Theoret. Appl. Mech., Vol. 30, No. 1, pp. 55-82, Belgrade 2003

Numerical simulation of complex 3D compressible viscous flows through rotating blade passages

M. Despotović * **M. Babić** †
D. Milovanović ‡ **V. Šušterčić** §

Abstract

This paper describes a three-dimensional compressible Navier-Stokes code, which has been developed for analysis of turbocompressor blade rows and other internal flows. Despite numerous numerical techniques and statement that Computational Fluid Dynamics has reached state of the art, issues related to successful simulations represent valuable database of how particular technique behave for a specific problem. This paper deals with rapid numerical method accurate enough to be used as a design tool. The mathematical model is based on system of Favre averaged Navier-Stokes equations that are written in relative frame of reference, which rotates with constant angular velocity around axis of rotation. The governing equations are solved using finite volume method applied on structured grids. The numerical procedure is based on the explicit multistage Runge-Kutta scheme that is coupled with modern numerical procedures for convergence acceleration. To demonstrate the accuracy of the described numerical method developed software is applied to numerical analysis of flow through impeller of axial turbocompressor, and obtained results are compared with available experimental data.

*Faculty of Mech. Eng., Kragujevac, Yugoslavia (m.despot@ptt.yu)
†Faculty of Mech. Eng., Kragujevac, Yugoslavia (nastasija@ptt.yu)
‡Faculty of Mech. Eng., Kragujevac, Yugoslavia (dobrica@knez.uis.kg.ac.yu)
§Faculty of Mech. Eng., Kragujevac, Yugoslavia (vanjas@knez.uis.kg.ac.yu)

1 Introduction

Effective progress in turbocompressors design requires understanding of flows through their passages, which are very complex. The main challenges that stand before the turbocompressor designers or analysts are: to increase loading - that is to increase the pressure ratio for a given amount of hardware, to increase efficiency - that is to reduce the power required to achieve the pressure rise, and to increase the regions of high efficiency. Improvements in each of the above are extremely demanding. For successfully solving all these tasks understanding of flow conditions within turbocompressors plays very important role. In recent years, thanks to advantages in relation to experimental research, the computational fluid dynamics (CFD), has been used much more widely, resulting in shorter development cycle. In many instances, CFD simulations provide the only detailed flow field information, as experimental testing of turbocompressors with detailed measurement in rotating passages is very expensive, and in many cases, impossible.

Turbulent flow through impeller of turbocompressor is probably one of the most difficult problem in CFD. Numerical simulations of high Re number turbulent flows, occurring in complex geometry from inlet to outlet of turbocompressor impeller, require a very large number of grid points and extremely small grid spacing to resolve phenomena in near-wall region.

For engineering applications, at this stage, numerical solving of the averaged Navier-Stokes equations [13] represents an optimal choice for simulation of turbulent flows through complex domains. In order to improve performance of flow computations, it is necessary to apply effective acceleration techniques. The multistage Runge-Kutta scheme, developed by Jameson [8], in conjunction with local time stepping, implicit residual smoothing and the multigrid method is a powerful numerical tool and with much success has been applied in CFD applications for solving turbomachinery flow problems [1], [5].

Objective of this paper is to present a numerical method for simulation of three-dimensional compressible viscous flows through rotating impellers. The application of the three-dimensional compressible turbulent flow computation through isolated rotor of axial turbocompressor is described and compared with experimental data [2].

2 Introductory studies, starting equations

The three-dimensional, unsteady Navier-Stokes equations, which describe the compressible, viscous flow, can be written in Cartesian rotating frame of reference as:

$$\frac{\partial Q}{\partial t} + \frac{\partial (F - F_\nu)}{\partial x} + \frac{\partial (G - G_\nu)}{\partial y} + \frac{\partial (H - H_\nu)}{\partial z} = S, \tag{1}$$

where vector S on the right hand side represents the source term, which takes into consideration influence of Coriolis and centrifugal forces on fluid flow. If impeller rotates with constant angular velocity Ω around x axis, then the source term is given as:

$$S = \begin{bmatrix} 0 \\ 0 \\ \rho\Omega(\Omega y + 2w) \\ \rho\Omega(\Omega z - 2v) \\ 0 \end{bmatrix}. \tag{2}$$

The Cartesian frame of reference is not the most suitable for simulation in complex geometrical domains. Therefore it is desirable to transform system of equations (1) into boundary fitted coordinate system (Fig. 1).

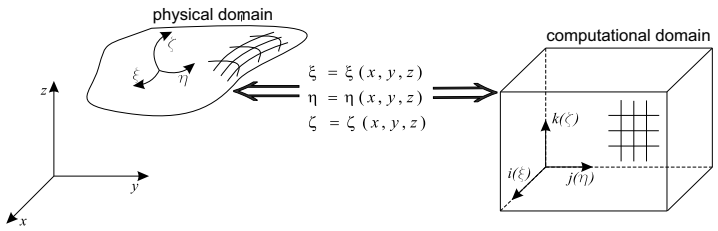

Figure 1: Coordinate transformation

After general three-dimensional transformation between the Cartesian variables (x, y, z) and the generalized coordinates (ξ, η, ζ), which is defined by Jacobian of transformation:

$$\mathbb{J} = \left[\frac{\partial(\xi, \eta, \zeta)}{\partial(x, y, z)}\right] = \begin{bmatrix} \xi_x & \xi_y & \xi_z \\ \eta_x & \eta_y & \eta_z \\ \zeta_x & \zeta_y & \zeta_z \end{bmatrix}, \quad J = \det \mathbb{J}, \tag{3}$$

equation (1) can be written as:

$$\frac{\partial \hat{Q}}{\partial t} + \frac{\partial (\hat{F} - \hat{F}_\nu)}{\partial \xi} + \frac{\partial (\hat{G} - \hat{G}_\nu)}{\partial \eta} + \frac{\partial (\hat{H} - \hat{H}_\nu)}{\partial \zeta} = \hat{S}, \tag{4}$$

where,

$$\hat{Q} = \frac{1}{J} \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho e \end{bmatrix}, \quad \hat{S} = \frac{1}{J} \begin{bmatrix} 0 \\ 0 \\ \rho\omega(\omega y + 2w) \\ \rho\omega(\omega z - 2v) \\ 0 \end{bmatrix}. \tag{5}$$

The inviscid flux terms $\hat{F}$, $\hat{G}$ and $\hat{H}$ have the form of the following expression:

$$\hat{F} = \frac{1}{J} \begin{bmatrix} \rho U \\ \rho u U + \xi_x p \\ \rho v U + \xi_y p \\ \rho w U + \xi_z p \\ (\rho e + p) U \end{bmatrix}, \hat{G} = \frac{1}{J} \begin{bmatrix} \rho V \\ \rho u V + \eta_x p \\ \rho v V + \eta_y p \\ \rho w V + \eta_z p \\ (\rho e + p) V \end{bmatrix}, \hat{H} = \frac{1}{J} \begin{bmatrix} \rho W \\ \rho u W + \zeta_x p \\ \rho v W + \zeta_y p \\ \rho w W + \zeta_z p \\ (\rho e + p) W \end{bmatrix}, \tag{6}$$

where U, V, W are the contravariant velocities, given by

$$\begin{aligned} U &= \xi_x u + \xi_y v + \xi_z w, \\ V &= \eta_x u + \eta_y v + \eta_z w, \\ W &= \zeta_x u + \zeta_y v + \zeta_z w. \end{aligned} \tag{7}$$

