Journal of Computational Physics 229 (2010) 461–480

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

# Pressure based finite volume method for calculation of compressible viscous gas flows

Kiril S. Shterev *, Stefan K. Stefanov

*Institute of Mechanics, Bulgarian Academy of Sciences, Acad G. Bonchev Str., Block 4, Sofia 1113, Bulgaria*

ARTICLE INFO

*Article history:*
Received 7 September 2008
Received in revised form 25 September 2009
Accepted 29 September 2009
Available online 6 October 2009

*PACS:*
65Z05
76M12
76J20
76N15
76P05

*Keywords:*
Finite volume method
DSMC
Compressible flows
Microflows

ABSTRACT

A pressure based, iterative finite volume method is developed for calculation of compressible, viscous, heat conductive gas flows at all speeds. The method does not need the use of under-relaxation coefficient in order to ensure a convergence of the iterative process. The method is derived from a general form of system of equations describing the motion of compressible, viscous gas. An emphasis is done on the calculation of gaseous microfluidic problems. A fast transient process of gas wave propagation in a two-dimensional microchannel is used as a benchmark problem. The results obtained by using the new method are compared with the numerical solution obtained by using SIMPLE (iterative) and PISO (non-iterative) methods. It is shown that the new iterative method is faster than SIMPLE. For the considered problem the new method is slightly faster than PISO as well. Calculated are also some typical microfluidic subsonic and supersonic flows, and the Rayleigh–Bénard convection of a rarefied gas in continuum limit. The numerical results are compared with other analytical and numerical solutions.

## 1. Introduction

The rapid growth of microfluidic and nanofluidic technologies suggests new challenges for computational fluid dynamics. For example, the computational analysis of many problems, concerning gaseous flows in micron- and submicron-size devices, cannot be based on the classical continuum fluid models, which are no longer valid at micron and submicron scales. In such flows the mean free path of the gas molecules is comparable to the characteristic size of the device and the kinetic effects of rarefaction and non-equilibrium must be taken into consideration [1–3]. For small Knudsen number $Kn < 0.1$ ($Kn = \ell_0/L$ , where $\ell_0$ is the mean free path of the gas molecules and $L$ is the characteristic length), a continuum approach based on modified Navier–Stokes–Fourier or extended hydrodynamic [4–6] continuum models with corresponding velocity-slip and temperature-jump boundary conditions is still applicable and, respectively, preferable. However, for larger Knudsen numbers, where the non-equilibrium effects are significant, a molecular approach, based on kinetic theory models [7,8], or the particle DSMC (Direct Simulation Monte Carlo) method [9] has to be used. In the numerical examples, given in the present paper, we restrict ourself to the use of Navier–Stokes–Fourier continuum model [10] with state-depended transport coefficients determined by the first approximation of the Chapman–Enskog theory for low Knudsen numbers. In our opinion this model captures the basic flow effects of the motion of compressible viscous heat-conducting gas in continuum

[*] Corresponding author. Tel.: +359 2 979 6488; fax: +359 2 870 7498.
*E-mail address:* kshterev@imbm.bas.bg (K.S. Shterev).

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.09.042

limit. The mentioned above reasons lead to corresponding modifications of the numerical method for calculation of gaseous microflows. To this aim, a new iterative pressure based finite volume method (FVM) is developed, in which the iterative process has an improved within a time step convergence compared to the other standard FVM iterative schemes. Since the developed approach is valid also for other classic compressible flow models, the equation system and the computational considerations are given in a unified two-dimensional form. The extension of the proposed approach for calculation of three-dimensional gaseous flows is straightforward. The widely used pressure based methods [11–20] can be classified into two groups: SIMPLE-like methods used for calculation of steady and low or moderate speed unsteady flows and some methods specially developed for calculation of transient flows. The representative of SIMPLE-like methods are SIMPLE, SIMPLER, SIMPLEST-ANL, SIMPLEC. All of them have an important disadvantage, when are applied for calculation of unsteady flows. It is related to the existence of a term $\partial\rho/\partial t = (\rho - \rho^{n-1})/\Delta t$ in the equation for pressure, which makes the standard iterative scheme unstable. To make the iterative scheme converging one has use under-relaxation coefficients. A similar problem has been considered by Issa [19,21], by using his method PISO, a non-iterative method developed for transient fluid flow calculations. In PISO density in the mentioned term is substituted with pressure derived from the equation of state. We have adopted this idea in the new iterative algorithm by substituting density with pressure in the mentioned term and rearranging the order of calculated equations in order to keep the equation of state satisfied within each iteration. Thus, the energy and pressure equations are calculated in a separate internal iterative loop placed in the computational algorithm after the calculation of the pseudo velocities and before the calculation of velocities. In this way we ensure the convergence of the iterative process without using under-relaxation coefficients. A detailed comparison of the numerical results obtained by the standard iterative SIMPLE, the non-iterative PISO and the new iterative method SIMPLE-TS (TS – Time Step) is presented in the paper for two flows: a low speed gas flow in a microchannel and a supersonic flow past a square confined in a channel. The efficiency analysis of the methods shows that the SIMPLE-TS is much faster than the standard SIMPLE. The efficiency of the new method is better than that of PISO in the cases when the calculations are performed by both methods with the same accuracy. The proposed approach is addressed mainly to problems in microfluidics and their numerical analysis, however its universality makes it applicable also for calculation of classic compressible Navier–Stokes–Fourier flows at all speeds. The numerical results obtained for the considered microfluidic problems are compared also to the Arkilic's analytical solution [22] (for a plain channel flow) and to the computational data obtained by the authors using the direct Monte Carlo simulation (DSMC) method [9,10].

## 2. Basic equations and computational considerations

Seeking more generality we consider a unified two-dimensional system of equations describing the time evolution of a unsteady, viscous, compressible, heat-conducting gas flow. The analysis can be extended to a three-dimensional form in a straightforward manner. After an appropriate scaling the considered non-dimensional equation system reads as follows:

$$\frac{\partial\rho}{\partial t}+\frac{\partial(\rho u)}{\partial x}+\frac{\partial(\rho v)}{\partial y}=0 \tag{1}$$

$$\begin{aligned}\frac{\partial(\rho u)}{\partial t}+\frac{\partial(\rho uu)}{\partial x}+\frac{\partial(\rho vu)}{\partial y}=\rho g_x-A\frac{\partial p}{\partial x}+B\left[\frac{\partial}{\partial x}\left(\Gamma\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left(\Gamma\frac{\partial u}{\partial y}\right)\right]\\+B\left\{\frac{\partial}{\partial x}\left(\Gamma\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left(\Gamma\frac{\partial v}{\partial x}\right)-\frac{2}{3}\frac{\partial}{\partial x}\left[\Gamma\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}\right)\right]\right\}\end{aligned} \tag{2}$$

$$\begin{aligned}\frac{\partial(\rho v)}{\partial t}+\frac{\partial(\rho uv)}{\partial x}+\frac{\partial(\rho vv)}{\partial y}=\rho g_y-A\frac{\partial p}{\partial y}+B\left[\frac{\partial}{\partial x}\left(\Gamma\frac{\partial v}{\partial x}\right)+\frac{\partial}{\partial y}\left(\Gamma\frac{\partial v}{\partial y}\right)\right]\\+B\left\{\frac{\partial}{\partial y}\left(\Gamma\frac{\partial v}{\partial y}\right)+\frac{\partial}{\partial x}\left(\Gamma\frac{\partial u}{\partial y}\right)-\frac{2}{3}\frac{\partial}{\partial y}\left[\Gamma\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}\right)\right]\right\}\end{aligned} \tag{3}$$

$$\frac{\partial(\rho T)}{\partial t}+\frac{\partial(\rho uT)}{\partial x}+\frac{\partial(\rho vT)}{\partial y}=C^{T1}\left[\frac{\partial}{\partial x}\left(\Gamma^{\lambda}\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(\Gamma^{\lambda}\frac{\partial T}{\partial y}\right)\right]+C^{T2}\cdot\Gamma\cdot\Phi+C^{T3}\frac{Dp}{Dt} \tag{4}$$

$$p=\rho T \tag{5}$$

where:

$$\Phi=2\left[\left(\frac{\partial u}{\partial x}\right)^2+\left(\frac{\partial v}{\partial y}\right)^2\right]+\left(\frac{\partial v}{\partial x}+\frac{\partial u}{\partial y}\right)^2-\frac{2}{3}\left(\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}\right)^2 \tag{6}$$

$u$ is the horizontal component of velocity, $v$ is the vertical component of velocity, $p$ is pressure, $T$ is temperature, $\rho$ is density, $t$ is time, $x$ and $y$ are coordinates of a Cartesian coordinate system. The reference quantities used to scale the system of equations (1)–(5) will be defined later, separately for each considered problem. Parameters $A$, $B$, $g_x$, $g_y$, $C^{T1}$, $C^{T2}$, $C^{T3}$ and diffusion coefficients $\Gamma$ and $\Gamma^{\lambda}$, given in Eqs. (1)–(5), depend on the gas model and the equation non-dimensional form.

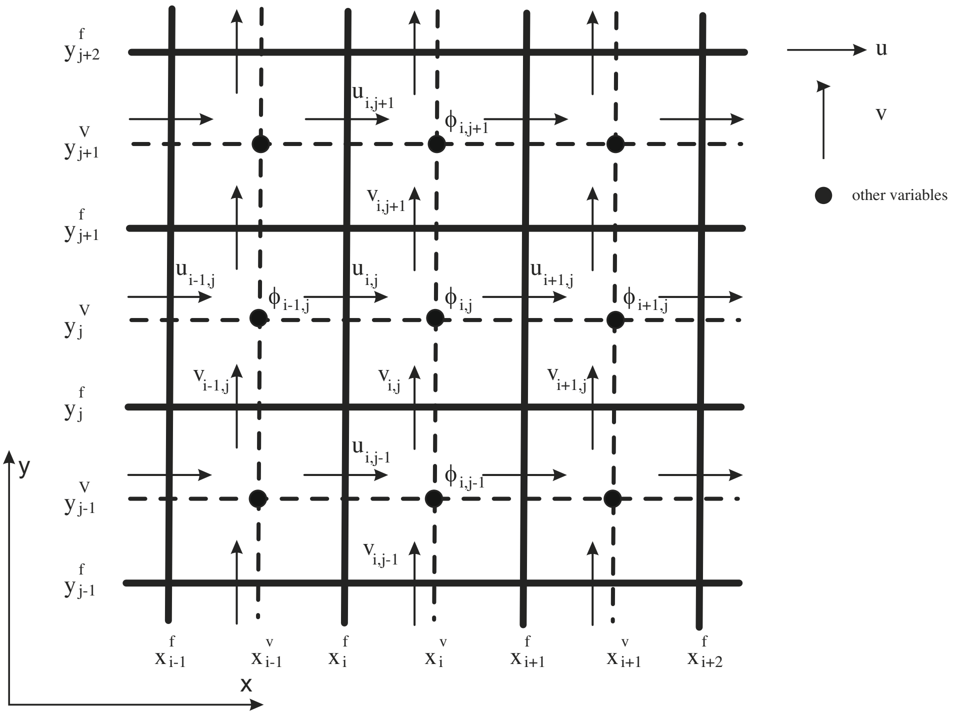


**Fig. 1.** Cell volume.

The discretization of the equation system is accomplished by using a backward staggered velocity grid, in which all dependent field variables (pressure, temperature and density) are calculated at a cell centre and all flow variables (velocity components) are calculated at the surfaces of a cell (Fig. 1). For interpolation between two neighbor points a piecewise-linear profile is used. In that way we approximate the derivations of second order [16]. For the interpolation of convective terms a first order upwind scheme is used. In Fig. 1 the following grid variables are denoted:

$x_i^f$ – the left frontier $x$-coordinate of the control volume of node $i$

$y_j^f$ – bottom frontier $y$-coordinate of the control volume of node $j$

$\Delta x_i = x_{i+1}^f - x_i^f$ – step on OX

$\Delta y_j = y_{j+1}^f - y_j^f$ – step on OY

$x_i^v$ – $x$-coordinate of the centre of the control volume of node $i$, $x_i^v = x_i^f + 0.5\Delta x_i$

$y_j^v$ – $y$-coordinate of the centre of the control volume of node $j$, $y_j^v = y_j^f + 0.5\Delta y_j$

$\phi_{i,j}$ – field variables defined at point $\left(x_i^v, y_j^v\right)$

$u_{i,j}$ – the horizontal component of velocity $\left(x_i^f, y_j^v\right)$

$v_{i,j}$ – the vertical component of velocity defined at point $\left(x_i^v, y_j^f\right)$

$F_{i,j}^x$ – the convective mass flux through the surface between control volumes $(i-1,j)$ and $(i,j)$ (in horizontal direction)

$$F_{i,j}^x = \rho_{i,j}^u u_{i,j} \Delta y_j \tag{7}$$

$F_{i,j}^y$ – the convective mass flux through the surface between control volumes $(i,j-1)$ and $(i,j)$ (in vertical direction)

$$F_{i,j}^y = \rho_{i,j}^v v_{i,j} \Delta x_i \tag{8}$$

where:

$$\rho_{i,j}^u = \begin{cases} 0.5(\rho_{i-1,j} + \rho_{i,j}) & \text{if } u_{i,j} = 0 \\ \rho_{i-1,j} & \text{if } u_{i,j} > 0 \\ \rho_{i,j} & \text{if } u_{i,j} < 0 \end{cases} \tag{9}$$

$$\rho_{i,j}^v = \begin{cases} 0.5(\rho_{i,j-1} + \rho_{i,j}) & \text{if } v_{i,j} = 0 \\ \rho_{i,j-1} & \text{if } v_{i,j} > 0 \\ \rho_{i,j} & \text{if } v_{i,j} < 0 \end{cases} \tag{10}$$

Density $(\rho)$ is computed at the middle points, i.e. $\rho^u_{i,j}$ at point $\left(x^f_i, y^v_j\right)$ and $\rho^v_{i,j}$ at point $\left(x^v_i, y^f_j\right)$, by using upwind scheme [23,24]. The upwind calculation of density is of first order accuracy, but it brings more stability of the calculations, when a supersonic fluid flow is considered.

