Fluid Flow Simulation with Lattice Boltzmann Methods on High Performance Computers

Mathias J. Krause, Adrian Kummerländer, Stephan Simonis

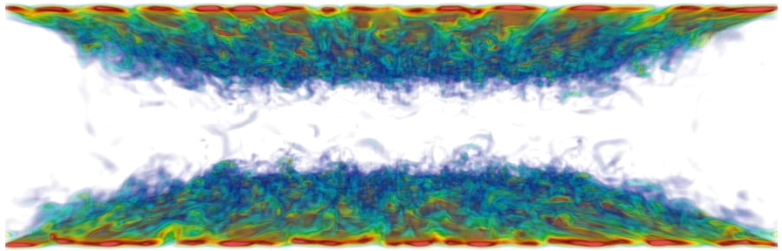
NASA AMS Seminar Series, December 1, 2020

Lattice Boltzmann Research Group (LBRG)
Institute for Applied and Numerical Mathematics (IANM)
Institute of Mechanical Process Engineering an Mechanics (MVM)
Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany

Challenges in (Computational) Fluid Mechanics

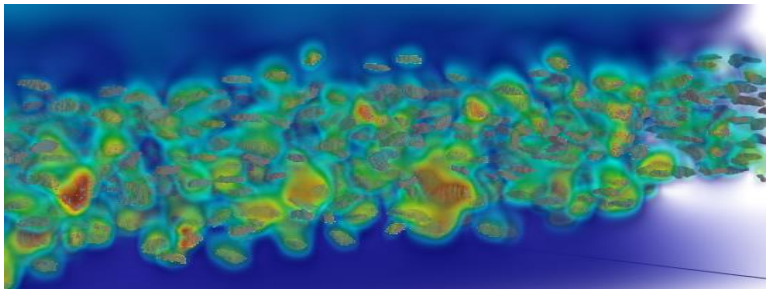
Challenge 1: Turbulence

- capture small scales
- models inaccurate or expensive



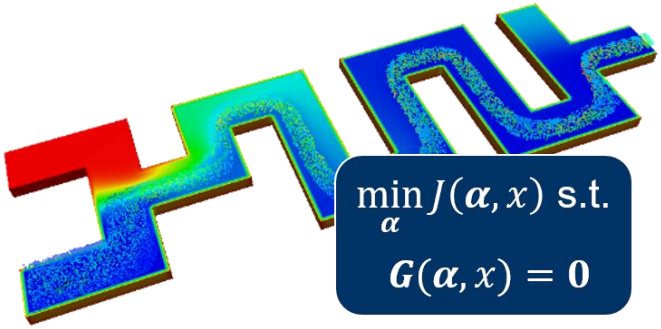
Challenge 2: Suspensions

- capture effects of small particles
- models inaccurate or expensive



Challenge 3: Optimal Control / Optimization

- enable model calibration & optimization
- formulation problem dependent, expensive

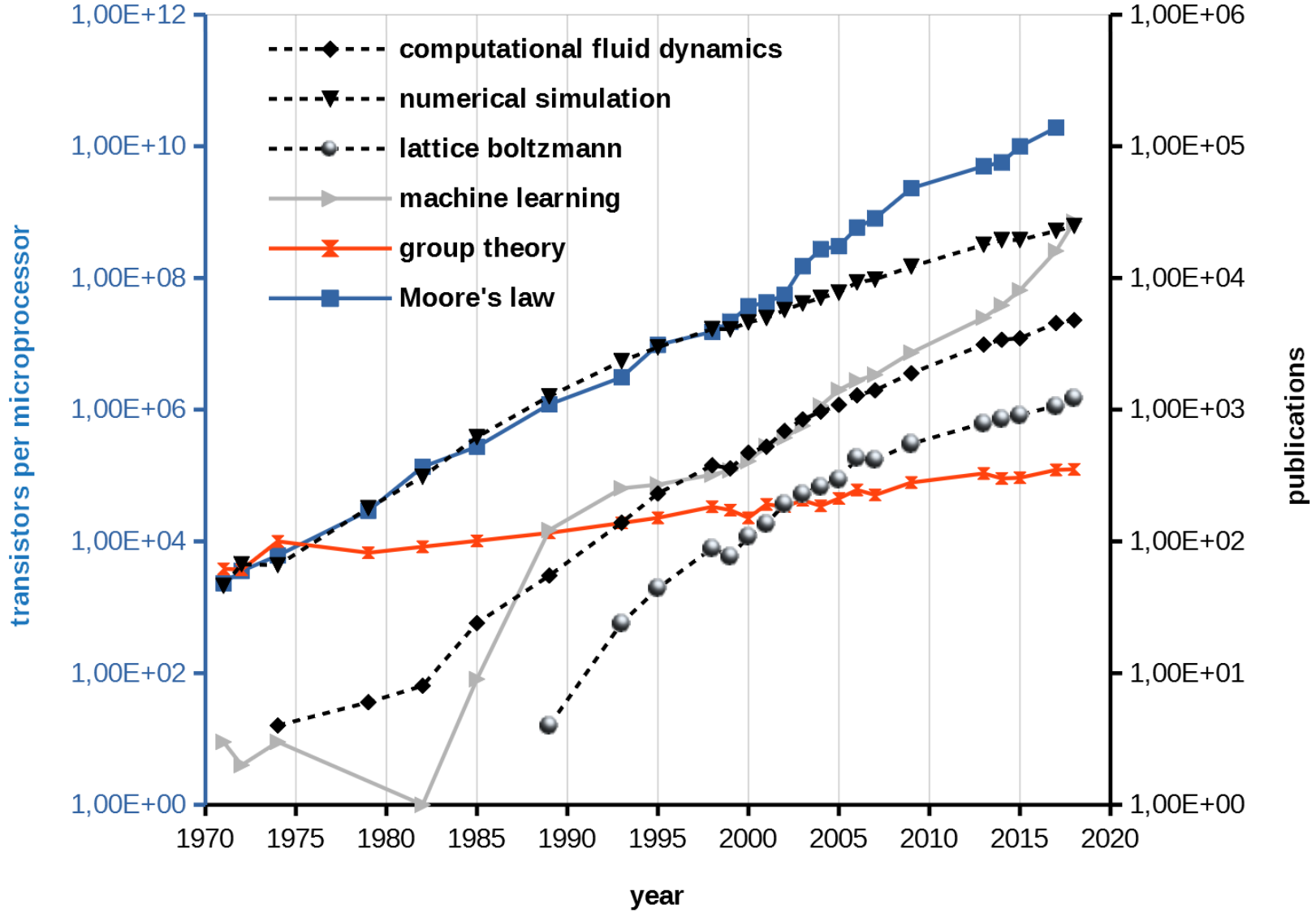


$$\min_{\boldsymbol{\alpha}} J(\boldsymbol{\alpha}, x) \text{ s.t. } \boldsymbol{G}(\boldsymbol{\alpha}, x) = \mathbf{0}$$

Kwak, D., Kiris, C., Kim, C. S. (2005) Comput Fluids, 34(3), pp.283-299

Slotnick, J., Khodadoust, A., Alonso, J. *et al.* (2014). NASA *TR*, no. NASA/CR-2014-218178

Facing the Challenges: Compute Power Available



Facing the Challenges: LBRG's Solution Approach

Parallel Lattice Boltzmann Methods (LBM)

- physical mesoscopic model
- algorithmic properties / parallelism
- LB approach as PDE solver

Sustainable Research & Education

- beyond one PhD cycle
- open (source) community
- method AND application view
- interdisciplinary
- modern C++, CI, GIT, ..

Challenge 1:

DNS/LES
instead of RANS

Challenge 2:

resolve particles' shape, force, …

Challenge 3:

algorithmic differentiation & adjoints, combine measurement & simulation

Agenda

I. Facing Challenges in CFD with LBM and OpenLB	**Mathias J. Krause**
II. LBM on Heterogeneous HPC Platforms	Adrian Kummerländer
III. LBM for Turbulent Flows	Stephan Simonis

Agenda

I. Facing Challenges in CFD with LBM and OpenLB

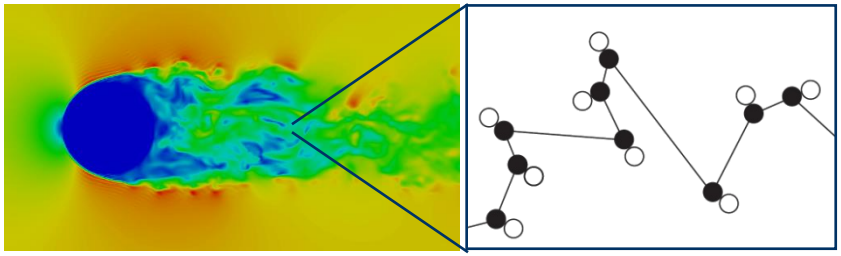
1. Motivation
2. **Introduction to Lattice Boltzmann Methods (LBM)**
3. Introduction to OpenLB
4. Overview of Applications

II. LBM on Heterogeneous HPC Platforms

III. LBM for Turbulent Flows

Hierarchical Modelling - Overview

Characteristic length L



Mean free path of molecules l_f

Macroscopic view ~200.000 : 1 microscopic view

- **Microscopic:**

  Newton's laws – motion and interaction of single molecules

- **Mesoscopic: $6*10^{23}$ molecules per mole!**

  Statistical physics – distribution of molecules ➔ **LBM**

- **Macroscopic:**

  Continuum – neglect of molecular interaction

Modelling Flows of Incompressible Newtonian Fluids

macroscopic:

Navier-Stokes equation
$\boldsymbol{u}, p$ in $I \times \Omega$

$h \to 0$

[1]

mesoscopic:

Boltzmann equation /
BGK-Boltzmann equation
f in $I \times \Omega \times \mathbb{R}^d$