The viscous flux terms $\hat{F}_\nu$, $\hat{G}_\nu$ and $\hat{H}_\nu$ have the following form:

$$\hat{F}_\nu = \frac{1}{J}\begin{bmatrix} 0 \\ \xi_x\tau_{xx} + \xi_y\tau_{xy} + \xi_z\tau_{xz} \\ \xi_x\tau_{xy} + \xi_y\tau_{yy} + \xi_z\tau_{yz} \\ \xi_x\tau_{xz} + \xi_y\tau_{yz} + \xi_z\tau_{zz} \\ \xi_x b_x + \xi_y b_y + \xi_z b_z \end{bmatrix}, \hat{G}_\nu = \frac{1}{J}\begin{bmatrix} 0 \\ \eta_x\tau_{xx} + \eta_y\tau_{xy} + \eta_z\tau_{xz} \\ \eta_x\tau_{xy} + \eta_y\tau_{yy} + \eta_z\tau_{yz} \\ \eta_x\tau_{xz} + \eta_y\tau_{yz} + \eta_z\tau_{zz} \\ \eta_x b_x + \eta_y b_y + \eta_z b_z \end{bmatrix},$$

$$\hat{H}_\nu = \frac{1}{J}\begin{bmatrix} 0 \\ \zeta_x\tau_{xx} + \zeta_y\tau_{xy} + \zeta_z\tau_{xz} \\ \zeta_x\tau_{xy} + \zeta_y\tau_{yy} + \zeta_z\tau_{yz} \\ \zeta_x\tau_{xz} + \zeta_y\tau_{yz} + \zeta_z\tau_{zz} \\ \zeta_x b_x + \zeta_y b_y + \zeta_z b_z \end{bmatrix},$$

where the shear stress and heat flux terms are defined in tensor notation as:

$$\begin{aligned} \tau_{x_i x_j} &= \frac{1}{\mathrm{Re}_\infty}\left[\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} + \lambda\frac{\partial u_k}{\partial x_k}\delta_{ij}\right)\right], \\ b_{x_i} &= u_j\tau_{x_i x_j} - \dot{q}_{x_i}, \\ \dot{q}_{x_i} &= -\frac{\mu}{\mathrm{Re}_\infty \mathrm{Pr}(\gamma-1)M_\infty^2}T_{x_i}. \end{aligned} \tag{8}$$

3 The closure problem

For closing system of averaged Navier-Stokes equations (4) the two equation $k-\omega$ turbulence model of Wilcox [13] is used, which is given in tensor notation as:

$$\begin{aligned} &\frac{\partial}{\partial t}(\rho k) + \frac{\partial}{\partial x_j}\left[\rho u_j k - (\mu + \sigma^*\mu_t)\frac{\partial k}{\partial x_j}\right] = P - c_k f_k \rho\omega k \ , \\ &\frac{\partial}{\partial t}(\rho\omega) + \frac{\partial}{\partial x_j}\left[\rho u_j \omega - (\mu + \sigma\mu_t)\frac{\partial \omega}{\partial x_j}\right] = c_{\omega 1} f_\omega \frac{\omega}{k}P - c_{\omega 2}\rho\omega^2 \ . \end{aligned} \tag{9}$$

Coefficients and dumping functions, which close the system of turbulence model equations are given as follows: $c_k = 9/100$, $c_\mu = 1$,

$c_{\omega 1} = 5/9$, $c_{\omega 2} = 3/40$, $\sigma = \sigma^* = 0.5$, $f_\mu = \frac{\frac{c_{\omega 2}}{3} + R_t/6}{1 + R_t/6}$, $f_k = \frac{\frac{5}{18} + (R_t/8)^4}{1 + (R_t/8)^4}$, and $f_\omega = \frac{0.1 + R_t/2.7}{(1 + R_t/2.7) f_\mu}$, where R_t is turbulent Reynolds number.

The molecular viscosity μ, which is computed according to Sutherland law, and molecular thermal conductivity k are replaced with:

$$\mu = \mu_l + \mu_t \quad , \tag{10}$$

$$k = c_p \left[\left(\frac{\mu}{\mathrm{Pr}} \right)_l + \left(\frac{\mu}{\mathrm{Pr}} \right)_t \right] \quad , \tag{11}$$

where c_p is the specific heat at constant pressure, Pr is the Prandtl number, and the subscripts l and t refer to laminar and turbulent, respectively.

The turbulence energy production P on the right hand side of closing equations is given in tensor notation as:

$$P = \left[\frac{1}{\mathrm{Re}_\infty} \mu_t \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \delta_{ij} \frac{\partial u_k}{\partial x_k} \right) - \frac{2}{3} \delta_{ij} \rho k \right] \frac{\partial u_i}{\partial x_j}. \tag{12}$$

4 Numerical algorithm

4.1 Spatial discretization

The governing equations are spatially discretized using vertex-centered finite volume method on H type structured grids. By means of Gauss' divergence theorem the integral form of eq. (4) can be easily rewritten following the standard finite -volume formulation:

$$V_j \frac{\partial}{\partial t} \bar{Q} + \oint_{S_j} \hat{\mathcal{F}}(\hat{Q}) \cdot n dS_j = V_j \bar{S}, \tag{13}$$

where $\bar{Q}$ and $\bar{S}$ denote averaged values of conserved variables and source terms over control volume V_j:

$$\bar{Q} = \frac{1}{V_j} \int_{V_j} \hat{Q} dV_j, \quad \bar{S} = \frac{1}{V_j} \int_{V_j} \hat{S} dV_j, \tag{14}$$

and n is the vector normal to surface S_j, which envelops control volume.

$\hat{\mathcal{F}}(\hat{Q})$ in eq. (13) represents set of vectors and tensors that define convective and diffusive terms of governing equations and is given as:

$$\hat{\mathcal{F}}(\hat{Q}) = \hat{F}(\hat{Q}) - \hat{F}_\nu(\hat{Q}) + \hat{G}(\hat{Q}) - \hat{G}_\nu(\hat{Q}) + \hat{H}(\hat{Q}) - \hat{H}_\nu(\hat{Q}). \qquad (15)$$

Since in the present paper the vertex-centered finite volume method is concerned, the control volume in computational domain is shown in Fig. 2.

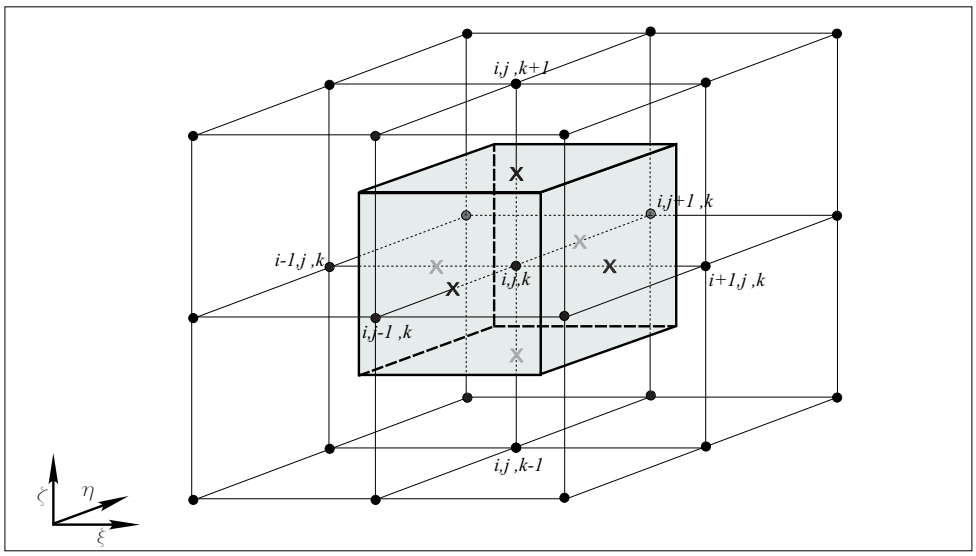

Figure 2: Control volume in computational domain

Introducing indexes of nodes of computational grid the unknown variables can be expressed as:

$$\hat{Q}_{i,j,k} = \hat{Q}(\xi, \eta, \zeta) = \hat{Q}(i\Delta\xi, j\Delta\eta, k\Delta\zeta). \qquad (16)$$

