using upwind-directed differences, by calculating locally optimized Successive Over Relaxation (SOR) parameters, and then solving by SOR iteration. The upwind differencing eliminated oscillations and caused numerical truncation error, which is not a problem unless an accurate solution of the original grid equations is important for optimization reasons. In this paper, a slight modification of the questions and a different numerical approach will avoid the problems of lack of linearity, stiffness, and accuracy.

Equations

A linear one-dimensional grid equation results when a modified form of Thompson's elliptic equations[12,13,14] is used. The equation is

$$r_{\eta\eta} + P_2 r_\eta = 0.0 \tag{1}$$

where η is the transformed coordinate and r is considered to be any appropriate coordinate such as x, y, or z in a Cartesian coordinate system, r, θ, or z in a cylindrical system, r, θ, or ϕ in a spherical system, or arc length s. This equation can be solved analytically in special cases.

Analytical solutions for two cases are of interest. First, if the grid control function P_2 is zero, then a constant step-size grid is generated. Second, if P_2 is a constant, then the solution is of exponential form, and the physical grid step size varies exponentially. Many grids are currently generated analytically for turbulent flow about simple geometric configurations with similar exponential functions. The preceding differential equation thus gives families of exponential grids. A constant outer grid and inner exponential grid would result from changing P_2 from a constant to zero in the interior of the domain, thus avoiding grid embedding, if Eq. (1) is solved numerically.

Since exponential grids are required for viscous flows, the numerical solution of Eq. (1) is also based on an exponential form. Roscoe[10] refers to such a numerical scheme as the unified difference representation (UDR), and the difference equation for Eq. (1) is

$$-[\mu(-P_2)+\mu(P_2)E_2]r_{j-1}+(1+E_2)r_j$$
$$-[\mu(P_2)+\mu(-P_2)E_2]r_{j+1}=0.0 \tag{2a}$$

$$\mu(P_2)=0, \quad \text{if } P_2 \le 0; \quad \mu(P_2)=1, \quad \text{if } P_2>0 \tag{2b}$$

$$Q_2 = |P_2| \tag{2c}$$

$$E_2 = \exp(-Q_2) \tag{2d}$$

If the grid control function P_2 is a constant, than an exact solution to Eq. (1) is obtained numerically for any number of grid points except for round-off error. Large errors result if the usual finite-difference approximations based on first- or second-order polynomials are used. When P_2 is not a constant, there are truncation errors in the UDR solution depending on the variation from an exponential. In all cases that have been attempted, Eqs. (2) have resulted in more accurate numerical solutions.

Since the one-dimensional grid equation is linear, the difference equation is linear and requires no iteration. Two boundary values are specified, and then a tridiagonal algorithm solves the linear system of difference equations very efficiently. The differential equation is solved accurately also, which may be important for grid optimization.

Grid Optimization

The truncation error of finite-difference flow solutions depends on the number of grid points, but the distribution of grid points can be optimized also to reduce truncation error.[9] A simple analytical example can illustrate how the grid can be optimized, and if Eq. (1) is solved numerically, an accurate solution is required.

Consider a power law velocity profile in a boundary layer. The power law profile in the physical coordinate is mapped into a second-order polynomial in the transformed coordinate: thus conventional second-order differencing of derivatives with respect to the transformed variable will give zero truncation error! The profile and its derivates are given by

$$U = Y^{1/m} = \bar{\eta}^2 \tag{3a}$$

$$U_{\bar{\eta}} = 2\bar{\eta} \tag{3b}$$

$$U_{\bar{\eta}\bar{\eta}} = 2 \tag{3c}$$

$$U\bar{\eta}\bar{\eta}\bar{\eta} = 0 \tag{3d}$$

where Y, U, and $\bar{\eta}$ are normalized to vary from zero to one. Note that all higher order derivatives in the transformed space of the velocity profile U, and thus truncation error, are zero. The derivatives of the physical coordinate with respect to the transformed coordinate are

$$Y_{\bar{\eta}} = 2m\bar{\eta}^{2m-1} \tag{4a}$$

$$Y_{\bar{\eta}\bar{\eta}} = 2m(2m-1)\bar{\eta}^{2(m-1)} \tag{4b}$$

(Note, however, that numerical approximations to these derivatives will introduce error.) By substituting in Eq. (1) and solving for P_2, the optimum grid control function for power law profiles is

$$P_2 = -(2m-1)/\bar{\eta} \tag{5}$$

The constant m is normally 2 for laminar boundary layers and changes to 7 for turbulent boundary layers. Thus, an optimized one-dimensional grid can be generated by numerically solving Eqs. (2) with the control function given by Eq. (5). An exact solution is not obtained by the UDR, but the error is less than that for a finite-difference approximation, especially when $m = 7$.

Results

As an illustration, a two-dimensional grid is generated based on the one-dimensional grid equation for a laminar boundary layer with a power law profile. A laminar boundary layer grows proportionally with the square root of the axial distance. Thus, the one-dimensional grid is scaled with the axial distance to give the grid shown in Fig. 1. The physical grid with the leading edge singularity maps to a rectangular region with a square grid in the two-dimensional transformed coordinates as discussed in the next section.