For brevity, here a finite volume discretization of the momentum equations is given for the $y$-momentum (3) only. The equation is written in the following convenient form:

$$\underbrace{\frac{\partial(\rho v)}{\partial t}}_{\text{Unsteady term}} + \underbrace{\frac{\partial(\rho u v)}{\partial x} + \frac{\partial(\rho v v)}{\partial y}}_{\text{Convective terms}} = - \underbrace{A\frac{\partial p}{\partial y}}_{\text{Pressure term}} + \underbrace{B\left[\frac{\partial}{\partial x}\left(\Gamma\frac{\partial v}{\partial x}\right) + \frac{\partial}{\partial y}\left(\Gamma\frac{\partial v}{\partial y}\right)\right]}_{\text{Diffusion terms}}$$
$$\underbrace{+\rho g_y + B\left\{\frac{\partial}{\partial y}\left(\Gamma\frac{\partial v}{\partial y}\right) + \frac{\partial}{\partial x}\left(\Gamma\frac{\partial u}{\partial y}\right) - \frac{2}{3}\frac{\partial}{\partial y}\left[\Gamma\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)\right]\right\}}_{\text{Source term } (S^v)} \tag{11}$$

The integration of (11) is accomplished for $v$ over a control volume (Fig. 1), as given in [25]: integrating from $x^f_i$ to $x^f_{i+1}$ for OX, from $y^v_{j-1}$ to $y^f_j$ for OY and integrating over the volume from $x^f_i$ to $x^f_{i+1}$ for OX and from $y^f_j$ to $y^v_j$ for OY. In the middle nodes, where $v$ is not defined, we have used a upwind interpolation scheme, instead the approach proposed in [25]. The coefficients are defined after adding the expressions obtained from both integrations and a corresponding rearrangement. The numerical equation for $v$ reads:

$$v_{i,j} = \frac{a^v_1 v_{i-1,j} + a^v_2 v_{i+1,j} + a^v_3 v_{i,j-1} + a^v_4 v_{i,j+1} + b^v}{a^v_0} - d^v_{i,j}(p_{i,j} - p_{i,j-1}), \tag{12}$$

where the coefficients are:

$$\begin{aligned}
&a^v_0 = a^v_1 + a^v_2 + a^v_3 + a^v_4 + 0.5\left(F^x_{i+1,j} - F^x_{i,j} + F^x_{i+1,j-1} - F^x_{i,j-1}\right) + \overline{F}^y_{i,j} - \overline{F}^y_{i,j-1} - S^v_p + (\rho_{i,j}\Delta y_j + \rho_{i,j-1}\Delta y_{j-1})\frac{\Delta x_i}{2\Delta t}\\
&a^v_1 = a^{vc}_1 + D^{vx}_{i,j}, \quad a^v_2 = a^{vc}_2 + D^{vx}_{i+1,j}, \quad a^v_3 = a^{vc}_3 + D^{vy}_{i,j}, \quad a^v_4 = a^{vc}_4 + D^{vy}_{i,j+1}\\
&b^v = \left(\rho^{n-1}_{i,j}\Delta y_j + \rho^{n-1}_{i,j-1}\Delta y_{j-1}\right)\frac{\Delta x_i}{2\Delta t} v^{n-1}_{i,j} + S^v_c, \quad d^v_{i,j} = \frac{A}{a^v_0}\Delta x_i
\end{aligned} \tag{13}$$

$$\begin{aligned}
&a^{vc}_0 = a^{vc}_1 + a^{vc}_2 + a^{vc}_3 + a^{vc}_4 + 0.5\left(F^x_{i+1,j} - F^x_{i,j} + F^x_{i+1,j-1} - F^x_{i,j-1}\right) + \overline{F}^y_{i,j} - \overline{F}^y_{i,j-1}\\
&a^{vc}_1 = 0.5\left[max\left(0, F^x_{i,j}\right) + max\left(0, F^x_{i,j-1}\right)\right]\\
&a^{vc}_2 = 0.5\left[max\left(0, -F^x_{i+1,j}\right) + max\left(0, -F^x_{i+1,j-1}\right)\right]\\
&a^{vc}_3 = max\left(0, \overline{F}^y_{i,j-1}\right), \quad a^{vc}_4 = max\left(0, -\overline{F}^y_{i,j}\right)
\end{aligned} \tag{14}$$

The term $\overline{F}^y_{i,j} = 0.5\left(F^y_{i,j} + F^y_{i,j+1}\right)$ is defined at point $\left(x^v_i, y^v_j\right)$. With $D$ we denote the diffusion conductance at cell face. To determine the value of $D$, we assume that the diffusion $\Gamma$ varies continuously between the adjacent control volumes and use the bilinear interpolation of the diffusion coefficients at the control volume surfaces to solve $\Gamma|_{x^f_i}$ in $D^{vx}_{i,j}$. For $D^{vy}_{i,j}$ no interpolation is needed, because the diffusion coefficient $\Gamma$ is defined in nodes $\left(x^v_i, y^v_j\right)$

$$D^{vx}_{i,j} = B\cdot\Gamma|_{x^f_i}\frac{\Delta y_j + \Delta y_{j-1}}{\Delta x_i + \Delta x_{i-1}}, \quad D^{vy}_{i,j} = B\cdot\Gamma_{i,j-1}\frac{\Delta x_i}{\Delta y_{j-1}} \tag{15}$$

The finite-difference representation of the source term $S^v$ (11) is expressed as $S^v = S^v_c + S^v_p v_{i,j}$. The coefficient $S^v_p$ is always less than or equal to zero; otherwise we could obtain an instability and divergence of the computational process, or physically unrealistic solutions [16]. After the integration over the control volume for $v_{i,j}$ and some additional rearrangements the source term takes the form:

$$\begin{aligned}
&S^v_p = -\frac{1}{3}\left(D^{vy}_{i,j} + D^{vy}_{i,j+1}\right)\\
&S^v_c = B\left[\frac{1}{3}\Gamma_{i,j}\frac{\Delta x_i}{\Delta y_j}v_{i,j+1} - \Gamma|_{x^f_i}(u_{i,j} - u_{i,j-1}) + \frac{1}{3}\Gamma_{i,j-1}\frac{\Delta x_i}{\Delta y_{j-1}}v_{i,j-1} + \frac{2}{3}\Gamma_{i,j-1}(u_{i+1,j-1} - u_{i,j-1}) + \Gamma|_{x^f_{i+1}}(u_{i+1,j} - u_{i+1,j-1})\right.\\
&\qquad\left.-\frac{2}{3}\Gamma_{i,j}(u_{i+1,j} - u_{i,j})\right] + g_y\left(\rho_{i,j}\frac{\Delta y_j}{2} + \rho_{i,j-1}\frac{\Delta y_{j-1}}{2}\right)\Delta x_i
\end{aligned} \tag{16}$$

One can see easily that $S^v_p$ is always less or equal to zero ($S^v_p = 0$, if $B = 0$). It is worth noting that when considering an extended hydrodynamic system of equations [4,8] high order terms such as thermal stresses etc. must enter in the source therm. In this case, it is important to transform the source term in a such way so that the condition (25) is fulfilled.

After integration over the control volume (from $x^f_i$ to $x^f_{i+1}$ for OX and from $y^f_j$ to $y^f_{j+1}$ for OY (Fig. 1)), the continuity equation (1) takes the form:

$$\frac{\partial\rho}{\partial t}\Delta x_i\Delta y_j + \left(\rho^u_{i+1,j}u_{i+1,j} - \rho^u_{i,j}u_{i,j}\right)\Delta y_j + \left(\rho^v_{i,j+1}v_{i,j+1} - \rho^v_{i,j}v_{i,j}\right)\Delta x_i = 0 \tag{17}$$

Here pseudo velocities are used in the same way to SIMPLER. Therefore, the numerical equations for $u$ and $v$ (12) can be written in form:

$$u_{i,j} = \hat{u}_{i,j} - d^u_{i,j}(p_{i,j} - p_{i-1,j}) \tag{18}$$

$$v_{i,j} = \hat{v}_{i,j} - d^v_{i,j}(p_{i,j} - p_{i,j-1}) \tag{19}$$

where $\hat{u}_{i,j}$ and $\hat{v}_{i,j}$ are pseudo velocities:

$$\hat{u}_{i,j} = \frac{a^u_1 u_{i-1,j} + a^u_2 u_{i+1,j} + a^u_3 u_{i,j-1} + a^u_4 u_{i,j+1} + b^u}{a^u_0} \tag{20}$$

$$\hat{v}_{i,j} = \frac{a^v_1 v_{i-1,j} + a^v_2 v_{i+1,j} + a^v_3 v_{i,j-1} + a^v_4 v_{i,j+1} + b^v}{a^v_0} \tag{21}$$

An important part of our scheme is obtained by the substitution of density with pressure in the unsteady term in Eq. (17) (a similar idea was used in PISO [19]):

$$\frac{\partial\rho}{\partial t}\Delta x_i\Delta y_j = \frac{\partial\left(\frac{p}{T}\right)}{\partial t}\Delta x_i\Delta y_j = \left(\frac{p_{i,j}}{T_{i,j}} - \frac{p^{n-1}_{i,j}}{T^{n-1}_{i,j}}\right)\frac{\Delta x_i\Delta y_j}{\Delta t} \tag{22}$$

Finally, after some rearrangements, the numerical equation for pressure is expressed as follows:

$$a^p_0 p_{i,j} = \left(a^{px}_{i,j}p_{i-1,j} + a^{px}_{i+1,j}p_{i+1,j} + a^{py}_{i,j}p_{i,j-1} + a^{py}_{i,j+1}p_{i,j+1}\right)\Delta t + b^p \tag{23}$$

where:

$$\begin{aligned}
a^p_0 &= \frac{1}{T_{i,j}}\Delta x_i\Delta y_j + \left(a^{px}_{i,j} + a^{px}_{i+1,j} + a^{py}_{i,j} + a^{py}_{i,j+1}\right)\Delta t\\
b^p &= \frac{p^{n-1}_{i,j}}{T^{n-1}_{i,j}}\Delta x_i\Delta y_j - \left(b^{px}_{i+1,j} - b^{px}_{i,j} + b^{py}_{i,j+1} - b^{py}_{i,j}\right)\Delta t\\
a^{px}_{i,j} &= \rho^u_{i,j}d^u_{i,j}\Delta y_j, \quad a^{py}_{i,j} = \rho^v_{i,j}d^v_{i,j}\Delta x_i\\
b^{px}_{i,j} &= \rho^u_{i,j}\hat{u}_{i,j}\Delta y_j, \quad b^{py}_{i,j} = \rho^v_{i,j}\hat{v}_{i,j}\Delta x_i
\end{aligned} \tag{24}$$

We multiply both sides of (23) by time step, making the numerical scheme more stable for small time steps. Scarborough [26] has shown that a sufficient condition for the convergence of an iterative method can be expressed in terms of the values of the coefficients of the numerical equations as follows:

$$\frac{\left|a^{px}_{i,j}\Delta t\right| + \left|a^{px}_{i+1,j}\Delta t\right| + \left|a^{py}_{i,j}\Delta t\right| + \left|a^{py}_{i,j+1}\Delta t\right|}{\left|a^p_0\right|}\begin{cases}\leqslant 1 & \text{at all nodes}\\ 1 & \text{at one node at least}\end{cases} \tag{25}$$

For SIMPLE-TS the validity of condition (25) can be proved easily. The coefficients (24) are positive, therefore, after the substitution of (24) in (25) the condition (25) takes the form:

$$\frac{\left(a^{px}_{i,j} + a^{px}_{i+1,j} + a^{py}_{i,j} + a^{py}_{i,j+1}\right)\Delta t}{\frac{1}{T_{i,j}}\Delta x_i\Delta y_j + \left(a^{px}_{i,j} + a^{px}_{i+1,j} + a^{py}_{i,j} + a^{py}_{i,j+1}\right)\Delta t} < 1 \tag{26}$$

A similar substitution done for methods containing term $\rho - \rho^{n-1}$ (like SIMPLE) in (25) gives the expression

$$\frac{\left(a^{px}_{i,j} + a^{px}_{i+1,j} + a^{py}_{i,j} + a^{py}_{i,j+1}\right)\Delta t}{\left(a^{px}_{i,j} + a^{px}_{i+1,j} + a^{py}_{i,j} + a^{py}_{i,j+1}\right)\Delta t} = 1 \tag{27}$$

and to make the iterative process converging one has to use under-relaxation coefficients appropriately included in the iteration procedure.

