

# 1 Flux Computation

The main part of the Gas-Kinetic-Scheme Implementation, is the Flux Computation. This Computation starts with the analytical solution of the BGK-Equation by the method of characteristics. The solution consists of two part. The first part is the way towards equilibrium, whereas the second part is the decay of the initial distribution.

$$f(x_{i+1/2}, y_j, t, u, v, \xi) = \frac{1}{\tau} \int_0^t g(x', y', t', u, v, \xi) e^{-\frac{t-t'}{\tau}} dt' + f_0(x_{i+1/2} - ut, y_j - vt) e^{-\frac{t}{\tau}} \quad (1)$$

In this equation,  $g$  denotes the equilibrium distribution function at the interface and  $f_0$  is the distribution function at the interface at the beginning of the time step.

$$f_0(x, y, u, v, \xi) = g_0(u, v, \xi) \cdot \left( \underbrace{1 + a x + b y}_{\text{equilibrium}} - \underbrace{\tau(a u + b v + A)}_{\text{non-equilibrium}} \right) \quad (2)$$

$$g(x, y, t, u, v, \xi) = g_0(u, v, \xi) \cdot (1 + a x + b y + A t) \quad (3)$$

$$g_0(u, v, \xi) = \rho_0 \left( \frac{\lambda_0}{\pi} \right)^{\frac{K+2}{2}} e^{-\lambda_0((u-U_0)^2 + (v-V_0)^2 + \xi^2)} \quad \int_{-\infty}^{\infty} g du dv d\xi = \left[ \frac{kg}{m^3} \right] \quad (4)$$

The coefficients  $a$ ,  $b$  and  $A$  are the normalized gradients of the equilibrium distribution:

$$a = \frac{1}{g} \frac{\partial g}{\partial x} \Big|_{x=0} = \left[ \frac{1}{m} \right] \quad , \quad b = \frac{1}{g} \frac{\partial g}{\partial y} \Big|_{y=0} = \left[ \frac{1}{m} \right] \quad , \quad A = \frac{1}{g} \frac{\partial g}{\partial t} \Big|_{t=0} = \left[ \frac{1}{s} \right] \quad (5)$$

These coefficients are than itself expressed as expansions of the microscopic variables:

$$\begin{aligned} a &= a_1 + a_2 u + a_3 v + a_4 \frac{1}{2} (u^2 + v^2 + \xi^2) \\ b &= b_1 + b_2 u + b_3 v + b_4 \frac{1}{2} (u^2 + v^2 + \xi^2) \\ A &= A_1 + A_2 u + A_3 v + A_4 \frac{1}{2} (u^2 + v^2 + \xi^2) \end{aligned} \quad (6)$$

Explanation from Prendergast and Xu (1993):

$$\chi_\alpha = (-\ln(A), -2\lambda U, -2\lambda V, 2\lambda)^T \quad \text{with } A = \frac{1}{\rho} \left( \frac{\lambda}{\pi} \right)^{-\frac{K+2}{2}} \quad (7)$$

$$\psi_\alpha = \left( 1, u, v, \frac{1}{2}(u^2 + v^2 + \xi^2) \right)^T \quad (8)$$

$$g = e^{-\chi_\alpha \psi_\alpha} = \frac{1}{A} e^{-\lambda(u^2 + v^2 + \xi^2 - 2uU - 2vV)} \text{ which is not equivalent to the above definition of } g \quad (9)$$

$$a = \frac{1}{g} \frac{\partial g}{\partial x} = \frac{1}{g} \frac{\partial g}{\partial \chi_\alpha} \frac{\partial \chi_\alpha}{\partial x} = \frac{1}{g} \psi_\alpha g \frac{\partial \chi_\alpha}{\partial x} = \psi_\alpha \frac{\partial \chi_\alpha}{\partial x} = a_\alpha \psi_\alpha \quad (10)$$

## Integration of the Analytical Solution

These are inserted into the analytical solution with  $x' = -u(t - t')$ ,  $x = -ut$ ,  $y' = -v(t - t')$  and  $y = -vt$ :

$$\begin{aligned} f &= \frac{g_0}{\tau} \int_0^t (1 - a u(t - t') - b v(t - t') + A t') e^{-\frac{t-t'}{\tau}} dt' \\ &+ g_0 (1 - a ut - b vt - \tau (a u + b v + A)) e^{-\frac{t}{\tau}} \end{aligned} \quad (11)$$

The Integral can be solved with integration by parts:

$$\begin{aligned} f &= \frac{g_0}{\tau} \left[ \tau (1 - a u(t - t') - b v(t - t') + A t') e^{-\frac{t-t'}{\tau}} \right]_0^t \\ &- \frac{g_0}{\tau} \int_0^t \tau (a u + b v + A) e^{-\frac{t-t'}{\tau}} dt' \\ &+ g_0 (1 - a ut - b vt - \tau (a u + b v + A)) e^{-\frac{t}{\tau}} \end{aligned} \quad (12)$$

The remaining integral can than be solved directly:

$$\begin{aligned} f &= \frac{g_0}{\tau} \left[ \tau (1 - a u(t - t') - b v(t - t') + A t') e^{-\frac{t-t'}{\tau}} \right]_0^t \\ &- \frac{g_0}{\tau} \left[ \tau^2 (a u + b v + A) e^{-\frac{t-t'}{\tau}} \right]_0^t \\ &+ g_0 (1 - a ut - b vt - \tau (a u + b v + A)) e^{-\frac{t}{\tau}} \end{aligned} \quad (13)$$

Then the limits of the integrals are applied.

$$\begin{aligned} f &= g_0 (1 + A t) - g_0 (1 - a ut - b vt) e^{-\frac{t}{\tau}} \\ &- g_0 \tau (a u + b v + A) + g_0 \tau (a u + b v + A) e^{-\frac{t}{\tau}} \\ &+ g_0 (1 - a ut - b vt - \tau (a u + b v + A)) e^{-\frac{t}{\tau}} \end{aligned} \quad (14)$$

