JOURNAL OF PROPULSION AND POWER
Vol. 9, No. 1, Jan.-Feb. 1993

Grid Generation for Three-Dimensional Turbomachinery Geometries Including Tip Clearance

Anton H. Basson,* Robert F. Kunz,† and Budugur Lakshminarayana‡
Pennsylvania State University, University Park, Pennsylvania 16802

An efficient technique for the generation of structured grids for viscous flow computations in turbomachinery blade rows (two- and three-dimensional), and a specialized embedded *H*-grid for application, particularly to tip clearance flows, are presented. The grid generation technique uses a combination of algebraic and elliptic methods to obtain smooth grids while maintaining strict control over grid spacing and orthogonality at domain boundaries. A geometric series scheme is used to distribute boundary points. Algebraically generated layers next to boundaries are used, thus excluding highly clustered regions from the elliptic generation procedure's domain. The computational efficiency of the elliptic generation procedure is greatly enhanced by the application of the minimal residual method. The embedded *H*-grid topology provides good resolution of tip clearance effects. This topology requires only minor modifications to flow solvers developed for conventional *H*-grids. The results obtained with an embedded *H*-grid are compared to those obtained using a thin-tip approximation. A linear compressor cascade with tip clearance was used as a test case. Both grid topologies capture the dominant flow structures associated with the leakage flow. The embedded *H*-grid provided better quantitative agreement with the experimental results.

Introduction and Overview of Present Technique

THE generation of structured grids for the computation of viscous flows in conventional turbomachinery blade rows, is considered in this article. Though particular attention is paid to cases with tip clearance, the grid generation methodology is applicable to other blade rows as well. The generation of an appropriate grid is an essential element of an accurate flow solution process. Adequate resolution of important flow phenomena, such as boundary layers and wakes, are crucial to obtaining physically representative flow solutions.

Structured grids have the advantage that connectivity arrays are trivial,[1] thus simplifying the calculation of higher derivatives encountered in viscous flow computations. The development presented below employs *H*-grids, but most of the techniques are also applicable to *C*- and *O*-grids. The approach followed here for the generation of grids for three-dimensional turbomachinery blade rows (as also used by Beach[2]) is to generate several "blade-to-blade" grids on axisymmetric surfaces, and then interpolate between these grids to define the three-dimensional grids. The near axisymmetry of turbomachine blade rows and flows make this approach totally adequate and it is very economical. Interpretation of a flow computation's results is also simplified by constructing a three-dimensional grid out of axisymmetric grid "slices."

Four particular aspects are considered: 1) the method used to distribute the boundary points; 2) the method used to generate a two-dimensional grid; 3) acceleration of the iteration process involved in generating the grid; and 4) the use of the embedded *H*-grid topology to resolve tip clearance flows.

Presented as Paper 91-2360 at the AIAA/SAE/ASME 27th Joint Propulsion Conference, June 24–26, 1991; received July 19, 1991; revision received Feb. 3, 1992; accepted for publication March 3, 1992.

*Graduate Student, Department of Aerospace Engineering; currently an Engineer at the Bureau for Mechanical Engineering, Stellenbosch, 7600, South Africa. Member AIAA.

†Graduate Student, Department of Aerospace Engineering; currently an Engineer at Knolls Atomic Power Laboratory, General Electric Company, Schenectady, NY, 12301. Member AIAA.

‡Evan Pugh Professor of Aerospace Engineering. Fellow AIAA.

The grid generation process is greatly simplified by determining the locations of grid points on the boundaries prior to the generation of the internal grid. Elliptic grid generation schemes can then be formulated as Dirichlet problems. Sorenson[3] uses a weighted equidistribution scheme to distribute boundary points. Other common approaches are the use of cosine[4] and hyperbolic tangent[2] functions. A new distribution scheme based on a geometric series, which provides better control than the cosine and hyperbolic tangent schemes, (but is easier to use than equidistribution schemes) is presented here.

To maintain the accuracy of difference approximations of the derivatives in flow solvers, structured grids have to be smooth, i.e., the local radius of curvature of a grid line must be large compared to the local length of grid segments. But the smoothest grids will, in general, not provide sufficient grid resolution in regions of large gradients of flow properties, e.g., at leading and trailing edges, and in boundary layers and wakes. Another consideration is orthogonality. In turbulent flow analyses, the application of turbulence models are greatly simplified when the grid lines intersect the solid boundaries at right angles. Although orthogonality of the grid over the whole domain would be advantageous, it is not possible to enforce orthogonality in periodic grids for blade rows with nonzero inlet or outlet angles.

The boundary-value problem nature of grid generation for internal flows (the domain is completely enclosed by the boundaries) and the smoothness requirement, make elliptic grid generation methods a natural choice.[1] Accordingly, such methods are used in a number of grid generation schemes.[2,3,5,6] The smoothing properties are modified in different ways to obtain the required resolution in regions of large flow gradients. Sorenson[3] and Shieh[6] used the source terms in the Poisson equations to control the grid spacing and orthogonality near the boundaries. In the experience of the present authors and other workers,[2] the amount of control that can be obtained by the source terms is inadequate for cascade grids with thick leading and trailing edges. The extremely small grid spacings at the solid boundaries required for computations using low Reynolds number turbulence models, can also not be reliably produced solely through the use of source terms.

Some researchers totally abandon the use of a single-block grid structure,[7] deeming their use for turbomachinery geometries impractical. With suitable modifications to elliptic grid generators, however, suitable grids can be generated. One approach to overcome the inadequacies of elliptic grid generators, while retaining their useful properties, is to combine the elliptic methods with algebraic methods. Beach[2] generates an algebraic grid interactively, and then uses the elliptic grid generation equations to smooth selected regions. Another approach is followed in the method presented here. In this approach algebraic methods are only used to generate layers of grid points near solid boundaries, while the remainder of the grid is generated using the elliptic grid generation approach. The elliptic generator is also implemented to smooth the interface between the algebraic and elliptic grid regions.