Setting $\Delta\xi = \Delta\eta = \Delta\zeta = 1$, we get the vector of conserved variables as a function of indexes of computational grid:

$$\hat{Q}_{i,j,k} = \hat{Q}(i, j, k). \qquad (17)$$

With adopted notation the convective $\hat{\mathcal{F}}^c$ and diffusive $\hat{\mathcal{F}}^d$ part of eq. (15) get the form of following expressions:

$$\begin{aligned}\hat{\mathcal{F}}^c_{i,j,k}(\hat{Q}) &= (\delta_\xi \hat{F})_{i,j,k} + (\delta_\eta \hat{G})_{i,j,k} + (\delta_\zeta \hat{H})_{i,j,k} \\ &= \hat{F}_{i+\frac{1}{2},j,k} - \hat{F}_{i-\frac{1}{2},j,k} + \hat{G}_{i,j+\frac{1}{2},k} - \hat{G}_{i,j-\frac{1}{2},k} + \hat{H}_{i,j,k+\frac{1}{2}} - \hat{H}_{i,j,k-\frac{1}{2}},\end{aligned} \tag{18}$$

$$\begin{aligned}\hat{\mathcal{F}}^d_{i,j,k}(\hat{Q}) &= [\delta_\xi(\hat{F}_\nu)]_{i,j,k} + [\delta_\eta(\hat{G}_\nu)]_{i,j,k} + [\delta_\zeta(\hat{H}_\nu)]_{i,j,k} \\ &= (\hat{F}_\nu)_{i+\frac{1}{2},j,k} - (\hat{F}_\nu)_{i-\frac{1}{2},j,k} + (\hat{G}_\nu)_{i,j+\frac{1}{2},k} - \\ &\quad (\hat{G}_\nu)_{i,j-\frac{1}{2},k} + (\hat{H}_\nu)_{i,j,k+\frac{1}{2}} - (\hat{H}_\nu)_{i,j,k-\frac{1}{2}}.\end{aligned} \tag{19}$$

The viscous fluxes at the cell faces are approximated with central differencing, and convective fluxes are discretized using flux difference splitting scheme of Roe [10]. Thus the flux difference in ξ direction could be written as:

$$\begin{aligned}(\delta_\xi \hat{F})_{i,j,k} &= \frac{1}{2}\left[\hat{F}(Q_L) + \hat{F}(Q_R) - |\tilde{A}|(Q_R - Q_L)\right]_{i+\frac{1}{2},j,k} \\ &- \frac{1}{2}\left[\hat{F}(Q_L) + \hat{F}(Q_R) - |\tilde{A}|(Q_R - Q_L)\right]_{i-\frac{1}{2},j,k},\end{aligned} \tag{20}$$

where $\hat{A}$ represents flux Jacobian matrix, which is known as Roe's matrix, and Q_L and Q_R are interpolated values of variables from left and right sides of control volumes, which are defined by monotone MUSCL scheme.

After calculating, the corresponding dissipation term of Roe's scheme can be written in following form, which is suitable for programming [5]:

$$|\tilde{A}|(Q_R - Q_L) \equiv |\tilde{A}|\Delta Q = \begin{bmatrix} \alpha_4 \\ \tilde{u}\alpha_4 + \hat{\xi_x}\alpha_5 + \alpha_6 \\ \tilde{v}\alpha_4 + \hat{\xi_y}\alpha_5 + \alpha_7 \\ \tilde{w}\alpha_4 + \hat{\xi_z}\alpha_5 + \alpha_8 \\ \tilde{H}\alpha_4 + \tilde{u}\alpha_6 + \tilde{v}\alpha_7 + \tilde{w}\alpha_8 - \dfrac{\tilde{a}^2\alpha_1}{\gamma - 1} \end{bmatrix}, \tag{21}$$

where:

$$\begin{aligned}
\alpha_1 &= \left|\frac{\nabla\xi}{J}\right| \left|\tilde{\bar{U}}\right| \left(\Delta\rho - \frac{\Delta p}{\tilde{a}^2}\right), \\
\alpha_2 &= \frac{1}{2\tilde{a}^2} \left|\frac{\nabla\xi}{J}\right| \left|\tilde{\bar{U}} + \tilde{a}\right| (\Delta p + \tilde{\rho}\tilde{a}\Delta\bar{U}), \\
\alpha_3 &= \frac{1}{2\tilde{a}^2} \left|\frac{\nabla\xi}{J}\right| \left|\tilde{\bar{U}} - \tilde{a}\right| (\Delta p - \tilde{\rho}\tilde{a}\Delta\bar{U}), \\
\alpha_4 &= \alpha_1 + \alpha_2 + \alpha_3 \\
\alpha_5 &= \tilde{a}(\alpha_2 - \alpha_3) \\
\alpha_6 &= \left|\frac{\nabla\xi}{J}\right| \left|\tilde{\bar{U}}\right| (\tilde{\rho}\Delta u - \hat{\xi}_x\tilde{\rho}\Delta\bar{U}), \\
\alpha_7 &= \left|\frac{\nabla\xi}{J}\right| \left|\tilde{\bar{U}}\right| (\tilde{\rho}\Delta v - \hat{\xi}_y\tilde{\rho}\Delta\bar{U}), \\
\alpha_8 &= \left|\frac{\nabla\xi}{J}\right| \left|\tilde{\bar{U}}\right| (\tilde{\rho}\Delta w - \hat{\xi}_z\tilde{\rho}\Delta\bar{U}).
\end{aligned} \tag{22}$$

Variables $\hat{\xi_x}$, $\hat{\xi_y}$, $\hat{\xi_z}$, and $\tilde{\bar{U}}$ are defined with following expressions:

$$\hat{\xi_x} = \frac{\xi_x}{|\nabla\xi|}, \quad \hat{\xi_y} = \frac{\xi_y}{|\nabla\xi|}, \quad \hat{\xi_z} = \frac{\xi_z}{|\nabla\xi|}. \tag{23}$$

$$\tilde{\bar{U}} = \frac{1}{|\nabla\xi|}(\xi_x\tilde{u} + \xi_y\tilde{v} + \xi_z\tilde{w}). \tag{24}$$

4.2 Iterative procedure

The system of governing equations can be rewritten in semi-discrete form by following expression:

$$\frac{\partial\hat{Q}}{\partial t} = RHS, \tag{25}$$

where RHS denotes right hand side, which takes into account convective, diffusive and source terms.

Since control volumes do not vary during the time eq. (25) can be written as:

$$\frac{1}{J}\frac{\partial Q}{\partial t} = RHS. \tag{26}$$

The system of differential equations (26) is advanced in time using an explicit four-stage Runge-Kutta scheme, which has the form of following expression:

$$\begin{aligned}
Q^{(0)} &= Q^{(n)}, \\
Q^{(1)} &= Q^{(0)} + \alpha_1 R\left(Q^{(0)}\right), \\
Q^{(2)} &= Q^{(0)} + \alpha_2 R\left(Q^{(1)}\right), \\
Q^{(3)} &= Q^{(0)} + \alpha_3 R\left(Q^{(2)}\right), \\
Q^{(4)} &= Q^{(0)} + \alpha_4 R\left(Q^{(3)}\right), \\
Q^{(n+1)} &= Q^{(4)},
\end{aligned} \tag{27}$$

where the residual $R(Q)$ according to eq. (26) is:

$$R(Q) = \Delta t \cdot J \cdot RHS(Q). \tag{28}$$

The coefficients α_i in eq. (27) have following values:

$$\alpha_1 = \frac{1}{4}, \quad \alpha_2 = \frac{1}{3}, \quad \alpha_3 = \frac{1}{2}, \quad \alpha_4 = 1. \tag{29}$$

Δt in eq. (28) denotes time step of iterative scheme. Since the property of time-stepping schemes is that solutions do not depend on chosen time step, the last should be as large as possible, but without disturbing stability of numerical method.