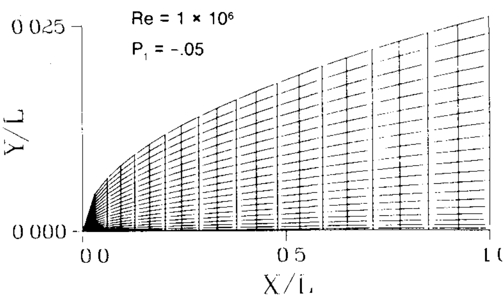

Fig. 1 Optimized grid for power law profile on flat plate (one-dimensional grid equation).

Two-Dimensional Parabolic Grid Generation

Two-dimensional grids can be generated by the numerical solution of parabolic partial differential equations that are similar to the preceding one-dimensional grid equation. A simple parabolic equation which has an analytical solution will be investigated first. Then two-dimensional elliptic equations will be parabolized by making some approximations similar to Nakamura.[5] However, the grid control function for variable grids will not be assumed to be zero, and the approximations and thus marching direction will be in an axial direction. The above one-dimensional grid technique is extended to the two-dimensional procedure. Although this approach applies in general, the equations are simplified for an axis-normal grid. Results are then given to illustrate the limitations of the approximations and solutions to the limitations.

Equations

A simple parabolic partial differential equation similar to the preceding one-dimensional grid equation can be used for two-dimensional grid generation in some cases. The following equation is proposed:

$$Rr_{\xi} = r_{\eta\eta} + P_2 r_{\eta} \tag{6}$$

where R is a smoothing parameter and ξ is the marching direction. This equation has an analytical solution of the form

$$r = e^{-(\lambda^2/R)\xi} e^{(P_2/2)\eta} (Ae^{C\eta} - Be^{-C\eta}) \tag{7}$$

where A, B, C, R, P_2, and λ are constants. Part of this solution has an exponential form similar to the one-dimensional grid equation given by Eq. (1), and will generate highly stretched grids for large Reynold's number. The solution is also multiplied by another exponential factor which will damp or smooth variations in the ξ coordinate direction over approximately $3\lambda^2/R$ steps.

Another form of smoothing is introduced in the numerical solution of Eq. (6). A first-order backward difference is used for the ξ derivative, and its truncation error term has a smoothing effect. The other derivatives are approximated by the UDR for nonhomogeneous equations.[10]

The one-dimensional grid equation is used to generate the initial boundary values to start the solution. At the next axial plane, the linear difference equation is solved by a tridiagonal algorithm. By marching in the axial direction, a grid is generated throughout the interior of the domain. At the last axial location in the domain, either a zero gradient or an extrapolation boundary condition are applied. Boundary points at this last location have no effect on the grid at previous locations and cannot be prescribed arbitrarily when Eq. (6) and the damping parameter are used.

The parabolic grid equation in Eq. (6) can be extended by adding a source term. This second form of a parabolic grid equation is

$$Rr_{\xi} = r_{\eta\eta} + P_2 r_{\eta} + S(\Delta r) \tag{8}$$

This form without the grid control $P_2 = 0$ is similar to the form proposed by Nakamura,[5] which was related to the elliptic grid equations.

The proposed form of the two-dimensional parabolic grid generation equations is derived by starting from the elliptic grid equations. The elliptic grid equations for mapping between the physical space and the transformed ξ,η space are of the form

$$\alpha_{11}(r_{\xi\xi} + P_1 r_{\xi}) + \alpha_{22}(r_{\eta\eta} + P_2 r_{\eta}) = -2\alpha_{12} r_{\xi\eta} \tag{9}$$

where r is chosen to be either $(x,y)^T$ in a Cartesian coordinate system or $(r,\theta)^T$ in a polar system. For a Cartesian system the coefficients are given in Thompson.[1] If only an axis-normal grid is required, one coordinate is prescribed (x, for example) or generated by one-dimensional equations, and the other coordinate (y) is obtained from Eq. (9).

To parabolize the elliptic equation in an axial direction (ξ direction), any second derivatives with respect to ξ are approximated by central differences. The second derivative with respect to ξ is split into the difference between the first derivatives. The equation becomes

$$\{\alpha_{11}[1+Q_1\mu(-P_1)]\}r_\xi|_{i-1/2}=\alpha_{22}(r_{\eta\eta}+P_2r_\eta)$$
$$+\{\alpha_{11}[1+Q_1\mu(P_1)]r_\xi|_{i+1/2}+2\alpha_{12}r_{\xi\eta}\} \qquad (10)$$

which is of the same form as Eq. (8). Thus, the part of the difference approximations at $i+1/2$ becomes the source term which must now be prescribed or approximated to obtain a marching solution. The source term is primarily a slope which can be approximated by prescribed points at some I^+ location. This I^+ location may be anywhere between the current axial solution station and the last axial location in the domain, and may be fixed or moving at a prescribed interval ahead of the solution. For example, I^+ may be $i+1$ or $i+5$, but limited to the last axial location ($i=I$MAX). The points are prescribed by solving the one-dimensional grid equation [Eq. (6)] at the I^+ location (identical to the "locally similar" procedure used by Ghia[15] as a guess for the elliptic equations). The localized orthogonality procedure used by Nakamura and Edwards could also be implemented at this point; it was not investigated for this paper, but has been done by Miller.[16]