Another disadvantage of elliptic grid generation methods, is the computational effort required. The nonlinearity of the equations give rise to the use of iterative solution techniques. Additionally, the generation of a grid on a new geometry typically requires a number of trials and revisions before a suitable grid is obtained. Therefore, the time required to generate a grid for a particular set of parameters is important. Full matrix inversion for large grids is impractical, while standard alternating direction implicit (ADI) or line relaxation schemes converge slowly. The minimal residual method is therefore employed in the scheme presented here to reduce computation time. Very significant CPU time improvements are obtained in this way.

The grid generation approach introduced above is applicable to a wide array of grid generation topologies and applications. The resolution of tip clearance flows, however, require specialized grids. The abrupt transition from the blade passage to the tip gap, with the sharp corners at the blade tip and the large gradients in flow properties, limits the usefulness of typical *H*- or *C*-grids. A commonly used approach, sometimes called the "thin blade approximation," is to cusp the blade tip.[8,9] This approach is reasonable for thin blade tips and is suitable for modeling the gross effects of tip clearance flows, though the tip geometry is not modeled precisely. Nonsmooth grid lines in the tip region also locally reduce the accuracy of the solution.

Better modeling of the tip gap has been obtained by embedding an *O*-grid within the main grid in the tip gap.[2] This method provides good resolution and accurate modeling of the actual geometry, but introduces singularities in the grid where the innermost line of the *O*-grid doubles back near the leading and trailing edges. The solution algorithm for such an embedded *O*-grid must further be formulated for the block-structured nature of the grid. To overcome these difficulties, an embedded *H*-grid topology is presented. This topology eliminates the singularities, while retaining the advantages of good resolution and accurate modeling of the geometry. In addition, the embedded *H*-grid topology does not require a block-structured solution algorithm.

Distribution of Boundary Points

The scheme used to distribute the grid points on the boundaries has a profound influence on the control over the accuracy with which the grid models the geometry. For cascade geometries, the leading and trailing edges are particular areas where control over the distribution of points is required. Highly cambered cascades, e.g., turbine nozzles, also require control over the spacing of points over the remainder of the blade profile to resolve large gradients that occur there. The most flexible approach for distributing boundary points, is the use of an equidistribution scheme with suitably chosen weighting functions. The use of these methods, however, either requires very detailed inputs or complex coding logic which is difficult to generalize. A much simpler approach is to use trigonometric functions to distribute the points. Such approaches only require the specification of one parameter at either end, e.g., the grid spacing. The grid stretching along the boundary is completely determined by the function used in the distribution scheme. These approaches, therefore, suffer from a lack of control over the grid spacing away from the end points and can lead to excessively large grid spacings.

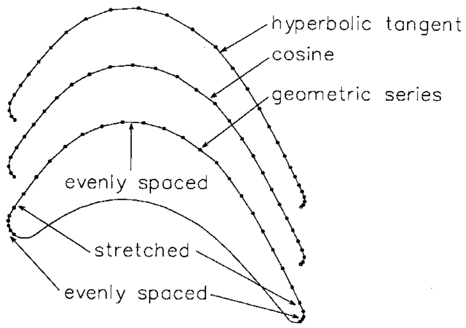

Fig. 1 Comparison of distribution schemes.

The geometric series distribution scheme provides more control than the trigonometric and hyperbolic distribution schemes, but is much easier to control than equidistribution schemes. The basic approach is illustrated in Fig. 1 for the distribution of points on one surface of a highly cambered turbine blade. In this figure, it is seen that the geometric series distribution scheme provides good leading- and trailing-edge resolution, while also providing smaller grid spacings at midchord than the trigonometric schemes. The parameters for this scheme, for a given boundary length and total number of grid points, are 1) the grid spacing at each end point; 2) the number of grid segments that have this constant spacing; 3) and the stretching factor (ratio of the length of two consecutive grid segments). The distribution procedure first allocates the boundary points that are uniformly spaced at the ends of the boundary, followed by the points where the spacing is increased by the stretching factor. At a certain stage the point spacing will have increased sufficiently to cover the remainder of the boundary by equally spaced points. This approach can easily be used to obtain a fine resolution of the leading and trailing edges, and directly control the stretching of the grid spacing. The grid spacing away from the end points can be controlled by means of the stretching factor. This ability to control the grid spacing away from the end points independently from the spacings at the end points, is the main advantage of the geometric series distribution scheme.

For a given set of parameters, there is obviously a minimum value for the stretching parameter. This minimum value can be derived by considering the sequence of grid lengths as a geometric series

$$S = a + \alpha a + \alpha^2 a + \alpha^3 a + \cdots + \alpha^{(n-1)}a + \alpha^{(m-1)}b + \cdots + \alpha^3 b + \alpha^2 b + \alpha b + b \tag{1}$$

where S is the length of the boundary over which $n + m$ stretched grid segments ($n + m + 1$ points) must be distributed, a and b are the end spacings, and α is the stretching factor. These parameters are then related as follows:

$$m - n = [\log(a/b)]/\log(\alpha)$$

$$\alpha^n = \frac{(\alpha - 1)S + (a + b)}{[a + \alpha^{(m-n)}b]} \tag{2}$$

Generation of the Algebraic Layers

The use of algebraic layers is the essential distinction between the grid generation method presented here and the purely elliptic grid generation schemes. In elliptic grid generation schemes, the inherent smoothing properties of the Laplace operator are modified by the source terms of the Poisson equation to obtain the required grid spacing and orthogonality at the boundaries. The strength of the source terms at a particular location on the boundary is directly related to how much the required grid deviates from the "smoothest" grid, as would be produced by the use of a pure Laplace operator. Highly clustered grids required by the viscous flow analysis procedures, combined with the effects of the leading and trailing edges, large stagger, or large camber, require large source terms to produce an adequate grid. These source terms, however, adversely affect the numerical stability of the grid generation scheme, and give rise to local regions of severe grid skewness. To illustrate this, a grid was generated for the turbine nozzle shown in Fig. 2. Figure 3 shows the leading- and trailing-edge grids obtained with an H-grid version of the GRAPE[3,10] grid generation code. Note that the tendency of the grid lines to bundle close to the leading and trailing edge was not prevented by the use of source terms. In addition, the smallest grid spacing normal to the wall for which the elliptic generation scheme converged, was about 50 times larger than that which would be required by a low Reynolds number turbulence model.

