

On the implicit density based OpenFOAM solver for turbulent compressible flows

Jiří Fůrst^a

Czech Technical University in Prague, Karlovo nám. 13, 121 35 Praha 2, Czech Republic

Abstract. The contribution deals with the development of coupled implicit density based solver for compressible flows in the framework of open source package OpenFOAM. However the standard distribution of OpenFOAM contains several ready-made segregated solvers for compressible flows, the performance of those solvers is rather weak in the case of transonic flows. Therefore we extend the work of Shen [15] and we develop an implicit semi-coupled solver. The main flow field variables are updated using lower-upper symmetric Gauss-Seidel method (LU-SGS) whereas the turbulence model variables are updated using implicit Euler method.

1 Introduction

The OpenFOAM package [20] is widely used open source toolbox for computational fluid dynamics (CFD). The keystone of the package is a library written in object oriented C++ allowing simple manipulation with fields related to cells, vertices, or nodes of three-dimensional polyhedral meshes. The toolbox contains also several ready-made solvers for incompressible as well as compressible flows based on the finite volume method. The OpenFOAM originated from the so-called pressure based segregated velocity-pressure correction methods devised originally for the incompressible Navier-Stokes equations, see eg. SIMPLE [4] or PISO [7]. However these segregated methods have been later extended to compressible flow problems including transonic or supersonic flows [12], the efficiency of the segregated methods is in the case of high speed flows usually worse when comparing with the so-called density based or coupled methods available in some commercial packages and in some academic in-house codes.

So far several attempts have been done in order to develop density based solvers for OpenFOAM. First of all, there is already a solver based on central-upwind scheme of Kurganov and Tadmor [6] in the standard distribution. Next, the OpenFOAM-extend version of the package contains some coupled solvers based on AUSM+up [9], HLLC [17], Roe or Rusanov fluxes combined with an explicit Runge-Kutta method in time. Similar capabilities has also the AeroFOAM package developed at the Politecnico di Milano. The main deficiency of the above mentioned solvers is the explicit time integration which puts a severe stability bound on the maximum time step and the efficiency of these explicit methods is much lower than the efficiency of implicit methods. This

is especially true for high speed flows at high Reynolds number where the correct resolution of thin boundary layer requires very fine meshes. Let's note also the hybrid pressure-density based method [21], which is based on the AUSM+up fluxes, but still uses a pressure correction loop similar to PISO algorithm and the maximal time step is therefore also limited.

The fully implicit coupled methods (see eg. [5], [3]) provides much better efficiency usually at the price of much higher complexity of the code and higher memory requirements. Besides the fully implicit method the matrix-free LU-SGS (lower-upper symmetric Gauss-Seidel, [3]) method shares the appealing properties of fully implicit methods (i.e. the unconditional stability) with low code complexity and low memory overhead. In their recent paper [15] Shen et al. give detailed description of the implementation of LU-SGS method in the framework of OpenFOAM for the case of inviscid compressible flows. We follow their work and extend the LU-SGS based solver to turbulent flows described by the set of Reynolds-averaged Navier-Stokes equations combined with an additional turbulence model provided by the OpenFOAM package.

2 Mathematical model

2.1 Governing equations

The governing equations expressing the conservation of mass, momentum, and energy are given as follows

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \mathbf{U}) = 0, \quad (1)$$

$$\frac{\partial(\rho \mathbf{U})}{\partial t} + \nabla(\rho \mathbf{U} \otimes \mathbf{U}) + \nabla p = \nabla \boldsymbol{\tau}, \quad (2)$$

$$\frac{\partial(\rho E)}{\partial t} + \nabla[(\rho E + p)\mathbf{U}] = \nabla(\boldsymbol{\tau} \mathbf{U}) + \nabla(\lambda \nabla T), \quad (3)$$

^a Corresponding author: Jiri.Furst@fs.cvut.cz

where ρ is the density, \mathbf{U} is the velocity vector, p is the pressure, $\boldsymbol{\tau}$ is the (effective) stress tensor, E is the specific total energy, λ is the (effective) thermal conductivity, and T is the temperature. The system is closed by the equation of state for ideal gas $p/\rho = rT$ and constant specific heat capacity c_p .

Spatial derivatives at the left hand side of the previous equations represent inviscid terms whereas the right hand side represents viscous terms. The above equations can be expressed in the integral form suitable for finite volume approximation as follows (assuming fixed control volume Ω)

$$\frac{d}{dt} \int_{\Omega} \mathbf{W} d\Omega + \oint_{\partial\Omega} (\mathbf{F}^c - \mathbf{F}^v) dS = 0, \quad (4)$$

where $\mathbf{W} = [\rho, \rho\mathbf{U}, \rho E]$ is the vector of conservative variables and \mathbf{F} and \mathbf{F}^v represent inviscid and viscous fluxes.

