Numerical Investigation of Hot Streaks in Turbines

Björn Krouthén* and Michael B. Giles†
Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

A numerical method is developed for simulation of hot-streak redistribution in a two-dimensional model of a turbine rotor. The flow domain is divided into a viscous region near the blade where the Reynolds-averaged, thin shear-layer Navier-Stokes equations are solved using an implicit finite-volume technique, and an inviscid core region where the Euler equations are solved using an explicit finite-volume method. The computational mesh consists of an O-mesh and an H-mesh patched together smoothly to cover the domain of interest. Computations are performed using two different flow conditions. The first test case uses hot streaks with a temperature ratio of 1.2 and a tangential inflow angle of 40 deg, whereas the second test case is run with a temperature ratio of 2.0 and a tangential inflow angle of 45 deg. The computed solution from both test cases predicts a migration of hot gas to the pressure surface, which also has been observed experimentally.

Introduction

A MODERN jet engine is designed to have an extremely high temperature gas leaving the combustor. The temperature tolerance of the guide vanes is usually based on an average value of the exit combustor temperature. Due to introduction of cooling air in the nozzle, the turbine entry temperature (TET) measured behind the guide vanes is lower than the combustor exit temperature. The temperature tolerance of the blades in the first rotor row is based on an averaged TET. Recent investigations, however, have shown that hot gas migrates to the pressure surface of the rotor blade. This can lead to peak temperatures on the rotor blade that might exceed acceptable metal temperatures and hence lead to blade failures. This indicates that the TET, which is currently used, is too low an estimate for the rotor surface temperature.

In an earlier work by Butler et al.,[1] an experimental and analytical investigation of the redistribution process for an axial turbine stage was presented. In the experiment, a streak of hot air seeded with CO_2 was introduced at one circumferential location upstream of the inlet guide vane. The redistribution of the hot streak was determined by measuring the concentration of CO_2 inside the turbine stage. Measurements of CO_2 taken on the rotor surface indicated that hot and cold gas had been segregated with the cold gas migrating to the suction surface and the hot gas to the pressure surface. In the same paper it was postulated that the segregation effect was due to the difference in rotor relative inlet angles of the hot and cold gases. The postulate is based on an observation by Kerrebrock and Mikolajczak[2] in their work on wake transport in compressors.

The observed segregation phenomenon[1] has been investigated numerically by Rai and Dring.[3] Their paper includes a comparison between Rai's computations and Butler's measurements. The computations did not give the same high temperature excess as was observed experimentally. The discrepancies between calculations and experiment were blamed on three basic differences in flow conditions: 1) the experiment was fully three-dimensional; 2) the temperature ratio between hot and cold gas was 1.2 in the calculations and 2.0 in the experiment; and, finally, 3) the flow coefficient (ratio betwen axial inlet velocity and circumferential velocity) was 0.78 in the calculations and 0.68 in the experiment. It was believed[3] that the lower flow coefficient would accentuate the hot-streak accumulation on the rotor pressure side. In a previous experiment by Stabe, Whitney, and Moffitt,[4] it was found that a temperature ratio of 1.2 between hot and cold gas did not affect the turbine performance.

Viscous phenomena such as growth of a boundary layer near the blade surface and vortex shedding from the trailing edge are very local phenomena. Their redistribution and transport in large regions of the flow path is basically governed by the inviscid Euler equations. It is therefore reasonable to assume that in a narrow region around the blade the flow is governed by the viscous Navier-Stokes equations and elsewhere by the Euler equations. Considering the high computational cost involved in the simulation of a viscous, unsteady flow, it would be advantageous if the computational domain could be divided into a viscous and an inviscid region. Instead of using an expensive implicit, time-accurate method in the whole domain, its use could then be restricted to the viscous region close to the blade surface, and a less expensive explicit method could be used in the outer, inviscid region.

In a typical turbine blade design the blade will have substantial thickness, it will be highly cambered, and its leading and trailing edges will be rounded. It is therefore clear that, for a viscous calculation, an O-type structured mesh is best suited to solve the boundary layer around the blade. However, due o the spatial periodicity, the O-mesh will inevitably become very skewed at the inflow and periodic boundaries and hence increase the numerical errors introduced in the solution. The strong nonlinearity of the incoming hot streak could introduce numerical errors if the cells at the inflow boundary are too skewed. An H-mesh, on the other hand, has very good inlet-outlet properties but has the disadvantage of resolving the leading and the trailing edges very poorly. This leads to the conclusion that for a typical turbine configuration, no single grid system offers satisfactory grid properties in the entire turbine stator or rotor passage.

The idea of dividing the computational mesh in a turbine passage into patched subgrids that match the different characteristic flow regions was introduced by Norton, Thompkins, and Haimes.[5] They divided the computational mesh into an O-mesh in the viscous region close to the blade and an H-mesh in the core flow region. As a numerical algorithm for solving the Reynolds-averaged, Navier-Stokes equations, they used an implicit cell-centered finite-volume technique. In both regions the full Navier-Stokes equations were solved with the difference that in the inner O-mesh region a turbulence model was applied. Due to a pointer system in the solution algorithm, the

Presented as Paper 88-3015 at the AIAA/ASME/SAE/ASEE 24th Joint Propulsion Conference, Boston, MA, July 11-13, 1988; received Nov. 21, 1988; revision received July 24, 1989.

*Research Assistant; also with The Aeronautical Research Institute of Sweden. Member AIAA.

†Charles Stark Draper Assistant Professor. Member AIAA.

scheme became grid transparent and hence very flexible. The idea of splitting the computational grid into different subgrids was carried even further by Nakahashi and Obayashi.[6] In their paper they used a structured C-mesh in the wake- and boundary-layer regions and an unstructured, triangular grid in the remaining region. In the wake- and boundary-layer regions, the Reynolds-averaged thin-layer Navier-Stokes equations were solved using an implicit finite-difference algorithm. In the outer region the viscous terms were neglected, and the Euler equations were solved with a finite-element method. The method was shown to give very good results for a bi-airfoil case. Nakahashi et al.[7] also successfully used their patched zonal method for a two-dimensional turbine cascade configuration.