4.3 Acceleration techniques

4.3.1 Local time stepping

In order to improve convergence acceleration local time stepping is applied. The local time step limit is obtained as maximum permissible time step at each grid point computed as function of spectral radii of flux Jacobian matrices of convective terms. In the present work, the local time step limit Δt is computed accounting for both the convective (Δt_c) and diffusive (Δt_d) contributions [1]:

$$\Delta t = CFL \left(\frac{\Delta t_c \Delta t_d}{\Delta t_c + \Delta t_d} \right). \tag{30}$$

Specifically, for the inviscid and viscous time steps, the following expressions have been adopted:

$$\Delta t_c = \frac{1}{\lambda_\xi + \lambda_\eta + \lambda_\zeta}, \tag{31}$$

$$\Delta t_d = \frac{1}{K_t \frac{\gamma\mu}{\rho Pr} J^2 \left(S_\eta^2 S_\zeta^2 + S_\xi^2 S_\zeta^2 + S_\xi^2 S_\eta^2 \right)}, \tag{32}$$

where

$$S_\xi^2 = x_\xi^2 + y_\xi^2 + z_\xi^2, \quad S_\eta^2 = x_\eta^2 + y_\eta^2 + z_\eta^2, \quad S_\zeta^2 = x_\zeta^2 + y_\zeta^2 + z_\zeta^2. \tag{33}$$

K_t in expression (32) is a constant whose value is set equal to 2.5, based on numerical experiments [1].

4.3.2 Implicit residual smoothing

Further improvements in robustness and stability limit are obtained by implicit residual smoothing with variable coefficients, which are function of spectral radii of flux Jacobian matrices and CFL numbers of smoothed and unsmoothed schemes. Numerical investigations [5] have shown that this technique with variable coefficients saves CPU time 2-3 times. For the case of three-dimensional flow, the implicit residual smoothing has the form:

$$(1-\beta_\xi\nabla_\xi\Delta_\xi)(1-\beta_\eta\nabla_\eta\Delta_\eta)(1-\beta_\zeta\nabla_\zeta\Delta_\zeta)\tilde{R}_{i,j,k}=R_{i,j,k}\ \ , \tag{34}$$

where $\tilde{R}$ represents residual vector after smoothing in ξ, η and ζ directions with coefficients β_ξ, β_η and β_ζ. For viscous simulation on highly stretched meshes Swanson and Turkel [11] proposed variable coefficients formulations, which have been used in the present paper and which are as follows:

$$\begin{aligned}
\beta_\xi &= MAX\left\{0,\frac{1}{4}\left[\left(\frac{CFL}{CFL^*}\frac{\lambda_\xi}{\lambda_\xi+\lambda_\eta+\lambda_\zeta}\Phi_\xi\right)^2-1\right]\right\},\\
\beta_\eta &= MAX\left\{0,\frac{1}{4}\left[\left(\frac{CFL}{CFL^*}\frac{\lambda_\eta}{\lambda_\xi+\lambda_\eta+\lambda_\zeta}\Phi_\eta\right)^2-1\right]\right\},\\
\beta_\zeta &= MAX\left\{0,\frac{1}{4}\left[\left(\frac{CFL}{CFL^*}\frac{\lambda_\zeta}{\lambda_\xi+\lambda_\eta+\lambda_\zeta}\Phi_\zeta\right)^2-1\right]\right\}.
\end{aligned} \tag{35}$$

CFL and CFL^* in the previous expression are the Courant number of the smoothed and unsmoothed schemes respectively. Coefficients Φ_ξ, Φ_η, and Φ_ζ in expression (35) are defined as follows:

$$\begin{aligned}
\Phi_\xi &= 1+\left(\frac{\lambda_\eta}{\lambda_\xi}\right)^\sigma+\left(\frac{\lambda_\zeta}{\lambda_\xi}\right)^\sigma,\\
\Phi_\eta &= 1+\left(\frac{\lambda_\xi}{\lambda_\eta}\right)^\sigma+\left(\frac{\lambda_\zeta}{\lambda_\eta}\right)^\sigma,\\
\Phi_\zeta &= 1+\left(\frac{\lambda_\xi}{\lambda_\zeta}\right)^\sigma+\left(\frac{\lambda_\zeta}{\lambda_\zeta}\right)^\sigma,
\end{aligned} \tag{36}$$

where λ_ξ, λ_η, and λ_ζ are the spectral radii of the flux Jacobian matrices for the convective terms:

$$\lambda_\xi = |U| + a\sqrt{\xi_x^2 + \xi_y^2 + \xi_z^2},$$

$$\lambda_\eta = |V| + a\sqrt{\eta_x^2 + \eta_y^2 + \eta_z^2}, \tag{37}$$

$$\lambda_\zeta = |W| + a\sqrt{\zeta_x^2 + \zeta_y^2 + \zeta_z^2},$$

and a is the speed of sound.

The equation (34) is solved by parsing into three items according to each coordinate direction, the result of which set of three tridiagonal system of algebraic equations are obtained.

I STEP

$$(1 - \beta_\xi \nabla_\xi \Delta_\xi)\tilde{R}^{**}_{i,j,k} = R_{i,j,k}, \tag{38}$$

$$\tilde{R}^{**}_{i,j,k} - \beta_\xi \left(\tilde{R}^{**}_{i+1,j,k} - 2\tilde{R}^{**}_{i,j,k} + \tilde{R}^{**}_{i-1,j,k}\right) = R_{i,j,k}, \tag{39}$$

$$-\beta_\xi \tilde{R}^{**}_{i-1,j,k} + (1 + 2\beta_\xi)\tilde{R}^{**}_{i,j,k} - \beta_\xi \tilde{R}^{**}_{i+1,j,k} = R_{i,j,k}. \tag{40}$$

do $j = 2, jm-1$
  do $k = 2, km-1$

$i = 2 : -\beta_{\xi_{1,j,k}} \tilde{R}^{**}_{1,j,k} + (1 + 2\beta_{\xi_{2,j,k}})\tilde{R}^{**}_{2,j,k} - \beta_{\xi_{3,j,k}} \tilde{R}^{**}_{3,j,k} = R_{2,j,k}$

$i = 3 : -\beta_{\xi_{2,j,k}} \tilde{R}^{**}_{2,j,k} + (1 + 2\beta_{\xi_{3,j,k}})\tilde{R}^{**}_{3,j,k} - \beta_{\xi_{4,j,k}} \tilde{R}^{**}_{4,j,k} = R_{3,j,k}$

$\vdots$

$i = im-1 : -\beta_{\xi_{im-2,j,k}} \tilde{R}^{**}_{im-2,j,k} + (1 + 2\beta_{\xi_{im-1,j,k}})\tilde{R}^{**}_{im-1,j,k} - \beta_{\xi_{im,j,k}} \tilde{R}^{**}_{im,j,k} = R_{im-1,j,k}$

  enddo
enddo

II STEP

$$(1 - \beta_\eta \nabla_\eta \Delta_\eta)\tilde{R}^{*}_{i,j,k} = \tilde{R}^{**}_{i,j,k}, \tag{41}$$

$$\tilde{R}^{*}_{i,j,k} - \beta_\eta \left(\tilde{R}^{*}_{i,j+1,k} - 2\tilde{R}^{*}_{i,j,k} + \tilde{R}^{*}_{i,j-1,k}\right) = \tilde{R}^{**}_{i,j,k}, \tag{42}$$

$$-\beta_\eta \tilde{R}^*_{i,j-1,k} + (1+2\beta_\eta)\tilde{R}^*_{i,j,k} - \beta_\eta \tilde{R}^*_{i,j+1,k} = \tilde{R}^{**}_{i,j,k}. \tag{43}$$

do $i = 2, im-1$
  do $k = 2, km-1$
$j = 2 : -\beta_{\eta_{i,1,k}} \tilde{R}^*_{i,1,k} + (1+2\beta_{\eta_{i,2,k}})\tilde{R}^*_{i,2,k} - \beta_{\eta_{i,3,k}} \tilde{R}^*_{i,3,k} = \tilde{R}^{**}_{i,2,k}$