In the case of turbulent flow the above mentioned system of equations is coupled to an additional turbulence model (e.g. a two-equation SST model [10]) which provides components of the Reynolds stress tensor and the turbulent heat flux.

2.2 Spatial discretization

The numerical solution is obtained with the finite volume method in semi-coupled fashion. It means that the code updates in each step the main flow variables \mathbf{W} and then it updates the turbulence model variables. Since we are using turbulence models provided by the OpenFOAM package, we do not give detailed description of its discretization and we focus the attention to the discretization of the equations for main flow variables (4).

The integral form of the equations (4) discretized in space using collocated finite volume method (FVM) by taking

$$\mathbf{W}_i(t) = \frac{1}{|\Omega_i|} \iiint_{\Omega_i} \mathbf{W}(\mathbf{x}, t) d\Omega, \quad (5)$$

where Ω_i is the mesh cell i . Hence the equation (4) can be approximated as

$$|\Omega_i| \frac{d\mathbf{W}_i}{dt} = -R(\mathbf{W})_i = - \sum_{j \in N_i} (\mathbf{F}_{ij} + \mathbf{F}_{ij}^v), \quad (6)$$

where N_i is the set of indices of neighbor cells and \mathbf{F}_{ij} and \mathbf{F}_{ij}^v are the numerical fluxes approximating the inviscid and viscous fluxes. Viscous fluxes are discretized directly with OpenFOAM built-in schemes (e.g. central scheme). On the other hand the discretization of inviscid fluxes is made using well established approximate Riemann solvers. The first order Rusanov convective flux is

$$\mathbf{F}_{ij}^{Rus} = \frac{\mathbf{F}(\mathbf{W}_i) + \mathbf{F}(\mathbf{W}_j)}{2} \cdot \mathbf{S}_{ij} - \frac{\lambda_{ij}}{2} (\mathbf{W}_i - \mathbf{W}_j), \quad (7)$$

where the inviscid flux matrix \mathbf{F} is

$$\mathbf{F}(\mathbf{W}) = \begin{bmatrix} \rho\mathbf{U} \\ \rho\mathbf{U} \otimes \mathbf{U} + p\mathbf{I} \\ (\rho E + p)\mathbf{U} \end{bmatrix}, \quad (8)$$

and the stabilization parameter λ_{ij} is

$$\lambda_{ij} = |\mathbf{U}_{ij} \cdot \mathbf{S}_{ij}| + |\mathbf{S}_{ij}| a_{ij}. \quad (9)$$

Here \mathbf{U}_{ij} is the velocity at the face shared by cell i and j (it can be taken as an arithmetic average of \mathbf{U}_i and \mathbf{U}_j) and a_{ij} is the sound speed at the face. The Rusanov flux together with the viscous terms forms a low order residual

$$\mathbf{R}^{lo}(\mathbf{W})_i = \sum_{j \in N_i} (\mathbf{F}^{Rus}(\mathbf{W}_{ij}^L, \mathbf{W}_{ij}^R, \mathbf{S}_{ij}) + \mathbf{F}_{ij}^v). \quad (10)$$

The residual can be further simplified using the fact that $\sum_j \mathbf{S}_{ij} = 0$, so

$$\begin{aligned} \mathbf{R}^{lo}(\mathbf{W})_i &= \frac{1}{2} \sum_{j \in N_i} (\mathbf{F}(\mathbf{W}_j) \cdot \mathbf{S}_{ij} + \lambda_{ij} \mathbf{W}_j) - \\ &\quad - \frac{1}{2} \left(\sum_{j \in N_i} \lambda_{ij} \right) \mathbf{W}_i + \sum_{j \in N_i} \mathbf{F}_{ij}^v. \end{aligned} \quad (11)$$

It is well known that the Rusanov flux is overly diffusive and it is not a good choice for high speed flows. On the other hand thanks to its simplicity one can easily calculate the Jacobian matrix $\partial \mathbf{R} / \partial \mathbf{W}$, see next sections. In order to improve the accuracy a better Riemann solver should be used in the evaluation of the residual. In our the AUSM+up flux [9] or HLLC [17] flux is employed together with a piece-wise linear reconstructions with Barth [2] or Venkatakrisnan [18], thus the high order residual is

$$\mathbf{R}^{ho}(\mathbf{W})_i = \sum_{j \in N_i} (\mathbf{F}^{ho}(\mathbf{W}_{ij}^L, \mathbf{W}_{ij}^R, \mathbf{S}_{ij}) + \mathbf{F}_{ij}^v). \quad (12)$$

Here \mathbf{F}^{ho} is the AUSM+up or HLLC flux evaluated from the reconstructed states at the left (\mathbf{W}^L) and right (\mathbf{W}^R) hand side of the face.

