Contents

1 Streaming Step

During the streaming step, distributions are propagated into the neighbour cells in the corresponding direction. There are several ways to implement the streaming step in memory. The simplest approach is the A-B streaming pattern, which uses two distribution fields. Distributions are read from old locations in field A and written into new locations in field B. This doubles memory consumption.

To prevent the second field, it is necessary to find an algorithm that always writes data into the same locations where it read from. One of such algorithms is the A-A streaming, which combines streaming and collision steps together and alters between odd and even timesteps.

There is also another way that does not require altering between odd and even timesteps presented in [4], called Periodic Shift.

1.1 Periodic Shift Streaming

Article [4] presents the algorithm ”Periodic Shift” (PS). This algorithm requires the ”Structure of Arrays” (SoA) organization so that each distribution array $\underline{f_q}$ is saved separately in a continous block of memory. Instead of physically writing data into new positions during the streaming step, it uses periodic shifting of the memory index, independent for each distribution.

1.1.1 Basic idea

Before formalizing the algorithm, let's illustrate the idea on a D2Q9 scheme (Figure 1):

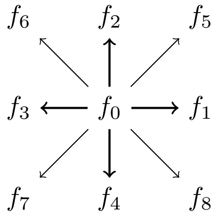

Figure 1: D2Q9 scheme

Consider a field with 3 rows and 4 columns with cells labelled using row-major indexing as shown in Figure 2:

0	1	2	3
4	5	6	7
8	9	10	11

Figure 2: D2Q9 field with cell labels

During streaming, values of each distribution array $\underline{f_q}$ move through the field in the direciton given by the velocity set. This means, after streaming, in cell $\overline{2}$ we want to find the original value of f_1 from cell 1, original value of f_2 from cell 6, etc. Values f_4, f_7, f_8 are unknown (”?”) in cell 2 and need to be filled by a boundary condition. Table 1 shows, for every cell and every distribution f_q, from where the value was streamed.

cell i	0	1	2	3	4	5	6	7	8	9	10	11
f_q	**In cell i, where was the value of f_q streamed from?**											
f_0	0	1	2	3	4	5	6	7	8	9	10	11
f_1	?	0	1	2	?	4	5	6	?	8	9	10
f_2	4	5	6	7	8	9	10	11	?	?	?	?
f_3	1	2	3	?	5	6	7	?	9	10	11	?
f_4	?	?	?	?	0	1	2	3	4	5	6	7
f_5	?	4	5	6	?	8	9	10	?	?	?	?
f_6	5	6	7	?	9	10	11	?	?	?	?	?
f_7	?	?	?	?	1	2	3	?	5	6	7	?
f_8	?	?	?	?	?	0	1	2	?	4	5	6

Table 1: Streaming map

Taking a look at Table 1, it seems that the cells always stay ordered. During streaming, they are all just shifted by a constant, which is different for every distribution. This is used by the PS algorithm.

1.1.2 Full PS algorithm

Let's now consider a 3 dimensional case, using the $D3Q27$ scheme with the following velocity set (Table 2):

Table 2: D3Q27 velocity set

	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{18}	f_{19}	f_{20}	f_{21}	f_{22}	f_{23}	f_{24}	f_{25}	f_{26}
c_x	0	1	−1	0	0	0	0	1	−1	1	−1	−1	1	0	0	−1	1	0	0	−1	1	−1	1	1	−1	−1	1
c_y	0	0	0	0	0	−1	1	0	0	0	0	−1	1	1	−1	1	−1	1	−1	1	−1	−1	1	−1	1	−1	1
c_z	0	0	0	−1	1	0	0	−1	1	1	−1	0	0	−1	1	0	0	1	−1	−1	1	1	−1	−1	1	−1	1
w	$\frac{8}{27}$	$\frac{2}{27}$	$\frac{2}{27}$	$\frac{2}{27}$	$\frac{2}{27}$	$\frac{2}{27}$	$\frac{2}{27}$	$\frac{1}{54}$	$\frac{1}{54}$	$\frac{1}{54}$	$\frac{1}{54}$	$\frac{1}{54}$	$\frac{1}{54}$	$\frac{1}{54}$	$\frac{1}{54}$	$\frac{1}{54}$	$\frac{1}{54}$	$\frac{1}{54}$	$\frac{1}{54}$	$\frac{1}{216}$	$\frac{1}{216}$	$\frac{1}{216}$	$\frac{1}{216}$	$\frac{1}{216}$	$\frac{1}{216}$	$\frac{1}{216}$	$\frac{1}{216}$

The field will be block-shaped, sized $N = N_x \cdot N_y \cdot N_z$. Cells will be indexed i, j, k to represent x, y, z position and the 3-dimensional index will be mapped onto 1D index idx as

$$idx(i, j, k) = k \cdot (N_x \cdot N_y) + j \cdot N_x + i \tag{1.1}$$

The index idx determines placement of data for each distribution array of length N in memory. To read distribution f_1 in cell i, j, k we call $f_1 = \underline{f_1}[idx(i, j, k)]$.