In the present work, the deficiencies in the elliptic generation scheme are overcome by the introduction of algebraically generated layers of grid cells near the boundaries (Fig. 4). By using algebraic methods to generate the grid cells where most of the stretching occurs (in the direction normal to the boundary), the elliptic generator is required to maintain a much larger minimum boundary spacing. In this way, the strength of the source terms in the Poisson equation are significantly reduced, leading to much smoother meshes and improvements in the numerical stability of the elliptic grid generation scheme. The enhanced numerical stability of the elliptic scheme makes it possible to improve the convergence rate of the elliptic generator even further by employing convergence acceleration techniques.

The algebraically generated layers near the boundaries provide better control over grid orthogonality near the walls. In regions of the algebraic layers where grid lines that are orthogonal to the boundary would overlap, (e.g., near the lead-

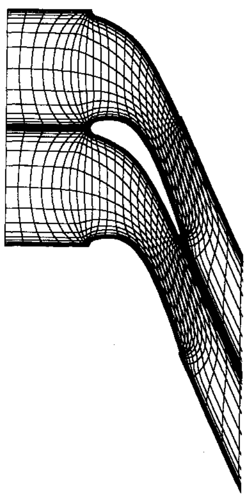

Fig. 2 Turbine nozzle grid (every second line shown).

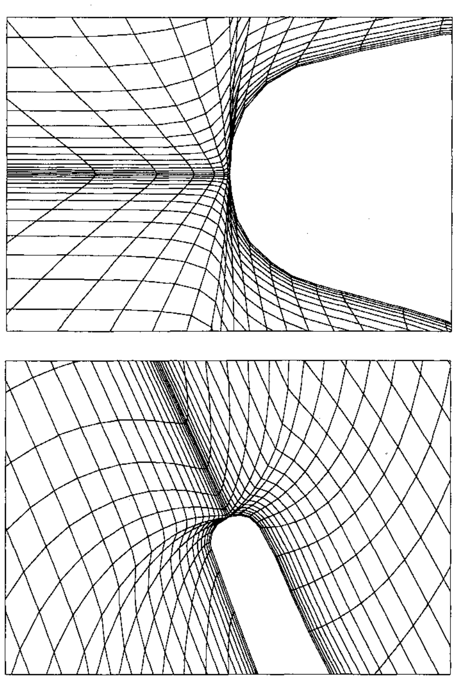

Fig. 3 Nozzle leading- and trailing-edge grids produced by purely elliptic generation.

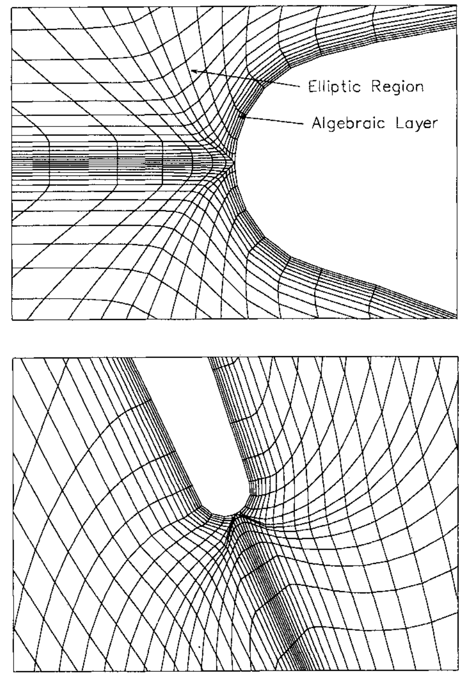

Fig. 4 Nozzle leading- and trailing-edge grids after elliptic generation and before smoothing of algebraic-elliptic interface.

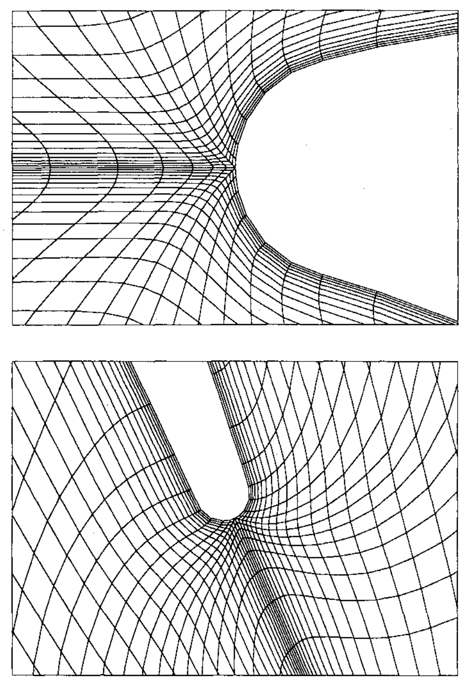

Fig. 5 Nozzle leading- and trailing-edge grids after smoothing of algebraic-elliptic interface.

ing and trailing edges) the grid lines are modified by using Bezier curves. With the use of algebraic layers, the inability of the elliptic generator to strictly control orthogonality does not affect the grid near the walls. The interface between the algebraic and elliptic regions of the grid, as shown in Fig. 4, is, however, not acceptable. The grid in these regions is therefore smoothed after the generation of the elliptic region. The requirement for a posteriori smoothing also relaxes the requirements for the algebraic generation procedure, as this final smoothing also improves this part of the grid. Figure 5 shows the final grid near the leading and trailing edges of the turbine nozzle used as an example here.