2.3 Temporal discretization

The temporal discretization is achieved by using the implicit Euler method, thus

$$\mathbf{W}_i^{n+1} = \mathbf{W}_i^n - \frac{\Delta t}{|\Omega_i|} \mathbf{R}^{ho}(\mathbf{W}^{n+1})_i. \quad (13)$$

The large system of non-linear equations is then linearized

$$\mathbf{W}_i^{n+1} \approx \mathbf{W}_i^n - \frac{\Delta t}{|\Omega_i|} \left(\mathbf{R}^{ho}(\mathbf{W}^n)_i + \sum_{j \in N_i \cup \{i\}} \frac{\partial \mathbf{R}_i^{ho}}{\partial \mathbf{W}_j} \Delta \mathbf{W}_j \right), \quad (14)$$

with $\Delta \mathbf{W} = \mathbf{W}^{n+1} - \mathbf{W}^n$, thus

$$\sum_{j \in N_i \cup \{i\}} \left[\frac{|\Omega_i|}{\Delta t} \delta_{ij} + \frac{\partial \mathbf{R}_i^{ho}}{\partial \mathbf{W}_j} \right] \Delta \mathbf{W}_j = -\mathbf{R}^{ho}(\mathbf{W}^n)_i. \quad (15)$$

Finally the Jacobian matrix of high order residual at the left hand side is replaced by its low order approximation, so

$$\sum_{j \in N_i \cup \{i\}} \left[\frac{|\Omega_i|}{\Delta t} \delta_{ij} + \frac{\partial \mathbf{R}_i^{lo}}{\partial \mathbf{W}_j} \right] \Delta \mathbf{W}_j = -\mathbf{R}^{ho}(\mathbf{W}^n)_i. \quad (16)$$

The matrix of the linear system (16) can be divided into block diagonal part D , the lower triangular matrix L and the upper triangular matrix. Replacing the Jacobian matrices of viscous fluxes by their spectral radii [3] and assuming local time step one finally arrives to diagonal operator D in the form

$$D_i = \left(\frac{\Omega_i}{\Delta t_i} + \frac{1}{2} \sum_{j \in N_i} \lambda_{ij}^* \right) \mathbf{I}. \quad (17)$$

The spectral radius λ_{ij}^* is composed of the inviscid contribution λ_{ij} given by (9), an over-relaxation parameter ω in the range $1 \leq \omega \leq 2$, and the viscous spectral radius λ_{ij}^v [3]

$$\lambda_{ij}^v = \frac{|\mathbf{S}_{ij}|}{\|\mathbf{x}_i - \mathbf{x}_j\|} \max \left(\frac{4}{3\rho_{ij}}, \frac{\gamma}{\rho_{ij}} \right) \left(\frac{\mu}{Pr} + \frac{\mu_T}{Pr_T} \right), \quad (18)$$

thus

$$\lambda_{ij}^* = \omega \lambda_{ij} + \lambda_{ij}^v. \quad (19)$$

Let us define the sets of cells belonging to lower and upper part of the matrix:

$$L_i = \{j \in N_i : j < i\}, \quad (20)$$

$$U_i = \{j \in N_i : j > i\}. \quad (21)$$

Then the matrix-free LU-SGS scheme can be written using the following two step procedure:

$$D_i \Delta \mathbf{W}_i^{(1)} = -\mathbf{R}_i^{ho} - \frac{1}{2} \sum_{j \in L_i} \left[\Delta \mathbf{F}_j^{(1)} \cdot \mathbf{S}_{ij} + \lambda_{ij}^* \Delta \mathbf{W}_j^{(1)} \right],$$

$$D_i \Delta \mathbf{W}_i = D_i \Delta \mathbf{W}_i^{(1)} - \frac{1}{2} \sum_{j \in U_i} \left[\Delta \mathbf{F}_j \cdot \mathbf{S}_{ij} + \lambda_{ij}^* \Delta \mathbf{W}_j \right].$$

Here

$$\Delta \mathbf{W}_i^{(1)} = \mathbf{W}_i^{(1)} - \mathbf{W}_i^n, \quad (22)$$

$$\Delta \mathbf{F}_i^{(1)} = \mathbf{F}(\mathbf{W}_i^{(1)}) - \mathbf{F}(\mathbf{W}_i^n), \quad (23)$$

$$\Delta \mathbf{F}_i = \mathbf{F}(\mathbf{W}_i^{n+1}) - \mathbf{F}(\mathbf{W}_i^n). \quad (24)$$

2.4 Coupling with a turbulence model

The LU-SGS solver is combined with an additional turbulence model in a weakly-coupled manner. It means that the turbulence quantities (e.g. k and ω in the case of two-equation SST model) are kept fixed during one iteration of LU-SGS model. Then these variables are updated using appropriate model keeping the vector of conservative variable constant. Hence (for the case of SST model:

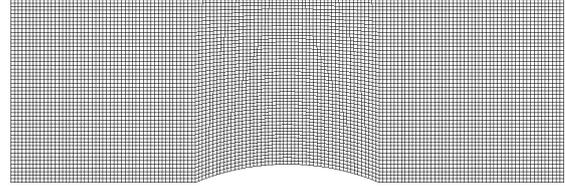


Fig. 1. Structured mesh for the flow over a bump.