In the new algorithm the temperature is calculated from the energy equation (4). It is worth noting that by using continuity equation the term $Dp/Dt$ in the right-hand side in the equation is transformed into new one, not having a time derivative of pressure:

$$\underbrace{\frac{\partial(\rho T)}{\partial t}}_{\text{Unsteady term}} + \underbrace{\frac{\partial(\rho uT)}{\partial x} + \frac{\partial(\rho vT)}{\partial y}}_{\text{Convective terms}} = \underbrace{C^{T1}_f\left[\frac{\partial}{\partial x}\left(\Gamma^\lambda\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(\Gamma^\lambda\frac{\partial T}{\partial y}\right)\right]}_{\text{Diffusion terms}} + \underbrace{C^{T2}_f\cdot\Gamma\cdot\Phi + C^{T3}_f\cdot p\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)}_{\text{Source term }(S^T)} \tag{28}$$

where:

$$C_f^{T1} = \frac{C^{T1}}{1 - C^{T3}}, \quad C_f^{T2} = \frac{C^{T2}}{1 - C^{T3}}, \quad C_f^{T3} = \frac{C^{T3}}{C^{T3} - 1} \tag{29}$$

The discrete equation for temperature is obtained by integration of Eq. (28) over a control volume for the field variables (Fig. 1):

$$a_0^T T_{i,j} = \Delta t\left(a_1^T T_{i-1,j} + a_2^T T_{i+1,j} + a_3^T T_{i,j-1} + a_4^T T_{i,j+1} + S^T_{c_{i,j}}\right) + \rho^{n-1}_{i,j} T^{n-1}_{i,j} \Delta x_i \Delta y_j \tag{30}$$

where:

$$\begin{aligned} a_0^T &= \Delta t\left(a_1^T + a_2^T + a_3^T + a_4^T + F^x_{i+1,j} - F^x_{i,j} + F^y_{i,j+1} - F^y_{i,j}\right) + \rho_{i,j}\Delta x_i \Delta y_j \\ a_1^T &= max\left(0, F^x_{i,j}\right) + D^{Tx}_{i,j}, \quad a_3^T = max\left(0, F^y_{i,j}\right) + D^{Ty}_{i,j} \\ a_2^T &= max\left(0, -F^x_{i+1,j}\right) + D^{Tx}_{i+1,j}, \quad a_4^T = max\left(0, -F^y_{i,j+1}\right) + D^{Ty}_{i,j+1} \end{aligned} \tag{31}$$

The diffusion coefficients are:

$$D^{Tx}_{i,j} = C_f^{T1} \cdot \Gamma^\lambda|_{x_i^f} \frac{\Delta y_j}{0.5(\Delta x_i + \Delta x_{i-1})}, \quad D^{Ty}_{i,j} = C_f^{T1} \cdot \Gamma^\lambda|_{y_j^f} \frac{\Delta x_i}{0.5(\Delta y_j + \Delta y_{j-1})} \tag{32}$$

A harmonic average between two neighboring nodes is used to calculate $\Gamma^\lambda|_{x_i^f}$ and $\Gamma^\lambda|_{y_j^f}$:

$$\Gamma^\lambda|_{x_i^f} = \frac{(\Delta x_{i-1} + \Delta x_i)\Gamma^\lambda_{i-1,j}\Gamma^\lambda_{i,j}}{\Delta x_{i-1}\Gamma^\lambda_{i,j} + \Delta x_i \Gamma^\lambda_{i-1,j}}, \quad \Gamma^\lambda|_{y_j^f} = \frac{(\Delta y_{j-1} + \Delta y_j)\Gamma^\lambda_{i,j-1}\Gamma^\lambda_{i,j}}{\Delta y_{j-1}\Gamma^\lambda_{i,j} + \Delta y_j \Gamma^\lambda_{i,j-1}} \tag{33}$$

The finite-difference representation of the source term $S^T$ is expressed as:

$$S^T = S_c^T + S_p^T T_{i,j}, \tag{34}$$

where:

$$\begin{aligned} S_p^T &= 0 \\ S^T_{c_{i,j}} &= C_f^{T2} \cdot \Gamma_{i,j}\left\{2\left[\left(\frac{u_{i+1,j} - u_{i,j}}{\Delta x_i}\right)^2 + \left(\frac{v_{i,j+1} - v_{i,j}}{\Delta y_j}\right)^2\right] + \left(\frac{v(x^f_{i+1}, y^v_j) - v(x^f_i, y^v_j)}{\Delta x_i} + \frac{u(x^v_i, y^f_{j+1}) - u(x^v_i, y^f_j)}{\Delta y_j}\right)^2\right. \\ &\left. - \frac{2}{3}\left(\frac{u_{i+1,j} - u_{i,j}}{\Delta x_i} + \frac{v_{i,j+1} - v_{i,j}}{\Delta y_j}\right)^2\right\}\Delta x_i \Delta y_j + C_f^{T3} p_{i,j}\left(\frac{u_{i+1,j} - u_{i,j}}{\Delta x_i} + \frac{v_{i,j+1} - v_{i,j}}{\Delta y_j}\right)\Delta x_i \Delta y_j \end{aligned} \tag{35}$$

To interpolate velocities $u\left(x^v_i, y^f_{j+1}\right)$, $u\left(x^v_i, y^f_j\right)$, $v\left(x^f_{i+1}, y^v_j\right)$ and $v\left(x^f_i, y^v_j\right)$ a bilinear interpolation between four neighboring nodes is used for each one.

## 3. Algorithm SIMPLE-TS

Using the results of the previous section we are ready to construct the general algorithm of the new method. The SIMPLE-TS algorithm for solving unsteady, compressible, viscous and heat-conducting gas flows consists of the following steps:

**The SIMPLE-TS algorithm**

```
Step 1. Initialize $u$, $v$, $p$, $T$ and compute $\rho$ using Eq. (5). Set time step $\Delta t$.
Start loop 1:
  Step 2. Set the initial condition for the calculated time step: $t := t + \Delta t$, $u^{n-1} = u$, $v^{n-1} = v$, $p^{n-1} = p$, $T^{n-1} = T$, $\rho^{n-1} = \rho$.
  Start loop 2 (calculating a state for a new time step):
    Step 3. Calculate convective and diffusion fluxes: $F^x$ (7), $F^y$ (8), $D^{ux}$, $D^{uy}$, $D^{vx}$ (15), $D^{vy}$ (15), $D^{Tx}$ (32), $D^{Ty}$ (32).
    Step 4. Calculate pseudo velocities $\hat{u}$ (20), $\hat{v}$ (21), coefficients $d^u$ and $d^v$ (13), and the coefficients for pressure equation $a^{px}$, $a^{py}$, $b^{px}$, $b^{py}$ (24).
    Start loop 3:
      Step 5. Solve the coupled equations for energy (30) and pressure (23).
    Stop loop 3. In most cases two iterations are sufficient. Further in the paper this point will be discussed in detail.
    Step 6. Compute $\rho$ (5), using $p$ and $T$ calculated within step 5, as well as velocities: $u$ (18) and $v$ (19).
  Convergence of loop 2: Check for convergence of the iteration process for the current time step by using iteration criteria (36). If the iteration criteria are not satisfied go to step 3, otherwise continue.
Convergence of loop 1: If the final time is not reached go to step 2, otherwise stop the calculation.
```

The following iteration criteria must be satisfied:

$$\begin{aligned}&|p-p^{old}|_{max}<\epsilon^p, \quad |T-T^{old}|_{max}<\epsilon^T\\ &|u-u^{old}|_{max}<\epsilon^u, \quad |v-v^{old}|_{max}<\epsilon^v\end{aligned} \tag{36}$$

where $p$, $T$, $u$ and $v$ are values obtained from the current iteration, $p^{old}$, $T^{old}$, $u^{old}$ and $v^{old}$ are values obtained from the previous iteration, $\epsilon^p$, $\epsilon^T$, $\epsilon^u$ and $\epsilon^v$ are iteration criteria for $p$, $T$, $u$ and $v$, respectively. When under-relaxation coefficients are included in the iteration scheme the convergence criteria take the following form:

$$\begin{aligned}&\frac{|p-p^{old}|_{max}}{\alpha_p}<\epsilon^p, \quad \frac{|T-T^{old}|_{max}}{\alpha_T}<\epsilon^T\\ &\frac{|u-u^{old}|_{max}}{\alpha_u}<\epsilon^u, \quad \frac{|v-v^{old}|_{max}}{\alpha_v}<\epsilon^v\end{aligned} \tag{37}$$

where $\alpha_p$, $\alpha_T$, $\alpha_u$ and $\alpha_v$ are under-relaxation coefficients for $p$, $T$, $u$ and $v$, respectively.

A basic idea in the algorithm SIMPLE-TS is to keep the equation of state maximally satisfied within the entire iterative process. This idea is accomplished by the calculation of pressure and temperature in loop 3 and the substitution of density with pressure into the unsteady term using equation of state. The loop 3 reduces the computational time for reaching a given accuracy (see Table 2). A second benefit from loop 3 is that the mass in the entire computational volume is conserved within given limits before finishing loop 2, since loop 3 is run until satisfying criteria $\epsilon^p$ and $\epsilon^T$.

The velocities, pressure and temperature fields are calculated at the end of the loop 2 using the values of the coefficients and pseudo velocities (step 3 and step 4) which depend directly on the initial fields, set in the beginning of loop 2.

The interpolation, applied to the middle points as explained in the previous section, is important for a successful implementation of the new algorithm. Having in mind this the following changes have to be accomplished in order to adapt the SIMPLE code to the SIMPLE-TS one:

1. The substitution of density into the unsteady term of the pressure equation using the equation of state is the most important step. In this way the coefficients for the pressure equations, used in SIMPLE-TS, satisfy a sufficient criterion for the convergence of the iterative process (26).
2. The calculation of energy and pressure equations in one loop (loop 3 of SIMPLE-TS), after calculation of pseudo velocities and before calculation of velocities.
3. The calculation of pseudo velocities.
4. The calculation of the absolute pressure instead of the pressure-correction. To stress on the importance of the first issue let us consider a simplified calculation of one-dimensional unsteady isothermal pressure driven flow in a duct.The example is simple but useful to highlight the meaning of the replace of density with pressure using the equation of state (22). The problem is calculated by using Eq. (17) with parameters $A = 1$ and $D^{ux} = 1$. The inlet and outlet pressures are fixed at $(p_{in} = 3)$ and $(p_{out} = 1)$, at both inlet and outlet the gradient of the velocity is zero $(\partial u/\partial x = 0)$. Initially the fluid is at rest under pressure equal to the outlet pressure. The step on OX is $\Delta x = 0.1$, the time step is $\Delta t = 0.1$ and the length of the duct is 10. The calculation follows the steps of the SIMPLE-TS algorithm. It can be shown easily that in the case without substitution of density (in the pressure equation exist term $\rho - \rho^{n-1}$) the exact pressure profile is linear demonstrating a typical incompressible behavior of the fluid while in the case of substitution (22) the solution takes into account the compressibility of the fluid. The pressure profiles shown on Fig. 2 are reached after the first iteration of the loop 2. In the case without substitution of density with pressure the equation of state is not satisfied within the iteration and a propagation

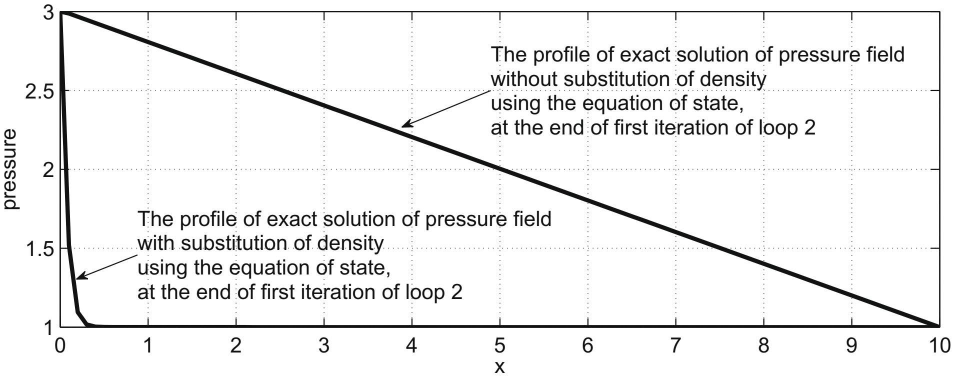


**Fig. 2.** Pressure profiles of the exact solution of the pressure equation calculated in loop 3, obtained with and without substitution of the density into unsteady term of the pressure equation.

of a disturbance from inlet to outlet is generated in each iteration making difficult the convergence of the iterative process. To ensure convergence of the algorithm in the case without substitution of density one must use under-relaxation coefficients and one iteration of the loop 3. The pressure profile of the exact solution of the pressure equation, when density is substituted, has a form of a compression wave (Fig. 2). Already after the first iteration the pressure profile is close to the final solution which is sought for within the given time step. Thus, the algorithm SIMPLE-TS maximally satisfies the equation of state during the iterative process and this leads to a better approximation of the solution after each iteration. In this way the number of iterations, and correspondingly the computational time, is reduced.

