AIAA 97-1828

Preconditioning for Numerical Simulation of Low Mach Number Three-Dimensional Viscous Turbomachinery Flows

Daniel L. Tweedt[†] and Rodrick V. Chima[‡]
NASA Lewis Research Center
Cleveland, Ohio 44135

Eli Turkel[§]
School of Mathematical Sciences
Tel-Aviv University, Tel-Aviv, Israel
and ICOMP, NASA Lewis Research Center

Abstract

A preconditioning scheme has been implemented into a three-dimensional viscous computational fluid dynamics code for turbomachine blade rows. The preconditioning allows the code, originally developed for simulating compressible flow fields, to be applied to the nearly-incompressible, low Mach number flows. A brief description is given of the compressible Navier-Stokes equations for a rotating coordinate system, along with the preconditioning method employed. Details about the conservative formulation of artificial dissipation are provided, and two different artificial dissipation schemes are discussed and compared. The preconditioned code was applied to a well-documented case involving the NASA large low-speed centrifugal compressor for which detailed experimental data are available for comparison. Performance and flow field data are compared for the near-design operating point of the compressor, with generally good agreement between computation and experiment. Further, significant differences between computational results for the different numerical implementations, revealing different levels of solution accuracy, are discussed.

Introduction

The general engineering discipline of turbomachinery fluid dynamics involves a wide spectrum of practical devices and machines, many of which involve incompressible or nearly incompressible fluid flows. A list of such devices would include, among others, pumps, hydraulic turbines, propellers, low-speed fans and blowers, and low-speed experimental test rigs. Despite this abundance of incompressible-flow turbomachinery, and other incompressible flows in general, it is an interesting historical facet of computational fluid dynamics (CFD) that a great many computer codes for fluid dynamic simulation have been developed based on numerical algorithms intended only for compressible flows. One reason for this is that most state-of-the-art fluids engineering is being performed in the aeropropulsion industry where the fluid dynamics is mostly compressible, and where new technology is worth the high cost of expensive research and development, including CFD code development.

It is well known that most compressible-flow CFD codes will not converge to an acceptable flow solution when the flow field Mach numbers become too low. Typically this occurs somewhere below about Mach number 0.1 where the gas becomes virtually incompressible; that is, the fluid density becomes nearly constant with changes in flow velocity. Within the past several years, however, methods for altering the compressible-flow numerical algorithm to allow convergence at very low Mach numbers have increasingly appeared in the literature [1-10]. These so-called preconditioning methods alter the eigenvalues of the system of compressible-flow equations so as to reduce, at low Mach numbers, the large disparity between the acoustic and convective wave speeds. Although it is possible to avoid this problem altogether by developing the numerical algorithm to solve the incompressible flow

---

[†] Aerospace Engineer

[‡] Aerospace Engineer, Associate Fellow AIAA

[§] Professor, Department of Mathematics, Member AIAA equations, the preconditioned compressible codes have several advantages and/or benefits, not the least of which is that many compressible-flow codes already exist for various applications. It is relatively easy to add a preconditioning scheme to an already existing code. Other advantages include code versatility and the capability to directly simulate flow fields involving both compressible and incompressible flow regions.

In this paper, a three-dimensional viscous CFD code originally developed for the simulation of the compressible flow field within a turbomachine blade row is considered. The code, designated RVC3D, solves a thin-layer formulation of the Reynolds-averaged three-dimensional Navier-Stokes equations [11, 12], and uses the Baldwin-Lomax algebraic turbulence model [13] to simulate the effects of boundary layer turbulence. Preconditioning has been incorporated into the code by implementing the work of Turkel [3, 4], Radespiel and Turkel [7], Radespiel et al. [6], and Turkel et al. [10]. The paper describes the numerical method, including the preconditioning scheme and some problems encountered, and then shows computed results for a centrifugal compressor impeller compared to detailed experimental data.

Governing Equations

The governing equations which are numerically integrated using the RVC3D code are summarized below. Only the thin-layer formulation of the viscous compressible equations as transformed to a generalized body-fitted coordinate system are presented, since further and more complete details can be found in the references for RVC3D [11, 12].

The Navier-Stokes equations are written for a Cartesian (*x, y, z*) coordinate system rotating with angular velocity Ω about the *x*-axis. The Cartesian equations have been mapped to a body-fitted (ξ, η, ζ) coordinate system, simplified using the thin-layer approximation, and nondimensionalized by arbitrary reference quantities ρ_0, a_0, and μ_0. Note that the ξ-coordinate direction is roughly parallel to the blade surface and wraps around it, while the η-direction is almost normal to the blade surface and the ζ-direction runs along the blade span. The resulting equations can be summarized as follows:

$$\partial_t \hat{q} + \partial_\xi \hat{E} + \partial_\eta \hat{F} + \partial_\zeta \hat{G} - Re^{-1}(\partial_\eta \hat{F}_V + \partial_\zeta \hat{G}_V) = \hat{H} \tag{1}$$

where

$$\hat{q} = J^{-1} q = J^{-1} \left[\rho,\ \rho u,\ \rho v,\ \rho w,\ e\right]^T$$

$$\hat{E} = J^{-1} \begin{bmatrix} \rho U' \\ \rho u U' + \xi_x p \\ \rho v U' + \xi_y p \\ \rho w U' + \xi_z p \\ e U' + p U \end{bmatrix} \qquad \hat{F} = J^{-1} \begin{bmatrix} \rho V' \\ \rho u V' + \eta_x p \\ \rho v V' + \eta_y p \\ \rho w V' + \eta_z p \\ e V' + p V \end{bmatrix}$$

$$\hat{G} = J^{-1} \begin{bmatrix} \rho W' \\ \rho u W' + \zeta_x p \\ \rho v W' + \zeta_y p \\ \rho w W' + \zeta_z p \\ e W' + p W \end{bmatrix} \qquad \hat{H} = J^{-1} \Omega \begin{bmatrix} 0 \\ 0 \\ -\rho w \\ \rho v \\ 0 \end{bmatrix}$$