After sorting

$$\begin{aligned} f &= g_0 (1 - \tau (a u + b v + A) + A t) \\ &- g_0 (1 - a ut - b vt - \tau (a u + b v + A)) e^{-\frac{t}{\tau}} \\ &+ g_0 (1 - a ut - b vt - \tau (a u + b v + A)) e^{-\frac{t}{\tau}} \end{aligned} \quad (15)$$

the last terms cancel out and the distribution function on the interface can be computed as:

$$f = g_0 (1 - \tau (a u + b v) + (t - \tau) A) \quad (16)$$

## Computation of Expansion Coefficients

Ansatz:

$$W_\alpha = \int \psi_\alpha g d\Xi = (\rho, \rho U, \rho V, \rho E)^T \quad (17)$$

$$\frac{\partial W_\alpha}{\partial x} = \int \psi_\alpha a g d\Xi = \underbrace{\int \psi_\alpha \psi_\beta g d\Xi}_{\rho M_{\alpha,\beta}} a_\beta \quad \Rightarrow \quad \left( \frac{1}{\rho} \frac{\partial W_\alpha}{\partial x} \right) = M_{\alpha,\beta} a_\beta \quad (18)$$

This system of linear equations can be solved analytically for  $a_\beta$ . The constants  $a_i$ ,  $b_i$  and  $A_i$  in the Taylor expansion of the Maxwellian distribution function are computed as:

$$\begin{aligned} A &= 2 \left( \frac{1}{\rho} \frac{\partial(\rho E)}{\partial x} \right) - \overbrace{\left( U^2 + V^2 + \frac{K+2}{2\lambda} \right)}^{2E} \left( \frac{1}{\rho} \frac{\partial \rho}{\partial x} \right) \left( = 2 \frac{\partial E}{\partial x} \right) = \left[ \frac{m}{s^2} \right] \\ B &= \left( \frac{1}{\rho} \frac{\partial(\rho U)}{\partial x} \right) - U \left( \frac{1}{\rho} \frac{\partial \rho}{\partial x} \right) \left( = \frac{\partial U}{\partial x} \right) = \left[ \frac{1}{s} \right] \\ C &= \left( \frac{1}{\rho} \frac{\partial(\rho V)}{\partial x} \right) - V \left( \frac{1}{\rho} \frac{\partial \rho}{\partial x} \right) \left( = \frac{\partial V}{\partial x} \right) = \left[ \frac{1}{s} \right] \end{aligned} \quad (19)$$

$$\begin{aligned} a_4 &= \frac{4\lambda^2}{K+2} (A - 2UB - 2VC) = \left[ \frac{s^2}{m^3} \right] \\ a_3 &= 2\lambda C - V a_4 = \left[ \frac{s}{m^2} \right] \\ a_2 &= 2\lambda B - U a_4 = \left[ \frac{s}{m^2} \right] \\ a_1 &= \left( \frac{1}{\rho} \frac{\partial \rho}{\partial x} \right) - U a_2 - V a_3 - \frac{1}{2} \left( U^2 + V^2 + \frac{K+2}{2\lambda} \right) a_4 = \left[ \frac{1}{m} \right] \end{aligned} \quad (20)$$

This computation is here only shown for the normal variation constants  $a_i$ . For the tangential and time variation the respective derivatives of the macroscopic variables need to be used in the above formulas. The values of the primitive variables are computed by third order interpolation in the normal direction of the interface.  $W_+$  denotes the cell on the positive side of the interface.

$$\underline{Z}_{i+\frac{1}{2},j} = \frac{7}{12} (\underline{Z}_{i,j} + \underline{Z}_{i+1,j}) - \frac{1}{12} (\underline{Z}_{i-1,j} + \underline{Z}_{i+2,j}) \quad \text{with} \quad \underline{Z} = (\rho, U, V, \lambda)^T \quad (21)$$

For the boundary Interfaces on linear Interpolation is used, since only one layer of Ghost-Cells is introduced.

The spacial derivatives of the conservative variables are computed from finite differences between then cell values. The normal derivatives are computed from the third order interpolation:

$$\left(\frac{1}{\rho} \frac{\partial W}{\partial x}\right)_{i+\frac{1}{2},j} = \frac{\frac{5}{4} (W_{i+1,j} - W_{i,j}) - \frac{1}{12} (W_{i+2,j} - W_{i-1,j})}{\rho \Delta x} \quad (22)$$

At the boundaries a again only first order interpolation is used. The tangential derivatives are computed by linear interpolation:

$$\left(\frac{1}{\rho} \frac{\partial W}{\partial y}\right)_{i+\frac{1}{2},j} = \frac{(W_{i,j+1} + W_{i+1,j+1}) - (W_{i,j} + W_{i+1,j})}{4\rho \Delta x} \quad (23)$$

For the time derivative a more complex computation is needed. The time derivatives can be expressed as:

$$\begin{pmatrix} \frac{\partial \rho}{\partial t} \\ \frac{\partial(\rho V)}{\partial t} \\ \frac{\partial(\rho U)}{\partial t} \\ \frac{\partial(\rho E)}{\partial t} \end{pmatrix} = - \int_{-\infty}^{\infty} \begin{pmatrix} 1 \\ u \\ v \\ \frac{1}{2}(u^2 + v^2 + \xi^2) \end{pmatrix} (au + bv) g \, dudvd\xi \quad (24)$$

The integral over all velocities can be computed analytically, since it is just a linear combination of different moments of the Maxwellian distribution function.

$$\rho \langle u \rangle = \int_{-\infty}^{\infty} u g \, dudvd\xi \quad , \quad \rho \langle u^2 \rangle = \int_{-\infty}^{\infty} u^2 g \, dudvd\xi \quad , \quad \dots \quad (25)$$

$$\rho \langle u^\alpha v^\beta \rangle = \int_{-\infty}^{\infty} u^\alpha v^\beta g \, dudvd\xi = \rho \left( \frac{1}{\rho} \int_{-\infty}^{\infty} u^\alpha g \, dudvd\xi \frac{1}{\rho} \int_{-\infty}^{\infty} v^\beta g \, dudvd\xi \right) = \rho \langle u^\alpha \rangle \langle v^\beta \rangle \quad (26)$$

