AIAA JOURNAL
Vol. 35, No. 10, October 1997

Implicit Method for Incompressible Flow Calculations in Three-Dimensional Ducts and Cascades

E. S. Politis,* K. C. Giannakoglou,† and K. D. Papailiou‡
National Technical University of Athens, 15710 Athens, Greece

A method is presented for the numerical solution of the incompressible Navier–Stokes equations in three-dimensional ducts and cascades. Both laminar and turbulent flows are resolved, the latter through the k–ε turbulence model. The method is based on an elliptic formulation, which consists of the sequential solution of the mean flow and the turbulence equations through an approximate factorization technique. The cost of factorization is minimized through the formation and inversion of a single matrix for all but the continuity equations. The continuity constraint is enforced by means of the pressure-correction technique for which a more efficient variant of the approximate factorization scheme is employed. The method exhibits good convergence characteristics, as depicted from the analyzed three-dimensional problems. The prediction capabilities of this method are demonstrated by analyzing the laminar flow in a three-dimensional bend and the turbulent flow in a linear compressor cascade, with zero and nonzero tip clearance. For cascade flow problems with nonzero tip clearances, a variant of the present method accommodating multiple, patched subdomains is devised.

Nomenclature

C	= chord
c_μ, c_1, c_2	= k–ε model constants
D_h	= hydraulic diameter
G	= production rate of turbulence
g^{jk}	= contravariant metric tensor
J	= Jacobian of transformation
k	= turbulent kinetic energy
p	= pressure
r	= radius
S	= source term
t	= tip gap height
u_i	= Cartesian velocity component
V^i	= contravariant velocity component
x_i	= Cartesian coordinate
y^+	= nondimensional distance
Γ	= diffusion coefficient
ε	= turbulent energy dissipation
ζ	= acceleration parameter
ν	= kinematic viscosity
ν_t	= eddy viscosity
ξ^i	= curvilinear coordinate
ϱ	= density
σ_k, σ_ε	= effective Prandtl numbers
Φ	= solution variable
ω	= underrelaxation factor

Subscripts

E, W, F, B, N, S	= nodes adjacent to node P
e, w, f, b, n, s	= midfaces over a finite volume
m	= midnode

Superscripts

c	= off-diagonal element of a matrix
d	= diagonal element
n	= iteration
l	= correction

Operators

$[A]$	= matrix
$\{b\}$	= array
$\|\|b\|\|$	= Euclidean norm of b

Introduction

SOLUTION methods for incompressible fluid flows, based on the partial or global parabolization of the flow or being in conformity to the thin-layer assumption, have found widespread use. There is a large body of literature on these techniques, as cited in review papers.[1] A notable advantage of these methods is the low requirements for computer memory and calculation time. On the other hand, because they are based on space marching procedures, they usually fail when large separation zones occur. This can be circumvented by methods based on fully three-dimensional solution algorithms, similar to the one proposed here.

For flow problems in complex three-dimensional geometries, the use of curvilinear, boundary fitted coordinates has become a common practice. Staggered or collocated grid arrangements, in conjunction with the appropriate finite volume schemes, are in use. The staggering technique[2] prevents calculations of nonphysical spatial pressure oscillations, but renders them cumbersome in handling complex geometries. On the other hand, the implementation of collocated grid arrangements is straightforward, whereas pressure and velocity decoupling can be alleviated through controllable smoothing mechanisms.[3,4] In the present method, the collocated grid system has been adopted.

In the collocated grid method, the momentum and turbulence equations are discretized using central schemes for the diffusion and the pressure terms. In the momentum equations, a third-order upwind scheme[5] is used for convection, whereas the turbulence equations are merely first-order upwinded, for the sake of stability. The divergence-free velocity constraint yields a pressure-correction equation. In the course of the solution algorithm, the convection-diffusion equations are sequentially solved, by defining a common coefficient matrix for all of them. Minor rearrangements are required, to account for differences in the diffusion coefficients among the turbulence and momentum equations. The assembled coefficient matrix is approximately factored, by means of two different techniques: 1) the strongly implicit procedure (SIP) proposed by Stone[6] for seven-diagonal matrices and 2) the seven-point formulation of the modified SIP (MSIP) proposed by Zedan and Schneider.[7] The performance of these schemes will be evaluated using a laminar flow test case. The computationally costly inversion procedure is carried out once per iteration, which contributes to the low computational

Received Feb. 1, 1996; revision received April 13, 1997; accepted for publication June 8, 1997.

*Postgraduate Student, Laboratory of Thermal Turbomachines, P.O. Box 64069.
†Lecturer, Laboratory of Thermal Turbomachines, P.O. Box 64069.
‡Professor, Laboratory of Thermal Turbomachines, P.O. Box 64069. Member AIAA.

requirements of the present method. Convergence is enhanced by a residual minimization scheme employed in the solution of the pressure-correction equation.

To model three-dimensional cascade flows with finite tip clearance, a two-domain approach has been adopted. The cascade passage is discretized using two H-type grids, one filling the main passage up to the blade tip and the second covering the remaining flow domain, which lies above the blade tip. Two-dimensional interpolations are required over a part of the interface, to exchange flow information between the adjacent domains. Along this interface, particular attention is given to the communication of the pressure-correction field.

Governing Equations and Geometrical Transformation

The equations governing the steady, three-dimensional incompressible turbulent flows will be presented in their Reynolds-averaged form. For laminar flows, governing equations are readily obtained by striking out the irrelevant terms. Hereafter the summation convention applies where repeated indices appear. Thus, the continuity equation yields

$$\frac{\partial u_j}{\partial x_j} = 0 \tag{1}$$

whereas the conservation equations for the momentum components, the turbulent kinetic energy, and the turbulent energy dissipation can be given in a compact form as

$$\frac{\partial}{\partial x_j}(u_j \Phi) = \frac{\partial}{\partial x_j}\left(\Gamma^{\Phi} \frac{\partial \Phi}{\partial x_j} \right) + S^{\Phi} \tag{2}$$

where Γ^{Φ} and S^{Φ} depend on the variable to be solved and are summarized in Table 1.

