

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

Taro Imamura<sup>a,\*</sup>, Kojiro Suzuki<sup>b</sup>, Takashi Nakamura<sup>a</sup>, Masahiro Yoshida<sup>a</sup>

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

<sup>b</sup> *Department of Advanced Energy, The University of Tokyo,  
5-1-5 Kashiwa-no-ha, Kashiwa, Chiba 277-8562, Japan*

**Journal of Computational Physics** 202 (2005) 645–663

Received 29 December 2003; revised 21 June 2004; accepted 2 August 2004

**Presenter:** Chen Peng Chung

# Outline

## GILBM and ISLBM

Extend **Interpolation-Supplemented LBM (ISLBM)** to general curvilinear coordinates through **conformal mapping**, without changing the lattice system. But the point of mapping is How can I deal with discrete particle velocity set.

### Key Features:

- Cartesian → Curvilinear via coordinate mapping
- Curved particle paths in computational domain
- Lattice structure remains **unchanged**
- Pull-back interpolation for streaming

$$f_\alpha(\vec{\xi}, t + \Delta t) = f_\alpha^*(\vec{\xi} - \Delta \vec{\xi}_\alpha, t) \quad (1)$$

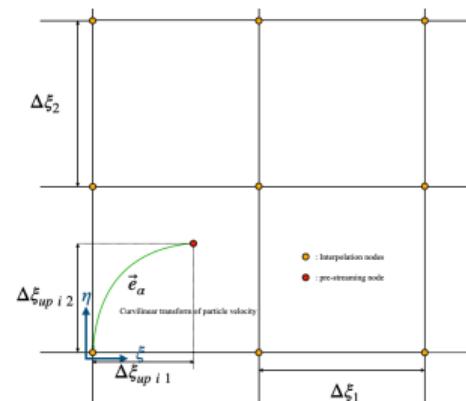


Figure: GILBM: curved paths in  $\xi$ -space

## Jacobian Transformation Matrix:

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

## Lamé Coefficient & Unit Vector:

$$h_1 \equiv \left| \frac{\partial \vec{r}}{\partial q_1} \right|, \quad \vec{e}_1 \equiv \frac{1}{h_1} \frac{\partial \vec{r}}{\partial q_1}$$

Position vector differentials:

$$\frac{\partial \vec{r}}{\partial \xi} = h_\xi \vec{e}_\xi = x_\xi \vec{e}_x + y_\xi \vec{e}_y$$

$$\frac{\partial \vec{r}}{\partial \eta} = h_\eta \vec{e}_\eta = x_\eta \vec{e}_x + y_\eta \vec{e}_y$$

## Physical Meaning:

- $h_i$ : Scale factor (metric coefficient)
- $\vec{e}_i$ : Local tangent direction
- $J$ : Jacobian determinant (area scaling)
- Transforms derivatives between coordinates

**Standard LBE (BGK Collision Operator):**

$$f_\alpha(\vec{x} + \vec{c}_\alpha \Delta t, t + \Delta t) = f_\alpha(\vec{x}, t) + \omega(f_\alpha - f_\alpha^{eq})$$

**LBE with MRT Collision Operator:**

$$\mathbf{M}\vec{f}(\vec{x} + \vec{c}_\alpha \Delta t, t + \Delta t) = \mathbf{M}\vec{f}(\vec{x}, t) + \mathbf{SM}(\vec{f} - \vec{f}^{eq})$$

**Collision Step:**

Position appears only at grid nodes; collision has no spatial derivatives.