1. calculate \mathbf{W}^{n+1} from \mathbf{W}^n using both steps of LU-SGS method (22) and assuming fixed k^n and ω^n ,
2. calculate k^{n+1} and ω^{n+1} using fixed \mathbf{W}^{n+1} .

The advantage of the semi-coupled approach is that it allows inclusion of built-in OpenFOAM turbulence models. Moreover, the implementation is much simpler than it would be in the fully-coupled scheme.

3 Results

3.1 Subsonic flow over a bump

The two-dimensional subsonic flow over a 10% circular bump has been chosen as a first benchmark. The inviscid flow passes through a channel of height $H = 1$ m and length $L = 3$ m. The total pressure at the inlet was $p_{tot,in} = 100$ kPa, the total temperature $T_{tot,in} = 293.15$ K and the inlet flow direction was parallel to x -axis. The average outlet pressure corresponds to isentropic Mach number $M_{is,out} = 0.1$. The solution was obtained using a structured mesh with 150×50 quadrilateral cells, see fig. 1.

The figure 2 shows contours of Mach number obtained with LU-SGS scheme combined with AUSM+up flux and piece-wise linear reconstruction with Venkatakrishnan limiter. The figure 3 shows the comparison of Mach number along the lower wall obtained with the LU-SGS scheme using AUSM+up flux and HLLC flux with the result of segregated solver based on SIMPLE loop. One can see that the HLLC scheme slightly underestimates the maximal Mach number with respect to other mentioned methods. The figure 4 compares the convergence history of LU-SGS scheme with AUSM+up and HLLC fluxes using either Barth or Venkatakrishnan limiter with the convergence history of segregated solver. One can see that the segregated solver is in this case more efficient than the LU-SGS scheme due to low Mach number character of the flow. Note that the efficiency of the LU-SGS scheme can be still improved using a proper preconditioning technique, see e.g. [14].

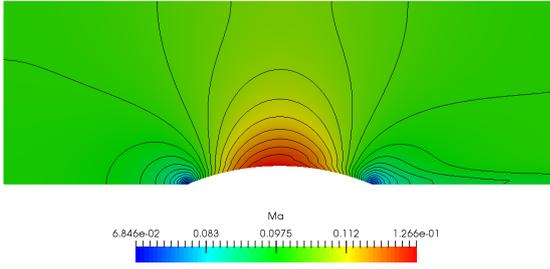


Fig. 2. Mach number distribution for subsonic flow over a bump.

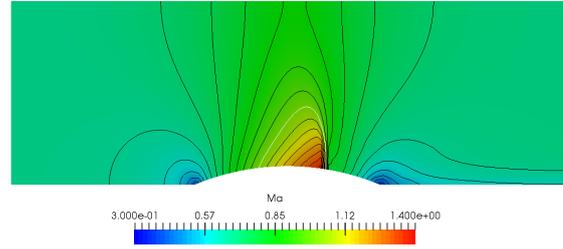


Fig. 5. Mach number distribution for transonic flow over a bump.

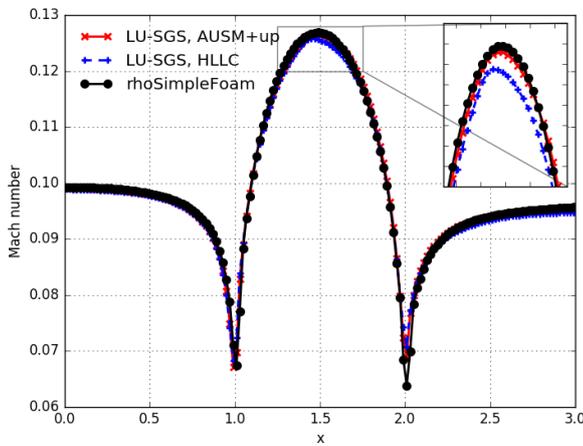


Fig. 3. Mach number distribution over the lower wall for subsonic flows over a bump.