During the streaming step, distributions should flow into neighbour cells in directions determined by the velocity set.

A distribution value located at $idx(i, j, k)$ after streaming was placed at $idx(i - c_x, j - c_y, k - c_z)$ before streaming. This gives, independently for each $\underline{f_q}$,

$$\begin{gathered} idx(i - c_x, j - c_y, k - c_z) \\ = \\ (k - c_z) \cdot (N_x \cdot N_y) + (j - c_y) \cdot N_x + (i - c_x) \\ = \\ k \cdot (N_x \cdot N_y) + j \cdot N_x + i - [c_z \cdot (N_x \cdot N_y) + c_y \cdot N_x + c_x] \\ = \\ idx(i, j, k) - s_q \end{gathered} \tag{1.2}$$

where the s_q term is constant and only depends on q (index of the distribution):

$$s_q = c_z \cdot (N_x \cdot N_y) + c_y \cdot N_x + c_x \tag{1.3}$$

For t streaming steps,

$$idx(i - tc_x, j - tc_y, k - tc_z) = idx(i, j, k) - ts_q \tag{1.4}$$

Instead of moving the data, it is possible to just read and write it into its original location everytime it is accessed. This can be formalized by making the index idx a function of iteration number t:

$$idx(i,j,k,t) = idx(i,j,k) - ts_q \tag{1.5}$$

To avoid negative indexes or indexes higher than N, just do

$$idx(i,j,k,t) = ((idx(i,j,k) - ts_q) \mod N + N) \mod N \tag{1.6}$$

Note that this makes all distributions that enter the field from the boundaries invalid, but these would have to be fixed by boundary conditions anyway.

When implemented, this results in by 27 parts coalescent memory access during the collision step. There is only a single large jump of the pointer per every distribution array. There is no memory access during the streaming step, except for some implementation of the index shifter. There is no second field required, and there is no difference between odd and even iterations.

2 Boundary Conditions

2.1 Moment based boundary conditions

Moment based boundary conditions in 3 dimensions are investigated in [3], [2]. The idea is to approximate raw moments as their Maxwell-Boltzman equilibrium and use those in a system of equations to calculate the unknown distributions. The equilibriums are calculated from ρ, u_x, u_y and u_z. It is required that those 4 quantities are known a priori by prescribing them or by restoring them from known distributions. A general method to restore macroscopic quantities from known distributions is shown later on.

One advantage of using moment based boundary conditions is locality [3]. It is only necessary to work with information from a single cell.

Another advantage is that the prescribed quantities have clearer physical meaning compared to some other options, where part of the definition is somewhat artificial [3].

Derivation of moment based boundary conditions is explained below.

2.1.1 General procedure

Raw moments are defined as [1]

$$m_{abc} = \sum_i f_i c_{xi}^a c_{yi}^b c_{zi}^c = \sum_{\text{known } i} f_i c_{xi}^a c_{yi}^b c_{zi}^c + \sum_{\text{unknown } i} f_i c_{xi}^a c_{yi}^b c_{zi}^c \tag{2.1}$$

Each raw moment is just a linear combination of distributions f, from which some are known and some are unknown.
Equilibrium moments can be expressed as

$$m_{abc}^{eq} = \sum_i f_i^{eq} c_{xi}^a c_{yi}^b c_{zi}^c \tag{2.2}$$

where f_i^{eq} is the Maxwell-Boltzman equilibrium:

$$f_i^{\text{eq}} = w_i \rho \left[1 + 3\,\underline{c}_i \cdot \underline{u} + \frac{9}{2}\left(\underline{c}_i \cdot \underline{u}\right)^2 - \frac{3}{2}\,\underline{u} \cdot \underline{u} \right]. \tag{2.3}$$

Maxwell-Boltzman equilibrium moments relevant to the task are shown in Table 3. For use in a later chapter, there is also a linearization shown where any members with second or higher power of u are neglected, which is reasonable if Ma << 1.