According to the Boussinesq hypothesis, ν_t is related to the turbulence quantities through

$$\nu_t = c_\mu (k^2/\varepsilon) \tag{3}$$

As in the standard Jones and Launder[8] k–ε model, the constants appearing in the preceding expressions share the values

$$c_\mu = 0.09, \qquad c_1 = 1.43, \qquad c_2 = 1.92, \qquad \sigma_k = 1.0, \qquad \sigma_\varepsilon = 1.3$$

To handle complex geometries, a body-fitted curvilinear coordinate system is generated. The transformation from the physical x_i to the computational ξ^i space, $i = 1, 2, 3$, is governed by the Jacobian matrix

$$J = \frac{\partial(x_1, x_2, x_3)}{\partial(\xi^1, \xi^2, \xi^3)} \tag{4}$$

and the transformed form of Eqs. (1) and (2) is deduced through routine calculations as

$$\frac{\partial}{\partial \xi^j}(JV^j) = 0 \tag{5}$$

$$\frac{\partial}{\partial \xi^j}(JV^j \Phi) = \frac{\partial}{\partial \xi^j}\left(J \Gamma^{\Phi} g^{jk} \frac{\partial \Phi}{\partial \xi^k} \right) + JS^{\Phi} \tag{6}$$

The three contravariant velocity components appearing in Eqs. (5) and (6) are the inner products of the Cartesian velocity vector and the contravariant base vectors, namely,

$$V^j = \boldsymbol{V} \cdot \nabla \xi^j, \qquad j = 1, 2, 3 \tag{7}$$

Table 1 Diffusion coefficients and source terms

Φ	Γ^{Φ}	S^{Φ}
u_i	$\nu + \nu_t$	$-\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left\{ \Gamma^{\Phi} \frac{\partial u_j}{\partial x_i} \right\}$
k	ν_t/σ_k	$G/\rho - \varepsilon$
ε	ν_t/σ_ε	$(c_1 G/\rho - c_2 \varepsilon)\varepsilon/k$

Boundary Conditions

At the inflow plane, the velocity components and the turbulent quantities are imposed, the latter in terms of the turbulence intensity and the mixing length distributions. The pressure is iteratively updated through an extrapolation of the calculated pressure field at the interior nodes. At the outflow plane, a uniform pressure distribution is enforced; inasmuch as in incompressible flows nothing but the pressure differences matters, a zero pressure level is imposed. At the same plane, velocity and turbulence variables are iteratively calculated through a zero gradient condition.

Along solid walls either the no-slip condition (for laminar flows) or a logarithmic velocity profile (for turbulent flows) is implied. To reduce the number of grid points, the high Reynolds k–ε model is used, where the wall function technique bridges the gap between the solid surface and the first node (typically lying at the distance of $y^+ = 30$–80). A second-order accurate Neumann pressure condition is derived from the momentum equation, projected onto the normal to the wall direction.

Because a cell-centered formulation is used, all boundary conditions are imposed in the so-called weak sense, at the corresponding faces of the cells formed around the first nodes adjacent to the boundaries. Dirichlet boundary conditions are employed by fixing the values over the control volume face lying on the boundary and extending the calculation domain through fake nodes, for which mirroring techniques are used.

When more than one domain is combined to form the complete calculation domain, interfacing conditions must be imposed. Because their description causes a proliferation of equations with the multidomain approach, this will be deferred to the Results section.

Numerical Solution

Momentum and Turbulence Equations

Following a cell-centered formulation, the equations are integrated over control volumes surrounding the nodal points at which flow quantities are stored. Figure 1 shows a typical control volume around node P, along with some neighboring nodes and midnodes, and clarifies the notation used hereafter.

The integration of the governing equations yields the balance between convective and diffusive fluxes crossing the cell faces. For the turbulence equations, the extra terms modeling the generation and dissipation of turbulence are lumped on the right-hand side. For the numerical discretization of these fluxes, central differences for the diffusion and a quadratic upstream interpolation scheme[5] (of third-order formal accuracy) for the convective terms of the momentum equations are implemented. For the sake of stability,[9] the convective terms in the turbulence equations are discretized via a first-order upwind scheme.

For stability reasons, and to employ a seven-node discretization stencil, the implicit operator is formed using the part of the convection terms that corresponds to the first-order upwind scheme and the part of the diffusion corresponding to the orthogonal metrics in Eq. (6). As a consequence, the matrix form of the discretized momentum and k–ε model equations yields

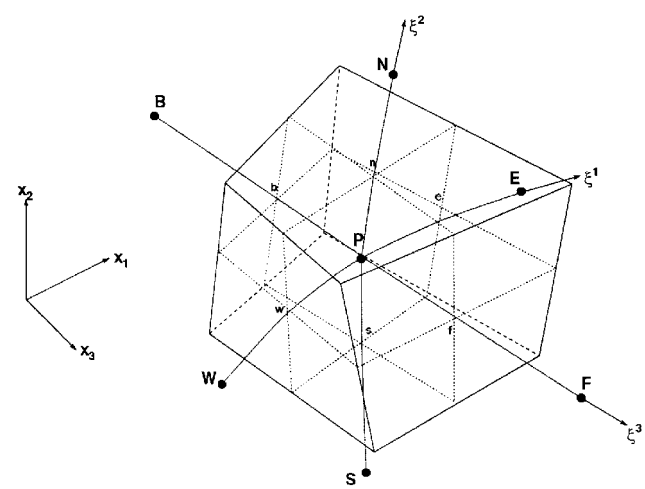


Fig. 1 Three-dimensional control volume and corresponding notation.

$$[A^i]\{\Phi_i\} = [C^i]\{\Phi_i\} + [B^i]\{p\} + [D^i], \qquad i = 1, 2, 3$$
$$[A^i]\{\Phi_i\} = [C^i]\{\Phi_i\} + [D^i], \qquad i = 4, 5 \tag{8}$$

where

$$\Phi_i = u_i, \qquad \Phi_4 = k, \qquad \Phi_5 = \varepsilon, \qquad i = 1, 2, 3$$

In Eq. (8), $[A^i]$, $i = 1, \ldots, 5$, are seven-diagonal matrices, involving the central node P and its six closest neighbors. All nonorthogonal diffusion terms that involve the cross metrics $g^{ij}(i \neq j)$ are treated explicitly and are included in $[C^i]$, $i = 1, \ldots, 5$. $[B^i]$, $i = 1, 2, 3$, discretize the pressure gradient terms in the momentum equations. Finally, $[D^i]$, $i = 1, \ldots, 5$, summarize all remaining terms, such as those resulting from the implementation of a higher-order scheme for the convection terms in the momentum equations, the part of the viscous terms not included in $[C^i]$, and the source terms in the k–ε equations.