⇒ Direct substitution:  $\vec{x} \rightarrow \vec{\xi}$

$$f_i^*(\vec{\xi}, t) = f_i + \omega(f_i^{eq} - f_i)$$

**Streaming Step:**

Requires path integration of particle velocity in curvilinear space.

$$f_\alpha(\vec{\xi}, t) = f_\alpha^*(\vec{\xi} - \Delta\vec{\xi}_\alpha, t)$$

(Pull-back from pre-streaming position via interpolation)

## Definition

Transform Cartesian lattice velocity to curvilinear coordinates:

$$\vec{e}_\alpha^j = c_\alpha^i \frac{\partial \xi_j}{\partial x_i} \dots \text{(assume lattice speed = 1)}$$

- $c_\alpha^i$ : Dimensionless discrete velocity (e.g., D2Q9: 0,  $\pm 1$ )
- $\frac{\partial \xi_j}{\partial x_i}$ : Coordinate transformation (from Jacobian)
- $\vec{e}_\alpha^j$ : Shows **velocity distortion** due to curvature

## Volumetric Lattice Boltzmann in Curvilinear Coordinates:

$$\vec{e}_\alpha = \underbrace{c_\alpha^i}_{\text{an integer}} \cdot \underbrace{\vec{g}_i(\vec{\xi})}_{\text{tangent vector}} \cdot \underbrace{\frac{\Delta x}{\Delta t}}_{\text{lattice speed} \approx 1}$$

Redefine the particle's discrete velocity: Enforce the direction of velocity to be the same as the tangent vector.

# Path Integration: 2nd Order Runge-Kutta

From straming step :

$$f_\alpha(\vec{\xi}, t + \Delta t) = f_\alpha^*(\vec{\xi} - \delta\vec{\xi}_\alpha, t)$$

In curvilinear coordinate, we have to calculate the position of the pre-streaming particles.

**Streaming Length:**

$$\delta\vec{\xi}_\alpha = \int_0^{\Delta t} \vec{e}_\alpha(\vec{\xi}(t)) dt$$

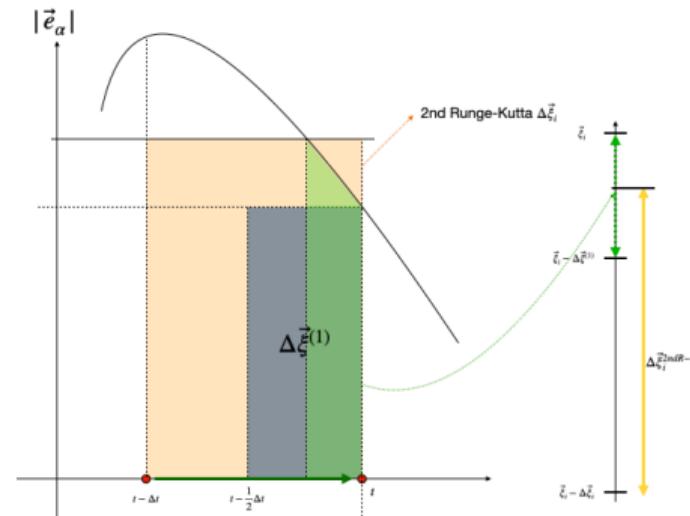
**ZXZX2nd**

**Or-  
der  
Runge-  
Kutta:**

Step 1:  $\Delta\vec{\xi}_\alpha^{(1)} = \frac{1}{2}\Delta t \vec{e}_\alpha(\vec{\xi})$

Step 2:  $\Delta\vec{\xi}_\alpha = \Delta t \vec{e}_\alpha(\vec{\xi} - \Delta\vec{\xi}_\alpha^{(1)})$

**Why  
not**



**Three Time Scales:**

- $K^{(0)}$ : Collision & streaming (mesoscopic)
- $K^{(1)}$ : Convection / advection (macroscopic)
- $K^{(2)}$ : Diffusion (macroscopic)

**Scale Operators:**

$$\begin{aligned}\partial_t &= K\partial_t^{(1)} + K^2\partial_t^{(2)} \\ \vec{\nabla} &= K\vec{\nabla}^{(1)}\end{aligned}$$

**Distribution Expansion:**

$$f_\alpha = f_\alpha^{eq} + Kf_\alpha^{(1)} + K^2f_\alpha^{(2)} + \dots$$

**Order  $K^1$  Equation:**

$$\left(\partial_t^{(1)} + \vec{e}_\alpha \cdot \vec{\nabla}^{(1)}\right) f_\alpha^{eq} = -\frac{1}{\tau} f_\alpha^{(1)}$$

