



Analysis of the Accuracy of OpenFOAM Solvers for the Problem of Supersonic Flow Around a Cone

Alexander E. Bondarev^(✉)  and Artem E. Kuvshinnikov 

Keldysh Institute of Applied Mathematics RAS, Moscow, Russia
bond@keldysh.ru, kuvsh90@yandex.ru

Abstract. The numerical results of comparing the accuracy for some OpenFOAM solvers are presented. The comparison was made for the problem of inviscid compressible flow around a cone at zero angle of attack. The results obtained with the help of various OpenFOAM solvers are compared with the known numerical solution of the problem with the variation of cone angle and flow velocity. This study is a part of a project aimed to create a reliable numerical technology for modelling the flows around elongated bodies of rotation (EBR).

Keywords: Flow around a cone · Computational fluid dynamics
OpenFOAM · Accuracy comparison

1 Introduction

In recent years, there has been a fairly frequent situation where it is necessary to calculate the flow of elongated bodies of rotation (EBR) under specific conditions. Such calculations are usually carried out for practical purposes, taking into account all technological features of the body in question. Naturally, for such calculations, there is a desire to apply some of CFD software packages, which have been widely used in recent times. However, when trying to solve practical problems with the help of such packages, there are some difficulties. The catalogs of mathematical models and finite-difference schemes used in such complexes are imperfect. The acceptability of many models for solving complex problems and determining the limits of their applicability are the subject of a separate study. This refers to the problems of flow around the elongated bodies of rotation and the implementation of turbulence simulation methods for them. For a particular class of EBR it is required to carry out a large number of test calculations to show the applicability of the chosen numerical method and the chosen model of turbulence. These methodological studies are often neglected. Therefore, a user of similar software packages encounters the need to specify a variety of variable parameters, which in practice provides an indefinite result.

In this situation, obviously, we need a computational technology that would be a kind of standard for solving the problems of flow around the EBR and would help to regulate the tunable parameters of both numerical methods and models of turbulence in various software packages. In this capacity, it was decided to recreate on the modern

level the computational technology developed earlier in the Keldysh Institute of Applied Mathematics. In the late 80's - early 90's this computational technology allowed to make mass industrial computing for a flow around EBR with a high degree of reliability. The error of aerodynamic drag coefficients did not exceed 2–3% in comparison with the experimental results. The essence of this technology was that the aerodynamic drag coefficient C_x , was considered as a sum of three components: C_p – coefficient for inviscid flow, C_f - coefficient for viscous friction and C_d – coefficient for near wake pressure. Such an approach was widely used for industrial analysis of aerodynamic properties of EBR and proved to be very effective. The work presented is a part of the general project to create a similar technology [1, 2]. To calculate the friction coefficient, a computational technique [2] is realized. The technique is based on an approximate semi-empirical model which combines the results of experimental studies and the method of effective length. This computational technology is designed to determine the friction coefficient and estimate the characteristics of the boundary layer for EBR. To calculate the aerodynamic characteristics for inviscid flow around the elongated bodies of rotation, it was proposed to use the OpenFOAM software package (Open Source Field Operation and Manipulation CFD Toolbox) [3]. OpenFOAM is actively used in industry and in science. OpenFOAM contains a number of solvers [4–7] having different computational properties.

Therefore, it is necessary to make methodological calculations that allow to evaluate the effectiveness of these solvers for practical application. This paper presents a comparative analysis of the OpenFOAM solvers accuracy for the problem of inviscid flow around cones with different angles and different flow velocities at zero angle of attack. Tabular solutions [8] are used as an exact solution for comparison. Presented in a tabular form solutions [8] have high accuracy and for many years are used for testing the computational properties of numerical methods. It should be noted that similar comparisons of solvers were carried out in [9, 10]. However, these comparisons do not give full and clear recommendations on the accuracy of solvers.