With these relations the integral can be computed as:

$$\begin{aligned} \left(\frac{1}{\rho} \frac{\partial \rho}{\partial t}\right) &= - \left( \begin{aligned} &a_1 \langle u \rangle + a_2 \langle u^2 \rangle + a_3 \langle u \rangle \langle v \rangle + a_4 \frac{1}{2} \left( \langle u^3 \rangle + \langle u \rangle \langle v^2 \rangle + \langle u \rangle \langle \xi^2 \rangle \right) \\ &+ b_1 \langle v \rangle + b_2 \langle u \rangle \langle v \rangle + b_3 \langle v^2 \rangle + b_4 \frac{1}{2} \left( \langle u^2 \rangle \langle v \rangle + \langle v^3 \rangle + \langle v \rangle \langle \xi^2 \rangle \right) \end{aligned} \right) \\ \left(\frac{1}{\rho} \frac{\partial(\rho U)}{\partial t}\right) &= - \left( \begin{aligned} &a_1 \langle u^2 \rangle + a_2 \langle u^3 \rangle + a_3 \langle u^2 \rangle \langle v \rangle + a_4 \frac{1}{2} \left( \langle u^4 \rangle + \langle u^2 \rangle \langle v^2 \rangle + \langle u^2 \rangle \langle \xi^2 \rangle \right) \\ &b_1 \langle u \rangle \langle v \rangle + b_2 \langle u^2 \rangle \langle v \rangle + b_3 \langle u \rangle \langle v^2 \rangle + b_4 \frac{1}{2} \left( \langle u^3 \rangle \langle v \rangle + \langle u \rangle \langle v^3 \rangle + \langle u \rangle \langle v \rangle \langle \xi^2 \rangle \right) \end{aligned} \right) \end{aligned}$$

$$\left(\frac{1}{\rho} \frac{\partial(\rho V)}{\partial t}\right) = - \left( \begin{aligned} & a_1 \langle u \rangle \langle v \rangle + a_2 \langle u^2 \rangle \langle v \rangle + a_3 \langle u \rangle \langle v^2 \rangle + a_4 \frac{1}{2} \left( \langle u^3 \rangle \langle v \rangle + \langle u \rangle \langle v^3 \rangle + \langle u \rangle \langle v \rangle \langle \xi^2 \rangle \right) \\ & b_1 \langle v^2 \rangle + b_2 \langle u \rangle \langle v^2 \rangle + b_3 \langle v^3 \rangle + b_4 \frac{1}{2} \left( \langle u^2 \rangle \langle v^2 \rangle + \langle v^4 \rangle + \langle v^2 \rangle \langle \xi^2 \rangle \right) \end{aligned} \right)$$

$$\left(\frac{1}{\rho} \frac{\partial(\rho E)}{\partial t}\right) = - \left( \begin{aligned} & a_1 \frac{1}{2} \left( \langle u^3 \rangle + \langle u \rangle \langle v^2 \rangle + \langle u \rangle \langle \xi^2 \rangle \right) \\ & + a_2 \frac{1}{2} \left( \langle u^4 \rangle + \langle u^2 \rangle \langle v^2 \rangle + \langle u^2 \rangle \langle \xi^2 \rangle \right) \\ & + a_3 \frac{1}{2} \left( \langle u^3 \rangle \langle v \rangle + \langle u \rangle \langle v^3 \rangle + \langle u \rangle \langle v \rangle \langle \xi^2 \rangle \right) \\ & + a_4 \frac{1}{2} \left( \frac{1}{2} \left( \langle u^5 \rangle + \langle u \rangle \langle v^4 \rangle + \langle u \rangle \langle \xi^4 \rangle \right) + \langle u^3 \rangle \langle v^2 \rangle + \langle u^3 \rangle \langle \xi^2 \rangle + \langle u \rangle \langle v^2 \rangle \langle \xi^2 \rangle \right) \\ & + b_1 \frac{1}{2} \left( \langle u^2 \rangle \langle v \rangle + \langle v^3 \rangle + \langle v \rangle \langle \xi^2 \rangle \right) \\ & + b_2 \frac{1}{2} \left( \langle u^3 \rangle \langle v \rangle + \langle u \rangle \langle v^2 \rangle + \langle u \rangle \langle v \rangle \langle \xi^2 \rangle \right) \\ & + b_3 \frac{1}{2} \left( \langle u^2 \rangle \langle v^2 \rangle + \langle v^4 \rangle + \langle v^2 \rangle \langle \xi^2 \rangle \right) \\ & + b_4 \frac{1}{2} \left( \frac{1}{2} \left( \langle u^4 \rangle \langle v \rangle + \langle v^5 \rangle + \langle v \rangle \langle \xi^4 \rangle \right) + \langle u^2 \rangle \langle v^3 \rangle + \langle u^2 \rangle \langle v \rangle \langle \xi^2 \rangle + \langle v^3 \rangle \langle \xi^2 \rangle \right) \end{aligned} \right)$$

The Flux over the interface is computed from the distribution function on the interface.

$$\underline{F} = \int_{-\infty}^{\infty} u \begin{pmatrix} 1 \\ u \\ v \\ \frac{1}{2}(u^2 + v^2 + \xi^2) \end{pmatrix} \left( \rho - \tau(a u + b v) + (t - \tau)A \right) g \, du dv d\xi$$

The Flux is split into three parts for the analytical time integration over one time step.