The parabolic difference equation for Eq. (10) can now be obtained. The ξ derivatives are approximated by first-order directed differences, and η derivatives by the UDR. The difference equation is linear if the coefficients are constant, and can be solved by a tridiagonal algorithm without iteration. Since the coefficients depend on the solution, they must be approximated from the solution at the previous axial station or from the following one-dimensional solutions that are generated at I^+. An option to iterate on the coefficients is available, but is usually unnecessary.

The parabolic grid generation procedure for axis-normal grids can be summarized by the following:

1) Lower ($j=1$) and upper ($j=J$MAX) axial boundary points are prescribed or obtained by one-dimensional grid generation [Eqs. (2)].

2) The initial axial boundary points are solved at $i=1$ and $i=2$ by one-dimensional grid generation [Eqs. (2)].

3) The I^+ location is prescribed and the one-dimensional grid generation [Eqs. (2)] provides the approximation to the source terms.

4) The metrics and coefficients are approximated.

5) The two-dimensional equation is solved by a tridiagonal algorithm starting at $i=2$ ($i=1$ for a symmetry plane).

6) Steps 3-5 are repeated by marching in an axial direction.

7) The last axial station is approximated by extrapolation or by a zero gradient. Thus, two tridiagonal solutions are required at each axial location depending on I^+ (only one if I^+ is the last axial location). If the coefficients were iterated, an additional tridiagonal solution would be required per iteration.

Results

Grids were generated using the two-dimensional procedure primarily to investigate any limitations or problems with the parabolic grid equations. The simple parabolic grid equation without a source term was investigated first. Grids on convex and concave corners were used to illustrate any problems caused by large curvature and to demonstrate the smoothing ability of the parabolic technique. The parabolized grid equation that approximates the elliptic grid equations was then investigated.

Convex Corner

Grids were first generated on a convex corner using the simple parabolic grid equation [Eq. (6)]. Grids (21×22) on a convex 45 deg corner are shown in Fig. 2. For $R=0$, which reduces to a one-dimensional grid, the discontinuity in boundary slope caused discontinuities in the grid slopes (and thus metrics) in the interior of the domain as shown in Fig. 2a. A 90 deg convex corner would exaggerate this even more. For $R=3$, the discontinuities are smoothed dramatically as shown in Fig. 2b. The amount of smoothness required, however, is a function of the flow solution and algorithm. This illustrates that the parabolic grid equation smooths the grid as expected based on the analytical solution. The truncation error caused by the first-order derivative approximation is a second way in which the grid is smoothed. Note that both of these ways only smooth the grid downstream in the marching direction, which is expected from the parabolic nature of the grid equation. No problem with one-to-one mapping occurred even when P_2 was very large.

Concave Corner

Grids which were generated on a concave corner using the simple parabolic grid equation [Eq. (6)] did have a one-to-one mapping problem. The grid crossover at the boundary with a 45 deg slope is shown in Fig. 3 for $R=1$. This grid is unacceptable since the Jacobian of the transformation goes through zero at some point, and therefore the transformation between physical space and transformed space does not exist. The damping parameter R must be reduced to a small value. Then the boundary discontinuity causes discontinuities in the metrics very similar to those occurring in one-dimensional grid generation.

Concave Corner with Source

The one-to-one mapping problem for the concave corner can be solved by adding a slope or source term to Eq. (6). A more appropriate source term can be obtained from an ap-

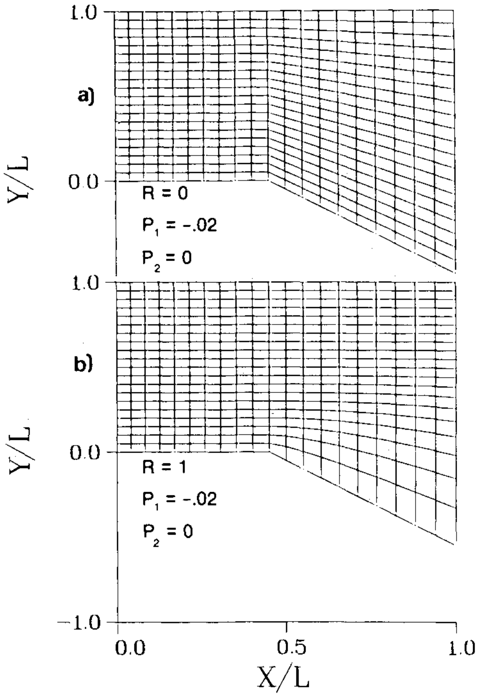

Fig. 2 Convex corner with and without smoothing by simple two-dimensional parabolic grid equation.

proximation of the elliptic equations. The parabolic difference equation with a source term was solved. The resulting grid with $I^+ = i+1$ is shown in Fig. 4a. With the source term a one-to-one mapping problem did not occur on the convex corner, and the discontinuities were smoothed. More smoothing is shown in Fig. 4b with $I^+ = i+5$. Note that the grid is now smoothed before the discontinuity that is characteristic of elliptic grids. However, this influence can be controlled if desired (5 grid lines here!). Prescribing I^+ at the last axial boundary would decrease the computation time by a factor of almost half, but this may cause too much smoothing and could lead to a one-to-one mapping problem for other cases. Take a wavy wall for example; I^+ should be less than the number of grid points in each quarter cycle for the wavy wall. Thus, this limitation of parabolic grid generation can be easily solved.

