5.Taro Imamura (2005a).tex

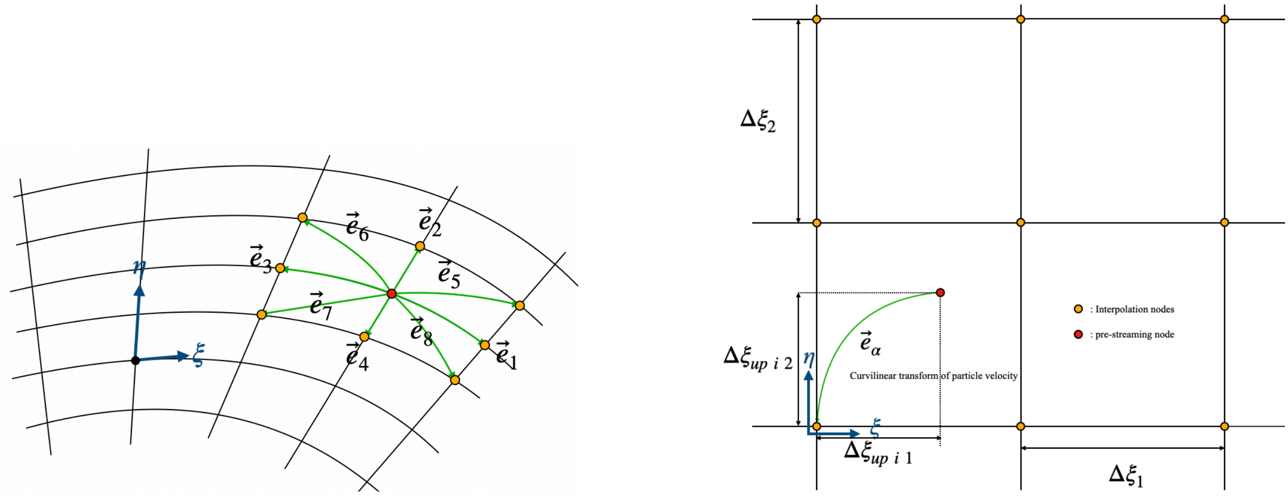
Chen Peng Chung

February 4, 2026

Contents

1 General Interpolation LBM

This note presents an extension of the Interpolation-Supplemented Lattice Boltzmann Method (ISLBM) for use in curvilinear coordinate systems. The first strategy is to transform the Cartesian coordinate system to a general curvilinear coordinate system through conformal mapping. Different from previous papers, this method extends **ISLBM** without changing the lattice system to accommodate curved motion. Note that because the transformation is based on coordinate mapping, we still compute the curved particle paths in the computational domain, as shown in the figure below:



(a) curvilinear coordinate system for SLBM **(b)** General Interpolation LBM

Figure 1.1: 2 types of curvilinear coordinate system

2 Transformation

2.1 Jacobian relation

This section will give the proof of the Jacobian relation:

$$\begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} = \frac{1}{\mathrm{J}} \begin{bmatrix} y_\eta & -x_\eta \\ -y_\xi & x_\xi \end{bmatrix} \tag{2.1}$$

For general curvilinear coordinates in two dimensions, we have to define the Lamé coefficient for analysis:

$$\vec{dr}_{+1}|_{q_2,q_3}(\vec{r}) \equiv \vec{r}(q_1+\Delta q_1, q_2, q_3) - \vec{r}(q_1, q_2, q_3) = \left.\frac{\partial \vec{r}}{\partial q_1}\right|_{q_2,q_3} dq_1 = \frac{\left.\frac{\partial \vec{r}}{\partial q_1}\right|_{q_2,q_3}}{\left|\left.\frac{\partial \vec{r}}{\partial q_1}\right|_{q_2,q_3}\right|} \left|\left.\frac{\partial \vec{r}}{\partial q_1}\right|_{q_2,q_3}\right| dq_1 \tag{2.2}$$

From the expression above, we can define unit vector and coefficient for differential geometry.

$$\begin{aligned} h_1 &\equiv \left|\frac{\partial \vec{r}}{\partial q_1}\right|_{q_2,q_3} \\ \vec{e}_1 &\equiv \frac{1}{h_1} \left.\frac{\partial \vec{r}}{\partial q_1}\right|_{q_2,q_3} \end{aligned} \tag{2.3}$$

where the symbol h_1 is a Lamé coefficient for the coordinate component. Let's take the definition to the differential of the position vector about variable (ξ, η).