$h \in \mathbb{R}_{>0}$

$$\omega \sim h^2, \; l_f \sim h, \; c_s \sim 1/h$$

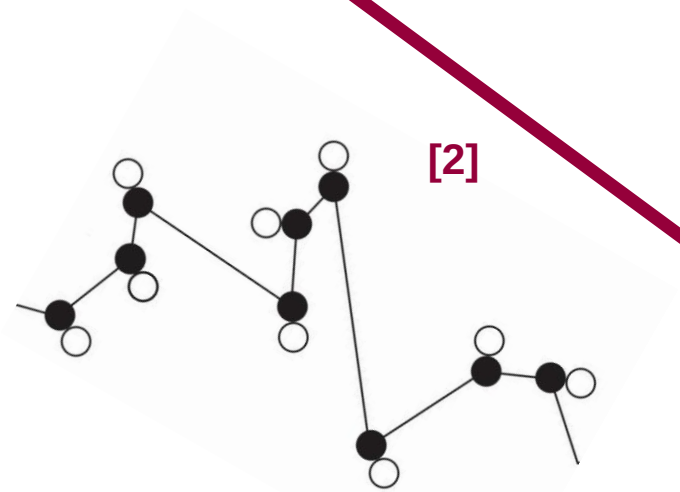
[3] **Consistent discretization: LBM**

$$\Delta t = h^2, \; \Delta r_k = h, \; v_i \sim 1/h$$

Lattice Boltzmann equation
f_i in $I_h \times \Omega_h \times Q$

- explicit scheme
- efficient parallelization

[2]



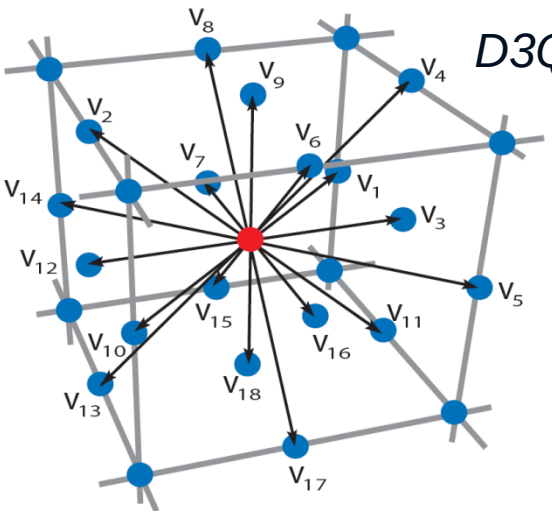
[1] Saint-Raymond, L. (2003). Ann. Sci. Ecole Norm. Sup.(4), 36, 271-317.

[2] Junk, M., Klar, A., Luo, L. S. (2005). J. Comput. Phys., 210(2), 676-704.

[3] Krause, M. J. (2010)., KIT Karlsruhe.

Lattice Boltzmann Methods (LBM)

Idea: coupling model parameter $h \in \mathbb{R}_{>0}$ with discretization parameter: *Lattice DdQq*



Macroscopic moments:

density $\rho = \sum_{i=0}^{q-1} f_i$, velocity $\rho \boldsymbol{u} = \sum_{i=0}^{q-1} \boldsymbol{v_i} f_i$.

Time loop $t = t_0, t_0 + h^2, t_0 + 2h^2, ..., t_1$

Position space loop $\boldsymbol{r} \in \Omega_h$

(1) Collision $\tilde{f}_i(t, \boldsymbol{r}) = f_i(t, \boldsymbol{r}) - \frac{1}{3\nu + 1/2} \left(f_i(t, \boldsymbol{r}) - M_{f_i}^{eq}(t, \boldsymbol{r}) \right)$

(2) Streaming $f_i(t + h^2, \boldsymbol{r} + h^2 \boldsymbol{v_i}) = \tilde{f}_i(t, \boldsymbol{r})$

Optimal Control Solution Strategies

Problem: find control α and state f such that $J(f,\alpha)$ is minimised and $G(f,\alpha)=0$ holds

Side condition G governed by e.g. a BGK-Boltzmann equation

Two solution strategies [3]

- Iterative, gradient-based (Steepest Descent, LBFGS)
- Discretization: LBM
- Parallel

Approach I

First-discretize-then-optimize

- (Discrete) adjoint-based [1, 2]
- Algorithmic Differentiation [3]

Approach II [3]

First-optimize-then-discretize

- Adjoint-based

[1] Pingen, G., Evgrafov, A., Maute, K. (2007). Struct Multidisc Optim 34(6), 507-524.

[2] Tekitek, M. M., Bouzidi, M., Dubois, F. *et al.* (2006). Comput Fluids, 35(8-9), 805-813.

[3] Krause, M. J. (2010)., KIT Karlsruhe.

Agenda

I. Facing Challenges in CFD with LBM and OpenLB

1. Motivation
2. Introduction to Lattice Boltzmann Methods (LBM)
3. **Introduction to OpenLB**
4. Overview of Applications

II. LBM on Heterogeneous HPC Platforms

III. LBM for Turbulent Flows

Facts and Figures

2D and 3D fluid flow and transport simulations based on LBM

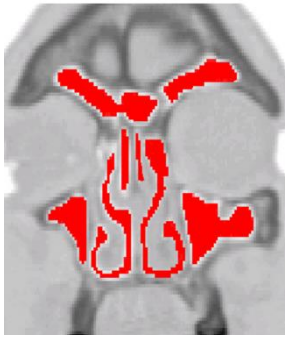
Realization

- Started in 2006 by Jonas Latt & Mathias J. Krause
- Open source (GPL2)
- C++, object oriented, template-based, modular, extensible
- **Hybrid parallelization (MPI & OpenMP)**

Features in latest release 1.4

- Various lattice types: D2Q9, D3Q15, D3Q19, ...
- Local, non-local, on- and off-lattice boundary conditions
- Collision models: BGK, MRT, LES, multiphase, thermal
- **Build-in pre-processing from e.g. STL-files**
- Unit conversion for problem set-up in SI-units
- XML interface for input parameters
- Visualization (built-in and VTK)

Built-in Geometry Creation and Meshing



CT/MRT

CAD

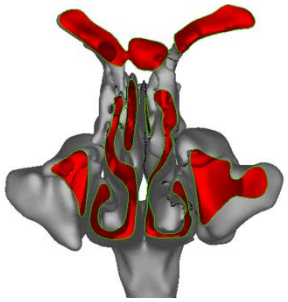
Step 1b):
Import

Indicator Functors

Step 1c):
Built-in

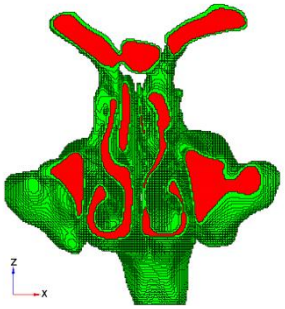
Step 1a):
Segmentation,
Smoothing

VMTK, ITK, …



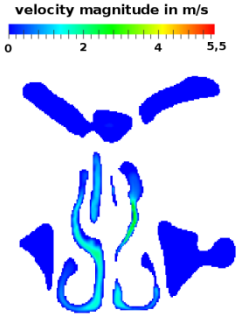
STL Surface Mesh **and/or**
Geometry Primitives

Step 2:
Built-in voxelizing



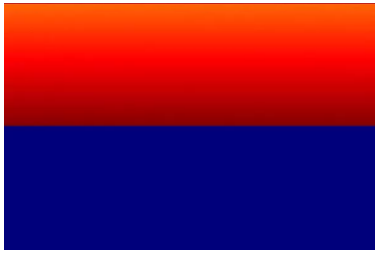
Voxel
Mesh

Step 3:
Setting of
material-
dependent
boundary
conditions



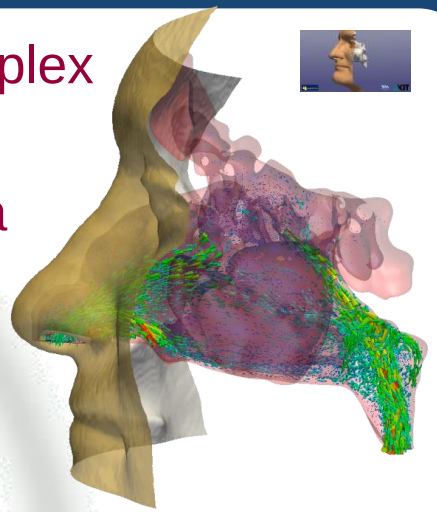
Simulation
Results

Applications

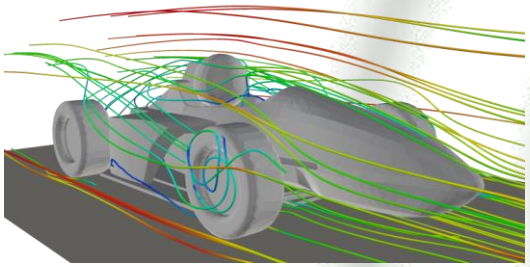


Multiphase flows

Flows in complex geometries, porous media

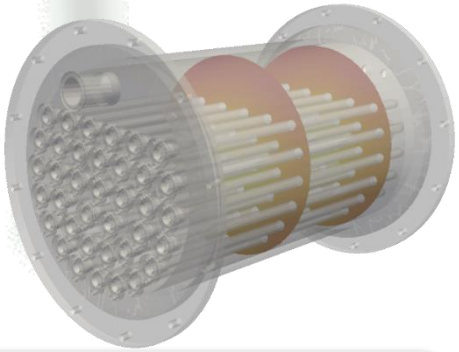


Turbulent flows

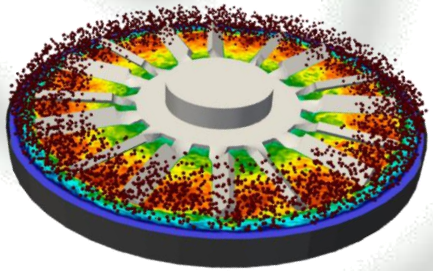


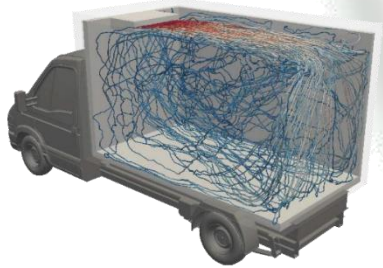
OpenLB Applications

Radiative transport



Particle flows





Thermal flows

Krause, M. J., Kummerländer, A., Avis, S. J., Kusumaatmaja, H., Dapelo, D., Klemens, F., Gaedtke, M., Hafen, N., Mink, A., Trunk, R., Marquardt, J. E., Maier, M.-L., Haussmann, M., Simonis, S. (2020). Comput Math Appl, in Press.