Table 3: Maxwell–Boltzmann equilibrium moments and linearized equilibrium

Moment	**Maxwell–Boltzmann equilibrium**	**Linearized equilibrium**
m_{000}	ρ	ρ
m_{100}	$\rho\, u_x$	$\rho\, u_x$
m_{010}	$\rho\, u_y$	$\rho\, u_y$
m_{001}	$\rho\, u_z$	$\rho\, u_z$
m_{200}	$\frac{1}{3}\rho + \rho\, u_x^2$	$\frac{1}{3}\rho$
m_{020}	$\frac{1}{3}\rho + \rho\, u_y^2$	$\frac{1}{3}\rho$
m_{002}	$\frac{1}{3}\rho + \rho\, u_z^2$	$\frac{1}{3}\rho$
m_{011}	$\rho\, u_y u_z$	0
m_{101}	$\rho\, u_x u_z$	0
m_{110}	$\rho\, u_x u_y$	0
m_{111}	0	0
m_{210}	$\frac{1}{3}\rho\, u_y + \rho\, u_x^2 u_y$	$\frac{1}{3}\rho\, u_y$
m_{201}	$\frac{1}{3}\rho\, u_z + \rho\, u_x^2 u_z$	$\frac{1}{3}\rho\, u_z$
m_{120}	$\frac{1}{3}\rho\, u_x + \rho\, u_x u_y^2$	$\frac{1}{3}\rho\, u_x$
m_{021}	$\frac{1}{3}\rho\, u_z + \rho\, u_y^2 u_z$	$\frac{1}{3}\rho\, u_z$
m_{102}	$\frac{1}{3}\rho\, u_x + \rho\, u_x u_z^2$	$\frac{1}{3}\rho\, u_x$
m_{012}	$\frac{1}{3}\rho\, u_y + \rho\, u_y u_z^2$	$\frac{1}{3}\rho\, u_y$
m_{022}	$\frac{1}{9}\rho + \frac{1}{3}\rho\, u_y^2 + \frac{1}{3}\rho\, u_z^2 + \rho\, u_y^2 u_z^2$	$\frac{1}{9}\rho$
m_{202}	$\frac{1}{9}\rho + \frac{1}{3}\rho\, u_x^2 + \frac{1}{3}\rho\, u_z^2 + \rho\, u_x^2 u_z^2$	$\frac{1}{9}\rho$
m_{220}	$\frac{1}{9}\rho + \frac{1}{3}\rho\, u_x^2 + \frac{1}{3}\rho\, u_y^2 + \rho\, u_x^2 u_y^2$	$\frac{1}{9}\rho$
m_{211}	$\frac{1}{3}\rho\, u_y u_z$	0
m_{121}	$\frac{1}{3}\rho\, u_x u_z$	0
m_{112}	$\frac{1}{3}\rho\, u_x u_y$	0

It is required that ρ, u_x, u_y, u_z are defined a priori, then all equilibrium moments can be calculated.

The idea is to set raw moments from equation (2.1) equal to their Maxwell-Boltzman equilibrium from Table 3. This gives a system of linear equations that allows to find unknown distributions:

$$\sum_{\text{known } i} f_i c_{xi}^a c_{yi}^b c_{zi}^c + \sum_{\text{unknown } i} f_i c_{xi}^a c_{yi}^b c_{zi}^c = m_{abc}^{eq} \tag{2.4}$$

Approximating the moments as their equilibrium is exact for raw moments up to first order. For higher order moments, using the equilibrium means that the non-equilibrium part is neglected. This approximation still recovers incompressible Navier-Stokes equations correctly [2].

The specific formulation of equations depends on type and orientation of the cell. A face cell is a cell that does not have neighbours in one direction. A edge cell does not have neighbours in two directions. A corner cell misses neighbours in three directions. A general approach to the derivation process is needed to handle all different kinds of cells [2]. A section showing the equations for one of the cell orientations as an example, follows. The procedure presented below can be algorithmized and used to automatically generate code for all different cell types.

2.1.2 Corner cell example

As an example to explain the procedure, let's use the most "difficult" case of a corner cell with outer normal $(1, -1, 1)$ (so it has neighbours in X direction but misses neighbours in positive Y and negative Z). After streaming, the distributions with $c_x = -1, c_y = 1$ or $c_z = -1$ are unknown.

Let's put all known distributions into a vector $\underline{f_k}$ and unknown distributions into $\underline{f_u}$ as displayed in Table 4.