$$\begin{aligned}
\frac{\partial \vec{r}}{\partial \xi} &= h_\xi \vec{e}_\xi = \vec{e}_x x_\xi + \vec{e}_y y_\xi \\
\frac{\partial \vec{r}}{\partial \eta} &= h_\eta \vec{e}_\eta = \vec{e}_x x_\eta + \vec{e}_y y_\eta \\
\frac{\partial \vec{r}}{\partial x} &= \vec{e}_x = h_\xi \vec{e}_\xi \xi_x + h_\eta \vec{e}_\eta \eta_x \\
\frac{\partial \vec{r}}{\partial y} &= \vec{e}_y = h_\xi \vec{e}_\xi \xi_y + h_\eta \vec{e}_\eta \eta_y
\end{aligned} \tag{2.4}$$

we can do some simple computation, and get the result below :

$$\begin{aligned}
\frac{h_\xi}{x_\xi}\vec{e}_\xi - \frac{h_\eta}{x_\eta}\vec{e}_\eta &= \vec{e}_y\left(\frac{y_\xi}{x_\xi} - \frac{y_\eta}{x_\eta}\right) = \vec{e}_y \frac{y_\xi x_\eta - y_\eta x_\xi}{x_\eta x_\xi} \\
\frac{h_\xi}{y_\xi}\vec{e}_\xi - \frac{h_\eta}{y_\eta}\vec{e}_\eta &= \vec{e}_x\left(\frac{x_\xi}{y_\xi} - \frac{x_\eta}{y_\eta}\right) = \vec{e}_x \frac{x_\xi y_\eta - x_\eta y_\xi}{y_\eta y_\xi}
\end{aligned} \tag{2.5}$$

Furthermore, we can rewrite this as:

$$\begin{aligned}
-x_\eta h_\xi \vec{e}_\xi + x_\xi h_\eta \vec{e}_\eta &= \vec{e}_y\left(x_\xi y_\eta - x_\eta y_\xi\right) \\
y_\eta h_\xi \vec{e}_\xi - y_\xi h_\eta \vec{e}_\eta &= \vec{e}_x\left(x_\xi y_\eta - x_\eta y_\xi\right)
\end{aligned} \tag{2.6}$$

Substitution the equation (2.6) into (2.4)

$$\begin{bmatrix} \xi_x & \xi_y \\ \eta_x & \eta_y \end{bmatrix} = \frac{1}{(x_\xi y_\eta - x_\eta y_\xi)} \begin{bmatrix} y_\eta & -x_\eta \\ -y_\xi & x_\xi \end{bmatrix} \tag{2.7}$$

The equation above is Jacobian relation.

2.2 Transform for Lattice Boltzmann Equation

First, review the basic governing equation for the standard lattice Boltzmann method, i.e., the lattice Boltzmann equation in Cartesian coordinates with 3 dimensions:

$$f_i(\vec{x} + \vec{c}_i \Delta t, t+1) = f_i(\vec{x}, t) + \Omega(f_i(\vec{x}, t), f_i^{eq}(\rho, \vec{u}, t)) \tag{2.8}$$

where the symbol Ω is a collision operator, so it can be separated into two types of forms.

$$f_i(\vec{x} + \vec{c}_i \Delta t, t+1) = f_i(\vec{x}, t) + \omega(f_i(\vec{x}, t) - f_i^{eq}(\rho, \vec{u}, t)) \tag{2.9}$$

If the collision operator is chosen as the BGK operator, we call the equation the **LBGK** equation. On the other hand, we can allow each momentum component to have a different relaxation effect. To achieve this, we introduce a basis transformation matrix that transforms the function from velocity space to momentum space. This means that the streaming step and collision step are performed in separate spaces. Let's show the transform below:

$$\begin{aligned}\mathbf{M}\vec{f}(\vec{x}+\vec{c}_i\Delta t,t+1) &= \mathbf{M}\vec{f}(\vec{x},t)+\mathbf{SM}(\vec{f}(\vec{x},t)-\vec{f}^{eq}(\rho,\vec{u},t)) \\ &= \mathbf{M}\vec{f}(\vec{x},t)+\mathbf{S}\left(\vec{m}(\vec{x},t)-\vec{m}^{eq}(\vec{x},t,f_i^{eq}(\rho,\vec{u},t))\right)\end{aligned} \tag{2.10}$$

In this work, we first apply the LBGK equation and use coordinate mapping to modify the streaming process, thereby extending ISLBM to curvilinear coordinate systems. The limitation of this approach is that we have not yet extended the MRT operator to general coordinate systems, as we have not fully grasped the underlying mathematical theory and physical mechanisms.

