Article

Recalibration of LBM Populations for Construction of Grid Refinement with no Interpolation.

Arseniy Berezin [1,2]*, Anastasia Perepelkina [2], Anton Ivanov [2] and Vadim Levchenko[2]

[1] National Research Nuclear University "MEPhI", 31, Kashirskoe Highway, Moscow, 115409 Russia
[2] Keldysh Institute of Applied Mathematics, Miusskaya sq. 4, 125047 Russia
* Correspondence: ArsenBRS@mail.ru

Abstract: Grid refinement is used to save computing costs while preserving high precision of fluid simulation. In the Lattice Boltzmann Method, grid refinement often uses interpolated values. We found a method, where interpolation in space and time is not required. For this purpose, we used the moment matching condition and rescaling the non-equilibrium part of populations, and developed the recalibration procedure that allows to transfer information between different LBM stencils in the simulation domain. Then, we build non-uniform lattice which uses stencils with different shapes on the transition. The resulting procedure is verified on the 2D Poisselle flow and the advected vortex benchmark. In prospect, grids with adaptive geometry can be built with the use of the proposed method.

Keywords: Lattice Boltzmann Method; Grid refinement; non-uniform grid; moment matching

1. Introduction

Lattice Boltzmann Method (LBM) is one of the state of the art methods for computational fluid dynamics. It is a curious case of how the cellular automata application [1] had evolved to model fluids, and subsequently was proven to be a rigorous numerical scheme for integrating the kinetic Boltzmann equation with the use of Gauss cubatures [2]. From the lattice gas stage of the evolution of the method, it inherited the use of rectangular meshes, which are more common in LBM simulations rather than in other hydrodynamic schemes, such as finite element and discontinuous Galerkin methods [3].

Compared to other methods in computational fluid dynamics (CFD), LBM stands out as a lightweight, efficient method with high locality and many options for parallelism [4,5]. The downside is the limited fluid parameter range where LBM, in its classical formulation, works. It is known to lack Galilean invariance, be working in the range of low Mach numbers, and for isothermal flows. Even in that range, many applications are found, such as mesoscale modeling in additive manufacturing [6]. Moreover, with high popularity of the method, many advanced variations were developed for compressible flows, thermal flows and additional physics, and LBM is used in almost every area of CFD.

In this work, we study the adaptive meshes for LBM. In a simulation of a model with several characteristic scales, where both large-scale background flow and small-scale flow variations have to be reproduced, one can use finer mesh for the areas where it is required, and save the computational cost by coarsening the mesh in the area with lower gradients of fluid properties. Adaptive meshes are also used to cover the areas with fine geometry features, such as thin channels. In the flow around the geometry with high curvature, or flat boundary that is not parallel to the Cartesian axes, the rectangular mesh has to be either sufficiently fine to resolve it, or replaced by a mesh that is conformal with the fluid-solid boundaries in the model. Since the classical LBM is constructed for rectangular meshes with a mesh step and time step equal to unity in non-dimensional units, in all these cases, there is an issue of information transfer between different mesh resolutions, different mesh geometries, and an issue of writing LBM for non-rectangular mesh.

 doi:10.20944/preprints202305.0015.v1

Over the years, many researchers have solved the issues by supplementing LBM with some other numerical method. When mesh is refined in some regions, the information is transferred by interpolation in space and/or time. There are two ways of refining the lattice: node-based and cell-based [7,8]. In a node-based approach, a node is added in between each pair of cells. In a cell-based approach, a lattice node is interpreted as a cubic cell, and each cell is subdivided into smaller cubes.

In the node-based methods [7,9–11], the order of interpolation can be increased by using more nodes in the reconstruction of the populations [11,12], or by the use of the information on fluid moments to increase the accuracy of the relevant quantities only [13].

Cell-based methods [8,14] have the mass-conserving property. However, in the conventional method [8], the information transfer from coarse to fine mesh, which is often referred to as 'explode', is, essentially, an interpolation with a polynomial of the order zero: the values are just copied to the nearest cells. An extension with improved accuracy was proposed [7,14]. The mass-conserving method is used in software like PARAMESH [15], waLBerla [16].

There are other methods to modify LBM so as to work with non-uniform meshes, such as finite difference LBM [17–19], finite volume and finite element methods [20–24] which can adapt LBM to curvilinear, tetrahedral and unstructured mesh geometries, Taylor series expansion and least-squares methods [25].

Since in the areas with high space resolution finer time step is desired as well, interpolation in time is also used, however, it can be avoided [26,27].

The motivation of the current study is based on the following observation. As a numerical scheme for the kinetic Boltzmann equation, LBM uses a set of discrete velocities to compute moments of the particle distribution function with Gauss quadrature rules. Various sets can be chosen as long as there are enough points in the quadrature to integrate polynomials of a given order [28]. The discretization of the Boltzmann equation in space-time is performed by integration in the direction of the characteristic [29]. The integration along the characteristic is first order, but, with a change of variables, the second order scheme is obtained. The velocity sets are chosen to form a uniform lattice in space, so that the streaming in the direction of the characteristic moves the particle from one lattice node to another.

What if we choose an arbitrary lattice, that fits the spatial and temporal scales and the geometry of our model, and choose a velocity set for each node in a way that every node is connected to its neighbors? If it is possible, than no other scheme except for the basic LBM is introduced in the model, interpolation is avoided, and the order of accuracy depends only on the proper LBM construction.

Methods, where LBM adapts to a stretched [30] or non-rectangular [31] mesh exist. The fact that a composition of different stencil was not used for adaptive grids before is due to the following issue. A lattice with varied geometry requires different discrete velocity sets in some nodes, and a discrete particle distribution function corresponds to each velocity. These discrete particle distribution functions are the populations that are updated in the LBM. We come to a situation where, in general, there are nodes in which the numerical scheme is constructed differently and operates on a different number of populations, and it is not known how these population sets can be converted into one another.

A well-known method of rescaling was used in the original grid refinement method [9], and improved in further works [32]. This was motivated by the change in the value of the time step, which required change in the collision parameter that controls the viscosity, and, in turn, the non-equilibrium part of the distribution functions. The change in the quadrature rule requires further transformation.

Our solution for the issue is motivated by a success in another area of advanced LBM variations. In [33], a revolutionary method of variable rescaling was introduced for modeling high-Mach flows. Even with classical LBM collision operators and no other modifications, flows with Mach numbers over 10^3 were reproduced. The work had put a start for many new development for complex flow simulations. The scheme was extended

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 May 2023 doi:10.20944/preprints202305.0015.v1

and applied for various compressible flows [34–39] Recalibrating the populations to the flow velocity temperature for the collision step is known to improve stability [40]

Here we propose to use this method of rescaling for construction of non-uniform grids. Based on the moment matching condition from [33], we develop a population recalibration method that allows to perform streaming between LBM nodes in which the numerical parameters used in construction of LBM method (time and spatial scales, and the velocity set) are different. In general, we identify the numerical parameters that are used to discretize the Boltzmann equations, and find a method to convert a set of LBM populations from a LBM node to an LBM node where LBM was constructed with different parameters.

In the following text, in Section 2, we remind the construction of the LBM method, and what is required to recalibrate LBM variables from one set to another; in Section 3, we propose an algorithm for LBM simulation on a non-uniform grid with the use of the proposed recalibration; in Section 4, we evaluate the resulting method on the trial problems, in Section 5, we discuss the impact of the results.

2. Theoretical Background

2.1. Lattice Boltzmann Method and its Parameters

The kinetic Boltzmann equation is

$$\frac{\partial f}{\partial t} + \boldsymbol{\xi}\frac{\partial f}{\partial \boldsymbol{r}} = \Omega, \tag{1}$$

where $f(t, \boldsymbol{r}, \boldsymbol{\xi})$ — is a single particle distribution function, and its arguments are time, spatial coordinate and particle velocity, and Ω is the collision term.

Macroscopic flow parameters (density $\rho(t, \boldsymbol{r})$, flow speed $\mathbf{u}(t, \boldsymbol{r})$, and temperature $T(t, \boldsymbol{r})$) are the moments of $f(t, \boldsymbol{r}, \boldsymbol{\xi})$:

$$\rho = \int\limits_{\mathbb{R}^D} f d^D\xi, \qquad \rho\mathbf{u} = \int\limits_{\mathbb{R}^D} \boldsymbol{\xi} f d^D\xi, \qquad \rho(u^2 + DT) = \int\limits_{\mathbb{R}^D} \xi^2 f d^D\xi, \tag{2}$$

where D — is the number of spatial dimensions.

In the equilibrium, the collision term is zero, and the solution of (1), which satisfies (2), is

$$f^{\mathrm{eq}}(\boldsymbol{\xi}) = \frac{\rho}{(\sqrt{2\pi}\xi_0)^D} e^{-(\boldsymbol{\xi}-\mathbf{u})^2/2\xi_0^2}, \qquad T = \xi_0^2. \tag{3}$$

In LBM, the integration in (2) is performed numerically. Taking into account the solution (3), the integration kernel is

$$\omega(\xi) = \frac{1}{(\sqrt{2\pi}\xi_0)^D} e^{-\xi^2/2\xi_0^2}. \tag{4}$$

Let us take a quadrature rule with the order of accuracy equal to n

$$\int\limits_{\mathbb{R}^D} g(\boldsymbol{\xi}) d^D\xi = \int\limits_{\mathbb{R}^D} \frac{g(\boldsymbol{\xi})}{\omega(\xi)} \omega(\xi) d^D\xi = \left/ \begin{array}{c} \boldsymbol{\xi}=\xi_0\mathbf{v} \\ d^D\xi = \xi_0^D d^D v \end{array} \right/ = \frac{1}{(2\pi)^{D/2}} \int\limits_{\mathbb{R}^D} \frac{g(\xi_0\mathbf{v})}{\omega(\xi_0 v)} e^{-v^2/2} d^D v \simeq \sum_i \frac{w_i g(\mathbf{c}_i)}{\omega(c_i)}, \tag{5}$$

where $\mathbf{c}_i = \xi_0\mathbf{v}_i$, $\mathbf{v}_i$ are the quadrature points, and w_i are the corresponding weights. The expression is exact when $g(\boldsymbol{\xi})$ is a polynomial of the order no higher than n.