Table 4: Known and unknown distributions for a cell with $(1, -1, 1)$ outer normal

Known distributions	$\underline{f_k} = (f_0, f_1, f_4, f_5, f_9, f_{14}, f_{16}, f_{20})^T$
Unknown distributions	$\underline{f_u} = (f_2, f_3, f_6, f_7, f_8, f_{10}, f_{11}, f_{12}, f_{13}, f_{15}, f_{17}, f_{18}, f_{19}, f_{21}, f_{22}, f_{23}, f_{24}, f_{25}, f_{26})^T$

Equilibrium moments will be stored in a vector $\underline{m} \in \mathbb{R}^{23}$:

Table 5: Vector of equilibrium moments

Equilibrium moments	$\underline{m} = \begin{pmatrix} m_{000},\ m_{100},\ m_{010},\ m_{001}, \\ m_{200},\ m_{020},\ m_{002},\ m_{011},\ m_{101},\ m_{110}, \\ m_{111}, m_{210},\ m_{201},\ m_{120},\ m_{021},\ m_{102},\ m_{012}, \\ m_{022},\ m_{202},\ m_{220},\ m_{211},\ m_{121},\ m_{112} \end{pmatrix}^T$

This allows to express equation 2.4 in a matrix form:

$$\underline{\underline{\underline{M_{fk}}}}\ \underline{f_k} + \underline{\underline{\underline{M_{fu}}}}\ \underline{f_u} = \underline{m} \tag{2.5}$$

where $\underline{\underline{\underline{M_{fk}}}} \in \mathbb{Z}^{23 \times \mathrm{len}(\underline{f_k})}$ is the matrix of f_k coefficients:

$$\underline{\underline{\underline{M_{fk}}}} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & -1 & 0 & -1 & -1 & -1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \tag{2.6}$$

and $\underline{\underline{\underline{M_{fu}}}} \in \mathbb{Z}^{23\times \mathrm{len}(\underline{f_u})}$ is the matrix of f_u coefficients:

$$\underline{\underline{\underline{M_{fu}}}} = \begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
-1 & 0 & 0 & 1 & -1 & -1 & -1 & 1 & 0 & -1 & 0 & 0 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & -1 & 1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\
0 & -1 & 0 & -1 & 1 & -1 & 0 & 0 & -1 & 0 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\
1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & -1 & 0 & 0 & -1 & 1 & 1 & -1 & -1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & 1 & -1 & -1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 1 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\
0 & 0 & 0 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & -1 & 0 & 0 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & -1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\
0 & 0 & 0 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & -1 & -1 & 1 & 1
\end{bmatrix} \tag{2.7}$$

Equation (2.5) gives 23 scalar equations. For a corner cell with outer normal $(1, -1, 1)$ there are 19 unknown distributions, so from (2.5) it is necessary to pick 19 equations that have linearly independent rows of $\underline{\underline{\underline{M_{fu}}}}$. It is suggested to select those corresponding to lower order moments, as they have clearer physical meaning [3].

To select the equations, let's define a selector matrix $\underline{\underline{S}} \in \mathbb{Z}^{\mathrm{len}(\underline{f_u})\times 23}$. $\underline{\underline{S}}$ looks almost like an identity matrix, but 4 rows are left out to only keep the desired 19 equations:

$$\underline{\underline{S}} = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0
\end{bmatrix} \tag{2.8}$$

By multiplying (2.5) by $\underline{\underline{S}}$, the correct 19 equations are selected:

$$\underline{\underline{S}}\,\underline{\underline{\underline{M_{fk}}}}\,\underline{f_k} + \underline{\underline{S}}\,\underline{\underline{\underline{M_{fu}}}}\,\underline{f_u} = \underline{\underline{S}}\,\underline{m} \tag{2.9}$$

$\left(\underline{\underline{S}}\underline{\underline{\underline{M_{fu}}}}\right)$ is regular, so it can be inverted. This allows to find $\underline{f_u}$:

$$\underline{f_u} = \left(\underline{\underline{S}}\underline{\underline{\underline{M_{fu}}}}\right)^{-1} \left(\underline{\underline{S}}\underline{m} - \underline{\underline{S}}\underline{\underline{\underline{M_{fk}}}}\underline{f_k}\right) \tag{2.10}$$

As $\left(\underline{\underline{S}}\ \underline{\underline{\underline{M_{fu}}}}\right)$ is relatively small and nice, it can be inversed using a symbolic solver.