Distinguishing between momentum and turbulence equations yields

$$[A^1] = [A^2] = [A^3] = [A]$$
$$[A^i] = [A] + [C_T^i]\{\Phi_i\}, \qquad i = 4, 5 \tag{9}$$

where the additional terms $[C_T^i]\{\Phi_i\}$, $i = 4, 5$, denote extra diffusion terms due to differences in the diffusion coefficients involved in the k–ε equations (see Table 1). In the solution procedure, these extra terms are treated explicitly.

Following the preceding analysis, all convection–diffusion equations can be written in the common matrix form

$$[A]\{\Phi_i\} = \{S_i\}, \qquad i = 1, \ldots, 5 \tag{10}$$

allowing a single seven-diagonal matrix inversion to be utilized.

For stability reasons, Eqs. (10) are underrelaxed and are finally cast in the form

$$\{(1/\omega)[A^d]^{(n)} + [A^c]^{(n)}\}\{\Phi_i\}^{(n+1)} = \{S_i\} + [(1-\omega)/\omega][A^d]^{(n)}\{\Phi_i\}^{(n)} \tag{11}$$

where $[A^d]$ and $[A^c]$ include the main-diagonal and the off-diagonal elements of the initial coefficient matrix $[A]$, respectively, whereas (n) and $(n+1)$ define known and unknown quantities.

In the proposed algorithm, the momentum and turbulence equations remain loosely coupled and can be sequentially solved using the same implicit procedure. Two alternative incomplete factorization techniques are employed, through which the unique coefficient matrix $[A]$ is approximately decomposed into lower and upper triangular matrices $[L]$ and $[U]$, i.e.,

$$[A] \simeq [P] = [L][U] \tag{12}$$

The first procedure is SIP, proposed by Stone[6] for seven-diagonal matrices. According to SIP, $[L]$ and $[U]$ have the same structure as the lower and upper parts of $[A]$, respectively. Seeking an $[L][U]$ product as close as possible to the original matrix $[A]$, while avoiding the appearance of new diagonals, all additional entries in the $[L][U]$ product are Taylor expanded (weighted by a parameter α using a recommended value $\alpha = 0.9$) to the existing diagonals.

The second procedure is based on the MSIP (see Ref. 7). MSIP was originally conceived for 19-diagonal matrices and, consequently, its application to 7-diagonal ones through zeroing all additional entries is straightforward.[7] This results in matrices $[L]$ and $[U]$ featuring a 13- rather than a 7-diagonal layout. As in SIP, a parameter α cancels out all additional entries (the recommended value of $\alpha = 0.4$).

The factorization step, which is common for all equations, is followed by five two-step procedures, which are schematically given as follows.

Step 1:

$$[L]\{\gamma_i\} = \{S_i\} \tag{13a}$$

Step 2:

$$[U]\{\Phi_i\} = \{\gamma_i\}, \qquad i = 1, \ldots, 5 \tag{13b}$$

The solution of Eq. (13) for $i = 1, 2, 3$ yields a provisional velocity field u_i^* that, in general, fails to conform to the continuity constraint. A divergence-free velocity field will be deduced through the solution of a pressure-correction equation. On the other hand, Eqs. (13) provide the final $k^{(n+1)}$ and $\varepsilon^{(n+1)}$ fields, and the eddy viscosity coefficient $\nu_t^{(n+1)}$ is directly updated using Eq. (3).

Pressure-Correction Equation

To set up the pressure-correction equation, Eq. (11) is written as[10]

$$u_i^{P*} = \frac{\omega}{A_P}\left(\sum_{k=E,W,N,S,F,B} A_k u_i^{k*} + C_i\right) - \frac{\omega}{A_p}\left[\frac{\partial}{\partial \xi^k}\left(J\frac{\partial \xi^k}{\partial x_i}p\right)\right]_P + (1-\omega)u_i^{P(n)}, \qquad i = 1, 2, 3 \tag{14}$$

and the following corrections of the Cartesian velocity components and the static pressure are defined:

$$u_i^{(n+1)} = u_i^* + u_i', \qquad i = 1, 2, 3 \tag{15a}$$
$$p^{(n+1)} = p^{(n)} + p' \tag{15b}$$

whereas the divergence-free velocity constraint reads

$$[JV^1]_w^e + [JV^2]_s^n + [JV^3]_b^f = 0 \tag{16}$$

By shifting Eq. (14), already satisfied over the computational cells enclosing nodal points (integer nodes), to the centers of the cell faces, the contravariant velocity components can be expressed in the form[10]

$$(V^\pi)_m^* = \frac{\omega}{\overline{A_m}}\overline{\left(\sum A_k u_i^k|^* + C_i\right)}_m \left.\frac{\partial \xi^\pi}{\partial x_i}\right|_m - \frac{\omega}{\overline{A_m}}\left[\frac{\partial}{\partial \xi^k}\left(J\frac{\partial \xi^k}{\partial x_i}p\right)\right]_m \left.\frac{\partial \xi^\pi}{\partial x_i}\right|_m + (1-\omega)\overline{(V^\pi)}_m, \qquad \pi = 1, 2, 3 \tag{17}$$

where subscript m is for the six cell faces and the overbars are values obtained through algebraic averaging over the two neighboring integer nodes. The last terms in Eqs. (14) and (17) are treated explicitly to prevent any dependency of the solution on the value of the underrelaxation factor ω.

By virtue of Eqs. (15) and (17), Eq. (16) leads to the pressure-correction equation

$$-B_P p_P' + \sum_{k=E,W,N,S,F,B} B_k p_k' = S^* \Leftrightarrow [B]\{p'\} = \{S^*\} \tag{18}$$

where S^* is the nonzero divergence of the provisional velocity field u_i^*, $i = 1, 2, 3$. The pressure terms in Eq. (17), which appear for $i = \pi$, are discretized using local central differences to circumvent pressure-velocity decoupling.