In the present work, a few of the ideas described above are used together with some completely new approaches. The computational domain is divided into a viscous region close to the turbine blade and an inviscid region consisting of the core flow. Therefore, the computational mesh is divided into two patched grids, an O-mesh in the viscous boundary-layer region and an H-mesh in the core flow region. In the viscous region an implicit finite-volume technique is used to solve the thin shear layer (TSL) Navier-Stokes equations. A two-layer algebraic turbulence model due to Baldwin and Lomax is used to model the turbulent viscosity. The same finite-volume discretization is used for the Euler equations in the inviscid region. Since the same discretization is used in the two regions, the mesh interface will become transparent to the finite-volume operator. In the inviscid region an explicit Runge-Kutta time integration algorithm is used to advance the solution in time. To be able to simulate hot streaks coming in from the stator, a special technique for treating the inflow and outflow boundaries is applied.

The present numerical algorithm is used to simulate redistribution of hot streaks in a turbine rotor. Simulations are performed under two different flow conditions in order to give results that can be compared with both previous calculations and experiment. Snapshots of the solution at different time levels of the cycle are given in the form of isotherms in which the convecton of the hot streak through the rotor can be studied. A comparison between results from the present method, Rai and Dring's computation,[3] and Butler et al.'s experiment[1] is presented for the time-averaged pressure distribution and the time-averaged temperature distribution.

Governing Equations

The two-dimensional Navier-Stokes equations can be written in nondimensional form as

$$\frac{\partial \bar{q}}{\partial \bar{t}} + \frac{\partial \bar{F}}{\partial \bar{x}} + \frac{\partial \bar{G}}{\partial \bar{y}} = \frac{1}{Re_c}\left(\frac{\partial \bar{F}^v}{\partial \bar{x}} + \frac{\partial \bar{G}^v}{\partial \bar{y}}\right) \tag{1}$$

where

$$\bar{q} = \begin{bmatrix} \bar{\rho} \\ \bar{\rho}\bar{u} \\ \bar{\rho}\bar{v} \\ \bar{e} \end{bmatrix} \quad \bar{F} = \begin{bmatrix} \bar{\rho}\bar{u} \\ \bar{\rho}\bar{u}^2 + \bar{p} \\ \bar{\rho}\bar{u}\bar{v} \\ \bar{u}(\bar{e} + \bar{p}) \end{bmatrix} \quad \bar{G} = \begin{bmatrix} \bar{\rho}\bar{v} \\ \bar{\rho}\bar{u}\bar{v} \\ \bar{\rho}\bar{u}^2 + \bar{p} \\ \bar{v}(\bar{e} + \bar{p}) \end{bmatrix}$$

Here ρ is the density, u and v are the Cartesian velocity components, p is the static pressure, e is the total energy per unit volume, Re_c is the Reynolds number based on the axial chord and the inlet stagnation speed of sound. The bar indicates that the equations are written in nondimensional form. For the nondimensionalization the following quantities are used: axial chord length c, inlet stagnation density ρ_∞, inlet stagnation speed of sound a_∞, inlet stagnation temperature T_∞, and inlet coefficient of viscosity μ_∞.

Since the intention is to divide the flow region into a viscous region and an inviscid region, two cases of Eq. (1) can be distinguished between

1) $Re \to \infty$, i.e., inviscid flow. In this case the viscous flux terms become

$$\bar{F}^v = \mathbf{0}, \qquad \bar{G}^v = \mathbf{0}$$

2) Viscous flow, i.e., $Re < \infty$, where the viscous flux terms become

$$\bar{F}^v = \begin{bmatrix} 0 \\ \bar{\mu}2/3(2\bar{u}_x - \bar{v}_y) \\ \bar{\mu}(\bar{u}_y + \bar{v}_x) \\ \bar{\mu}\bar{v}(\bar{u}_y + \bar{v}_x) + \bar{\mu}\bar{u}2/3(2\bar{u}_x - \bar{v}_y) + \dfrac{\bar{\mu}}{(\gamma - 1)Pr}\bar{T}_x \end{bmatrix}$$

$$\bar{G}^v = \begin{bmatrix} 0 \\ \bar{\mu}(\bar{u}_y + \bar{v}_x) \\ \bar{\mu}2/3(2\bar{v}_y - \bar{u}_x) \\ \bar{\mu}\bar{u}(\bar{u}_y + \bar{v}_x) + \bar{\mu}\bar{v}2/3(2\bar{v}_y - \bar{u}_x) + \dfrac{\bar{\mu}}{(\gamma - 1)Pr}\bar{T}_y \end{bmatrix}$$

where Pr is the Prandtl number that relates the coefficients of viscosity and thermal conductivity and is assumed to be constant. For a perfect gas the equation of state is

$$\bar{p} = (\gamma - 1)\,[\bar{e} - 1/2\bar{\rho}(\bar{u}^2 + \bar{v}^2)], \qquad \bar{T} = \gamma\bar{p}/\bar{\rho} \tag{2}$$

Finally the coefficient of viscosity is related to the temperature using Sutherland's formula

$$\bar{\mu} = \bar{T}^{3/2}\left(\frac{1 + S_k}{\bar{T} + S_k}\right) \tag{3}$$

where S_k is a constant.

Numerical Method

The basic philosophy in the construction of the numerical scheme is to divide the computational region into two subregions. The idea is that one can save a lot of computational time by choosing a clever combination of numerical schemes. The first region is the boundary-layer region that is defined close to the cascade blade. In this region the flow is highly viscous and the computational mesh must be very fine to be able to resolve the boundary layer. In this region the thin shear-layer Navier-Stokes equations are solved on an O-mesh, using an implicit finite-volume technique. By using an implicit scheme one can avoid a CFL restriction due to small cells. The second region is the core-flow region that is defined outside the boundary-layer region. In this region the flow is basically inviscid and governed by the Euler equations. The equations are solved on a structured H-mesh, using an explicit finite-volume technique that is much faster than the implicit method.