Details of the viscous flux vectors $\hat{F}_V$ and $\hat{G}_V$ are given in References 11 and 12. The absolute velocity components *u*, *v* and *w* point in the *x, y,* and *z* coordinate directions, respectively, and the relative contravariant velocity components are given by

$$\begin{aligned} U' &= \xi_x u + \xi_y v' + \xi_z w' \\ V' &= \eta_x u + \eta_y v' + \eta_z w' \\ W' &= \zeta_x u + \zeta_y v' + \zeta_z w' \end{aligned} \tag{2}$$

where the relative velocity components are:

$$\begin{aligned} u' &= u \\ v' &= v - \Omega z \\ w' &= w + \Omega y \end{aligned} \tag{3}$$

Assuming an ideal gas with constant specific heats, then $\gamma\ (= c_p / c_v\ )$ is constant, and the energy and static pressure are given by

$$e = \rho\left[\frac{a^2}{\gamma(\gamma - 1)} + \frac{1}{2}(u^2 + v^2 + w^2)\right] \tag{4}$$

$$p = (\gamma - 1)\left[e - \frac{1}{2}\rho(u^2 + v^2 + w^2)\right] \tag{5}$$

where the sonic velocity *a* is related to static pressure and density by the equation of state:

$$a^2 = \frac{\gamma p}{\rho} \tag{6}$$

The metric terms are defined by the following relations:

$$\begin{bmatrix} \xi_x & \eta_x & \zeta_x \\ \xi_y & \eta_y & \zeta_y \\ \xi_z & \eta_z & \zeta_z \end{bmatrix} = J \begin{bmatrix} y_\eta z_\zeta - y_\zeta z_\eta & y_\zeta z_\xi - y_\xi z_\zeta & y_\xi z_\eta - y_\eta z_\xi \\ x_\zeta z_\eta - x_\eta z_\zeta & x_\xi z_\zeta - x_\zeta z_\xi & x_\eta z_\xi - x_\xi z_\eta \\ x_\eta y_\zeta - x_\zeta y_\eta & x_\zeta y_\xi - x_\xi y_\zeta & x_\xi y_\eta - x_\eta y_\xi \end{bmatrix} \tag{7}$$

where the Jacobian J and its inverse J^{-1} are

$$\begin{aligned} J^{-1} = {} & x_\xi y_\eta z_\zeta + x_\zeta y_\xi z_\eta + x_\eta y_\zeta z_\xi \\ & - x_\xi y_\zeta z_\eta - x_\eta y_\xi z_\zeta - x_\zeta y_\eta z_\xi \end{aligned} \tag{8}$$

The inlet (inflow) boundary condition for the above system of equations involves specification of the spanwise distributions of total pressure, total temperature, and absolute circumferential velocity component, v_θ. The upstream-running Riemann-invariant based on the meridional velocity component is extrapolated upstream from the interior to the inlet boundary.

At the exit boundary four of the five conservation variables, namely ρ, ρu, ρv, and ρw, are extrapolated downstream to the boundary, and the static pressure is specified at the inner (hub) boundary and integrated in the spanwise direction using simple radial equilibrium:

$$\frac{dp}{dr} = \frac{\bar{\rho}\bar{v}_\theta^2}{r} \tag{9}$$

where r is local radius and the overbar denotes a circumferential-averaged quantity. The resulting local spanwise static pressures are then used either as constants (circumferential invariant exit static pressures), or as nominal values for calculating circumferential pressure distributions using a method described by Giles [14].

Circumferential-periodic boundaries between the blades are solved like interior points using a dummy grid line outside of the domain. At blade and endwall surfaces, the no-slip, adiabatic wall boundary condition is used (for viscous simulations).

Numerical Method

The governing equations are discretized using a node-centered finite-difference scheme, with second-order central differences used throughout. The multistage Runge-Kutta scheme developed by Jameson, Schmidt, and Turkel [15] is used to advance the flow equations in time from an initial guess to a steady state. With the exception of the 5-stage scheme, the physical and artificial dissipation terms are calculated only at the first stage and then held constant for subsequent stages. Further details of the Runge-Kutta solution scheme can be found in References 11 and 12.

The time step limit resulting from a three-dimensional linear stability analysis can be expressed as follows:

$$\Delta t \le \frac{\Lambda^*}{\Delta t_\xi^{-1} + \Delta t_\eta^{-1} + \Delta t_\zeta^{-1} + \Delta t_V^{-1}} \tag{10}$$

where Λ^* is the maximum Courant number for the particular multistage scheme. The inverse one-dimensional time steps in each grid direction are equal to the absolute value of the convective eigenvalues of largest magnitude (for the inviscid equations without preconditioning):

$$\begin{aligned} \Delta t_\xi^{-1} &= \lambda_\xi = |U'| + a\sigma_\xi \\ \Delta t_\eta^{-1} &= \lambda_\eta = |\mathrm{V}'| + a\sigma_\eta \\ \Delta t_\zeta^{-1} &= \lambda_\zeta = |\mathrm{W}'| + a\sigma_\zeta \end{aligned} \tag{11}$$

where

$$\begin{aligned} |U'| &= |\xi_x u| + |\xi_y \mathrm{v}'| + |\xi_z w'| \\ |\mathrm{V}'| &= |\eta_x u| + |\eta_y v'| + |\eta_z w'| \\ |\mathrm{W}'| &= |\zeta_x u| + |\zeta_y v'| + |\zeta_z w'| \end{aligned} \tag{12}$$

and

$$\begin{aligned} \sigma_\xi &= \sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2} \\ \sigma_\eta &= \sqrt{\eta_x^2 + \eta_y^2 + \eta_z^2} \\ \sigma_\zeta &= \sqrt{\zeta_x^2 + \zeta_y^2 + \zeta_z^2} \end{aligned} \tag{13}$$

The viscous time-step [16] contribution Δt_V^{-1} is given by

$$\Delta t_V^{-1} = k_t \frac{\gamma \mu}{\rho Pr} (\sigma_\xi^2 + \sigma_\eta^2 + \sigma_\zeta^2) \tag{14}$$

where the constant k_t is assigned a value of 4.0.