Agenda

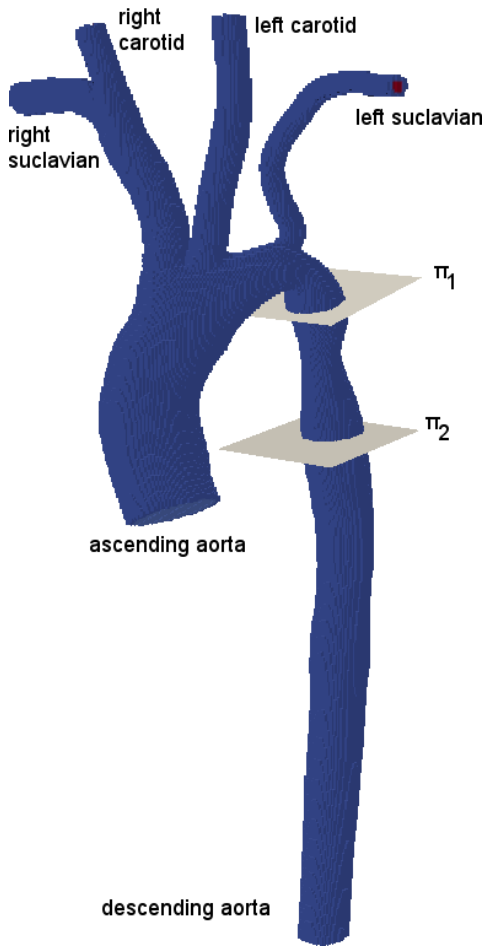
I. Facing Challenges in CFD with LBM and OpenLB

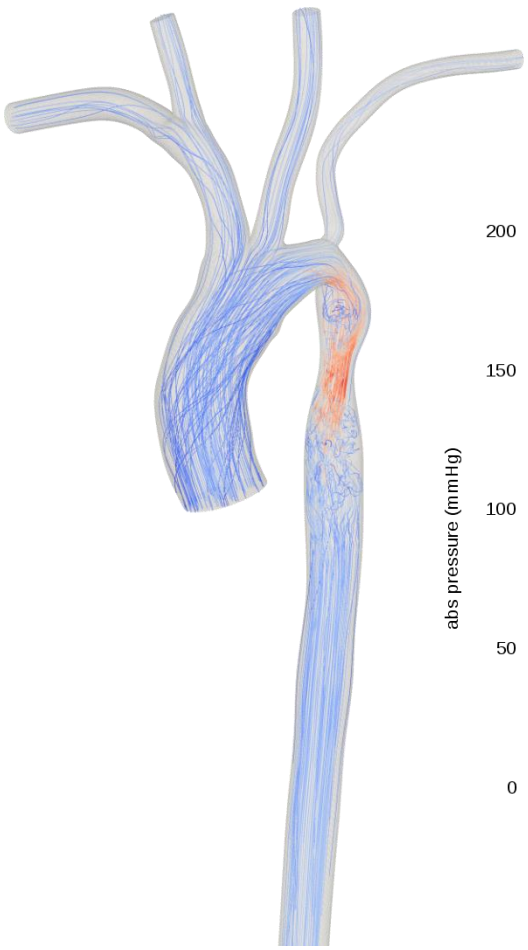
1. Motivation
2. Introduction to Lattice Boltzmann Methods (LBM)
3. Introduction to OpenLB
4. **Overview of Applications**

II. LBM on Heterogeneous HPC Platforms

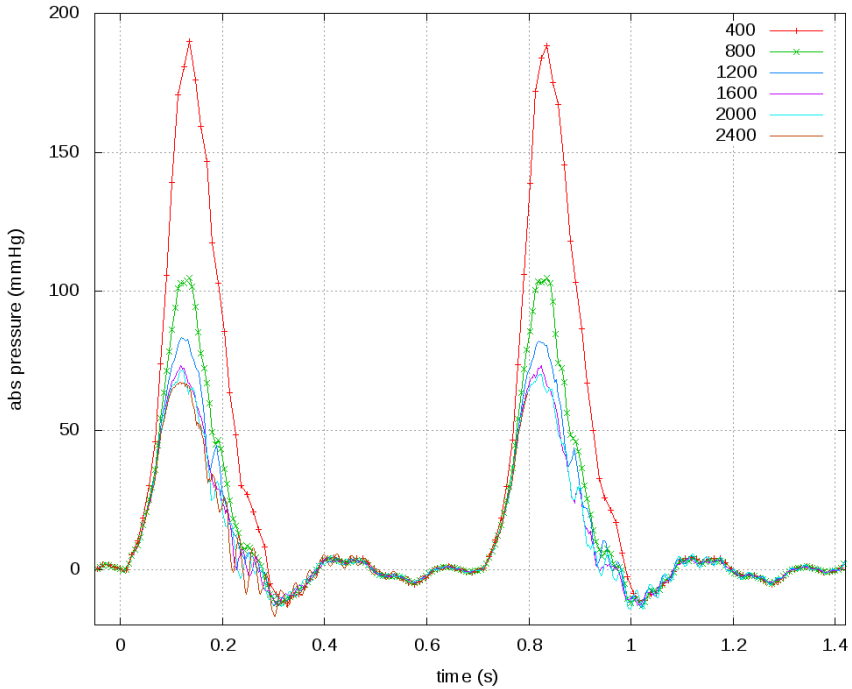
III. LBM for Turbulent Flows

Aorta Benchmark, DNS





Goal: Validate DNS LBM

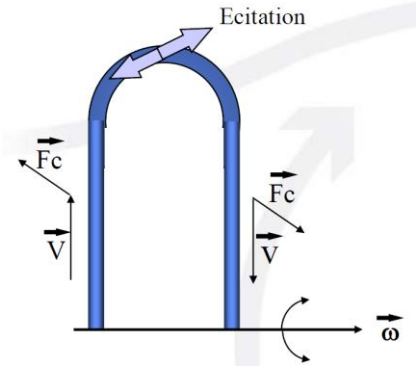


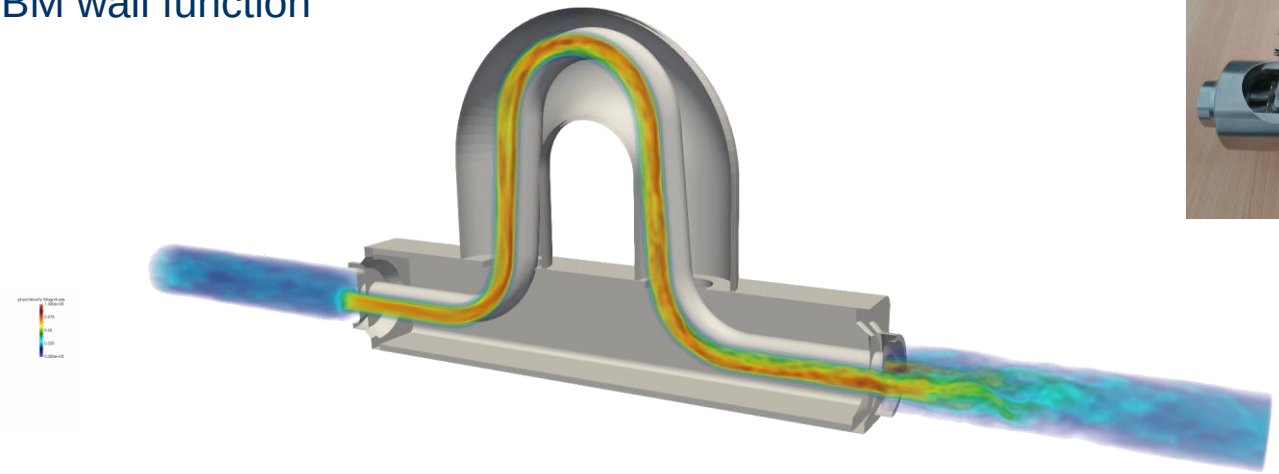
Henn, T., Krause, M. J. *et al.* (2012). In proceedings MICCAI12 STACOM12 7746, 34.

Coriolis Mass Flowmeter Simulation, LES

Goal: Improve measurement accuracy

- Investigation of pressure drop
  - Comparison with experimental data
- Investigation of vortex phenomena
  - LBM Large Eddy Simulation Smagorinsky model
  - LBM wall function





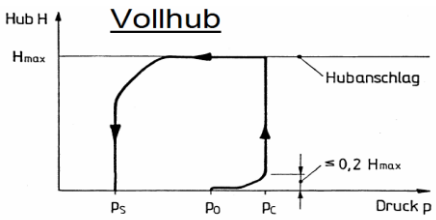
Haussmann, M., Reinshaus, P., Simonis, S. *et al.* (2020). Preprint arXiv:2005.04070 [physics.comp-ph].

Haussmann, M., Barreto, A. C., Kouyi, G. L. *et al.* (2019). Comput. Math. with Appl., 78(10), 3285.

Safety Valve Simulation, LES

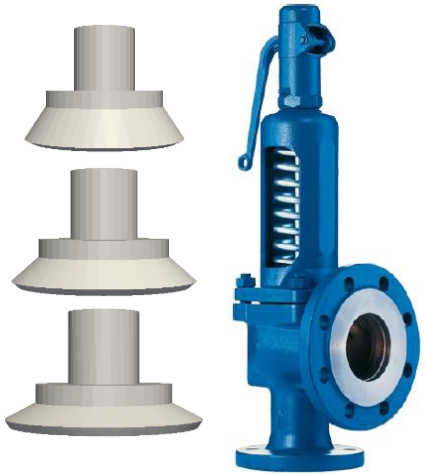
Goal: avoid chatter

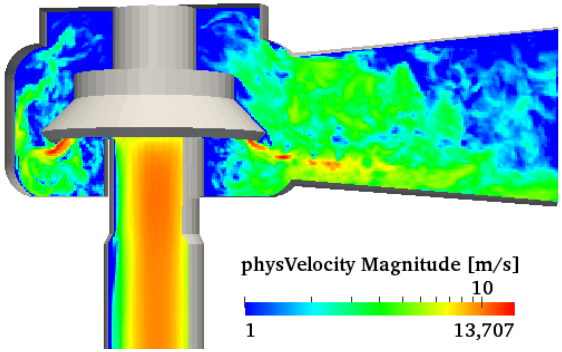
➔ vary shape of disk

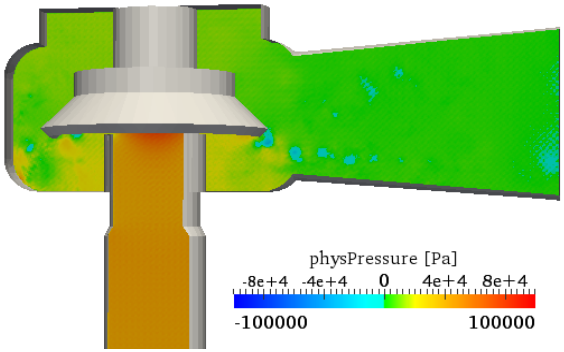


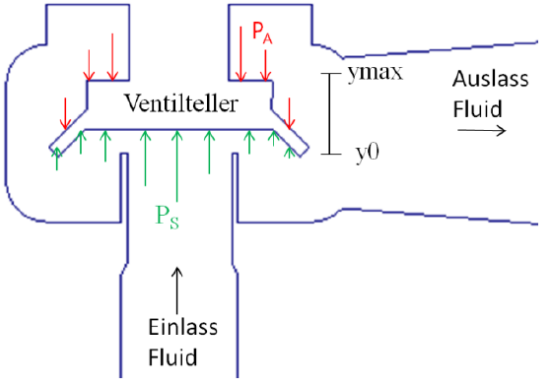
- 3D transient turbulent simulation
- 1 billion degrees of freedom
- parallelization: 30 days ➔ 1 day
  64 cores ➔ 2.048 cores
- optimize shape of disk