2 Formulation of the Problem

The statement of the problem is presented in full accordance with [8], where the results of the inviscid flow around cones with different angles at various Mach numbers are considered. We consider the case of a cone placed in a uniform supersonic flow of an ideal gas at zero angle of attack $\alpha = 0^\circ$ with a Mach number of 2 to 7. The body under investigation is a cone with an angle $\beta = 10\text{--}35^\circ$ in steps of 5° . Here angle β is a half of cone angle as shown in Fig. 1. The conditions of the input flow are denoted by the index “ ∞ ”, and at the output by the index ξ , since the solution is self-similar and depends on the dimensionless variable. The Euler equations system is used for the calculation. The system is supplemented by the equation of state of an ideal gas.

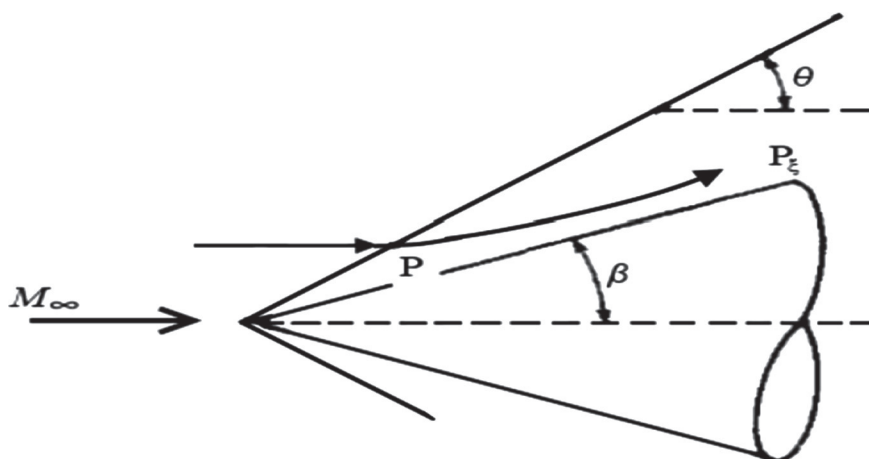


Fig. 1. Flow scheme.

3 OpenFOAM Solvers

For comparison, 4 solvers were selected from the OpenFOAM software package:

RhoCentralFoam is based on a central-upwind scheme, which is a combination of central-difference and upwind schemes [4, 5]. The essence of the central-upwind schemes consists in a special choice of a control volume containing two types of domains: around the boundary points, the first type; around the center point, the second type. The boundaries of the control volumes of the first type are determined by means of local propagation velocities. The advantage of these schemes is that, using the appropriate technique to reduce the numerical viscosity, it is possible to achieve good solvability for discontinuous solutions — shock waves in gas dynamics, and for solutions in which viscous phenomena play a major role.

SonicFoam is based on the PISO algorithm (Pressure Implicit with Splitting of Operator) [6]. The basic idea of the PISO method is that two difference equations are used to calculate the pressure for the correction of the pressure field obtained from discrete analogs of the equations of moments and continuity. This approach is due to the fact that the velocities corrected by the first correction may not satisfy the continuity equation, therefore, a second corrector is introduced which allows us to calculate the velocities and pressures satisfying the linearized equations of momentum and continuity.

RhoPimpleFoam is based on the PIMPLE algorithm, which is a combination of the PISO and SIMPLE (Semi-Implicit Method for Pressure-Linked Equations) algorithms. An external loop is added to the PISO algorithm, thanks to which the method becomes iterative and allows to count with the Courant number greater than 1.

PisoCentralFoam is a combination of a Kurganov-Tadmor scheme [4] with the PISO algorithm [7].

For all solvers the calculations were carried out using the OpenFOAM version 2.3.0. Solver *sonicFoam* in the standard version does not support dynamic time step change, so the necessary corrections have been made to the code of solver. Also the calculations were made for *pimpleCentralFoam* solver. This solver exists only for OpenFOAM version 3.0.1 and higher. The results for this solver were similar to the results of *pisoCentralFoam*, so it was decided not to include these results in the tables below.