To accelerate convergence to a steady state, the maximum permissible time step at each grid point is used, giving a constant Courant number everywhere. To further accelerate convergence, implicit residual smoothing [17] is used to allow a time step typically two to three times larger than the stability limit given by Equation 10. Eigenvalue scaling [18] of the residual smoothing coefficients, and of the artificial dissipation where applicable, is employed and greatly increases the accuracy and robustness of the numerical scheme. A blending function nearly the same as one proposed by Kunz and Lakshminarayana [19] is used to blend the eigenvalues from all three directions.

Preconditioning

The implementation of preconditioning into the RVC3D code is based on the work of Turkel [3, 4], Radespiel and Turkel [7], Radespiel et al. [6], and Turkel et al. [10], where generalized preconditioning schemes have been developed and summarized. The most recent publications by the above authors present a generalization of the preconditioners originally introduced by Turkel [3, 4], and by Choi and Merkle [5], and they also present some further developments since then. In contrast to "standard" numerical schemes using a pseudo-compressibility approach, the preconditioners they describe not only accelerate the convergence of the numerical solution to a steady state, but also change the steady-state solution because of the choice of artificial viscosity terms. As discussed in Reference 2, the standard pseudo-compressibility schemes do not converge to the solution of the incompressible equations as the Mach number approaches zero. With proper preconditioning, however, the numerical scheme does behave appropriately at low Mach numbers. The recent publications also show that preconditioning can be combined with well-known convergence acceleration techniques such as residual smoothing and multigrid.

Only one of several possible schemes has been incorporated into the RVC3D code, which is a scheme closely resembling the preconditioner of Choi and Merkel [5]. The scheme as implemented has been formulated to maintain a conservative form of artificial dissipation.

For convenience, Equation 1 is rewritten in a terse finite-difference form:

$$\Delta \hat{q} = -\Delta t\,[R_I - (R_V + D)] \qquad (15)$$

where R_I is the inviscid residual including source term, R_V is the viscous residual, and D is the artificial dissipation term. The artificial dissipation term can be expanded as follows:

$$D = (\Delta_\xi \nabla_\xi + \Delta_\eta \nabla_\eta + \Delta_\zeta \nabla_\zeta)\,\hat{q} \qquad (16)$$

where each coordinate-direction appears explicitly, and ∇_ξ, ∇_η, and ∇_ζ, are operators defined later which depend on the type of artificial dissipation scheme. Denoting the preconditioning matrix by Γ, the preconditioned form of Equation 15 can be expressed as follows:

$$\Delta\hat{\omega} = -\Gamma\Delta t\left[R_I - \left\{R_V + (\Delta_\xi\Gamma^{-1}\nabla_\xi + \Delta_\eta\Gamma^{-1}\nabla_\eta + \Delta_\zeta\Gamma^{-1}\nabla_\zeta)\,\hat{\omega}\right\}\right] \qquad (17)$$

where the solution vector $\hat{q}$ has been replaced by $\hat{\omega}$:

$$\hat{\omega} = J^{-1}\omega = J^{-1}\left[p,\ u,\ v,\ w,\ h\right]^T \qquad (18)$$

and h is static enthalpy:

$$h = a^2/(\gamma - 1) \qquad (19)$$

The preconditioning matrix Γ and its inverse Γ^{-1} are given by

$$\Gamma = \frac{1}{\rho}\begin{bmatrix} \beta^2\rho & 0 & 0 & 0 & 0 \\ -u & 1 & 0 & 0 & 0 \\ -v & 0 & 1 & 0 & 0 \\ -w & 0 & 0 & 1 & 0 \\ \left(\beta^2 + \frac{q^2}{2} - \frac{a^2}{\gamma - 1}\right) & -u & -v & -w & 1 \end{bmatrix} \qquad (20)$$

$$\Gamma^{-1} = \begin{bmatrix} 1/\beta^2 & 0 & 0 & 0 & 0 \\ u/\beta^2 & \rho & 0 & 0 & 0 \\ v/\beta^2 & 0 & \rho & 0 & 0 \\ w/\beta^2 & 0 & 0 & \rho & 0 \\ \left(\frac{h}{\beta^2} + \frac{q^2}{2\beta^2} - 1\right) & \rho u & \rho v & \rho w & \rho \end{bmatrix} \qquad (21)$$

where $q^2 = u^2 + v^2 + w^2$, and for the rotating coordinate system

$$\begin{aligned} \beta^2 &= \min[\max(q'^2, k_p q'^2_{ref}), a^2] \\ q'^2 &= u^2 + v'^2 + w'^2 \end{aligned} \qquad (22)$$

The parameters k_p and q'_{ref} are specified constants, with k_p typically between 0.1 and 0.3 for viscous simulations, and q'_{ref} being a nominal reference velocity in the relative frame-of-reference.

The largest eigenvalues for the resulting preconditioned (inviscid) equations are, for each coordinate direction, the following:

$$\begin{aligned}\lambda_\xi &= \kappa|U'| + \sqrt{\kappa^2|U'|^2 + \beta^2\left(\sigma_\xi^2 - \frac{|U'|^2}{a^2}\right)}\\ \lambda_\eta &= \kappa|\mathrm{V}'| + \sqrt{\kappa^2|V'|^2 + \beta^2\left(\sigma_\eta^2 - \frac{|V'|^2}{a^2}\right)}\\ \lambda_\zeta &= \kappa|\mathrm{W}'| + \sqrt{\kappa^2|W'|^2 + \beta^2\left(\sigma_\zeta^2 - \frac{|W'|^2}{a^2}\right)}\end{aligned} \qquad (23)$$

where

$$\kappa = \frac{1}{2}\left(1 + \frac{\beta^2}{a^2}\right) \qquad (24)$$

and the relative contravariant velocity magnitudes $|U'|$, $|V'|$, and $|W'|$, are obtained from Equation 12. Note that the resulting condition number for this scheme is approximately 2.6 as Mach number approaches zero.