2.2.1 Transform Collision Step

Equations (2.9) and (2.10) are described in three-dimensional Cartesian coordinates. To transform these equations to general curvilinear coordinates, we note that the position vector only appears at discrete grid nodes, and the collision operator does not involve derivatives with respect to position. Therefore, we can directly substitute $\vec{x}$ with $\vec{\xi}$, where $\vec{x}$ denotes the Cartesian coordinate position and $\vec{\xi}$ denotes the curvilinear coordinate position.

We can say that position information only has position variables located at the grid nodes, so we can simply transform by substituting.

$$f_i^{\star}(\vec{\xi},t)=f_i(\vec{\xi},t)+\omega(f_i(\vec{\xi},t)-f_i^{eq}(\rho,\vec{u},t)) \tag{2.11}$$

2.2.2 Transform Streaming Step

Definition of normalized discrete velocity set In general curvilinear coordinates, the variation of the position vector or velocity or any physical variable related to the length dimension needs to consider the transform factor for basis transformation. This is complex for analyzing the motion of particles. Let's define the normalized discrete velocity set.

$$\vec{e}_\alpha = \underbrace{c_\alpha^i}_{\text{non-dimension}} \quad \underbrace{\vec{g}_i(\vec{\xi})}_{\text{curvature tangent vector}} \quad \underbrace{\frac{\Delta x}{\Delta t}}_{\text{(lattice speed)}\approx 1} \tag{2.12}$$

where the vector $\vec{e}_\alpha$ is the non-dimensional discrete particle velocity set, and the vector $\vec{g}_i$ is the tangent vector related to the curvature at the position. $\frac{\Delta x}{\Delta t}$ is the lattice speed, and its function in this definition is velocity dimension. So we rewrite the definition to another form below based on differential geometry.

$$\vec{e}_\alpha^{\,j}=c_\alpha^i\frac{\partial\xi_j}{\partial x_i} \tag{2.13}$$

where x represents the index of the Cartesian coordinate and ξ is the index of the general curvilinear coordinate, and the differential represents the basis transform from Cartesian coordinates to curvilinear coordinates. Therefore, we can see the particle velocity distortion through the factor of the transform.

Path integration of the particle velocity The streaming step equation

$$f_\alpha(\vec{x}, t) = f_\alpha^\star(\vec{x} - \vec{c}_\alpha \delta t, t)$$

We can see that the essential issue is computing the integration of the discrete particle velocity to know the position where the post-streaming position and streaming length are. The length of the streaming step is defined as below:

$$\delta\vec{\xi}_\alpha = \int_0^{\Delta t} d\vec{\xi}_\alpha = \int_0^{\Delta t} \vec{e}_\alpha dt \tag{2.14}$$

1st explicit Euler method:

$$\delta\vec{\xi}_\alpha \approx \vec{e}_\alpha \Delta t \ldots O(\Delta t) \tag{2.15}$$

2nd order Runge-Kutta method:
Using the 2nd order Runge-Kutta method to solve the integration, it is separated into two steps:

$$\begin{aligned} \text{step 1}: \Delta\vec{\xi}_\alpha^{(1)} &= \frac{1}{2}\Delta t \vec{e}_\alpha \\ \text{step 2}: \Delta\vec{\xi}_\alpha &= \Delta t \vec{e}_\alpha(\vec{\xi} - \vec{\xi}_\alpha^{(1)}) \end{aligned} \tag{2.16}$$

The result of discrete integration (2.16) has 2nd order accuracy in time space. The figure shows the method of 2nd order Runge-Kutta to treat integration.