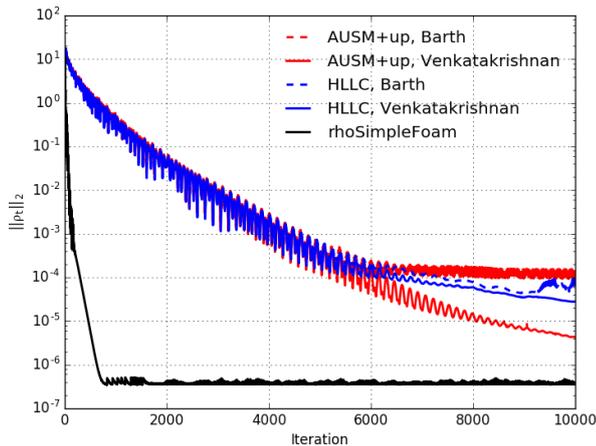


Fig. 4. Convergence history for subsonic flows over a bump.

3.2 Transonic flow over a bump

The second benchmark is a transonic flow over the bump using the same geometry as in the previous case. The same boundary conditions were used at the inlet and the average outlet pressure was set to a value corresponding to isentropic Mach number $M_{is,out} = 0.675$. In this case the flow accelerates over the bump and reaches supersonic speed. Then it passes through a shock wave and continues towards the outlet.

The figure 5 shows the contours of Mach number obtained with LU-SGS scheme with AUSM+up flux and piecewise linear reconstruction with Venkatakrishnan limiter. The sonic line (i.e. the line where the magnitude of the velocity equals to local sound speed) is plotted in other color in order to depict the shape of the supersonic region. The comparison of the Mach number over the lower wall is shown in the figure 6. One can see that all methods including the transonic variant of SIMPLE solver give similar results. Nevertheless the resolution of the shock wave is much better with the AUSM or HLLC based LU-SGS scheme than with the SIMPLE solver which spreads the shock over 4 cell due to higher amount of numerical viscosity. The figure 7 compares the convergence history for all solvers. One can see that the LU-SGS scheme is more efficient than the segregated approach in this case. Note that the CPU time for 10000 iterations was approximately the same using all methods including segregated SIMPLE solver.

3.3 Transonic flow through a turbine cascade

The next benchmark is two-dimensional compressible turbulent flow through test turbine cascade SE-1050. The geometry and boundary conditions correspond to the test from ERCOFTAC QNET database [13]. The inlet total pressure and total temperature were $p_{tot,in} = 100$ kPa and $T_{tot,in} = 293.15$ K and the inlet angle was $\alpha_{in} = 19.3^\circ$. The average outlet pressure corresponds to isentropic outlet Mach number $M_{is,out} = 1.2$, the Reynolds number related to outlet flow conditions and blade pitch was $Re = 1.2 \times 10^6$ and the inlet turbulence intensity was $Tu_{in} = 2\%$.

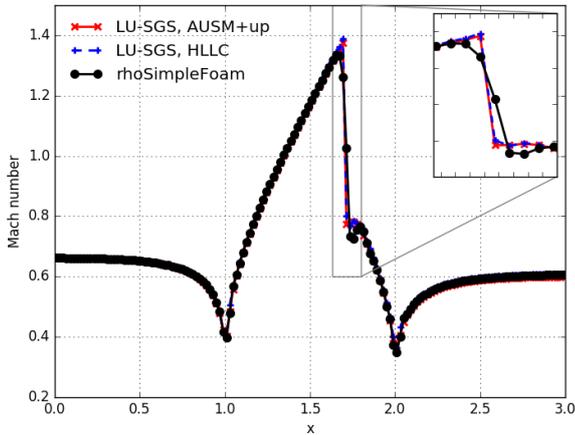


Fig. 6. Mach number distributed over the lower wall for transonic flows over a bump.

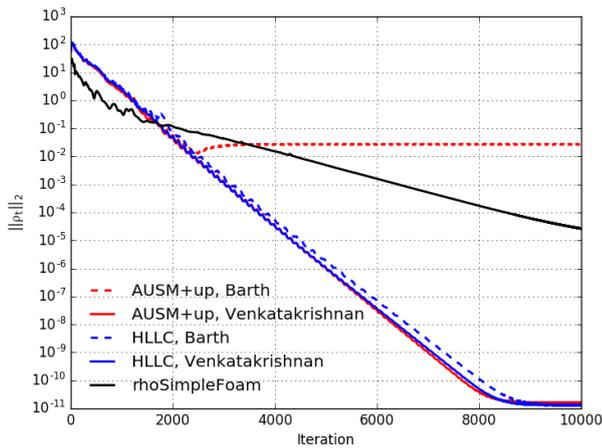


Fig. 7. Convergence history for transonic flows over a bump.