4 Computations and Results

4.1 Mesh Generation, Initial and Boundary Conditions

Figure 2 shows the computational domain. On the upper boundary indicated as “top”, the zero gradient condition for the gas dynamic functions, is specified. The same conditions are set on the right border, denoted by “outlet”. On the left border, designated as “inlet”, the parameters of the oncoming flow are set: pressure $P = 101325$ Pa, temperature $T = 300$ K, speed U from 694.5 m/s (Mach number = 2) to 2430.75 m/s (Mach number = 7). On the boundary of the cone (“cone”) for pressure and temperature, the condition of zero gradient is given, for the speed is given the condition “slip”, corresponding to the non-flow condition for the Euler equations. To model the axisymmetric geometry in the OpenFoam package, a special “wedge” condition is used for the front (“front”) and back (“back”) borders. The OpenFoam package also introduces a special “empty” boundary condition. This condition is specified in cases when calculations in a given direction are not carried out. In our case, this condition is used for the “axis” border.

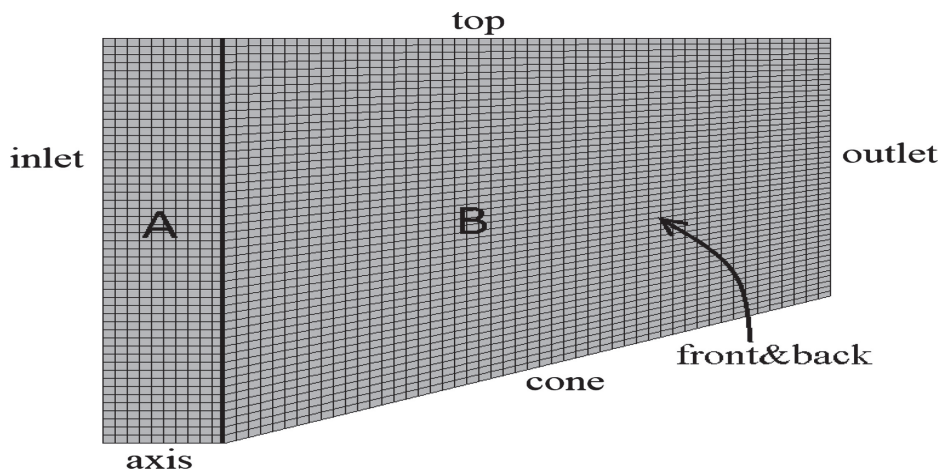


Fig. 2. Computational domain and boundaries.

The number of grid cells is 13200. The initial conditions correspond to the boundary conditions on the inlet edge, that is, the initial conditions are used for the parameters of the oncoming stream. The molar mass $M = 28.96$ kg/mol and the specific heat at constant pressure $C_p = 1004$ were also set.

To estimate the effect of the grid partition on the accuracy of calculations, the calculations were carried out on three grids, denoted as *coarse*, *fine*, *finest*. The number of cells: *coarse* – 3000, *fine* – 12000, *finest* – 48000.

4.2 Parameters of Solvers

In the OpenFOAM package, there are two options for approximating differential operators: directly in the solver’s code or using the fvSchemes and fvSolution configuration files. In order for the comparison to be correct, we used the same parameters where

possible. In the fvSchemes file: ddtSchemes – *Euler*, gradSchemes – *Gauss linear*, divSchemes – *Gauss linear*, laplacianSchemes – *Gauss linear corrected*, interpolationSchemes – *vanLeer*. In the fvSolution file: solver – *smoothSolver*, smoother *symGaussSeidel*, tolerance – $1e-09$, nCorrectors – 2, nNonOrthogonalCorrectors – 1.