$$\left(\underline{\underline{S}}\ \underline{\underline{\underline{M_{fu}}}}\right)^{-1} =$$

$$\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
1 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 1/2 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & 0 & 0 & -1/2 \\
0 & 0 & 0 & 1/2 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & -1/2 & 0 & 0 \\
-1 & -1/2 & 0 & -1/2 & 1/2 & 1 & 1/2 & 0 & 0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 & -1/2 & 0 & -1/2 \\
0 & 0 & -1/2 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & -1/2 & 0 & 0 \\
0 & 1/2 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & -1/2 & 0 \\
-1 & 0 & 1/2 & -1/2 & 1 & 1/2 & 1/2 & 0 & 0 & 0 & -1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & -1/2 & -1/2 \\
-1 & -1/2 & 1/2 & 0 & 1/2 & 1/2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & -1/2 & -1/2 & -1/2 & 0 \\
0 & 0 & 0 & 1/2 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 & 0 & 0 & -1/2 & 0 \\
0 & 0 & -1/2 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/2 \\
1 & 1/2 & -1/2 & 1/2 & -1/2 & -1/2 & -1/2 & -1/4 & 1/4 & -1/4 & 1/4 & -1/4 & -1/4 & -1/4 & -1/4 & 1/4 & 1/4 & 1/4 & 1/4 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/4 & 0 & 0 & 0 & 0 & 0 & 1/4 & 0 & -1/4 & 1/4 & 0 & 0 \\
0 & -1/2 & 0 & 0 & -1/2 & 0 & 0 & 0 & -1/4 & 1/4 & 1/4 & -1/4 & 1/4 & 0 & 1/4 & 0 & 0 & 1/4 & 1/4 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1/4 & -1/4 & 0 & 1/4 & 0 & 0 & 0 & 0 & 0 & 1/4 \\
0 & 0 & 0 & -1/2 & 0 & 0 & -1/2 & 1/4 & -1/4 & 0 & 0 & 1/4 & 0 & 1/4 & -1/4 & 1/4 & 1/4 & 1/4 & 0 \\
0 & 0 & 1/2 & 0 & 0 & -1/2 & 0 & 1/4 & 0 & 1/4 & -1/4 & 0 & -1/4 & -1/4 & 0 & -1/4 & 1/4 & 0 & 1/4 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/4 & 0 & 0 & 1/4 & 0 & 0 & 1/4 & 0 & 0 & 1/4 & 0
\end{bmatrix} \tag{2.11}$$

2.2 Reconstruction of macroscopic quantities from known distributions

The moment based boundary conditions allow to calculate $\underline{f_u}$ from $\underline{f_k}$ and macroscopic quantities ρ, u_x, u_y, u_z. In case that not all macroscopic quantities are explicitly prescribed, it is possible to reconstruct them from $\underline{f_k}$.

2.2.1 Corner cell example with unknown ρ, u_x, u_y, u_z

Let's consider a corner cell with $(1, -1, 1)$ outer normal again. Let's first illustrate a case where no macroscopic quantity is prescribed, to show how it is possible to reconstruct all ρ, u_x, u_y, u_z just from $\underline{f_k}$.

Known and unknown distributions for such case were previously displayed in Table 4. Matrices $\underline{\underline{\underline{M_{fk}}}}$ and $\underline{\underline{\underline{M_{fu}}}}$ were listed in equations (2.6) and (2.7), respectively.

In the previous chapter, the goal was to calculate $\underline{f_u}$, which required finding linearly independeng rows of $\underline{\underline{\underline{M_{fu}}}}$. Now the situation is reversed: goal is to find at least 4 independent equations for ρ, u_x, u_y, u_z where no unknown f_i is present.

This can be achieved by forming linear combinations of the available equations. Let $\underline{c} \in \mathbb{R}^{23}$ denote a column vector of coefficients, where each component multiplies the corresponding equation. Then the unknown distributions get eliminated if for all $\underline{f_u}$ it holds that

$$\left(\underline{\underline{\underline{M_{fu}}}}\ \underline{f_u}\right) \cdot \underline{c} = 0 \tag{2.12}$$

This can be reformulated as

$$\left(\underline{\underline{\underline{M_{fu}}}}\ \underline{f_u}\right) \cdot \underline{c} = \underline{c}^T \left(\underline{\underline{\underline{M_{fu}}}}\ \underline{f_u}\right) = \left(\underline{c}^T\ \underline{\underline{\underline{M_{fu}}}}\right) \underline{f_u} = \underline{f_u}^T \left(\underline{c}^T\ \underline{\underline{\underline{M_{fu}}}}\right)^T = \underline{f_u}^T \left(\underline{\underline{\underline{M_{fu}}}}^T\ \underline{c}\right) = 0 \tag{2.13}$$

which will hold for all $\underline{f_u}$ only if

$$\underline{\underline{\underline{M_{fu}}}}^T\ \underline{c} = \underline{0} \tag{2.14}$$