Although incomplete factorization also could have been used in the numerical solution of Eq. (18), a more efficient,[10] in terms of convergence rate, numerical scheme was finally adopted. This numerical scheme has been previously used for laminar flow calculations[10] and can be extended to turbulent flows. To advance the pressure-correction solution from the (n) to the $(n+1)$ iteration level, an acceleration factor ζ is used as follows:

$$\{p'\}^{(n+1)} = \zeta[P]^{-1}\{S^*\} = \zeta\{R'\} \tag{19}$$

where $[P]^{-1}$ is an approximate inverse of $[B]$, obviously calculated through incomplete factorization. Equation (19) may also reflect the standard incomplete lower–upper (LU) decomposition (SIP or MSIP), provided that $\zeta = 1$. In the present method, a ζ value different from 1 was used. This (constant over the whole field) value

is estimated at the end of each iteration by imposing residual minimization constraints. A rather lengthy mathematical analysis leads to the following expression for ζ:

$$\zeta = \frac{\alpha_1}{(\alpha_1^2 + \alpha_2^2)} \tag{20}$$

where

$$\alpha_1 = \left[[P\prime]^{-1}[B\prime] \frac{\{R\prime\}^{(n)}}{\|\{R\prime\}^{(n)}\|} \right] \cdot \frac{\{R\prime\}^{(n)}}{\|\{R\prime\}^{(n)}\|} \tag{21a}$$

$$\alpha_2 = \left\| [P\prime]^{-1}[B\prime] \frac{\{R\prime\}^{(n)}}{\|\{R\prime\}^{(n)}\|} - \alpha_1 \frac{\{R\prime\}^{(n)}}{\|\{R\prime\}^{(n)}\|} \right\| \tag{21b}$$

Results and Discussion

Laminar Flow in a Bend of Square Cross Section

As a first case, the three-dimensional laminar flow in a 90-deg bend of square cross section, experimentally investigated by Humphrey et al.,[11] has been used to validate the proposed method. In the experiment, a long straight upstream extension ensured a fully developed flow at the entry to the bend. The Reynolds number of the flow is equal to 790, with the bulk velocity and the hydraulic diameter D_h the reference velocity and length, respectively. A perspective view of the channel, along with the corresponding notation, is provided in Fig. 2. Because of symmetry in the bend geometry and inlet profiles, only the upper-half of the flow domain is considered in this work, defined by the symmetry plane, which is denoted by α and β in Fig. 2. The inner radius is equal to $r_i = 0.072$ m (corresponding to α), and the outer radius is $r_o = 0.112$ m (corresponding to β). The mesh filling the upper-half of the flow domain consists of $76 \times 26 \times 31$ nodes in the streamwise, the spanwise (normal to the symmetry plane, denoted by z, $0 < z < z_{\max} = D_h/2$), and the radial direction, respectively. To cover the streamwise direction, 37 out of 76 cross planes have been placed inside the bend in equal angular distances.

For the sake of compatibility with the measurements, a fully developed laminar velocity profile[12] was imposed at the inlet of the computational domain, located at a distance of $5\,D_h$ upstream from the entrance of the bend ($\phi = 0$ deg). A zero pressure level was imposed at the exit of the computational domain located at $2.5\,D_h$ downstream of the exit of the bend ($\phi = 90$ deg).

In Fig. 3 the nondimensional stream velocity patterns at six cross sections are presented. For each cross section, the velocity profiles are plotted at two spanwise positions, namely, at the symmetry plane ($z = 0$) and at the plane lying in the middle of the half-duct height ($z = z_{\max}/2$). In the same figure, the calculated profiles are compared with the measurements of Humphrey et al.[11] Very good agreement between the two curves can be observed at all but the $\phi =$ 60-deg planes, where some discrepancies emerge. As expected, after the $\phi = 30$-deg plane the flow is accelerated close to the outer radius wall and strong secondary flow effects appear. These effects can be easily identified in Fig. 4, where the secondary flowfield at the $\phi = 90$-deg cross section is plotted.

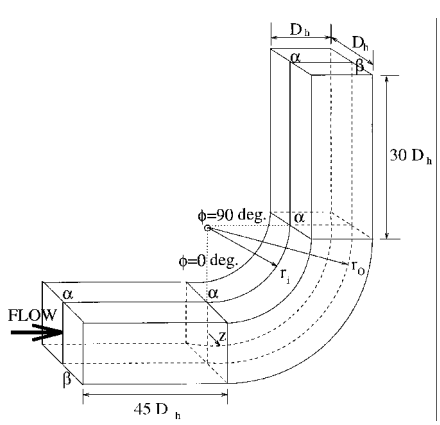
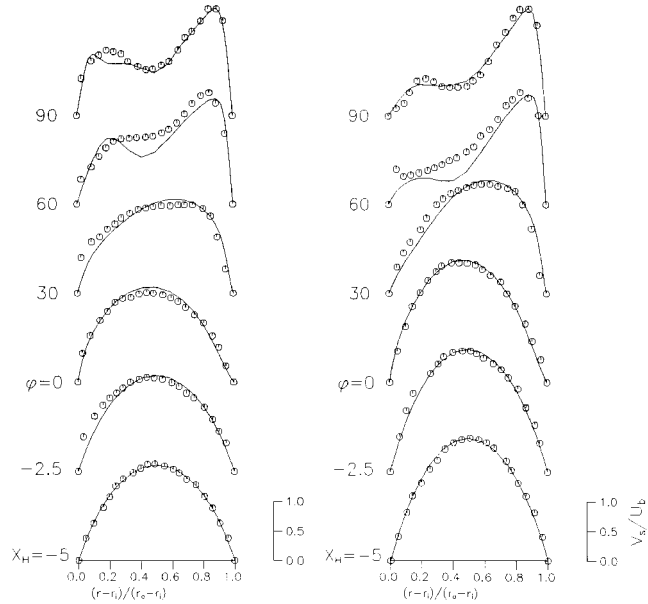


Fig. 2 Perspective view of the three-dimensional bend.



a) $z = z_{\max}/2$ **b) $z = 0$**

Fig. 3 Calculated (——) and measured (symbols) velocity profiles.