4.3 Calculation of the Axisymmetric Flow

Figure 3 presents the steady-state flow field for pressure when using the solver rhoCentralFoam. The figure indicates that, as a result of the establishment, a qualitative picture of the flow is obtained that corresponds to the known solutions [8].

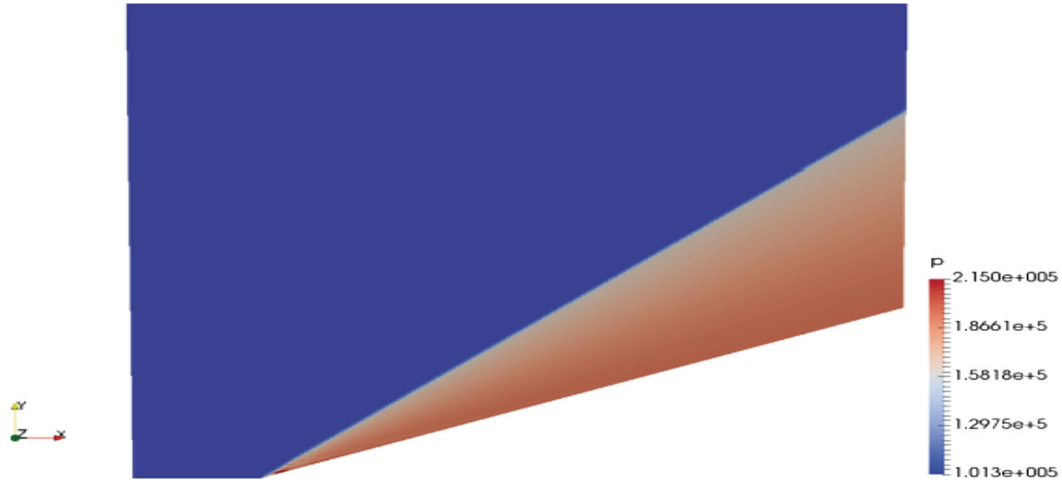


Fig. 3. Pressure field for steady flow.

Tables from 1, 2, 3, 4, 5, 6, 7, 8 and 9 show the results of calculations in the form of an analog of the L_2 norm:

$$\sqrt{\sum_m |y_m - y_m^{exact}|^2 V_m} / \sqrt{\sum_m |y_m^{exact}|^2 V_m} \quad (1)$$

In Tables 1, 2 and 3 y_m is the velocity U_x , U_y , pressure p and density ρ in the cell, V_m is the cell volume for the cone angle $\beta = 20^\circ$ and the Mach number $M = 2$. These tables show the grid convergence for the variant considered. Grid convergence for all variants was considered similarly.

In Tables 4, 5, 6, 7, 8 and 9 y_m is the pressure p in the cell, V_m is the cell volume for the cone angle $\beta = 10-35^\circ$ in steps of 5° and the Mach numbers $M = 2-7$. The minimum values are highlighted in bold. The symbol “x” in the tables means that at a given speed and given cone angle, the solver became unstable. The values of y_m^{exact} are obtained by interpolating tabular values from [8] into grid cells. It should be noted that the authors of the tables [8] indicate the admissibility of interpolation for all parameters and table values. Further we will use abbreviations for solvers. rCF (rhoCentralFoam), pCF (pisoCentralFoam), sF (sonicFoam), rPF (rhoPimpleFoam).

Table 1. Deviation from the exact solution for coarse grid

	rCF	pCF	sF	rPF
U_x	0.009062	0.008929	0.008366	0.010155
U_y	0.043725	0.050789	0.050932	0.060268
p	0.024054	0.027705	0.033429	0.037406
ρ	0.018327	0.021848	0.028965	0.033199

Table 2. Deviation from the exact solution for fine grid

	rCF	pCF	sF	rPF
U_x	0.006268	0.006482	0.005809	0.007588
U_y	0.029656	0.034403	0.033814	0.043562
p	0.016989	0.019515	0.022465	0.026656
ρ	0.012834	0.015182	0.019085	0.022994