For the spatial part of Eq. (1), the finite-volume technique given by Eriksson[8] and Müller and Rizzi[9] is used for the inviscid part of the equation, and a thin shear-layer form of the finite-volume technique described by Swanson and Turkel[10] is used for the viscous terms. That is, only viscous terms in the direction normal to the blade are considered. To derive the spatial finite-volume operator, consider Eq. (1) written in intergral form:

$$\int_\Omega \frac{\partial q}{\partial t}\,dV + \oint_{\partial\Omega} H(q)\cdot n\,ds = \frac{1}{Re_c}\oint_{\partial\Omega} H^v(q)\cdot n\,ds \tag{4}$$

where the matrix $H(q)$ has the inviscid flux vectors F and G as columns and $H^v(q)$ has the viscous flux vectors F^v and G^v as columns. Assuming that the mesh is fixed in time, Eq. (4) can be approximated by

$$\frac{dq_{ij}}{dt}+\frac{1}{A_{ij}}\left[\sum_{k=1}^{4}H_{pk}\cdot S_{pk}\right]=\frac{1}{Re_c}\frac{1}{A_{ij}}\left[\sum_{k=1}^{2}H^v_{pk}\cdot S_{pk}\right] \tag{5}$$

where S_{pk} is the face vector at face k and A_{ij} is the area of cell ij. These quantities are identified in Fig. 1. The finite-volume formulation [Eq. (5)] can be shown to be second-order accurate. The inviscid fluxes are evaluated on the cell faces by averaging

$$H_{pk}=1/2\ (H_k+H_p)$$

Also see Fig. 1 for notations.

The sum of the viscous fluxes in the right-hand side of Eq. (5) only includes two terms. This is due to the thin shear-layer approximation. Consider Fig. 2 that shows the computational cells close to a solid wall. It can be seen that $a', c' \ll b', d'$, and hence the contribution to the line integral comes mainly from the fluxes through the faces b' and d'. At the faces $p1$ and $p2$ the gradient of the Cartesian velocity components u and v and the gradient of the temperature T has to be evaluated. Following Müller and Rizzi[9] and others, the gradient of an arbitrary function φ at the point p' (see Fig. 2) can, in the TSL limit, be approximated by

$$\nabla\varphi\simeq\frac{1}{A'}\int_{\omega'}\nabla\varphi\ dA=\frac{1}{A'}\oint_{\partial\omega'}\varphi\ ds\simeq\frac{1}{A'}\sum_{k=1}^{2}\varphi k S'_k$$

where the line integral is carried out around the cell with thick lines (see Fig. 2). The cells used for gradient calculations form a second mesh that is shifted a half cell in the normal direction in relation to the original computational mesh.

To simulate a turbulent boundary layer, a Baldwin-Lomax[11] two-layer turbulence model is implemented. Since the finite-volume scheme, Eq. (5), is nondissipative, an artificial fourth-difference damping term is added.

The semidiscretized Euler and TSL Navier-Stokes equation, Eq. (5), forms a large system of ODEs:

$$\begin{cases}\dfrac{dq_{ij}}{dt}=R_{ij}\\ q_{ij}(0)=q^0_{ij}\end{cases} \tag{6}$$

which is solved using an explicit Runge-Kutta method for the Euler equations and an implicit Beam-Warming method for the TSL Navier-Stokes equations.

To optimize the computational speed and hence minimize the computational cost, a fast and computationally relatively inexpensive explicit Runge-Kutta method is used in the large inviscid core flow region. In this work a three-stage Runge-Kutta scheme for ODEs is used. The scheme is defined by

$$\begin{cases}q^* = q^n-\Delta t\ R(q^n)\\ q^{**}=q^n-1/2\Delta t\ [R(q^n)+R(q^*)]\\ q^{n+1}=q^n-1/2\Delta t\ [R(q^n)+R(q^{**})]\end{cases}$$

The algorithm is second-order accurate and has a CFL limit of 2.

To avoid the problem of having a severe CFL restriction in the viscous boundary-layer region due to cells with large aspect ratios, an implicit algorithm is used. Beam and Warming[12] suggested the following scheme to solve a system of ODEs such as Eq. (6):

$$\frac{3}{2\Delta t}\Delta q^n-\frac{\psi}{2\Delta t}\Delta q^{n-1}=-R(q^{n+1}) \tag{7}$$

where

$$\Delta q^n=q^{n+1}-q^n$$

After linearization of the right-hand side and splitting the resulting Jacobian into I and J directions (matrices A and B), the scheme becomes

$$\left(I+\frac{2\Delta t}{3}\delta_i A+\frac{2\Delta t}{3}\delta_j B\right)\Delta q=-\frac{2\Delta t}{3}R^n+\frac{1}{3}\Delta q^{n-1} \tag{8}$$

For a hyperbolic model equation this scheme is unconditionally stable. For flows with strong nonlinearities, however, the scheme can have a CFL restriction based on streamwise grid spacing of as low as 10.

Boundary Conditions

For cascade geometries there are two types of boundary conditions, namely solid wall conditions at the blade and far-field conditions at the inlet and outlet. For the mesh topologies used in this work an additional boundary condition is needed in the interface between the viscous and the inviscid region.

The no-slip condition at a solid, impermeable wall requires

$$u_w=v_w=0$$

where the subscript w denotes the wall value. The temperature boundary condition can be of two different types. It can be either a specified wall temperature as a function of the arc length:

$$T_w=T_w(l)$$

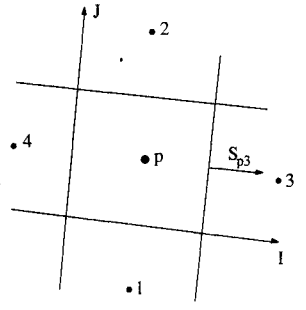


Fig. 1 Identifying cell centers in the computational mesh.