Generalized Minimal Residual Method

The numerical solution of the elliptic grid generation equations (which are nonlinear) requires the repeated solution of a set of simultaneous linear equations. For typical grids, the system of equations is too large to use full matrix inversion. The generalized nonlinear minimal residual method has been shown to improve the convergence rate of iterative schemes.[11,12] In the present work, the linear form of this method was implemented in the grid generation procedure as an alternative to traditional methods, such as line relaxation. Numerical experiments have shown that the nonlinear form of the minimal residual method does not perform better than the linear form in this application, while the nonlinear form incurs significant increases in complexity.

The form of the generalized minimal residual method to be applied to the elliptic grid generation equations will briefly be derived here to introduce the essential concepts and notation. A detailed development is given by Huang, et al.[12] An explicit iterative solution scheme can be cast in a quasi-time-dependent form. The resulting discretized equation takes the following form:

$$\phi_{\tau+1} = \phi_\tau + \Delta\tau(L\phi_\tau + f) \tag{3}$$

where ϕ is the dependent variable, τ is the pseudotime variable, L is the discrete approximation of the linear difference operator and f is the forcing function. The residual is defined as follows:

$$r_\tau = (\phi_{\tau+1} - \phi_\tau)/\Delta\tau \tag{4}$$

Considering the linearity of the operator L, straightforward Taylor series expansions in τ can be applied to obtain expressions for $\phi_{\tau+1}$ and $r_{\tau+1}$. The essence of the generalized form of the minimal residual method, is to consider the time steps, $\Delta\tau$, in these expansions to be composed of smaller, but not necessarily equal, time steps. Accordingly, these expansions can be written

$$\phi_{\tau+1} = \phi_\tau + \omega_1 r_\tau + \omega_2 L(r_\tau) + \omega_3 L^2(r_\tau) + \cdots \tag{5}$$

$$r_{\tau+1} = r_\tau + \omega_1 L(r_\tau) + \omega_2 L^2(r_\tau) + \omega_3 L^3(r_\tau) + \cdots \tag{6}$$

The basic premise of the minimum residual method is that the optimum value for each ω_i can be determined from the condition that the residual at the next time step, $r_{\tau+1}$, must be minimized. To minimize the L^2 norm of the residual, the derivative with respect to each ω_i is taken and is set equal to zero. The following expression is thus obtained:

$$\int_\Omega \left\{ [r_\tau \cdot L^i(r_\tau)] + \sum_j \omega_j [L^i(r_\tau)] \cdot [L^j(r_\tau)] \right\} = 0 \tag{7}$$

where Ω is the solution domain.

Equation (7) describes a system of linear equations that can be solved for the ω_i. These ω_i can then be substituted into Eq. (5) to give the values of ϕ that will minimize the residual at the next time step, in other words, give the optimum convergence rate.

The application of the minimal residual method to the grid generation equations will now be demonstrated by using only the x-field equation. The development for the corresponding y-equation is similar. Using Sorenson's[3] notation, the x-equation is as follows:

$$\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} = -J^2(Px_\xi + Qx_\eta) \tag{8}$$

where ξ and η are the curvilinear coordinates, α, β, and γ are terms containing first derivatives of x and y, J is the Jacobian of the transformation, P is the source term for the Poisson equation for ξ, Q is the corresponding source term for η, and the subscripts are used to indicate partial derivatives.

For the application of the linear form of the minimal residual method, α, β, γ, and the right side of Eq. (8) are taken to be independent of the solution, but do not have to be uniform across the domain. The equivalent time-dependent form for an explicit iteration scheme for Eq. (8) is

$$\frac{\partial x}{\partial \tau} = Ax_{\xi\xi} - 2Bx_{\xi\eta} + Cx_{\eta\eta} - F_x \tag{9}$$

where

$$A = \alpha\left(\frac{\alpha}{\Delta\xi^2} + \frac{\gamma}{\Delta\eta^2}\right)^{-1}$$

$$F_x = -J^2(Px_\xi + Qx_\eta)\left(\frac{\alpha}{\Delta\xi^2} + \frac{\gamma}{\Delta\eta^2}\right)^{-1}$$

with similar expressions for B and C.

By definition, the right side of Eq. (9) is the residual. Using second-order difference approximations, the linear operator L, takes the following form:

$$\begin{aligned} L(\phi) = {} & A(\phi_{i+1,j} - 2\phi_{i,j} + \phi_{i-1,j}) - 2B\tfrac{1}{4}(\phi_{i+1,j+1} \\ & - \phi_{i+1,j-1} - \phi_{i-1,j+1} + \phi_{i-1,j-1}) + C(\phi_{i,j+1} \\ & - 2\phi_{i,j} + \phi_{i,j-1}) \end{aligned} \tag{10}$$

The $L^i(\phi)$ terms in Eq. (7) are obtained by repeated application of Eq. (10)

$$L^2(\phi) = L[L(\phi)] \tag{11}$$

Equations (10) and (11), applied to the residual field, give the coefficients of the linear system in Eq. (7). The solution of this linear system is the ω_i to be used in Eq. (5), with x replacing ϕ. The number of ω_i's used has to be chosen by balancing memory requirements and computational effort. Numerical experiments indicated that the use of three ω's, in

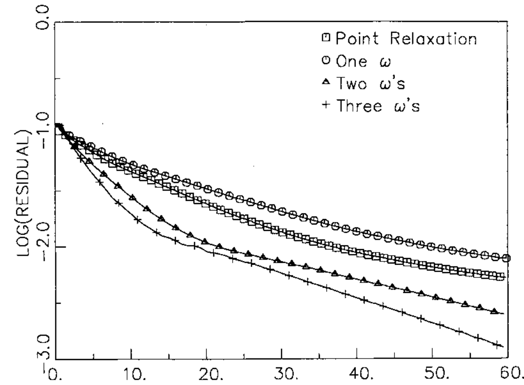

Fig. 6 Comparison of convergence histories of point relaxation and minimal residual method using 1, 2, or 3 ω's.