$$\underline{F} = \underline{F}^1 - \tau \underline{F}^2 + (t - \tau) \underline{F}^3 \quad \Rightarrow \quad \int_{t_n}^{t_{n+1}} \underline{F} dt = \Delta t \underline{F}^1 - \tau \Delta t \underline{F}^2 + \left( \frac{\Delta t^2}{2} - \tau \Delta t \right) \underline{F}^3$$

$$\underline{F}^1 = \rho \begin{pmatrix} \langle u \rangle \\ \langle u^2 \rangle \\ \langle uv \rangle \\ \frac{1}{2}(\langle u^3 \rangle + \langle u \rangle \langle v^2 \rangle + \langle u \rangle \langle \xi^2 \rangle) \end{pmatrix}$$

$$\underline{F}^2 = \rho \begin{pmatrix} \left[ \begin{array}{l} a_1 \langle u^2 \rangle + a_2 \langle u^3 \rangle + a_3 \langle u^2 \rangle \langle v \rangle + a_4 \frac{1}{2} (\langle u^4 \rangle + \langle u^2 \rangle \langle v^2 \rangle + \langle u^2 \rangle \langle \xi^2 \rangle) \\ + b_1 \langle u \rangle \langle v \rangle + b_2 \langle u^2 \rangle \langle v \rangle + b_3 \langle u \rangle \langle v^2 \rangle + b_4 \frac{1}{2} (\langle u^3 \rangle \langle v \rangle + \langle u \rangle \langle v^3 \rangle + \langle u \rangle \langle v \rangle \langle \xi^2 \rangle) \end{array} \right] \\ \left[ \begin{array}{l} a_1 \langle u^3 \rangle + a_2 \langle u^4 \rangle + a_3 \langle u^3 \rangle \langle v \rangle + a_4 \frac{1}{2} (\langle u^5 \rangle + \langle u^3 \rangle \langle v^2 \rangle + \langle u^3 \rangle \langle \xi^2 \rangle) \\ + b_1 \langle u^2 \rangle \langle v \rangle + b_2 \langle u^3 \rangle \langle v \rangle + b_3 \langle u^2 \rangle \langle v^2 \rangle + b_4 \frac{1}{2} (\langle u^4 \rangle \langle v \rangle + \langle u^2 \rangle \langle v^3 \rangle + \langle u^2 \rangle \langle v \rangle \langle \xi^2 \rangle) \end{array} \right] \\ \left[ \begin{array}{l} a_1 \langle u^2 \rangle \langle v \rangle + a_2 \langle u^3 \rangle \langle v \rangle + a_3 \langle u^2 \rangle \langle v^2 \rangle + a_4 \frac{1}{2} (\langle u^4 \rangle \langle v \rangle + \langle u^2 \rangle \langle v^3 \rangle + \langle u^2 \rangle \langle v \rangle \langle \xi^2 \rangle) \\ + b_1 \langle u \rangle \langle v^2 \rangle + b_2 \langle u^2 \rangle \langle v^2 \rangle + b_3 \langle u \rangle \langle v^3 \rangle + b_4 \frac{1}{2} (\langle u^3 \rangle \langle v^2 \rangle + \langle u \rangle \langle v^4 \rangle + \langle u \rangle \langle v^2 \rangle \langle \xi^2 \rangle) \end{array} \right] \\ \frac{1}{2} \left( \begin{array}{l} a_1 (\langle u^4 \rangle + \langle u^2 \rangle \langle v^2 \rangle + \langle u^2 \rangle \langle \xi^2 \rangle) \\ + a_2 (\langle u^5 \rangle + \langle u^3 \rangle \langle v^2 \rangle + \langle u^3 \rangle \langle \xi^2 \rangle) \\ + a_3 (\langle u^4 \rangle \langle v \rangle + \langle u^2 \rangle \langle v^3 \rangle + \langle u^2 \rangle \langle v \rangle \langle \xi^2 \rangle) \\ + a_4 \left( \frac{1}{2} (\langle u^6 \rangle + \langle u^2 \rangle \langle v^4 \rangle + \langle u^2 \rangle \langle \xi^4 \rangle) + \langle u^4 \rangle \langle v^2 \rangle + \langle u^4 \rangle \langle \xi^2 \rangle + \langle u^2 \rangle \langle v^2 \rangle \langle \xi^2 \rangle) \right. \\ + b_1 (\langle u^3 \rangle \langle v \rangle + \langle u \rangle \langle v^3 \rangle + \langle u \rangle \langle v \rangle \langle \xi^2 \rangle) \\ + b_2 (\langle u^4 \rangle \langle v \rangle + \langle u^2 \rangle \langle v^3 \rangle + \langle u^2 \rangle \langle v \rangle \langle \xi^2 \rangle) \\ + b_3 (\langle u^3 \rangle \langle v^2 \rangle + \langle u \rangle \langle v^4 \rangle + \langle u \rangle \langle v^2 \rangle \langle \xi^2 \rangle) \\ + b_4 \left( \frac{1}{2} (\langle u^5 \rangle \langle v \rangle + \langle u \rangle \langle v^5 \rangle + \langle u \rangle \langle v \rangle \langle \xi^4 \rangle) + \langle u^3 \rangle \langle v^3 \rangle + \langle u^3 \rangle \langle v \rangle \langle \xi^2 \rangle + \langle u \rangle \langle v^3 \rangle \langle \xi^2 \rangle) \right. \end{array} \left. \right) \end{pmatrix}$$

$$\underline{F}^3 = \rho \begin{pmatrix} A_1 \langle u \rangle + A_2 \langle u^2 \rangle + A_3 \langle u \rangle \langle v \rangle + A_4 \frac{1}{2} (\langle u^3 \rangle + \langle u \rangle \langle v^2 \rangle + \langle u \rangle \langle \xi^2 \rangle) \\ A_1 \langle u^2 \rangle + A_2 \langle u^3 \rangle + A_3 \langle u^2 \rangle \langle v \rangle + A_4 \frac{1}{2} (\langle u^4 \rangle + \langle u^2 \rangle \langle v^2 \rangle + \langle u^2 \rangle \langle \xi^2 \rangle) \\ A_1 \langle u \rangle \langle v \rangle + A_2 \langle u^2 \rangle \langle v \rangle + A_3 \langle u \rangle \langle v^2 \rangle + A_4 \frac{1}{2} (\langle u^3 \rangle \langle v \rangle + \langle u \rangle \langle v^3 \rangle + \langle u \rangle \langle v \rangle \langle \xi^2 \rangle) \\ \left[ \begin{array}{l} A_1 \frac{1}{2} (\langle u^3 \rangle + \langle u \rangle \langle v^2 \rangle + \langle u \rangle \langle \xi^2 \rangle) \\ + A_2 \frac{1}{2} (\langle u^4 \rangle + \langle u^2 \rangle \langle v^2 \rangle + \langle u^2 \rangle \langle \xi^2 \rangle) \\ + A_3 \frac{1}{2} (\langle u^3 \rangle \langle v \rangle + \langle u \rangle \langle v^3 \rangle + \langle u \rangle \langle v \rangle \langle \xi^2 \rangle) \\ + A_4 \frac{1}{2} (\frac{1}{2} (\langle u^5 \rangle + \langle u \rangle \langle v^4 \rangle + \langle u \rangle \langle \xi^4 \rangle) + \langle u^3 \rangle \langle v^2 \rangle + \langle u^3 \rangle \langle \xi^2 \rangle + \langle u \rangle \langle v^2 \rangle \langle \xi^2 \rangle) \end{array} \right] \end{pmatrix}$$