Convex Corner with Source

Although the source term was not required for convex corners or surfaces, the parabolic difference equation was also solved for the convex corner. The grid for $I^+ = i+5$, $P_2 = -.05$, and two iterations is shown in Fig. 5. Some smoothing occurs and a small step size is obtained at the corner. This step size is larger for $I^+ = i+1$. Therefore, $I^+ = i+1$ is recommended for convex corners. This corner does cause a problem if large values of P_2 are prescribed and a highly stretched grid is desired. A one-to-one mapping problem can occur at the corner, but can be solved by iterating at least once on the coefficients of Eq. (9). This was done to generate the grid in Fig. 5. Another solution is to decrease the source term magnitude near a convex boundary with large curvature. Another solution is to use intermediate transformations that change the curvature problem to only a problem with variations in the curvature.

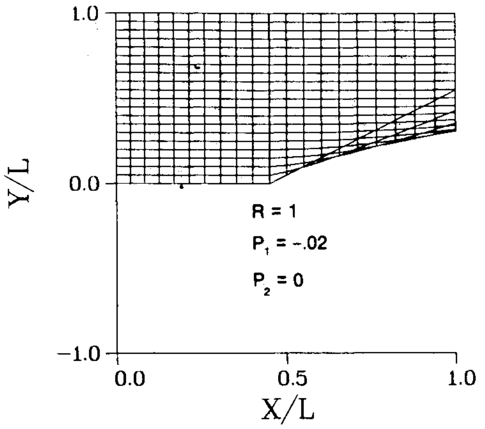

Fig. 3 Concave corner with smoothing by simple two-dimensional parabolic grid equation.

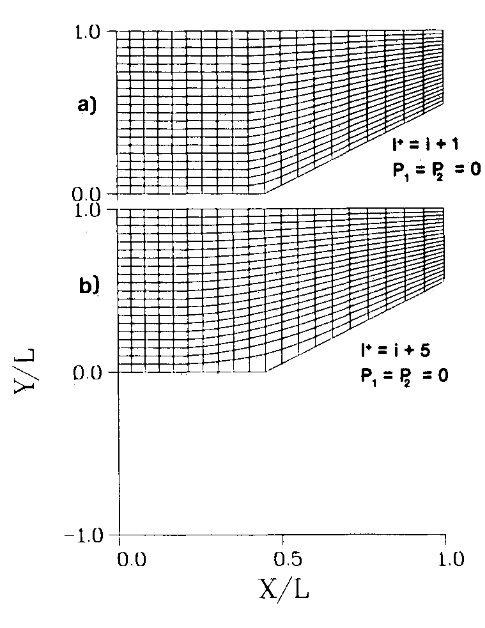

Fig. 4 Concave corner with smoothing by two-dimensional parabolic grid equation.

Polar Grid

An intermediate transformation can be used to minimize the problems encountered by parabolic grid generation for configurations with convex surfaces. For two dimensions, the grid can be generated in terms of polar coordinates. This will limit problems to changes in curvature. An example for a cylinder is shown in Fig. 6. A symmetry condition is necessary at each end of the grid, and can be approximated easily.

Flat Plate

The two-dimensional parabolic grid equations were solved on a flate plate, with the optimized grid control function for power law profiles in Eq. (5). Transition from laminar flow $m = 2$ was prescribed at the quarter chord, and fully turbulent flow $m = 7$ was prescribed at midchord. Thus, a discontinuity in the grid control was prescribed. The parabolic equations smoothed this discontinuity using $I^+ = i+5$, as shown in Fig. 7 for a Reynold's number of one million. This is only the first step towards optimized grid control. The parabolic grid generation technique can be efficiently used in a boundary-layer program where the optimized grid control is based on the computed profiles. The same grid shown in Fig. 7 was also generated in an initial plane by the three-dimensional parabolic grid equations.

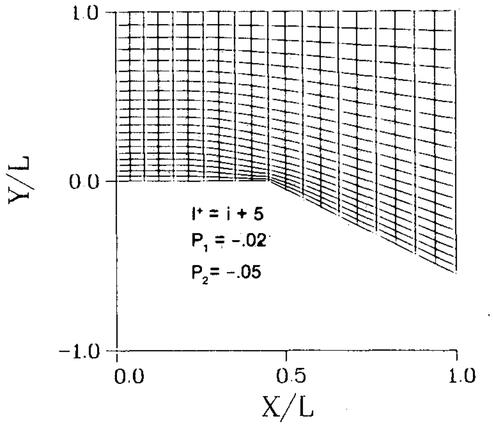

Fig. 5 Convex corner with smoothing by two-dimensional parabolic grid equation.