In the same way as for the unpreconditioned scheme, the above eigenvalues are used to determine the time step and to directionally scale the implicit residual smoothing and the artificial dissipation.

Theoretical considerations suggest that modifications to the boundary condition formulations might be needed in conjunction with the preconditioning scheme [6, 10]. Good numerical convergence behavior was observed for all cases investigated, however, without such modifications. Therefore no changes were made to any of the boundary condition formulations.

Artificial Dissipation Models

Two different artificial dissipation models are examined for use with the preconditioning scheme at low Mach numbers. The first model is that introduced by Jameson, Schmidt, and Turkel [15], and will be referred to as the baseline scheme. This scheme uses blended first- and third-order dissipation terms and is the one originally used in the RVC3D code as documented in the references [11, 12]. The second model involves scaling the artificial dissipation by the local minimum-eigenvalue. At low Mach numbers (below Mach 0.5) this scheme is identical to the Convective Upwind Split Pressure (CUSP) scheme introduced fairly recently by Jameson [20], and subsequently modified by Tatsumi, Martinelli, and Jameson [21, 22]. The Symmetric LImited Positive (SLIP) flux limiter introduced by Jameson [20] is used in connection with this scheme, which for brevity will be referred to in this paper simply as the SLIP scheme. A variation of the SLIP scheme is also introduced, the ASymmetric LImited Positive (ASLIP) scheme, which for the low Mach number levels considered appears to work slightly better than the SLIP scheme.

The baseline and the SLIP and ASLIP schemes can be described using the dissipation operators $\Delta_\xi\nabla_\xi$, $\Delta_\eta\nabla_\eta$, and $\Delta_\zeta\nabla_\zeta$ in Equation 16 as the starting point. Since each directional operator is similar to the others, only the ξ-direction need be considered. Also, note that q everywhere below is replaced by ω when preconditioning is applied.

Baseline Scheme

In the baseline scheme, the dissipation term $\Delta_\xi\nabla_\xi\hat{q}$ is finite-differenced according to

$$\begin{aligned}\Delta_\xi\nabla_\xi\hat{q} = {}& (J^{-1}C_\xi)_{i+1/2}\delta_{i+1/2} - \\ & (J^{-1}C_\xi)_{i-1/2}\delta_{i-1/2}\end{aligned} \qquad (25)$$

where the i-indexing corresponds to the ξ-direction and C_ξ is a coefficient obtained from blended eigenvalues [19]:

$$C_\xi = \lambda_\xi\sqrt{1 + \frac{\lambda_\eta + \lambda_\zeta}{\lambda_\xi}} \qquad (26)$$

The dissipative flux vector $\delta_{i+1/2}$ is given by

$$\begin{aligned}\delta_{i+1/2} = {}& V_2\Delta q_{i+1/2} - \\ & V_4(\Delta q_{i+3/2} - 2\Delta q_{i+1/2} + \Delta q_{i-1/2})\end{aligned} \qquad (27)$$

where $\Delta q_{i+1/2} = q_{i+1} - q_i$ and the coefficients V_2 and V_4 are the following:

$$\begin{aligned}V_2 &= \mu_2\max(\nu_{i-1}, \nu_i, \nu_{i+1}, \nu_{i+2})\\ V_4 &= \max(0, \mu_4 - V_2)\end{aligned} \qquad (28)$$

with

$$\nu_i = \frac{|p_{i+1} - 2p_i + p_{i-1}|}{\min(p_n, |p_{i+1} + 2p_i + p_{i-1}|)} \qquad (29)$$

The constant μ_2 scales a first-order artificial viscosity that is switched on at shocks detected by Equation 29. The denominator in Equation 29 is normally constant at the pressure p_n, a fixed low pressure at either the grid inlet or exit, making the operator roughly symmetric across shocks. The more commonly applied term $|p_{i+1} + 2p_i + p_{i-1}|$ is included to switch on the second-difference dissipation when the pressure becomes very small, usually due to numerical problems. The constant μ_4 scales a uniform third-order artificial viscosity that is switched off at shocks by V_2. Typically, values for μ_2 and μ_4 are $1/4$ and $1/32$, respectively.

SLIP and ASLIP Schemes

In the SLIP and ASLIP schemes, the dissipation term $\Delta_\xi \nabla_\xi \hat{q}$ is finite-differenced according to

$$\Delta_\xi \nabla_\xi \hat{q} = (J^{-1}\sigma_\xi)_{i+1}\delta_{i+1/2} - (J^{-1}\sigma_\xi)_{i-1}\delta_{i-1/2} \tag{30}$$

where σ_ξ is given by Equation 13, and the dissipative flux vector $\delta_{i+1/2}$ is given by

$$\delta_{i+1/2} = \frac{1}{2}\alpha\, a_{i+1/2}(q_R - q_L) \tag{31}$$

The subscripts R and L denote right and left states, respectively, for the state vector q. The left and right states are determined by either the SLIP or the ASLIP flux limiter, as described later. The coefficient α is given by

$$\alpha = \varsigma^{+}\begin{cases} |M'_\xi| & |M'_\xi| \geq \varsigma^{-}\varepsilon_0 \\ \dfrac{1}{2}\left(\varsigma^{-}\varepsilon_0 + \dfrac{M'^2_\xi}{\varsigma^{-}\varepsilon_0}\right) & |M'_\xi| < \varsigma^{-}\varepsilon_0 \end{cases} \tag{32}$$

where the relative contravariant Mach number M'_ξ is

$$M'_\xi = \frac{\overline{U'}}{(a\sigma_\xi)_{i+1/2}} = \frac{U'_R + U'_L}{2(a\sigma_\xi)_{i+1/2}} \tag{33}$$

When preconditioning is applied, $q_R - q_L$ is replaced by $\Gamma^{-1}_{i+1/2}(\omega_R - \omega_L)$ in Equation 31.