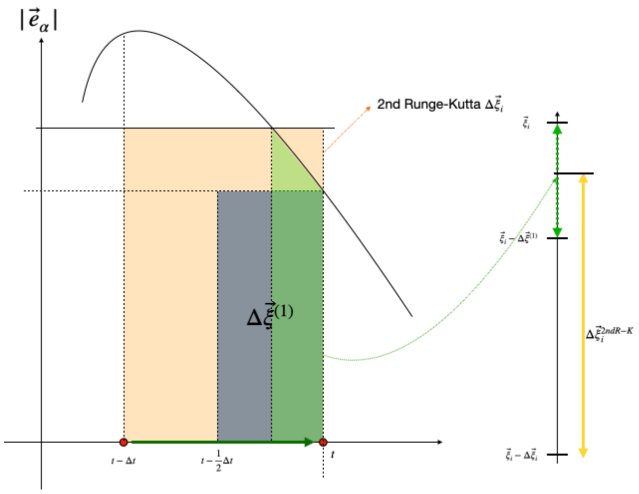


Figure 2.2: integration skill-2nd Runge-Kutta

In Interpolation-Supplemented LBM, the streaming step is implemented by pulling data backward from the pre-streaming position to the current grid node through interpolation. This approach ensures that information does not spread from grid nodes to non-physical locations. Therefore, when integrating the particle path, we integrate backward from $\vec{\xi}_\alpha$ to $\vec{\xi}_\alpha - \Delta\vec{\xi}_\alpha$, rather than forward from $\vec{\xi}_\alpha$ to $\vec{\xi}_\alpha + \Delta\vec{\xi}_\alpha$.

Streaming step after transformation When we get the result of the path integration of particle discrete velocity along the direction, we can rewrite the streaming step equation below: The discrete process is a little similar to the process of distribution for time-space discretizing. However, the term discretized is different; one is time integration of distribution and the other is time integration of particle path. In curvilinear coordinates, we have

$$f_\alpha(\vec{\xi}, t) = f^\star_\alpha(\vec{\xi} - \Delta\vec{\xi}_\alpha, t) \tag{2.17}$$

where $\Delta\vec{\xi}_\alpha$, the change of position, has 2nd order accuracy in time space, and the R.H.S. term is the post-collision function.

Accuracy effect in time space for lattice Boltzmann equation - BGK operator LBGK in curvilinear coordinates can be rewritten below:

$$f_\alpha(\vec{\xi}, t+1) - f_\alpha(\vec{\xi} - \Delta\vec{\xi}_\alpha, t) = \omega(f_\alpha(\vec{\xi} - \Delta\vec{\xi}_\alpha, t) - f^{eq}_\alpha(\rho, \vec{u}, t)) \ldots \text{(curvilinear coordinate)} \tag{2.18}$$

and we can define a material derivative for the discrete-velocity Boltzmann equation:

$$\mathcal{D}_t = \partial_t + \frac{\delta\vec{\xi}_\alpha}{\delta t} \cdot \vec{\nabla}_\xi \tag{2.19}$$

We call the above operator the "discrete-velocity material derivative". Using Taylor expansion on equation (2.18):

$$\left(\partial_t + \frac{\delta\vec{\xi}_\alpha}{\delta t} \cdot \vec{\nabla}_\xi\right) f_\alpha \Delta t + \frac{1}{2}\left(\partial_t + \frac{\delta\vec{\xi}_\alpha}{\delta t} \cdot \vec{\nabla}_\xi\right)^2 f_\alpha \Delta t^2 \Bigg|_{(\vec{\xi} - \Delta\vec{\xi}_\alpha, t)} = \omega(f_\alpha(\vec{\xi} - \Delta\vec{\xi}_\alpha, t) - f^{eq}_\alpha(\rho, \vec{u}, t)) \tag{2.20}$$

Left-hand side and right-hand side are both divided by Δt, which can recover the N-S equation in differential form.

$$\left(\partial_t + \frac{\delta\vec{\xi}_\alpha}{\delta t} \cdot \vec{\nabla}_\xi\right) f_\alpha + \frac{1}{2}\left(\partial_t + \frac{\delta\vec{\xi}_\alpha}{\delta t} \cdot \vec{\nabla}_\xi\right)^2 f_\alpha \Delta t \Bigg|_{(\vec{\xi} - \Delta\vec{\xi}_\alpha, t)} + O(\Delta t^2) = \omega(f_\alpha(\vec{\xi} - \Delta\vec{\xi}_\alpha, t) - f^{eq}_\alpha(\rho, \vec{u}, t)) \tag{2.21}$$