With (2) and (5), we find

$$\rho = \sum_i \frac{w_i f(t, \boldsymbol{r}, \mathbf{c}_i)}{\omega(c_i)} = \sum_i f_i, \tag{6}$$

where the discrete populations f_i are defined as

$$f_i(t, \boldsymbol{r}) \equiv \frac{w_i f(t, \boldsymbol{r}, \mathbf{c}_i)}{\omega(c_i)} = (\sqrt{2\pi}\xi_0)^D w_i f(t, \boldsymbol{r}, \mathbf{c}_i) e^{c_i^2/2\xi_0^2}. \tag{7}$$

Other moments are obtained in a similar fashion:

$$\rho \mathbf{u} = \sum_i f_i \mathbf{c}_i, \qquad \rho(u^2 + DT) = \sum_i f_i c_i^2. \tag{8}$$

The equilibrium discrete populations are

$$f_i^{\mathrm{eq}} = \rho w_i e^{(2\mathbf{c}_i \cdot \mathbf{u} - u^2)/2\xi_0^2}. \tag{9}$$

The standard polynomial representation of f_i^{eq} [29] can be obtained by expanding this expression in the Taylor series:

$$f_i^{\mathrm{eq}} = \rho w_i \left(1 - \frac{u^2}{2\xi_0^2} + \frac{\mathbf{c}_i \cdot \mathbf{u}}{\xi_0^2} + \frac{(\mathbf{c}_i \cdot \mathbf{u})^2}{2\xi_0^4}\right). \tag{10}$$

By inserting (10) and $T = \xi_0^2$ in (6), (8) the moments are obtained exactly if the chosen quadrature rule is at least of the fourth order of accuracy.

In this derivation, ξ_0 has been introduced as a scheme parameter in (4). It allows to scale $\mathbf{c}_i$, therefore, it provides a connection between time and space scales and controls the lattice geometry. Taking the expression for f^{eq} into account, it is also associated with the model temperature and the speed of sound. In the current work, athermal LBM is considered, and ξ_0 remains a numerical parameter in the scheme construction. For athermal LBM, the 5-th order quadrature is enough [28].

The LBM equations are obtained by letting $\boldsymbol{\xi} = \mathbf{c}_i$ in (1) and multiplying by $w_i/\omega(c_i)$ for each i:

$$\frac{\partial f_i}{\partial t} + \mathbf{c}_i \frac{\partial f_i}{\partial \boldsymbol{r}} = \Omega_i, \qquad \Omega_i \equiv \frac{w_i}{\omega(c_i)} \Omega|_{\boldsymbol{\xi}=\mathbf{c}_i}. \tag{11}$$

In this work, the collision operator is taken in the Bhatnagar-Gross-Krook (BGK) form [41,42]:

$$\Omega = \frac{f^{\mathrm{eq}} - f}{\tau}, \qquad \Omega_i = \frac{f_i^{\mathrm{eq}} - f_i}{\tau}, \tag{12}$$

where τ is the collision parameter that controls the relaxation rate of populations and the fluid viscosity.

Let us take a uniform lattice with mesh step Δx. Then, (11) is discretized on a time step Δt in two successive steps:

- streaming: $$\frac{\partial f_i}{\partial t} + \mathbf{c}_i \frac{\partial f_i}{\partial \boldsymbol{r}} = 0 \quad \Rightarrow$$
$$f_i(t + \Delta t, \boldsymbol{r} + \Delta \boldsymbol{r}) - f_i(t, \boldsymbol{r}) = 0, \quad \Delta \boldsymbol{r} \equiv \mathbf{c}_i \Delta t; \tag{13}$$

- collision: $$\frac{\partial f_i}{\partial t} = \frac{f_i^{\mathrm{eq}} - f_i}{\tau} \quad \Rightarrow$$
$$f_i(t + \Delta t, \boldsymbol{r}) - f_i(t, \boldsymbol{r}) = \frac{\Delta t}{\tau}\left(f_i^{\mathrm{eq}}(t, \boldsymbol{r}) - f_i(t, \boldsymbol{r})\right). \tag{14}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 May 2023 doi:10.20944/preprints202305.0015.v1

In (13), $\Delta \boldsymbol{r}$ should point from one node to another, therefore, every velocity in the set $\{\mathbf{c}_i\}$ should be scaled accordingly. The ξ_0 parameter is used to scale $\{\mathbf{c}_i\}$. For example, the widely used D2Q9 set, which is

$$\left\{\mathbf{v}_i^{\mathrm{D2Q9}}\right\} = \Big\{(0,0),\ (0,\pm\sqrt{3}),\ (\pm\sqrt{3},0),\ (\pm\sqrt{3},\pm\sqrt{3})\Big\}, \tag{15}$$

corresponds to $\xi_0 = \Delta x/(\sqrt{3}\Delta t)$.

Thus, according to (13), (14), LBM method consists of the two steps: streaming from one node to another in the direction of $\mathbf{c}_i$, and a local collision operation.

Let us define the LBM stencil as a complete set of scheme parameters that uniquely define LBM streaming: quadrature points $\{\mathbf{v}_i\}$ and weights, scaling coefficient ξ_0, time step Δt. We denote the stencil by inserting these parameters in the brackets, i.e. $DNQM(\Delta t,\ \xi_0^2)$. The common D2Q9 LBM stencil (15) with $\Delta t = \Delta x = 1$ is denoted as D2Q9$(1, 1/3)$ hereafter.

The fluid viscosity ν is derived with the Chapman-Enskog analysis as [29]:

$$\nu = \xi_0^2\left(\tau - \frac{\Delta t}{2}\right). \tag{16}$$

Finally, given a lattice and a fluid with viscosity equal to ν, to use classical athermal LBM with BGK collision, we have to ensure that each vector $\xi_0\mathbf{v}_i\Delta t$ points exactly from one lattice node to another, the quadrature set is enough to integrate with the 5th order of accuracy, and (16) is satisfied.

2.2. Recalibration of Populations

In LBM with adaptive grids, lattice nodes may have varying stencils. For the information exchange between LBM nodes that operate with different LBM stencils, we need to know how the population sets are converted into one another. To construct the method of recalibration of populations from one stencil to another, let us find out what changes are required with the change of Δt, ξ_0, and the set of quadrature points.

Let us remark that we allow the stencil, and, thus, $\mathbf{c}_i$, vary in time and space, and the derivation of (11) requires inserting $\omega(\mathbf{c}_i)$ under the derivative [43]. There is no contradiction here. In each node, we construct LBM as if it is on a uniform grid. In the following, we find out how the information is converted between LBM constructed for different stencils.

2.2.1. Recalibration with Δt

According to (13), (14), the equations of LBM are:

$$f_i(t+\Delta t, \boldsymbol{r}+\mathbf{c}_i\Delta t) = f_i(t,\boldsymbol{r}) + \frac{\Delta t}{\tau}\Big(f_i^{\mathrm{eq}}(t,\boldsymbol{r}) - f_i(t,\boldsymbol{r})\Big). \tag{17}$$

In other words, the populations are collided in $(t,\boldsymbol{r})$ and travel to $(t+\Delta t, \boldsymbol{r}+\mathbf{c}_i\Delta t)$.

To find the relationship between population which travel from the node $(t,\boldsymbol{r})$ according to different stencils, let us use Chapman-Enskog analysis. Taking into account only the term that is linear in the Knudsen number ϵ,

$$f_i \simeq f_i^{\mathrm{eq}} + \epsilon f_i^{(1)}, \qquad f_i^{(1)} = -\tau\Big(\partial_t^{(1)} f_i^{\mathrm{eq}} + \mathbf{c}_i\partial_{\boldsymbol{r}}^{(1)} f_i^{\mathrm{eq}}\Big), \qquad \partial^{(1)} = \frac{\partial}{\epsilon}, \tag{18}$$

and

$$f_i \simeq f_i^{\mathrm{eq}} - \tau\Big(\partial_t f_i^{\mathrm{eq}} + \mathbf{c}_i\partial_{\boldsymbol{r}} f_i^{\mathrm{eq}}\Big) = f_i^{\mathrm{eq}} - \tau D f_i^{\mathrm{eq}}, \qquad D \equiv \partial_t + \mathbf{c}_i\partial_{\boldsymbol{r}}. \tag{19}$$

Let us consider conversion between stencils with differ only in Δt: D2Q$M(\Delta t_c,\ \xi_0)$ and D2Q$M(\Delta t_f,\ \xi_0)$. Let us take a coarse stencil with time step Δt_c and relaxation parameter τ_c, and a fine stencil with Δt_f and τ_f. By inserting (19) into (17) we get

$$f_{i,c}(t+\Delta t_c, \boldsymbol{r}+\mathbf{c}_i\Delta t_c) - f_i^{\mathrm{eq}}(t,\boldsymbol{r}) = (\Delta t_c - \tau_c) D f_i^{\mathrm{eq}}(t,\boldsymbol{r}). \tag{20}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 May 2023 doi:10.20944/preprints202305.0015.v1

and a similar relation for the fine stencil. By equating $Df_i^{\mathrm{eq}}(t,\boldsymbol{r})$ in the expression for the fine and the coarse stencils, we get a well-known relationship [9]

$$\frac{f_{i,c}(t+\Delta t_c,\boldsymbol{r}+\mathbf{c}_i\Delta t_c)-f_i^{\mathrm{eq}}(t,\boldsymbol{r})}{f_{i,f}(t+\Delta t_f,\boldsymbol{r}+\mathbf{c}_i\Delta t_f)-f_i^{\mathrm{eq}}(t,\boldsymbol{r})}=\frac{\Delta t_c-\tau_c}{\Delta t_f-\tau_f}, \tag{21}$$

and the relationship between τ_c and τ_f is obtained be requiring the invariance of fluid viscosity (16) .

$$\nu=\xi_0^2\left(\tau_c-\frac{\Delta t_c}{2}\right)=\xi_0^2\left(\tau_f-\frac{\Delta t_f}{2}\right)\qquad\Rightarrow\qquad\frac{2\tau_c-\Delta t_c}{2\tau_f-\Delta t_f}=1. \tag{22}$$