## Update of conservative variables and time Integration

The Gas-Kinetic Scheme is a Finite Volume method, which uses the Gas-Kinetic to compute the interface fluxes. The Computation of the Fluxes was shown in the previous sections. The values of the conservative variables in the cells are then updated with these Fluxes:

$$\underline{W}_{i,j}^{n+1} = \underline{W}_{i,j}^n + \frac{1}{\Delta V} \sum_l \int_{t^n}^{t^{n+1}} \vec{F}_l \cdot \vec{\Delta} S_l dt \quad \text{with} \quad \underline{W} = [\rho, \rho U, \rho V, \rho E]^T \quad (27)$$

For the implementation the time integration can be solved explicitly. Moreover the Flux density tensor  $\vec{F}_l$  is formulated in a local coordinate system and has only the form of a Vector of the Fluxes in Interface normal direction. The current implementation supports only structured rectangular grids. Therefore only vertical and horizontal interfaces occur. The projection of the Flux density tensor on the interface normal then simplifies to a swap of the momentum fluxes for the horizontal interfaces and a projection of the interface normal (positive coordinate direction) onto the outward facing normal of the cell face. the latter one results only in a positive (right and top) or negative (left and bottom) sign  $s$ .

$$\underline{W}_{i,j}^{n+1} = \underline{W}_{i,j}^n + \frac{1}{\Delta V} \sum_l s \underline{R} [\underline{F}^1 t - \tau t \underline{F}^2 + (\frac{1}{2} t^2 - \tau t) \underline{F}^3]_0^{\Delta t} \quad (28)$$

$$\underline{W}_{i,j}^{n+1} = \underline{W}_{i,j}^n + \frac{1}{\Delta V} \sum_l s \underline{R} (\underline{F}^1 \Delta t - \tau \Delta t \underline{F}^2 + (\frac{1}{2} \Delta t^2 - \tau \Delta t) \underline{F}^3) \quad (29)$$

The swap of the momentum components is shown by the Rotation Matrix  $\underline{R}$  which just rotates the second and third component.

## Boundary Conditions

The Boundary Conditions are implemented with the help of a layer of Ghost-Cells around the domain. The Interfaces between the domain and the Ghost-Cells exactly represent the boundary. Two types of Boundary Conditions are implemented for every primitive variable. The value of the Ghost-Cell for Diriclet Boundary Conditions (fixed value on boundary) is computed with linear interpolation:

$$Z_{\alpha, \text{GhostCell}} = 2Z_{\alpha, \text{Boundary}} - Z_{\alpha, \text{Cell in Domain}} \quad (30)$$

The homogeneous Neumann Boundary Conditions (zero Gradient at Boundary) are implemented in a similar way:

$$Z_{\alpha, \text{GhostCell}} = Z_{\alpha, \text{Cell in Domain}} \quad (31)$$

The values of the conservative variables are then computed from the primitive variables.

## 2 Some Formulas

The ideal gas law:

$$p = \rho RT = \frac{\rho}{2\lambda} \quad (32)$$

From the Rayleigh-Bernard-Paper (Xu, Lui, 1999):

$$\lambda = \frac{1}{2RT} = \left[ \frac{s^2}{m^2} \right] \quad (33)$$

$$\tau = \frac{\nu}{RT} = 2\lambda\nu \quad (34)$$

From the GKS-Book (Xu, 2015):

$$\lambda = \frac{(K+2)\rho}{4 \left( \rho E - \frac{1}{2} \frac{(\rho U)^2 + (\rho V)^2}{\rho} \right)} \quad (35)$$

$$\rho E = \frac{(K+2)\rho}{4\lambda} + \frac{1}{2} \frac{(\rho U)^2 + (\rho V)^2}{\rho} \quad (36)$$

The first moments of the Maxwell distribution:

$$\begin{aligned} \langle 1 \rangle &= 1 \\ \langle u \rangle &= U \\ \langle u^2 \rangle &= U^2 + \frac{1}{2\lambda} = \frac{1}{\rho} (\rho U^2 + p) \\ \langle u^3 \rangle &= U^3 + U \frac{1}{2\lambda} + \frac{1}{\lambda} U = U^3 + \frac{3}{2\lambda} U \\ \langle u^4 \rangle &= U^4 + \frac{3}{2\lambda} U^2 + \frac{3}{2\lambda} \left( U^2 + \frac{1}{2\lambda} \right) = U^4 + \frac{3}{\lambda} U^2 + \frac{3}{4\lambda^2} \end{aligned} \quad (37)$$

Euler equation in flux formulation from Xu (2001):

$$\frac{\partial}{\partial t} \begin{pmatrix} \rho \\ \rho U \\ \rho V \\ \rho E \end{pmatrix} + \frac{\partial}{\partial x} \begin{pmatrix} \rho U \\ \rho U^2 + p \\ \rho UV \\ (\rho E + p)U \end{pmatrix} + \frac{\partial}{\partial y} \begin{pmatrix} \rho V \\ \rho UV \\ \rho V^2 + p \\ (\rho E + p)V \end{pmatrix} = 0 \quad (38)$$



