Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

Journal of Computational Physics 202 (2005) 645–663

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

Acceleration of steady-state lattice Boltzmann simulations on non-uniform mesh using local time step method

Taro Imamura [a,*], Kojiro Suzuki [b], Takashi Nakamura [a], Masahiro Yoshida [a]

[a] *Japan Aerospace Exploration Agency, Institute of Space Technology and Aeronautics, 7-44-1, Jindai-ji Higashi-mati, Chofu-si, Tokyo 182-8522, Japan*

[b] *Department of Advanced Energy, The University of Tokyo, 5-1-5 Kashiwa-no-ha, Kashiwa, Chiba 277-8562, Japan*

Received 29 December 2003; received in revised form 21 June 2004; accepted 2 August 2004
Available online 15 September 2004

Abstract

In this paper, the local time step method is applied to the lattice Boltzmann method (LBM) on non-uniform grid (generalized form of interpolation supplemented LBM: GILBM) for steady flow simulations. The local time step method is known to be an effective technique to accelerate the solution to the converged steady state on non-uniform grid. First, code validation for global time step method is performed by solving flow around a cylinder at low Reynolds numbers. Our results show good agreement with previous studies. Two-dimensional flow simulations around an airfoil were performed in order to validate the code using the local time step method and the results are compared with the global time step solutions. The pressure distribution and the aerodynamic coefficients of global and local time step results are in good agreement. They are also consistent with the previous studies. At the same time, CPU time required to obtain the steady-state solutions using the local time step are reduced by 70–80% compared with that using global time step.

Keywords: Lattice Boltzmann method; Generalized coordinate; Local time step method

1. Introduction

The lattice Boltzmann method (LBM) has been proved to be an promising method in simulating incompressible flow, porous-media flow, and multi-phase flow [1]. Since various modifications have been applied

[*] Corresponding author. Tel.: +81 422 40 1016; fax: +81 422 40 1026.
E-mail address: imamura.taro@jaxa.jp (T. Imamura).

doi:10.1016/j.jcp.2004.08.001

to LBM, the accuracy and the efficiency of the simulation has improved compared with the early stage of its development.

It is important to reduce the computational time in order to use LBM as a practical computational fluid dynamics (CFD) tool. However, the grid used in the standard LBM, commonly referred to as Lattice Bhatnagar–Gross–Krook (LBGK) model [2], is restricted to orthogonal grid with equal spacing. In order to solve high Reynolds number flows with sufficient spatial accuracy, larger computer resources are required compared with the traditional Navier–Stokes solvers on non-uniform mesh.

Recently, many studies have been dedicated to the extension of the LBM on non-uniform mesh. The strategies are classified into two groups. The first type is to patch fine orthogonal grid only to the region where high resolution is required. The adaptive mesh refinement (AMR) technique or composite grid system are applied to LBM [3–7]. Although the grid spacing is different between the grids, the algorithm applied to the each grid is exactly the same as LBGK. A special treatment is implemented only to the boundary of the grids where the exchange of values between the grids is necessary. The second type is to use body-fitted grids and has been formulated by many authors [8–15]. Using these methods, flow around various configurations can be solved. Among these methods, interpolation supplemented lattice Boltzmann method (ISLBM) [12–14] and characteristic Galerkin finite element method for the discrete Boltzmann equation (CGDBE) [15] seem to give results without excessive numerical dissipation.

Although the computational time is considerably reduced by placing the grid points to the region where the high resolution is required, the calculation time is still longer than that by the conventional incompressible Navier–Stokes solvers, such as marker-and-cell (MAC) method [16] and so on. The time steps for both LBM and MAC method must satisfy the advection term stability condition, which is called as Courant–Friedrichs–Lewy (CFL) condition. However, the advection speed used to evaluate the time step Δt is different. While MAC method uses flow velocity (macroscopic velocity), LBM uses the speed of the particle, which is the order of the sound speed. If the same grid is used for the both simulations, the time step defined from the CFL condition becomes inversely proportional to the advection speeds of the methods. When calculating low Mach number flow with LBM, macroscopic fluid velocity is defined smaller by one order compared with the sound speed. Thus, the time step used in the LBM simulation also becomes smaller by one order compared with the MAC method. Even the CPU time required for one time step is smaller for LBM, the total calculation time becomes several times larger. Further reduction in CPU time is necessary.

There are requirements in obtaining steady-state solution for incompressible flow simulations. Particularly in aerospace problems, drag prediction is an important issue in designing process and drag force is often obtained from the steady-state solution. Furthermore, steady-state solution is often used as an initial condition for unsteady flow calculation, in order to reduce the total computational time. Several ways to accelerate the solutions to converged state have been proposed using grid refinement [4] and multi-grid methods [7]. In this paper, we will apply the local time step method [17] to LBM on non-uniform grid in order to obtain steady-state solution with reduction in computational time. The local time step method is an effective technique to accelerate the solution to the converged steady state on non-uniform grid. The local time step method means that each grid point uses a time step, which is based on the local advection term stability condition. The convergence to the steady-state flow is expected to be accelerated, since larger time step can be used for larger grid cell. In this study, generalized form of ISLBM is used as LBM solver on generalized coordinates.

This paper is organized as follows. Section 2 briefly describes the LBGK model. Section 3 presents the LBM on body-fitted grid. A generalization of the ISLBM [12–14] to arbitrary structured grids is proposed and is named as generalized form of interpolation supplemented LBM (GILBM). Section 4 presents numerical procedure to apply the local time step method to GILBM. Section 5 presents the computational results both for global and local time step. Section 6 concludes this paper. All the descriptions made in this paper are for a two-dimensional plane.

2. Lattice BGK model

Frequently, the nine-velocity model (2D-9V model) [18] is used for two-dimensional calculations. The governing equation of LBM, lattice Boltzmann equation (LBE), is described as

$$f_i(\mathbf{x}+\mathbf{c}_i\Delta t, t+\Delta t) - f_i(\mathbf{x},t) = -\frac{1}{\omega}[f_i(\mathbf{x},t) - f_i^{\text{eq}}(\mathbf{x},t)], \tag{1}$$

where $i = 0,\ldots,8$, $\omega \equiv \tau/\Delta t$. ω is the relaxation time.

The particle velocity vectors set for the 2D-9V model is

$$\begin{aligned}
&\mathbf{c}_0 = (0,0),\\
&\mathbf{c}_i = c\left(\cos\left(\frac{i-1}{2}\pi\right), \sin\left(\frac{i-1}{2}\pi\right)\right) \quad \text{for } i = 1,2,3,4,\\
&\mathbf{c}_i = \sqrt{2}c\left(\cos\left(\frac{i-5}{2}\pi+\frac{\pi}{4}\right), \sin\left(\frac{i-5}{2}\pi+\frac{\pi}{4}\right)\right) \quad \text{for } i = 5,6,7,8,
\end{aligned} \tag{2}$$

where c is the streaming speed of the particle. The reference length L, mean flow velocity U_0, reference time $t_0(= L/U_0)$, and reference density ρ_0 are used in order to describe the variables in non-dimensional form, which is the same as that of incompressible Navier–Stokes equations. The LBE consists of collision and advection steps. By dividing Eq. (1), each step can be described independently as

$$f_i^*(\mathbf{x},t) = f_i(\mathbf{x},t) - \frac{1}{\omega}[f_i(\mathbf{x},t) - f_i^{\text{eq}}(\mathbf{x},t)], \tag{3}$$

$$f_i(\mathbf{x},t+\Delta t) = f_i^*(\mathbf{x}-\mathbf{c}_i\Delta t, t), \tag{4}$$

where $f_i^*(\mathbf{x},t)$ indicates the post-collision distribution function. The collision term is calculated at position $\mathbf{x}$ according to Eq. (3). Then, the post-collision distribution functions move to the neighboring nodes, depending on the particle velocity $\mathbf{c}_i$ as Eq. (4).