## 4. Applications

The system of equations (1)–(5) is given in a general form of the Navier–Stokes–Fourier equations. For gaseous microflow description we use the model of a compressible, viscous hard-sphere gas with diffusion coefficients determined by the first approximation of the Chapman–Enskog theory for low Knudsen numbers [10]. For a hard-sphere gas, the viscosity coefficient $\mu$ and the heat conduction coefficient $\lambda$ read (first approximations are sufficient for our considerations) as:

$$\mu = \mu_h\sqrt{T}, \quad \mu_h = (5/16)\rho_0 l_0 V_{th}\sqrt{\pi} \tag{38}$$

$$\lambda = \lambda_h\sqrt{T}, \quad \lambda_h = (15/32)c_p\rho_0 l_0 V_{th}\sqrt{\pi} \tag{39}$$

The dimensionless system of equations (1)–(5) is scaled by the following reference quantities, as given in [10]: molecular thermal velocity $V_0 = V_{th} = \sqrt{2RT_0}$ for velocity, a flow characteristic length $L$ (defined for each problem) for length, $t_0 = L/V_0$ for time, the reference pressure and temperature definition depends on the considered problem. The corresponding non-dimensional parameters in the equation system (1)–(5) read as follows:

$$\begin{gathered} A = 0.5, \quad B = \frac{5\sqrt{\pi}}{16}Kn, \quad \Gamma = \Gamma^{\lambda} = \sqrt{T} \\ C^{T1} = Kn\sqrt{\pi\frac{225}{1024}}, \quad C^{T2} = \frac{\sqrt{\pi}}{4}Kn, \quad C^{T3} = \frac{2}{5} \end{gathered} \tag{40}$$

For the Rayleigh–Bénard flow a gravity acceleration $g$ is used in the opposite direction of axis OY, so that the Froude number is equal to $Fr = V_0^2/(g \cdot L)$. Velocity-slip and temperature-jump boundary conditions (BC) [27] are implemented in the algorithm following the standard approach [28].

The velocity-slip BC is given as:

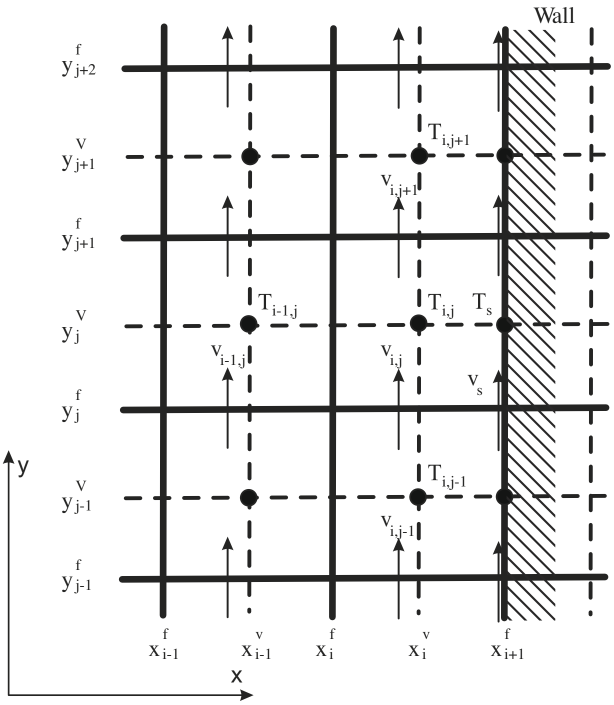


**Fig. 3.** Cell volume of $v$ and $T$ to the wall.

$$v_s - v_w = \zeta \frac{\partial v}{\partial n}\bigg|_s \tag{41}$$

where $v_s$ is velocity of the gas at the solid wall surface, $v_w$ is velocity of the wall, $\zeta = 1.1466 \cdot Kn_{local} = 1.1466 \cdot Kn/\rho_{local}$, $Kn_{local}$ is the local Knudsen number, $\rho_{local}$ is the local density, $\frac{\partial v}{\partial n}|_s$ is the derivative of velocity normal to the wall surface. When the velocity-slip BC is applied to a vertical wall (Fig. 3) $v_s$ takes coordinates $\left(x^f_{i+1}, y^f_j\right)$. The implementation of velocity-slip BC in the computational scheme is arranged in accordance with the finite volume method requirements:

$$v_s = \left(a^v_{BC} \cdot v_{i,j} + v_w\right)/\left(a^v_{BC} + 1\right) \tag{42}$$

where $a^v_{BC} = \zeta/(0.5 \cdot \Delta x_i) = 1.1466 \cdot Kn/\left(\rho^v_{i,j} \cdot 0.5 \cdot \Delta x_i\right)$. The temperature-jump boundary condition is applied in a similar way:

$$T_s - T_w = \tau \frac{\partial T}{\partial n}\bigg|_s \tag{43}$$

where $T_s$ is temperature of the gas at the wall surface, $T_w$ is temperature of the wall, $\tau = 2.1904 \cdot Kn_{local} = 2.1904 \cdot Kn/\rho_{local}$, $\frac{\partial T}{\partial n}|_s$ is the derivative of temperature normal to the wall surface. Applied on the vertical wall (Fig. 3) $T_s$ takes coordinates $\left(x^f_{i+1}, y^v_i\right)$ and the temperature-jump BC reads as follows:

$$T_s = \left(a^T_{BC} \cdot T_{i,j} + T_w\right)/\left(a^T_{BC} + 1\right) \tag{44}$$

where $a^T_{BC} = \tau/(0.5 \cdot \Delta x_i) = 2.1904 \cdot Kn/(\rho_{i,j} \cdot 0.5 \cdot \Delta x_i)$.

The velocity-slip and temperature-jump boundary conditions satisfy the sufficient condition for a convergent iterative method (25).

The examined problems are considered in a local Cartesian coordinate system on a uniform mesh with $N_x$ and $N_y$ numbers of cells on OX and OY directions.

Some of the calculations are performed by the authors by using also the DSMC method and the obtained results from both methods are compared. The DSMC algorithm used in the present calculations follows the basic steps of the "No Time Counter" scheme, proposed by Bird [9], and it is described in detail in [29]. It uses a hard-sphere model of mono-atomic gas. The diffuse reflection boundary condition is used at the microchannel and square walls.

### 4.1. Pressure driven gas flow in a microchannel

The flow geometry of microchannel is shown in Fig. 4. The inlet boundary conditions ($BC_{in}$) at $x = 0$ are:

$$p = p_{in}, \quad \frac{\partial v}{\partial x} = 0, \quad T = T_{in} \tag{45}$$

At the outlet $x = L$ we impose the following boundary conditions ($BC_{out}$):

$$p = p_{out}, \quad \frac{\partial v}{\partial x} = 0, \quad \frac{\partial T}{\partial x} = 0 \tag{46}$$

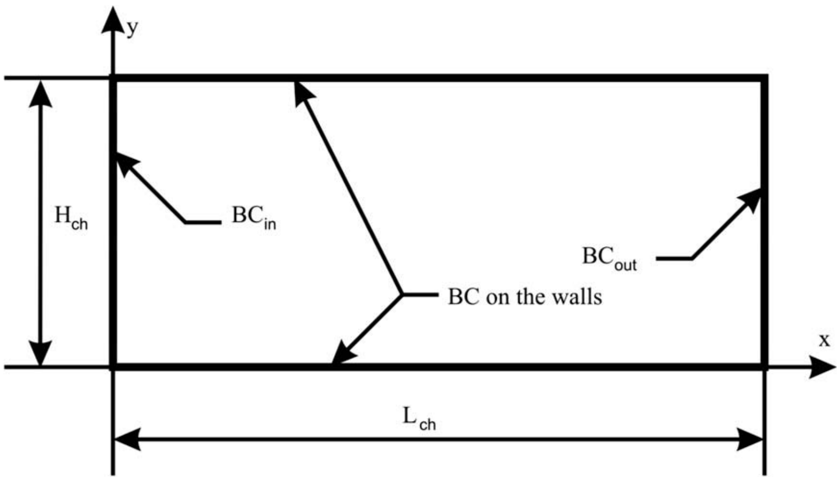


**Fig. 4.** Geometry of a microchannel with length $L_{ch}$ and height $H_{ch}$.

The boundary conditions for the velocity $u$ at both inlet and outlet are obtained by integration of the continuity equation (1) over each cell in the first or last column of the grid along the $x$-axis. In this way we ensure a better conservation of mass. Thus, velocity $u$ at the channel outlet is computed by using the expression:

$$u_{i+1,j} = -\left(\rho_{i,j} - \rho_{i,j}^{n-1}\right)\frac{\Delta x_i}{\Delta t}\frac{1}{\rho^u_{i+1,j}} - \left(\rho^v_{i,j+1} v_{i,j+1} - \rho^v_{i,j} v_{i,j}\right)\Delta x_i \frac{1}{\rho^u_{i+1,j}\Delta y_j} + \rho^v_{i,j} u_{i,j}\frac{1}{\rho^u_{i+1,j}} \tag{47}$$

where, $i = N_x$. Velocity $u$ at the inlet is computed in a similar manner.

The parameters specific for the considered problem that complete the problem formulation are: $p_0 = p_{out},\ T_0 = T_w,\ T_{in} = T_w$ and $L = H_{ch}$.

The result obtained from SIMPLE-TS calculations are compared to the available analytical solution of viscous, compressible isothermal flow in a long microchannel [22] and as well as to the DSMC data. The analytical solution (AS) [22], given for the pressure $p^{AS}$ (48) and the horizontal component of velocity $u^{AS}$ (49), can be rewritten in a non-dimensional form according to the given scales as follows:

$$p^{AS}(x) = p_{out}\left[-r + \sqrt{r^2 + (1 + 2\cdot r)\frac{x}{L_{ch}} + (P^2 + 2\cdot r\cdot P^2)\left(1 - \frac{x}{L_{ch}}\right)}\right] \tag{48}$$

$$u^{AS}(x,y) = \frac{1}{A\cdot 5\cdot Kn\sqrt{\pi}}\frac{dp^{AS}(x)}{dx}\left[1 - 4\left(\frac{y - 0.5}{H_{ch}}\right)^2 + \frac{4\cdot F\cdot Kn}{p^{AS}(x)}\right] \tag{49}$$

where $r = 6\cdot F\cdot Kn,\ P = p_{in}/p_{out}$ is the pressure ratio and $F = 1$ is the slip coefficient.

The analytical solutions $p^{AS}$ and $u^{AS}$ uses the well-known Maxwell velocity-slip BC. The same BC on the channel walls are used in the SIMPLE-TS calculations. For velocity-slip BC (41), the variable $\zeta$ is equal to $\zeta = 1\cdot Kn/\rho_{local}$. The temperature of the gas nearby the channel wall is equal to the wall temperature $T_w$.

A long enough channel covered with a uniform grid with steps $\Delta x = \Delta y = \Delta = 0.05$ is considered so that the total number of grid cells is equal to $(1000 \times 20)$. This problem is solved for $Kn = 0.05$ (defined at the channel outlet), $T_{in} = T_w = 1,\ p_{in} = 3,\ p_{out} = 1$, the aspect ratio is $A = L_{ch}/H_{ch} = 50$ and the pressure ratio is $P = 3$. The results are presented in Figs. 5–7.

As the pressure does not depend on $y$ the macroscopic variable profiles are shown only on the centreline of the channel $y = H_{ch}/2$, Fig. 5. The temperature profile is very close to constant calculated by both SIMPLE-TS and DSMC methods. The convergence criteria used for this steady-state problem are $\epsilon^p = \epsilon^T = \epsilon^u = \epsilon^v = 10^{-12}$. The mass flow rate is calculated at both inlet ($F_{in}$) and outlet ($F_{out}$) as follows:

$$F_{in} = \sum_{j=1\dots N_y-1} \rho_{0,j}\cdot u_{0,j}\cdot \Delta y_j, \quad F_{out} = \sum_{j=1\dots N_y-1} \rho_{N_x-1,j}\cdot u_{N_x-1,j}\cdot \Delta y_j \tag{50}$$

The calculated difference between both values is very small, $|(F_{in} - F_{out})/F_{in}| \approx 2.5\times 10^{-8}$, illustrating a very good conservation of mass in the results obtained by the SIMPLE-TS method. The graphics illustrate also the very good agreement between the analytical solution, SIMPLE-TS and DSMC data.