$j = 3 : -\beta_{\eta_{i,2,k}} \tilde{R}^*_{i,2,k} + (1+2\beta_{\eta_{i,3,k}})\tilde{R}^*_{i,3,k} - \beta_{\eta_{i,4,k}} \tilde{R}^*_{i,4,k} = \tilde{R}^{**}_{i,3,k}$

$\vdots$

$j = jm-1 : -\beta_{\eta_{i,jm-2,k}} \tilde{R}^*_{i,jm-2,k} + (1+2\beta_{\eta_{i,jm-1,k}})\tilde{R}^*_{i,jm-1,k} - \beta_{\eta_{i,jm,k}} \tilde{R}^*_{i,jm,k} = \tilde{R}^{**}_{i,jm-1,k}$
  enddo
enddo

III STEP

$$(1 - \beta_\zeta \nabla_\zeta \Delta_\zeta)\tilde{R}_{i,j,k} = \tilde{R}^*_{i,j,k}, \tag{44}$$

$$\tilde{R}_{i,j,k} - \beta_\zeta \left(\tilde{R}_{i,j,k+1} - 2\tilde{R}_{i,j,k} + \tilde{R}_{i,j,k-1} \right) = \tilde{R}^*_{i,j,k}, \tag{45}$$

$$-\beta_\zeta \tilde{R}_{i,j,k-1} + (1+2\beta_\zeta)\tilde{R}_{i,j,k} - \beta_\zeta \tilde{R}_{i,j,k+1} = \tilde{R}^*_{i,j,k}. \tag{46}$$

do $i = 2, im-1$
  do $j = 2, jm-1$
$k = 2 : -\beta_{\zeta_{i,j,1}} \tilde{R}_{i,j,1} + (1+2\beta_{\zeta_{i,j,2}})\tilde{R}_{i,j,2} - \beta_{\zeta_{i,j,3}} \tilde{R}_{i,j,3} = \tilde{R}^*_{i,j,2}$

$k = 3 : -\beta_{\zeta_{i,j,2}} \tilde{R}_{i,j,2} + (1+2\beta_{\zeta_{i,j,3}})\tilde{R}_{i,j,3} - \beta_{\zeta_{i,j,4}} \tilde{R}_{i,j,4} = \tilde{R}^*_{i,j,3}$

$\vdots$

$k = km-1 : -\beta_{\zeta_{i,j,km-2}} \tilde{R}_{i,j,km-2} + (1+2\beta_{\zeta_{i,j,km-1}})\tilde{R}_{i,j,km-1} - \beta_{\zeta_{i,j,km}} \tilde{R}_{i,j,km} = \tilde{R}^*_{i,j,km-1}$
  enddo
enddo

For solving previous set of equations very fancy Thomas algorithm is used.

4.3.3 Multigrid

One of the most successful and widely used methods of convergence acceleration is multigrid by means of which convergence can be accelerated several hundred times [6]. The basic idea of multigrid method is to apply sequence of grids to solve a discrete problem. The low-frequency error components on the fine grid are precisely the error components that dramatically slow the convergence. With suitable coarse grid approximations of the fine grid problem, the low-frequency error components on the fine grid appear as high-frequency error components on the coarser grids. Because the coarse grids require less computational work, the objective of multigrid method is to spend much more time on the coarse grids than on the fine grid. The other advantage of working with coarse grids is that coarse grids allow bigger time steps.

In the present paper the full multigrid algorithm is applied. The auxiliary coarser meshes are obtained by eliminating every other mesh line in each coordinate direction. Both the solution vector and the residual vector are transferred to coarse mesh using rule which conserves mass, momentum and energy. Restriction of solution vector has the following form:

$$Q_{2h} = I_h^{2h} Q_h, \tag{47}$$

where I_h^{2h} denotes restriction operator from fine (h) to coarse ($2h$) grid, Q_{2h} is restricted solution vector on coarse grid, and Q_h is current solution of discretized equations, obtained after implementation of one or more Runge-Kutta cycles.

Since the grid points and locations where the variables are stored for the case of vertex-centered finite volume method overlap, restriction of solutions is simple injection of solution values from the fine grid points:

$$Q_{2h(IC,JC,KC)} = Q_{h(i,j,k)}. \tag{48}$$

IC, JC, KC in expression (48) are indexes of grid points on coarse grids defined as:

$$IC = \frac{1}{2}(i+1), \quad JC = \frac{1}{2}(j+1), \quad KC = \frac{1}{2}(k+1). \tag{49}$$

The restriction of residuals is according following expression [5], [9], [11]:

$$I_h^{2h} R_{h(i,j,k)} = 8\mu_\xi^2 \mu_\eta^2 \mu_\zeta^2 R_{h(i,j,k)}, \tag{50}$$

where μ_ξ, μ_η and μ_ζ denote standard averaging operators along ξ, η and ζ direction, which are as follows:

$$\begin{aligned}
\mu_\xi R_{h(i,j,k)} &= \frac{1}{2}(R_{h(i+1/2,j,k)} + R_{h(i-1/2,j,k)}), \\
\mu_\eta R_{h(i,j,k)} &= \frac{1}{2}(R_{h(i,j+1/2,k)} + R_{h(i,j-1/2,k)}), \\
\mu_\zeta R_{h(i,j,k)} &= \frac{1}{2}(R_{h(i,j,k+1/2)} + R_{h(i,j,k-1/2)}).
\end{aligned} \tag{51}$$

For the sake of simplicity the derived form of equation (50) is given with following expression:

$$\begin{aligned}
\mu_\xi^2 \mu_\eta^2 \mu_\zeta^2 R_{h(i,j,k)} = \frac{1}{64} \big(& R_{h(i-1,j+1,k+1)} + 2R_{h(i,j+1,k+1)} + R_{h(i+1,j+1,k+1)} + \\
& 2R_{h(i-1,j,k+1)} + 4R_{h(i,j,k+1)} + 2R_{h(i+1,j,k+1)} + \\
& R_{h(i-1,j-1,k+1)} + 2R_{h(i,j-1,k+1)} + R_{h(i+1,j-1,k+1)} + \\
& 2R_{h(i-1,j+1,k)} + 4R_{h(i,j+1,k)} + 2R_{h(i+1,j+1,k)} + \\
& 4R_{h(i-1,j,k)} + 8R_{h(i,j,k)} + 4R_{h(i+1,j,k)} + \\
& 2R_{h(i-1,j-1,k)} + 4R_{h(i,j-1,k)} + 2R_{h(i+1,j-1,k)} + \\
& R_{h(i-1,j+1,k-1)} + 2R_{h(i,j+1,k-1)} + R_{h(i+1,j+1,k-1)} + \\
& 2R_{h(i-1,j,k-1)} + 4R_{h(i,j,k-1)} + 2R_{h(i+1,j,k-1)} + \\
& R_{h(i-1,j-1,k-1)} + 2R_{h(i,j-1,k-1)} + R_{h(i+1,j-1,k-1)} \big).
\end{aligned}$$

After the residuals are transferred to coarse grid, in order to respect the fine grid approximation, forcing function P_{2h} is defined on the coarse

grid and added to the governing equations. This forcing function is defined as:

$$P_{2h} = I_h^{2h} R_h(Q_h) - R_{2h}(Q_{2h}). \tag{52}$$

Now, the equations on coarse grid have the form:

$$\frac{1}{J_{2h}} \frac{\partial Q_{2h}}{\partial t} = R_{2h} + P_{2h}. \tag{53}$$

When the hybrid fourth order Runge-Kutta scheme is applied on equation (53), the solution on coarse grid can be written as:

$$Q_{2h}^{(p)} = Q_{2h}^{(p-1)} + \alpha_p \Delta t J_{2h} \left(R_{2h}^{(p-1)} + P_{2h} \right), \tag{54}$$

where *(p=1,2,3,4)*.

It is obvious from equation (54) that during the first numerical update on the coarse grid (for $p = 1$):

$$Q_{2h}^{(1)} = Q_{2h}^{(0)} + \alpha_p \Delta t J_{2h} \left(I_h^{2h} R_h(Q_h) \right), \tag{55}$$

the coarse grid residual drops out. This ensures zero corrections from the coarse grid if the restricted residual from the fine grid vanishes.