different disk angle







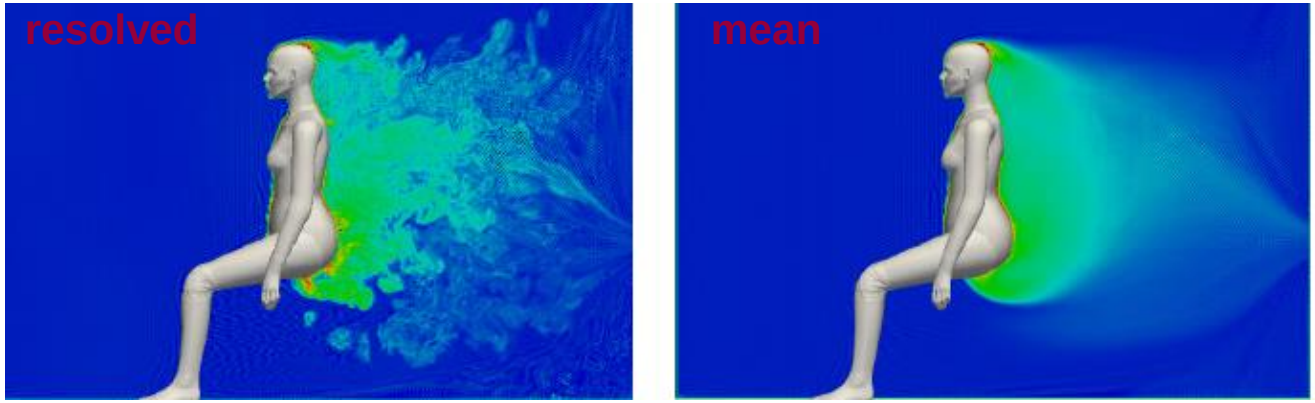


Thermal Flow for Thermal Comfort, LES

Goal: Improve thermal comfort

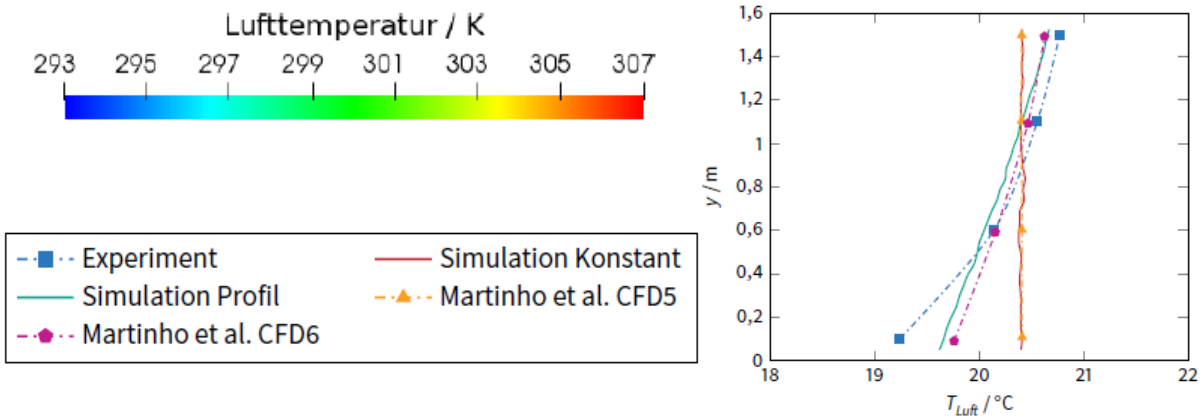
control flow patterns by change of design and flow conditions of

- Heating
- Air condition
- Ventilator



Benchmark study:

- Re=29,000
- Pe=20,600
- LES - Smagorinsky type
- 130 mio. grid cells



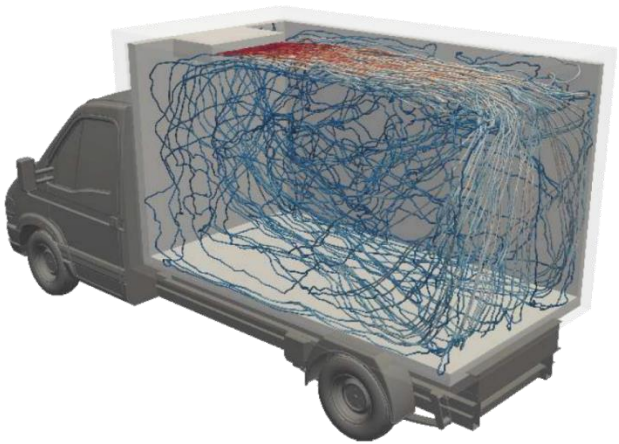
Siodlaczek, M., Gaedtke, M., Simonis S. *et al.* (2020). Submitted to Build Environ.

Thermal Flow in Refrigerated Vehicles, LES

Goal: Improve the insulation efficiency

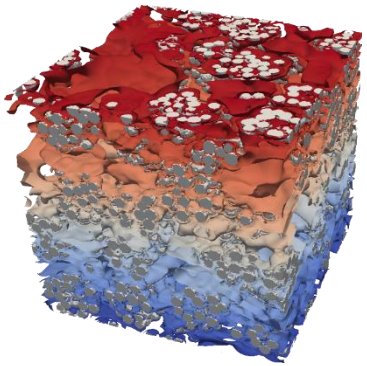
➔ exchange insulation material

- extruded polysterol (XPS) by
- vacuum insulation panels (VIP)



Convection in vehicle's cooling chamber:

- Air conditioning volume flow of $990\,\frac{m^3}{h}$
- Turbulent free jet, $\mathrm{Re} = 28{,}000$
- Large eddy simulation (LES) Smagorinsky
- Resolved heat flux through insulation walls
- Utilizing conjugated heat transfer implementation



Gaedtke, M., Wachter, S., Kunkel, S. *et al.* (2019). Heat and Mass Transfer, 1-13.

Gaedtke, M., Wachter, S., Raedle, M. *et al.* (2018). Comput. Math. with Appl., 76(10), 2315-2329.

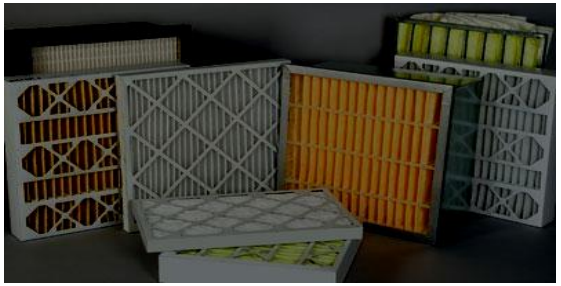
Ross-Jones, J., Gaedtke, M., Sonnick, S. *et al.* (2019). Comput. Math. with Appl., 77(1), 209-221.

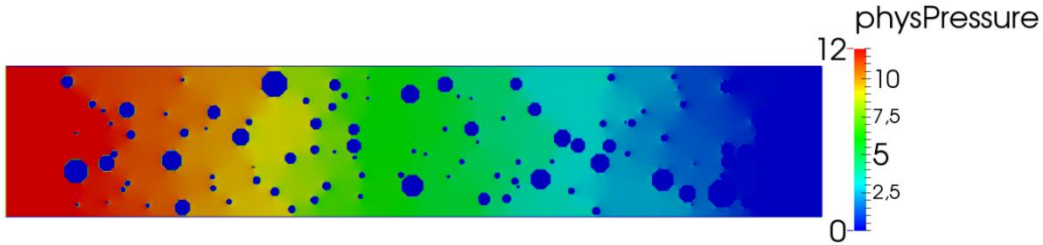
Micro Filtration, Particle

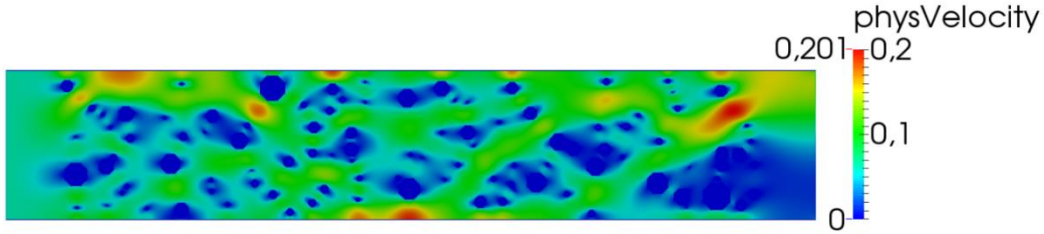
Goal: design of an efficient filter

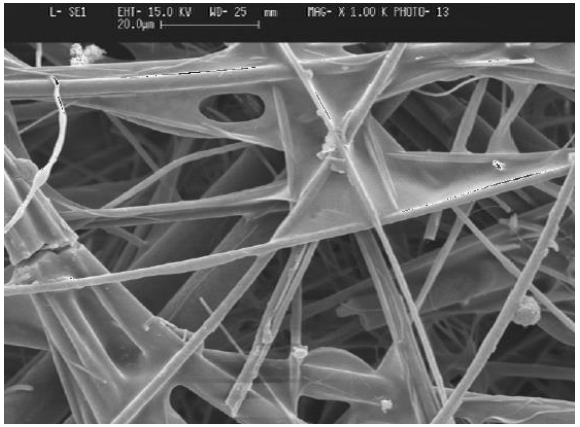
➔ vary shape of filter and flow conditions

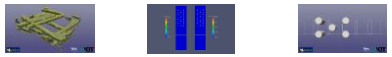
- geometry from µCT scans
- 2D and 3D transient simulation slip flow
  - particles (Lagrange)
  - air as density (Euler)







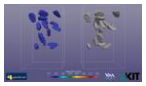




Augusto, L. D. L. X., Ross-Jones, J., Lopes G. C. *et al.* (2018). Commun Comput Phys, 23, 910-931.