The DSMC calculations are performed on a grid with 800 × 80 cells with a total number of $5.67\times 10^6$ particles.

### 4.2. Unsteady pressure driven gas flow in microchannel

To compare the efficiency between SIMPLE-TS, SIMPLE and PISO [30] we calculate the non-stationary pressure driven gas flow in microchannel. The flow geometry is the same as the considered in the previous stationary case (Fig. 4). Correspond-

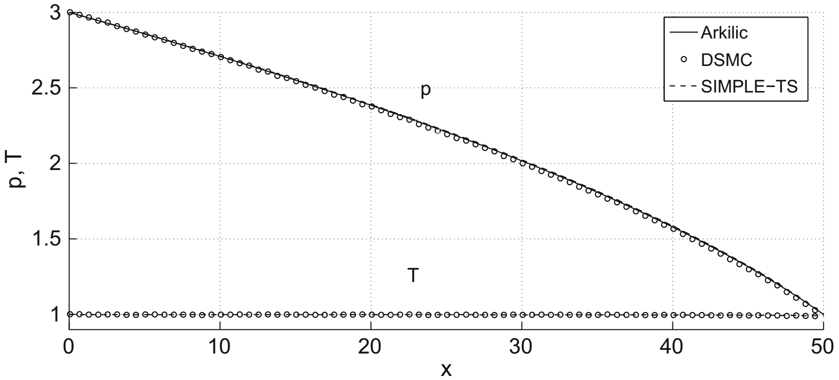


**Fig. 5.** Pressure and temperature distribution along OX at centre line of the channel ($y = H_{ch}/2$). Pressure and temperature profiles obtained from analytical solution [22], SIMPLE-TS and DSMC methods.

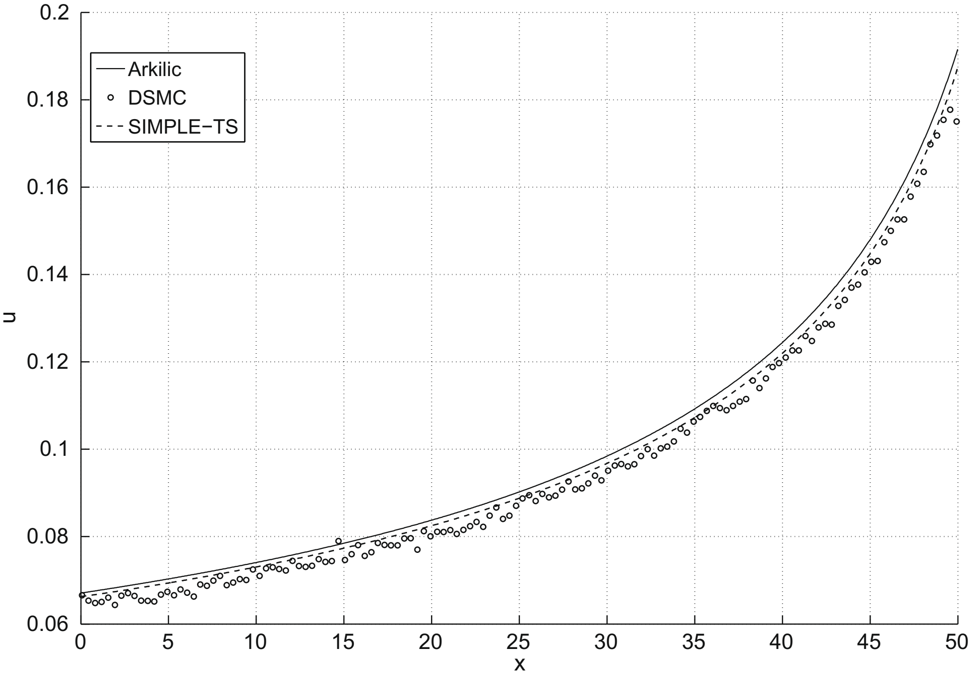


**Fig. 6.** Horizontal component of velocity ($u$) along the centre line of the channel ($y = H_{ch}/2$). $u$-velocity profiles, from analytical solution [22], SIMPLE-TS and DSMC methods.

ingly, at the channel walls we have velocity-slip (41) and temperature-jump (43) boundary conditions with coefficients (42) and (44). The initial condition is a gas at rest (the velocity fields are equal to zero), the pressure of the gas in the channel is equal to the pressure at the outlet ($p_{out}$), the gas temperature is equal to the temperature of the channel walls ($T_w$). At zero-time $t = 0$ the inlet is opened suddenly and the gas under pressure $p_{in}$ and temperature ($T_{in} = T_w$) enters the channel volume. We consider the initial part of the transient period and compare at several different times the compression wave propagation, calculated by all considered methods. A channel length of $L_{ch} = 2$ is sufficient for our aim. The consideration of a short channel allows us to carry out a set of runs sufficient for analysis of the influence of the time and spatial step variation. The results are obtained for five spatial steps $\Delta x = \Delta y = \Delta = 0.05$, 0.025, 0.01, 0.005 and 0.0025. The profiles of the macroscopic variables along and normal to the channel are shown in Figs. 8 and 9. As one can see the results convergence with the decrease of the spatial step so that the difference between the profiles for $\Delta = 0.01$ (200 × 100) and $\Delta = 0.0025$ (800 × 400) at time $t = 0.2$ is less than 2.5%. The chosen time step is $\Delta t = 10^{-4}$. For a fixed spatial step $\Delta = 0.01$ the difference in the fields obtained for $\Delta t = 10^{-4}$ and $\Delta t = 10^{-5}$ at time $t = 0.2$ is less than 0.11%. for the same case the CPU times of SIMPLE, PISO and SIMPLE-TS calculations are compared. All calculations are done on a PC Pentium D 2.8 GHz, with 2 GB RAM/667 MHz.

The Standard SIMPLE method is realized by using one pass of the pressure-correction equation. In order to take into account the corrections in the pressure field, the density is computed from the state equation after the velocity evaluation. Then the fluxes ($F^x$ and $F^y$) are calculated by using the last computed values of the density and velocities. Finally, temperature is computed from the energy equation (30) by using all final values of pressure, velocities, density and fluxes. After that the density is re-calculated with the new temperature value. Some runs with different under-relaxation coefficients have been performed in order to find an optimal under-relaxation coefficient $\alpha_p = 1.575 \times 10^{-5}$ for pressure. No under-relaxation coefficients for $u$, $v$ and $T$ are applied. The maximum residuals of the equations are $\epsilon^p = \epsilon^T = \epsilon^u = \epsilon^v = 10^{-9}$ (37). About 1776 s of CPU time is needed to reach the benchmark time $t = 0.2$ (40 iterations in mean per a time step).

The PISO method contains a prediction step, which takes approximately the same CPU time such as the correction step. Finally, it is worth noting that there is not any need to include under-relaxation coefficients in PISO [30] and SIMPLE-TS. The results, obtained from the calculations carried out by PISO and SIMPLE-TS, are summarized in Tables 1 and 2, where the maximum difference between the current residual and a fixed one $\epsilon = 10^{-9}$, obtained with unlimited number of iterations or correction steps for $u$, $v$, $p$ and $T$ fields, is given. The number of iterations, when unlimited iterations for SIMPLE-TS and unlimited correction steps for PISO are used, are rounded off to the closest integer value in order to avoid eventual confusions. As one can see from the results for $t = 0.2$ the maximum difference between the calculated by SIMPLE and SIMPLE-TS macroscopic fields ($u$, $v$, $p$ and $T$) is $0.929 \times 10^{-3}$. The maximum difference between the calculated by PISO and SIMPLE-TS fields ($u$, $v$, $p$ and $T$) at $t = 0.2$, is $1.002 \times 10^{-8}$, which is within the used accuracy. Tables 2 and 3 show that for the same CPU time a SIMPLE-TS run reaches a better accuracy or for a fixed accuracy it needs less CPU time than required for a correspond-

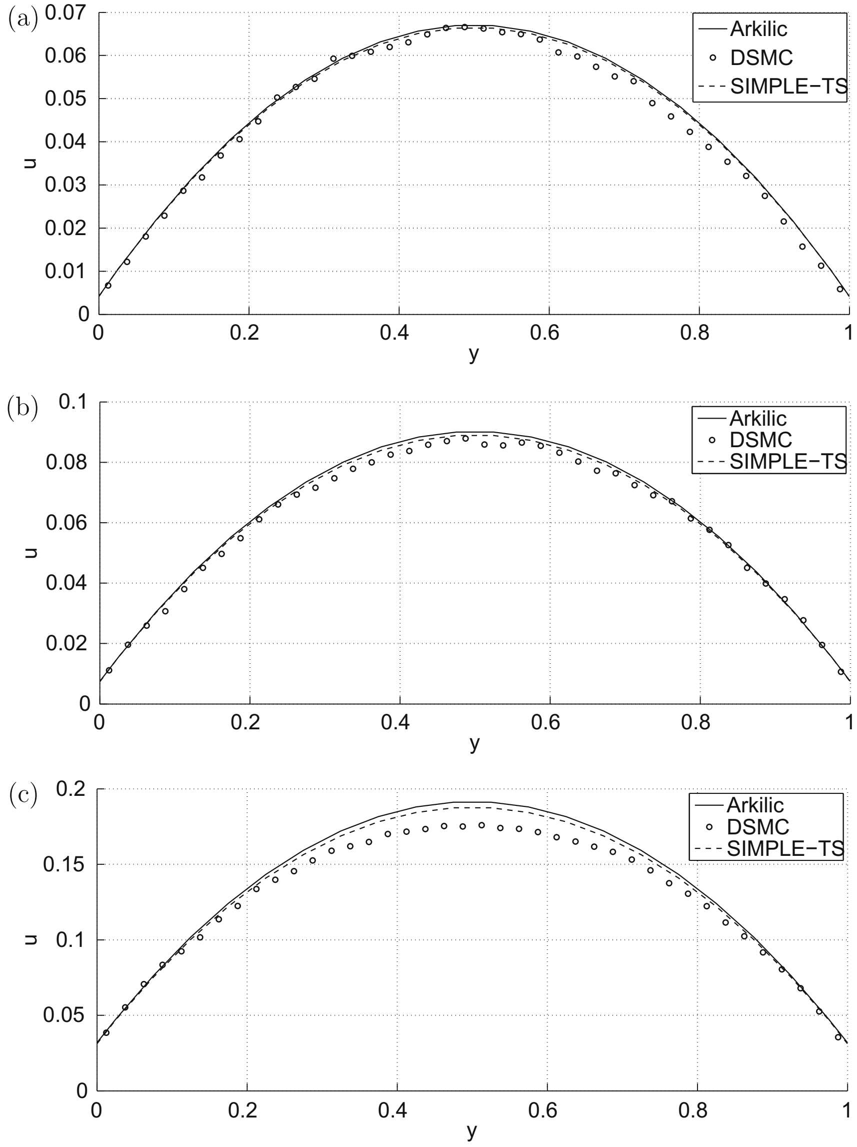


**Fig. 7.** Horizontal component of velocity ($u$) calculated from analytical solution [22], SIMPLE-TS and DSMC in several normal to the channel axis sections: (a) ($x = 0$), (b)($x = L_{ch}/2$), and (c) ($x = L_{ch}$).

ing PISO calculation. Generally speaking, the SIMPLE-TS calculations take much less CPU time, than the SIMPLE and less time than the PISO ones.

### *4.3. Flow past a confined square in microchannel*

As a second example we consider a 2D steady-state laminar flow around a small square particle with size $a$ confined in a plane microchannel (height $H_{ch}$) as shown in Fig. 10. The reference length is equal to the square size ($L = a$). The blockage ratio $B = a/H_{ch}$ is equal to $B = 10$, the inflow length is $L_a$. The problem is considered in a local Cartesian coordinate system, which is moving with the particle. Thus for an observer moving along with the particle the problem is transformed to a consideration of a gas flow past a stationary square confined in a microchannel with moving walls. Reference parameters for this problem are: $p_0 = p_{in},\ T_0 = T_{in}$ and $L = a$. Temperature of the square and channel walls are equal to $T_0$. On the walls of the channel and the square velocity-slip (41) and temperature-jump (43) boundary conditions are imposed. The Knudsen number is $Kn = 0.05$.