2.2.2. Recalibration with both Δt and ξ_0

Let us take two stencils $\mathrm{D}NQM(\Delta t_f,\,\xi_{0,f}^2)$ and $\mathrm{D}NQM(\Delta t_c,\,\xi_{0,c}^2)$ which correspond to the same quadrature $\{\mathbf{v}_i\}$, but different ξ_0, Δt, τ. We have

$$\{\mathbf{c}_{i,f}\}\neq\{\mathbf{c}_{i,c}\},\qquad D_f\neq D_c,\qquad f_{i,f}^{\mathrm{eq}}\neq f_{i,c}^{\mathrm{eq}}, \tag{23}$$

and, instead of (22),

$$\nu=\xi_{0,c}^2\left(\tau_c-\frac{\Delta t_c}{2}\right)=\xi_{0,f}^2\left(\tau_f-\frac{\Delta t_f}{2}\right)\qquad\Rightarrow\qquad\frac{2\tau_c-\Delta t_c}{2\tau_f-\Delta t_f}=\frac{\xi_{0,f}^2}{\xi_{0,c}^2}. \tag{24}$$

Taking (23) into account, (20) leads to the following expression:

$$\frac{f_{i,c}(t+\Delta t,\boldsymbol{r}+\mathbf{c}_{i,c}\Delta t)-f_{i,c}^{\mathrm{eq}}(t,\boldsymbol{r})}{f_{i,f}(t+\Delta t,\boldsymbol{r}+\mathbf{c}_{i,f}\Delta t)-f_{i,f}^{\mathrm{eq}}(t,\boldsymbol{r})}=\frac{(\Delta t_c-\tau_c)D_cf_{i,c}^{\mathrm{eq}}}{(\Delta t_f-\tau_f)D_ff_{i,f}^{\mathrm{eq}}}. \tag{25}$$

Let us find how Df_i^{eq} depends on the stencil. Let us express it through the derivatives of the physical quantities ρ, $\mathbf{u}$. From (10) and (19), we get

$$\begin{aligned}Df_i^{\mathrm{eq}}=w_i\epsilon\Bigg(&\partial_t^{(1)}\rho-\frac{1}{2\xi_0^2}\partial_t^{(1)}\rho u^2+\frac{c_{i,\beta}}{\xi_0^2}\partial_t^{(1)}\rho u_\beta+\frac{c_{i,\beta}c_{i,\gamma}}{2\xi_0^4}\partial_t^{(1)}\rho u_\beta u_\gamma+\\&c_{i,\alpha}\partial_\alpha^{(1)}\rho-\frac{c_{i,\alpha}}{2\xi_0^2}\partial_\alpha^{(1)}\rho u^2+\frac{c_{i,\alpha}c_{i,\beta}c_{i,\gamma}}{2\xi_0^4}\partial_\alpha^{(1)}\rho u_\beta u_\gamma+\frac{c_{i,\alpha}c_{i,\beta}}{\xi_0^2}\partial_\alpha^{(1)}\rho u_\beta\Bigg).\end{aligned} \tag{26}$$

Time derivatives in (26) are replaced by the space derivatives with the use of the conservation laws

$$\partial_t^{(1)}\rho+\partial_\alpha^{(1)}(\rho u_\alpha)=0,\qquad\partial_t^{(1)}(\rho u_\alpha)+\partial_\gamma^{(1)}\Pi_{\alpha\gamma}^{\mathrm{eq}}=0, \tag{27}$$

where

$$\Pi_{\alpha\gamma}^{\mathrm{eq}}=\rho u_\alpha u_\gamma+\rho\xi_0^2\delta_{\alpha\gamma}. \tag{28}$$

The time derivative is expanded,

$$\partial_t^{(1)}\rho u_\beta u_\gamma=u_\beta\partial_t^{(1)}\rho u_\gamma+u_\gamma\partial_t^{(1)}\rho u_\beta-u_\beta u_\gamma\partial_t^{(1)}\rho. \tag{29}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 May 2023 doi:10.20944/preprints202305.0015.v1

Finally, with the use of (27) and (28), (26) becomes

$$Df_i^{\mathrm{eq}} = w_i\epsilon\left(-\rho\partial_\alpha^{(1)}u_\alpha + \frac{c_{i,\alpha}c_{i,\beta}}{\xi_0^2}\rho\partial_\alpha^{(1)}u_\beta - \frac{c_{i,\beta}}{\xi_0^2}\partial_\gamma^{(1)}\rho u_\beta u_\gamma - \frac{c_{i,\alpha}}{2\xi_0^2}\partial_\alpha^{(1)}\rho u^2 + \frac{c_{i,\alpha}c_{i,\beta}c_{i,\gamma}}{2\xi_0^4}\partial_\alpha^{(1)}\rho u_\beta u_\gamma + \mathcal{O}(u^3)\right). \quad (30)$$

According to (18), here, we can use ∂ instead of $\partial^{(1)}$, and insert $\mathbf{c}_i = \xi_0\mathbf{v}_i$ to find how Df_i^{eq} depends on ξ_0.

$$Df_i^{\mathrm{eq}} = w_i\left(\underline{-\rho\partial_\alpha u_\alpha + v_{i,\alpha}v_{i,\beta}\,\rho\partial_\alpha u_\beta} - \frac{v_{i,\beta}}{\xi_0}\partial_\gamma\rho u_\beta u_\gamma - \frac{v_{i,\alpha}}{2\xi_0}\partial_\alpha\rho u^2 + \frac{v_{i,\alpha}v_{i,\beta}v_{i,\gamma}}{2\xi_0}\partial_\alpha\rho u_\beta u_\gamma + \mathcal{O}(u^3)\right). \quad (31)$$

Here the underlined terms do not depend on ξ_0. These terms are unchanged as long as the same quadrature rule is used. The following terms can be neglected in the low compressibility limit. Therefore, the recalibration expression is similar to (21):

$$\frac{f_{i,c}(t+\Delta t, \boldsymbol{r}+\mathbf{c}_{i,c}\Delta t) - f_{i,c}^{\mathrm{eq}}(t,\boldsymbol{r})}{f_{i,f}(t+\Delta t, \boldsymbol{r}+\mathbf{c}_{i,f}\Delta t) - f_{i,f}^{\mathrm{eq}}(t,\boldsymbol{r})} = \frac{\Delta t_c - \tau_c}{\Delta t_f - \tau_f}, \quad (32)$$

and the relationship between τ_f and τ_c is (24).

2.2.3. Recalibration with a change in quadrature

Let us construct the conversion between populations in D2Q$M_1(\Delta t, \xi_0^2)$ and in D2Q$M_2(\Delta t, \xi_0^2)$.

The conversions in (21), (32) are between the stencils of the same quadrature $\{\mathbf{v}_i\}$, but different τ. Therefore, these are the conversions between LBMs with different collision operators, which are essentially different LBM equations, and the conversions are giving the correct non-equilibrium part of the populations.

When the τ parameter remains the same, the moment matching condition [33–36] is used for conversion. It is required that the change in the LBM populations does not lead to the change in the physical moments:

$$\left(\sum_i c_{i,x}^p c_{i,y}^q f_i\right)\Big|_{\mathrm{D2Q}M_1(\Delta t,\xi_0^2)} = \left(\sum_i c_{i,x}^p c_{i,y}^q f_i\right)\Big|_{\mathrm{D2Q}M_2(\Delta t,\xi_0^2)}. \quad (33)$$

The number of equations in the linear system (33) depends on the order of approximation of the quadrature and its symmetries [28,29,44]. For athermal fluid physics, we require at least

$$p+q \le 5 \qquad p,q \in \mathbb{N}_0. \quad (34)$$

Additionally, an explicit relationship can be found for f_0 (which corresponds to the zero discrete velocity vector) with (7):

$$(f_0/w_0)|_{\mathrm{D2Q}M_1(\Delta t,\xi_0^2)} = (f_0/w_0)|_{\mathrm{D2Q}M_2(\Delta t,\xi_0^2)}. \quad (35)$$

Similar relationship can be written whenever stencils have a share discrete velocity.

Relations (33), (35) are a system of linear equations, that has to be solved to find the populations of one stencil, while the populations of another stencil are known. In general, the system can be underdetermined or inconsistent. For example, in the conversion from D2Q9 to D2Q15 we have to find 15 unknown populations from computable moments. In

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 May 2023 doi:10.20944/preprints202305.0015.v1

this case, we append the systems by equating relationships from (33) to the equilibrium moments [45].

2.2.4. Recalibration with the change of stencil

In total, in the recalibration of populations from D2Q$M_1(\Delta t_1, \xi^2_{0,1})$ to D2Q$M_2(\Delta t_2, \xi^2_{0,2})$ two conversions are applied.

First, the quadrature rule is fixed, and Δt, τ, ξ_0 are modified with (24), (32). This converts D2Q$M_1(\Delta t_1, \xi^2_{0,1})$ to D2Q$M_1(\Delta t_2, \xi^2_{0,2})$. Second, the quadrature is changed with fixed τ, ξ_0, and the stencil is converted to D2Q$M_2(\Delta t_2, \xi^2_{0,2})$. The two steps can be performed in any order. In any case, in the middle of the conversion, there is a virtual LBM stencil (here, D2Q$M_1(\Delta t_2, \xi^2_{0,2})$) for which neither streaming nor collision takes place.

3. Grid Refinement Interface without Interpolation

3.1. Grid Geometry

The recalibration method allows to construct a transition between grids with different time and space steps, if only we construct the stencils that point to/from the lattice points exactly in the uneven grid transition region.

Let us take a grid refinement boundary with coarse uniform grid ($\Delta t_c = \Delta x_c = 1$) on the left side and fine uniform grid ($\Delta t_f = \Delta x_f = 1/2$) on the right side (Fig. 1).