The parameters ς^{+} and ς^{-} rescale the artificial dissipation near the wall for viscous flows, for reasons discussed in Reference 23. For convenience and computational efficiency, the coefficients obtained from blended eigenvalues (see Equation 26) are used:

$$\begin{aligned} \varsigma^{+} &= \max(1, [1 - 2|M'_\xi|]\varsigma_\xi) \\ \varsigma^{-} &= \min(1, \varsigma_\xi^2) \\ \varsigma_\xi &= \frac{C_\eta + C_\zeta}{C_\xi} \end{aligned} \tag{34}$$

The parameter ς^{+} as given above is used only in the ξ-direction, with ς^{+} being set identically to one for the η- and ζ-directions (where physical viscosity terms exist). The parameters ε_0 is defined by

$$\varepsilon_0 = k_v M'_{ref} \tag{35}$$

where k_v and M'_{ref} are specified constants, with k_v typically given a value of 0.05 or less for viscous simulations, and M'_{ref} being a nominal reference Mach number in the relative frame-of-reference.

As alluded to earlier, either the SLIP or ASLIP flux limiter is used to determine the left and right states. This is done by extrapolation from nearby data, subject to a limiter to preserve monotonicity. This approach is similar to van Leer's MUSCL scheme [24] and provides first-order dissipation near "discontinuities", that is, non-smooth regions, and third-order dissipation in smooth regions of the flow field.

From the work of Swanson, Radespiel, and Turkel [23], a slightly modified form of the SLIP scheme can be written as follows for the right and left states in the dissipative flux vector $\delta_{i+1/2}$:

$$\begin{aligned} q_L &= q_i + \frac{1}{2}L(\Delta q_{i+3/2}, \Delta q_{i+1/2}, \Delta q_{i-1/2}) \\ q_R &= q_{i+1} - \frac{1}{2}L(\Delta q_{i+3/2}, \Delta q_{i+1/2}, \Delta q_{i-1/2}) \end{aligned} \tag{36}$$

where $\Delta q_{i+1/2} = q_{i+1} - q_i$, and the operator L is defined as

$$L\left(\Delta q^{+}, \Delta q, \Delta q^{-}\right) = S\left(\Delta q^{+}, \Delta q^{-}\right)\left[(1-\mu_4)\Delta q + \mu_4\frac{\Delta q^{+} + \Delta q^{-}}{2}\right] \tag{37}$$

with the flux limiter S as

$$S\left(\Delta q^{+}, \Delta q^{-}\right) = 1 - \left| \frac{\Delta q^{+} - \Delta q^{-}}{\left|\Delta q^{+}\right| + \left|\Delta q^{-}\right| + \varepsilon} \right|^{\chi} \tag{38}$$

The exponent χ is a positive number given a value of 2.0 in the RVC3D code. The constant μ_4 scales the third-order dissipation such that in smooth regions of the flow ($S \approx 1$):

$$q_R - q_L \approx -\mu_4 \Delta^3 q_{i+1/2} \tag{39}$$

Typically μ_4 is specified to have a value of 1.0. Note that whenever Δq^{+} and Δq^{-} have the opposite sign then $S \approx 0$, and where the flow is smooth then $S \approx 1$. The parameter ε in Equation 38 is a threshold to prevent the denominator from getting close to zero when both Δq^{+} and Δq^{-} are small.

The ASLIP limiter is similiar to the SLIP limiter except that the operator L is split into two parts:

$$L\left(\Delta q^{+}, \Delta q, \Delta q^{-}\right) = \frac{1}{2} L^{+}\left(\Delta q^{+}, \Delta q\right) + \frac{1}{2} L^{-}\left(\Delta q^{-}, \Delta q\right) \tag{40}$$

where

$$\begin{aligned} L^{+}\left(\Delta q^{+}, \Delta q\right) &= S\left(\Delta q^{+}, \Delta q\right) \\ &\quad \left[(1-\mu_4)\Delta q + \mu_4 \frac{\Delta q^{+} + \Delta q}{2}\right] \\ L^{-}\left(\Delta q^{-}, \Delta q\right) &= S\left(\Delta q^{-}, \Delta q\right) \\ &\quad \left[(1-\mu_4)\Delta q + \mu_4 \frac{\Delta q^{-} + \Delta q}{2}\right] \end{aligned} \tag{41}$$

The flux limiter S is defined the same as above. Note that in smooth regions of the flow the ASLIP scheme is virtually the same as the SLIP scheme, but in non-smooth regions the scheme becomes asymmetric about $i + 1/2$ due to the independent action of the two flux limiters, one for L^{+} and one for L^{-}.

Comparison of Baseline, SLIP, and ASLIP Schemes

A rigorous comparison of the three schemes is well beyond the scope of this paper, but a couple of preliminary comparisons are provided to demonstrate significant differences between the schemes. In particular, results for the nearly-incompressible laminar boundary-layer flow over a flat plate are shown for two conditions, freestream Mach numbers $M_\infty = 0.30$ and $M_\infty = 0.05$, respectively. For both flows the same grid was used and the Reynolds number based on plate length was $Re_L = 1.0 \times 10^6$. Regarding the grid (which is not shown due to space limitations), it should be noted that grid characteristics such as coarseness, wall spacing, and wall stretching, were selected to be representative of typical RVC3D computations, and that these characteristics substantially influence the numerical solutions. Typical values for the numerical parameters were used as well ($\mu_4 = 1/32$ for the baseline scheme, and $\mu_4 = 1$ for the SLIP and ASLIP schemes). Note, however, that instead of RVC3D, a two-dimensional CFD code [25] was used for these simulations, but the numerical formulations are in most respects nearly the same for both codes. Finally, it should be emphasized that the comparisons made here apply only to freestream flow velocity levels under Mach 0.5.