Execution of one or several time steps on the coarse grid yields corrections of the form:

$$\Delta Q_{2h} = Q_{2h}^k - Q_{2h}^0. \tag{56}$$

These corrections are defined in the grid points of coarse grid, and it is necessary somehow to redistribute them to locations where the variables on the fine grid are stored, in order to correct solutions on the fine grid, which is called prolongation. This is done by three-linear interpolation, so the corrected solutions on fine grid have the form of following expression:

$$Q_h^{cor} = Q_h + I_{2h}^h \Delta Q_{2h}, \tag{57}$$

where I_{2h}^h denotes operator of prolongation corrections from coarse to fine grid.

In order to improve stability of numerical algorithm the implicit residual smoothing procedure is applied on interpolated corrections:

$$(1-\beta_\xi\nabla_\xi\Delta_\xi)(1-\beta_\eta\nabla_\eta\Delta_\eta)(1-\beta_\zeta\nabla_\zeta\Delta_\zeta)\Delta\tilde{Q}_{2h(i,j,k)} = \left(I_{2h}^{h}\Delta Q\right)_{(i,j,k)}, \tag{58}$$

where β_ξ, β_η, and β_ζ constants which have value 0.6-0.8.

Finally, after smoothing corrections, the solution have the form:

$$Q_h^{cor} = Q_h + \Delta\tilde{Q}_{2h}. \tag{59}$$

4.4 Boundary conditions

The very important segment of numerical computation is specifying the boundary conditions because they ensure the unique solution of governing equations. Since blade passages are considered, four types of boundary conditions are used: inlet, outlet, solid walls and periodic boundary conditions.

Specifying the boundary conditions at the inlet is based on theory of characteristics. Number of boundary conditions needed depends on the way information prolong along characteristics, in other words, on eigenvalues of flux Jacobian matrices:

$$\begin{aligned} \lambda_1 &= \lambda_2 = \lambda_3 = U, \\ \lambda_4 &= U + a\sqrt{\xi_x^2+\xi_y^2+\xi_z^2}, \\ \lambda_5 &= U - a\sqrt{\xi_x^2+\xi_y^2+\xi_z^2}. \end{aligned} \tag{60}$$

For subsonic inflow four eigenvalues are positive and one is negative. Accordingly, four boundary conditions should be specified and one value should be extrapolated from interior of domain. At the inlet total pressure, total temperature, turbulent kinetic energy, specific dissipation rate, and two flow angles in perpendicular planes are prescribed, whereas the Riemann invariant R^- is extrapolated from interior of the domain at each iteration. The total velocity at inlet is then found as follows:

$$V_{in} = \frac{(\gamma-1)R^- + \sqrt{2(1-\gamma)(R^-)^2 + 4(\gamma+1)c_pT_0}}{\gamma+1}\ . \tag{61}$$

With known values for flow angles, now is possible to calculate velocity components, and static pressure and density are found from isentropic relations. Knowing the value of velocity at inlet in each iteration and having in mind the prescribed value for total temperature, the temperature at the inlet could be found from following expression:

$$T_0 = T + \frac{V^2}{2 \cdot c_p}, \tag{62}$$

and pressure follows from:

$$p = \frac{p_0}{\left(\frac{T_0}{T}\right)^{\frac{\kappa}{\kappa - 1}}}. \tag{63}$$

Instead of extrapolating Riemann invariant it is possible to extrapolate pressure at the inlet and then by virtue of total pressure and total temperature to obtain velocity at the inlet. For the sake of investigation which one of these two approaches is more eligible both approaches have been used in this paper and no significant difference has been noticed with respect to number of iterations toward steady solutions.

The way of specifying boundary conditions at outlet is very similar, i.e. boundary conditions are specified according to theory of characteristics. A value for pressure at the hub is prescribed and pressure distribution is found from radial equilibrium equation:

$$\frac{\partial p}{\partial r} = \frac{\rho v_\theta^2}{r}. \tag{64}$$

Concerning the other variables they are found by extrapolating mass fluxes from interior with conservation of continuity equation. In the case that the last two grid planes defined by indexes im and $im-1$ are identical resulting that the Jacobians in these cells are equal, extrapolation comes to extrapolation of density and velocity components:

$$
\begin{aligned}
u_{im} &= u_{im-1}, \\
v_{im} &= v_{im-1}, \\
w_{im} &= w_{im-1}, \\
\rho_{im} &= \rho_{im-1}.
\end{aligned}
\tag{65}
$$

Boundary conditions that are used on solid walls are no-slip boundary conditions for velocity components, while pressure, temperature and density are found assuming adiabatic walls: $\frac{\partial T}{\partial n}\big|_{wall} = 0$, $\frac{\partial p}{\partial n}\big|_{wall} = 0$, $\frac{\partial \rho}{\partial n}\big|_{wall} = 0$. In this paper simulation was carried out assuming that there is no tip clearance between shroud and blades. Since the rotating frame of reference is employed, the stationary walls are treated as moving walls with the rotation velocity of the frame in reverse direction. Turbulent kinetic energy is set to zero at solid walls, and the specific dissipation rate following Menter asymptotically approach [7].

Upwind and downwind from the blades periodic boundary conditions are applied for radial and tangential velocity components and all scalar values.

5 Application

The method described above has been implemented in three-dimensional solver and applied to numerical simulation of flow through rotating impeller of axial turbocompressor designated as NASA Stage 37. That compressor was designed in NASA-Lewis Research Center as an inlet stage of an aircraft engine. The whole stage consists of inlet guide vanes, rotor and downstream stator blades.

The numerical predictions as well as experimental measurements were performed only on rotor part, in order to avoid any interactions from the upstream inlet guide vanes or downstream stator blades. The number of rotor blades is 36, and design pressure ratio is 2.106 at mass flow of 20.19 kg/s and rotational speed of 17188.72 min^{-1}. The mesh model of rotor is shown in Fig. 3 (not all grid lines are shown), and computational domain, which has 250000 grid points is shown in Fig.

4. The meridional plane of rotor is shown in Fig. 5, whereby value for axial direction x=0 corresponds to intersection of hub and leading edge.

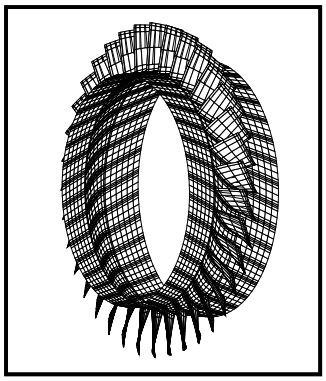

Figure 3: The mesh model of the rotor.

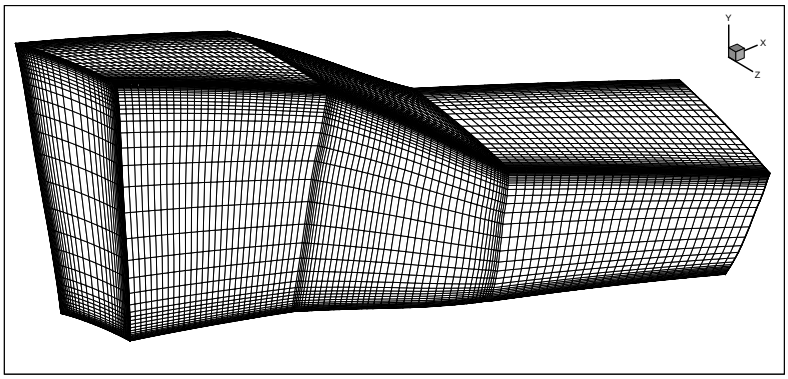

Figure 4: The computational grid.

The measurement data [2] to which computation is compared, were obtained at the NASA Lewis Research Center using aerodynamic probes and laser anemometry. The measurement locations from which the data were picked correspond to axial coordinate of x=-41.9, -5% chord, 20% chord, 45.7, 101.6 and 106.7 in five different spanwise sections (30%, 50%, 70%, 75% and 95%).