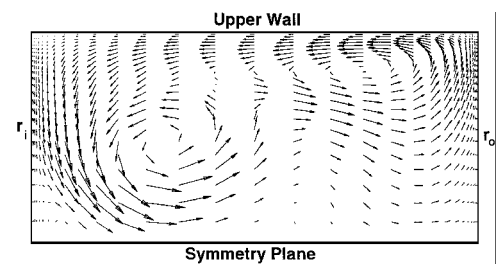


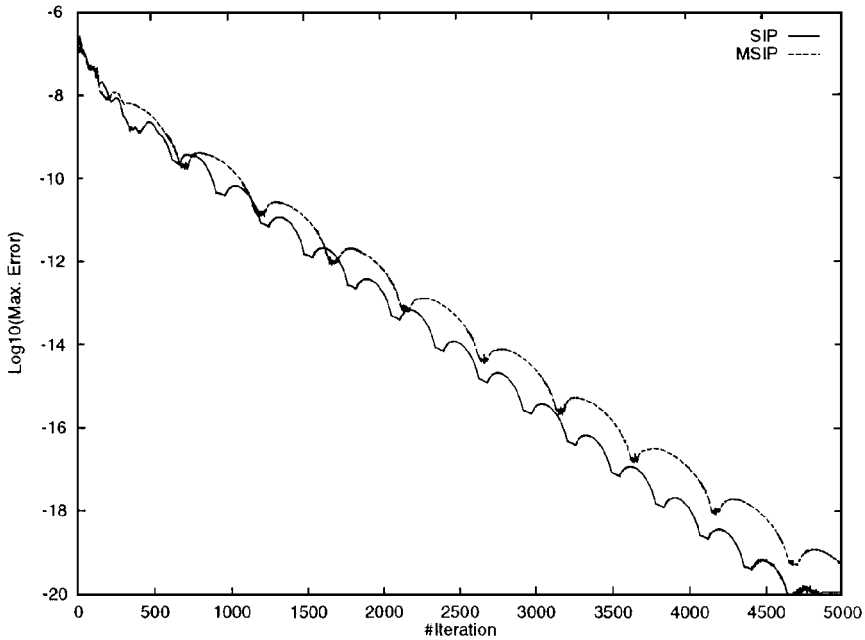
Fig. 4 Secondary flowfield at the exit of the bend.

As far as the convergence history is concerned, the evolution of the maximum errors for the three momentum and the pressure-correction equations is plotted in Fig. 5. Curves obtained using SIP and MSIP are included. A very fast tendency to machine accuracy is observed for both procedures. In addition, the comparison between SIP and MSIP highlights the better convergence characteristics of the former; therefore, it will be used as the standard option in the algorithm for the ensuing analyses. Both approximate factorization schemes revealed significantly enhanced stability, but SIP performs better than MSIP, which translates into approximately a 10% economy in the number of required iterations. In addition, the recursive algorithm of SIP is simpler than MSIP, yielding decreased computing requirements per iteration and rendering SIP even more attractive.

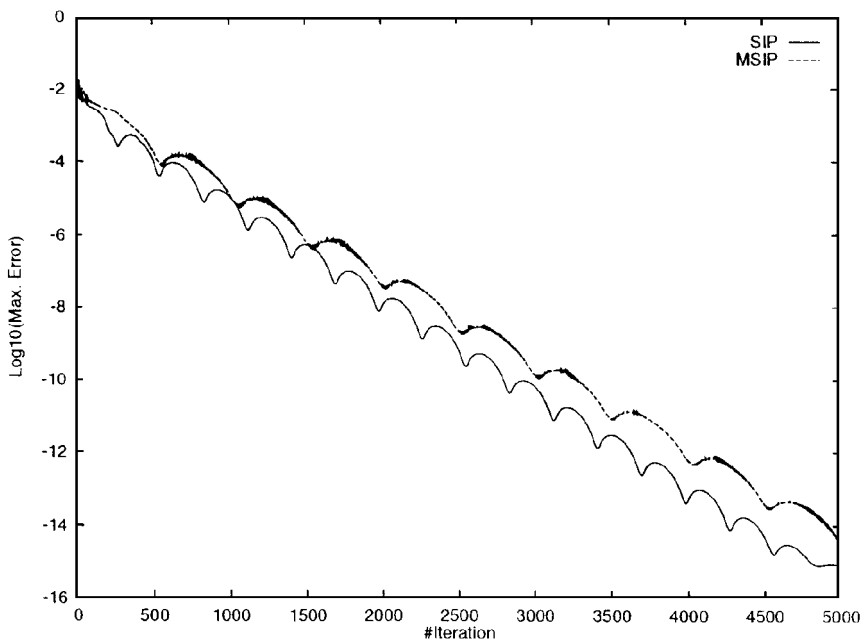
Linear Compressor Cascade with and Without Tip Clearance

A highly loaded, low-speed linear compressor cascade[13,14] was the second case examined. The cascade consisted of NACA 65-12-A10-10 blades, with a chord length $C = 0.13$ m and a stagger angle of 30 deg. The solidity was equal to 1.25, and the span to chord ratio was 2.1. According to the available data, two cases were analyzed, the first with no tip clearance (case A) and the second with a tip gap height equal to $0.011C$ (case B). Available experimental data[13] dictate that slightly different inlet conditions prevail in each case. Nevertheless, in the present study, a unique inlet velocity profile was imposed in both cases at $0.667C$ upstream of the leading edge. The present case has been selected for analysis because it involves a multidomain configuration, as is usually the case in practical turbomachinery applications. For all runs performed, the SIP was used as the standard iterative solver.