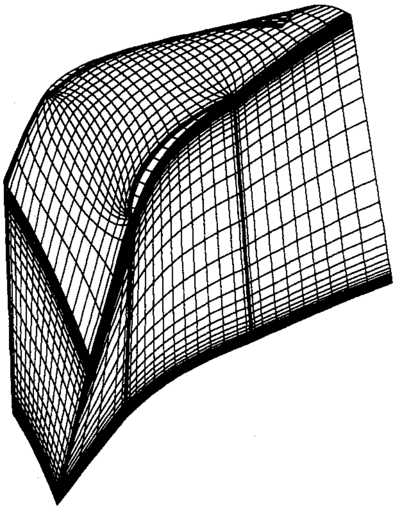

Fig. 7 Example of a three-dimensional grid.

general, gives close to the maximum convergence rate, while requiring storage of only three extra variables per grid point.

Figure 6 demonstrates the improvement in convergence rate that can be obtained by using the generalized minimal residual method compared to point relaxation for the generation of the grid in Fig. 2. The generalized minimal residual method further obviates the need to experimentally determine an optimum relaxation factor required by conventional solution techniques.

Generation of Three-Dimensional Grids for Cascades

The well-defined geometry of turbomachine blade rows permits the generation of three-dimensional grids by mapping two-dimensional grids onto cylindrical, conical, or axisymmetric surfaces. Developed onto a flat plane, the generation of these two-dimensional grids reduces to the familiar two-dimensional cascade grid case. The methods described above are employed to generate such two-dimensional blade-to-blade grids. With fixed values for the various control parameters, the grid generation scheme will give similar blade-to-blade grid surfaces at different spanwise positions.

A two-dimensional grid need not be generated for each grid plane that will be required, in fact this would be less likely to produce a smooth grid.[2] Only a few two-dimensional grids need to be generated. Cubic splines are then used to find other grid points by interpolation, resulting in a smooth grid. The geometric series distribution scheme described above is used to determine the appropriate spanwise spacing of the blade-to-blade grid surfaces. Figure 7 gives an example of a grid generated with this procedure.

Grids for Blade Rows with Tip Clearance

Blade rows with tip clearance present a number of additional grid generation complications. Although the tip gap is small compared to the major flow passage dimensions, tip clearance flows can influence about 20% (30% in low-aspect ratio blade rows) of the blade passage. Therefore, it is important to adequately resolve the details of the flow through the tip gap. To obtain adequate spatial resolution for these geometries (while maintaining compatibility with flow solvers for simple grids) embedded *H*-grids, such as used by Moore and Moore,[13] can be generated using the present procedures. The grid in the computational space is shown in Fig. 8. In the region "below" the tip gap, the blade profile is mapped into the rectangular region where dashed lines are shown in Fig. 8. The computational space then has the usual inlet, outlet, periodic, and solid boundaries encountered with *H*-grids, but two "new" solid boundaries are encountered near the leading and trailing edges. In the tip gap region, the computational space includes the dashed lines shown in Fig. 8. No solid boundaries therefore appear in the blade-to-blade grid surfaces. Figures 9 and 10, respectively, show leading-edge details of the resulting grids above and below the tip gap for a C4 cascade.

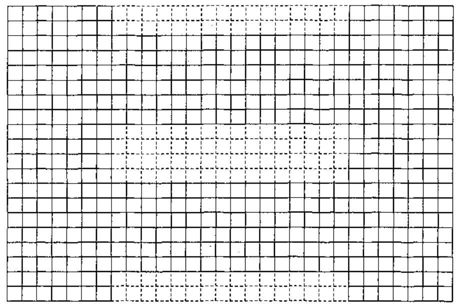

Fig. 8 Computational space for embedded *H*-grid topology with dashed lines only used in tip gap (two blade passages shown).

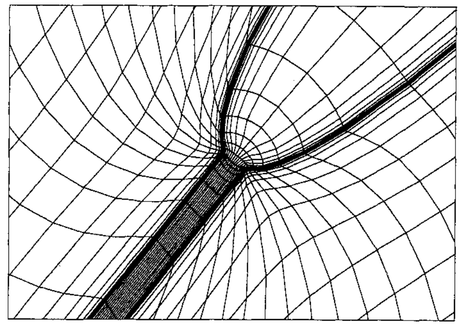

Fig. 9 Leading-edge grid detail for embedded *H*-grid for a C4 cascade. Grid in tip gap.

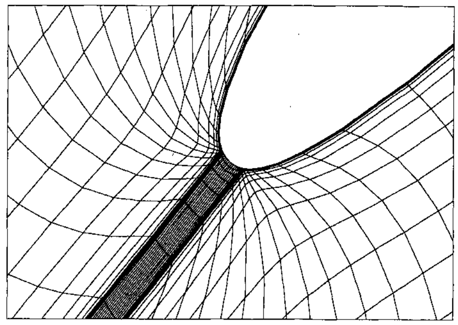

Fig. 10 Leading-edge grid detail for embedded *H*-grid for a C4 cascade. Grid below tip gap.