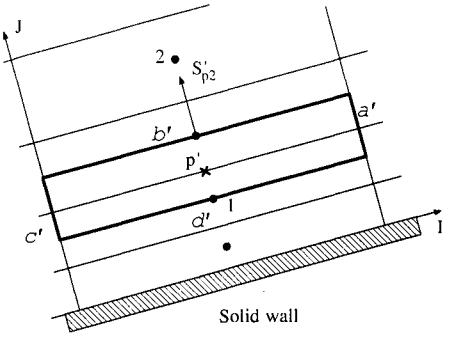


Fig. 2 Viscous cells close to a solid wall.

or a zero heat-flux condition for an adiabatic wall:

$$\left(\frac{\partial T}{\partial n}\right)_w = 0$$

where n denotes the wall-normal.

At the inflow and outflow boundaries the method described by Giles[13] is used. The method is based on an analysis of the locally linearized, unsteady one-dimensional Euler equations. The linearized one-dimensional Euler equations can be written as

$$\frac{\partial \Delta q}{\partial t} + A_0 \frac{\partial \Delta q}{\partial x} = 0 \tag{9}$$

where A_0 is a function of the steady flow variables and $\Delta q = q - q_0$ is the perturbation from the steady flow. Since A_0 is nondefective, it can be diagonalized by a similarity transform. The transformed equation becomes

$$\frac{\partial \Phi}{\partial t} + \Lambda_0 \frac{\partial \Phi}{\partial x} = 0 \tag{10}$$

where the solution $\Phi = T_0^{-1}\Delta q$ is the vector of linear characteristics and $\Lambda_0 = T_0^{-1}A_0T_0 = \text{diag}[u, u+a, u-a]$ is a diagonal matrix with the eigenvalues of A_0 on the diagonal, a being the local speed of sound. By assuming no variation in the y direction, in the two-dimensional case the fourth equation becomes

$$\frac{\partial \phi_4}{\partial t} + u \frac{\partial \phi_4}{\partial x} = 0 \tag{11}$$

and the four eigenvalues become

$$\lambda_0 = \text{diag}[u, u+a, u-a, u]$$

The eigenvalues now determine the slope of the isoclines corresponding to each characteristic variable $\Phi = [\phi_1, \phi_2, \phi_3, \phi_4,]^T$ in the x,t space. For a subsonic case, characteristic variables corresponding to the eigenvalues u, u and $u+a$ will have isoclines with positive slope, and the characteristic variable corresponding to $u-a$ has an isocline with negative slope. To prevent spurious wave reflections at the inflow boundary, ϕ_1, ϕ_2, and ϕ_4 are specified to be zero, and ϕ_3 is extrapolated from the interior. At the outflow ϕ_3 is specified to be zero, and the other three are extrapolated from the interior.

To allow for specified unsteadiness such as hot streaks at the inflow boundary, we require modification of the characteristic variables ϕ_1, ϕ_2, ϕ_3, and ϕ_4 so that these are perturbations of some known inlet disturbance. Explicitly the characteristic variables are

$$\begin{aligned}
\phi_1 &= (\Delta\rho) - \frac{\Delta p}{a_{is}^2} \\
\phi_2 &= (\Delta u) + \frac{\Delta p}{\rho_{is} a_{is}} \\
\phi_3 &= -(\Delta u) + \frac{\Delta p}{\rho_{is} a_{is}} \\
\phi_4 &= \Delta v
\end{aligned} \tag{12}$$

where $\Delta\rho = \rho - \rho_{\text{inl}}$, $\Delta u = u - u_{\text{inl}}$, $\Delta v = v - v_{\text{inl}}$, and $\Delta p = p - p_{\text{inl}}$. The subscript inl refers to the prescribed unsteady inlet value and the subscript *is* refers to the steady flow variables. To be consistent with the physical problem, it is assumed that in the wake frame the prescribed flow is parallel. It is also assumed that the Mach number and the static and total pressures in the hot streak are the same as in the free flow. The initial temperature excess used by Butler et al.[1] is approximately a square well. However, since we will only compute the field around the rotor, we assume that the square well after convection through the stator row will have been diffused. The smoothed temperature distribution is given by a cubic spline function $W(\xi)$ that has a smooth transition region spanning over 10% of the pitch. Under these assumptions the inlet flow quantities are given by

$$\begin{aligned}
p_{\text{inl}} &= p_{is} \\
T_{\text{inl}} &= T_{is}\,\{1 + DW[s(\eta)]\} \\
\rho_{\text{inl}} &= \rho_{is}\,\{1 + DW[s(\eta)]\}^{-1} \\
u_{\text{inl}} &= u_{is}\,\{1 + DW[s(\eta)]\}^{1/2} \\
v_{\text{inl}} &= (v_{is} + V)\,\{1 + DW[s(\eta)]\}^{1/2} - V
\end{aligned} \tag{13}$$

The subscript inl denotes inlet hot-streak values, *is* denotes the inlet-steady values obtained from a steady-state calculation in the rotor frame, M_{in} is the inlet Mach number, D is the fractional temperature excess, W is the smooth distribution function, and V is the rotor velocity, which is needed to convert between wake-relative and rotor-relative velocities.

Interface Region

As long as each cell in the computational domain has only four closest neighbor cells, the previously described spatial finite-volume operator will always be consistent, which implies at least first-order accuracy. The most critical point along the intersection between the two mesh topologies is the point where the H-mesh splits and is joined by the O-mesh. As can be seen in Fig. 3, all cells in this critical region have only four faces and hence only four closest neighbors. The spatial operator along the interface line will always have four points in one region and one point in the other. Provided that the interface line is defined to be located outside the boundary layer, i.e., the viscous terms in the last layer of cells in the O-mesh can be neglected, the steady-state spatial operator is consistent along this line.