The macroscopic variables such as density and velocity in non-dimensional form are obtained as

$$\rho = \sum_i f_i, \quad \rho u_\alpha = \sum_i f_i c_{i,\alpha}. \tag{5}$$

The equilibrium distribution functions for the discrete velocities $\mathbf{c}_i$ are defined as

$$f_i^{\text{eq}}(\mathbf{x},t) = w_i\rho\left[1 + \frac{3(c_{i,\alpha}u_\alpha)}{c^2} + \frac{9(c_{i,\alpha}u_\alpha)^2}{2c^4} - \frac{3u^2}{2c^2}\right]. \tag{6}$$

The coefficient w_i is a constant which depends on the velocity model. The constants for the 2D-9V model are

$$\begin{aligned}
&w_0 = 4/9,\\
&w_1 = \cdots = w_4 = 1/9,\\
&w_5 = \cdots = w_8 = 1/36.
\end{aligned} \tag{7}$$

By applying Taylor and Chapman–Enskog expansions to the LBE, the equation of continuity and Navier–Stokes equations are derived as

$$\frac{\partial\rho}{\partial t} + \frac{\partial(\rho u_\alpha)}{\partial x_\alpha} = 0 + \mathrm{O}(\delta^2), \tag{8}$$

$$\frac{\partial(\rho u_\alpha)}{\partial t}+\frac{\partial(\rho u_\alpha u_\beta)}{\partial x_\beta}=-\nabla p+\Pi^{(1)}_{\alpha\beta}+\mathrm{O}(\delta^2)+\mathrm{O}(M^3), \tag{9}$$

where $M = U/c$, δ is small parameter which is the same order of Δt, and $\Pi^{(1)}_{\alpha\beta}$ is the stress tensor for a Newtonian fluid [19].

The relation between the viscosity and relaxation time for LBGK is $\nu = (1/6)(2\omega - 1)c^2\Delta t$. Thus the Reynolds number, which is the only parameter for the non-dimensional incompressible Navier–Stokes equations, is described as

$$Re=\frac{6}{(2\omega-1)\Delta t c^2}, \tag{10}$$

and the pressure becomes

$$p=\frac{\rho}{3}c^2=\rho c_s^2, \tag{11}$$

where $c_s \equiv c/\sqrt{3}$ is the sound speed.

3. Generalized form of interpolation supplemented lattice Boltzmann method

3.1. Numerical method for GILBM

In this section, LBGK model on generalized coordinate is proposed based on the idea of ISLBM [12–14]. Since the transformation between physical plane and computational plane must be described by analytical function for ISLBM, a generalization of ISLBM is necessary to calculate on arbitrary structured grid. Numerical procedure for general cases is introduced hereafter.

The governing equations (3) and (4) on an orthogonal coordinates are transformed into generalized coordinates. The physical and computational planes are described as $\mathbf{x} \equiv (x_1,x_2) \equiv (x,y)$ and $\boldsymbol{\xi} \equiv (\xi_1,\xi_2) \equiv (\xi,\eta)$, respectively.

The transformation of the collision term equation into generalized coordinates is simple, because only information local to the current grid node is required. Thus, by replacing $\mathbf{x}$ in Eq. (3) to $\boldsymbol{\xi}$, collision term calculation on generalized coordinates is obtained as

$$f_i^*(\boldsymbol{\xi},t)=f_i(\boldsymbol{\xi},t)-\frac{1}{\omega}[f_i(\boldsymbol{\xi},t)-f_i^{\mathrm{eq}}(\boldsymbol{\xi},t)], \tag{12}$$

where $f_i^*(\boldsymbol{\xi},t)$ indicates the post-collision distribution function.

After the collision step, the advection term calculation is performed. Transformation of the advection term equation requires some discussions. Fig. 1 shows the $\mathbf{c}_1 = c(1,0)$ vectors on the physical and compu-

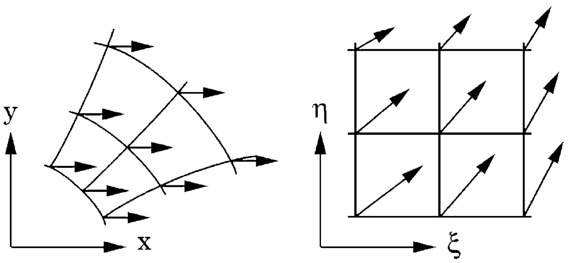

Fig. 1. $c(1,0)$ vectors on physical plane and computational plane.

tational planes, respectively. They are constant vectors on the physical plane, but the contravariant velocities $\tilde{\mathbf{c}}_i(\equiv \tilde{c}_i, \alpha)$ defined as

$$\tilde{c}_{i,\alpha} = c_{i,\beta}\frac{\partial \xi_\alpha}{\partial x_\beta}, \tag{13}$$

on the computational plane are not constant. Consequently, integration of the contravariant velocity over time step Δt must be performed to calculate the position where the distribution function comes from.

The transformed equation of Eq. (4) becomes

$$f_i(\boldsymbol{\xi}, t + \Delta t) = f_i^*(\boldsymbol{\xi} - \Delta\boldsymbol{\xi}_{up,i}, t), \tag{14}$$

where

$$\Delta\boldsymbol{\xi}_{up,i} = \int_0^{\Delta t} \mathrm{d}\boldsymbol{\xi}_i = \int_0^{\Delta t} \tilde{\mathbf{c}}_i \,\mathrm{d}t. \tag{15}$$

In order to solve Eq. (14) numerically, numerical accuracy of the discretization methods are important. The numerical errors must be suppressed to the same order of the truncation error of LBE when deriving Navier–Stokes equations. The estimation of the contravariant velocity $\tilde{\mathbf{c}}_i$ at each node, the integration scheme applied to Eq. (15), and the interpolation method applied to the right-hand side of Eq. (14) are important to suppress numerical errors.

First, for the calculation of the contravariant velocities at each node, the estimation of the metrics (e.g. $\xi_x \equiv \partial\xi/\partial \mathrm{x}$) is necessary

$$\begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} = J\begin{bmatrix} y_\eta & -x_\eta \\ -y_\xi & x_\xi \end{bmatrix}, \tag{16}$$

where the Jacobian J is defined as

$$J = x_\xi y_\eta - x_\eta y_\xi. \tag{17}$$

The central-differences are used for the calculation of the derivatives, such as x_ξ. Applying the results of Eq. (16) to Eq. (13), the contravariant velocities at the grid nodes are obtained with second-order accuracy in space.

Second, the integration of the contravariant velocity over the time step Δt is performed. The integration of the contravariant velocity using first-order Euler explicit integration becomes

$$\Delta\boldsymbol{\xi}_{up,i} = \Delta t\tilde{\mathbf{c}}_i(\boldsymbol{\xi}) + \mathrm{O}(\Delta t^2). \tag{18}$$

However, the numerical accuracy cannot be maintained using first-order Euler explicit integration. The two-step Runge–Kutta method is necessary when the grid is clustered around the body and is calculated as

$$\Delta\boldsymbol{\xi}_{up,i}^{(1)} = \frac{1}{2}\Delta t\tilde{\mathbf{c}}_i(\boldsymbol{\xi}) \quad : \mathrm{1st - step}, \tag{19}$$

$$\Delta\boldsymbol{\xi}_{up,i} = \Delta t\tilde{\mathbf{c}}_i(\boldsymbol{\xi} - \Delta\boldsymbol{\xi}_{up,i}^{(1)}) + \mathrm{O}(\Delta t^3) \quad : \mathrm{2nd - step}. \tag{20}$$

The LBE on generalized coordinate can be described as

$$f_i(\boldsymbol{\xi} + \Delta\boldsymbol{\xi}_{up,i}, t + \Delta t) = f_i(\boldsymbol{\xi}, t) - \frac{1}{\omega}[f_i(\boldsymbol{\xi}, t) - f_i^{\mathrm{eq}}(\boldsymbol{\xi}, t)], \tag{21}$$

by simply combining Eqs. (12) and (14). By applying the Taylor expansion to the above equation,

$$\frac{\partial f_i}{\partial t} + \bar{c}_{i,\alpha}\frac{\partial f_i}{\partial \xi_\alpha} + \frac{1}{2}\left(\frac{\partial}{\partial t} + \bar{c}_{i,\alpha}\frac{\partial}{\partial \xi_\alpha}\right)^2 f_i \Delta t + \mathrm{O}(\Delta t^2) = -\frac{1}{\omega \Delta t}(f_i - f_i^{\mathrm{eq}}) \tag{22}$$

is derived, where $\bar{c}_{i,\alpha} \equiv \Delta\boldsymbol{\xi}_{up,i}/\Delta t$. The stress tensor term of the Navier–Stokes equation is derived from the third term in LHS of Eq. (22), and the order of this term is $\mathrm{O}(\Delta t)$. When first-order Euler explicit integration is used for the estimation of $\Delta\boldsymbol{\xi}_{up,i}$, the leading error term of $\bar{c}_{i,\alpha}$ is $\mathrm{O}(\Delta t)$. The error from the second term becomes the same order from the third term in LHS of Eq. (22). This indicates that using first-order Euler explicit integration is insufficient because the error from the second term affects the third term. Therefore, the integration of the contravariant velocities must be higher than second order, in order to suppress the leading error term of $\bar{c}_{i,\alpha}$ to $\mathrm{O}(\Delta t^2)$. Some results are shown later to confirm this point.