Fine Particle Fractionation, Particle

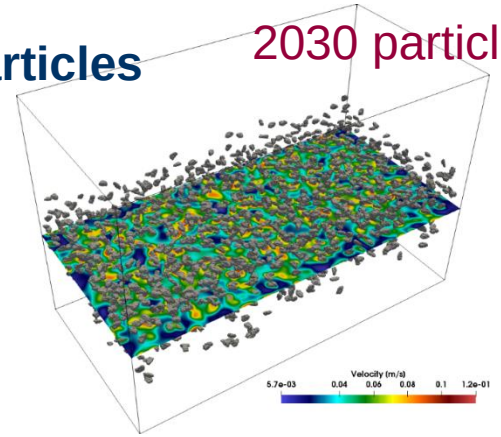
Challenge: Low selectivity in the range from $100\ nm$ to $10\ \mu m$



Goal: Improvement of separation processes

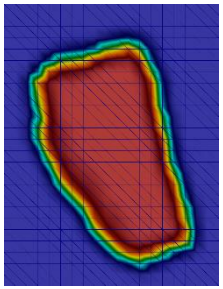
➔ Simulation of a large number of **arbitrary shaped particles**

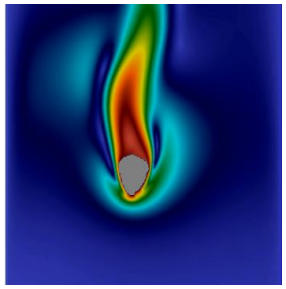
2030 particles

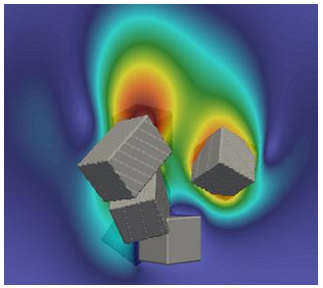


Method needs to account for surface structure

Contact treatment for non-spherical particles is complex





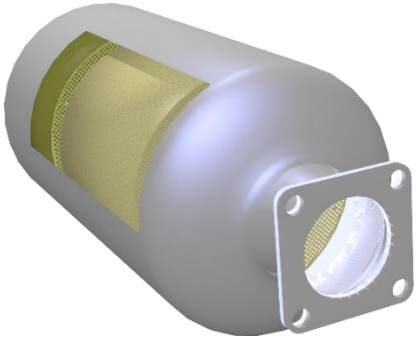


Trunk, R., Marquardt, J., Thäter, G. *et al.* (2018). Comput Fluids, 172, 621-631.

Krause, M. J., Klemens, F., Henn, T. *et al.* (2017). Particuology, 34, 1-13.

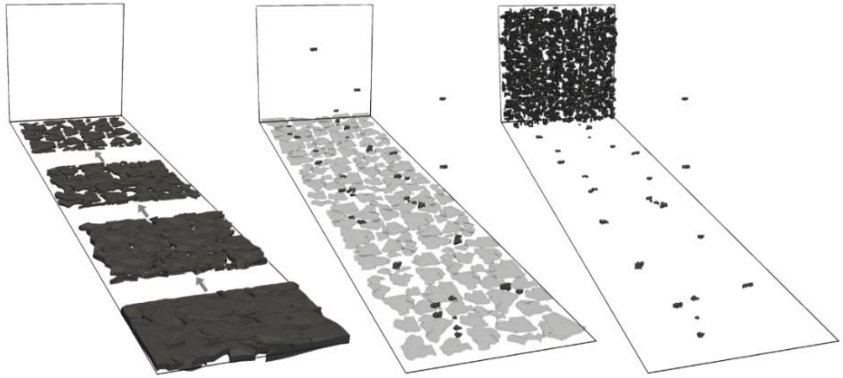
Exhaust Treatment by Wall-flow Filters, Particle

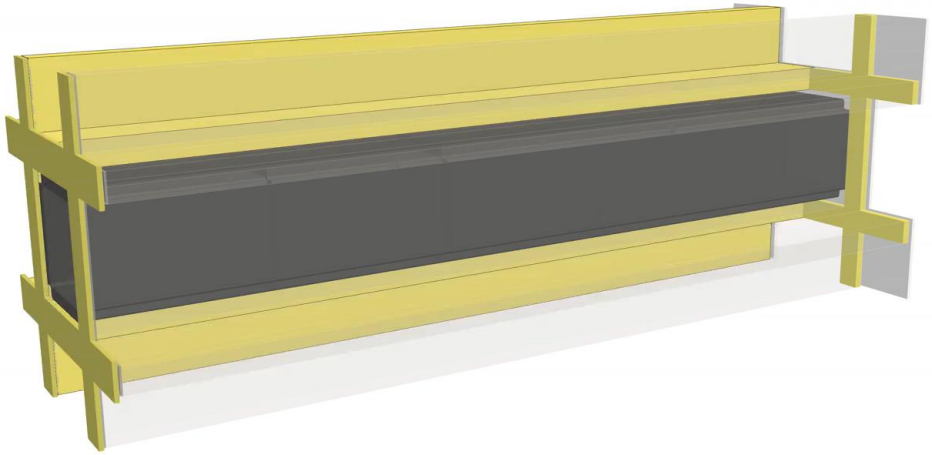
Goal: Investigation of particle-layer rearrangement



➔ simulation of resolved particulate flows

- Ash accumulates, forms specific deposition patterns
- Patterns evolve due to oxidation during the filter regeneration
- Effect of deposition patterns:
  - change in filter efficiency
  - increase of pressure loss



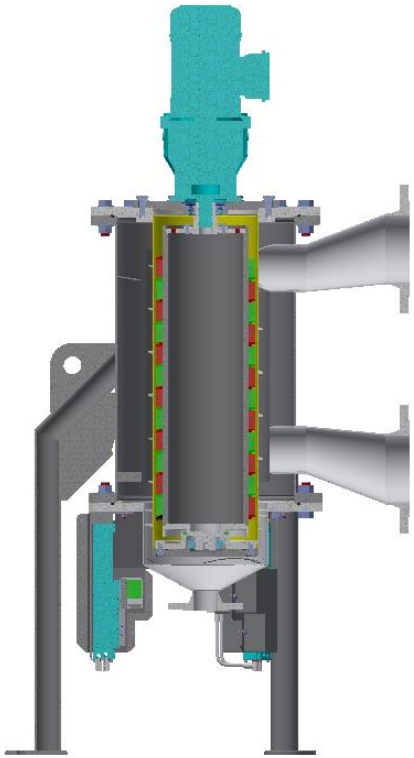


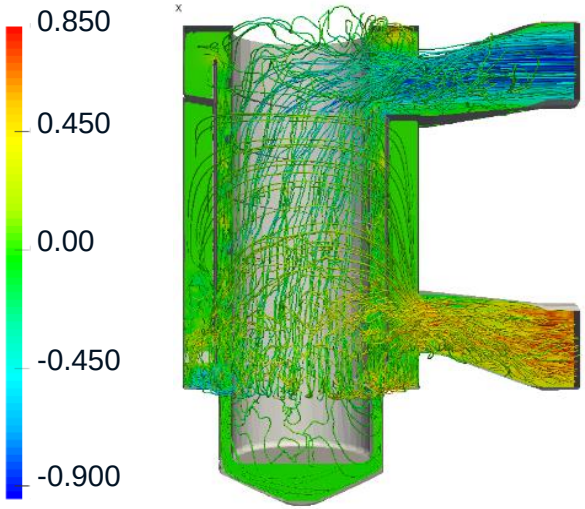
Hafen, N., Dittler, A., Krause, M. J. (2020). Submitted to Philos. Trans. Royal Soc. A.

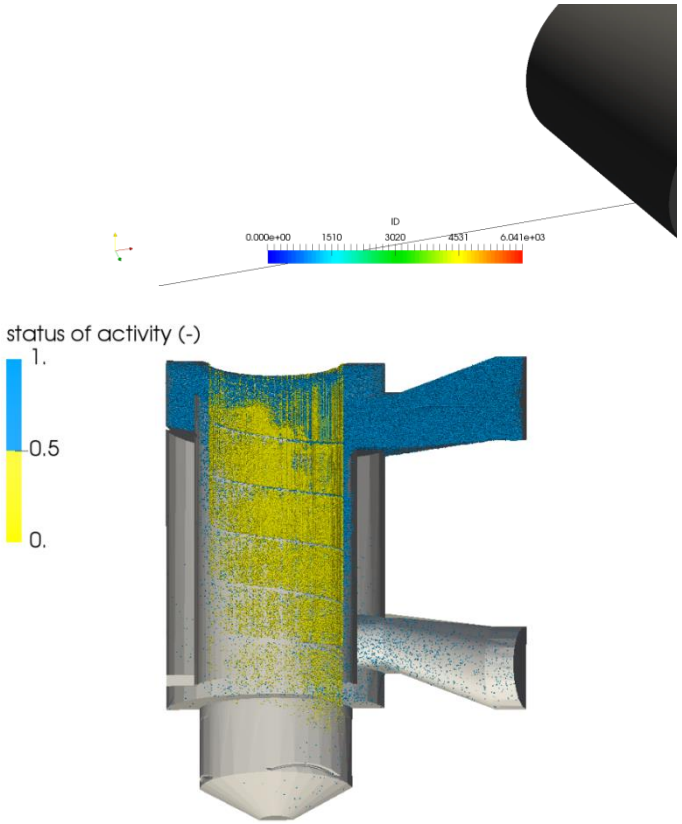
Magnetic Spiral Separator, Particle

Goal: basic understanding, increase efficiency

3D simulation with LBM
- carrier fluid (Euler)
- magnetic field (Euler)
- magnetic particles (Lagrange)







Maier, M. L., Milles, S., Schuhmann, S. *et al.* (2018). Comput. Math. with Appl., 76(11-12), 2744-2757.