The second term of the velocity-discrete Boltzmann equation corresponds to the shear stress field through Chapman-Enskog expansion. If we use the explicit Euler method to discretize the particle path integration, the first term has only zeroth-order temporal accuracy, while the stress term has only first-order temporal accuracy. This is insufficient to recover the Navier-Stokes equation. Therefore, we must use a method that provides higher-order temporal accuracy for the path integration, ensuring that the differential term of the velocity-discrete Boltzmann equation has at least second-order temporal accuracy.

3 Multiple Scale Expansion

For collision and streaming in mesoscopic space, and advection and diffusion in macroscopic space, we have three types of time scales:

1. $K^{(0)}$: The first time scale, related to collision and streaming in mesoscopic space.
2. $K^{(1)}$: The second time scale, related to convection in macroscopic space.

3. $K^{(2)}$: The third time scale, related to diffusion in macroscopic space.

If K = 0.01, then we can define three types of time variables: (the value is about $K^{(0)}$, $K^{(1)}$, $K^{(2)}$)

$$
\begin{aligned}
& t \in K^{(2)} \\
& t^{(1)} \equiv K * t \in K^{(1)} \\
& t^{(2)} \equiv K^2 * t \in K^{(0)}
\end{aligned}
\tag{3.1}
$$

and define two types of space variables:

$$
\vec{r}^{(1)} \equiv K * \vec{r} \tag{3.2}
$$

Therefore, the differential about the time variable is:

$$
\frac{\partial}{\partial t} = K \frac{\partial}{\partial t^{(1)}} + K^2 \frac{\partial}{\partial t^{(2)}} = K \partial_t^{(1)} + K \partial_t^{(2)} \tag{3.3}
$$

In physical space, we have the relation about two variables:

$$
\begin{aligned}
& \frac{\partial}{\partial \vec{\xi}} = K \frac{\partial}{\partial \vec{\xi}^{(1)}} \\
& \Rightarrow \vec{\nabla}_{\vec{\xi}} = K \vec{\nabla}_{\vec{\xi}^{(1)}}
\end{aligned}
\tag{3.4}
$$

Finally, for three types of time variables, the multi-scale expansion of the distribution function:

$$
f_i = f_i^{eq} + K f_i^{(1)} + K^2 f_i^{(2)} + \dots \tag{3.5}
$$

We have to know that the time scale is a domain for time space like $K^{(0)}$, $K^{(1)}$, and the element of the time scale is a variable, and their scales have different properties.

Multi-Scale Technique and LBGK Before analyzing the LBGK, first, review the form of the basic equation:

$$
f_\alpha(\vec{r} + \vec{e}_\alpha \delta t, t + \delta t) - f_\alpha(\vec{r}, t) = -\frac{\delta t}{\tau} \left[f^{neq}(\vec{r}, t) \right] \tag{3.6}
$$

Using particle material derivative:

$$
\mathcal{D}_t = \partial_t + \vec{e}_\alpha \cdot \vec{\nabla}
$$

and Taylor expansion to analyze the left-hand side of the LBGK:

$$
\mathcal{D}_t \delta t + \frac{\delta t^2}{2} \mathcal{D}_t^2 f_\alpha \Big|_{\vec{r},t} = -\frac{\delta t}{\tau} \left[f^{neq}(\vec{r}, t) \right] \tag{3.7}
$$

$$
\left(\partial_t + \vec{e}_\alpha \cdot \vec{\nabla} \right) f_\alpha + \frac{\delta t^1}{2} \left(\partial_t + \vec{e}_\alpha \cdot \vec{\nabla} \right)^2 f_\alpha = -\frac{1}{\tau} \left[f^{neq}(\vec{r}, t) \right] \tag{3.8}
$$

substitute the above equation using the discrete velocity Boltzmann equation:

$$\left(\partial_t + \vec{e}_\alpha \cdot \vec{\nabla}\right) f_\alpha = -\frac{1}{\tau}\left[f^{neq}(\vec{r},t)\right]$$

we have the equation below:

$$\left(\partial_t + \vec{e}_\alpha \cdot \vec{\nabla}\right) f_\alpha - \frac{\delta t^1}{2}\left(\partial_t + \vec{e}_\alpha \cdot \vec{\nabla}\right)\frac{1}{\tau}\left[f_\alpha^{neq}(\vec{r},t)\right] = -\frac{1}{\tau}\left[f_\alpha^{neq}(\vec{r},t)\right] \tag{3.9}$$