The hybrid mesh was composed of structured quadrilateral layer around the blade and unstructured triangular part in the rest of the domain. The mesh contained approximately 66×10^3 cells and the near-wall refinement corresponded to $y_1^+ \approx 0.2$. Details of the mesh around leading and trailing edges is given at the figure 8. The figure 8 shows contours of the Mach number in two periods of the mesh obtained using LU-SGS scheme with HLLC flux. One can see here both branches of the shock emanating from the trailing edge as well as the interaction of the inner branch of the shock wave with the blade. The figure 10 shows the comparison of the pressure along the blade with the results obtained with segregated SIMPLE solver as well as with the experimental data [13]. One can see that both numerical methods shows very good agreement with experimental data.

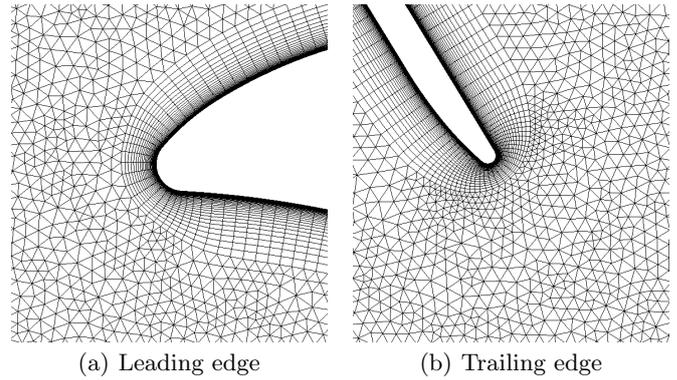


Fig. 8. Hybrid mesh for test turbine cascade.

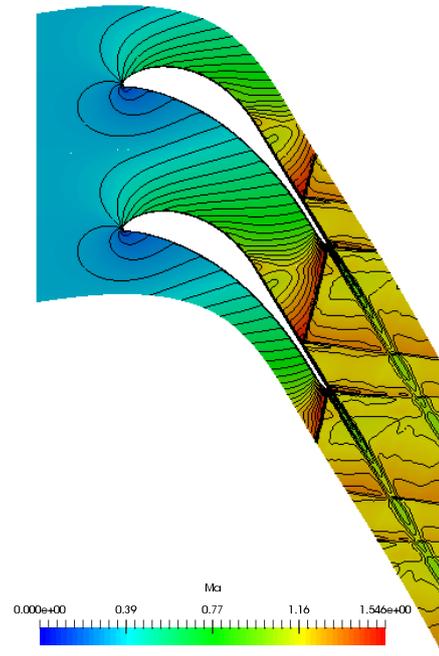


Fig. 9. Calculated Mach number distribution in the test turbine cascade.

3.4 Transonic flow through 3D test turbine cascade

The next case is the 3D flow through a test turbine cascade with prismatic blades. The geometry of blades correspond to the previous case and the blade has finite span, see e.g. [16]. The flow regime corresponds to outlet isentropic Mach number $M_{is,out} = 1.2$ and Reynolds number related to chord length $Re = 1.34 \times 10^6$. Inlet conditions include a simple model of boundary layer corresponding to conditions in the experiment, see [16] for details. The numerical solution was obtained on an unstructured mesh with 3.3×10^6 cells with near wall refinement to $y_1^+ \approx 0.2$. The figure 11 shows the distribution of the pressure at the blades and the side wall and the distribution of entropy in the test plane. The regions with higher level of entropy show re-distribution of the energy losses

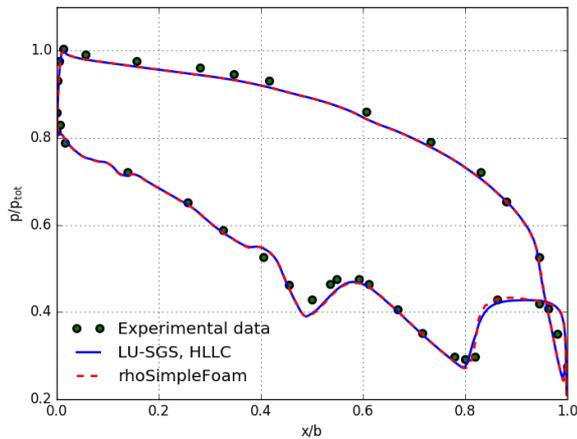


Fig. 10. Pressure distribution over the test turbine blade, comparison with experimental data [13].

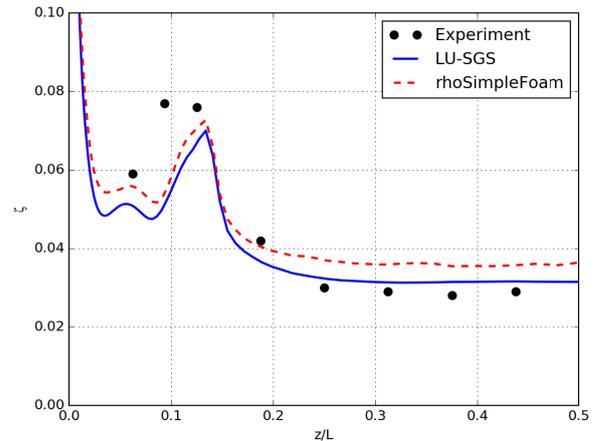


Fig. 12. Spanwise distribution of pitch averaged loss coefficient ζ , comparison with experiment.