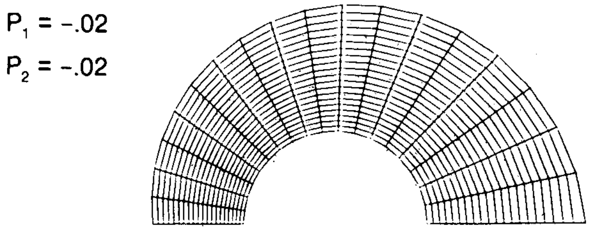

Fig. 6 Circular cylinder in polar coordinates by two-dimensional parabolic grid equation.

Three-Dimensional Parabolic Grid Generation

Once the procedures for two-dimensional parabolic grid generation have been developed, the procedures for parabolic three-dimensional grid generation are almost identical. The three-dimensional elliptic equations can be approximated to march only in one direction, and an iterative solution (equivalent to solution of two-dimensional elliptic equations) is required at each surface in general, or at each axial plane if an axis-normal grid is required. The three-dimensional elliptic equations can also be approximated to march in two directions, and then three-dimensional grid generation is analogous to a combination of two-dimensional parabolic grid generations, except for the addition of more terms to each equation. In either case, the planes are tied together so discontinuities are smoothed.

Equations

The three-dimensional elliptic grid equations can be written in general[14,17] as

$$\alpha_{11}(r_{\xi\xi}+P_1 r_\xi)+\alpha_{22}(r_{\eta\eta}+P_2 r_\eta)$$
$$+\alpha_{33}(r_{\zeta\zeta}+P_3 r_\zeta)=-2(\alpha_{12}r_{\xi\eta}+\alpha_{13}r_{\xi\zeta}+\alpha_{23}r_{\eta\zeta}) \qquad (11a)$$

where

$$\alpha_{ij}=\sum_{n=1}^{3}\gamma_{ni}\gamma_{nj} \qquad (11b)$$

and γ_{ij} is the ijth signed cofactor of the following matrix

$$\begin{bmatrix} r_\xi & r_\eta & r_\zeta \\ \theta_\xi & \theta_\eta & \theta_\zeta \\ z_\xi & z_\eta & z_\zeta \end{bmatrix} \qquad (11c)$$

where $r=(z,r,\theta)^T$ for cylindrical coordinates. The singularity in cylindrical coordinates is avoided,[9,15] and all coordinates are treated nondimensionally. Essentially, an intermediate transformation between the physical coordinates and the transformed coordinates (ξ,η,ζ) is introduced. Also, Cartesian coordinates with $r=(z,y,x)^T$ or spherical coordinates with $r=(\phi,r,\theta)^T$ are generated by the same equation. The physical coordinates are rotated in the definition of r for programming convenience, but are defined in a right-hand system.

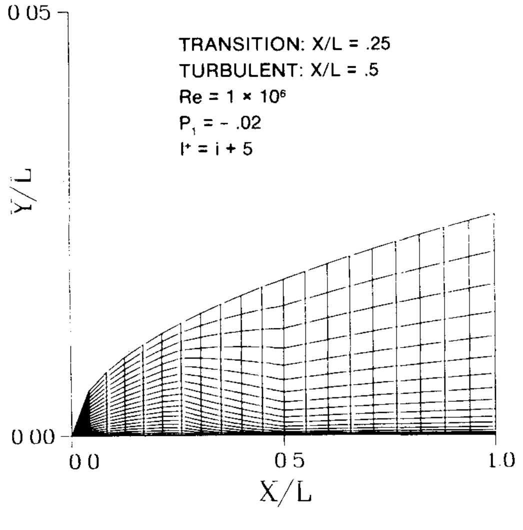

Fig. 7 Optimized grid for power law profiles with transition on flate plate (two-dimensional parabolic grid equation).

For an axis-normal grid, the coefficients simplify dramatically since $z=z(\xi)$ only. In addition, $\theta=\zeta$ is also chosen. For this case, only one of the three equations represented by Eq. (11a) must be solved. The general procedure is illustrated, however, and numerically each equation is similar.

The elliptic equation is parabolized in both the ξ and ζ directions in an analogous fashion to the procedure used for the two-dimensional equations. Similar difference approximations give the final difference equation for a three-dimensional axis-normal grid as