The calculation of the contravariant velocity and its integration are performed only once at the beginning of the simulation, since these values do not change during the simulation. Although extra memory is required to store the integrated contravariant velocities in an array, nearly 50% CPU time is saved each time step.

Third, the interpolation function is necessary to calculate the right-hand side of Eq. (14). Also the same interpolation function is used in Eq. (20). In both equations, the value between the grid points is required. It is pointed out in previous study [13] that using a second-order scheme is important to suppress numerical dissipation. Second-order upwind quadratic interpolation is used in the same manner as ISLBM. For two-dimensional case, the interpolation function g_i becomes,

$$\begin{aligned} g_i(\boldsymbol{\xi} - \Delta\boldsymbol{\xi}_{up,i}) &= a_{i,0,2}(a_{i,0,1}g_{i,00} + a_{i,1,1}g_{i,10} + a_{i,2,1}g_{i,20}) \\ &\quad + a_{i,1,2}(a_{i,0,1}g_{i,01} + a_{i,1,1}g_{i,11} + a_{i,2,1}g_{i,21}) \\ &\quad + a_{i,2,2}(a_{i,0,1}g_{i,02} + a_{i,1,1}g_{i,12} + a_{i,2,1}g_{i,22}) \\ &\equiv \sum_{m=0}^{2}\sum_{l=0}^{2} a_{i,m,2}a_{i,l,1}g_{i,lm}, \end{aligned} \tag{23}$$

where the $g_{i,lm}$ are the stencil as shown in Fig. 2 depending on the position of $\boldsymbol{\xi}$-$\Delta\boldsymbol{\xi}_{up,i}$. The coefficients before $g_{i,lm}$ are

$$\begin{aligned} a_{i,0,\alpha} &= \frac{1}{2}(|\Delta\xi_{up,i,\alpha}| - 1)(|\Delta\xi_{up,i,\alpha}| - 2), \\ a_{i,1,\alpha} &= -|\Delta\xi_{up,i,\alpha}|(|\Delta\xi_{up,i,\alpha}| - 2), \\ a_{i,2,\alpha} &= \frac{1}{2}|\Delta\xi_{up,i,\alpha}|(|\Delta\xi_{up,i,\alpha}| - 1), \end{aligned} \tag{24}$$

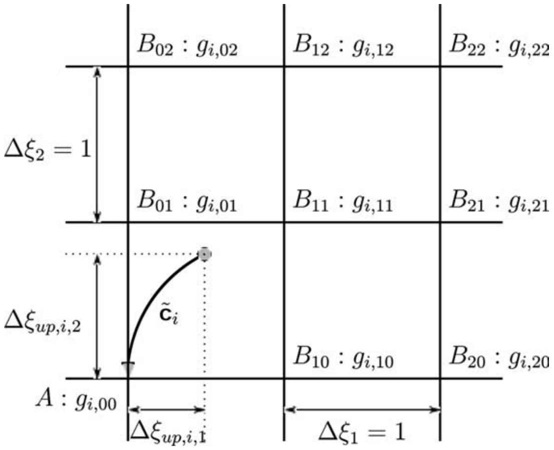

Fig. 2. Stencil used for the interpolation function.

where the grid spacing is $\Delta\xi_1 = \Delta\xi_2 = 1$. Since multi-dimensional interpolation function is used, the present algorithm becomes exactly the same as LBGK model, if the calculation is performed on orthogonal grid with equal spacing. The time step Δt_g for the present method can be given from the Courant–Friedrichs–Lewy (CFL) condition, as it is given in traditional CFD methods

$$\Delta t_g = \lambda \min_{i,\alpha,j,k} \left[\frac{1}{\tilde{c}_{i,\alpha}\big|_{j,k}} \right], \tag{25}$$

where λ is the CFL number ($0 < \lambda \leqslant 1$) and (j,k) is the coordinate of grid point on the computational domain. Since Δt_g satisfies the numerical stability condition at every computational points, it is called global time step. It is pointed out by Lee and Lin [20] that CFL condition must be satisfied in order to perform stable calculation.

The set up of the parameters used in the calculation is summarized as follows:

1. Set the following value; Reynolds number (Re), $U/c(=0.1)$, $\lambda(=1.0)$.
2. Calculate the contravariant velocity; Eq. (13).
3. Calculate the global time step; Eq. (25).
4. Calculate the relaxation time; Eq. (10).

3.2. Boundary conditions of GILBM

In this study, suitable wall boundary condition for the generalized coordinates is obtained based on the idea of the incompressible Navier–Stokes solvers. The wall boundary condition for the incompressible Navier–Stokes solvers is defined as follows. Since the normal gradient of the pressure vanishes at the wall boundary, pressure on the wall is extrapolated from the node in the computational domain next to the wall boundary node. In LBM, similar expression for density is obtained since $p \sim \rho$ (see Eq. (11)). The flow velocity at the wall is given as wall velocity.

From the boundary condition of the macroscopic variables (ρ,u,v), the boundary condition of the distribution function is calculated. If we assume $f_i|_{bc} = f_i^{\text{eq}}$, distribution function can be calculated using Eq. (6). However, this assumption is not accurate enough since viscosity effect appears in the first order of non-equilibrium term. The distribution function is estimated up to the first order of non-equilibrium using the Chapman–Enskog expansion as

$$\begin{aligned} f_i|_{bc} &= f_i^{\text{eq}} + f_i^{(1)} + \cdots \\ &= f_i^{\text{eq}} \left[1 - \omega\Delta t \left(\frac{3U_{i,\alpha}U_{i,\beta}}{c^2} - \delta_{\alpha\beta} \right) \frac{\partial u_\alpha}{\partial x_\beta} \right] + \mathrm{O}(\delta^2), \end{aligned} \tag{26}$$

where $U_{i,\alpha} = c_{i,\alpha} - u_\alpha$, δ is small parameter, and $\delta_{\alpha\beta}$ is Kronecker delta. The derivation of above equation is described in Appendix A. The spatial derivatives of the velocity $\partial u_\alpha/\partial x_\beta$ in Eq. (26) are calculated as

$$\frac{\partial u_\alpha}{\partial x_\beta} = \frac{\partial u_\alpha}{\partial \xi_\gamma} \frac{\partial \xi_\gamma}{\partial x_\beta}. \tag{27}$$

The first term in right-hand side $(\partial u_\alpha/\partial \xi_\gamma)$ is calculated using second-order one-sided finite difference method. The second term $(\partial \xi_\gamma/\partial x_\beta)$ is the metrics at the wall boundary, which is already calculated.

4. GILBM using local time step method

The calculation time is considerably reduced by using GILBM compared with that on orthogonal grid, because grid points can be significantly reduced at region far away from the wall boundary. However, the total calculation time is still longer than that by the conventional incompressible Navier–Stokes solvers, such as the MAC method [16]. The time step for the GILBM and MAC methods are defined from the CFL condition. While the macroscopic fluid velocity U is used as the advection speed in the MAC method, GILBM uses the speed of the particle c, which is on the order of the sound speed. If the same grid is used for the both simulations, the time step defined from CFL condition becomes inversely proportional to the advection speeds of the methods. Thus, the time step used in the GILBM simulation is approximately 10% (when $U/c = 0.1$) of that in the MAC method. Although the calculation time required for one time step is smaller than MAC method, the total calculation time of GILBM becomes several times larger. Further reduction in the calculation time is required.