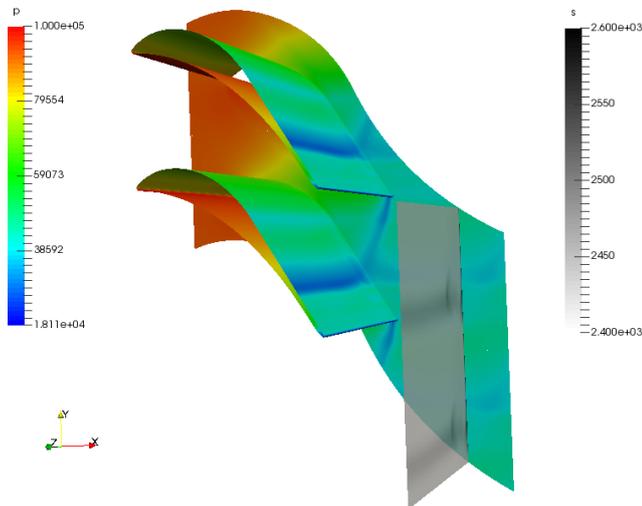


Fig. 11. Pressure distribution at the blades and side wall and the entropy distribution in test plane for the 3D flows over test turbine cascade.

in the vicinity of side walls. The figure 12 shows the comparison of the spanwise distribution loss coefficient ζ (see [16]) obtained using LU-SGS scheme with HLLC flux and the transonic version of segregated SIMPLE solver with experimental data. One can see that the loss coefficient is better predicted by LU-SGS method especially in the middle of the channel.

3.5 Transitional flow through VKI turbine cascade

The last case demonstrates the combination of the LU-SGS scheme for main flow variables with transitional turbulence models. The test case corresponds to the experimental study of Arts et al. [1] performed at the von Karman Institute for Fluid Dynamics. The 2D flow through a turbine cascade was measured in terms of surface heat flux. The numerical solution was obtained with 62 000

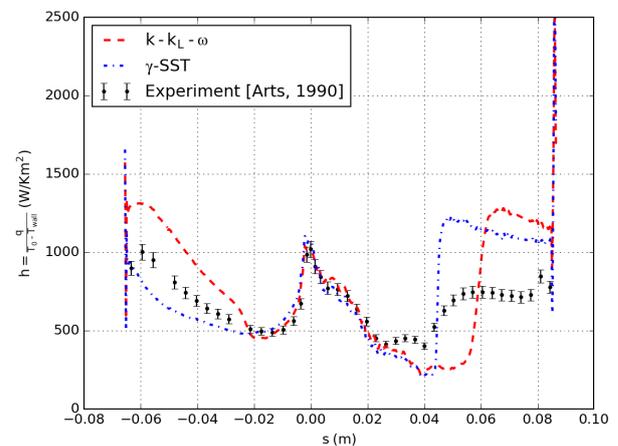


Fig. 13. Surface heat transfer coefficient for VKI MUR 241 Cascade.

cells for the regime corresponding to MUR 241 case from [1], i.e. the outlet Mach number was $M_2 = 1.089$, the Reynolds number related to outlet parameter and chord length was $Re = 2.1 \times 10^6$ and the free-stream turbulence intensity was $Tu = 6\%$. The figure 13 shows the comparison of experimental data with numerical results obtained with a three-equation $k - k_L - \omega$ model developed by Walters and Cokljat, see [19] or [8], and with a three-equation $\gamma - SST$ model by Menter et al. [11]. One can see that both models capture very well the heat flux in the laminar part of the flow, the transition onset at the suction side is better predicted by the $\gamma - SST$ model, and the heat flux is over-predicted by both models in the rear part of the suction side.

4 Conclusion

The matrix-free steady-state LU-SGS solver for compressible turbulent flow is described. The solvers shows very

good accuracy for several test cases ranging from inviscid low Mach number flow over a bump to transonic turbulent flow through turbine cascade. The convergence to steady state is superior to segregated SIMPLE solver in the case of transonic flows. The convergence is a bit worse for low Mach number flows but it should be possible to improve it using a preconditioning. The LU-SGS solver provides very robust method which doesn't need any case dependent tuning (like e.g. relaxation factors in SIMPLE method for transonic flows). The main advantage of OpenFOAM platform is that the LU-SGS solver can be combined with a lot of built-in turbulence models as well as with the post-processing or parallel computing provided by the platform.