$$-\alpha_{22}D\{[\mu(-P_2)+\mu(P_2)E_2]r_{i,j-1,k}$$
$$+[\mu(P_2)+\mu(-P_2)E_2]r_{i,j+1,K+}\}+\{\alpha_{11}[1+Q_1\mu(-P_1)$$
$$+[1+Q_1\mu(P_1)]/(I^+-i)]+\alpha_{33}[1+Q_3\mu(-P_3)$$
$$+[1+Q_3\mu(P_3)]/(K^+-k)]$$
$$+\alpha_{22}D(1+E_2)\}r_{i,j,k}=\alpha_{11}\{[1+Q_1\mu(-P_1)]r_{i-1,j,k}$$
$$+[1+Q_1\mu(P_1)]/(I^+-i)r_{I+,j,k}\}+\alpha_{33}\{[1+Q_3\mu(-P_3)]r_{i,j,k-1}$$
$$+[1+Q_3\mu(P_3)]/(K^+-k)r_{i,j,K+}\}+\alpha_{12}(r_{I+,j+1,k}-r_{i-1,j+1,k}$$
$$-r_{I+,j-1,k}+r_{i-1,j-1,k})/(I^+-i+1)+2\alpha_{13}(r_{I+,j,K+}-r_{I+,j,k-1}$$
$$-r_{i-1,j,K+}+r_{i-1,j,k-1})/(I^+-i+1)/(K^+-k+1)$$
$$+\alpha_{23}(r_{i,j+1,K+}-r_{i,j-1,K+}$$
$$-r_{i,j+1,k-1}+r_{i,j-1,k-1})/(K^+-k+1) \qquad (12a)$$

$$D=Q/(1-E) \qquad \text{if } Q\geq 10^{-6}$$
$$=1/(1-Q/2+Q^2/6-Q^3/24+Q^4/120) \quad \text{if } Q<10^{-6} \qquad (12b)$$

where $\mu(P)$, Q, and E are defined in Eqs. (2) with a subscript added to correspond to P_1 or P_2 or P_3. The limiting form of D near $P=0$ is handled by a five-term Maclaurin's expansion. Note that source terms which are approximated by one-dimensional grid generation [Eqs. (2)] occur in both marching directions. Both I^+ and K^+ must be prescribed. This difference equation is also linear if the coefficients are approximated, and is solved without iteration (or could be iterated) by a tridiagonal algorithm.

The three-dimensional parabolic grid generation procedure is almost identical to the preceding two-dimensional procedure summary, except for the extra marching direction and the handling of boundary points if they are generated and not prescribed. Boundary points can be prescribed or

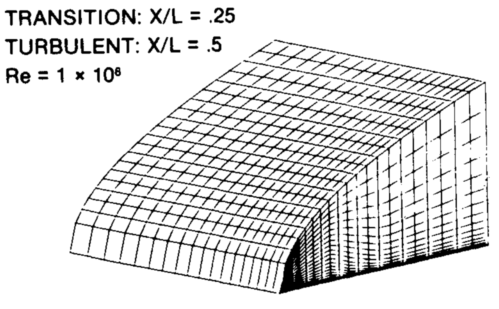

Fig. 8 Optimized grid for power law profiles with transition on flate plate (three-dimensional parabolic grid equations).

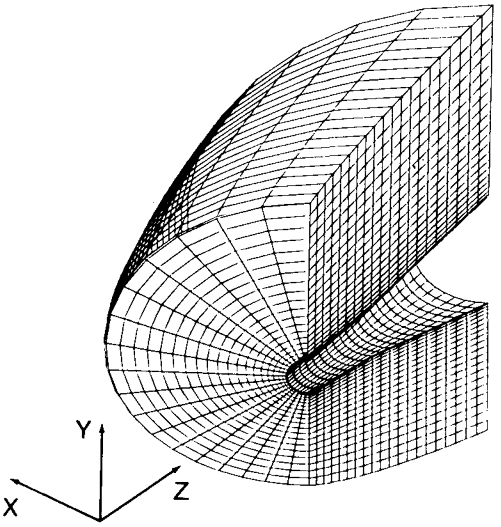

Fig. 9 Cylindrical grid on biconic by three-dimensional parabolic grid equations.

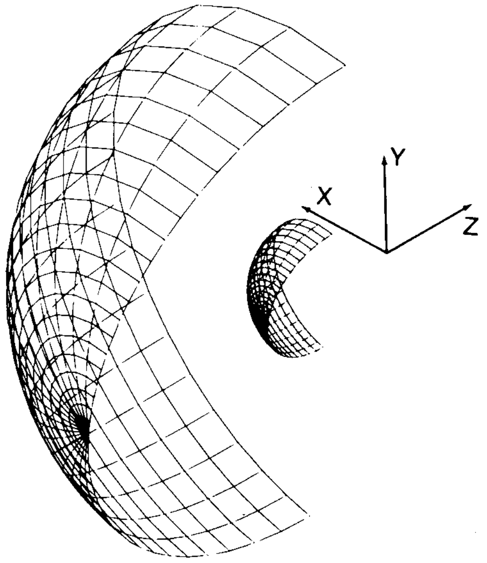

Fig. 10 Spherical grid on spheroid by three-dimensional parabolic grid equations.

generated by one-dimensional, two-dimensional, or three-dimensional grid generation equations. The three-dimensional equations would be used for a symmetry plane on a boundary, and complicates the procedure because of the multiple options, some of which must be in the marching loops. Points at corners are prescribed or generated by one-dimensional grid generation. Thus, the major complication in going to three dimensions was not the interior grid generation, but obtaining the boundary points and visualizing the results.