Maier, M. L., Henn, T., Thaeter, G. *et al.* (2017). Chem Eng Technol, 40(9), 1591-1598.

Photobioreactor Simulation, Complex System

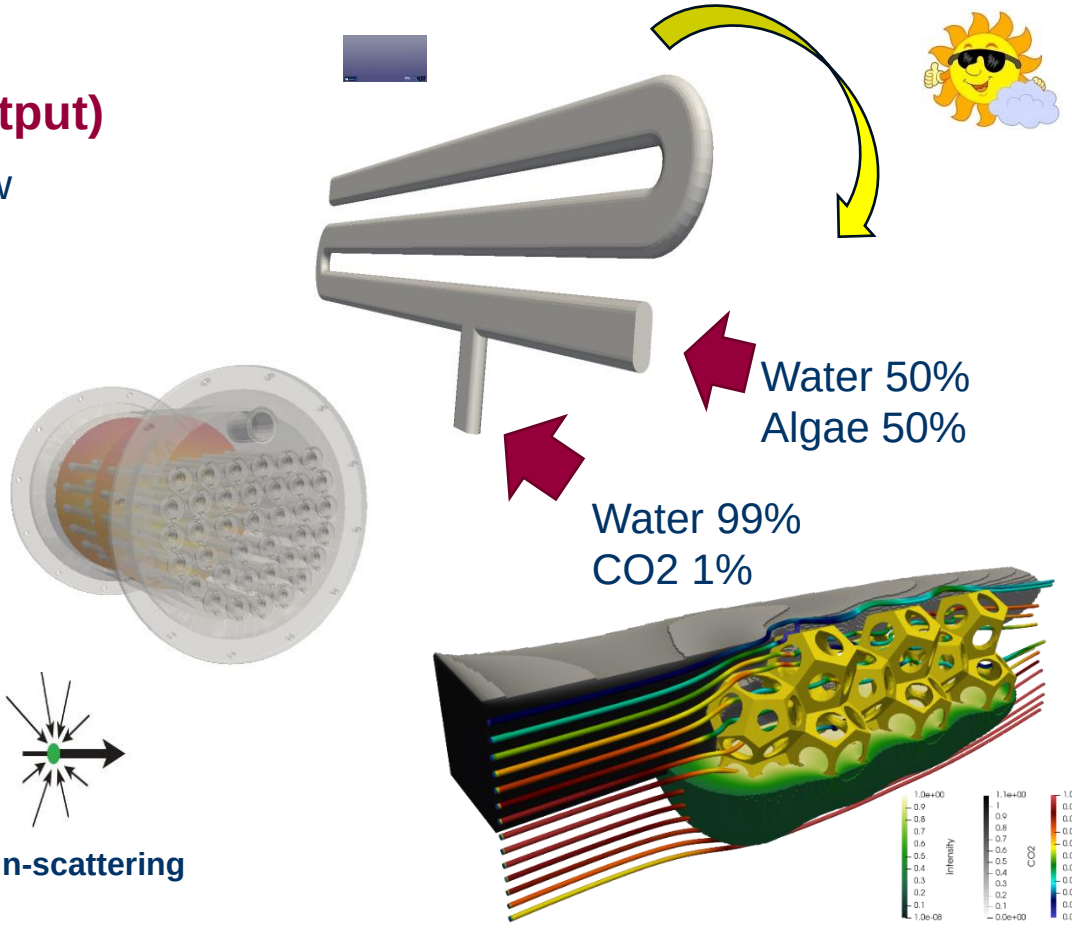
Goal: optimize energy balance

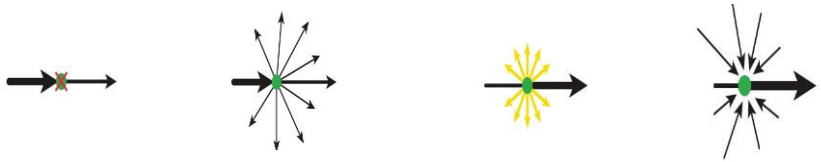
(energy input / bio mass output)

➔ change shape, position, inflow

- 3D suspension simulation
  - algae as particles (Lagrange)
  - CO2 , Water as density (Euler)
  - algae consume CO2
- 3D light simulation

next: vary pipe shape, sun position





absorption **out-scattering** **emission** **in-scattering**

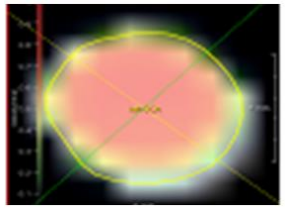
Mink, A., McHardy, C., Bressel, L. *et al.* (2020). J Quant Spectrosc Radiat Transf, 243, 106810.

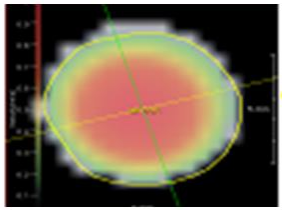
Mink, A., Thäter, G., Nirschl, H. *et al.* (2016). J Comput Sci, 17, 431-437.

CFD-MRI: Basic Algorithm, Optimization

Step 1: Noisy MRI scan velocity u^*, boundary ➔ start "porosity" d_0

Step 2: CFD simulation $u(d_i)$





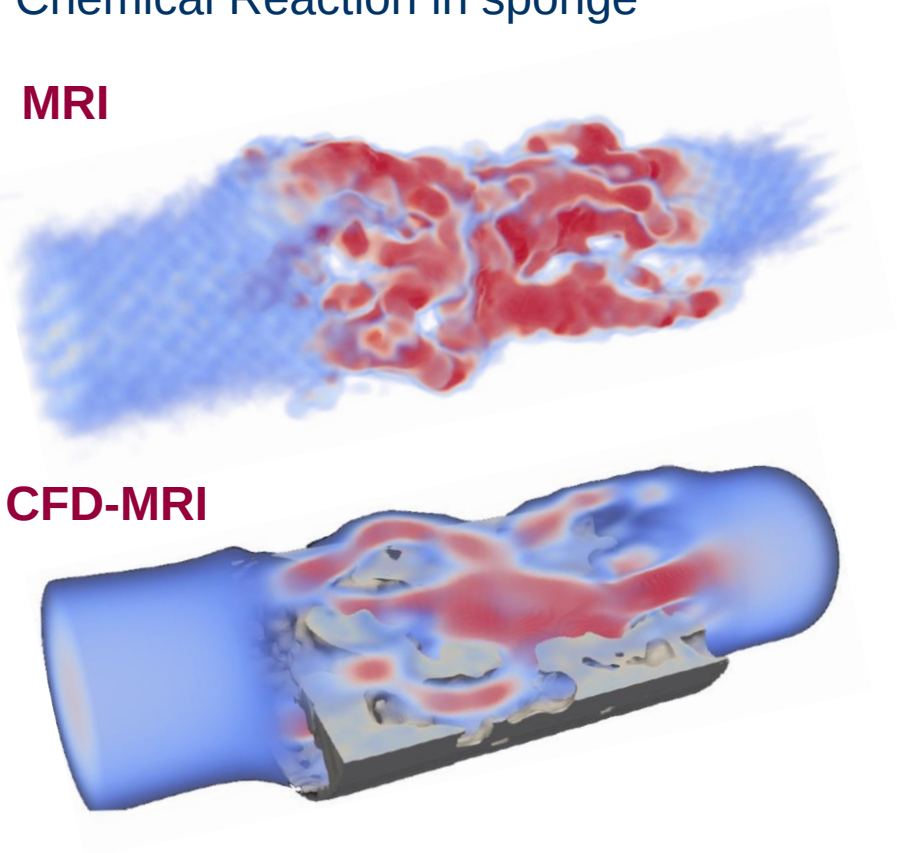
Goal:
Reduce noise,
identify flow topology

Step 3: Calculate
a) difference between CFD and MRI $J = \int_\Omega (u(d_i) - u^*)^2$,
b) gradient dJ and with this,
c) new parameter for porosity d_{i+1}

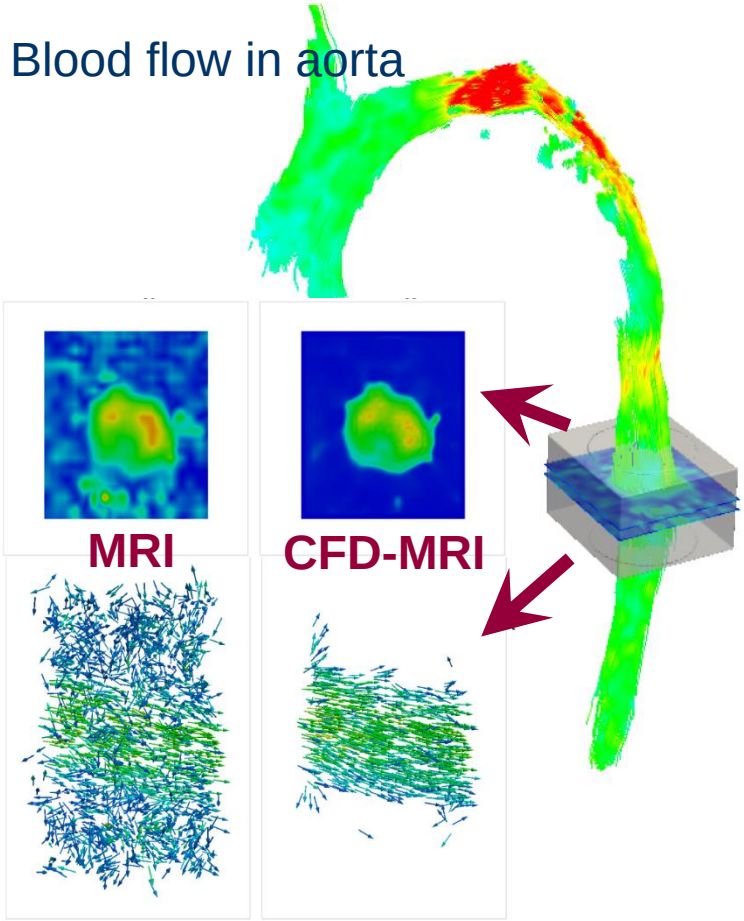
Krause, M. J., Förster, B., Mink, A. *et al.* (2016). In *HPC in Sci and Eng 16* (pp. 337-353).