Acknowledgements. This research has been realized using the support of the Czech Science Foundation under grant P101/12/1271.

References

- [1] ARTS, T., LAMBERT DE ROUVROIT, M., and RUTHERFORD, A.W. *Aero-thermal investigation of a highly loaded transonic linear turbine guide vane cascade: A test case for inviscid and viscous flow computations*. Technical Note 174. Von Karman Institute for Fluid Dynamics, 1990, p. 97.
- [2] BARTH, Timothy and JESPERSEN, Dennis. "The design and application of upwind schemes on unstructured meshes". In: *27th Aerospace Sciences Meeting*. Reston, Virginia: American Institute of Aeronautics and Astronautics, 1989.
- [3] BLAZEK, Jiri. *Computational fluid dynamics : principles and applications*. Butterworth-Heinemann, 2015, p. 466.
- [4] FERZIGER, Joel H. and PERIC, Milovan. *Computational Methods for Fluid Dynamics*. Springer Science & Business Media, 2012, p. 426.
- [5] FÜRST, Jiří. "A weighted least square scheme for compressible flows". In: *Flow, Turbulence and Combustion* 76.4 (2006), pp. 331–342.
- [6] GREENSHIELDS, Christopher J. et al. "Implementation of semi-discrete, non-staggered central schemes in a colocated, polyhedral, finite volume framework, for high-speed viscous flows". In: *International Journal for Numerical Methods in Fluids* 63.1 (2010), pp. 1–21.
- [7] ISSA, R.I. "Solution of the implicitly discretised fluid flow equations by operator-splitting". In: *Journal of Computational Physics* 62.1 (1986), pp. 40–65.
- [8] KOŽIŠEK, Martin et al. "Implementation of k-kL- ω Turbulence Model for Compressible Flow into OpenFOAM". In: *Applied Mechanics and Materials* 821 (2016), pp. 63–69.
- [9] LIOU, Meng-Sing. "A sequel to AUSM, Part II: AUSM+-up for all speeds". In: *Journal of Computational Physics* 214.1 (2006), pp. 137–170.
- [10] MENTER, F. R. "Two-equation eddy-viscosity turbulence models for engineering applications". en. In: *AIAA Journal* 32.8 (1994), pp. 1598–1605.
- [11] MENTER, Florian R. et al. "A one-equation local correlation-based transition model". In: *Flow, Turbulence and Combustion* 95.4 (2015), pp. 583–619.
- [12] MOUKALLED, F. and DARWISH, M. "A High-Resolution Pressure-Based Algorithm for Fluid Flow at All Speeds". In: *Journal of Computational Physics* 168.1 (2001), pp. 101–130.
- [13] ŠAFARŮK, Pavel. "Experimental data from optical measurement tests on a transonic turbine blade cascade". In: *13th Symposium on Measuring Techniques for Transonic and Supersonic Flow in Cascades and Turbomachines*. Ed. by G. Gyarmathy) C. GROSSWEILER. ETH Zurich, 1996, pp. 0–14.
- [14] SHEN, Chun, SUN, Fengxian, and XIA, Xinlin. "Implementation of density-based solver for all speeds in the framework of OpenFOAM". In: *Computer Physics Communications* 185.10 (2014), pp. 2730–2741.
- [15] SHEN, Chun et al. "Implementation of density-based implicit LU-SGS solver in the framework of OpenFOAM". In: *Advances in Engineering Software* 91 (2016), pp. 80–88.
- [16] ŠIMURDA, David, FÜRST, Jiří, and LUXA, Martin. "3D flow past transonic turbine cascade SE 1050 - Experiment and numerical simulations". In: *Journal of Thermal Science* 22.4 (2013), pp. 311–319.
- [17] TORO, Eleuterio F. "The HLL and HLLC Riemann Solvers". In: *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997, pp. 293–311.
- [18] VENKATAKRISHNAN, V. "On the accuracy of limiters and convergence to steady state solutions". In: *31st Aerospace Sciences Meeting & Exhibit* (1993), pp. 1–10.
- [19] WALTERS, D. Keith and COKLJAT, Davor. "A three-equation eddy-viscosity model for Reynolds-averaged Navier–Stokes simulations of transitional flow". In: *Journal of Fluids Engineering* 130.12 (2008), pp. 121401–121401–14.
- [20] WELLER, H. G. et al. "A tensorial approach to computational continuum mechanics using object-oriented techniques". In: *Computers in Physics* 12.6 (1998), p. 620.
- [21] XISTO, Carlos M., PÁSCOA, José C., and OLIVEIRA, Paulo J. "A pressure-based high resolution numerical method for resistive MHD". In: *Journal of Computational Physics* 275.151 (2014), pp. 323–345.