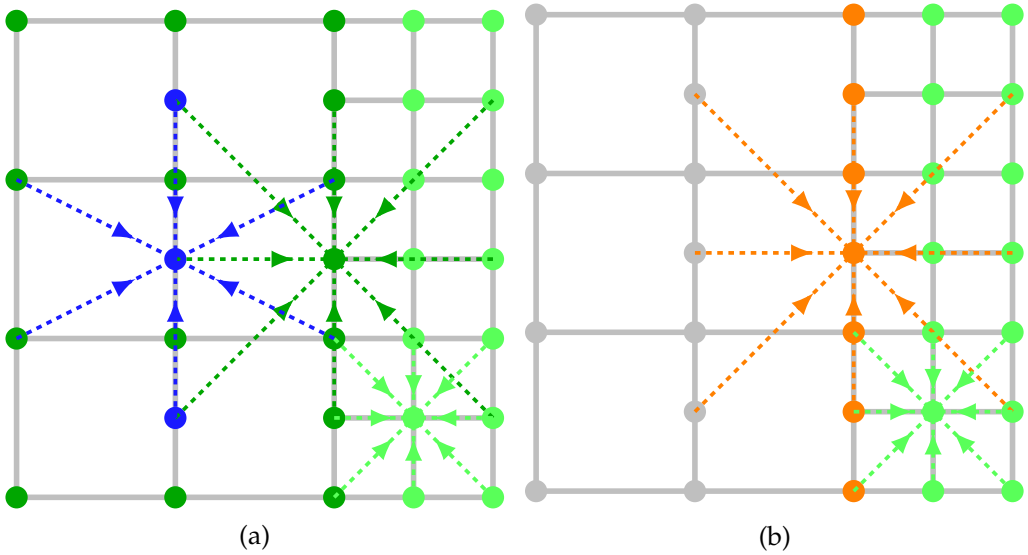


Figure 1. The scheme of the discrete velocity sets of the stencils near the grid refinement boundary, at the integer (a) and half-integer (b) time steps. Dark-green: D2Q9$(1, 1/3)$; light-green: D2Q9$(1/2, 1/3)$; blue: D2Q7$(1, 1/4)$; orange: D2Q9$(1/2, 4/3)$; gray: no streaming at half-integer steps.

LBM stencils can be used in the 'Pull' or 'Push' paradigm, which is relevant for the conversion purposes. In the 'Push' method, the populations are streamed from a node according to the stencil of that node. After the streaming, the populations in any one node may come from different stencils, and there is no full set of populations in some nodes. In the 'Pull' method, populations are converted into the stencil of the node before they are streamed into that node. The 'Pull' method is in the original work [33], and is used in the current study. In case of the uniform grids, no stencil conversions are used and the 'Pull' and 'Push' methods are equivalent. The difference of rescaling incoming and outgoing populations in refined grids supplemented with interpolation is reported in [32].

3.2. Stencils and Recalibration

Let us construct an LBM algorithm, in which coarse and fine grids use the classical LBM stencil, and the transition between grids do not require interpolated values, that is,

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 May 2023 doi:10.20944/preprints202305.0015.v1

all $\mathbf{c}_i$ point from one node to another. This is possible by introducing several transitional LBM stencils.

Let us use D2Q9$(1, 1/3)$ (dark-green in Fig. 1) and D2Q9$(1/2, 1/3)$ (light-green) for the coarse and the fine grids respectively. This leads to the classical unaltered D2Q9 LBM in the fine and coarse grids away from the boundary.

For the grid transition, the well-known quadrature derivation rules can be used [46]. D2Q15$(1, 25/38)$ (see Appendix A for the derivation) is a stencil of the 5th order which uses the existing nodes, thus, its can reproduce the moments of the equilibrium distribution at least as good as D2Q9 which is used away from the grid transition. We have also tried the D2Q7$(1, 1/4)$ (details in Appendix A). This stencil performs correct integration for less moments, but is more local. The tests with both stencils are reported in Section 4. In Fig. 1, D2Q7$(1, 1/4)$ is depicted, but D2Q15$(1, 25/38)$ can be used instead.

To perform grid transition, we use nodes with the D2Q7$(1, 1/4)$ (or D2Q15$(1, 25/38)$) stencil (blue in Fig. 1) on the integer time steps, and the D2Q9$(1/2, 4/3)$ on half-integer time steps (orange nodes in Fig. 1). This is just one of the many possibilities that can be constructed on the boundary transition.

Other configurations can be used depending on the requirements of the model.

When the introduced stencils are used in the 'Pull' paradigm, the streaming operation at a node requests populations from the neighboring nodes. At the node from where the population is requested from, the recalibration in the target stencil takes place. If it is a recalibration between different variations of D2Q9, (24) is used. If any other stencil is involved, the recalibration is performed in two steps according to Section 2.2.4.

$$\text{D2Q9}(\dots) \rightleftarrows \text{D2Q9}(1,\ 1/4) \rightleftarrows \text{D2Q7}(1,\ 1/4), \tag{36}$$

$$\text{D2Q9}(\dots) \rightleftarrows \text{D2Q9}(1,\ 25/38) \rightleftarrows \text{D2Q15}(1,\ 25/38), \tag{37}$$

Here D2Q9$(1, 1/4)$ and D2Q9$(1, 25/38)$ are the virtual stencils that are used only as an intermediate state in the conversion of populations.

3.3. Full Grid Transition Algorithm

The initial state $t = t_0$ corresponds to Fig. 1(a).

1. Perform streaming on the coarse grid $t_0 \to t_0 + \Delta t_c$ with the use of the stencils
   - (a) D2Q7$(1, 1/4)$ (or D2Q15$(1, 25/38)$) for the blue nodes;
   - (b) D2Q9$(1, 1/3)$ for dark green nodes. Here the incoming populations at the nodes which are exactly on the boundary are saved in a separate temporary buffer to be used in step 5, since the pre-streaming populations are still needed in the next step.
2. Perform streaming on the fine grid $t_0 \to t_0 + \Delta t_f$ (Fig 1(b)) with the use of the stencils
   - (a) D2Q9$(1/2, 1/3)$ at the light-green nodes;
   - (b) D2Q9$(1/2, 4/3)$ at the orange nodes.
3. Perform collision on the fine grid at the orange and light-green nodes (Fig 1(b)).
4. Perform second streaming $t_0 + \Delta t_f \to t_0 + \Delta t_c$ to the light-green nodes of the fine grid (Fig 1(a)).
5. Restore the values of the boundary nodes from the buffer.
6. Perform collision on all nodes according to the stencils in Fig 1(a).

4. Benchmarks

The proposed algorithm was implemented in C++ with the use of the Zipped Data Structure for Adaptive Mesh Refinement (ZAMR) [47] library in the Aiwlib [48] package. The library provides convenient tools to work with binary refined grids based on the Z-order curve traversal. It allows to perform operations on all nodes of uneven grids, and to set flags on the nodes where operations differ. With it, we mark different color of nodes in Fig. 1 with different flags and implement the described algorithm.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 May 2023 doi:10.20944/preprints202305.0015.v1

4.1. Poiseuille Flow

The stationary solution of the Navier-Stokes equation

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\boldsymbol{\nabla})\mathbf{u} = -\frac{1}{\rho}\boldsymbol{\nabla} p + \nu\Delta\mathbf{u} \tag{38}$$

in the $-H_x/2 \le x \le H_x/2, 0 \le y \le H_y$ domain with no-slip boundary in x and periodic boundary in y:

$$\mathbf{u}\left(-\frac{H_x}{2}, y\right) = \mathbf{u}\left(\frac{H_x}{2}, y\right) = \mathbf{0}, \qquad \mathbf{u}(x, y + H_y) = \mathbf{u}(x, y), \qquad \boldsymbol{\nabla} p = (0, -g), \tag{39}$$

subject to the pressure gradient

$$\boldsymbol{\nabla} p = (0, -g), \tag{40}$$

is

$$u_x = 0, \qquad u_y(x) = \frac{g H_x^2}{8\rho\nu}\left(1 - \frac{4x^2}{H_x^2}\right). \tag{41}$$

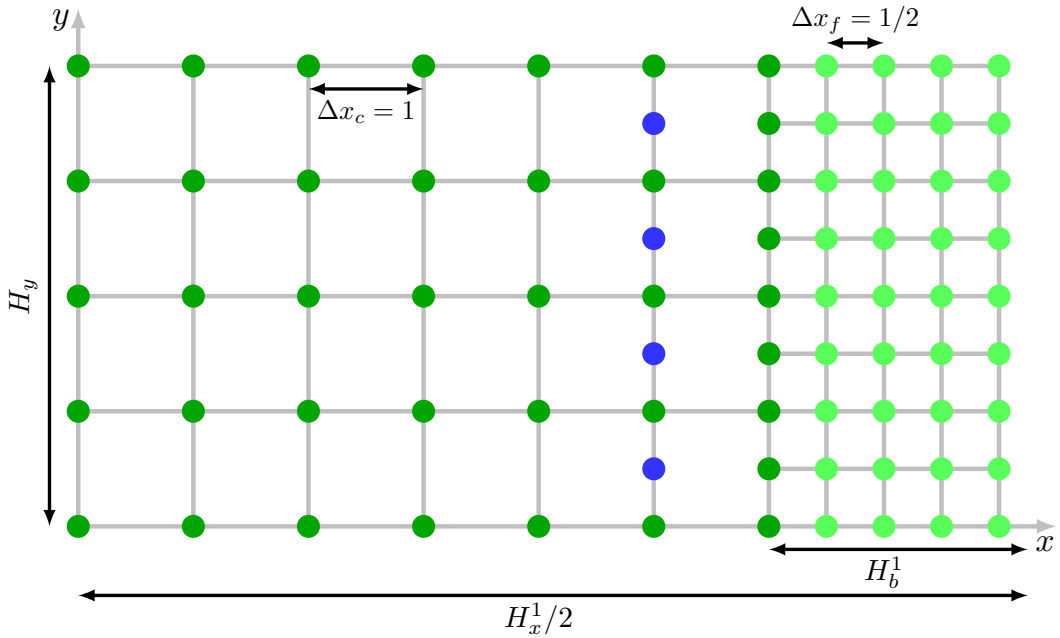


Figure 2. Grid geometry in the Poiseuille flow benchmark

The flow is modeled with the use of the bounce-back boundary conditions in x [29]. The pressure gradient is simulated with an additional term $\Delta u_y = g\tau/\rho$ in the flow velocity in the computation of equilibrium distribution [49] for the collision step. The initial state is equilibrium where

$$\rho = 1, \qquad u_x = u_y = 0. \tag{42}$$

The flow evolves to its equilibrium state (41).