The temporal matching between the viscous and inviscid regions can be described as follows:

1) Calculate the flux through the interface between the inviscid and viscous region at time level n, i.e., $F^n_{\text{int}} = 1/2\,(F^n_{\text{visc}} + F^n_{\text{inv}})$.

2) Make a predictor step with the Runge-Kutta formula throughout the inviscid mesh, $q^* = q^n + \Delta t R^n$ (Euler forward step).

3) Update the boundary cells in the viscous region using the extrapolation formula $q^* = q^n + \Delta q^{n-1}$.

4) Calculate the flux through the interface between the inviscid and viscous region at time level $n+1$, i.e., $F^*_{\text{int}} = 1/2\,(F^*_{\text{visc}} + F^*_{\text{inv}})$.

5) Make a corrector step with the Runge-Kutta formula throughout the inviscid mesh, $q^{**} = q^n + 1/2\Delta t(R^* + R^n)$.

6) Calculate the new flux through the interface between the

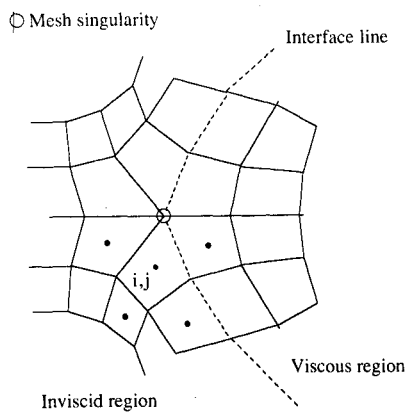


Fig. 3 Points in the spatial finite-volume operator and mesh singularities.

inviscid and viscous region at time level $n+1$, i.e., $F^{**}_{int} = 1/2F^{**}_{visc} + F^{**}_{int}$).

7) Make a final corrector step with the Runge-Kutta formula throughout the inviscid mesh, $q^{n+1} = q^n + 1/2\Delta t\,(R^{**} + R^n)$.

8) Save the flux through the interface calculated by $F^{n+1}_{int} = 1/2(F^n_{int} + F^{**}_{int})$.

9) Make an implicit step throughout the viscous region using the interface flux calculated in the explicit step.

Note that the extrapolation update done in the viscous region is not used as a permanent update but just to enforce consistency in the inviscid region. Note also that the solution at substages * and ** are not solutions at separate time levels but are in fact predicted and corrected solutions at time level $n+1$.

The fourth difference smoothing of the fluxes is imposed at each stage of the explicit scheme and in the beginning of the implicit step. To smooth the last two cells before the interface in both the regions, two points in the other region are needed for the five-point smoothing stencil. Two layers of cells next to the interface in each region are buffered and stored independently to be used for smoothing. The buffer values in the viscous region follow the updating described above, and therefore the smoothing is always performed at the correct time level.

The algorithm described above is second-order accurate everywhere except in the interface region. In the interface region, however, the scheme is consistent and therefore at least first-order accurate since the intermediate extrapolation update ($q^{n+1} = q^n + \Delta q^{n-1}$) in the viscous region is consistent. The algorithm just presented is adequate in the interface region up to the second corrector step when the fluxes through the interface of the boundary cell are based on the predicted value in the viscous region rather than on the corrected values. The first step is identical on both sides of the interface boundary since the Euler step and the extrapolation step are both first-order accurate. However, the errors introduced to the scheme by averaging at different stages that might give local first-order accuracy are not going to affect the global second-order accuracy of the full scheme. This statement is based on a theorem by Gustafsson[14] that states that for convergence, the accuracy of the boundary conditions to an interior scheme can be of one order less than the scheme itself.

Results

The two-dimensional configuration on which the calculations are performed is taken to be the midspan section of the experimental setup used by Butler et al.[1] The original experiment was run at almost incompressible conditions with an outflow Mach number of approximately 0.2. The steady inflow and outflow boundary conditions, upon which the unsteady boundary conditions are based, were obtained by performing a steady computation using the inviscid Euler solver UNSFLO.[13] The unsteady computations were started from a field initialized by averaging the inflow and outflow steady conditions.

To be able to compare the results both with the experimental data and with previous calculations, two different simulations were performed. The test cases were run at two different flow coefficients ϕ and temperature ratios λ where

$$\phi = \frac{u_{inlet}}{V_{rot}}, \qquad \lambda = \frac{T_{hot}}{T_{cold}}$$

In the first test case the flow coefficient was $\phi = 0.78$ and the temperature ratio was $\lambda = 1.2$, which is consistent with Rai and Dring's[3] simulation. The second computation used $\phi = 0.68$ and $\lambda = 2.0$, which correspond to the actual experiment. In both test cases the boundary layer was assumed to be fully turbulent. Furthermore, it was assumed that the blade wall is adiabatic, i.e., there is no heat flux at the blade surface. In both computations the Reynolds number, based on axial chord and inlet speed of sound, was set to 10^6 to match the experimental data as well as Rai and Dring's computational data.

The two computations were performed on the same computational mesh. An O-mesh was generated in the viscous region close to the blade, and an H-mesh was generated in the inviscid through flow region. The H-mesh was created using an elliptic grid generator and the O-mesh using an algebraic technique using splines. The two meshes were patched together and the interface region was smoothed in order to prevent numerical errors. The inviscid H-mesh consisted of 97×51 mesh points and the viscous O-mesh of 129×21 mesh points, which gave a total of 7656 mesh points. For the O-mesh the stretching factor in the normal direction was 1.1. The mesh is shown in Fig. 4. To get good temporal resolution, 3000 iterations were performed for each cycle (blade passing). For all test cases, this gave a CFL restriction corresponding to CFL = 1.2 in the inviscid region and CFL = 4.50 in the viscous region. Seven cycles were needed to obtain a periodically converged solution from the freestream condition. All computations are performed on an ETA 10-P supercomputer. The computational cost is 3.5 CPU seconds per iteration, which gives a total of 3.0 CPU hours per cycle.