For analyzing the equation above, review the multiple scale relation as shown below:

$$\begin{aligned}
\partial_t &= K\partial_t^{(1)} + K^2\partial_t^{(2)} + \dots \\
\vec{\nabla} &= K\vec{\nabla}^{(1)} + K^2\vec{\nabla}^{(2)} + \dots \\
f_\alpha^{neq} &= f_\alpha - f_\alpha^{eq} = Kf_\alpha^{(1)} + K^2 f_\alpha^{(2)} + \dots
\end{aligned} \tag{3.10}$$

and go back to the equation (3.9), we have

$$\begin{aligned}
&K\left(\partial_t^{(1)} + \vec{e}_\alpha \cdot \vec{\nabla}^{(1)}\right) f_\alpha = -K\frac{1}{\tau}\left[f_\alpha^{(1)}(\vec{r},t)\right] \\
&K^2\left(\partial_t^{(2)} + \vec{e}_\alpha \cdot \vec{\nabla}^{(2)}\right) f_\alpha - \frac{\delta t K^2}{2\tau}\left(\partial_t^{(1)} + \vec{e}_\alpha \cdot \vec{\nabla}^{(1)}\right)\left[f_\alpha^{(1)}(\vec{r},t)\right] = -K^2\frac{1}{\tau}\left[f_\alpha^{(2)}(\vec{r},t)\right]
\end{aligned} \tag{3.11}$$

But $f_\alpha = f_\alpha^{eq} + f_\alpha^{neq} = f_\alpha^{eq} + Kf_\alpha^{(1)} + \dots$, the equation can be shown below :

$$\begin{aligned}
&K\left(\partial_t^{(1)} + \vec{e}_\alpha \cdot \vec{\nabla}^{(1)}\right) f_\alpha^{eq} = -K\frac{1}{\tau}\left[f_\alpha^{(1)}(\vec{r},t)\right] \\
&K^2\left(\partial_t^{(2)} + \vec{e}_\alpha \cdot \vec{\nabla}^{(2)}\right) f_\alpha^{eq} - \frac{\delta t K^2}{2\tau}\left(\partial_t^{(1)} + \vec{e}_\alpha \cdot \vec{\nabla}^{(1)}\right)\left[f_\alpha^{(1)}(\vec{r},t)\right] + K^2\left(\partial_t^{(1)} + \vec{e}_\alpha \cdot \vec{\nabla}^{(1)}\right) f_\alpha^{(1)} \\
&= -K^2\frac{1}{\tau}\left[f_\alpha^{(2)}(\vec{r},t)\right]
\end{aligned} \tag{3.12}$$

4 Boundary Condition

In multiple scale analysis, the first time scale non-equilibrium distribution function is:

$$-\omega\delta t\left(\partial_t^{(1)} + \vec{e}_\alpha \cdot \vec{\nabla}^{(1)}\right) f_\alpha^{eq} = \left[f_\alpha^{(1)}(\vec{r},t)\right] \tag{4.1}$$

where the differential variable is the second time variable $t^{(1)}$, which belongs to the advection time scale $K^{(1)}$. And the Maxwell-Boltzmann equilibrium distribution function with Hermite polynomials expansion has:

$$f_\alpha^{eq} = w_\alpha\rho\left(1 + \frac{e_\alpha^i u_i}{c_s^2} + \frac{u_i u_j\left(e_\alpha^i e_\alpha^j - c_s^2\delta^{ij}\right)}{2c_s^4}\right) \tag{4.2}$$

so we have to deal with the macroscopic derivative:

$$\begin{aligned}
\partial_t^{(1)} f_\alpha^{eq} &= \frac{\partial f_\alpha^{eq}}{\partial u_i}\partial_t^{(1)} u_i + \frac{\partial f_\alpha^{eq}}{\partial \rho}\partial_t^{(1)} \rho \\
\vec{\nabla}^{(1)} f_\alpha^{eq} &= \frac{\partial f_\alpha^{eq}}{\partial u_i}\vec{\nabla}^{(1)} u_i + \frac{\partial f_\alpha^{eq}}{\partial \rho}\vec{\nabla}^{(1)} \rho
\end{aligned} \tag{4.3}$$