#### *4.3.1. Subsonic gas flow*

In the subsonic case the Mach number is 0.1, $L_{ch} = 40,\ L_a = 15.5$. The spatial steps are $\Delta x = \Delta y = \Delta = 0.025$ ($1600 \times 400$). The channel walls are moving with a constant velocity equal to $u = 0.09129$ (according to Mach number 0.1). The inflow

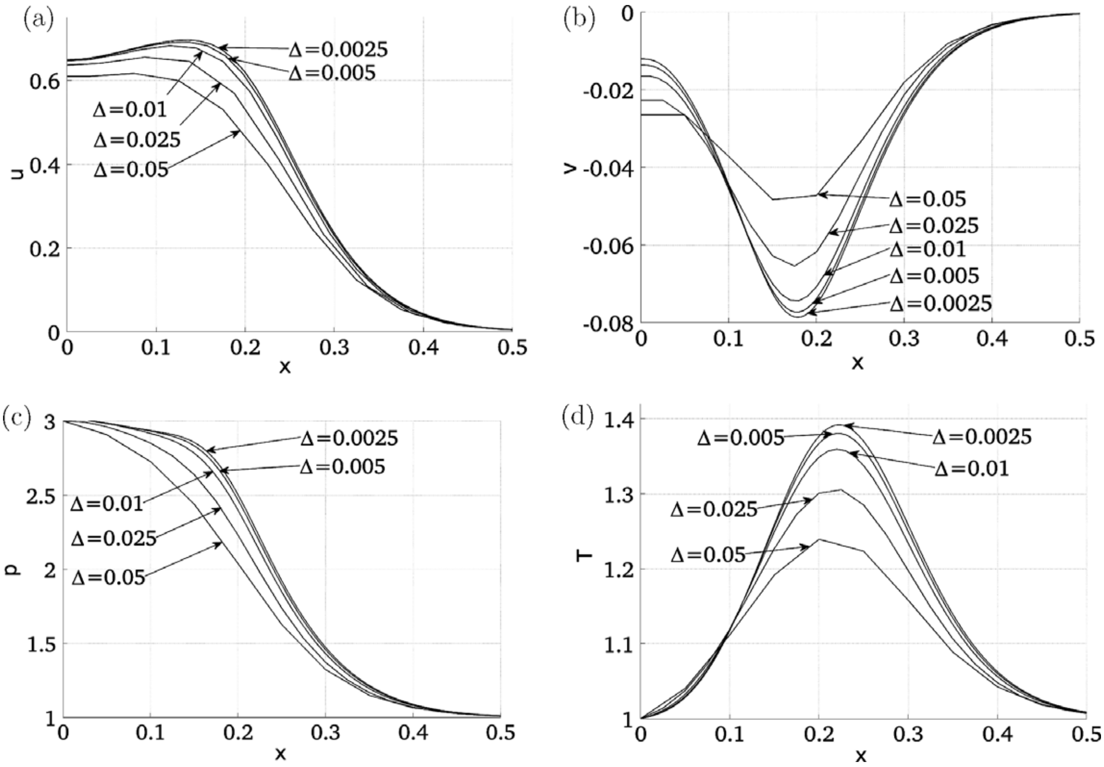


**Fig. 8.** The influence of the spatial step size on the calculated profiles along the channel axis at time $t = 0.2$. (a) $u$ at $(y = H_{ch}/2)$, (b) $v$ at $y = 0.05$, (c) $p$ at $y = H_{ch}/2$ and (d) $T$ at $y = H_{ch}/2$.

boundary conditions $(BC_{in})$ are: $u_{in}$ is calculated from the continuity equation for first control volume on OX, similarly to the pressure driven flow in a channel, $\partial v/\partial x = 0,\ p_{in} = 1$ and $T_{in} = 1$. The outflow boundary conditions $(BC_{out})$ are: $u_{out}$ is calculated using the continuity equation defined for the last control volume on OX (46), $\partial v/\partial x = 0,\ \partial p/\partial x = 0$ and $\partial T/\partial x = 0$. For these boundary conditions, the difference between inflow and outflow fluxes, calculated using (50), is $|(F_{out} - F_{in})/F_{in}| \approx 8.5 \times 10^{-10}$. The same problem is calculated also by using the DSMC method. The obtained by SIMPLE-TS drag coefficient $C_D^{SIMPLE\text{-}TS} = 7.493$ is in a good agreement with the drag coefficient $C_D^{DSMC} = 7.467$ calculated by the DSMC method. The DSMC calculations are performed on a grid with $1600 \times 400$ cells with a total number of particles $16.6 \times 10^6$. The drag coefficient is defined as $C_D = D/(0.5\rho_0' U^2 S)$ , where $\rho_0'$ is density, $D$ is the drag force [N], $U$ is velocity and $S$ is the projected area. From Figs. 11 and 12 one can see that the agreement between SIMPLE-TS and DSMC data is very good. It is important to note that for small Mach number to obtain DSMC data with a reduced variance one needs to carry out very long calculations compared to those of SIMPLE-TS.

### 4.3.2. Supersonic gas flow

In the supersonic calculations the Mach number is 2.4261. The channel length is $L_{ch} = 50$, the distance between square and the channel inlet is $L_a = 5.5$. The channel walls are moving with a constant velocity $u = 2.2147$ (according to Mach number 2.4261). The inflow boundary conditions $(BC_{in})$ are: $u_{in} = 2.2147,\ v = 0,\ p_{in} = 1$ and $T_{in} = 1$. The outflow boundary conditions $(BC_{out})$ are: $\partial u/\partial x = 0,\ \partial v/\partial x = 0,\ \partial p/\partial x = 0$ and $\partial T/\partial x = 0$.

The results for the drag coefficient of square are shown in Table 3 for different spatial steps $\Delta = \Delta x = \Delta y$ of the computational grid given in the first column of Table 3. The difference between inlet and outlet fluxes (50) is less than $|(F_{out} - F_{in})/F_{in}| \approx 10^{-8}$ for all calculated cases. The profiles of $u$, $v$, $p$ and $T$ computed for different spatial steps in the midplane of the channel are shown in Figs. 13 and 14. The differences between the solutions obtained by using different spatial steps are computed in percent (%) for the first local maximum of $u$ behind the square (see Fig. 13), are given below:

- difference between $\Delta = 0.05$ and $\Delta = 0.025$: $\frac{0.99562 - 0.69239}{u_{in}} \approx 13.7\%$
- difference between $\Delta = 0.025$ and $\Delta = 0.0125$: $\frac{0.69239 - 0.45352}{u_{in}} \approx 10.8\%$
- difference between $\Delta = 0.0125$ and $\Delta = 0.00625$: $\frac{0.45365 - 0.36468}{u_{in}} \approx 4.02\%$

One can see that the results converge with the grid refinement. We accept that the numerical solution is accurate enough for a spatial step $\Delta = 0.00625$ ($8000 \times 1600$).

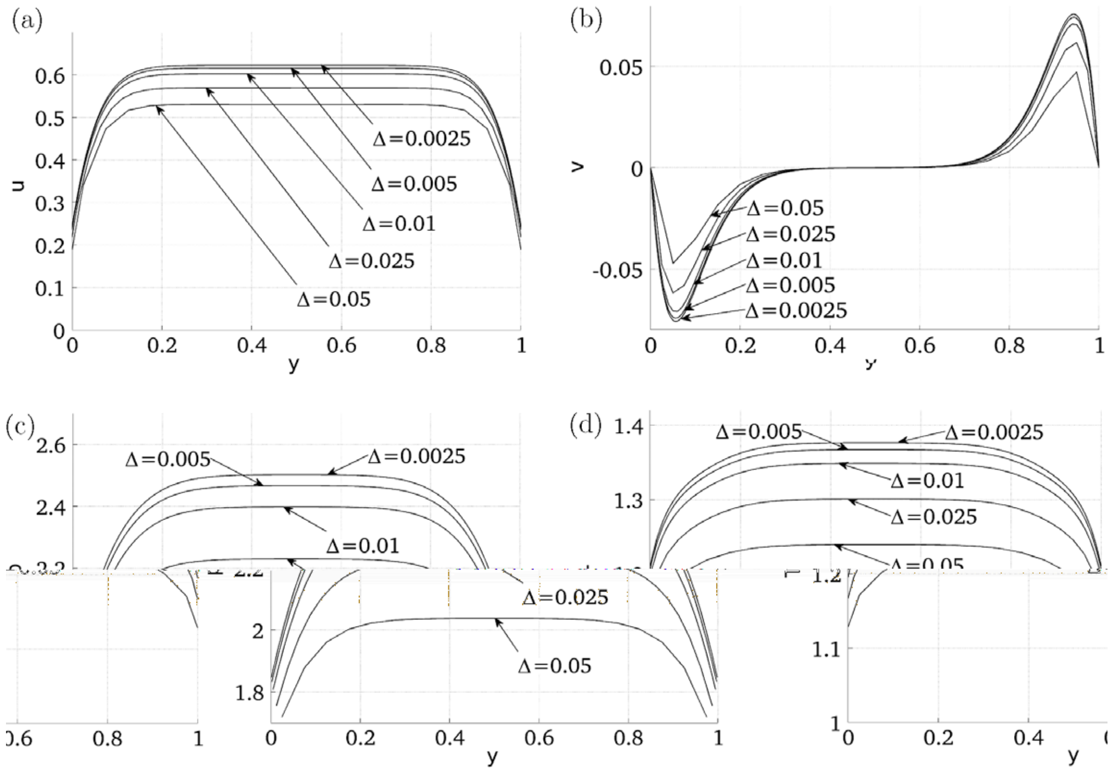


**Fig. 9.** The influence of the spatial step size on the calculated profiles for (a) $u$, (b) $v$, (c) $p$ and (d) $T$ at a section $x = 0.2$ (normal to the channel axis) at time $t = 0.2$.

**Table 1**
CPU times for PISO to reach $t = 0.2$.

| Number of correction steps | CPU time (s) | Maximum difference (%) |
|---|---|---|
| 2 | 123.25 | 6.28 |
| 3 | 168.75 | 1.01 |
| 4 | 213.50 | 0.130 |
| 5 | 259.75 | 0.0178 |
| 10 | 497.00 | 0 |

**Table 2**
CPU times for SIMPLE-TS to reach $t = 0.2$.

| Number of iterations | | CPU time (s) | Maximum difference (%) |
|---|---|---|---|
| Loop 2 | Loop 3 | | |
| 2 | 1 | 86.00 | 16.7 |
| 2 | 2 | 97.25 | 8.99 |
| 2 | 3 | 108.50 | 8.23 |
| 3 | 1 | 128.00 | 1.35 |
| 3 | 2 | 145.25 | 0.597 |
| 3 | 3 | 162.50 | 0.591 |
| 4 | 1 | 170.75 | 0.119 |
| 4 | 2 | 194.00 | 0.0463 |
| 4 | 3 | 217.00 | 0.0463 |
| 5 | 1 | 213.25 | 0.0109 |
| 5 | 2 | 242.00 | 0.00379 |
| 5 | 3 | 271.00 | 0.00379 |
| 10 | 1 | 424.75 | 0 |
| 7 | 2 | 339.25 | 0 |
| 7 | 3 | 384.25 | 0 |
| 7 | 4 | 409.25 | 0 |

**Table 3**
The drag coefficient of square for different meshes.

| $\Delta$ | Mesh | $C_D^{SIMPLE\text{-}TS}$ |
|---|---|---|
| 0.05 | 1000 × 200 | 1.875 |
| 0.025 | 2000 × 400 | 1.876 |
| 0.0125 | 4000 × 800 | 1.860 |
| 0.00625 | 8000 × 1600 | 1.848 |

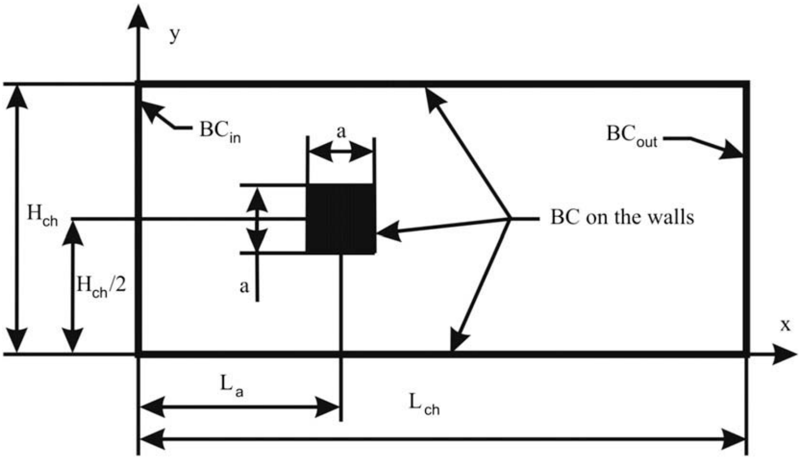


**Fig. 10.** Flow geometry for a square-shaped particle with size $a$ confined in a channel with length $L_{ch}$ and height $H_{ch}$.