For the discretization of the no-tip case, a standard H-type grid of $85 \times 41 \times 41$ nodes in the streamwise, pitchwise, and spanwise directions, respectively, was used. The resulting two-dimensional blade-to-blade layout is shown in Fig. 6. In case B, a two-domain approach was devised and an appropriate multidomain variant of



Momentum equations



Pressure-correction equation

Fig. 5 Convergence history for the laminar bend.

the code was implemented. Two H-type grids are in use (as shown in Fig. 7). The first domain is similar to the single domain used in case A but covers only the part of the passage below the blade tip. This grid will be referred to as the main passage grid (MPG) and has $85 \times 41 \times 35$ nodes. Both the MPG (case B) and the case A grid were generated by first meshing their common two-dimensional layout (see Fig. 6) and then stacking it in the spanwise direction, up to the shroud (for case A) or up to the tip of the blade (for case B). The mesh generation for the two-dimensional grid was accomplished using Poisson type equations,[15] with appropriate source terms controlling the clustering and the orthogonality of grid lines. The second grid in case B [tip grid (TG)] is still of H type ($85 \times 47 \times 7$ nodes) and fills the remaining part of the domain, above the blade tip. The two grids communicate along their interface, which coincides with a blade-to-blade surface at the blade tip height. The two grids are built in such a way that the two blade-to-blade mesh layouts remain identical in their major part, as the first 37 transversal grid lines, starting from the suction side, do coincide. This is shown in Fig. 8, where a detailed view of the two-dimensional grids over the interface is given; only the region close to the leading edge is shown. If appropriate boundary conditions along the interface are imposed during the generation of the two grids, then the communication procedure in this part, during the solution, is straightforward. Interpolation is deliberately introduced only in the remaining small part of the interface, as implied by Fig. 8. The complete two-dimensional layout of the TG, meshed in a similar way, is shown in Fig. 9.

An important issue is the flow variable communication along the interface, during the iterative solution. A layer of fake cells is generated by extending each grid toward the adjacent one, and in the iterative calculation algorithm a communication task seeks to correctly assign all flow variables to the fake nodes. The extension of the grid is carried out so that the extended nodes are located at the same radius as the corresponding grid nodes of the adjacent grid. Because of the coinciding interface nodes and the form of the grid lines crossing the major part of the common surface, the allocation of flow variables to the fake nodes is readily

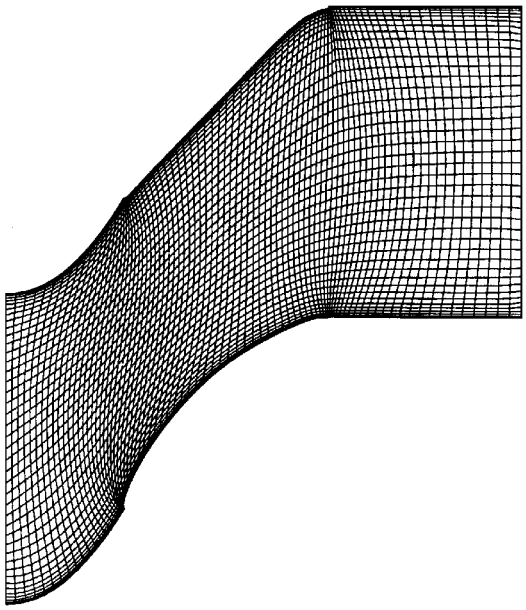


Fig. 6 Two-dimensional layout of the case A grid.

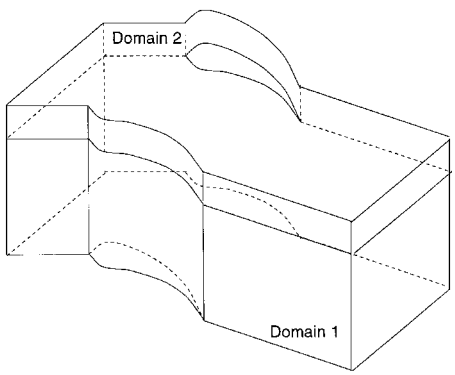


Fig. 7 Grid system used in tip-clearance flow study.

accomplished for all but the pressure-correction variable. On the other hand, a two-dimensional interpolation is used for the non-coinciding nodes at the remaining part of the interface through Coon's interpolation scheme. For all scalar quantities, boundary conditions are indirectly imposed by lumping the contribution of the extended nodes into the source terms. The pressure-correction still requires a particular treatment, and this will be discussed subsequently.

In a single-domain solution method, pressure-correction is a restarting variable within each iteration. In a multidomain environment, the pressure-correction boundary conditions are imposed as follows. Over the fake nodes, the pressure-correction values retain their values from the most recent iteration, even if the same variable is always restarted in the interior of the domain. By explicitly treating the contribution of the phantom nodes, the fluxes crossing the interface are controlled, giving rise to a smoother pressure field at the end of each iteration, and this in turn provides a smoother velocity field close to the interface. In this way, the performance degradation due to the communication scheme is kept low. The convergence history retains the behavior shown in Fig. 5, with an expected degradation of approximately 15%, because it is now penalized by the interfacing task.

A comparison between predicted and measured nondimensional velocity distributions along the blade surface at midspan is presented in Fig. 10 for cases A and B. The agreement can be seen to be very good, and slight differences between predictions and measurements in case B are attributed to differences between the imposed and the measured inlet velocity profiles.

The radial distributions of the axial and peripheral velocities for the two cases are plotted in Fig. 11 against the experimental curves

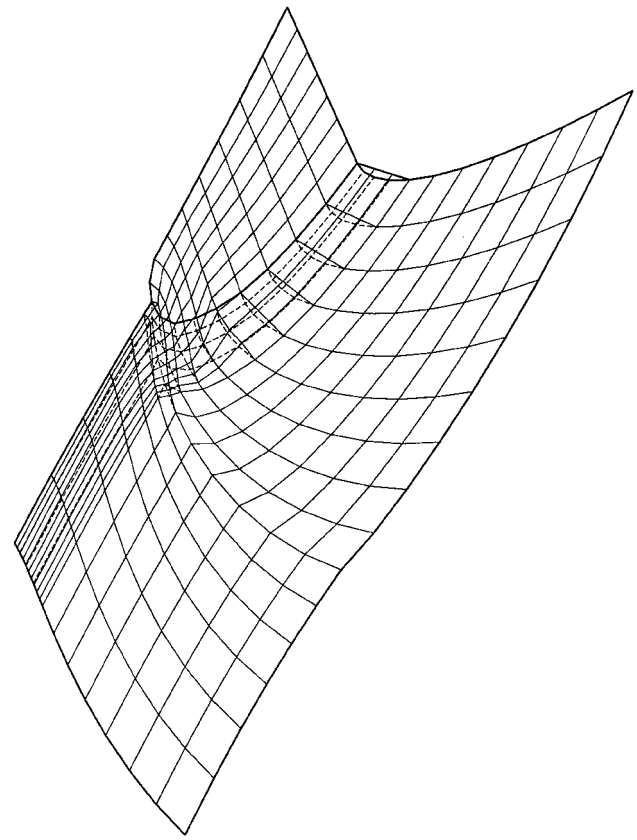


Fig. 8 Comparison of two-dimensional layouts of the grids used, in the region close to leading edge: ——, TG and - - - -, main passage grid.