## 2.1 CFL-Condition

From Guo and Liu (2008):

$$\Delta t = CFL \frac{\Delta x}{u_{max} + c_s} \quad \text{for} \quad Re \gg 1 \quad (39)$$

$$\Delta t \leq \frac{\Delta x^2}{2^d \nu} \quad \text{for} \quad Re \approx 1 \quad (40)$$

From Tannehill et al. (1997) Chapter 9 Eq. 9.14

$$\Delta t \leq CFL \frac{\Delta x}{(u_{max} + c_s)(1 + 2/Re^*)} \quad (41)$$

The Problem with the latter condition is, that  $Re^* = \frac{u_{max} \Delta x}{\nu}$  becomes zero for stagnating flows, which then results in zero time steps. Therefore currently a modified version of the latter condition is implemented:

$$\Delta t \leq CFL \frac{\Delta x}{u_{max} + c_s + \frac{2\nu}{\Delta x}} \quad (42)$$

## 2.2 Forcing

the forcing is implemented as:

$$\begin{pmatrix} \rho \\ \rho U \\ \rho V \\ \rho E \end{pmatrix}^{n+1} = \begin{pmatrix} \rho \\ \rho U \\ \rho V \\ \rho E \end{pmatrix}^n + Fluxes + \begin{pmatrix} 0 \\ \Delta t \rho F_x \\ \Delta t \rho F_y \\ 0 \end{pmatrix} \quad (43)$$

### 3 Poiseuille Flow

$$\begin{aligned}
 u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} &= -\frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + \frac{G}{\rho} \\
 0 &= -\frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \frac{\partial^2 u}{\partial y^2} + \frac{G}{\rho} && \text{with } v = 0, \frac{\partial u}{\partial x} = 0 \\
 \frac{\partial^2 u}{\partial y^2} &= -\frac{1}{\nu} \left( -\frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{G}{\rho} \right) \\
 u(y) &= \frac{1}{2\nu} (Hy - y^2) \left( -\frac{1}{\rho} \frac{\partial p}{\partial x} + \frac{G}{\rho} \right) \\
 u(y) &= \frac{1}{2\nu} (Hy - y^2) \frac{G}{\rho} && \text{with } \frac{\partial p}{\partial x} = 0 \\
 u_{max} &= \frac{GH^2}{8\nu\rho}
 \end{aligned} \tag{44}$$

A very simple test case, with which I test my implementation is the Poiseuille flow driven by a volume force  $G$ , where the analytical solution is known.

The set up of the simulation is the following:

- computation domain of  $1m \times 1m$
- one Cell in flow direction (x-direction)
- varying number of cells in cross flow direction (y-direction, 64 Cells for the figures)
- periodic boundary condition in x-direction (by assigning the Pointers to the cell itself)
- Ghost Cell boundary conditions on top and bottom
  - zero Gradient for density and temperature (in the form of  $\lambda$ )
  - fixed value for both velocities  $u = v = 0.0 \frac{m}{s}$ .
- $K = 1$ ,  $\nu = 0.01 \frac{m^2}{s}$ ,  $\rho = 1.0 \frac{kg}{m^3}$
- $G = 1 \cdot 10^{-4} \frac{m}{s^2}$
- $\lambda = 1.0 \frac{s^2}{m^2}$  (random value) and  $\lambda = 8.2 \cdot 10^{-6} \frac{s^2}{m^2}$  (corresponds to  $T = 293.15K$  for Argon)
- $Re = 0.125$
- $Ma = 1.77 \cdot 10^{-3}$  and  $Ma = 5.06 \cdot 10^{-6}$  for the respective simulations
- $CFL = 0.1$  (the value of  $CFL = 0.5$  seemed to yield unstable results)

The result of the simulation is shown in the figures 1 and 2. The stream wise velocity corresponds with analytical solution and reduces with first order with a refinement in y-direction for both values of  $\lambda$ . The cross stream velocity should be zero. The simulation with  $\lambda = 1.0$  yields non zero cross stream velocities that are not negligible small, only 5 orders of magnitude smaller than the stream wise velocity at double precision, so round off and cut off can be ruled out. These v-velocities are also stationary and do not disappear with time. Moreover they do not disappear with a grid refinement.