To so-called "mesh singularities" at leading and trailing edges obtained with standard *H*-grids do not arise in this grid topology. Accordingly, the embedded *H*-grid is particularly well-suited to thick blades. Additionally, this topology is well-suited to flow solution algorithms that are not formulated for block structured grids, because the *H*-grid connectivity pattern is retained. Also, the clustered region emanating from the trailing edge, similar to the region ending at the leading edge shown in Fig. 10, provides improved wake resolution. For thick blade applications (i.e., turbine blades), utilization of an embedded *H*-grid topology may increase the required number of grid points in the blade-to-blade direction, since a larger percentage of grid points lie in the vicinity of the leading and trailing edges, as compared to a standard *H*-grid.

Comparison of Computations on Embedded *H*-Grid and Pinched Tip *H*-Grid

A three-dimensional Navier-Stokes solver, developed for turbomachinery *H*-grids, was modified to accommodate the embedded *H*-grid topology. Briefly, the technique employs a

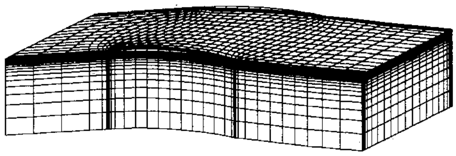

Fig. 11 Three-dimensional embedded *H*-grid (every second grid line shown).

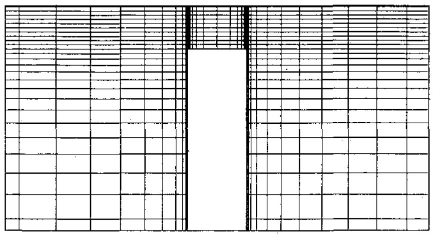

Fig. 12 Embedded *H*-grid crossplane tip detail at midchord.

Runge-Kutta numerical procedure and a low-Reynolds number form of the k-ε turbulence model. Second-order central differences are used to discretize the spatial derivatives which appear in the inviscid and viscous flux vectors of the Navier-Stokes and turbulence transport equations. For the subsonic cases considered here, a fourth-difference artificial dissipation operator is added to the discrete equation set to provide additional damping required for stability. An inviscid flux Jacobian eigenvalue scaling and a near-wall local velocity scaling are applied to the artificial dissipation operators to enhance stability and to help reduce corruption of physical diffusion by the artificial dissipation operators in near-wall and near-wake regions. Further details are available in Kunz and Lakshminarayana.[8]

To minimize the modifications required to the code, the grid points that would normally only be used in the tip gap (indicated by the dashed lines in Fig. 8), were retained throughout computational space. The following minor modifications to the code were then required to accommodate the embedded *H*-grid topology:

1) A block preprocessor is called, which applies Dirichlet boundary conditions along blade surfaces and the blade tip.

2) For stability, local time steps are set to zero at each iteration for grid points on and within the blades.

3) Another boundary condition routine is called that applies appropriate boundary conditions for pressure and density on the blade surfaces and the blade tip.

4) A block postprocessor is used to set the transport variables within the blades to values which allow convenient graphics postprocessing.

An extensive experimental investigation into tip clearance effects in a C4 cascade by Lakshminarayana and Horlock[14,15] was chosen to compare the results obtained with a pinched tip *H*-grid and an embedded *H*-grid. This test case was chosen because the effects of tip clearance could be isolated from end-wall effects. In the experimental setup, the end-wall effect was eliminated by splitting the blades of a two-dimensional cascade at midspan. The equivalent tip gap for each half-blade row is then half of the spanwise gap. In this way the end-wall boundary is replaced by a symmetry boundary located midway in the gap between the blade halves. The blade chord was 152 mm and the gap/chord ratio was 4%.

The number of grid lines used for both the grid topologies is 69 (streamwise) × 59 (pitchwise) × 35 (spanwise). This gives about 140,000 grid points. In the tip gap, 11 grid lines from the blade tip to the symmetry plane were used. The embedded *H*-grid had 17 grid lines across the blade tip in the pitchwise direction. Figure 11 shows a three-dimensional view of the embedded *H*-grid. The spanwise clustering of the grids was limited to the tip region as this was the only region of interest. Figures 9 and 10 show the leading- and trailing-edge details of the embedded *H*-grid used. Figures 12 and 13, respectively, show the grid in a plane at midchord for the embedded *H*-grid and the pinched tip *H*-grid.

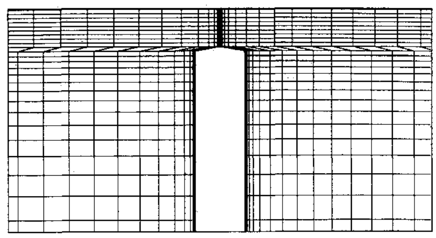

Fig. 13 Pinched tip *H*-grid crossplane tip detail at midchord.

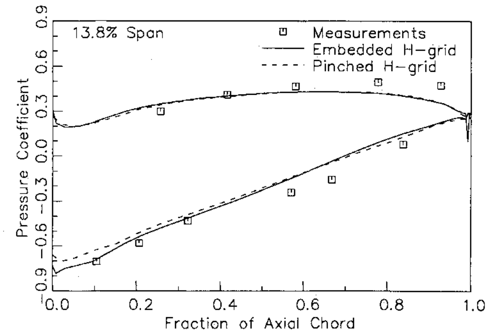

Fig. 14a Comparison of blade surface pressure coefficients, $(p - p_{in})/(\rho q_{in}^2)$, at 13.8% span below blade tip (p, ρ, q = pressure, density, velocity magnitude).

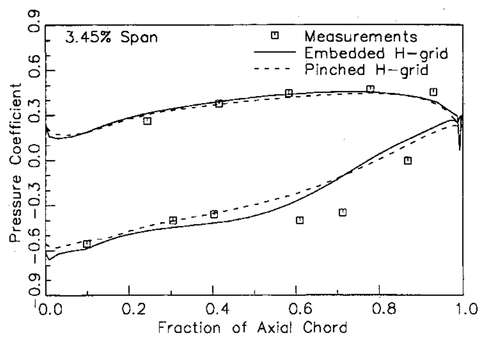

Fig. 14b Comparison of blade surface pressure coefficients, $(p - p_{in})/(\rho q_{in}^2)$, at 3.45% span below blade tip (p, ρ, q = pressure, density, velocity magnitude).