Although the flow over the entire plate length was simulated in each case, it is sufficient to compare the results at only one plate location, $Re_x = 5.0 \times 10^5$. Similar differences were observed at other plate locations.

The results for freestream Mach number $M_\infty = 0.30$ are shown in Figure 1 where the normalized errors $\Delta u / u_\infty$ in the u-velocity profiles are compared for the baseline scheme (with and without preconditioning) and the SLIP and ASLIP schemes (only with preconditioning). A local error of zero corresponds to a local u-velocity which is equal to that of the "exact" Blasius solution. The dimensionless distance from the wall is the standard similarity parameter $\eta = (y/x)\sqrt{Re_x/2}$.

The comparisons in Figure 1 reveal a relatively large difference in accuracy for the baseline and the SLIP/ASLIP schemes, with the latter performing considerably better. It is noteworthy that the baseline scheme is improved substantially by the use of preconditioning, which in this case reduces the artificial dissipation (see Equations 25 and 26) due the to rescaling of eigenvalues (compare Equations 11 and 23). Although outside the scale of the graph, the maximum error of around -7.5 percent for the baseline scheme without preconditioning is reduced down to about -3 percent with preconditioning.

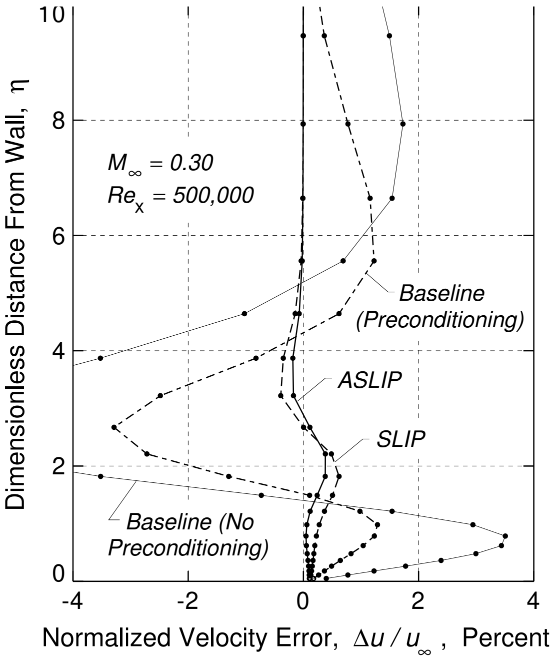


Figure 1 — Error in computed solutions of laminar Blasius velocity profiles using the baseline, SLIP, and ASLIP schemes ($M_\infty = 0.30$)

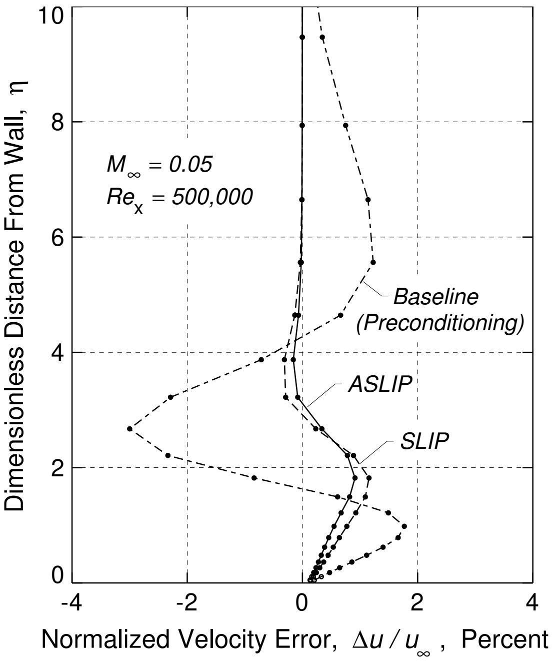


Figure 2 — Error in computed solutions of laminar Blasius velocity profiles using the baseline, SLIP, and ASLIP schemes ($M_\infty = 0.05$)

This result illustrates the fact that at low Mach numbers the preconditioning methods being implemented alter the steady-state solution, as noted earlier, and in a favorable way. The SLIP and ASLIP schemes both perform well and, although not demonstrated, should not depend strongly on whether or not preconditioning is used. The ASLIP scheme is slightly more accurate than the SLIP, as can be seen, and this result has been confirmed in general for a wide range of (undocumented) practical low Mach number simulations for various turbomachinery cases. It should be noted, however, that the ASLIP flux-limiter has been observed to work well only for low Mach number flows, whereas the SLIP flux-limiter has a far wider range of application [20-22]. Finally, it can be seen in Figure 1 that the SLIP and ASLIP schemes completely eliminate the velocity "overshoot" oscillation at the outer edge of the layer, an effect commonly observed with respect to the baseline scheme at virtually all Mach number conditions. For a transonic airfoil example involving such an overshoot and its elimination by using the CUSP scheme with the SLIP flux limiter [20-22], see Reference 26.

Laminar boundary layer results similar to those above can be seen in Figure 2 for the freestream Mach number $M_\infty = 0.05$. In this figure, however, no curve is included for the baseline scheme without preconditioning since the low freestream Mach number makes it impractical. Qualitatively the results are the same as for $M_\infty = 0.30$, although quantitatively the near-wall errors for the SLIP and ASLIP schemes are somewhat larger.