where $t^{(1)} \in K^{(1)}$, which is time scale about advection. from equilibirium distribution function, the differential term have the result below :

$$\begin{aligned}
\frac{\partial f_\alpha^{eq}}{\partial u_i}(\vec{r},t) &= w_\alpha \rho \left[\frac{e_\alpha^i}{c_s^2} + \frac{u_j\left(e_\alpha^i e_\alpha^j - c_s^2 \delta^{ij}\right)}{2c_s^4}\right] \\
&= w_\alpha \rho \left[\frac{1}{c_s^2}\left(e_\alpha^i - u_i\right) + \frac{u_j e_\alpha^i e_\alpha^j}{2c_s^4}\right] \\
&\approx \frac{e_\alpha^i - u_i}{c_s^2} f_\alpha^{eq} \\
\frac{\partial f_\alpha^{eq}}{\partial \rho}(\vec{r},t) &= \frac{1}{\rho} f_\alpha^{eq}(\vec{r},t)
\end{aligned} \tag{4.4}$$

Using macroscopic conservation law (include mass conservation and momentum conservation) to solve the time differential of density and vleocity.

$$\begin{aligned}
\frac{\partial^{(1)} \rho}{\partial t} &= -\vec{\nabla}^{(1)} \rho \vec{u} \\
\frac{\partial^{(1)} \vec{u}}{\partial t} &= -\vec{u}\cdot\vec{\nabla}^{(1)}\vec{u} - \frac{1}{\rho}\vec{\nabla}^{(1)} p
\end{aligned} \tag{4.5}$$

For equilibilirium distribution function, the first scale time variable derivation have :

$$\begin{aligned}
\partial_t^{(1)} f_\alpha^{eq} &= f_\alpha^{eq}\left(\frac{e_\alpha^i - u_i}{c_s^2}(-u_\beta \frac{\partial^{(1)}}{\partial x_\beta} u_i - \frac{1}{\rho}\frac{\partial^{(1)}}{\partial x_i} p) + \frac{-1}{\rho}\frac{\partial^{(1)}}{\partial x_\beta}\rho u_\beta\right) \\
\vec{\nabla}^{(1)} f_\alpha^{eq} &= f_\alpha^{eq}\left(\frac{e_\alpha^i - u_i}{c_s^2}(\vec{\nabla}^{(1)} u_i) + \frac{1}{\rho}\vec{\nabla}^{(1)}\rho\right) \\
e_\alpha^\gamma \frac{\partial^{(1)}}{\partial x_\gamma} f_\alpha^{eq} &= e_\alpha^\gamma f_\alpha^{eq}\left(\frac{e_\alpha^i - u_i}{c_s^2}(\frac{\partial^{(1)}}{\partial x_\gamma} u_i) + \frac{1}{\rho}\frac{\partial^{(1)}}{\partial x_\gamma}\rho\right)
\end{aligned} \tag{4.6}$$

Finally, we can get the **the first scale non-equilibilirium distribution function**

$$f_\alpha^{(1)} = \omega \delta t \left(A \cdot [-u_\beta \frac{\partial^{(1)}}{\partial x_\beta} u_i \underbrace{- \frac{1}{\rho}\frac{\partial^{(1)}}{\partial x_i} p}_{ignore} + e_\alpha^\gamma \frac{\partial^{(1)}}{\partial x_\gamma} u_i] + \underbrace{B \cdot [-\frac{\partial^{(1)} \rho u_\beta}{\partial x_\beta} + e_\alpha^\gamma \frac{\partial^{(1)} \rho}{\partial x_\gamma}]}_{ignore} \right) f_\alpha^{eq} \tag{4.7}$$

where

$$A = \frac{e_\alpha^i - u_i}{c_s^2}$$
$$B = \frac{1}{\rho} \tag{4.8}$$

In this section , we have to give the assumption :

1. : pressure field is a constant field, so $\frac{\partial^{(1)}}{\partial x_i} p = 0$
2. : density field is a constant field, so $\frac{\partial^{(1)}}{\partial x_\beta} \rho = 0$