Results

Grids were generated using the three-dimensional parabolic procedure with very little computer time. Limitations for the three-dimensional procedure were analogous to those in two dimensions, and problems because of boundary curvature were handled in an analogous way. Three examples are presented consisting of a case in Cartesian coordinates (plate) for use in a boundary-layer solution, a case in cylindrical coordinates (aft section of a biconic) for use in an axis-normal PNS solution, and a case in spherical coordinates (spheroid with offset outer boundary) for a PNS blunt body starting solution.

A three-dimensional ($21 \times 21 \times 19$) grid for a flat plate is shown in Fig. 8. The grid was generated in Cartesian coordinates $r = (z,y,x)^T$. The outer boundary was prescribed in terms of the theoretical solution for boundary-layer thickness for a Reynold's number of one million. The leading edge singularity is shown, and causes the only problem in the grid generation if too much smoothing is implemented by prescribing a large I^+ value. A gap between the last generated $j = J\text{MAX} - 1$ grid points and the outer boundary $j = J\text{MAX}$ would occur. For the case shown, $I^+ = i + 1$ and $K^+ = k + 1$ was prescribed, and an excellent grid was generated. The grid control function of P_1 and $P_3 = -0.02$ was prescribed in the x and z directions. The grid control in the normal (y) direction was optimized according to the laminar and turbulent power law profile given in Eq. (5). The corner points at the axis center and singularity were trivially generated by the one-dimensional grid equations. The $k = 1$ surface on the z axis and the $i = 1$ surface on the x axis were generated by the three-dimensional parabolic grid equations with coefficients that essentially reduce to two-dimensional parabolic grid equations. This simple configuration does not illustrate the generality of the procedure, but provides an excellent test case for code checkout.

Biconic

A three-dimensional axis-normal ($21 \times 21 \times 19$) grid for a 10 deg/20 deg biconic is shown in Fig. 9. The grid was generated in cylindrical coordinates. The outer boundary was circular, with an offset along the y axis. The radius was arbitrarily increased axially similar to laminar growth of a boundary layer. The z points and θ points were both generated by one-dimensional grid generation with P_1 and $P_3 = -.02$. The power law profile was arbitrarily used for P_2. The corner points at $i = 1$ and $k = 1$ were generated by the one-dimensional equations. The $k = 1$ ($\theta = -\pi/2$) and $i = 1$ ($z = 0$) surfaces were generated by the three-dimensional parabolic grid equations that essentially reduce to two-dimensional parabolic equations. Extrapolation conditions were used at the $i = I\text{MAX}$ ($z = 1$) boundary, and zero gradient conditions at $k = K\text{MAX}$ ($\theta = \pi/2$). The grid was smoothed using $I^+ = i + 5$ and $K^+ = k + 1$. The smoothing at the compression surface (concave corner) cannot be seen well in the three-dimensional perspective view, but is similar to that shown for two dimensions in Fig. 4. The $i = 1$, $k = 1$, $k = K\text{MAX}$, $j = 1$, and $j = J\text{MAX}$ surfaces are shown in Fig. 9. No problems were encountered for this grid. The only problem ever encountered was identical to a two-dimensional problem with a combination of a very fine grid at a large expansion angle (convex corner). The solution to this problem also was to iterate at least twice on the coefficients, or to reduce the effect of the source terms near convex surfaces.

Spheroid

A three-dimensional ($21 \times 21 \times 19$) grid for a spheroid was generated and is shown in Fig. 10. The grid was generated in spherical coordinates. The outer boundary is spherical and the inner boundary or body is axisymmetric and varies in radius from the negative y axis to the positive y axis. The only major difference in this grid is that it is not an axis-normal grid, and the spherical coordinate singularity has been generated without any difficulty. This is more of a surface-normal grid for the solution of elliptic flow equations in a blunt nose region; the grid is not orthogonal and a local orthogonality condition could be implemented for this case.

Conclusions and Recommendations

Parabolic grid generation equations were used to successfully generate numerous two- and three-dimensional grids in terms of Cartesian, polar, cylindrical, and spherical coordinates.

Unique extensions were made in generating one-dimensional grids by accurately and efficiently solving a linear differential grid equation. Efficiency was obtained by choosing a linear grid equation which could be solved without iteration. The linear equation also admits exponential grid solutions appropriate for high Reynold's number turbulent flow, which is also solved accurately numerically by assuming an exponential solution form. Optimization of the grid to minimize flow solution truncation error was demonstrated for a power law velocity profile. This one-dimensional grid generation procedure was then extended to both two and three dimensions.

Unique extensions were made in generating two-dimensional grids in terms of Cartesian or polar coordinates using parabolic grid equations. Elliptic equations were parabolized to allow marching in an axial direction consistent with flow solutions of parabolic equations such as PNS and boundary layer. Grid-control functions which allow highly stretched grids, especially those normal to a surface, were incorporated. Thus, grid embedding and interpolation are not necessary! The parabolic grid equations are solved accurately and efficiently without iteration in most cases. The ability to smooth the grid even with surface discontinuities was demonstrated and can be controlled for concave corners. Solutions to a one-to-one mapping problem at a convex corner with a very fine grid are identified. These solutions are also appropriate for three-dimensional grids.