Unlike the LBGK model, the grid size for GILBM is not uniform. Thus, a local time step method [17] can be applied to GILBM in order to obtain steady-state solution with reduction in CPU time. For larger grid sizes, a larger time step, which satisfies the CFL condition, can be used. The time step is defined locally from the local CFL condition as

$$\Delta t_l\big|_{j,k} = \lambda \min_{i,\alpha}\left[\frac{1}{\tilde{c}_{i,\alpha}\big|_{j,k}}\right], \tag{28}$$

where λ is the CFL number. This time step $\Delta t_l|_{j,k}$ is called local time step. Having different time step sizes for each grid point corresponds to a calculation of

$$f_i(\boldsymbol{\xi} + \tilde{\mathbf{c}}_i \Delta t_l\big|_{j,k}, t + \Delta t_l\big|_{j,k}) - f_i(\boldsymbol{\xi}, t) = -\frac{1}{\omega_a\big|_{j,k}}[f_i(\boldsymbol{\xi}, t) - f_i^{\mathrm{eq}}(\boldsymbol{\xi}, t)]. \tag{29}$$

In order to describe the constant Reynolds number, the relaxation time $\omega_a|_{j,k}$ is defined locally as

$$\omega_a\big|_{j,k} = \frac{3}{Re\Delta t_l\big|_{j,k} c^2} + \frac{1}{2}, \tag{30}$$

using the local time step $\Delta t_l|_{j,k}$. Since Re, Δt_l and c are all positive values, $\tau > 1/2$ is also maintained when the local time step method is used.

By applying the Taylor and Chapman–Enskog expansions to Eq. (29), the following equations, which are similar to the equation of continuity and the Navier–Stokes equations on physical plane, are derived as

$$\frac{1}{a\big|_{j,k}}\frac{\partial\rho}{\partial t} + \frac{\partial(\rho u_\alpha)}{\partial x_\alpha} = 0 + \mathrm{O}(\delta^2), \tag{31}$$

$$\frac{1}{a\big|_{j,k}}\frac{\partial(\rho u_\alpha)}{\partial t} + \frac{\partial(\rho u_\alpha u_\beta)}{\partial x_\beta} = -\nabla p + \Pi^{(1)}_{\alpha\beta} + \mathrm{O}(\delta^2) + \mathrm{O}(M^3), \tag{32}$$

where $a|_{j,k} \equiv \Delta t_l|_{j,k}/\Delta t_g$ is the acceleration factor, and Δt_g is a global time step defined by Eq. (25). There are two differences between the above equations and Eqs. (8) and (9). The first difference is the introduction of the inverse of the acceleration factor $1/a|_{j,k}$ to the time derivative term. If a steady state is found for the system above, the time derivative term must be zero irrespective of the value of the acceleration factor $a|_{j,k}$. Therefore, a steady-state solution of Eq. (29) becomes that of the original equations (Eq. (1)) as well. The second difference appears in δ. Since the leading error term δ is proportional to time step Δt, slight difference in the results should be observed. However, this error is negligible and we will confirm this point by checking the calculated results in the following section.

Numerical procedure to calculate with local time step is described from now on. When time step and relaxation time vary between grid points, modification must be applied to the distribution functions. The similar discussion is made in the previous studies [3,4] when exchange between coarse and fine grid system is necessary, since the time step and relaxation time differs between the grid systems.

The distribution function, which is described up to the first-order non-equilibrium term, can be written as

$$\begin{aligned} f_i &= f_i^{\mathrm{eq}} + f_i^{(1)} + \cdots \\ &= f_i^{\mathrm{eq}} - \omega_a|_{j,k}\Delta t_l|_{j,k}\left(\frac{1}{a|_{j,k}}\frac{\partial f_i^{\mathrm{eq}}}{\partial t} + c_{i,\alpha}\frac{\partial f_i^{\mathrm{eq}}}{\partial x_\alpha}\right) + \mathrm{O}(\delta^2) \end{aligned} \tag{33}$$

for the accelerated system. By assuming a steady-state solution, the time derivative term converges to zero. The first-order non-equilibrium term $f_i^{(1)}$ is linear to the product of relaxation time $\omega_a|_{j,k}$ and time step $\Delta t_l|_{j,k}$. This product becomes a function of $a|_{j,k}$ as

$$\omega_a|_{j,k}\Delta t_l|_{j,k} = \frac{3}{Re\,c^2} + \frac{a|_{j,k}\Delta t_g}{2}. \tag{34}$$

This indicates that $f_i^{(1)}$ depends on the acceleration factor $a|_{j,k}$. For example, in order to calculate the advection step in the two-dimensional GILBM, eight points to the upwind side are necessary for the construction of the interpolation function. See Fig. 2 once again which shows the stencil of the distribution function f_i corresponding to the $\mathbf{c}_i$. Now focusing on point A, we will calculate the value f_i for the next time step. B_{lm} represents the points on the upwind side, and B represents an arbitrary point of B_{lm}. The time step and relaxation time at points A and B are Δt_A, ω_A, Δt_B, and ω_B, respectively. In order to construct the interpolation function for point A (see Eq. (23)), the distribution functions at points B must be re-estimated to values corresponding to Δt_A and ω_A. This re-estimation is necessary because the first order of non-equilibrium term depends on $a|_{j,k}$ as shown in Eq. (34).

The re-estimation is applied to $f_i|_B$ before the collision term calculation and the re-estimated distribution function $\tilde{f}_i|_B$ is obtained as

$$\tilde{f}_i|_B = f_i^{\mathrm{eq}}|_B + (f_i|_B - f_i^{\mathrm{eq}}|_B)\frac{\omega_A\Delta t_A}{\omega_B\Delta t_B}, \tag{35}$$

where $f_i|_B - f_i^{\mathrm{eq}}|_B \simeq f_i^{(1)}|_B$. After the re-estimation, the collision term calculation using the relaxation time at point A is performed. The re-estimation and collision calculation described in one equation is written as

$$\begin{aligned} \tilde{f}_i^*|_B &= \tilde{f}_i|_B - \frac{1}{\omega_A}(\tilde{f}_i|_B - f_i^{\mathrm{eq}}|_B) \\ &= f_i^{\mathrm{eq}}|_B + (f_i|_B - f_i^{\mathrm{eq}}|_B)\frac{(\omega_A - 1)\Delta t_A}{\omega_B\Delta t_B}, \end{aligned} \tag{36}$$

where $\tilde{f}_i^*|_B$ is the re-estimated post-collision distribution function at point B. This calculation is applied to the all of the grid points at the upwind side of point A. On the other hand, at point A, re-estimation is unnecessary and only collision calculation is performed. However, the same equation can be use for point A, since Eq. (36) can be reduced to simple collision equation by substituting $\omega_B = \omega_A$ and $\Delta t_B = \Delta t_A$.

After the re-estimation and the collision calculation, the advection term calculation is performed using the interpolation function, Eq. (23). The overall procedure to calculate using the local time method is shown in Fig. 3. The present local time step method can be applied to other types of LBM formulated on non-uniform grid systems for the purpose of reducing the computational time.