$\Rightarrow$  First-order non-equilibrium:

$$f_\alpha^{(1)} = -\tau \left(\partial_t^{(1)} + \vec{e}_\alpha \cdot \vec{\nabla}^{(1)}\right) f_\alpha^{eq}$$

## Maxwell-Boltzmann Equilibrium (Hermite Expansion):

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

Derivatives of  $f_{\alpha}^{eq}$ :

$$\frac{\partial f_{\alpha}^{eq}}{\partial u_i} \approx \frac{e_{\alpha}^i - u_i}{c_s^2} f_{\alpha}^{eq}, \quad \frac{\partial f_{\alpha}^{eq}}{\partial \rho} = \frac{f_{\alpha}^{eq}}{\rho}$$

Conservation Laws (1st order scale):

$$\partial_t^{(1)} \rho = -\vec{\nabla}^{(1)} \cdot (\rho \vec{u})$$

$$\partial_t^{(1)} \vec{u} = -\vec{u} \cdot \vec{\nabla}^{(1)} \vec{u} - \frac{1}{\rho} \vec{\nabla}^{(1)} p$$

## Assumptions at Boundary:

- ① Constant pressure:  $\nabla p = 0$
- ② Incompressible:  $\nabla \rho = 0$

## Distribution Function at Boundary:

$$f_{\alpha}|_{bc} = f_{\alpha}^{eq} + \omega \delta t \left( \frac{(e_{\alpha}^i - u_i)(e_{\alpha}^{\beta} - u_{\beta})}{c_s^2} \frac{\partial u_i}{\partial x_{\beta}} - \frac{\partial u_{\beta}}{\partial x_{\beta}} \right) f_{\alpha}^{eq}$$

## Velocity Gradient in Curvilinear Coordinates:

$$\frac{\partial u_i}{\partial x_{\beta}} = \frac{\partial u_i}{\partial \xi_j} \cdot \frac{\partial \xi_j}{\partial x_{\beta}}$$

## 2nd Order One-Side Finite Difference:

$$\frac{\partial u}{\partial \xi} \Big|_{i+\frac{1}{2}} = \frac{3u_i - 4u_{i-1} + u_{i-2}}{2\Delta \xi} + O(\Delta \xi^2)$$

**GILBM Initialization:**

- ➊ Set  $Re$ , relaxation factor  $\omega$
- ➋ Contravariant velocity:

$$e_{\alpha}^i = c_{\alpha}^{\beta} \frac{\partial \xi_i}{\partial x_{\beta}}$$

- ➌ Global time step (CFL):

$$\Delta t = \text{CFL} \cdot \min \left| \frac{1}{e_{\alpha}^i} \right|$$

- ➍ Relaxation time:  $\tau = \Delta t / \omega$
- ➎ Viscosity:  $\nu = (\tau - 0.5\Delta t)c_s^2$

**Runtime Loop:**

- ➏ **Collision:**  $f_i^* = f_i + \omega(f_i^{eq} - f_i)$
- ➐ **Streaming:**  $f_i(\vec{\xi}) = f_i^*(\vec{\xi} - \Delta \vec{\xi}_i)$  (with interpolation)
- ➑ **Boundary:** Apply  $f_{\alpha}|_{bc}$
- ➒ **Macroscopic:**  $\rho = \sum_i f_i, \vec{u} = \frac{1}{\rho} \sum_i f_i \vec{c}_i$

Pre-calculation reduces runtime by  $\sim 50\%$

### GILBM: Key Points

- ① **Coordinate Transform:** Jacobian relation maps Cartesian  $\leftrightarrow$  curvilinear
- ② **Collision:** Direct substitution  $\vec{x} \rightarrow \vec{\xi}$  (no spatial derivatives)
- ③ **Streaming:** Path integration with contravariant velocity  $e_\alpha^j$
- ④ **Accuracy:** 2nd order RK required for N-S recovery
- ⑤ **Boundary:** Distribution from Chapman-Enskog with  $\nabla p = 0, \nabla \rho = 0$
- ⑥ **Efficiency:** Pre-calculated contravariant velocity saves  $\sim 50\%$  runtime

**Limitation:** MRT operator extension to curvilinear coordinates not yet developed.