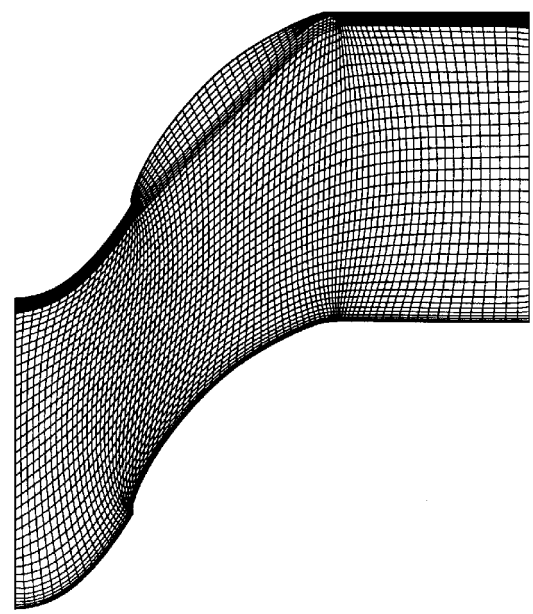
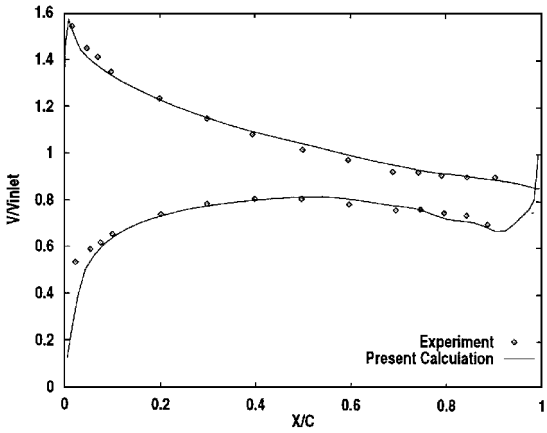
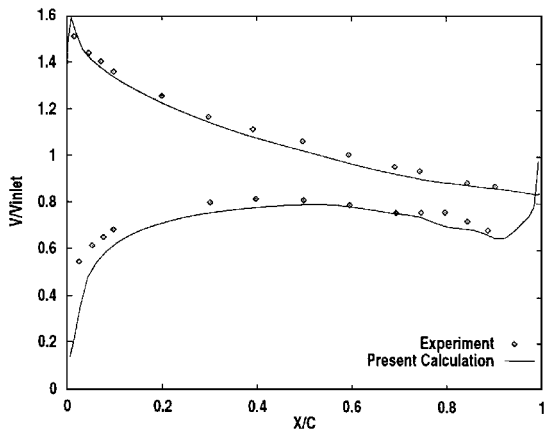


Fig. 9 Two-dimensional layout of the TG used in case B.

for case A. The peripheral velocity prediction for case A does not perfectly match the measured profile, and the deficiencies of the incorporated k–ε turbulence model remain to be examined. Figure 12 shows the calculated velocity vectors at midspan for case A and at a distance of 5 mm from the shroud for case B. The predicted flow patterns are in good agreement with the corresponding figure published in Ref. 13 (not repeated here). One may see the strong interaction between the jet, which emerges from the tip clearance and the main flow. The interaction of the leakage jet and the secondary flow results in a stagnation line, which starts at the leading edge and extends farther downstream.



Case A



Case B

Fig. 10 Nondimensional velocity distribution along the blade surface at midspan.

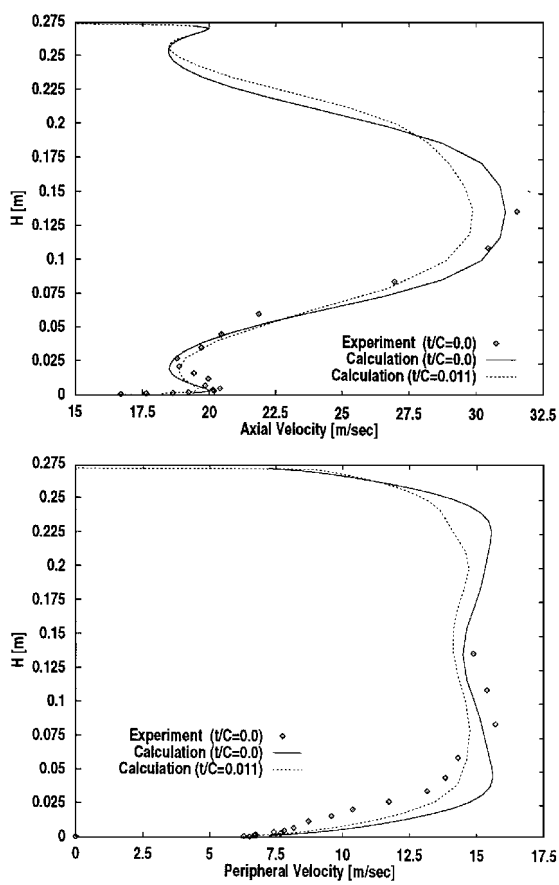


Fig. 11 Spanwise distributions of axial and peripheral velocity at exit.

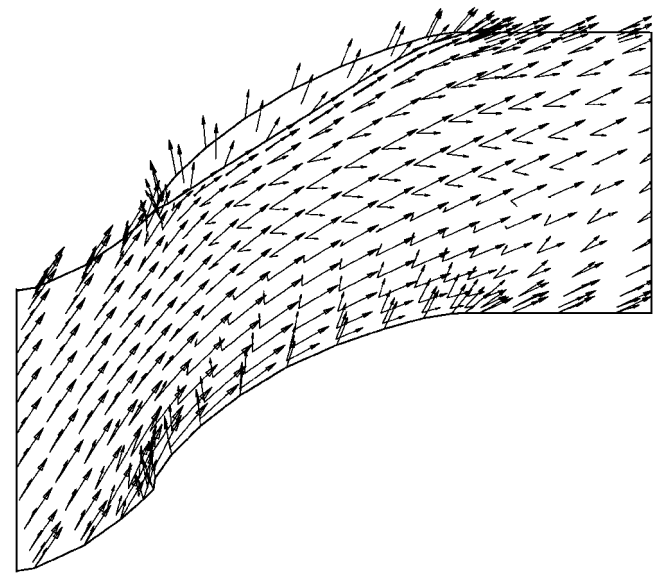


Fig. 12 Velocity vectors at midspan (case A) and at a 5-mm distance from the shroud (case B).