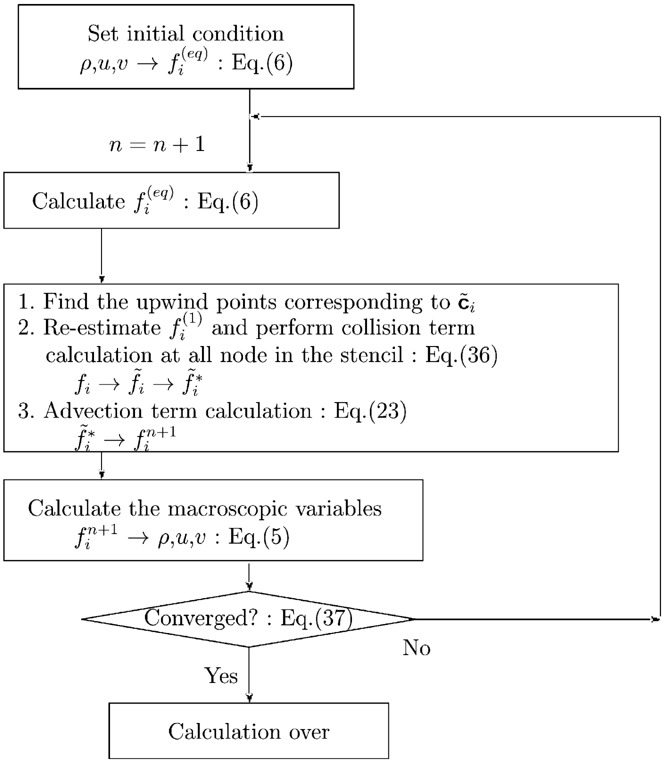
Set initial condition
$\rho,u,v \rightarrow f_i^{(eq)}$: Eq.(6)
$n = n + 1$
Calculate $f_i^{(eq)}$: Eq.(6)
1. Find the upwind points corresponding to $\tilde{\mathbf{c}}_i$
2. Re-estimate $f_i^{(1)}$ and perform collision term calculation at all node in the stencil : Eq.(36)
$f_i \rightarrow \tilde{f}_i \rightarrow \tilde{f}_i^*$
3. Advection term calculation : Eq.(23)
$\tilde{f}_i^* \rightarrow f_i^{n+1}$
Calculate the macroscopic variables
$f_i^{n+1} \rightarrow \rho,u,v$: Eq.(5)
Converged? : Eq.(37)
No
Yes
Calculation over

Fig. 3. Calculation procedure using local time step method.

5. Computational results

5.1. Flow around a circular cylinder

The numerical simulations of two-dimensional flow around a circular cylinder are performed. The Reynolds number is based on the free stream velocity and the diameter of the circular cylinder. The grid used in the simulation is shown in Fig. 4. The grid type is an O-grid, and the resolution is 181 × 241. The minimum grid spacing is defined by $\Delta x_{min} = 0.1/\sqrt{Re}$. The circular outer boundary is located at a distance equal to 20 times the diameter of the cylinder. The Chapman–Enskog and non-reflecting boundary conditions are applied to the wall and to the outer boundary, respectively. A periodic boundary condition is applied at the grid-cut in the wake. The free stream velocity is defined as $U_\infty/c = 0.1$. The uniform flow or the potential flow solutions can be used as an initial condition for the simulations. For the present calculations, latter condition was used in the same manner as He et al. [12].

At low Reynolds numbers (Re = 10, 20 and 40), this calculation is often carried out as a test problem to validate a flow solver on body-fitted coordinates. When the Reynolds number is less than the critical Reynolds number ($Re_{\mathrm{cr}} \sim 47$), a steady recirculating region appears behind the circular cylinder. Convergence to the steady-state solution is evaluated by measuring the maximum of the ratio of the relative velocity change δ, defined as

$$\delta = \mathrm{Max}\frac{|\mathbf{u}(t-\Delta t) - \mathbf{u}(t)|}{|\mathbf{u}(t)|} \leqslant 1.0 \times 10^{-7}. \tag{37}$$

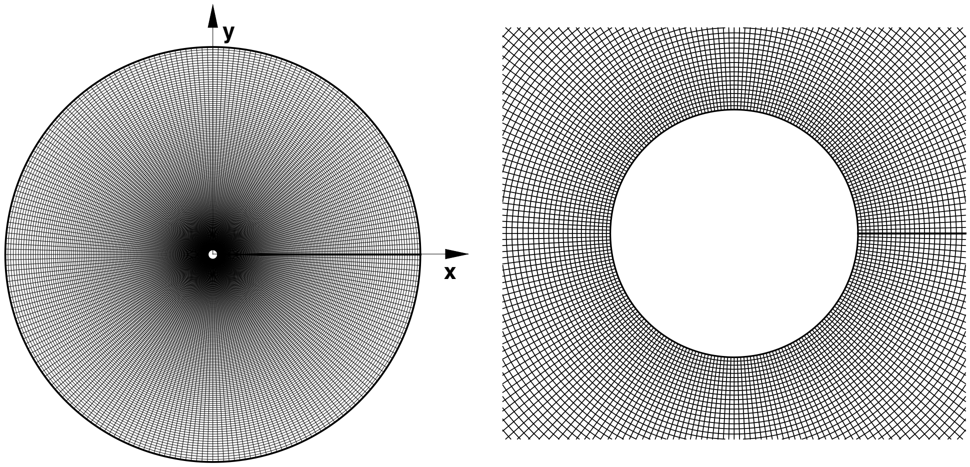

Fig. 4. Body-fitted grid used in circular cylinder flow simulations.

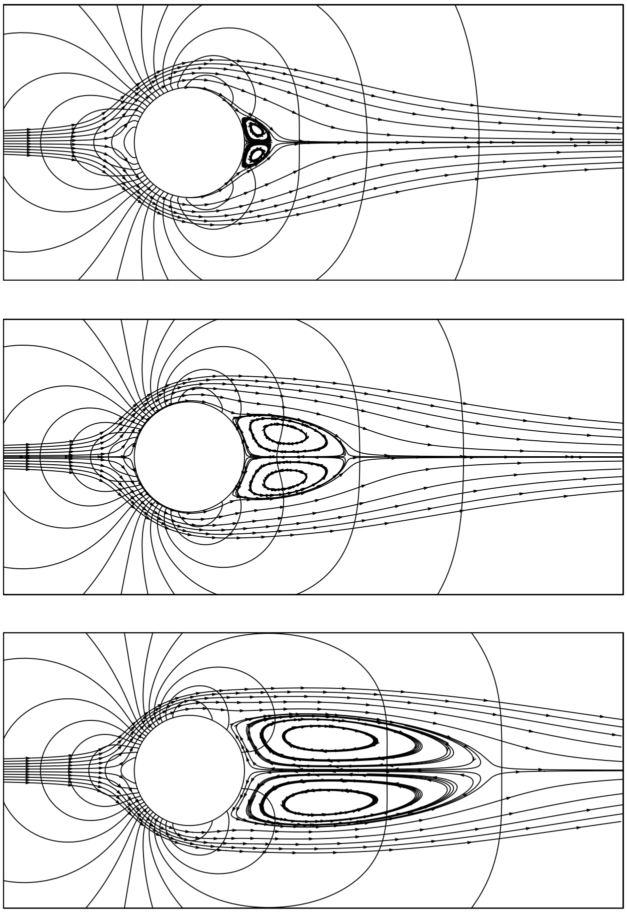

Fig. 5. Pressure contours and streamlines of cylinder flow at $Re = 10$, 20, and 40.

The above calculation is applied only to nodes where $|\mathbf{u}(t)| \geqslant \mathrm{Max}|\mathbf{u}(t)| \times 0.01$ is satisfied in order to avoid division by zero.

Fig. 5 shows the pressure contours and streamlines at $Re = 10$, 20, and 40. A pair of vortices is observed behind the circular cylinder. As the Reynolds number increases, the length of the recirculating region L and the separation angle θ_s (see Fig. 5(c)) increase.

The aerodynamic coefficients and geometrical parameters at low Reynolds number are listed in Table 1 and compared with previous studies [11,12,21,22]. The results by Mei and Shyy [11] were obtained by FDLBM, and the results by He et al. [12] were obtained by ISLBM. Also, the results by Nieuwstadt and Keller [21] and Dennis and Chang [22] were obtained by a finite-difference method.

The drag coefficient C_d and the pressure coefficient C_p are defined as

$$C_d = \frac{F_x}{\frac{1}{2}\rho U^2}, \quad C_p = \frac{(p - p_\infty)}{\frac{1}{2}\rho U^2}, \tag{38}$$

where F_x is the force component which is parallel to the mean flow. F_x is calculated by the stress tensor and normal direction of the body surface.

The drag coefficients obtained by the present method show similar trends relative to the values obtained by finite-difference methods. The stagnation pressure coefficient $C_{p,s}$, and the base pressure coefficient at the rear stagnation point $C_{p,r}$, compare well with the results of previous studies. The separation angle is slightly smaller than the results obtained by finite-difference methods. However, the length of the recirculating region, normalized by the radius, and the separation angle agree with other studies for all three Reynolds numbers.