Therefore, the searched coefficient vectors $\underline{c}$ are given by the null space of $\underline{\underline{M_{fu}}}^T$. Let's assemble all n linearly independent vectors $\underline{c}$ into a matrix $\underline{\underline{C}}$:

$$\underline{\underline{C}} = \begin{bmatrix} \vert & \vert & \vert & \vert \\ \underline{c}_0 & \underline{c}_1 & \dots & \underline{c}_n \\ \vert & \vert & \vert & \vert \end{bmatrix} \tag{2.15}$$

where $\underline{c_0}, \underline{c_1} \dots$ are ordered smallest to largest using the key

$$\text{key} = \sum_{i}^{\text{len}(\underline{c})} w_i, \quad w_i = \begin{cases} \text{order of moment } m_i \text{ if } c_i \neq 0 \\ 0 \text{ otherwise} \end{cases} \tag{2.16}$$

which puts low order moment combinations first. In case of outer normal $(1, -1, 1)$, $\underline{\underline{C}}$ yields

$$\underline{\underline{C}} = \begin{bmatrix} -1 & 1 & -1 & 1 \\ -1 & 0 & -1 & 1 \\ 1 & -1 & 0 & -1 \\ -1 & 1 & -1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & -1 & 0 & 0 \\ -1 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.17}$$

That suggests how to deal with f_u. To relate remaining f_k to ρ, u_x, u_y, u_z, equilibrium moments will be used once again. However, Maxwell-Boltzman equilibrium moments as shown in Table 3 are nonlinear. Therefore the linearized equilibrium will be used to provide a linear set of equations.

Let's define a vector of independent macroscopic quantities $\underline{q}$:

$$\underline{q} = (\rho, \rho u_x, \rho u_y, \rho u_z)^T \tag{2.18}$$

Then linearized equilibrium moments can be expressed as

$$\underline{m} = \underline{\underline{Q}}\, \underline{q} \tag{2.19}$$

where $\underline{\underline{Q}} \in \mathbb{Z}^{23 \times 4}$:

$$\underline{\underline{Q}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & 0 \\ 0 & 0 & 0 & \frac{1}{3} \\ 0 & \frac{1}{3} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{3} \\ 0 & \frac{1}{3} & 0 & 0 \\ 0 & 0 & \frac{1}{3} & 0 \\ \frac{1}{9} & 0 & 0 & 0 \\ \frac{1}{9} & 0 & 0 & 0 \\ \frac{1}{9} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{2.20}$$

Now, inserting (2.19) into (2.5) and multiplying both sides from the left by $\underline{\underline{C}}^T$ gives

$$\underline{\underline{C}}^T \left(\underline{\underline{M_{fk}}}\ \underline{f_k} + \underline{\underline{M_{fu}}}\ \underline{f_u} \right) = \underline{\underline{C}}^T\ \underline{\underline{Q}}\ \underline{q} \tag{2.21}$$

As follows from above, $\underline{\underline{C}}^T$ is built so that $\underline{f_u}$ gets cancelled, so

$$\underline{\underline{C}}^T \underline{\underline{M_{fk}}}\ \underline{f_k} = \underline{\underline{C}}^T\ \underline{\underline{Q}}\ \underline{q} \tag{2.22}$$

Equation (2.22) provides n scalar equations for n vectors $\underline{c}$. The unknown is $\underline{q}$. In a general case it may happen that n is higher than 4 which is the number of unknowns. Therefore we need to select such equations that have linearly independent rows of $\left(\underline{\underline{C}}^T\ \underline{\underline{Q}} \right)$ and which use the lowest order moments. Thanks to the ordering (2.16), the preferred rows are placed on top.

To select the equations, let's use a new selector matrix $\underline{\underline{S_q}} \in \mathbb{Z}^{4 \times \text{columns}(\underline{\underline{C}})}$. In case of outer normal $(1, -1, 1)$, $\underline{\underline{S}}$ yields:

$$\underline{\underline{S_q}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.23}$$

Multiplying (2.22) by $\underline{\underline{S_q}}$ selects the correct equations:

$$\underline{\underline{S_q}}\underline{\underline{C}}^T \underline{\underline{M_{fk}}}\ \underline{f_k} = \underline{\underline{S_q}}\underline{\underline{C}}^T \underline{\underline{Q}}\ \underline{q} \tag{2.24}$$

where $\left(\underline{\underline{S_q}}\underline{\underline{C}}^T \underline{\underline{Q}} \right)$ is regular, so it can be inverted. This allows to find $\underline{q}$:

$$\underline{q} = \left(\underline{\underline{S_q}}\underline{\underline{C}}^T \underline{\underline{Q}} \right)^{-1} \underline{\underline{S_q}}\underline{\underline{C}}^T \underline{\underline{M_{fk}}}\ \underline{f_k} \tag{2.25}$$