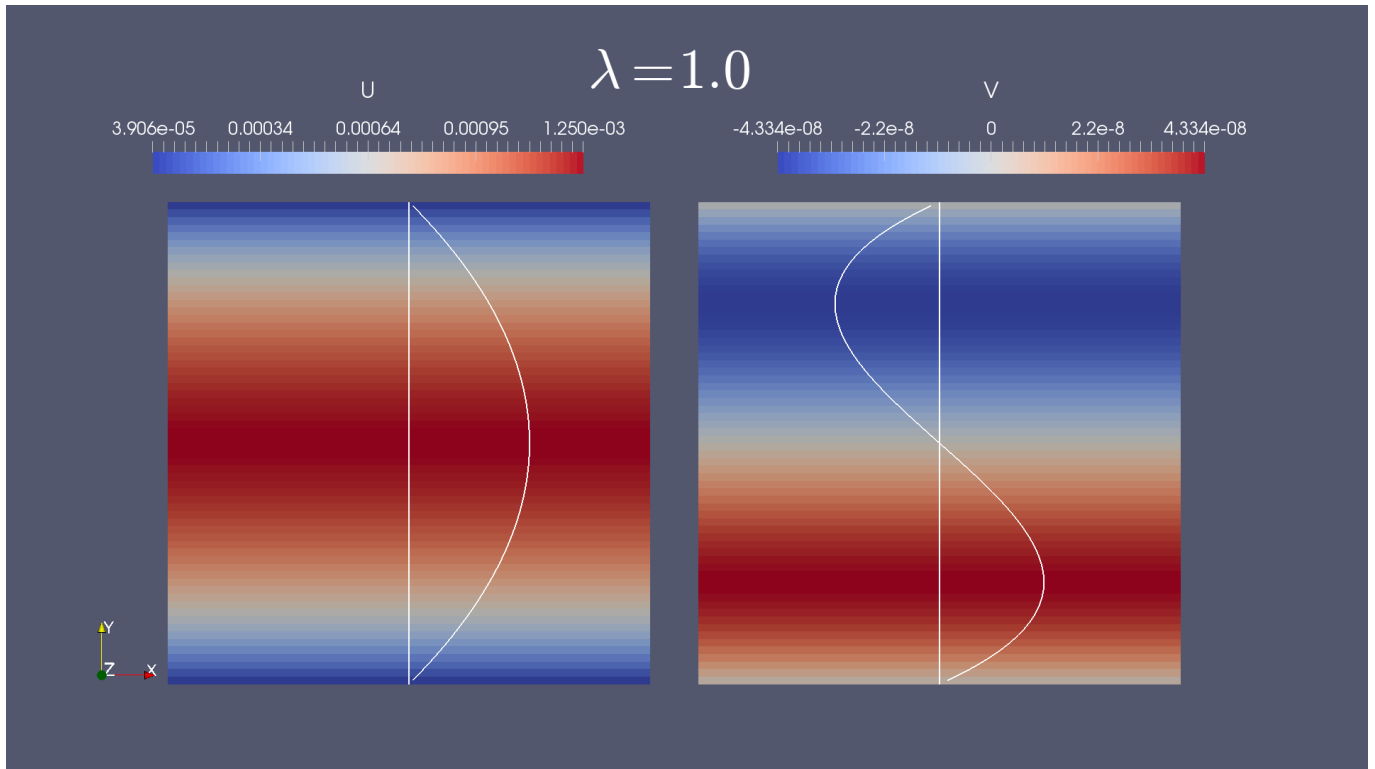


Figure 1: Velocity Field of the Poiseuille flow with the velocity profiles with example values

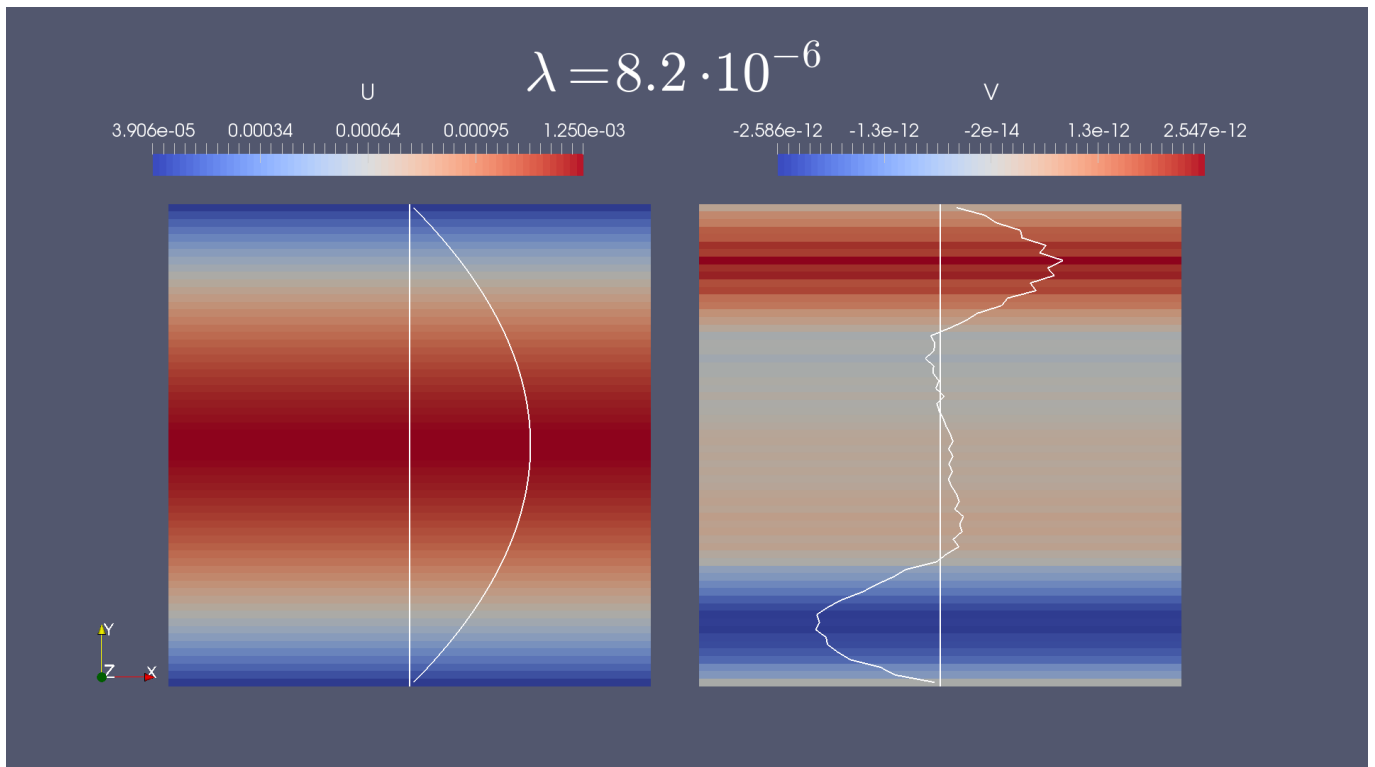


Figure 2: Velocity Field of the Poiseuille flow with the velocity profiles with physically more realistic values

## 4 Implementation

My Implementation is based on your structured Code and the corresponding Paper by Xu and Lui (1999). From there I used the incompressible GKS from the 4th Chapter of Xu's GKS-Book (2015) with the simplification of smooth flow, which is also reported in Su, Xu et al. (1999).

I implemented everything in C++, making use of the object orientation. There is one class containing the mesh and all the functionality on the whole mesh. The mesh itself contains pointers to all Cells and Interfaces, which in turn are represented by Objects of the respective classes. These two classes contain the data of the flow field and the functionality limited to a cell or an interface. In addition to that a Cell has the pointers to the four adjacent Interfaces and an interface has the pointers to the two adjacent cells. These pointers are used to get all the needed data for the flux computation (interfaces) and the update of the macroscopic variables (cells).