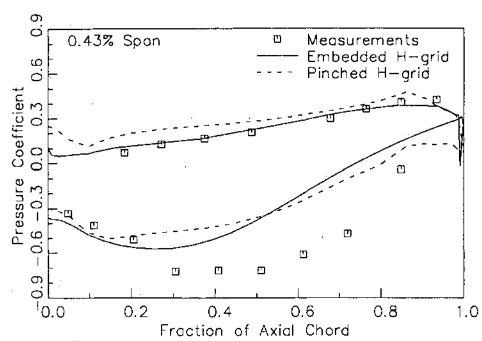

Fig. 14c Comparison of blade surface pressure coefficients, $(p - p_{in})/(\rho q_{in}^2)$, at 0.43% span below blade tip (p, ρ, q = pressure, density, velocity magnitude).

Three aspects of the solutions for the two grid topologies were selected for comparison to the experiments, i.e., blade surface pressure coefficient distributions, outlet dynamic head distributions, and outlet pitchwise flow angle distributions. The pressure coefficient distributions were chosen as a measure of the blade unloading. Both the dynamic pressure and flow angle serve as indicators of the leakage vortex strength and location. The outlet flow angle is also an indicator of the

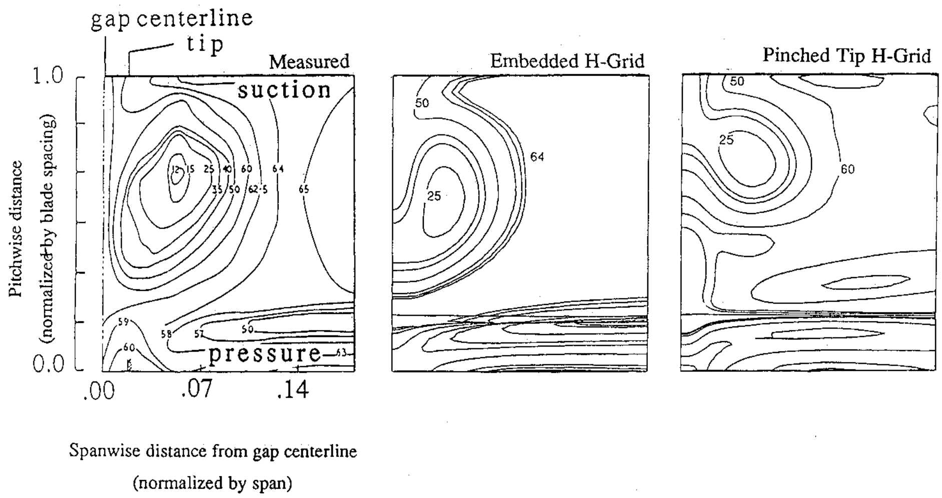

Fig. 15 Comparison of dynamic pressure contours, $(\rho q^2/\rho q_{in}^2) \times 100$ (ρ, q = density, velocity magnitude).

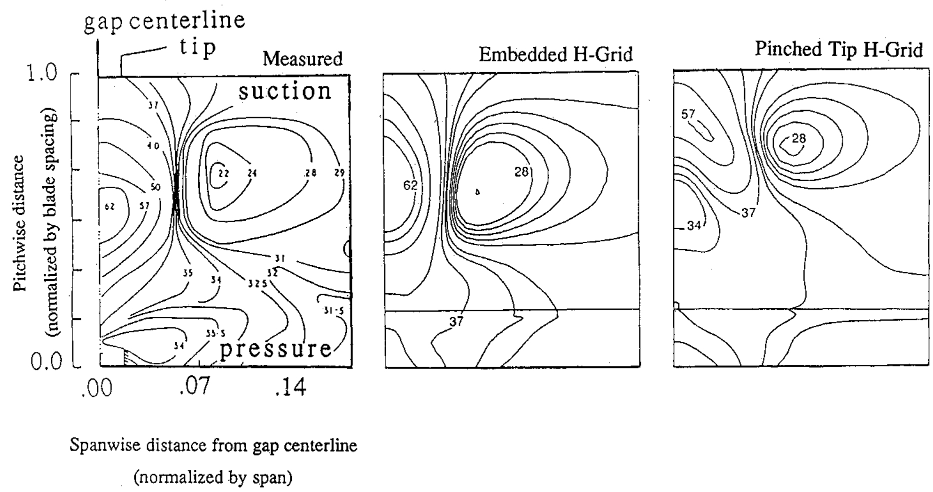

Fig. 16 Comparison of pitchwide flow angle contours, $\tan^{-1}(v/u)$ in degrees (u, v = Cartesian axial and pitchwise velocity components).

pressure rise through the blade row and the three-dimensionality of the flow. The computed results are compared to experimental measurements in Figs. 14–16.

Figure 14 compares the blade surface pressure coefficient distributions at three spanwise locations, i.e, 13.8, 3.45, and 0.43% span below the blade tip. The differences between results obtained with the two grid types are, as expected, the largest near the blade tip. At 13.8% below the tip, agreement with the measurements is good and there is little difference between the results for the two grid topologies. At the location closest to the tip, the embedded H-grid produces better results on the pressure side. In both cases, the suction side unloading near the tip is not accurately captured, probably because only 33 grid points were used in the streamwise direction from the leading to trailing edge. The embedded H-grid does, however, exhibit more of the typical downstream shift of the suction peak near the tip gap.