In this benchmark, classical LBM with BGK collision and bounce-back boundary has a well-known problem. The effective channel width H_x can depend on the collision parameter τ [50]. In this benchmark we fix $\tau_f = (\sqrt{3}+2)\Delta t_f/4$ on the fine grid. Then the viscosity is $\nu = \sqrt{3}\Delta t_f/12$, and the boundary is considered to be at the distance $\Delta x_f/2$ from the boundary nodes [29].

We refine the grid near the no-slip boundaries. The sizes of the grid are

$$H_x^r = 2^{3+r}\Delta x_c + \Delta x_f, \qquad H_b^r = 2^r \Delta x_c + \frac{\Delta x_f}{2} \approx \frac{H_x^r}{8}, \qquad H_y = 4\Delta x_c. \tag{43}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 May 2023 doi:10.20944/preprints202305.0015.v1

Where r is the parameter that controls the grid resolution. With $r = 1$, half of the grid geometry is in Fig. 2.

The fine grid takes approximately a quarter of the domain. In our benchmarks, such grid geometry performed slightly better than if the grid is refined in the half of the simulation region.

To evaluate the order of approximation of the proposed scheme, the numerical error dependency on r is studied. With the change of r with fixed Δx, the effective channel width and velocity maximum increase. Thus, the error is normalized to $(H_x^0/H_x^r)^2$. Note that in evaluating the L_p norm in case of non-uniform grids, the contribution of nodes is scaled to the effective area of the node $dS(\text{node})$ to ensure correct numerical representation of the finite integral.

$$L_\infty(H_x^r) = \left(\frac{H_x^0}{H_x^r}\right)^2 \max_{\text{nodes}} |u_y^{r,\text{num}} - u_y^r|, \tag{44}$$

$$L_n(H_x^r) = \left(\frac{H_x^0}{H_x^r}\right)^2 \sqrt[n]{\frac{\sum\limits_{\text{nodes}} |u_y^{r,\text{num}} - u_y^r|^n dS(\text{node})}{\sum\limits_{\text{nodes}} dS(\text{node})}}, \qquad n \in \{1,2\}, \tag{45}$$

$$dS(\text{coarse}) = \Delta x_c^2, \qquad dS(\text{fine}) = \Delta x_f^2, \qquad dS(\text{border}) = (\Delta x_c^2 + \Delta x_f^2)/2, \tag{46}$$

The blue nodes in Fig 2 serve auxiliary purpose in the scheme and are not taken into account in the total error.

The benchmark results are reported in Fig. 3, 4. The results are compared to LBM simulation on a uniform fine grid (light green lines).

We observe that the proposed method of grid transition works correctly.

The D2Q7(1, 1/4) stencil performs better in the space time discretization, and gives the result almost as good as a uniform fine grid. However, the stencil does not reproduce high order velocity moments, and suffers from Galilean invariance errors. That is why its performance with grid refinement is more consistent with lower g (Fig. 4).

The discretization errors for the D2Q15(1, 25/38) are higher and show the first order of accuracy. The source of the error is yet to be found. The dependencies of this stencil in the space-time grid are the longest of all other stencils in the constructed scheme. The low angle of the space-time characteristics may cause such behavior.

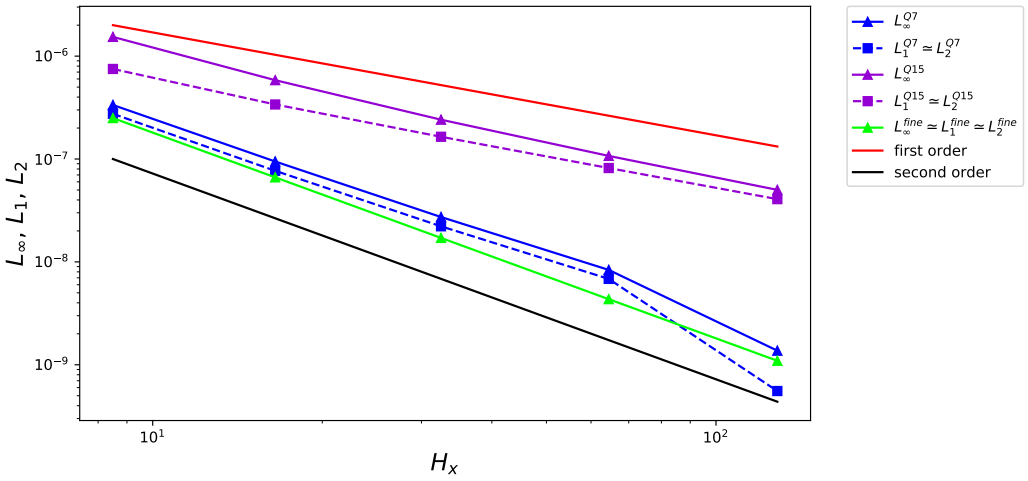


Figure 3. The L_{inf}, L_1 and L_2 errors for Poiseuille flow with $g = 1 \times 10^{-6}$, $\nu = \sqrt{3}/24$.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 May 2023 doi:10.20944/preprints202305.0015.v1

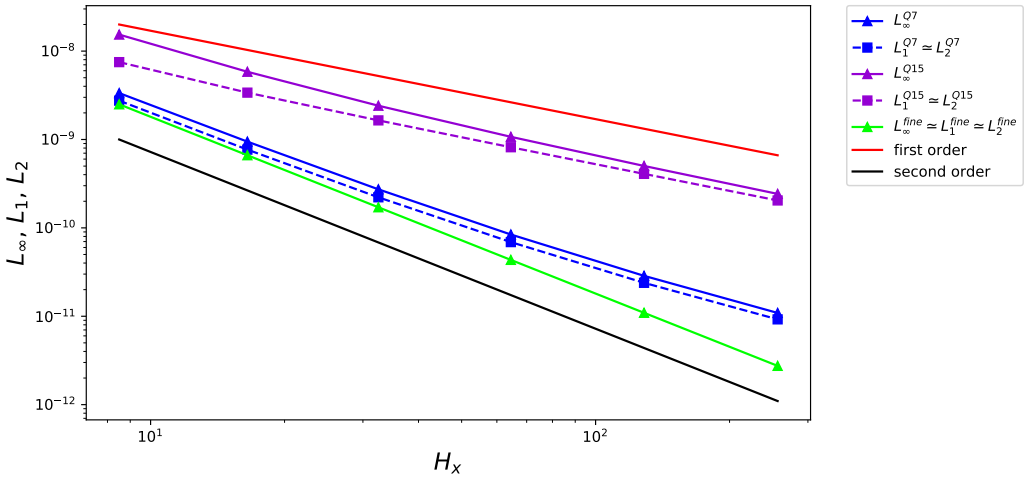


Figure 4. The L_{inf}, L_1 and L_2 errors for Poiseuille flow with $g = 1 \times 10^{-8}$, $\nu = \sqrt{3}/24$.

Let us compare the order of accuracy of our algorithm with the already known schemes for constructing non-uniform grids. When using stencil D2Q7$(1, 1/4)$, we get almost the second order of accuracy in each norm, which is more stable than both algorithms with initial-value problem (IVP) and boundary-value problem (BVP) types of interpolation in [26], and similar to cell-vertex and cell-centered with linear explosion hybrid-recursive regularized (HRR) algorithms in [7]. When using stencil D2Q15$(1, 25/38)$, we get the order of accuracy close to unity, which is resemble method with IVP interpolation in [26] and cell-centred with uniform explosion HRR algorithm in [7].

Let us note that the benchmark results can be improved by the use of advanced collision and boundary conditions [29,50]. Here we demonstrate only the impact of the proposed recalibration method, thus, the classical LBM with BGK collision is used.

In Fig. 5, the difference between numerical and theoretical solution is plotted vs the x coordinate. Here $r = 1$, $g = 1 \times 10^{-6}$, $\nu = \sqrt{3}/24$. It can be seen that the grid transition introduced an error an leads to the non-physical current in the x direction, and the error, indeed, appears on the grid boundary, and is not caused by the bounce-back boundary.

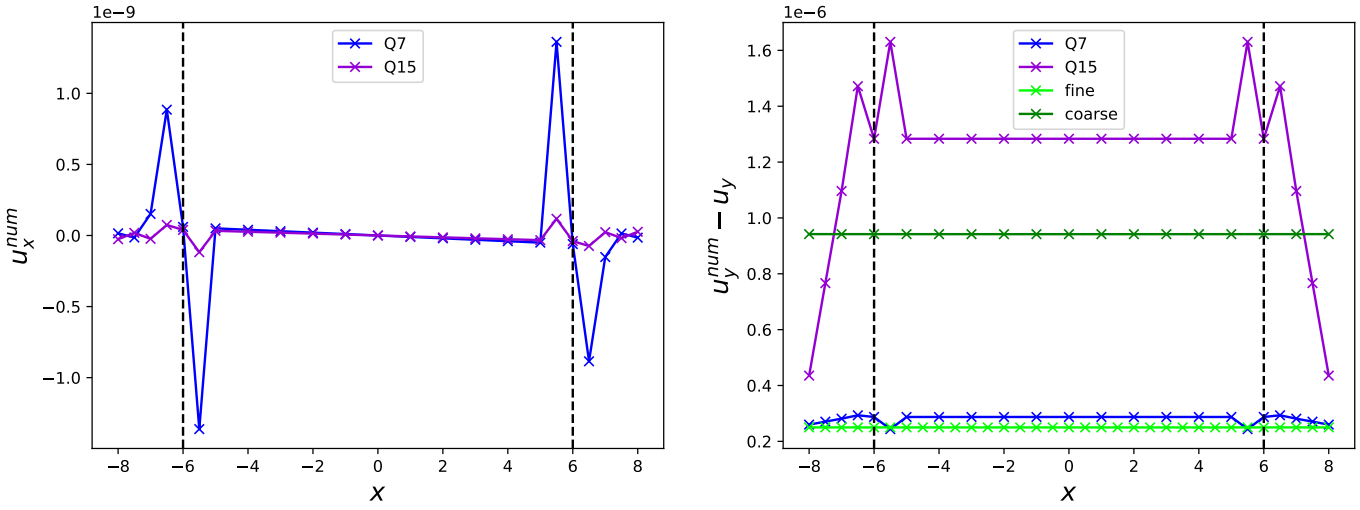


Figure 5. The absolute error of the Poiseuille flow solution vs the x coordinate. Here, $r = 1$, $g = 1 \times 10^{-6}$, $\nu = \sqrt{3}/24$.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 May 2023 doi:10.20944/preprints202305.0015.v1