The above comparisons suggest that improved accuracy might be obtained in general by using the SLIP or ASLIP scheme rather than the baseline scheme. However, there are some disadvantages worth mentioning. A primary disadvantage is the direct increased computational overhead. For the 4-stage Runge-Kutta scheme with implicit residual smoothing, the overall CPU-time per iteration increases by about 26 percent for a typical low-speed viscous simulation, while for the 2-stage Runge-Kutta scheme the increase is about 50 percent. Note that, as implemented in RVC3D, each scheme applies the artificial dissipation only once per time-step. Another notable disadvantage is that the SLIP and ASLIP schemes sometimes reduce the code stability with 4-stage Runge-Kutta time-stepping, although the most frequent adverse effect is just poor convergence behavior. The exact cause of this effect has not been detemined, but it is thought that a multigrid scheme (not in RVC3D) might eliminate or reduce it while also improving code stability. Interestingly, good stability and convergence behavior is always achieved when using the SLIP and ASLIP schemes in connection with 2-stage Runge-Kutta time-stepping where, with implicit residual smoothing, CFL numbers of 2.5 to 3.0 can be routinely used.

Numerical Source Terms

Application of the RVC3D code to low Mach number flows revealed that some approximations related to the finite-differencing were inadequate, even though no significant effects were encountered at higher Mach numbers where the code has been routinely applied. In particular, two basic modifications were necessary in order to avoid substantial numerical sources and get good convergence behavior with undistorted flow field solutions. First, the expressions for the finite-difference metrics had to be replaced with more rigorously formulated, but more complicated, ones. Second, a corrective source term had to be added to account for errors introduced by coordinate-system rotation.

Finite-Difference Metrics

Until recently the RVC3D code used second-order central differences to evaluate the metric terms in Equation 7. However, as was pointed out years ago by Pulliam et al. [27], in three dimensions this scheme can introduce a freestream error, although it is noteworthy that Reference 27 mentions no discernible effects from ignoring the numerical source error other than the inability to maintain the free stream. At low enough Mach numbers, however, the resulting numerical sources produce large flow field distortions and can hinder or prevent numerical convergence.

Metric finite-difference formulas based on cell-face projected areas have been in existence for some time, and can be found clearly presented in Reference 28. For convenience, the numerical formulas are also given here:

$$
\begin{aligned}
2J^{-1}\xi_x &= \Delta_{y(0,1,1)}\Delta_{z(0,-1,1)} - \Delta_{y(0,-1,1)}\Delta_{z(0,1,1)} \\
2J^{-1}\xi_y &= \Delta_{z(0,1,1)}\Delta_{x(0,-1,1)} - \Delta_{z(0,-1,1)}\Delta_{x(0,1,1)} \\
2J^{-1}\xi_z &= \Delta_{x(0,1,1)}\Delta_{y(0,-1,1)} - \Delta_{x(0,-1,1)}\Delta_{y(0,1,1)} \\
2J^{-1}\eta_x &= \Delta_{y(1,0,1)}\Delta_{z(1,0,-1)} - \Delta_{y(1,0,-1)}\Delta_{z(1,0,1)} \\
2J^{-1}\eta_y &= \Delta_{z(1,0,1)}\Delta_{x(1,0,-1)} - \Delta_{z(1,0,-1)}\Delta_{x(1,0,1)} \\
2J^{-1}\eta_z &= \Delta_{x(1,0,1)}\Delta_{y(1,0,-1)} - \Delta_{x(1,0,-1)}\Delta_{y(1,0,1)} \\
2J^{-1}\zeta_x &= \Delta_{y(1,1,0)}\Delta_{z(-1,1,0)} - \Delta_{y(-1,1,0)}\Delta_{z(1,1,0)} \\
2J^{-1}\zeta_y &= \Delta_{z(1,1,0)}\Delta_{x(-1,1,0)} - \Delta_{z(-1,1,0)}\Delta_{x(1,1,0)} \\
2J^{-1}\zeta_z &= \Delta_{x(1,1,0)}\Delta_{y(-1,1,0)} - \Delta_{x(-1,1,0)}\Delta_{y(1,1,0)}
\end{aligned}
\tag{42}
$$

where the difference operator is defined as

$$\Delta_{x(I,J,K)} = \frac{1}{2}(x_{i+I,\,j+J,\,k+K} - x_{i-I,\,j-J,\,k-K})$$

Note that the indices *i*, *j*, and *k* correspond to the coordinate directions ξ, η, and ζ, respectively, and that these expressions for the metrics are fully conservative.

The metric Jacobian *J* is calculated as before, using central-finite-differences, since it does not significantly influence the steady-state solution. However, unlike the case of a stationary coordinate system there is some (negligible) influence due to the rotation source term on the right-hand-side of Equation 1.

Coordinate-System Rotation

The numerical source error due to coordinate-system rotation can be examined analytically by considering a special case; namely, that of steady uniform flow in the *x*-direction, in which case the velocity component *u* is a constant, the components *v* and *w* are zero, and all flow derivatives are zero. The *x*-momentum equation (see Equation 1) can then be reduced to the following expression:

$$
\begin{aligned}
0 = \Omega\,[\partial_\xi(-J^{-1}\xi_y z + J^{-1}\xi_z y) &+ \\
\partial_\eta(-J^{-1}\eta_y z + J^{-1}\eta_z y) &+ \\
\partial_\zeta(-J^{-1}\zeta_y z + J^{-1}\zeta_z y)&]
\end{aligned}
\tag{43}
$$

In general, for a non-zero angular velocity Ω, the right-hand side of Equation 43 is mathematically equal to zero, but is not identically zero for the numerical formulation, which produces a freestream error. This error occurs because the coordinates *y* and *z* bear no particular relationship to the finite-difference metrics, even those computed using Equation 42. That is, the location of the center node on a particular "cell-face" is not uniquely determined by the locations of the adjacent nodes from which the corresponding metric is computed.