My Algorithm is shown in figure 3. The time iteration is controlled in the global mesh class. First the global time step is computed as the minimum of all local time steps, which are computed by the CFL-Condition. Then I loop over all Interfaces and compute the fluxes. This is the right part of figure 3 and the main task in the algorithm.

I start with computing the values of the primitive variables at the interface by third order interpolation. The Interfaces at the boundary have only one neighbor outside of the domain (a Ghost Cell) and therefore I use linear interpolation for those interfaces. Then I compute the derivatives of conservative variables in the same manner. I also normalize all derivatives with the density. With the primitive Variables I then can compute the relaxation time. At this point I have to distinguish the Interfaces with respect to their orientation. I wrote only one code for all Interfaces (normal to x-axis and y-axis). For the Interfaces normal to the y-axis I need to swap the velocities and momenta (and the respective derivatives) since a normal velocity corresponds to a y-velocity in the global frame of reference, but to a x-velocity (normal) in the local interface frame of reference. Then I can compute the local slope constants  $a$  and  $b$  and the moments of the equilibrium distribution. With these two I then reconstruct the time slope of the conservative variables. The time slope is automatically normalized with density, since neither  $a$  and  $b$  nor the moments contain the density. The local slope constants with respect to time can then be computed by the same means as  $a$  and  $b$ .

Finally the flux over the Interface is assembled by splitting it into three parts for the time integration. The symbolic time integration and the integration over the area of the interface (by multiplying with the interface area) is also taken into account in this step. Then the Fluxes are rotated again into the global frame of reference.

With the fluxes computed I update the cell averaged values but adding all fluxes of the four adjacent interfaces to the current value. Then I apply the boundary conditions, by assigning the corresponding values to the ghost cells. Then I write my output data and start the iteration all over again.

### 4.1 Weidong Answer

Ans: I use the standard CFL condition. Since  $CFL * \frac{dx}{\lambda_{max}} \leq 1$ , where  $\lambda_{max}$  is the spectral radius and  $\lambda_{max} = (|u| + C) + v_{diffusion}$ ,  $C$  is the sound speed.  $v_{diffusion}$  is the characteristic velocity of diffusion and it is about  $2\nu/dx$ .

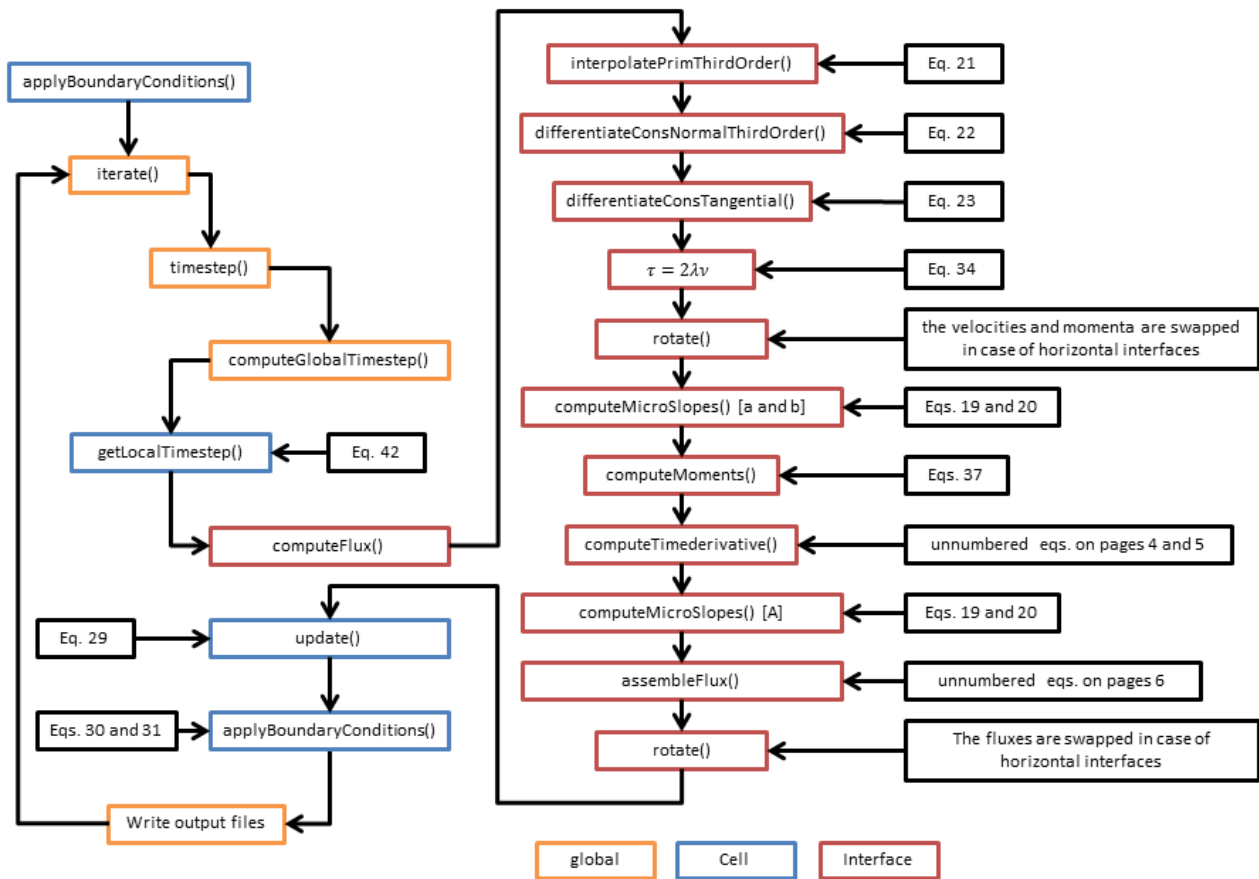


Figure 3: GKS-Algorithm for smooth flows