4.2. Athermal vortex

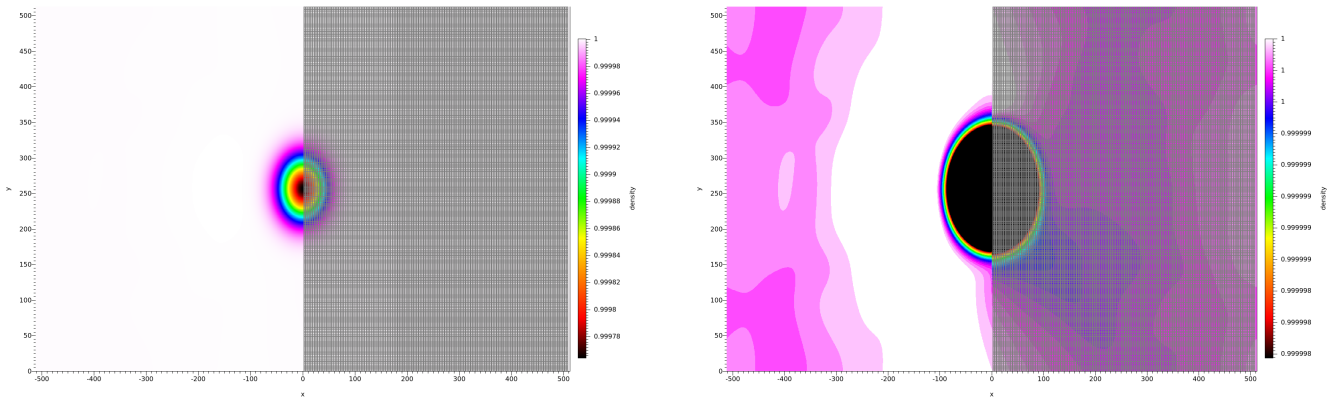
In the second benchmark, the dynamics of the advected athermal vortex [51,52] is modelled. The initial conditions are

$$u_\varphi(r)\big|_{t=0} = \frac{r\beta_0}{2\pi R}\exp\left(\frac{1-r^2/R^2}{2}\right), \quad \rho(r)|_{t=0} = \exp\left(-\frac{\beta_0^2}{8\pi^2 T}\exp\left(1-r^2/R^2\right)\right), \quad (47)$$

for the stationary vortex, and a constant u_0 is added to u_x to model advection. Here β_0 is the amplitude of the vortex rotation, R is the vortex radius, T is the lattice temperature. According to the Navier-Stokes solution, the vortex relaxes to uniform flow, and the relaxation rate is proportional to viscosity. In our benchmarks, we set lower viscosity and study the preservation of the vortex shape under the influence of the grid boundaries.

We take the simulation region size $-512 \le x \le 512$, $0 \le y \le 512$, the left half of the region is refined. All boundaries are periodic. The vortex is initialized in the left half by setting the equilibrium populations on the fine grid with the flow parameters according to (47). The numerical parameters are $\nu = 0.01$, $\beta_0 = 0.05$, $R = 40$, $T = 1/3$, $u_0 = 0.1$.

The D2Q15(1, 25/38) stencil is used on the grid transition. Fig. 6 shows the moment of vortex interaction with the grid boundary. Its shape is preserved in the interaction, and even after it passes back onto the fine grid (Fig. 7). The effect of the transition on the solution can be seen by magnifying the density range on the color map (Fig. 6(b)). The effect is several orders below the solution.



(a) (b)

Figure 6. Vortex density distribution on the color scale at $t = 3072$. The grid on the left is refined, and empty spaces in place of the finer nodes on the coarse area are shown in gray color. Left and right image differ in the colormap range.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 May 2023 doi:10.20944/preprints202305.0015.v1

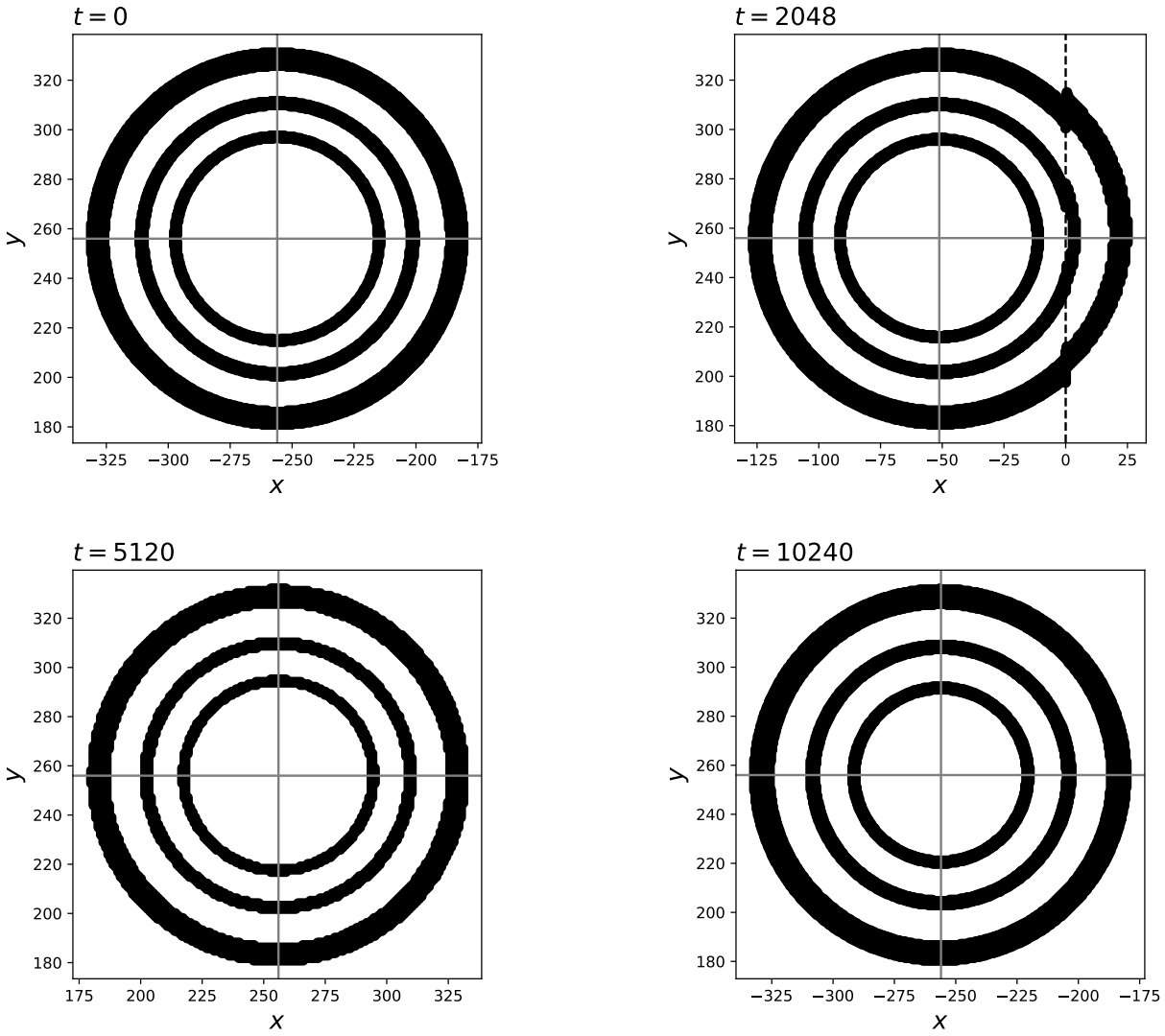


Figure 7. Density contours for $1-\rho = \{75 \div 125,\ 375 \div 425,\ 875 \div 925\} \times 10^{-7}$ for the 4 time instants.

5. Conclusion

As a conclusion, we have shown how the moment matching condition [11] can be used to perform conversions between LBM stencils on non-uniform grids. For LBM with BGK collision, we developed the method of recalibration of populations, which extends the one proposed in [9] to variable lattice temperatures, and supplemented it with the method of a transition to a different discrete velocity set.

We proved the recalibration to be valid with 2D benchmarks with one-dimensional grid transition interface. For this purpose, the algorithm that involves intermediate stencils on the transition interface was proposed.

We believe that this is an important contribution because it allows building non-uniform meshes while remaining in the LBM framework, without the use of interpolation, finite differences, series expansion, or other supplementary schemes. The recalibration method is based only on the correctness of quadrature rule in the velocity space, and the Chapman-Enskog analysis. This method can be used for different configuration of mesh interfaces, and even for dynamically refined or moving grids [53].

To apply the method further for more complex grid configurations, a combination of stencils for the grid interface has to be built. After that, the proposed method can be applied to perform the transitions.

The choice of the best configuration is yet to be found. In the following studies, we may search the source of errors that led to worse order of approximation in the Poiseuille flow benchmark.

Furthermore, it is interesting to formulate the recalibration in the 'Push' framework based on the rescaling of the outgoing populations [32] and moment matching with more than two population sets. By combining a 'Pull' and 'Push' types of streaming, a configuration of stencils that lead to a grid transition without the mass loss may be found.

In the work we used the basic BGK collision and the basic bounce-back boundary to focus on the impact of the recalibration method. For other collision operators, the recalibration procedure, and the adaptation of the collision parameter to the stencil, may differ, but may be built on the same principles that were discussed in the current text.

While the classical LBM is known to be limited in the range of its applicability, it is still valid in the vicinity of the equilibrium function with fixed velocity (usually, zero), low Mach number, and on a sufficiently detailed space-time lattice. With the use of the moment matching condition, the LBM can be locally rebuilt with different background flow [33,54,55], with different compressibility assumptions [45]. Thus, the LBM stencil can change both in time and in space to adapt to the flow conditions. The contribution of our work is the demonstration of the fact that the moment matching can also be used to adapt the stencil to the local geometry, or to an increased Reynolds flows.