CFD-MRI: Applications Sponge & Aorta, Optimization

Chemical Reaction in sponge



Blood flow in aorta



Klemens, F., Schuhmann, S., Guthausen, G. *et al.* (2018). Comput Fluids, 166, 218-224.

Klemens, F., Schuhmann, S., Balbierer, R. *et al.* (2020). Comput Fluids, 197, 104391.

Agenda

I. Facing Challenges in CFD with LBM and OpenLB	Mathias J. Krause
II. LBM on Heterogeneous HPC Platforms	**Adrian Kummerländer**
III. LBM for Turbulent Flows	Stephan Simonis

Agenda

I. Facing Challenges in CFD with LBM and OpenLB

II. LBM on Heterogeneous HPC Platforms

- **1. OpenLB Concept**
- 2. Parallelization Architecture and Data Structures
- 3. Performance Aspects: CSE and Propagation Pattern
- 4. Parallel Performance of OpenLB

III. LBM for Turbulent Flows

Basics of LBM Implementation

In theory:

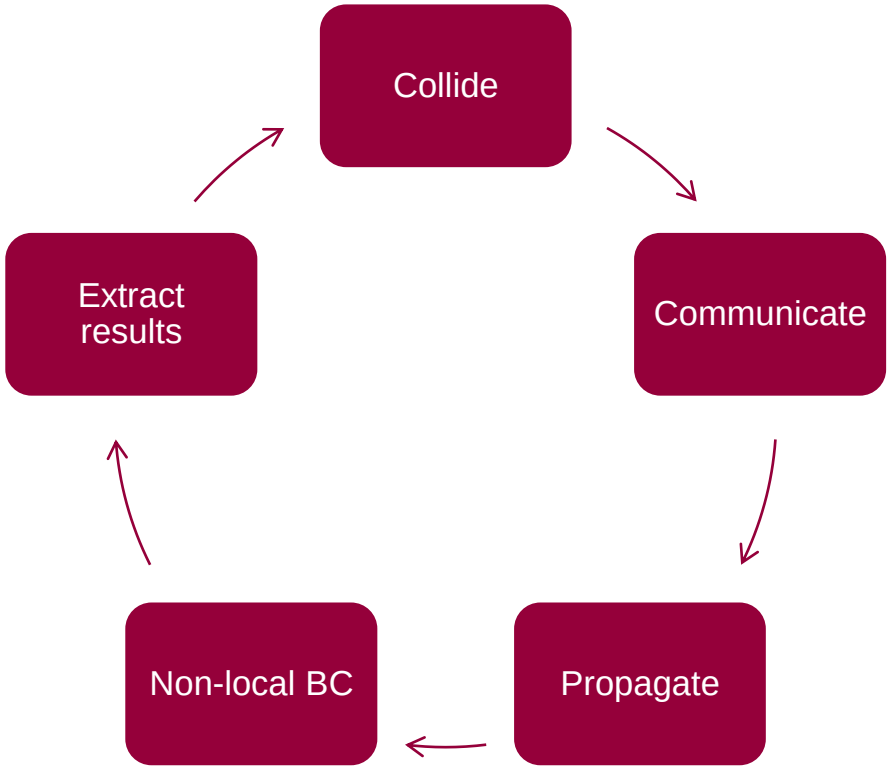
- Local collision step
- Neighborhood-local propagation step

In practice:

- Not all collisions are strictly local
- Not all boundary conditions are local

In OpenLB:

- Strictly local collision
- Generic propagation
- Post processors for non-local BCs, coupling
- Functors for results, error norms, ...



OpenLB Overview

- Continuously developed since 2006
- CPU-only C++14 code
- Hybrid parallelization reflected in class hierarchy
  - Super lattice contains block lattice contains n-dim array of cells
  - Cells have associated `Dynamics`, `Momenta`, `PostProcessor`
- Compromise between flexibility. performance and usability

Main challenges:

Adding GPU support, grid refinement, improving performance…
While preserving rich set of boundary conditions and collision models.

Functor Concept

Pre- and post-processing, geometry generation, boundary setup, …

share the need to work with **functions on spatial locations**

Use case	Function
Geometry generation	$f : \mathbb{R}^d \to \{0,1\}$
Boundary and initial conditions	$f : \mathbb{Z}^d \to \mathbb{R}^m$ (density, velocity, …)
Post-processing	$f : \mathbb{Z}^d \to \mathbb{R}^m$ (flow rate, error norms, …)

Function $f : \Omega \to \mathbb{R}^d, x \mapsto f(x)$ is realized as `GenericF<T,S>::operator()`

Generic interface for VTK, plotting, … output

Support for **functor arithmetic** and composition $f + g, \|\tilde{f} - f\|/\|f\|$ ….

▶ Flexible and convenient user interface

Agenda

I. Facing Challenges in CFD with LBM and OpenLB

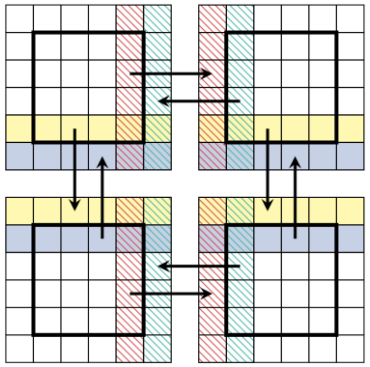
II. LBM on Heterogeneous HPC Platforms

1. OpenLB Concept
2. **Parallelization Architecture and Data Structures**
3. Performance Aspects: CSE and Propagation Pattern
4. Parallel Performance of OpenLB

III. LBM for Turbulent Flows

Parallelization Architecture

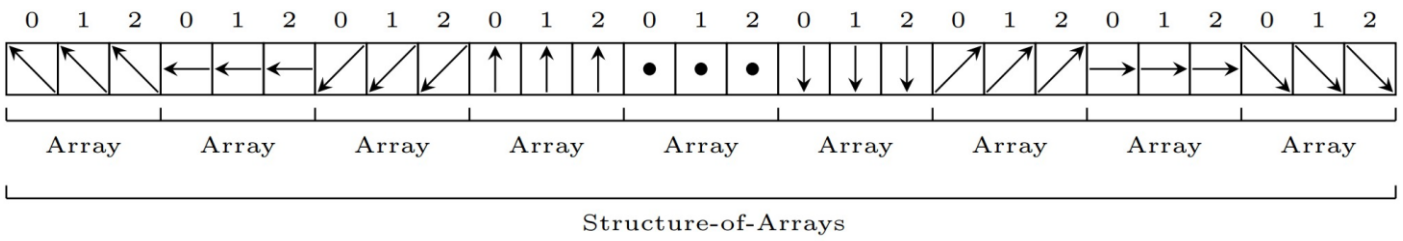
- Octree for domain decomposition, shrink blocks to geometry
- Overlap for communication
- Load balancer assigns blocks to processes
- Block-local processing is independent
  - OpenMP, SIMD, GPUs, Accelerators...
- Processes communicate using OpenMPI

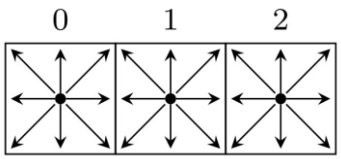


Krause, M. J. *et al.* (2009). Comput. Math. with Appl., 58(5), 1071-1080.

New Core Data Structures

Groundwork for GPU support, **SIMD**, increase flexibility





Adapt a data-first approach, break up strict object modeling

- Store all population data and associated fields in SoA column vectors
- Turn old data-classes into interfaces for legacy support

Cleaning up data-access patterns

- DdQq descriptor is template type that accepts field declarations
- `D3Q19<tag::MRT,FORCE>` declares layout of entire data structure

Old, New and Future Code Structures

Old: Existing class structure, derived from hybrid parallelization approach

```
using T = double; // or ADf…
CuboidGeometry<T> cGeometry(stlIndicator);
HeuristicLoadBalancer load(cGeometry);
SuperGeometry<T> sGeometry(cGeometry, load);
using DESCRIPTOR = D3Q19<tag::MRT,FORCE>;
SuperLattice<T,DESCRIPTOR>
sLattice(sGeometry);
```

Cell-focused, good UI but inconvenient for performance

```
Cell<T,DESCRIPTOR> cell = sLattice.get(loc);
cell.setField<FORCE>({1, 0, 0});
cell.collide(); // uses Dynamics, Momenta, ...
```

New: Raw SoA foundation, Cell only as interface for existing code

```
// Plain SoA, accessible to SIMD, CUDA, ...
FieldArrayD<T,DESCRIPTOR,POPULATION>
: ColumnVector<POPULATION::type,
               DESCRIPTOR::size<POPULATION>()>
```

Future: Strictly differentiate between data-focused Operators and Functors

```
// Operator implementable based on cells,
// SIMD packs, GPU kernel...
sLattice.apply<SmagorinskyBgkCollisionO>(
  bulkMask, /* params */);
```

Agenda

I. Facing Challenges in CFD with LBM and OpenLB

II. LBM on Heterogeneous HPC Platforms

1. OpenLB Concept
2. Parallelization Architecture and Data Structures
3. **Performance Aspects: CSE and Propagation Pattern**
4. Parallel Performance of OpenLB

III. LBM for Turbulent Flows

Performance Metric

Well optimized LBM codes are bound by memory bandwidth

For single time step and cell: Q loads and Q stores

Transfer 2*Q*sizeof(FPT)+idx bytes of memory per cell and time step

Millon **L**attice **UP**dates per second

Example	OpenLB 1.4	Experimental[1]	Limit[2]
LDC, D3Q19, double, 256^3	152 MLUPs	443 MLUPs	447 MLUPs

▶ Optimize code to be bound by memory bandwidth, then minimize per-cell transfer

[1] Using symbolically generated, hand-vectorized AVX256 benchmark code
[2] w.r.t. STREAM bandwidth on a dual socket Intel Xeon Silver 4114 node (Skylake) ~ 130 GiB/s

Performance Aspect: CSE

Common **S**ubexpression **E**limination

- CSE is a powerful tool for reaching the memory wall
- Some advanced models can be computed *for free*

e.g. Performance of Smagorinsky BGK (~800 FLOP) ≈ plain BGK (~500 FLOP)