Table 3. Deviation from the exact solution for finest grid

	rCF	pCF	sF	rPF
U_x	0.004372	0.004441	0.004057	0.005526
U_y	0.019862	0.022855	0.023113	0.030994
p	0.011611	0.013269	0.015143	0.018803
ρ	0.008715	0.010282	0.012684	0.015810

Table 4. Deviation from the exact solution, $U = 2M$

Cone angel	rCF	pCF	sF	rPF
10	0.006090	0.006973	0.010153	0.010341
15	0.012654	0.014446	0.019646	0.020645
20	0.016623	0.019353	0.022283	0.024951
25	0.018678	0.020948	0.020779	0.025426
30	0.020695	0.023130	0.025614	0.023267
35	0.032486	0.038658	0.074849	0.043179

Table 5. Deviation from the exact solution, $U = 3M$

Cone angle	rCF	pCF	sF	rPF
10	0.015309	0.019537	0.027152	0.027177
15	0.024608	0.030041	0.047813	0.041444
20	0.030440	0.035858	0.070564	0.045760
25	0.032486	0.038658	0.074849	0.043179
30	0.034040	0.040603	0.077408	0.040006
35	0.026334	0.029821	0.044853	0.027077

Table 6. Deviation from the exact solution, $U = 4M$

Cone angle	rCF	pCF	sF	rPF
10	0.028254	0.035251	0.058133	0.049334
15	0.040229	0.046494	0.106172	0.065384
20	0.046159	0.052687	0.126701	0.070649
25	0.045849	0.051912	0.134932	0.062785
30	0.040775	0.050619	0.109125	x
35	0.034277	0.042296	0.069668	x

Table 7. Deviation from the exact solution, $U = 5M$

Cone angle	rCF	pCF	sF	rPF
10	0.050834	0.055133	0.106710	0.075829
15	0.060069	0.063293	0.159880	0.090489
20	0.060174	0.064675	0.175666	x
25	0.059900	0.063284	0.175205	x
30	0.055975	0.062637	0.130201	x
35	0.043288	0.052737	0.090006	x

Table 8. Deviation from the exact solution, $U = 6M$

Cone angle	rCF	pCF	sF	rPF
10	0.061150	0.063986	0.148118	0.093482
15	0.077744	0.077303	0.215881	0.455342
20	0.076336	0.078191	0.210225	x
25	0.073101	0.075504	0.183841	x
30	0.063374	0.067209	0.144629	x
35	0.052961	0.062369	0.096355	x

Table 9. Deviation from the exact solution, $U = 7M$

Cone angle	rCF	pCF	sF	rPF
10	0.076287	0.074444	0.191676	0.112744
15	0.090901	0.089137	0.247274	0.543143
20	0.086889	0.089311	0.215352	x
25	0.085631	0.087697	0.188621	x
30	0.073957	0.077507	0.140091	x
35	0.063160	0.075632	0.111154	x