5.2. Flow around a NACA0012 airfoil

Before we begin the discussion of the local time step method, another validation of the generalized form of ISLBM (GILBM) using the global time step is carried out. A flow around a NACA0012 airfoil [23] was performed at laminar flow condition with $Re = 500$ and an angle of attack (AOA) of 0 [degree]. A body-fitted grid (C-grid) is used in the present simulations, and the resolutions are 257 × 65 and 373 × 141.

Table 1
Comparison with previous studies of aerodynamic coefficients and geometrical parameters for cylinder flow at low Reynolds number

Re	Author	Method	C_d	$C_{p,s}$	$-C_{p,r}$	$L/(0.5D)$	θ_s
10	Dennis and Chang	N.S.	2.846	1.489	0.742	0.53	29.6
	Nieuwstadt and Keller	N.S.	2.828	1.500	0.692	0.434	27.96
	He et al.	ISLBM	3.170	1.393	0.687	0.474	26.89
	Mei and Shyy	FDLBM	–	–	–	0.498	30.0
	Present work	GILBM	2.848	1.403	0.733	0.478	26.0
20	Dennis and Chang	N.S.	2.045	1.269	0.589	1.88	43.7
	Nieuwstadt and Keller	N.S.	2.053	1.274	0.582	1.786	43.37
	He et al.	ISLBM	2.152	1.233	0.567	1.842	42.96
	Mei and Shyy	FDLBM	–	–	–	1.804	42.1
	Present work	GILBM	2.051	1.251	0.589	1.852	43.3
40	Dennis and Chang	N.S.	1.522	1.144	0.509	4.69	53.8
	Nieuwstadt et al.	N.S.	1.550	1.117	0.554	4.357	53.34
	He et al.	ISLBM	1.499	1.113	0.487	4.490	52.84
	Mei et al.	FDLBM	–	–	–	4.38	50.12
	Present work	GILBM	1.538	1.156	0.514	4.454	52.4

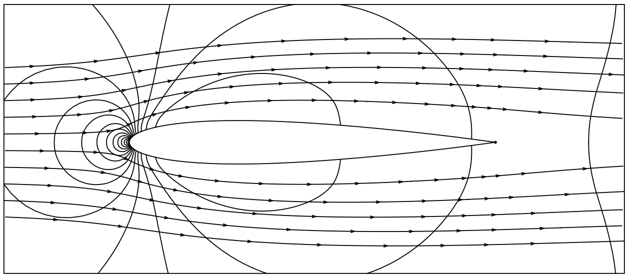

Fig. 6. Pressure contours and streamlines around a NACA0012 airfoil at $Re = 500$, AOA = 0 [degree].

Two different minimum grid sizes normal to the wall boundary are used as $0.1/\sqrt{Re} = 4.5 \times 10^{-3}$ and $0.01/\sqrt{Re} = 4.5 \times 10^{-4}$ in order to clarify the effect of the minimum grid size. The outer boundary is located at 20 times of the chord length. The ratio of macroscopic velocity of the far field to the particle velocity is defined as $U_\infty/c = 0.1$ following the most of the previous studies. Also, we have performed some test calculation varying U/c and around 0.1 was the optimum value considering the numerical error and the computational time. The present result is compared with the numerical simulations using CFL3D and PowerFLOW [24,25]. CFL3D is a Navier–Stokes solver using the finite-volume formulation on generalized coordinates, and PowerFLOW is a commercially available LBM solver using orthogonal grids. Convergence to steady state is confirmed by Eq. (37).

First of all, the pressure contours and streamlines obtained by the present method is shown in Fig. 6. The symmetric flow pattern is observed. The dependence of the aerodynamic coefficients on the grid resolution and the minimum grid spacing is studied in detail as shown in Table 2. The aerodynamic coefficients are compared with those of the other two methods. The drag coefficient obtained from the present method using the highest-resolution grid is about 0.9% smaller than that of CFL3D. As the resolution becomes higher, the drag coefficient obtained from the present method converges to that by CFL3D. On the other hand, PowerFLOW result steps away from the other two results due to insufficient resolution at the boundary, although resolution has increased. Also lift coefficient C_l shows excellent symmetry of the calculation in all cases.

In order to validate the accuracy of the scheme, three additional computation is performed [26]. We chose highest-resolution grid (373×141 grid with $\Delta x_{\min} = 4.5e - 4$, grid C) as a reference grid, and made two grid (A and B) with different resolutions which are listed in Table 3. The ratio of the grid size between the different grid is not 2 due to the restriction in computational resources as it is usually required in the accuracy estimation [26]. Thus, evaluation method proposed by Roache [27] when ratio of the grid size

Table 2
Summary of results for flow around a NACA0012 wing section

	Resolution (on wing)	$\Delta x_{\min}$	C_d	C_l
Present work	257 × 65 (173)	4.5e − 3	0.1682	1.0e − 13
	373 × 141 (251)	4.5e − 3	0.1672	1.0e − 13
	257 × 65 (173)	4.5e − 4	0.1736	1.0e − 13
	373 × 141 (251)	4.5e − 4	0.1725	1.0e − 13
PowerFLOW	159,060 (828)	1.2e − 3	0.1717	0.227e − 3
	418,800 (1275)	7.8e − 4	0.1807	−0.211e − 3
CFL3D	257 × 65	–	0.1762	0.115e − 6
	373 × 141	1.2e − 4	0.1741	−0.538e − 5

Table 3
Grid convergence study of NACA0012

	Resolution (on wing)	Δx_{min}	Grid size ratio	$C_{\text{p,s}}$
A	249 × 95 (167)	6.7e − 4	0.67	1.3198
B	323 × 123 (217)	5.2e − 4	0.87	1.2606
C	373 × 141 (251)	4.5e − 4	1.00	1.2389

is not 2 is used here. If we assume the exact solution as Φ, and solution on grid size of h as Φ_h, $\Phi = \Phi_h + \alpha h^p$ should be satisfied where α is a constant and p is the order of the scheme. By comparing the results by three different grid resolutions, exact solution can be eliminated as

$$\Phi_{h_\text{A}} + \alpha(h_\text{A})^p = \Phi_{h_\text{B}} + \alpha(h_\text{B})^p, \tag{39}$$

$$\Phi_{h_\text{B}} + \alpha(h_\text{B})^p = \Phi_{h_\text{C}} + \alpha(h_\text{C})^p, \tag{40}$$

where $\Phi_{h_\text{A}}, \dots, \Phi_{h_\text{C}}$ and $h_\text{A},\dots,h_\text{C}$ are the solution and the grid size for grids A–C, respectively. Since the ratio of the grid size between the different grids is not 2, the accuracy p is calculated recursively by

$$\frac{(h_\text{B})^p - (h_\text{A})^p}{(h_\text{C})^p - (h_\text{B})^p} = \frac{\Phi_{h_\text{A}} - \Phi_{h_\text{B}}}{\Phi_{h_\text{B}} - \Phi_{h_\text{C}}}. \tag{41}$$

The pressure at the stagnation point is compared to calculate the accuracy of the scheme, which is also listed in Table 3. Since all of the grids have a grid point on the stagnation, the effect of the additional interpolation does not appear when comparing between the different grids. From this analysis, the accuracy of the current code is p = 1.97, which is nearly 2. This shows that current scheme has second order of accuracy.

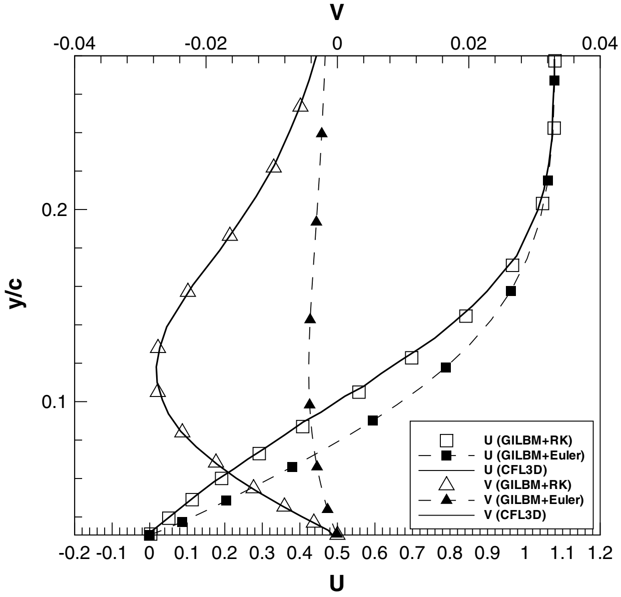

Fig. 7. Comparison of the boundary layer profile using different integration methods.