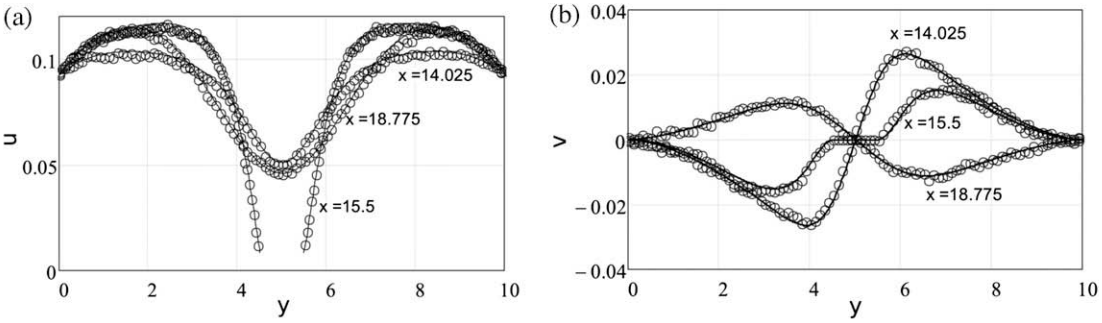


**Fig. 11.** Macroscopic profiles obtained by SIMPLE-TS (solid line) and DSMC (circles) data for: (a) horizontal velocity along the channel at $y = H_{ch}/2$ and (b) vertical velocity along the channel at $y = H_{ch}/4$.

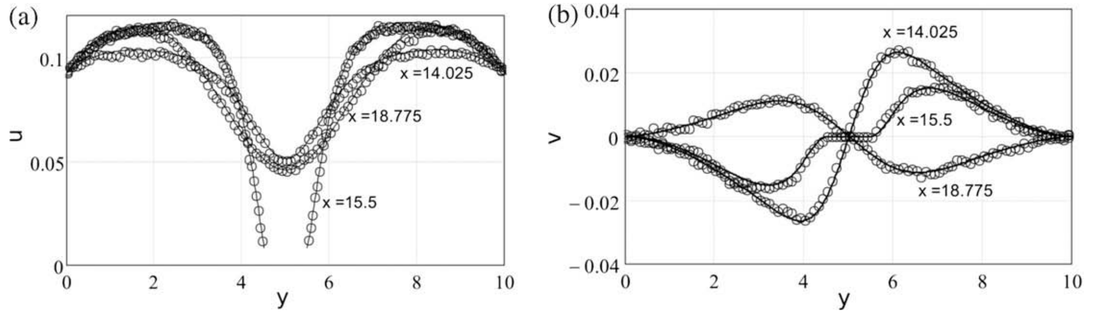


**Fig. 12.** Profiles of (a) the horizontal component of velocity and (b) the vertical component of velocity obtained by SIMPLE-TS (solid line) and DSMC (circles) data in sections normal to the channel axis in front of square ($x = 14.025$), in the middle of the square ($x = 15.5$), and behind the square ($x = 18.775$).

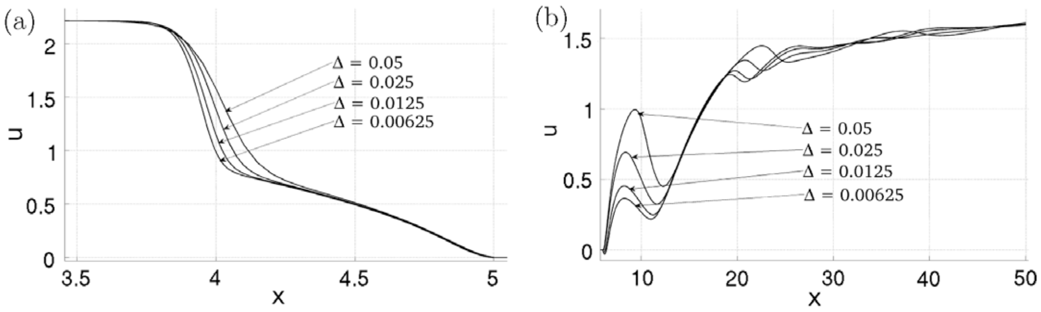


**Fig. 13.** Profiles of the horizontal velocity along the centre line of the channel ($y = H_{ch}/2$) for different spatial steps: (a) in front and (b) behind of the square.

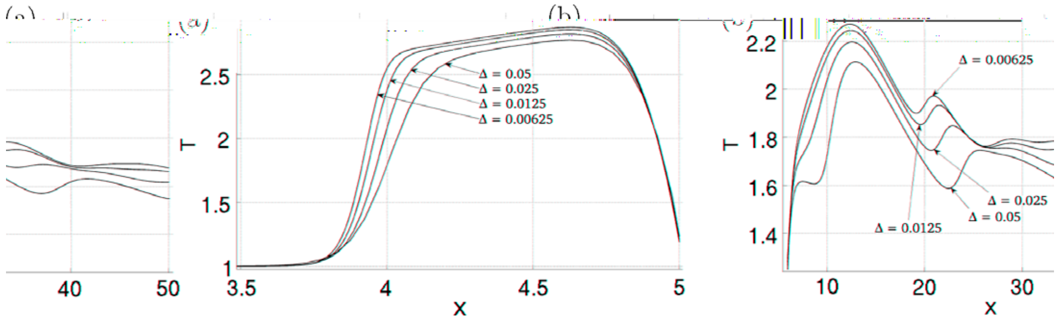


**Fig. 14.** Temperature profiles along the centre line of the channel ($y = H_{ch}/2$) for different spatial steps: (a) in front and (b) behind of the square.

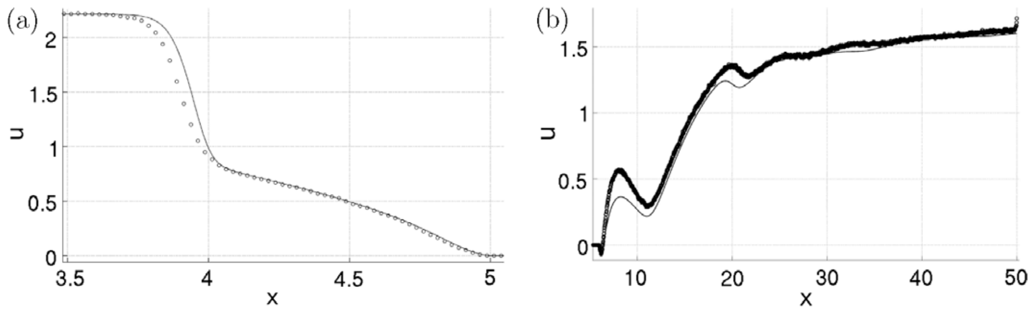


**Fig. 15.** Horizontal velocity profiles calculated by SIMPLE-TS (solid line) and DSMC (circles) in the mid-plane along the microchannel in front (a) and behind (b) of the square.

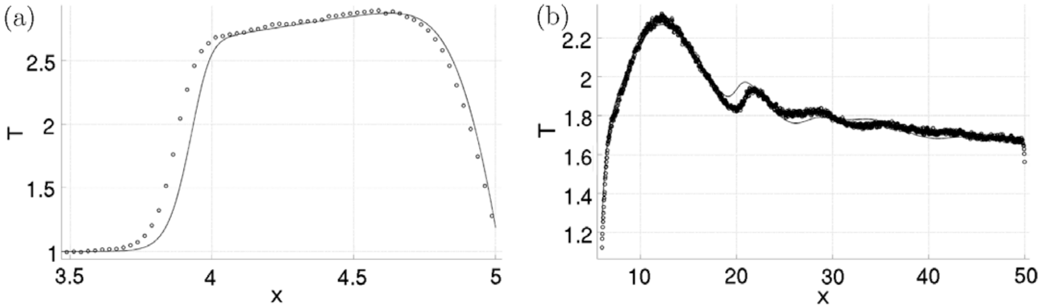


**Fig. 16.** Temperature profiles calculated by SIMPLE-TS (solid line) and DSMC (circles) in the mid-plane along the microchannel in front (a) and behind (b) of the square.

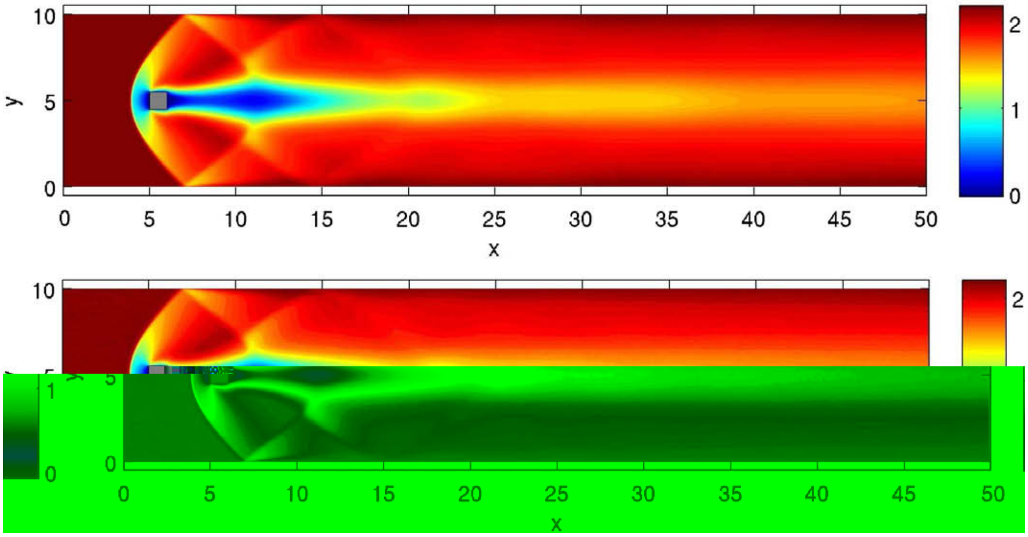


**Fig. 17.** Horizontal velocity field calculated by SIMPLE-TS (upper part) and DSMC (lower part).

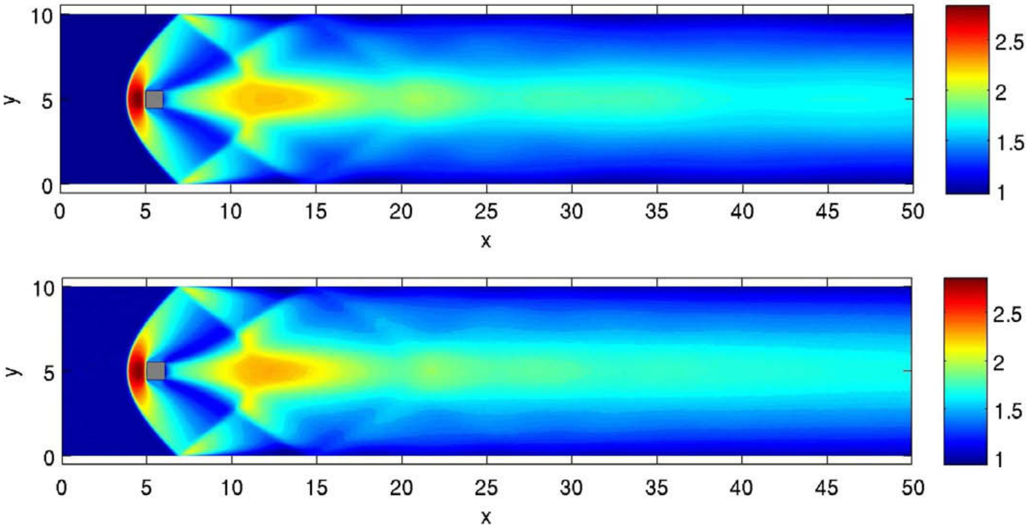


**Fig. 18.** Temperature field calculated by SIMPLE-TS (upper part) and DSMC (lower part).

The supersonic flow is calculated by using a parallel version of SIMPLE-TS code. In the parallel algorithm a decomposition of the computational domain is realized. The comparison of the results obtained by SIMPLE-TS (steps $\Delta = 0.00625$ $(8000 \times 1600)$ cells), and DSMC ($4000 \times 800$ cells and a total number of particles $96.1 \times 10^6$) in Figs. 15–18. The drag coefficient, calculated by DSMC, is $C_D^{DSMC} = 1.858$, which is in an excellent agreement with the value obtained by SIMPLE-TS (see Table 3).

The largest deviations of the SIMPLE-TS solution from the DSMC data are in the shock wave areas, where the field gradients are significant. From the graphics given for the horizontal component of velocity (Fig. 15) and the temperature (Fig. 16) one can detect a shock wave displacement between both solutions around of the square. A similar displacement between molecular and continuum results are pointed out in the works of other authors (for example, see [9]).