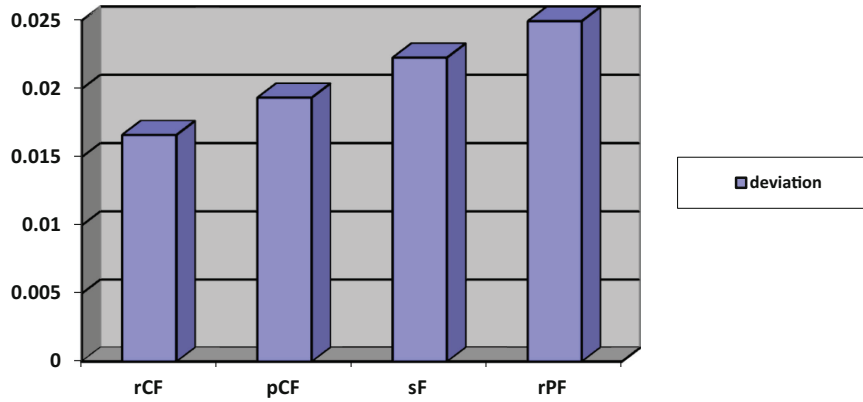
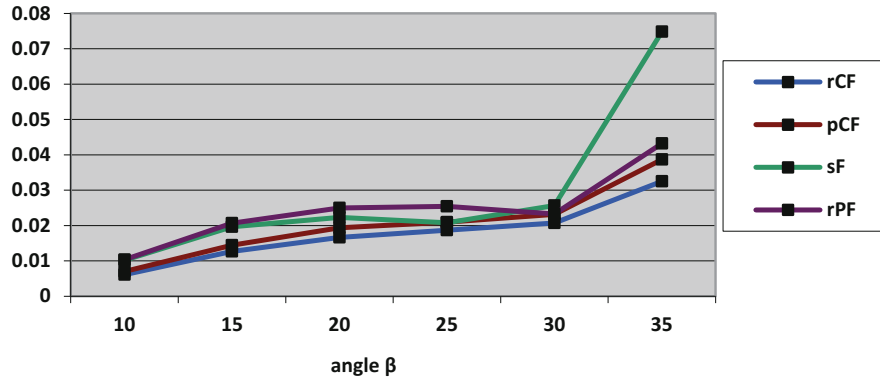
**Fig. 4.** Deviation from the exact solution for pressure $M = 2$, $\beta = 20$.**Fig. 5.** Change in deviation from the exact solution for pressure depending on the cone angle for all solvers, $M = 2$.

Figure 4 presents a diagram of the deviation from the exact solution in the analogue of the L_2 norm for the pressure for all used solvers by the example of the problem of flow past a cone with a cone angle $\beta = 20^\circ$ with Mach number $M = 2$. The smallest deviation from the exact solution is shown by the solver rhoCentralFoam, the maximum deviation is shown by the solver rhoPimpleFoam.

Figure 5 shows the change in the deviation from the exact solution in the analogue of the L_2 norm for the pressure for all solvers, depending on the cone angle for a fixed Mach number $M = 2$. The smallest deviation from the exact solution is shown by the

solver rhoCentralFoam, the largest deviation with an increase in the cone angle is shown by the sonicFoam solver.

Figure 6 shows the dependence of the deviation on the exact solution in the analog of the L_2 norm for the pressure for the solver rhoCentralFoam with the variation of the cone angle and the initial velocity. An increase in the Mach number of the oncoming stream has the greatest effect on the increase in the deviation of the numerical result from the exact solution.

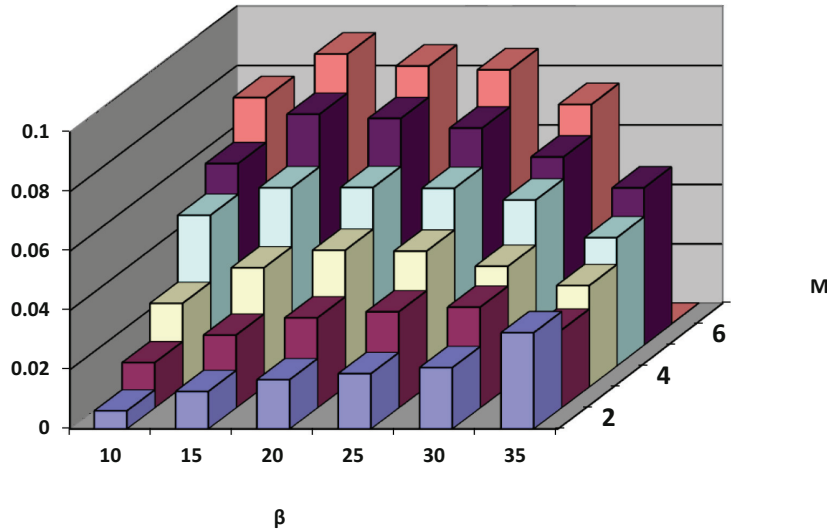


Fig. 6. Change in deviation from the exact solution for pressure depending on the cone angle and the velocity for the solver rhoCentralFoam.