Using the result of Equation 43, and denoting the right-hand side as S_Ω, a simple and straightforward source term correction can be applied in Equation 1. In particular, an additional source term $\hat{K}$ can be included to exactly cancel the freestream error:

$$\partial_t\hat{q} + \partial_\xi\hat{E} + \partial_\eta\hat{F} + \partial_\zeta\hat{G} - Re^{-1}(\partial_\eta\hat{F}_V + \partial_\zeta\hat{G}_V) = \hat{H} + \hat{K} \tag{44}$$

where

$$\hat{K} = S_\Omega q \tag{45}$$

Note that for efficiency, $\hat{K}$ need only be evaluated at the first stage of the Runge-Kutta scheme and then held constant along with the artificial dissipation and the viscous residual.

Numerical Results

The preconditioned RVC3D code has been applied to a well-documented experimental test case involving a large low-speed centrifugal compressor (LSCC) impeller tested at the NASA Lewis Research Center After describing the experimental test facility along with the CFD grid, several computational details are discussed, followed by comparisons of computational and experimental results.

Low-Speed Centrifugal Compressor Impeller

The LSCC is a large-scale low-speed compressor test rig designed to duplicate the essential flow field and flow physics of a high-speed centrifugal compressor impeller [29-31]. Due to its large size and low speed, extensive aerodynamic instrumentation of the impeller was made feasible, as were detailed flow field measurements using a laser anemometer system [31].

In order to provide an illustration of the test impeller, a coarse rendering of the CFD grid on the impeller hub and blade surfaces is shown in Figure 3 . The impeller has 20 full blades with 55 degrees backsweep and a design tip speed of 153 m/s (1917 rpm). The inlet diameter is 0.870 m with an inlet blade height of 0.218 m, and the exit diameter is 1.524 m with an exit blade height of 0.141 m. The tip clearance between shroud and impeller is uniform at 2.54 mm, which is 1.8 percent of blade height at the impeller exit. Fillet radii at the impeller hub are uniform at 9.525 mm. A vaneless radial diffuser is located downstream of the impeller.

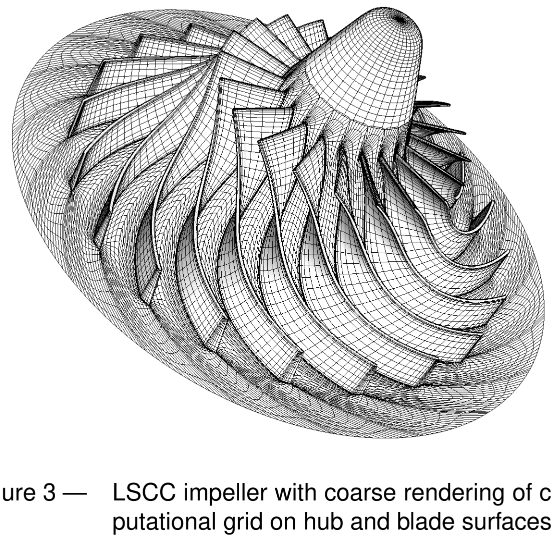
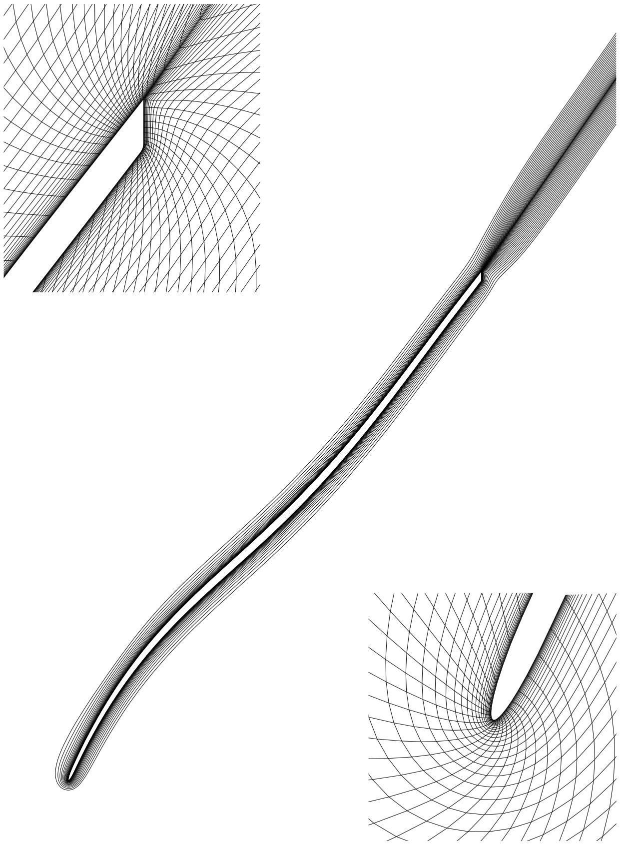


Figure 3 — LSCC impeller with coarse rendering of computational grid on hub and blade surfaces

Although Figure 3 shows all 20 impeller blades, the computational grid for RVC3D has only one blade (and blade channel), with blade-to-blade flow periodicity enforced by the boundary conditions. The RVC3D code uses C-type grids [11,12], and the same grid was used for all computational results presented below. The blade-to-blade grid near midspan is shown in Figure 4 which includes expanded views for the blade leading- and trailing-edges. The dimension of the grid in the ξ-direction (wrapping around the blade) is 225, while in the η-direction (normal to the blade surface) the grid has a dimension of 33, giving a total of 65 nodes blade-to-blade. In the ζ-direction (spanwise, hub-to-shroud), the grid dimension is 65 with more clustering near the shroud for better resolution of the blade tip clearance flow. As shown in Figure 5, the grid was stretched into the clearance gap region over the blade tip so as to provide for a direct simulation of the tip clearance flow, but without having to use a multi-block-grid flow solver. At the impeller hub, the blade-hub fillets



Adobe’s PostScript Language Reference Manual, 2nd Edition, section H.2.4
says your EPS file is not valid, as it calls clear

cleardictstack