Author Contributions: Conceptualization, funding acquisition, writing—review and editing A.P.; methodology, validation, formal analysis, investigation, benchmarks, visualization, writing—original draft preparation, A.B.; software, A.I.; supervision, V.L.

Funding: This research was funded by Russian Science Foundation grant number # 18-71-10004

Institutional Review Board Statement: Not applicable

Data Availability Statement: Data can be provided by the authors upon request.

Acknowledgments:

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

LBM	Lattice Boltzmann Method
CFD	Computational fluid dynamics
BGK	Bhatnagar-Gross-Krook
ZAMR	Zipped Data Structure for Adaptive Mesh Refinement
IVP	Initial-value problem
BVP	Boundary-value problem
HRR	Hybrid-recursive regularized

Appendix A Stencils for the grid transition

The stencils for the grid transitions are built according the basic Gauss-Hermite quadrature construction rules. For the constructed quadrature rule to be no worse than the classical D2Q9 variant, we require the expression

$$\frac{1}{2\pi\xi_0^2}\int\limits_{\mathbb{R}^2}\xi_x^p\xi_y^q e^{-(\xi_x^2+\xi_y^2)/2\xi_0^2}d^2\xi=\sum_i w_i c_{i,x}^p c_{i,y}^q,\qquad p,q\in\mathbb{N}_0,\qquad p+q\le 5,\tag{A1}$$

to be satisfied for all monomials up to the 5th order ($p+q\le 5$).

The quadrature points $\{\mathbf{c}_i\}$ are chosen in the grid refinement process. Then, the weights are found by solving (A1) for w_i and ξ_0. The quantities are assumed to satisfy the standard inequalities $0<w_i<1$, $\xi_0>0$.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 May 2023 doi:10.20944/preprints202305.0015.v1

This amounts to a total of $(n+1)(n+2)/2 = 21$ ($n = 5$) equations in general. If the quadrature point configuration is symmetrical in x and y, the equations for all odd powers in $c_{i,x}$ or $c_{i,y}$ are trivially satisfied. A total of six equations remain to be solved. Therefore, at least five independent weights are required in addition to the free parameter ξ_0..

As the most local configuration with five independent shells we choose (Fig. A1).

$$\{\mathbf{c}_i\} = \left\{(0,0),\ \left(0,\pm\frac{3}{2}\right),\ \left(\pm1,\pm\frac{3}{2}\right),\ \left(\pm1,\pm\frac{1}{2}\right),\ \left(\pm2,\pm\frac{1}{2}\right)\right\}, \tag{A2}$$

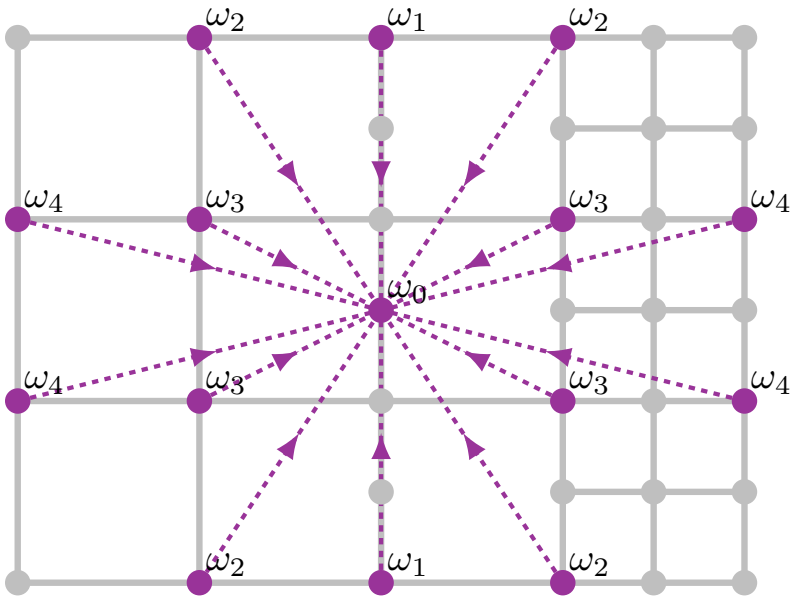


Figure A1. D2Q15(25/38, Δt) stencil

The equations to be satisfied are

$$p = 0,\ q = 0 \qquad 1 = w_0 + 2w_1 + 4w_2 + 4w_3 + 4w_4, \tag{A3}$$

$$p = 2,\ q = 0 \qquad \xi_0^2 = 4w_2 + 4w_3 + 16w_4, \tag{A4}$$

$$p = 0,\ q = 2 \qquad \xi_0^2 = 9w_1/2 + 9w_2 + w_3 + w_4, \tag{A5}$$

$$p = 4,\ q = 0 \qquad 3\xi_0^4 = 4w_2 + 4w_3 + 64w_4, \tag{A6}$$

$$p = 2,\ q = 2 \qquad \xi_0^4 = 9w_2 + w_3 + 4w_4, \tag{A7}$$

$$p = 0,\ q = 4 \qquad 3\xi_0^4 = (81w_1 + 192w_2 + 2w_3 + 2w_4)/8. \tag{A8}$$

The solution of the system is

$$w_0 = \frac{1249}{3249},\quad w_1 = \frac{6125}{103968},\quad w_2 = \frac{775}{23104},\quad w_3 = \frac{5375}{69312},\quad w_4 = \frac{925}{69312} \tag{A9}$$

and $\xi_0^2 = 25/38$.

In this work, we also use the stencil with points (Fig. A2)

$$\{\mathbf{c}_i\} = \left\{(0,0),\ (0,\pm1),\ \left(\pm1,\pm\frac{1}{2}\right)\right\}, \tag{A10}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 May 2023 doi:10.20944/preprints202305.0015.v1

For the fifth order of accuracy, the following relationships have to be satisfied

$$p = 0,\ q = 0 \qquad 1 = w_0 + 4w_1 + 2w_2, \tag{A11}$$

$$p = 2,\ q = 0 \qquad \xi_0^2 = 4w_1, \tag{A12}$$
$$p = 0,\ q = 2 \qquad \xi_0^2 = w_1 + 2w_2, \tag{A13}$$

$$p = 4,\ q = 0 \qquad 3\xi_0^4 = 4w_1, \tag{A14}$$
$$p = 2,\ q = 2 \qquad \xi_0^4 = w_1, \tag{A15}$$
$$p = 0,\ q = 4 \qquad 3\xi_0^4 = w_1/4 + 2w_2. \tag{A16}$$

It is impossible to satisfy all of these with just four parameters, so, among the fourth order equations we choose to satisfy only the equation for $p = q = 2$, while the equations for $p = 4$, $q = 0$ and $p = 0$, $q = 4$ are not satisfied. This is the only choice that results in a correct solution for the Poiseuille flow benchmark.

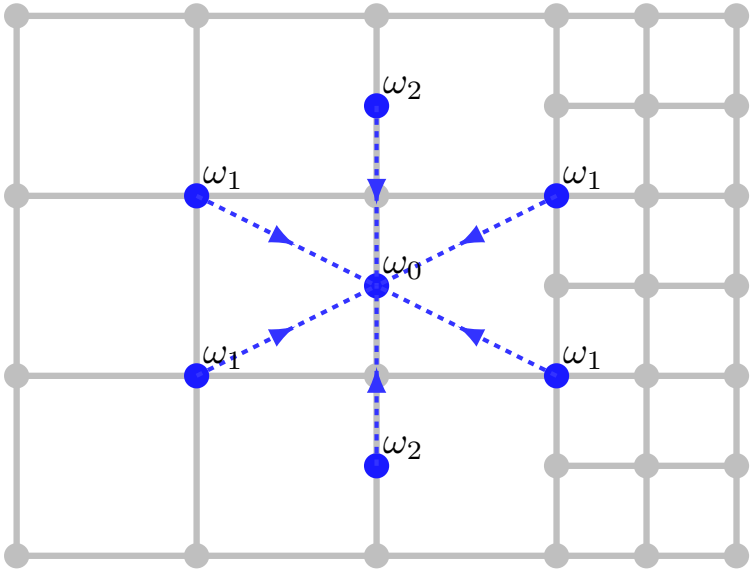


Figure A2. D2Q7(1/4, Δt) stencil

When the relationship for $p = q = 2$ is satisfied, the weights are

$$w_0 = \frac{9}{16}, \quad w_1 = \frac{1}{16}, \quad w_2 = \frac{3}{32} \tag{A17}$$

with $\xi_0^2 = 1/4$. This is the variant used in the current work.

1. Wolf-Gladrow, D.A. Lattice-gas cellular automata and lattice Boltzmann models: an introduction; Springer, 2004.
2. Shan, X.; Yuan, X.F.; Chen, H. Kinetic theory representation of hydrodynamics: a way beyond the Navier–Stokes equation. Journal of Fluid Mechanics **2006**, 550, 413–441.
3. Cockburn, B.; Shu, C.W. Runge–Kutta discontinuous Galerkin methods for convection-dominated problems. Journal of scientific computing **2001**, 16, 173–261.
4. Wittmann, M.; Haag, V.; Zeiser, T.; Köstler, H.; Wellein, G. Lattice Boltzmann benchmark kernels as a testbed for performance analysis. Computers & Fluids **2018**, 172, 582–592.
5. Perepelkina, A.; et al. Heterogeneous LBM Simulation Code with LRnLA Algorithms. Communications in Computational Physics **2023**, 33, 214–244. https://doi.org/10.4208/cicp.OA-2022-0055.