From the part of the equation above, I could find some zero term, let's show below :

$$\begin{aligned} &\frac{e_\alpha^i - u_i}{c_s^2} \cdot \left(\frac{-1}{\rho} \frac{\partial^{(1)}}{\partial x_i} p = 0 \right) \ldots \text{(second term of A)} \\ &\frac{1}{\rho} \left((e_\alpha^\beta - u_\beta) \frac{\partial \rho}{\partial x_\beta} \right) = 0 \ldots \text{(in B)} \\ &-\frac{\partial^{(1)} \rho u_\beta}{\partial x_\beta} = -u_\beta \frac{\partial^{(1)} \rho}{\partial x_\beta} - \rho \frac{\partial^{(1)} u_\beta}{\partial x_\beta} \end{aligned} \tag{4.9}$$

The distribution function is : (could be controlled by macroscopic variables)

$$\begin{aligned} f_\alpha &= f_\alpha^{eq} + f_\alpha^{(1)} \\ &= f_\alpha^{eq} + \omega \delta t \left(\frac{(e_\alpha^i - u_i)(e_\alpha^\beta - u_\beta)}{c_s^2} \frac{\partial^{(1)} u_i}{\partial x_\beta} - \frac{\partial^{(1)} u_\beta}{\partial x_\beta} \right) f_\alpha^{eq} \end{aligned} \tag{4.10}$$

where Knuson number is 1, and general distribution function can be calculate as above equation.

Boundary Condition In GILBM, we use macroscopic parameter limitations to enforce boundary conditions on the distribution function. At boundary nodes (i.e., nodes located between solid and fluid nodes), we make assumptions to construct the governing equations.

$$\begin{aligned} f_\alpha|_{b.c.} &= f_\alpha^{eq} + f_\alpha^{(1)} \\ &= f_\alpha^{eq} + \omega \delta t f_\alpha^{eq} \left(\frac{(c_\alpha^i - u_i)(c_\alpha^\beta - u_\beta)}{c_s^2} \frac{\partial u_i}{\partial x_\beta} - \frac{\partial u_\beta}{\partial x_\beta} \right) \end{aligned} \tag{4.11}$$

where gradient of velocity field :

$$\frac{\partial u_i}{\partial x_\beta} = \frac{\partial u_i}{\partial \xi_i} \frac{\partial \xi_i}{\partial x_\beta} \tag{4.12}$$

The first term is calculated using second order one side finite difference method :

$$\begin{aligned} \left.\frac{\partial u}{\partial \xi}\right|_{i+\frac{1}{2}} &= \frac{3u_i - 4u_{i-1} + u_{i-2}}{2\Delta \xi} + O(\Delta t^2) \\ \left.\frac{\partial u}{\partial \xi}\right|_{i-\frac{1}{2}} &= \frac{-3u_i + 4u_{i+1} - u_{i+2}}{2\Delta \xi} + O(\Delta t^2) \end{aligned} \tag{4.13}$$

and second is the term related to the curvature, have been calculated in initialization.

5 Algorithm

The contravariant velocity (i.e., the discrete velocity set in curvilinear coordinates) needs to be calculated at the beginning of the simulation process and requires an extra array in the code to store the data. However, this pre-calculated value can reduce the computational cost at every time step by close to 50 percent.

ISLBM initialization

1. setting Re number and relaxation factor
2. calculate relaxation time :
$$\tau = \frac{\Delta t}{\omega}$$
3. calculate viscosity :
$$\nu = c_s^2(\tau - 0.5dt)$$
4. calculate reference velocity :
$$U_{ref} = Re\, nu$$

GILBM initialization

1. setting Re number and relaxation factor
2. calculate contravariant velocity (i.e., normalized discrete particle velocity set) :
$$e_\alpha^i = c_\alpha^\beta \frac{\partial \xi_i}{\partial x_\beta}$$
3. calculate the global time step using contravariant velocity :
$$\Delta t_{glopbal} = CFL \cdot min_{i,j,k,\alpha} | \frac{1}{e_\alpha^i|_{j,k}} |$$
4. calculate the relaxation time :
$$\tau = \frac{\Delta t_{global}}{\omega}$$
5. calculate the viscosity
$$\nu = (\tau - 0.5\Delta t_{global})c_s^2$$