After numerical simulation at different boundary conditions, which correspond to mass flow rates that could be seen in Fig. 6 - Fig. 8, the

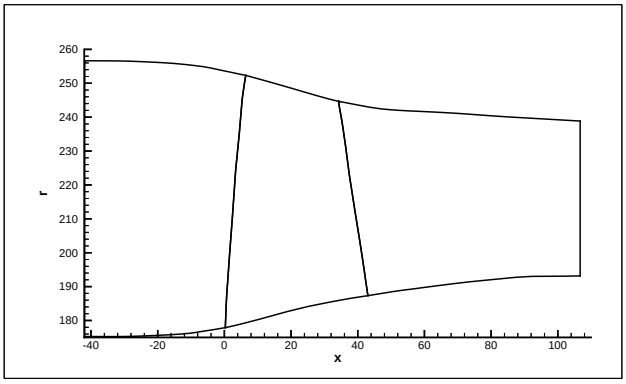

Figure 5: The meridional plane.

performance maps for isolated rotor are obtained. The total pressure ratio π_k is compared to experimental data and shown in Fig. 6. In Fig. 7 the isentropic efficiency is compared to measured data, and predicted polytropic efficiency at different mass flow rates is shown in Fig. 8. The efficiencies are calculated using averaged values from inlet and outlet of computational domain according the following expressions:

$$\eta_{is} = \frac{\pi_k^{\frac{\kappa-1}{\kappa}} - 1}{\pi_k^{\frac{n-1}{n}} - 1} \quad , \tag{66}$$

$$\eta_{pol} = \frac{n}{n-1} \cdot \frac{\kappa - 1}{\kappa} \quad , \tag{67}$$

where

$$n = \frac{\ln \frac{p_{II}}{p_I}}{\ln \frac{p_{II} \cdot T_I}{p_I \cdot T_{II}}} \quad . \tag{68}$$

The contours of relative Mach numbers at 90% span are shown in Fig. 9, and isodensity contours at 90% span are shown in Fig. 10.

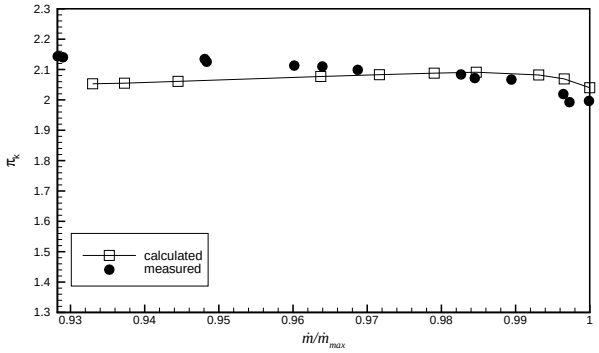

Figure 6: The total pressure ratio.

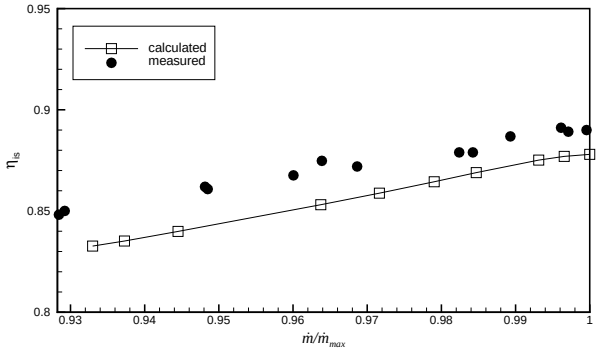

Figure 7: The isentropic efficiency.

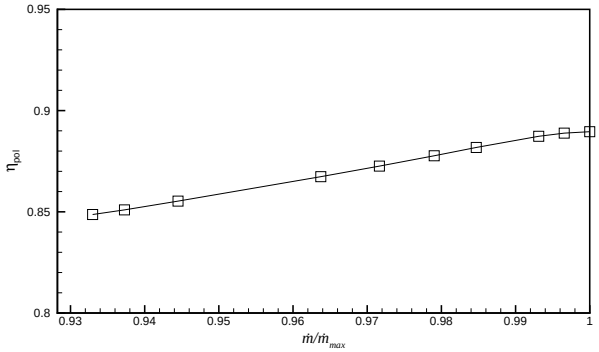

Figure 8: The polytropic efficiency.