Figures 15 and 16, respectively, give comparisons of computed and measured contours of local dynamic pressure and pitchwise flow angle at a measurement plane located half a chord axially downstream of the trailing edge. Figure 15 indicates that the location of the tip vortex was captured reasonably accurately on both grid topologies, the size of the vortex is captured more accurately by the embedded H-grid. Although both grid topologies gave qualitatively good estimations of outlet flow angle distributions, Fig. 16 shows that the embedded H-grid captured the strength of the vortex structure better than the pinched tip H-grid. The results obtained with the embedded H-grid exhibit the extent and magnitude of the overturning (28 deg) and underturning (62 deg) associated with the leakage vortex.

The accurate computation of the vortex location and strength is essential in assessing the overall aerodynamic efficiency and determining the three-dimensional, unsteady flowfield that would enter downstream blade rows. Although the pinched tip H-grid provides acceptable engineering approximations of the tip clearance physics for the compressor cascade considered here, the embedded H-grid topology apparently provides a somewhat more accurate estimation of the vortex strength and location. Because the embedded H-grid is inherently more capable of resolving the flow in the tip gap itself, its use is more generally applicable.

Concluding Remarks

An efficient technique for the generation of structured grids has been developed. The use of a combination of algebraic and elliptic methods makes it possible to obtain smooth grids while maintaining strict control over grid spacing and orthogonality near the domain boundaries. The geometric series distribution scheme provides more control over the positioning of boundary points than hyperbolic tangent or cosine distribution schemes.

The computational efficiency of the elliptic generation procedure is greatly enhanced by the application of the minimal residual method. The use of this method further obviates the need to experimentally determine the optimum relaxation factors for the iteration scheme.

The embedded H-grid topology, which provides good resolution of tip clearance effects, has been shown to produce better results than a H-grid using the thin tip approximation. The embedded H-grid topology provides improved resolution of the flow inside the tip clearance itself, while its use requires only minor modifications of flow solvers developed for conventional H-grids.

Although a pinched tip H-grid provides acceptable engineering approximations of the tip clearance physics, the embedded H-grid topology apparently provides a more accurate estimation of the vortex strength and location. Because the embedded H-grid is inherently more capable of resolving the flow in the tip gap itself, its use is more generally applicable.

Acknowledgments

A. H. Basson was supported by the Bureau for Mechanical Engineering of the University of Stellenbosch, South Africa. R. F. Kunz was supported by a United States Army Research Office fellowship, through Grant DAAL03-86-G-0044, monitored by T. L. Doligalski. The contributions of G. Dulikravich are gratefully acknowledged. The authors also acknowledge the National Science Foundation supported Pittsburgh Supercomputer Center (Grants MSM 860009 P and CBT 90015 P) as well as the National Aerodynamic Simulation facility at NASA Ames Research Center for providing computer resources.

References

[1]Eiseman, P. R., and Erlebacher, G., "Grid Generation for the Solution of Partial Differential Equations," NASA CR-178365, ICASE Rept. 87-57, 1987.

[2]Beach, T. A., "An Interactive Grid Generation Procedure for Axial and Radial Flow Turbomachinery," AIAA Paper 90-0344, Reno, NV, Jan. 1990.

[3]Sorenson, R. L., "A Computer Program to Generate Two-Dimensional Grids About Airfoils and Other Arbitrary Shapes by the Use of Poisson's Equation," NASA TM-81198, 1980.

[4]Dulikravich, G., private communication, Dept. of Aerospace Engineering, Pennsylvania State Univ., University Park, PA.

[5]Thompson, J. F., Warsi, Z. U. A., and Mastin, W. C., *Numerical Grid Generation, Foundations and Applications*, North-Holland, New York, 1985.

[6]Shieh, C. F., "Three-Dimensional Grid Generation Using Elliptic Equations with Direct Grid Distribution Control," *AIAA Journal*, Vol. 22, No. 3, March 1984, pp. 361–364.

[7]Lee, K. D., and Henderson, T. L., "Grid Generation Strategies for Turbomachinery Configurations," NASA CP-10045, 1990.

[8]Kunz, R. F., and Lakshminarayana, B., "Three Dimensional Navier-Stokes Computation of Turbomachinery Flows Using an Explicit Numerical Procedure and a Coupled k-ε Turbulence Model," *Journal of Turbomachinery*, Vol. 114, No. 3, pp. 627–642.

[9]Bansod, P., and Rhie, C. M., "Computation of Flow Through a Centrifugal Impeller with Tip Leakage," AIAA Paper 90-2021, Orlando, FL, July 1990.

[10]Gorski, J., "Generation of H-Type Grids Using the GRAPE Code," Dept. of Aerospace Engineering, Pennsylvania State Univ., PSU/TURBO R83-3, University Park, PA, 1983.

[11]Kennon, S. R., and Dulikravich, G. S., "Optimum Acceleration Factors for Iterative Solution of Linear and Nonlinear Differential Systems," *Computer Methods in Applied Mechanics and Engineering*, Vol. 47, North-Holland, New York, 1984, pp. 357–367.

[12]Huang, C-Y, Kennon, S. R., Dulikravich, G. S., "Generalized Nonlinear Minimal Residual (GNMLR) Method for Iterative Algorithms," *Journal of Computational and Applied Mathematics*, Vol. 16, North-Holland, New York, 1986, pp. 215–232.

[13]Moore, J., and Moore, J. G., "Shock Capturing and Loss Prediction for Transonic Turbine Blades," *Proceedings of International Symposium on Air Breathing Engines*, AIAA Paper 89-7017, 1989.

[14]Lakshminarayana, B., "Leakage and Secondary Flows in Axial Compressor Cascades," Vols. 1 and 2, Ph.D. Dissertation, Liverpool Univ., Liverpool, UK, Sept. 1963.

[15]Lakshminarayana, B., and Horlock, J. H., "Leakage and Secondary Flows in Compressor Cascades," Aeronautics Research Council, R&M 3483, Liverpool, UK, March 1965.