The unique extensions in the one- and two-dimensional grid generation were extended to three dimensions where elliptic grid generation is expensive. By marching in two directions, grids which approximate elliptic grids were generated without iteration at a cost of two tridiagonal solutions per grid line. For the axis-normal grids in this paper, only two times *I*MAX times *K*MAX tridiagonal solutions were necessary for the three-dimensional grids. In addition, the grid was only stored at three axial stations, thus saving computer memory. This has allowed grids to be calculated at each axial step in an explicit marching solution of the PNS equations where the axial step size is calculated based on stability (McCarty and Hodge[18]). The parabolic grids in this paper have been used with the boundary-layer equations.[19]

Further extension of this research is recommended in several areas. First, local orthogonality near surfaces that have been implemented by Nakamura and Edwards should be added. Second, the full parabolic grid equations should then be solved to generate surface-normal grids in addition to axis-normal grids (see Miller[16]). Third, if parabolic grid generation does not handle a particular configuration for some reason, then elliptic grid generation should be used with parabolic grid generation as an initialization procedure for the iterative elliptic equations. Finally, adaptive grid procedures should be attempted, especially when some type of space marching flow solution is used.

References

[1] Thompson, J. F., Editor, *Numerical Grid Generation,* Elsevier Science Publishing, 1982.

[2] Numerical Grid Generation Techniques, NASA CP-2166, 1980.

[3] Thompson, J. F., Thames, F. C., and Mastin, C. W., "Automatic Numerical Generation of Body-Fitted Curvilinear Coordinate System for Field Containing Any Number of Arbitrary Two Dimensional Bodies," *Journal of Computational Physics,* Vol. 15, July 1974, pp.299–319.

[4] Sorenson, R. L. and Steger, J. L., "Use of Hyperbolic Partial Differential Equations to Generate Body Fitted Coordinates," Numerical Grid Generation Techniques, NASA CP-2166, 1980, pp. 436–478.

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

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

[7] Edwards, T. A., "Noniterative Three Dimensional Grid Generation Using Parabolic Partial Differential Equations," AIAA paper 85-0485, 1985.

[8] Ghia, U., Hodge, J. K., and Hankey, W. L., "An Optimization Study for Generating Surface-Oriented Coordinates for Arbitrary Bodies in High RE Flow," AFFDL-TR-77-117, 1977.

[9] Hodge, J. K., Stone, A. L., and Miller, T. E., "Numerical Solution for Airfoils near Stall in Optimized Boundary-Fitted Curvilinear Coordinates," *AIAA Journal,* Vol. 17, May 1979, pp. 458–464.

[10] Roscoe, D. F., "New Methods for the Derivation of Stable Difference Representations for Differential Equations," *Journal Inst. Maths Applics,* Vol. 16, 1975, pp. 291–301.

[11] Hodge, J. K., "Numerical Solution of Incompressible Laminar Flow About Arbitrary Bodies in Body-Fitted Curvilinear Coordinates," Ph.D. dissertation, Mississippi State Univ., 1975.

[12] Shanks, S. P., "Numerical Simulation of Viscous Flow About Submerged Arbitrary Hydrofoils Using Non-Orthogonal, Curvilinear Coordinates," Ph.D. dissertation, Mississippi State Univ., 1977.

[13] Middlecoff, J. F. and Thomas, P. D., "Direct Control of the Grid Point Distribution in Meshes Generated by Elliptic Eqs.," AIAA Paper 79-1462, 1979.

[14] Thomas, P. D., "Numerical Generation of Composite Three Dimensional Grids by Quasilinear Elliptic Systems," *Numerical Grid Generation,* edited by J. F. Thompson, Elsevier Science Publishing, 1982, pp. 667–686.

[15] Ghia, U. and Ghia, K. N., "Boundary-Fitted Coordinates for Regions With Highly Curved Boundaries and Reentrant Boundaries," Numerical Grid Generation Techniques, NASA-CP-2166, 1980, pp. 295–306.

[16] Miller, S. G., "Numerical Grid Generation for Parabolic Partial Differential Equations Using Marching Techniques," M. S. Thesis, Air Force Institute of Technology, 1985.

[17] Sorenson, R. L. and Steger, J. L., "Grid Generation in Three Dimensions by Poisson Equations With Control of Cell Size and Skewness at Boundary Surfaces," *Advances in Grid Generation,* edited by U. Ghia, ASME, FED-Vol. 5, 1983, pp. 181–187; also AFWAL-TR-83-3129.

[18] McCarty, R. L. and Hodge, J. K., "Nonequilibrium Parabolized Navier-Stokes Solution Over a Noncatalyic-Catalytic Discontinuity," to be published as Air Force Institute of Technology Technical Report, 1987.

[19] Chen, A. J., "Numerical Solution of Laminar and Turbulent Boundary-Layer Equations. Including Transition, and Experimental Study of a Flat Plate With a Delta Wing at Incidence," M.S. Thesis, Air Force Institute of Technology, 1986.