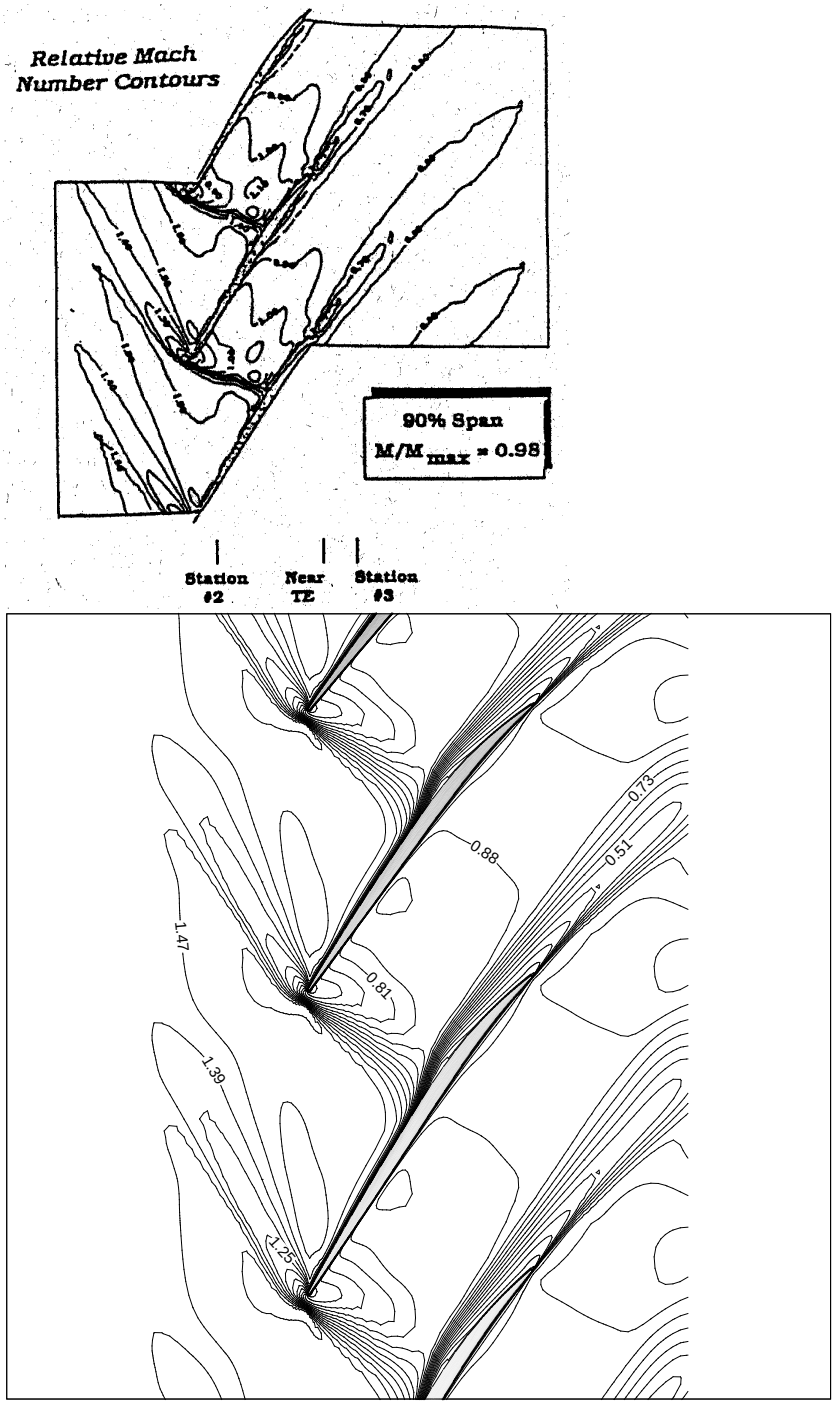

Figure 9: Contours of relative Mach numbers at 90% span

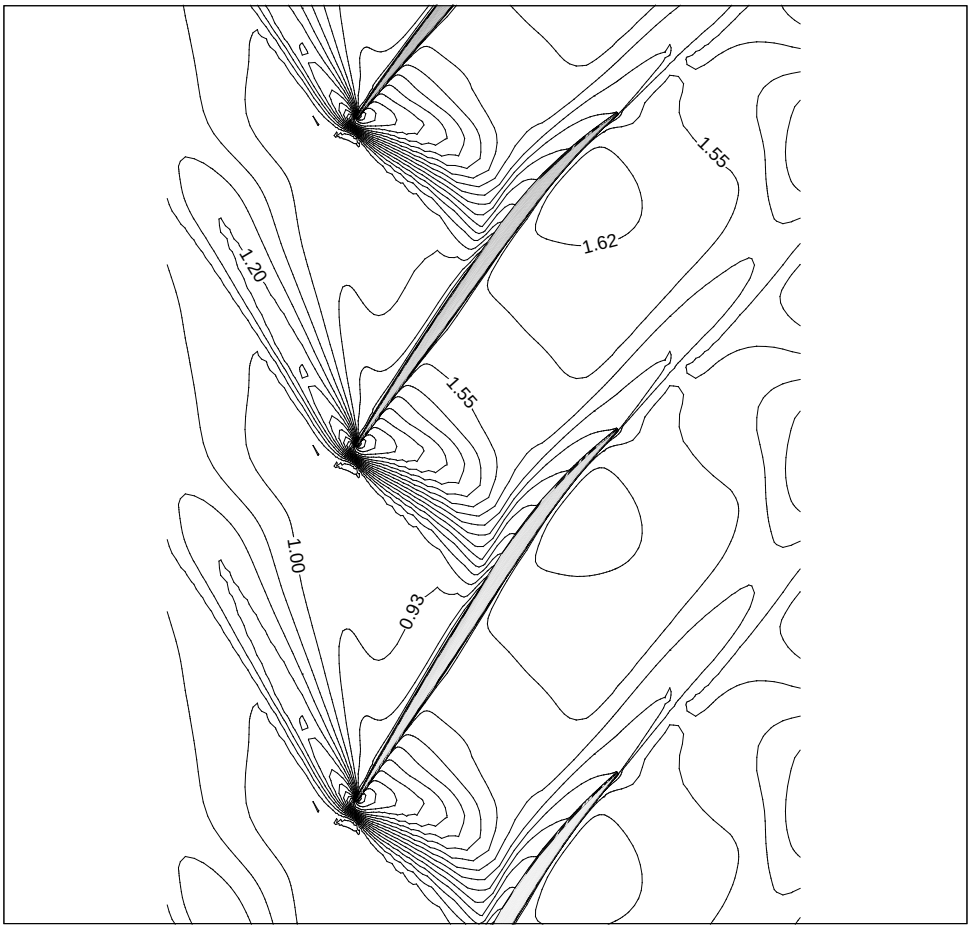

Figure 10: Isodensity contours at 90% span

6 Conclusion

The developed numerical model has been applied to numerical simulation of flow through rotating impeller of axial turbocompressor. The obtained results are compared with experimental data and it can be concluded that global features of flow field are very well resolved. Described procedure has been extended to prediction of performance maps of turbocompressors. For that purpose the calculations were carried out for different mass flow rates and obtained results were compared with experimental data. Both, the performance curve (total pressure ratio versus mass flow), and the isentropic efficiency are very well predicted with respect to experimental data. These results show that the developed methodology enables that numerical simulation on virtual model of compressors could be very useful and powerful tool for flow field prediction, compared to very expensive measurements. In addition, the general conclusion is that the presented numerical method could be very useful and powerful tool for compressor's performance maps prediction that could be used with confidence by designers, in the phase of preliminary design.

References

[1] Arnone A., *Multigrid Methods for Turbomachinery Navier-Stokes Calculations*, Solution Techniques for Large-scale CFD Problems, John Wiley & Sons, 1995.

[2] ASME Turbomachinery Committee *CFD Code Assessment in Turbomachinery - Data Report*, 1994.

[3] Chima V. R., Yokota W. J., *Numerical Analysis of Three-Dimensional Viscous Internal Flows*, AIAA Journal Vol. 28, No. 5, pp. 798-806, 1989.

[4] Chima V. R., *Inviscid and Viscous Flows in Cascades with an Explicit Multiple-Grid Algorithm*, AIAA Journal, Vol. 23, No. 10, pp. 1556- 1563, 1985.

[5] Despotović Z. M., *Investigation of Gasdynamic Characteristics of Compressor Cascades*, Ph. D. Thesis, University of Kragujevac, Yugoslavia, 2002.

[6] Despotović Z. M., *Convergence Acceleration - Multigrid*, Internal Report, Lehrstuhl für Strömungsmechanik, Universität Erlangen-Nürnberg, 2000.

[7] Hellsten A., *On the Solid-Wall Boundary Condition of ω in the $k-\omega$ Type Turbulence Models*, HUT Laboratory of Aerodynamics, Report No. B-50, 1998.

[8] Jameson A., Schmidt W., Turkel E., *Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes*, AIAA-81-1258, 1981.

[9] Kroll N., Radaspiel R., Rossow C., *Structured Grid Solvers I, Accurate and Efficient Flow Solvers for 3D Applications on Structured Meshes*, Special Course on Parallel Computing in CFD, AGARD Report R-807, 1995.

[10] Roe P. L., *Approximate Rieman Solvers, Parameter Vectors, and Diference Schemes*, Journal of Computational Physics, Vol. 43, pp. 357-372, 1981.

[11] Swanson C. R., Turkel E., *Multistage Schemes With Multigrid for Euler and Navier-Stokes Equations*, NASA Technical Paper 3631, 1997.

[12] Toro F. E., *Riemann Solvers and Numerical Methods for Fluid Dynamics*, Springer, 1997.

[13] Wilcox C. David, *Turbulence Modeling for CFD*, DCW Industries, 1998.

Submitted on January 2003; revised on May 2003.

Numerička simulacija složenih trodimenzijskih strujanja viskoznog stišljivog fluida kroz medjulopatične kanale radnog kola

UDK 519.6, 532.517.4, 533.6

U ovom radu se opisuje kompjuterski program koji je razvijen za analizu i simulaciju trodimenzijskog strujanja viskoznog fluida kroz kompresorske profilne rešetke. Uprkos brojnim numeričkim tehnikama i tvrdnji da je Proračunska dinamika fluida dostigla "state of the art" fazu, izveštaji o uspešnim simulacijama predstavljaju dragocenu bazu podataka o svojstvima pojedinih numeričkih tehnika primenjenih na odredjene probleme. U ovom radu se razmatra rapidan numerički algoritam koji je dovoljno brz da bi se mogao koristiti u projektovanju. Matematički model je zasnovan na sistemu Favre-ovo osrednjenih Navije-Stoksovih jednačina, koje su napisane u relativnom koordinatnom sistemu, koji rotira konstantnom ugaonom brzinom oko ose rotacije. Modelske jednačine su rešavane metodom konačnih zapremina na strukturnim mrežama. Numerička procedura je zasnovana na eksplicitnoj višestepenoj Runge-Kutta šemi, spregnutoj sa modernim numeričkim procedurama za ubrzavanje procesa konvergencije. Da bi se pokazala tvcnost opisanog numeričkog metoda, razvijeni softver primenjen je na numeričku analizu strujanja kroz radno kolo aksijalnog turbokompresora, a dobijeni rezultati su uporedjeni sa raspoloživim eksperimentalnim podacima.