Case 1: $\phi = 0.78$ and $\lambda = 1.2$

To be consistent with the computation performed by Rai and Dring,[3] the first test case was run with a flow coefficient $\phi = 0.78$. It was also assumed that the flow around the blade was fully turbulent and that the ratio between hot and cold gas was 1.2. Figure 5 shows the time-averaged pressure coefficient ($\bar{C}_p$) as a function of the normalized axial distance x/c. The pressure coefficient is defined as

$$\bar{C}_p = \frac{(p_{ave} - p_{ex})}{P_{0inl} - P_{ex}} \tag{14}$$

where p_{ave} is the static pressure time-averaged over one cycle, P_{0inl} is the inlet total pressure in the rotor frame, and p_{ex} is the exit static pressure.

Figure 5 also shows an experimentally obtained pressure distribution,[1] which was obtained without any incoming hot streaks. The agreement between computed and experimental data is very good. Also indicated in Fig. 5 is the maximum and minimum C_p on the blade over one cycle. One can clearly see that the case is truly unsteady.

Figure 6 shows the time-averaged temperature coefficient $\bar{C}_T$ along the blade surface. The surface length is nondimensionalized with respect to the axial chord c. The temperature

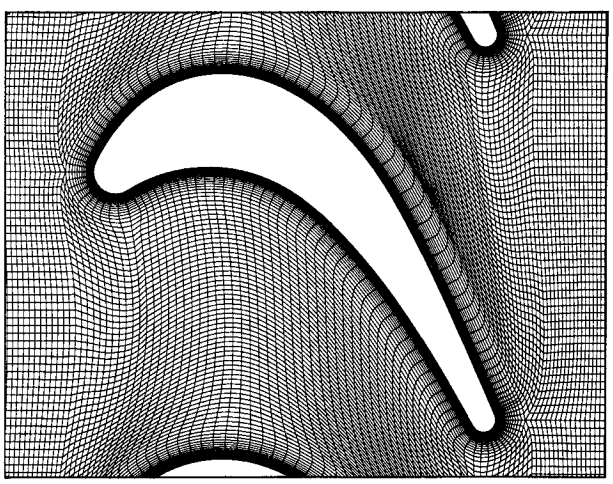


Fig. 4 Computational mesh around the rotor blade.

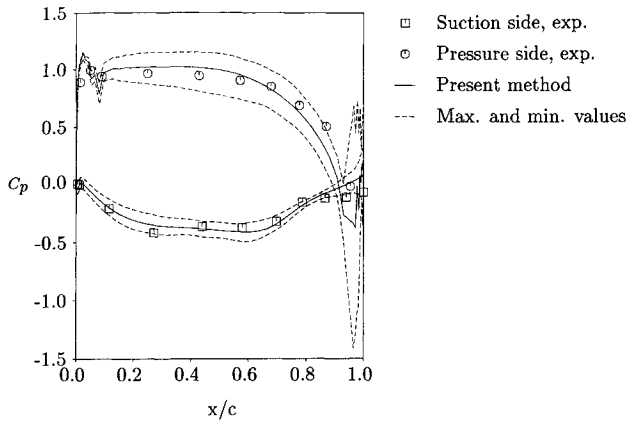


Fig. 5 Time-averaged surface pressure coefficient at $\phi = 0.78$ and $\lambda = 1.2$.

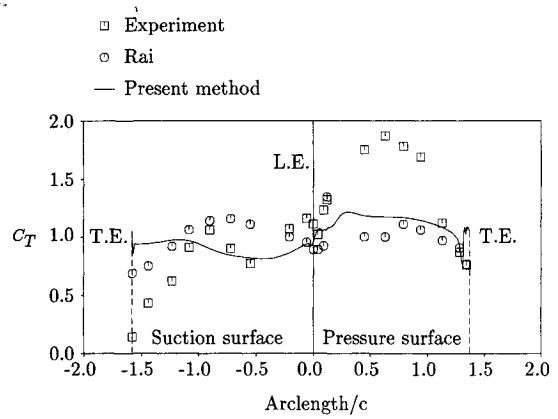


Fig. 6 Time-averaged surface temperature coefficient at $\phi = 0.78$ and $\lambda = 1.2$.

coefficient is defined as

$$\bar{C}_T = \frac{T - T_\infty}{T_{av} - T_\infty} \tag{15}$$

where T_{av} is the area averaged inlet temperature and T_∞ is the temperature in the unheated freestream. It can be seen that the temperature coefficient varies around 1 as expected. A peak in the temperature appears in the beginning of the pressure surface where the hot streak intersects with the blade. Following the pressure surface, the average temperature slowly rises from 1.15 to about 1.4 and finally fills beneath 1 at the trailing edge. On the suction surface the average temperature has a dip at the forward part and, then slowly increases to reach a maximum of about 1.0 at the aft part the blade. Finally the temperature decreases to a value of 0.9 at the trailing edge.

Figure 6 also shows the numerical results of Rai and Dring and the experimental results. The results obtained with the present method are in close agreement with Rai and Dring's data. On the suction surface, however, the present solution shows a lower value than the one obtained by Rai and Dring. With the flow conditions specified for this test case, neither the present method nor Rai and Dring's method predict the same amount of migration as was experimentally observed.