### *4.3.3. Rayleigh–Bénard flow of a rarefied gas*

The last example is linked to other group of problems that can be successfully analysed by SIMPLE-TS, namely, the problems of pattern formation in gaseous flows. The Rayleigh–Bénard convection of a rarefied gas [29,10] in continuum limit is considered for the case $Kn = 0.005,\ Fr = 50.0,\ T_h = 1$ and $T_c = 0.1$. Fig. 4 shows the geometry of the computational domain, where $T_h$ (the reference temperature) and $T_c$ are temperatures of the walls at $y = 0$ and $y = H_{ch}$, respectively. Character

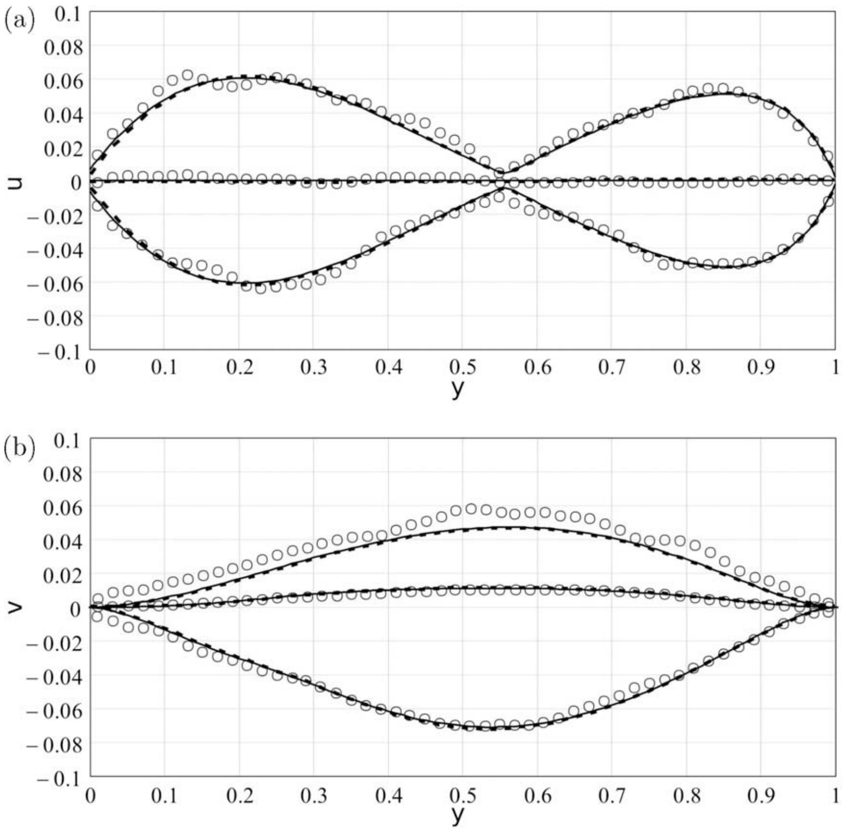


**Fig. 19.** The minimum, maximum and average profiles of (a) $u$-velocity and (b) $v$-velocity components computed by SIMPLE-TS (solid line), DSMC (circles) and finite difference (dash-line).

length is $L = H_{ch}$. The dimensions of the channel are $H_{ch} = 1$ and $L_{ch} = 2$. For this problem periodicity boundary conditions are used, instead $BC_{in}$ and $BC_{out}$. The boundary conditions at the walls are kept the same as defined in the previous examples. The initial condition is an equilibrium gas at rest $(u(x,y) = 0, v(x,y) = 0)$, with temperature and pressure equal to $(T(x,y) = 1, p(x,y) = 1)$. The grid steps are $\Delta x = \Delta y = \varDelta = 0.01$ $(200 \times 100)$. For more details of the problem formulation see [10]. The runs are performed by both SIMPLE-TS and DSMC methods. In SIMPLE-TS the iteration process (loop 2) within each time step is stopped, when convergence criteria (36) are fulfilled for $\epsilon = 10^{-10}$. The final state of the convection is a stable vortex flow. The conservation of mass in the whole computational domain is very good. For example, the averaged density is changed from 1 (at the beginning of calculation) to $1 + 7.4 \times 10^{-8}$ (at the end of the calculation). Fig. 19 illustrate the peaks (maximum and minimum) as well as the averaged velocity profiles of $u$ and $v$, which are obtained by applying oper-

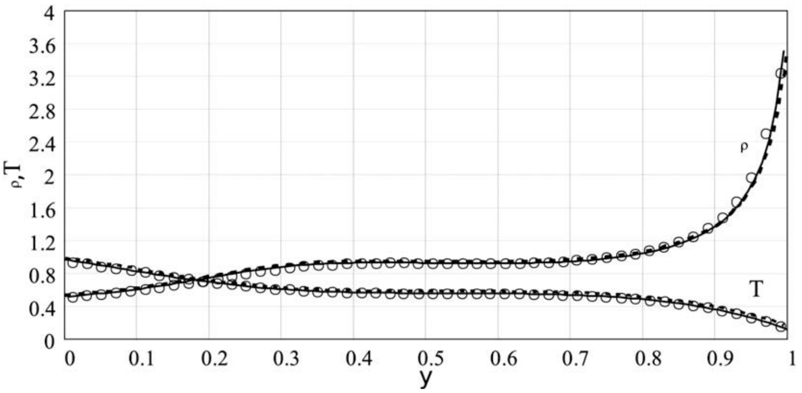
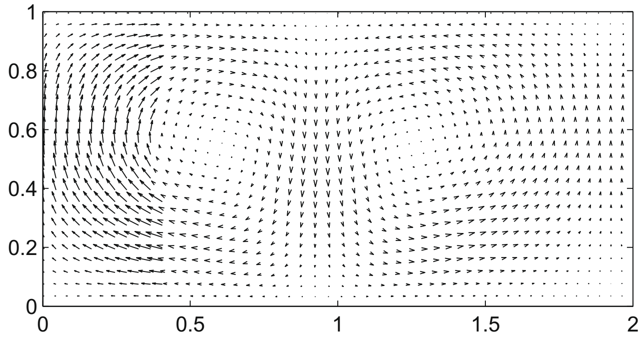


**Fig. 20.** The average density and temperature profiles computed by SIMPLE-TS (solid line), DSMC (circles) and finite difference (dash-line).



ators $max_{0<x<L_{ch}}(\cdot)$, $min_{0<x<L_{ch}}(\cdot)$, $mean_{0<x<L_{ch}}(\cdot)$, respectively, to the velocity fields $u(x,y)$ and $v(x,y)$. Fig. 20 illustrate the averaged density and temperature profiles, which are obtained by applying operator $mean_{0<x<L_{ch}}(\cdot)$, to the density $\rho(x,y)$ and temperature $T(x,y)$ fields. Fig. 21 shows the vector plot of velocity computed by SIMPLE-TS. The results, shown in Figs. 19–21, are compared with the corresponding results available in Stefanov et al. [10] and the agreement is very good.

## 5. Conclusions

The algorithm of method SIMPLE-TS is derived from the general form of equations of compressible viscous gas. The considered problems concern gas flows of very different nature. The pressure driven flow in a long microchannel and the subsonic flow past square in microchannel are low speed flows. The supersonic flow past a square confined in microchannel, illustrates a high-speed flow with shock waves and large gradients of the macroscopic variables. The calculations of the Rayleigh–Bénard convection flow of a rarefied gas demonstrates the ability of the proposed SIMPLE-TS to capture complex nonlinear phenomena such as lost of stability and pattern formation in compressible gas flow investigated in continuum limit. All these problems are calculated in a straightforward way by using SIMPLE-TS without any additional improvements or modifications. All obtained results are in a very good agreement with the other available results or obtained by other methods during the research work on the present paper. The comparison between SIMPLE, PISO and SIMPLE-TS calculations shows that the SIMPLE-TS method is more efficient than SIMPLE and slightly more efficient than PISO. Another advantage of SIMPLE-TS is that the method works effectively without under-relaxation coefficients.

## Acknowledgments

The author K. Shterev appreciates the support of the EUROPEAN SOCIAL FUND, OPERATIONAL PROGRAMME "HUMAN RESOURCES DEVELOPMENT", Grant No. BG051PO001/07/3.3-02-55/17.06.2008. The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 under grant agreement ITN GASMEMS No. 215504. These results were obtained by using the GRID sites of the South East European Regional Operating Centre of the EGEE project (http://public.eu-egee.org).

## References

[1] M. Gad-el-Hak, The fluid mechanics of microdevices – the Freeman scholar lecture, ASME J. Fluid Eng. 121 (1999) 5.
[2] G. Karniadakis, A. Beskok, N. Aluru, Microflows and Nanoflows. Fundamentals and Simulation, Springer Science + Business Media Inc., 2005.
[3] S. Colin, Rarefaction and compressibility effects on steady and transient flows in microchannels, Microfluid. Nanofluid. 1 (2005) 268.
[4] H. Struchtrup, Macroscopic Transport Equations for Rarefied Gas Flows, Springer, Germany, 2005.
[5] X.J. Gu, D.R. Emerson, A computational strategy for the regularized 13 moment equations with enhanced wall-boundary conditions, J. Comput. Phys. 225 (2007) 263.
[6] S. Mizzi, R.W. Barber, D.R. Emerson, J.M. Reese, S.K. Stefanov, A phenomenological and extended continuum approach for modelling non-equilibrium flows, Continuum Mech. Thermodynam. 19 (2007) 273.
[7] C. Cercignani, Rarefied Gas Dynamics. From Basic Concept to Actual Calculations, University Press, Cambridge, 2000.
[8] Y. Sone, Molecular Gas Dynamics: Theory, Techniques, and Applications, Birkhäser, Boston, 2007.
[9] G.A. Bird, Molecular, Gas Dynamics and the Direct Simulation of Gas Flows, Clarendon Press, Oxford, 1994.
[10] S. Stefanov, V. Roussinov, C. Cercignani, Rayleigh–Bénard flow of a rarefied gas and its attractors. I: Convection regime, Phys. Fluid 14 (2002) 2255, doi:10.1063/1.1483837.
[11] F.H. Harlow, J.E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, Phys. Fluid 8 (1965) 2182.
[12] A.A. Amsden, F.H. Harlow, The SMAC Method: A Numerical Technique for Calculating Incompressible Fluid Flows, Los Alamos Scientific Laboratory Report No. LA-4370, 1970.
[13] F.H. Harlow, A.A. Amsden, A numerical fluid dynamics calculation method for all speeds, J. Comput. Phys. 8 (1971) 197.
[14] S.V. Patankar, D.B. Spalding, A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows, Int. J. Heat Mass Transfer 15 (1972) 1787.
[15] C.W. Hirt, A.A. Amsden, J.L. Cook, An arbitrary Lagragian–Eulerian computing method for all flow speeds, J. Comput. Phys. 14 (1974) 227.

[16] S.V. Patankar, Numerical Heat Transfer and Fluid Flow, Hemisphere Publishing Corporation, New York, 1980.

[17] H.M. Domanus, R.C. Schmitt, W.T. Sha, V.L. Shah, New Implicit Numerical Solution Scheme in the COMMIX-1A Computer Program, Report Number(s)NUREG/CR-3435; ANL-83-64, 1983.

[18] J.P. Van Doormal, G.D. Raithby, Enhancement of the SIMPLE method for predicting incompressible fluid flows, Numer. Heat Transfer 7 (1984) 147–163.

[19] R.I. Issa, Solution of the implicitly discretised fluid flow equations by operator-splitting, J. Comput. Phys. 62 (1986) 40.

[20] K.C. Karki, S.V. Patankar, Pressure based calculation procedure for viscous flows at all speeds in arbitrary configurations, AIAA J. 27 (9) (1989) 1167.

[21] R.I. Issa, A.D. Gosman, A.P. Watkins, The computation of compressible and incompressible recirculating flows by a non-iterative implicit scheme, J. Comput. Phys. 62 (1986) 66.

[22] E.B. Arkilic, M.A. Schmidt, K.S. Breuer, Gaseous slip flow in long microchannels, J. Microelectromech. Syst. 6 (1997) 167.

[23] S.V. Patankar, Calculation of Unsteady Compressible Flows Involving Shocks, Rep. UF/TN/A/4, Mechanical Engineering Dept., Imperial College, London, 1971.

[24] R.I. Issa, F.C. Lockwood, On the prediction of two-dimensional supersonic viscous near walls, AIAA J. 15 (1977) 182.

[25] T.H. Chein, H.M. Domanus, W.T. Sha, COMMIX-PPC: A Three-Dimensional Transient Multicomponent Computer Program for Analysing Performance of Power Plant Condensers, vol. 1: Equations and Numerics, Argonne National Laboratory Report ANL-92/2-Vol.1, 1993. doi:10.2172/10147024.

[26] J.B. Scarborough, Numerical Mathematical Analysis, fourth ed., Johns Hopkins University Press, Baltimore, MD, 1958.

[27] C. Cercignani, The Boltzmann Equation and its Applications, Springer, New York, 1988.

[28] H.K. Versteeg, W. Malalasekra, An Introduction to Computational Fluid Dynamics: The Finite Volume Method, second ed., Prentice Hall, Pearson, 2007.

[29] S.K. Stefanov, C. Cercignani, Monte Carlo simulation of Bénard's instability in a rarefied gas, Eur. J. Mech. B: Fluid 11 (1992) 543.

[30] R.I. Issa, B. Ahmady-Befrui, K.R. Beshay, A.D. Gosman, Solution of the implicitly discretised reacting flow equations by operator-splitting, J. Comput. Phys. 93 (1991) 388310.