5 Conclusion

Using well-known problem of a supersonic inviscid flow around a cone at zero angle of attack we compared four OpenFOAM solvers with the exact solution. Grid convergence was shown for all solvers. According to the results obtained, the solver rhoCentralFoam has minimal error in almost all cases. The only drawback of rhoCentralFoam is the appearance of oscillations near the surface at the head of the cone. Solver pisoCentralFoam is in second place in accuracy, however, when using this solver, the appearance of oscillations is not observed. The methodical research can serve as a basis for selecting the OpenFOAM solver for calculating the inviscid supersonic flow around the elongated bodies of rotation. The results of solvers comparison can also be useful for developers of OpenFOAM software content. The results obtained made it possible to get a general idea of the calculation errors for all solvers.

In further studies it is proposed to make a similar comparison of solvers for the problem of flow around a cone with a variation of the angle of attack. It is also proposed to investigate the matrix of mutual errors for solutions obtained by different solvers by constructing elastic maps.

Acknowledgments. This work was supported by grant of RSF № 18-11-00215.

References

1. Bondarev, A.E., Kuvshinnikov, A.E.: Comparative study of the accuracy for OpenFOAM solvers. In: Proceedings of Ivannikov ISPRAS Open Conference (ISPRAS), p. 132. IEEE, Moscow (2017). <https://doi.org/10.1109/ispras.2017.00028>
2. Bondarev, A.E., Nesterenko, E.A.: Approximate method for estimation of friction forces for axisymmetric bodies in viscous flows. *Mathematica Montisnigri* **XXXI**, 54–63 (2014)
3. OpenFOAM. <http://www.openfoam.org>. Accessed 11 Feb 2018
4. Kurganov, A., Tadmor, E.: New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations. *J. Comput. Phys.* **160**, 241–282 (2000). <https://doi.org/10.1006/jcph.2000.6459>
5. Greenshields, C., Wellerr, H., Gasparini, L., Reese, J.: Implementation of semi-discrete, non-staggered central schemes in a colocated, polyhedral, finite volume framework, for high-speed viscous flows. *Int. J. Numer. Meth. Fluids* **63**(1), 1–21 (2010). <https://doi.org/10.1002/fld.2069>
6. Issa, R.: Solution of the implicit discretized fluid flow equations by operator splitting. *J. Comput. Phys.* **62**(1), 40–65 (1986). [https://doi.org/10.1016/0021-9991\(86\)90099-9](https://doi.org/10.1016/0021-9991(86)90099-9)
7. Kraposhin, M., Bovtrikova, A., Strijhak, S.: Adaptation of Kurganov-Tadmor numerical scheme for applying in combination with the PISO method in numerical simulation of flows in a wide range of Mach numbers. *Procedia Comput. Sci.* **66**, 43–52 (2015). <https://doi.org/10.1016/j.procs.2015.11.007>
8. Babenko, K.I., Voskresenskii, G.P., Lyubimov, A.N., Rusanov, V.V.: Three-Dimensional Ideal Gas Flow Past Smooth Bodies. Nauka, Moscow (1964). (in Russian)
9. Karvatskii, A., Pulinets, I., Lazarev, T., Pedchenko, A.: Numerical modelling of supersonic flow around a wedge with the use of free open software code OpenFOAM. *Space Sci. Technol.* **21**(2), 47–52 (2015). <https://doi.org/10.15407/knit2015.02.047>. (in Russian)
10. Gutierrez, L.F., Tamagno, J.P., Elaskar, S.A.: High speed flow simulation using OpenFOAM. In: *Mecanica Computacional*, Salta, Argentina, vol. XXXI, pp. 2939–2959 (2012)