Case 2: $\phi = 0.68$ and $\lambda = 2.0$

In order to match the actual experiment more closely, the flow coefficient for the second test case was set to be 0.68, which corresponds to a higher rotational speed and hence a higher tangential inflow angle. The temperature ratio λ was increased to 2.0, which was the same as in the experiment.[1] A comparison between the numerically and the experimentally obtained time-averaged pressure coefficient $\bar{C}_p$ is shown in

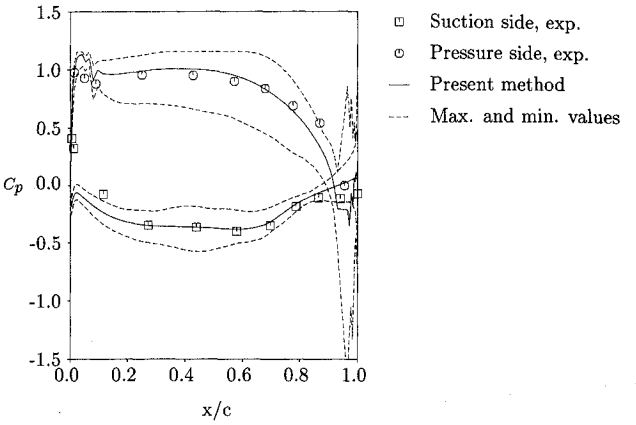


Fig. 7 Time-averaged surface pressure coefficient at $\phi = 0.68$ and $\lambda = 2.0$.

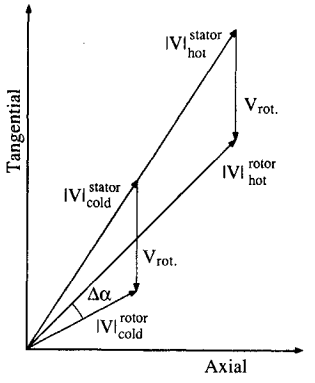


Fig. 8 Velocity triangles for hot and cold fluid.

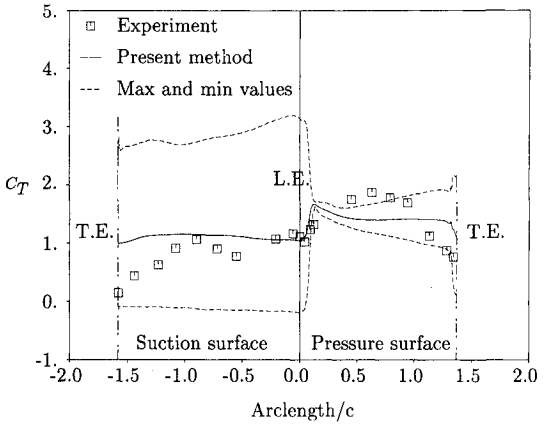
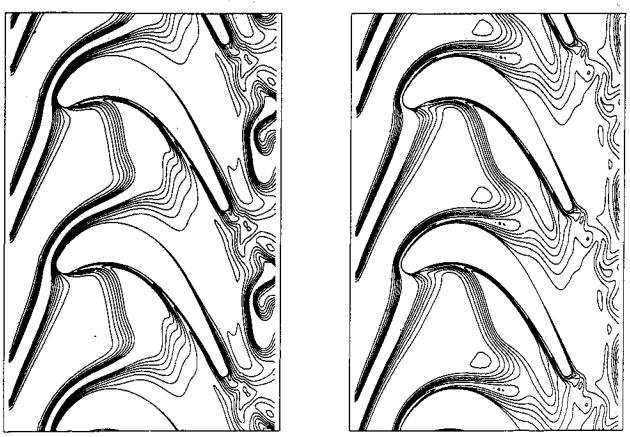


Fig. 9 Time-averaged maximum and minimum surface temperature coefficient at $\phi = 0.68$ and $\lambda = 2.0$.

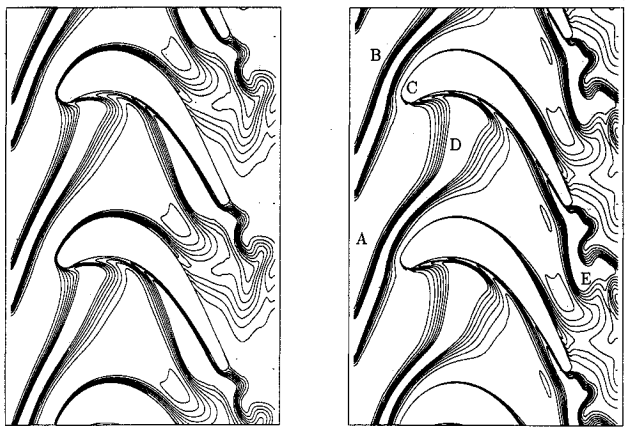
Fig. 7. The pressure coefficient was defined in Eq. (14). As for the case with $\phi = 0.78$, the experimental data were obtained without incoming hot streaks. The agreement between experimental and computational data is good with a small difference in the leading-edge region. The differences can be explained by the fact that the rotor relative flow angle is different in the hot and cold fluid. This can be understood by considering the velocity triangles shown in Fig. 8. Also indicated is the difference in flow angle $\Delta\alpha$ between the hot and cold fluid in the rotor frame of reference. A different time-averaged angle of attack will give rise to α different pressure distribution. The random fluctuations in the pressure at the end of the blade correspond to the points on the trailing-edge circle and are probably caused by numerical errors.

Figure 9 shows the averaged temperature coefficient as a function of the rotor blade surface length. Near the leading



a) b)

Fig. 10 Temperature contours: a) $t = 0.25$; b) $t = 0.5$.



a) b)

Fig. 11 Temperature contours: a) $t = 0.75$, b) $t = 1.0$.

edge on the pressure surface there is a peak in the temperature indicating the position where the hot streak first hits the blade. From this peak, the temperature decreases along the pressure surface. It starts at a value of 1.7 behind the peak and approaches 1.4 asymptotically to finally drop to 1.0 at the trailing-edge circle. On the suction surface the averaged temperature is almost constant at a value of 1.0.