6. Zakirov, A.; Belousov, S.; Bogdanova, M.; Korneev, B.; Stepanov, A.; Perepelkina, A.; Levchenko, V.; Meshkov, A.; Potapkin, B. Predictive modeling of laser and electron beam powder bed fusion additive manufacturing of metals at the mesoscale. Additive Manufacturing **2020**, 35, 101236.
7. Schukmann, A.; Schneider, A.; Haas, V.; Böhle, M. Analysis of Hierarchical Grid Refinement Techniques for the Lattice Boltzmann Method by Numerical Experiments. Fluids **2023**, 8, 103.
8. Rohde, M.; Kandhai, D.; Derksen, J.; Van den Akker, H.E. A generic, mass conservative local grid refinement technique for lattice-Boltzmann schemes. International journal for numerical methods in fluids **2006**, 51, 439–468.
9. Filippova, O.; Hänel, D. Grid refinement for lattice-BGK models. Journal of computational Physics **1998**, 147, 219–228.
10. Filippova, O.; Hänel, D. A novel lattice BGK approach for low Mach number combustion. Journal of Computational Physics **2000**, 158, 139–160.
11. Dorschner, B.; Frapolli, N.; Chikatamarla, S.S.; Karlin, I.V. Grid refinement for entropic lattice Boltzmann models. Physical Review E **2016**, 94, 053311.
12. Lagrava, D.; Malaspinas, O.; Latt, J.; Chopard, B. Advances in multi-domain lattice Boltzmann grid refinement. Journal of Computational Physics **2012**, 231, 4808–4822.
13. Tölke, J.; Krafczyk, M. Second order interpolation of the flow field in the lattice Boltzmann method. Computers & Mathematics with Applications **2009**, 58, 898–902.
14. Chen, H.; Filippova, O.; Hoch, J.; Molvig, K.; Shock, R.; Teixeira, C.; Zhang, R. Grid refinement in lattice Boltzmann methods based on volumetric formulation. Physica A: Statistical Mechanics and its Applications **2006**, 362, 158–167.
15. Yu, Z.; Fan, L.S. An interaction potential based lattice Boltzmann method with adaptive mesh refinement (AMR) for two-phase flow simulation. Journal of Computational Physics **2009**, 228, 6456–6478.
16. Bauer, M.; Eibl, S.; Godenschwager, C.; Kohl, N.; Kuron, M.; Rettinger, C.; Schornbaum, F.; Schwarzmeier, C.; Thönnes, D.; Köstler, H.; et al. waLBerla: A block-structured high-performance framework for multiphysics simulations. Computers & Mathematics with Applications **2021**, 81, 478–501. Development and Application of Open-source Software for Problems with Numerical PDEs, https://doi.org/https://doi.org/10.1016/j.camwa.2020.01.007.
17. Fakhari, A.; Lee, T. Finite-difference lattice Boltzmann method with a block-structured adaptive-mesh-refinement technique. Physical Review E **2014**, 89, 033310.
18. Mei, R.; Shyy, W. On the Finite Difference-Based Lattice Boltzmann Method in Curvilinear Coordinates. Journal of Computational Physics **1998**, 143, 426–448. https://doi.org/https://doi.org/10.1006/jcph.1998.5984.
19. Guo, Z.; Zhao, T.S. Explicit finite-difference lattice Boltzmann method for curvilinear coordinates. Phys. Rev. E **2003**, 67, 066709. https://doi.org/10.1103/PhysRevE.67.066709.
20. Peng, G.; Xi, H.; Duncan, C.; Chou, S.H. Finite volume scheme for the lattice Boltzmann method on unstructured meshes. Physical Review E **1999**, 59, 4675.
21. Xi, H.; Peng, G.; Chou, S.H. Finite-volume lattice Boltzmann schemes in two and three dimensions. Physical Review E **1999**, 60, 3380.
22. Li, Y.; LeBoeuf, E.J.; Basu, P. Least-squares finite-element lattice Boltzmann method. Physical Review E **2004**, 69, 065701.
23. Krämer, A.; Küllmer, K.; Reith, D.; Joppich, W.; Foysi, H. Semi-Lagrangian off-lattice Boltzmann method for weakly compressible flows. Phys. Rev. E **2017**, 95, 023305. https://doi.org/10.1103/PhysRevE.95.023305.
24. Wilde, D.; Krämer, A.; Reith, D.; Foysi, H. Semi-Lagrangian lattice Boltzmann method for compressible flows. Physical Review E **2020**, 101, 053306.
25. Chew, Y.; Shu, C.; Niu, X. A new differential lattice Boltzmann equation and its application to simulate incompressible flows on non-uniform grids. Journal of Statistical Physics **2002**, 107, 329–342.
26. Guzik, S.; Gao, X.; Weisgraber, T.; Alder, B.; Colella, P. An adaptive mesh refinement strategy with conservative space-time coupling for the lattice-Boltzmann method. In Proceedings of the 51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, 2013, p. 866.
27. Liu, Z.; Li, S.; Ruan, J.; Zhang, W.; Zhou, L.; Huang, D.; Xu, J. A New Multi-Level Grid Multiple-Relaxation-Time Lattice Boltzmann Method with Spatial Interpolation. Mathematics **2023**, 11, 1089.

28. Nie, X.; Shan, X.; Chen, H. Galilean invariance of lattice Boltzmann models. Europhysics Letters **2008**, 81, 34005.
29. Timm, K.; Kusumaatmaja, H.; Kuzmin, A.; Shardt, O.; Silva, G.; Viggen, E. The lattice Boltzmann method: principles and practice. Cham, Switzerland: Springer International Publishing AG **2016**.
30. Saadat, M.H.; Dorschner, B.; Karlin, I. Extended Lattice Boltzmann Model. Entropy **2021**, 23. https://doi.org/10.3390/e23040475.
31. Alexander, F.J.; Chen, S.; Sterling, J. Lattice Boltzmann thermohydrodynamics. Physical Review E **1993**, 47, R2249.
32. Dupuis, A.; Chopard, B. Theory and applications of an alternative lattice Boltzmann grid refinement algorithm. Physical Review E **2003**, 67, 066707.
33. Dorschner, B.; Bösch, F.; Karlin, I.V. Particles on demand for kinetic theory. Physical review letters **2018**, 121, 130602.
34. Zakirov, A.; Korneev, B.; Levchenko, V.; Perepelkina, A. On the conservativity of the Particles-on-Demand method for solution of the Discrete Boltzmann Equation. Keldysh Institute Preprints **2019**, pp. 1–19. https://doi.org/10.20948/prepr-2019-35-e.
35. Zipunova, E.; Perepelkina, A.; Zakirov, A.; Khilkov, S. Regularization and the Particles-on-Demand method for the solution of the discrete Boltzmann equation. Journal of Computational Science **2021**, 53, 101376.
36. Zipunova, E.; Perepelkina, A. Development of Explicit and Conservative Schemes for Lattice Boltzmann Equations with Adaptive Streaming. Keldysh Institute Preprints **2022**, pp. 1–20. https://doi.org/10.20948/prepr-2022-7.
37. Kallikounis, N.; Dorschner, B.; Karlin, I. Particles on demand for flows with strong discontinuities. Physical Review E **2022**, 106, 015301.
38. Kallikounis, N.; Karlin, I. Particles on Demand method: theoretical analysis, simplification techniques and model extensions. arXiv preprint arXiv:2302.00310 **2023**.
39. Sawant, N.; Dorschner, B.; Karlin, I.V. Detonation modeling with the particles on demand method. AIP Advances **2022**, 12, 075107.
40. Li, X.; Shi, Y.; Shan, X. Temperature-scaled collision process for the high-order lattice Boltzmann model. Physical review E **2019**, 100, 013301.
41. Bhatnagar, P.L.; Gross, E.P.; Krook, M. A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems. Physical review **1954**, 94, 511.
42. Qian, Y.H.; d'Humières, D.; Lallemand, P. Lattice BGK models for Navier-Stokes equation. EPL (Europhysics Letters) **1992**, 17, 479.
43. Philippi, P.C.; Hegele Jr, L.A.; Dos Santos, L.O.; Surmas, R. From the continuous to the lattice Boltzmann equation: The discretization problem and thermal models. Physical Review E **2006**, 73, 056702.
44. Karlin, I.; Asinari, P. Factorization symmetry in the lattice Boltzmann method. Physica A: Statistical Mechanics and its Applications **2010**, 389, 1530–1548.
45. Kallikounis, N.G.; Dorschner, B.; Karlin, I.V. Multiscale semi-Lagrangian lattice Boltzmann method. Phys. Rev. E **2021**, 103, 063305. https://doi.org/10.1103/PhysRevE.103.063305.
46. Spiller, D.; Dünweg, B. Semiautomatic construction of lattice Boltzmann models. Physical Review E **2020**, 101, 043310.
47. Ivanov, A.; Perepelkina, A. Zipped Data Structure for Adaptive Mesh Refinement. In Proceedings of the Parallel Computing Technologies: 16th International Conference, PaCT 2021, Kaliningrad, Russia, September 13–18, 2021, Proceedings 16. Springer, 2021, pp. 245–259.
48. Ivanov, A.; Khilkov, S. Aiwlib library as the instrument for creating numerical modeling applications. Scientific Visualization **2018**, 10, 110–127.
49. Sukop, M. DT Thorne, Jr. Lattice Boltzmann Modeling Lattice Boltzmann Modeling; Springer, 2006.
50. Ginzburg, I.; d'Humières, D. Multireflection boundary conditions for lattice Boltzmann models. Phys. Rev. E **2003**, 68, 066614. https://doi.org/10.1103/PhysRevE.68.066614.
51. Wissocq, G.; Boussuge, J.F.; Sagaut, P. Consistent vortex initialization for the athermal lattice Boltzmann method. Physical Review E **2020**, 101, 043306.
52. Astoul, T.; Wissocq, G.; Boussuge, J.F.; Sengissen, A.; Sagaut, P. Analysis and reduction of spurious noise generated at grid refinement interfaces with the lattice Boltzmann method. Journal of Computational Physics **2020**, 418, 109645.

 doi:10.20944/preprints202305.0015.v1

53. Yoo, H.; Bahlali, M.; Favier, J.; Sagaut, P. A hybrid recursive regularized lattice Boltzmann model with overset grids for rotating geometries. Physics of Fluids **2021**, 33, 057113.
54. Coreixas, C.; Latt, J. Compressible lattice Boltzmann methods with adaptive velocity stencils: An interpolation-free formulation. Physics of Fluids **2020**, 32, 116102.
55. Frapolli, N.; Chikatamarla, S.S.; Karlin, I.V. Lattice kinetic theory in a comoving Galilean reference frame. Physical review letters **2016**, 117, 010604.