As mentioned in the previous section, the effect of the numerical method in order to integrate the contravariant velocity, Eq. (15), is validated with this problem. The boundary layer profile at 75% chord is shown in Fig. 7. For both U and V distributions, the results using Euler integration differ from the other two results. In particular, the U distribution shows that the result calculated using Euler integration makes the boundary layer thinner compared with that obtained using the two-step Runge–Kutta method and CFL3D. Consequently, the drag coefficient C_d obtained using Euler integration become 0.2650, which is much higher than that obtained in the previous studies (around 0.17). In order to maintain the accuracy around the wall boundary, where the grid is clustered, the integration method applied to Eq. (15) must be a second-order scheme.

5.3. Acceleration using the local time step method

In this section, the local time step method is applied to GILBM and the same problem is solved using the same grid. The objective of the present calculation is to (1) compare the results using the local and global time step and check whether both results are the same, (2) see the reduction in calculation time. The

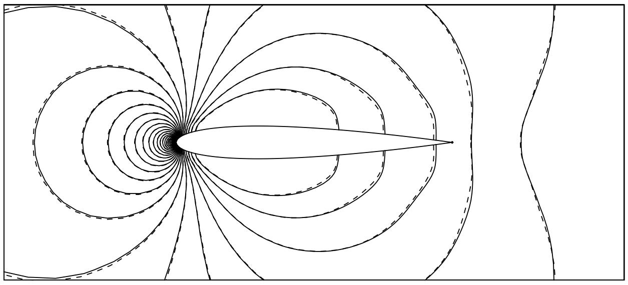

Fig. 8. Pressure distribution of NACA0012 flow with Re = 500, AOA = 0 [degree].

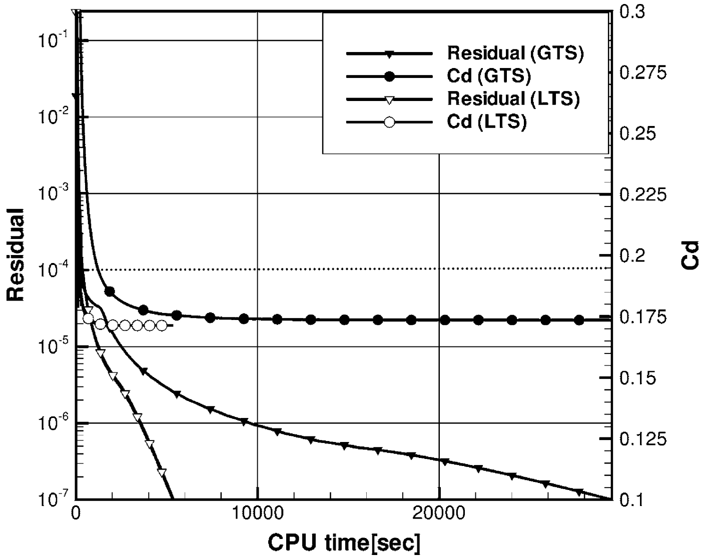

Fig. 9. Residual and drag coefficient history for NACA0012 calculation (257 × 65).

calculations are performed on two grids, 257 × 65 and 372 × 141, in order to investigate the effect of the grid dependency. The minimum grid size is defined as $0.01/\sqrt{Re}$. All of the results used in this section are calculated using single CPU PC (Pentium 4, 2 GHz) in order to compare the CPU time.

Fig. 8 shows the pressure contours using 257 × 65 grid. The pressure contour with line is the result calculated using local time step method, and that drawn with dashed line is the result calculated using global time step method. The excellent agreement is observed in the pressure contour.

Figs. 9 and 10 show the residual and C_d history of the present calculation. Figs. 9 and 10 correspond to the grid sized of 257 × 65 and 372 × 141, respectively. Every calculation is performed until Eq. (37) is satisfied and in both of the figures, C_d converge to constant value.

Table 4 lists the aerodynamic coefficients together with time step and CPU time required to satisfy the convergence criterion Eq. (37). There is a slight difference in C_d between the results by using global and local time step methods. For grid resolution of 257 × 65, the difference is 1.3%. However, this difference becomes smaller (0.3%) for grid resolution of 372 × 141. The lift coefficients C_l are maintained to almost 0 for all cases. From these facts, the results obtained using the local time step method is the same as that using the global time step method. Subsequently, we will focus on the CPU time listed in Table 4. The computational quantity increases over 20% per step when the local time step method is used. This increase is caused by the re-estimation procedure. Thus, the CPU time reduction ratio is smaller than the reduction in time step. Also, the reduction ratio for 257 × 65 grid is smaller than that of 372 × 141 grid, because

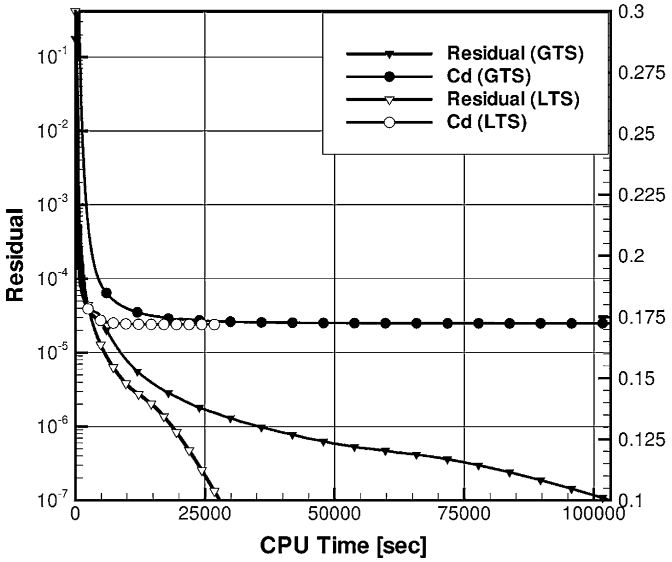

Fig. 10. Residual and drag coefficient history for NACA0012 calculation (372 × 141).

Table 4
Comparison of aerodynamic coefficients and computational time

Resolution:	257 × 65		372 × 141	
Method:	GTS	LTS	GTS	LTS
C_d	0.17357	0.17135	0.17247	0.17196
C_d	0.17357	0.17135	0.17247	0.17196
C_l	1.0e − 13	1.0e − 13	1.0e − 13	1.0e − 13
Time step	799,300 (1)	118,980 (0.15)	863,240 (1)	171,440 (0.20)
CPU time (s)	29,518 (1)	5354 (0.18)	103,268 (1)	27,866 (0.27)

the difference of minimum and maximum grid size is larger in the former case. The time step and CPU time are reduced by 70% to over 80% for the present cases.

6. Conclusions

The local time step method was applied to the Generalized form of ISLBM (GILBM), in order to obtain steady-state solutions with reduction in CPU time. The time step is defined locally from the local advection term stability.

First, code validation for global time step method was performed by solving flow around a cylinder at low Reynolds numbers. Our results show good agreement with previous studies. Two-dimensional flow around an airfoil was simulated in order to validate the present code. First, the global time step results were compared with the previous study in order to validate the present code. The excellent agreement was confirmed with the CFL3D results, and the importance of the estimation of the contravariant velocity is clarified. Second, the results obtained by the local time step method were compared with the global time step solutions, and its accuracy was confirmed. Also, the CPU times are reduced by 70–80% in all cases. Since the reduction in CPU time depends on the flow condition and grid configurations, it is not necessarily appropriate to suggest that the magnitude of the CPU time reduction is the same in other general cases. However, we have confirmed the effect of the local time step method with relatively small Reynolds numbers. As the Reynolds number becomes larger, the difference of minimum and maximum grid size becomes larger, and the reduction in CPU time is expected to become larger. Thus, the local time step method is an effective method on non-uniform mesh to accelerate the solution to the converged state in wide range of Reynolds numbers.