The experimental data are also shown in Fig. 9. The agreement between the experimental data and the computational results in this second case is much better than in the previous. The peak averaged temperature, however, does not reach the same high value in the computation as in the experiment. It is clear that the higher tangential flow angle α, due to a lower ϕ and the higher temperature ratio, indeed does accentuate the segregation of hot and cold gas between the pressure and suction surface. Indicated in Fig. 9 is also the maximum and minimum temperature along the blade for a cycle. It can be seen that the temperature differences are greatest on the suction side of the blade. Due to the fast convection of the hot streak, there is a rapid variation with a big amplitude of the temperature on the suction surface of the blade.

One of the big advantages of using a computational approach is that it yields, as a part of the calculation, not only surface temperatures but also the temperature distribution in the entire computational region. Figures 10a, 10b, 11a, and 11b show four snapshots of the temperature field in the rotor at four different times in the cycle: 25, 50, 75, and 100% of the cycle time. In all figures the temperature is represented by isotherms with the lower temperature levels filtered out. The hot streaks are in between the two narrow bands of isotherms representing the transition between hot and cold gas.

Figure 11b shows the solution at time level 1.0, which is equal to the solution at time level 0.0. Five different phases of the hot-streak redistribution can be identified. The first is the left-most stage in the inflow, where the hot streak is being connected towards the blade row, indicated by A in the figure. The second stage is close to the leading edge where the hot streaks have started to interact with the pressure perturbation from the blade. A first indication of bending of the streak can be noticed at station B in the figure. In the third stage the hot streak has impinged on the pressure surface of the blade (station C) and is slowly being convected along the surface. Due to the acceleration gradient in the high-speed region between the pressure and suction surface of two consecutive blades the sreaks startes to deform (D). This is referred to as bowing. In stage four the hot streak has been wrapped around the blade and the outer contours downstream of the trailing edge are being affected by the vortex shedding at the trailing edge (E). At this stage it can be seen that the redistribution of the hot streak is dominated by convection on the suction surface and by a combination of convection and heat diffusion on the pressure surface.

Conclusions

In this work a numerical algorithm for stimulating an unsteady, two-dimensional flow in a turbine rotor has been developed. The flow region is divided into two subregions with different flow characteristics. In the viscous region close to the blade, the unsteady, Reynolds-averaged, thin-layer Navier-Stokes equations are solved on an O-mesh using an implicit time integration method. In the inviscid core flow region, the unsteady Euler equations are solved on an H-mesh using an explicit time integration method. The two meshes are patched together and a first-order method is used to link the two methods in the interface region.

The numerical method was used to investigate the redistribution of hot streaks in a turbine rotor. Two different flow conditions were investigated. The first test case used a tangential inflow angle of 40 deg and a temperature ratio of 1.2. The result was in agreement with data from a previous numerical investigation.[3] In the second test case, the tangential inflow angle was 45 deg and the temperature ratio was 2.0. The computed solutions predicted a migration of hot gas to the pressure surface, which also has been observed experimentally.[1] The magnitude of hot gas migration to the pressure surface is significantly dependent on the differences in flow coefficient and temperature ratio. The differences between experiment and computation are due primarily to the three-dimensional nature of the flow that cannot be simulated by a two-dimensional code.

References

[1]Butler, T. L., Sharma, O. P., Joslyn, H. D., and Dring, R. P., "Redistribution of an Inlet Temperature Distortion in an Axial Flow Turbine Stage," AIAA Paper 86-1468, June 1986.

[2]Kerrebrock, J. L., and Mikolajczak, A. A., "Intra-Stator Transport of Rotor Wakes and Its Effect on Compressor Performance," *ASME Journal of Engineering for Power*, Oct. 1970.

[3]Rai, M. M., and Dring, R. P., "Navier-Stokes Analyses of the Redistribution of Inlet Temperature Distortions in a Turbine," AIAA Paper 87-2146, July 1987.

[4]Stabe, R. G., Whitney, W. J., and Moffitt, T. P., "Performance of a High Work Low Aspect Ratio Turbine Tested with a Realistic Inlet Radial Temperature Profile," AIAA Paper 84-1161, June 1984.

[5]Norton, R. J. G., Thompkins, W. T., and Haimes, R., "Implicit Finite Difference Schemes with Non-Simply Connected Grids—A Novel Approach," AIAA Paper 84-0003, Jan. 1984.

[6]Nakahashi, K., and Obayashi, S., "Viscous Flow Computations Using a Composite Grid," AIAA Paper 87-1128, June 1987.

[7]Nakahashi, K., Nozaki, O., Kikuchi, K., and Atsuhiro, T., "Navier-Stokes Computations of Two- and Three-Dimensional Cascade Flow Fields," AIAA Paper 87-1315, June 1987.

[8]Eriksson, L-E., "Simulation of Inviscid Flow Around Airfoils and Cascades Based on the Euler Equations," Aeronautical Research Institute of Sweden, Bromma, Sweden, TN 1985-20, 1985.

[9]Müller, B., and Rizzi, A. W., "Runge-Kutta Finite-Volume Simulations of Laminar Transonic Flow Over a NACA0012 Airfoil Using the Navier-Stokes Equations," Aeronautical Research Institute of Sweden, Bromma, Sweden, TN 1986-60, 1986.

[10]Swanson, C., and Turkel, E., "A Multi-Stage Time-Stepping Scheme for the Navier-Stokes Equations," AIAA Paper 5-0035, Jan. 1985.

[11]Baldwin, B. S., and Lomax, H., "Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows," AIAA Paper 78-257, Jan. 1978.

[12]Beam, R., and Warming, R., "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations," *AIAA Journal*, Vol. 16, No. 4, 1978, pp. 393–402.

[13]Giles, M. B., "UNSFLO: A Numerical Method for Unsteady Inviscid Flow in Turbomachinery," Computational Fluid Dynamics Lab., Massachusetts Inst. of Tech., Cambridge, MA, CFDL-TR-86-6, Dec. 1986.

[14]Gustafsson, B., "The Convergence Rate for Difference Approximations to Mixed Initial Boundary Value Problems," *Mathematical Computations*, Vol. 29, No. 130, April 1975, pp. 396–406.