As $\left(\underline{\underline{S_q}}\,\underline{\underline{C}}^T\underline{\underline{Q}}\right)$ is relatively small and nice, it can be inversed using a symbolic solver.

$$\left(\underline{\underline{S_q}}\,\underline{\underline{C}}^T\underline{\underline{Q}}\right)^{-1} = \begin{bmatrix} 2 & 3/2 & -3/2 & 3/2 \\ -1 & -3/2 & 0 & 0 \\ 1 & 0 & -3/2 & 0 \\ -1 & 0 & 0 & -3/2 \end{bmatrix} \tag{2.26}$$

2.2.2 Corner cell example with prescribed ρ or u_x, u_y, u_z

Let's modify the procedure for a case when either ρ is given or u_x, u_y, u_z are given. Compared to the previous derivation, $\underline{f_k}, \underline{f_u}, \underline{\underline{M_{fk}}}, \underline{\underline{M_{fu}}}, \underline{\underline{C}}$ stay the same. However, the vector of independent macroscopic quantities should be split in the ρ and u part. Let's define

$$\underline{u} = (u_x, u_y, u_z)^T \tag{2.27}$$

which results in new expression of $\underline{m}$:

$$\underline{m} = \rho\left(\underline{\underline{Q_u}}\;\underline{u} + \underline{Q_\rho}\right) \tag{2.28}$$

where

$$\underline{\underline{Q_u}} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \frac{1}{3} & 0 \\ 0 & 0 & \frac{1}{3} \\ \frac{1}{3} & 0 & 0 \\ 0 & 0 & \frac{1}{3} \\ \frac{1}{3} & 0 & 0 \\ 0 & \frac{1}{3} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{2.29}$$

and

$$\underline{Q_\rho} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ \frac{1}{3} \\ \frac{1}{3} \\ \frac{1}{3} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{9} \\ \frac{1}{9} \\ \frac{1}{9} \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{2.30}$$

That leads to

$$\underline{\underline{C}}^T \underline{\underline{M_{fk}}}\ \underline{f_k} = \underline{\underline{C}}^T \rho \left(\underline{\underline{Q_u}}\underline{u} + \underline{Q_\rho} \right) \tag{2.31}$$

reorganized

$$\frac{1}{\rho}\underline{\underline{C}}^T \underline{\underline{M_{fk}}}\underline{f_k} - \underline{\underline{C}}^T \underline{Q_\rho} = \underline{\underline{C}}^T \underline{\underline{Q_u}}\underline{u} \tag{2.32}$$

From here the case splits depending on if ρ or $\underline{u}$ is the unknown. If $\underline{u}$ is the unknown, we need to select such equations that have linearly independent rows of $\left(\underline{\underline{C}}^T \underline{\underline{Q_u}}\right)$. For that let's use a selector matrix $\underline{\underline{S_u}} \in \mathbb{Z}^{4 \times \text{columns}(\underline{\underline{C}})}$:

$$\underline{\underline{S_u}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{2.33}$$

Then

$$\underline{u} = \left(\underline{\underline{S_u}}\underline{\underline{C}}^T \underline{\underline{Q_u}} \right)^{-1} \left[\frac{1}{\rho}\underline{\underline{S_u}}\underline{\underline{C}}^T \underline{\underline{M_{fk}}}\underline{f_k} - \underline{\underline{S_u}}\underline{\underline{C}}^T \underline{Q_\rho} \right] \tag{2.34}$$

where

$$\left(\underline{\underline{S_u}}\underline{\underline{C}}^T\underline{\underline{Q_u}} \right)^{-1} = \begin{bmatrix} -1 & -3/2 & 0 \\ 1 & 0 & -3/2 \\ 1 & 3/2 & -3/2 \end{bmatrix} \tag{2.35}$$

If ρ is the unknown in (2.32), we can manipulate (2.32) to

$$\underline{\underline{C}}^T \underline{\underline{M_{fk}}}\underline{f_k} = \rho \left[\underline{\underline{C}}^T \underline{Q_\rho} + \underline{\underline{C}}^T \underline{\underline{Q_u}}\underline{u} \right] \tag{2.36}$$