Appendix A. Distribution function including first-order non-equilibrium term

If we apply the Chapman–Enskog expansion to the lattice Boltzmann equation [28], the first order of non-equilibrium is described in non-dimensional form as

$$f_i^{(1)} = -\omega\Delta t\left(\frac{\partial f_i^{\mathrm{eq}}}{\partial t} + c_{i,\alpha}\frac{\partial f_i^{\mathrm{eq}}}{\partial x_\alpha}\right). \tag{A.1}$$

The equilibrium distribution function of LBM is described by Eq. (6). The time and spatial derivatives of the equilibrium distribution function of Eq. (A.1) are replaced by the derivatives of the macroscopic variables as

$$\frac{\partial f_i^{\mathrm{eq}}}{\partial t} = \frac{\partial f_i^{\mathrm{eq}}}{\partial \rho}\frac{\partial \rho}{\partial t} + \frac{\partial f_i^{\mathrm{eq}}}{\partial u_\alpha}\frac{\partial u_\alpha}{\partial t}, \tag{A.2}$$

$$\frac{\partial f_i^{\mathrm{eq}}}{\partial x_\alpha} = \frac{\partial f_i^{\mathrm{eq}}}{\partial \rho}\frac{\partial \rho}{\partial x_\alpha} + \frac{\partial f_i^{\mathrm{eq}}}{\partial u_\beta}\frac{\partial u_\beta}{\partial x_\alpha}, \tag{A.3}$$

and the equilibrium distribution function differentiated by the macroscopic variables are

$$\frac{\partial f_i^{\mathrm{eq}}}{\partial \rho} = \frac{1}{\rho}f_i^{\mathrm{eq}}, \tag{A.4}$$

$$\begin{aligned}\frac{\partial f_i^{\mathrm{eq}}}{\partial u_\beta} &= w_i\rho\left[\frac{3}{c^2}(c_{i,\beta} - u_\beta) + \frac{9(c_{i,\alpha}u_{i,\alpha})c_{i,\beta}}{c^4}\right] \\ &\simeq \frac{3U_{i,\beta}}{c^2}f_i^{\mathrm{eq}},\end{aligned} \tag{A.5}$$

where $U_{i,\beta} = c_{i,\beta} - u_\beta$. Terms of a higher order than u^2 are ignored. The equations of continuity and the conservation law of momentum eliminate the time derivative of the macroscopic variables as

$$\frac{\partial \rho}{\partial t} = -\frac{\partial \rho u_\alpha}{\partial x_\alpha}, \tag{A.6}$$

$$\frac{\partial u_\alpha}{\partial t} = -u_\beta \frac{\partial u_\alpha}{\partial x_\beta} - \frac{1}{\rho}\frac{\partial p}{\partial x_\alpha}. \tag{A.7}$$

Substituting Eqs. (A.2)–(A.7) into Eq. (A.1), first order of non-equilibrium is given from the macroscopic variables as

$$f_i^{(1)} = -\omega \Delta t f_i^{\mathrm{eq}} \left(\frac{3U_{i,\alpha}U_{i,\beta}}{c^2} - \delta_{\alpha\beta} \right) \frac{\partial u_\beta}{\partial x_\alpha}. \tag{A.8}$$

Finally, the non-dimensional distribution function, including first order of non-equilibrium is described by the macroscopic variables.

$$\begin{aligned} f_i &= f_i^{\mathrm{eq}} + f_i^{(1)} \\ &= f_i^{\mathrm{eq}} \left[1 - \omega \Delta t \left(\frac{3U_{i,\alpha}U_{i,\beta}}{c^2} - \delta_{\alpha\beta} \right) \frac{\partial u_\alpha}{\partial x_\beta} \right]. \end{aligned} \tag{A.9}$$

References

[1] S. Chen, G. Doolen, Lattice Boltzmann method for fluid flows, Annu. Rev. Fluid Mech. 161 (1998) 329.

[2] H. Chen, S. Chen, W.H. Matthaeus, Recovery of the Navier–Stokes equations using a lattice Boltzmann method, Phys. Rev. A 45 (1991) R5339.

[3] O. Fillipova, D. Hänel, Grid refinement for lattice-BGK models, J. Comput. Phys. 147 (1998) 219.

[4] O. Filippova, D. Hänel, Acceleration of lattice-BGK schemes with grid refinement, J. Comput. Phys. 165 (2000) 407.

[5] O. Fillipova, S. Succi, F. Mazzocco, C. Arrighetti, G. Bella, D. Hänel, Multiscale lattice Boltzmann schemes with turbulence modeling, J. Comput. Phys. 170 (2001) 812.

[6] C.-L. Lin, Y.G. Lai, Lattice Boltzmann method on composite grids, Phys. Rev. E 62 (2) (2000) 2219.

[7] D. Yu, R. Mei, W. Shyy, A multi-block lattice Boltzmann method for viscous fluid flows, Int. J. Numer. Meth. Fluids 39 (2002) 99.

[8] H. Chen, Volumetric formulation of the lattice Boltzmann method for fluid dynamics: basic concept, Phys. Rev. E 58 (3) (1998) 3955.

[9] G. Peng, H. Xi, C. Duncan, Finite volume scheme for the lattice Boltzmann method on unstructured meshes, Phys. Rev. E 59 (4) (1999) 4675.

[10] H. Xi, G. Peng, S.-H. Chou, Finite-volume lattice Boltzmann method, Phys. Rev. E 59 (5) (1999) 6202.

[11] R. Mei, W. Shyy, On the finite difference-based lattice Boltzmann method in curvilinear coordinate, J. Comput. Phys. 143 (1998) 426.

[12] X. He, L.-S. Luo, M. Dembo, Some progress in lattice Boltzmann method. Part 1. Nonuniform mesh grids, J. Comput. Phys. 129 (1996) 357.

[13] X. He, G. Doolen, Lattice Boltzmann method on curvilinear coordinate system: flow around a circular cylinder, J. Comput. Phys. 134 (1997) 306.

[14] Q. Kang, D. Zhang, S. Chen, Unified lattice Boltzmann method for flow in multiscale porus media, Phys. Rev. E 66 (2002) 056307.

[15] T. Lee, C.-L. Lin, A characteristic Galerkin method for discrete Boltzmann equation, J. Comput. Phys. 171 (2001) 336.

[16] F.H. Harlow, E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluids with free surface, Phys. Fluids 8 (1965) 2182.

[17] J.D. AnderSon Jr., Computational Fluid Dynamics;The Basics with Applications, McGraw-Hill, New York, 1995.

[18] G. McNamara, G. Zanetti, Using the lattice Boltzmann equation to simulate lattice gas automata, Phys. Rev. Lett. 61 (1988) 2332.

[19] Q. Zou, S. Hou, S. Chen, G.D. Doolen, An improved incompressible lattice Boltzmann model for time-independent flows, J. Stat. Phys. 81 (1995) 35.

[20] T. Lee, C.-L. Lin, An Eulerian description of the streaming process in the lattice Boltzmann equation, J. Comput. Phys. 185 (2003) 445.

[21] F. Nieuwstadt, H.B. Keller, Viscous flow past circular cylinders, Comput. Fluids 1 (1973) 59.

[22] S.C.R. Dennis, G.Z. Chang, Numerical solutions for steady flow past a circular cylinder at Reynolds number up to 100, J. Fluid. Mech. 42 (1980) 471.

[23] H. Abbot, A.E. Von Doenhoff, Theory of Wing Sections, Including a Summary of Airfoil Data, Dover, New York, 1959.

[24] R.A. Shock, S. Mallick, H. Chen, V. Yakhot, R. Zhang, Recent results on two-dimensional airfoils using a lattice Boltzmann-based algorithm, J. Aircraft 39 (3) (2002) 434.

[25] D.P. Lockard, L. Luo, S.D. Milder, B.A. Singer, Evaluation of PowerFLOW for aerodynamic applications, J. Stat. Phys. 107 (2002) 423.

[26] J.H. Ferziger, M. Peric, Computational Methods for Fluid Dynamics, third ed., Springer, Berlin, 2002.

[27] P.J. Roache, Perspective: A method for uniform reporting of grid refinement studies, J. Fluid Eng. 116 (1994) 405.

[28] W.G. Vincenti, C.H. Kruger, Introduction to Physical Gas Dynamics, Wiley, New York, 1965, p. 375.