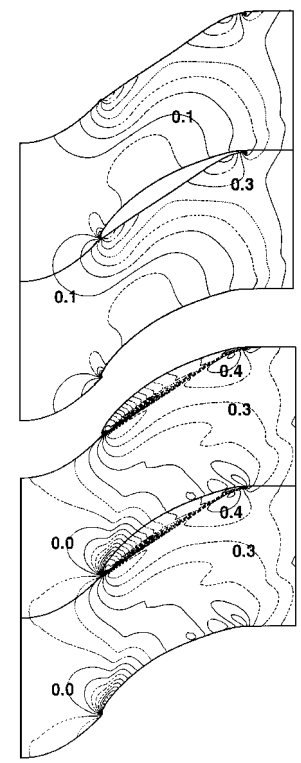


Fig. 13 Pressure coefficient contours at shroud for both cases: top, case A and bottom, case B.

Finally, the static pressure coefficient contours at the shroud of the cascade are plotted for both cases in Fig. 13. Even if data for the pressure fields are not available from measurements, the patterns appearing are in accord with our qualitative expectations for tip leakage flows.

Conclusions

In this work, a three-dimensional Navier–Stokes solution method for incompressible flow problems, which consists of the successive solution of mean flow and turbulence equations through a strongly implicit procedure, along with a pressure-correction step to provide a divergence-free velocity field, is presented. All but the continuity equations lead to the same coefficient matrix, so that the costly matrix inversion task is undertaken only once. Two approximate factorization schemes, SIP and MSIP, have been employed; the former has better convergence characteristics than the latter. An efficient

numerical scheme for the solution of the pressure-correction equation is proposed based on residual minimization constraints. This scheme is perfectly combined with the incomplete LU decomposition, used for the solution of all convection-diffusion type equations.

Laminar and turbulent duct and cascade flow problems have been used to validate the proposed method. In one of the examined cases that deals with linear cascades with nonzero tip clearance, a multi-domain version of the method was programmed and tested. It is shown that splitting of the original flow domain into subdomains (each of which is separately solved in an implicit fashion) can be treated in an efficient way, provided that the appropriate pressure-correction boundary conditions are imposed along the interface. The convergence degradation due to the two-domain formulation is about 15%.

Acknowledgment

A part of this work was funded by the European Economic Community AC3A Project AERO-CT92-0039 entitled Advanced Civil Core Compressor Aerodynamics.

References

[1]Lakshminarayana, B., "An Assessment of Computational Fluid Dynamic Techniques in the Analysis and Design of Turbomachinery—The 1990 Freeman Scholar Lecture," *Journal of Fluids Engineering*, Vol. 113, Sept. 1991, pp. 315–352.

[2]Patankar, S. V., *Numerical Heat Transfer and Fluid Flow*, Hemisphere, New York, 1978, pp. 118–120.

[3]Rhie, C. M., and Chow, W. L., "Numerical Study of the Turbulent Flow Past an Airfoil with Trailing Edge Separation," *AIAA Journal*, Vol. 21, No. 11, 1983, pp. 1525–1532.

[4]Majumdar, S., "Role of Underrelaxation in Momentum Interpolation for Calculation of Flow with Nonstaggered Grids," *Numerical Heat Transfer*, Vol. 13, 1988, pp. 125–132.

[5]Leonard, B. P., "A Stable and Accurate Convective Modelling Procedure Based on Quadratic Upstream Interpolation," *Computer Methods in Applied Mechanics and Engineering*, Vol. 19, 1979, pp. 59–98.

[6]Stone, H. L., "Iterative Solution of Implicit Approximations of Multidimensional Partial Differential Equations," *SIAM Journal on Numerical Analysis*, Vol. 5, No. 3, 1968, pp. 530–558.

[7]Zedan, M., and Schneider, G. E., "A Three-Dimensional Modified Strongly Implicit Procedure for Heat Conduction," *AIAA Journal*, Vol. 21, No. 2, 1983, pp. 295–303.

[8]Jones, W. P., and Launder, B. E., "The Prediction of Laminarization with a Two-Equation Model of Turbulence," *International Journal of Heat and Mass Transfer*, Vol. 15, 1972, pp. 301–314.

[9]Lien, F.-S., and Leschziner, M. A., "A Pressure-Velocity Solution Strategy for Compressible Flow and its Application to Shock/Boundary-Layer Interaction Using Second-Moment Turbulence Closure," *Journal of Fluids Enginnering*, Vol. 115, Dec. 1993, pp. 717–725.

[10]Giannakoglou, K. C., and Politis, E. S., "A Segregated Implicit Solution Algorithm for 2-D and 3-D Laminar Incompressible Flows," *International Journal for Numerical Methods in Fluids*, Vol. 21, No. 11, 1995, pp. 1067–1086.

[11]Humphrey, J. A. C., Taylor, A. M. K., and Whitelaw, G. H., "Laminar Flow in a Square Duct of Strong Curvature," *Journal of Fluid Mechanics*, Vol. 83, Pt. 3, 1977, pp. 509–527.

[12]White, F. M., *Viscous Fluid Flow*, McGraw–Hill, New York, 1974, pp. 119–125.

[13]Flot, R., "Contribution a l' Etude des Ecoulements Secondaires dans les Compresseurs Axiaux," Ph.D. Thesis, Univ. Claude-Bernard, Lyon, France, April 1975.

[14]Pouagare, M., and Delaney, R. A., "Study of Three-Dimensional Viscous Flows in an Axial Compressor Cascade Including Tip Leakage Effects Using a SIMPLE-Based Algorithm," International Gas Turbine Conf. and Exhibit (Dusseldorf, Germany), American Society of Mechanical Engineers, June 1996 (ASME Paper 86-GT-84).

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

G. M. Faeth
Associate Editor