To isolate ρ, we need to define $\underline{\underline{S_\rho}} \in \mathbb{Z}^{1 \times \text{columns}(\underline{\underline{C}})}$ so that it selects a single equation:

$$\underline{\underline{S_\rho}}\underline{\underline{C}}^T \underline{\underline{M_{fk}}}\underline{f_k} = \rho \left[\underline{\underline{S_\rho}}\underline{\underline{C}}^T \underline{Q_\rho} + \underline{\underline{S_\rho}}\underline{\underline{C}}^T \underline{\underline{Q_u}}\underline{u} \right] \tag{2.37}$$

Here we need to ensure that $\underline{\underline{S_\rho}}\,\underline{\underline{C}}^T\underline{Q_\rho} + \underline{\underline{S_\rho}}\,\underline{\underline{C}}^T\underline{\underline{Q_u}}\,\underline{u}$ is not 0. The dependency on $\underline{u}$ is uncomfortable, but by looking at size of elements in $\underline{Q_\rho}, \underline{\underline{Q_u}}$ and assuming that Ma << 1, it should be sufficient to ensure that $\underline{\underline{S_\rho}}\,\underline{\underline{C}}^T\underline{Q_\rho}$ is not 0. This is satisfied for the present case by

$$\underline{\underline{S_\rho}} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \tag{2.38}$$

Then

$$\rho = \left(\underline{\underline{S_\rho}}\,\underline{\underline{C}}^T\underline{Q_\rho} + \underline{\underline{S_\rho}}\,\underline{\underline{C}}^T\underline{\underline{Q_u}}\,\underline{u}\right)^{-1} \underline{\underline{S_\rho}}\,\underline{\underline{C}}^T\underline{\underline{M_{fk}}}\,\underline{f_k} \tag{2.39}$$

Note that $\left(\underline{\underline{S_\rho}}\,\underline{\underline{C}}^T\underline{Q_\rho} + \underline{\underline{S_\rho}}\,\underline{\underline{C}}^T\underline{\underline{Q_u}}\,\underline{u}\right)$ is a scalar, so there is no matrix inversion needed.

3 Classification of cells

Cell types are sorted in Table 6:

Cell type	**Placement**	**Given**	**Local**	**Algorithm**
Ignore cell	Anywhere	None	Yes	None
Fluid	Interior	None	Yes	1. Read $\underline{f}$ 2. Calculate ρ, u_x, u_y, u_z from $\underline{f}$ 3. Apply collision + forcing 4. Write $\underline{f}$
BC Bounceback	Anywhere	None	Yes	1. Read $\underline{f}$ 2. Apply bounceback 3. Write $\underline{f}$
BC FreeFlow	Face, Edge, Corner	None	Yes	1. Read $\underline{f_k}$ 2. Restore ρ, u_x, u_y, u_z from $\underline{f_k}$ 3. Apply MBBC 4. Apply collision + forcing 5. Write $\underline{f}$
BC Pressure	Face, Edge, Corner	ρ	Yes	1. Read $\underline{f_k}$ 2. Restore u_x, u_y, u_z from $\underline{f_k}$ 3. Apply MBBC 4. Apply collision + forcing 5. Write $\underline{f}$
BC Velocity	Face, Edge, Corner	u_x, u_y, u_z	Yes	1. Read $\underline{f_k}$ 2. Restore ρ from $\underline{f_k}$ 3. Apply MBBC 4. Apply collision + forcing 5. Write $\underline{f}$
BC PressureVelocity	Face, Edge, Corner	ρ, u_x, u_y, u_z	Yes	1. Read $\underline{f_k}$ 2. Apply MBBC 3. Apply collision + forcing 4. Write $\underline{f}$

Table 6: Classification of cell types

4 References

[1] Alessandro De Rosis. Nonorthogonal central-moments-based lattice boltzmann scheme in three dimensions. *Phys. Rev. E*, 95:013310, Jan 2017.

[2] Pavel Eichler, Radek Fučík, and Pavel Strachota. Investigation of mesoscopic boundary conditions for lattice boltzmann method in laminar flow problems. *Computers & Mathematics with Applications*, 173:87–101, 2024.

[3] Ivars Krastins, Andrew Kao, Koulis Pericleous, and Tim Reis. Moment-based boundary conditions for straight on-grid boundaries in three dimensional lattice boltzmann simulations. *International Journal for Numerical Methods in Fluids*, 92:1948–1974, 05 2020.

[4] Adrian Kummerländer, Márcio Dorn, Martin Frank, and Mathias J. Krause. Implicit propagation of directly addressed grids in lattice boltzmann methods. *Concurrency and Computation: Practice and Experience*, 35(8):e7509, 2023.