- OpenLB has manually CSE-optimized collision operators
- Experimental codes, future OpenLB uses SymPy to do this automatically

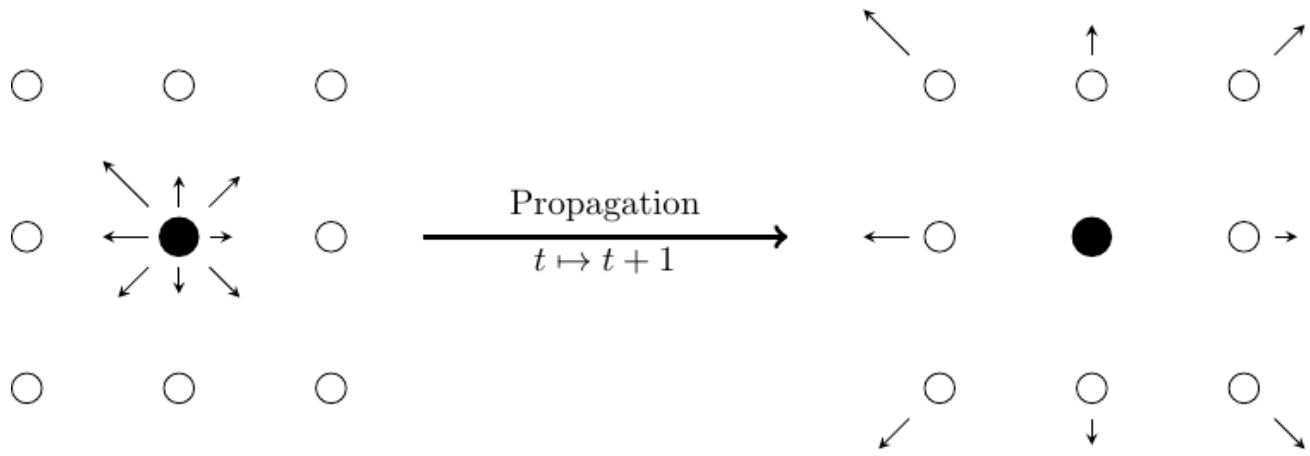
```
struct BgkCollideO {

template <typename V, typename T>
static void apply(V f_curr[19], V f_next[19], T tau) {
  /* [...] */
  V x28 = x15 + x18;
  V x29 = -u[1];
  V x30 = x29 + u[2];
  V x31 = x19 + u[1];
  V x32 = -x16;
  V x33 = x18 + x22;
  V x34 = T{6}*x5;
  V x35 = u[0] + u[1];
  V x36 = T{9}*(x35*x35);
  V x37 = T{6}*x8;
  V x38 = x9 + T{-2};
  V x39 = x29 + u[0];
  V x40 = -x18 + x22;
  V x41 = -u[2];
  V x42 = x41 + u[1];
  V x43 = x22 + x4;
  V x44 = x41 + u[0];
  f_next[0] = x1*(rho*(x12 + x17 + x3) - T{72}*f_curr[0]) +
f_curr[0];
  f_next[1] = -x1*(rho*(x18 + x23 - T{9}*x20*x20) +
T{72}*f_curr[1]) + f_curr[1];
  f_next[2] = x24*(rho*(x12 + x25) - T{36}*f_curr[2]) +
f_curr[2];
  /* [...] */
  f_next[16] = -x24*(rho*(-x25 + x38 + x4 + x6) +
T{36}*f_curr[16]) + f_curr[16];
  f_next[17] = -x1*(rho*(x4 + x40 - T{90}*x44*x44) +
T{72}*f_curr[17]) + f_curr[17];
  f_next[18] = -x1*(rho*(x16 - x3 + x43) + T{72}*f_curr[18]) +
f_curr[18];
}

};
```

Performance Aspect: Propagation Pattern

Collision is fixed set of bandwidth-bound computations



For sufficiently optimized LB collision kernels **propagation pattern is an essential performance aspect** due to bandwidth control

OpenLB uses **directly addressed** grids

Two grid Trivially perfectly parallel but requires two full lattice copies

One grid Uses only a single lattice but more complex implementation

Performance Aspect: Propagation Pattern

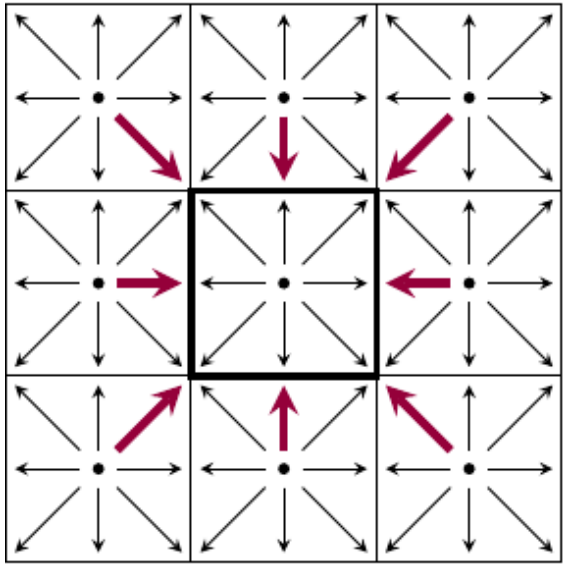
- Up to 1.3r1 OpenLB uses SWAP
- Cache-friendly, Inherently non-parallel, not data-transfer-optimal (3Q-1)
- Starting OpenLB 1.4: Periodic Shift (based on Shift-Swap-Streaming)
- Depends on Sweep SFC, SoA layout
- Implicit Propagation without additional data transfer

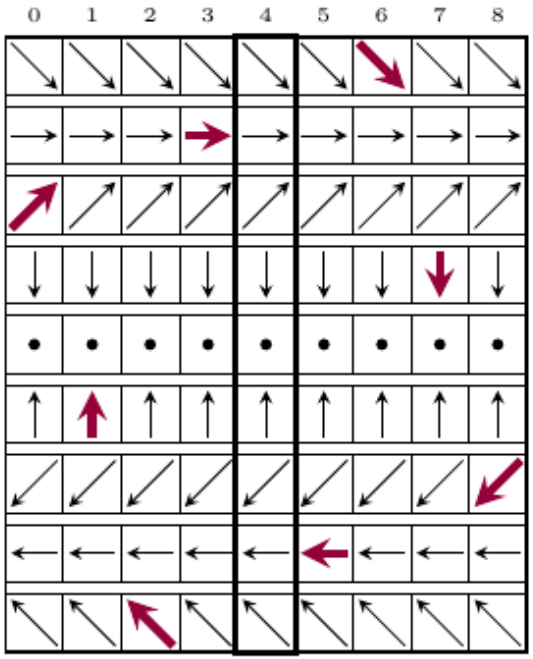
Mattila, K., Hyväluoma, J., Rossi, T. *et al.* (2007). Comput. Phys. Commun., 176(3), 200-210.

Kummerländer, A., Krause, M. J. (2020). In prep. for Comput. Fluids.

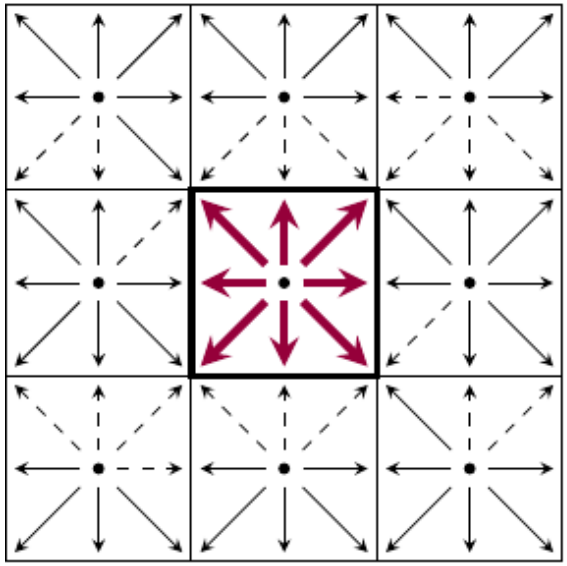
Mohrhard, M., Thäter, G., Bludau, J. *et al.* (2019). Comput. Fluids, 181, 1-7.

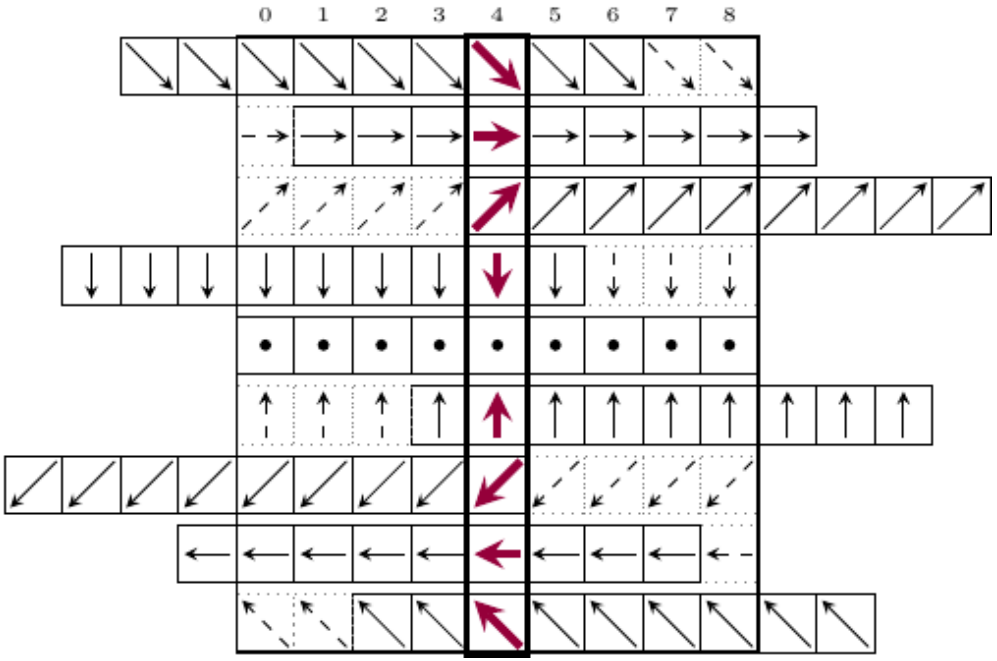
Periodic Shift Pattern





Periodic Shift Pattern





- Propagation is performed implicitly, without per-cell transfer

Agenda

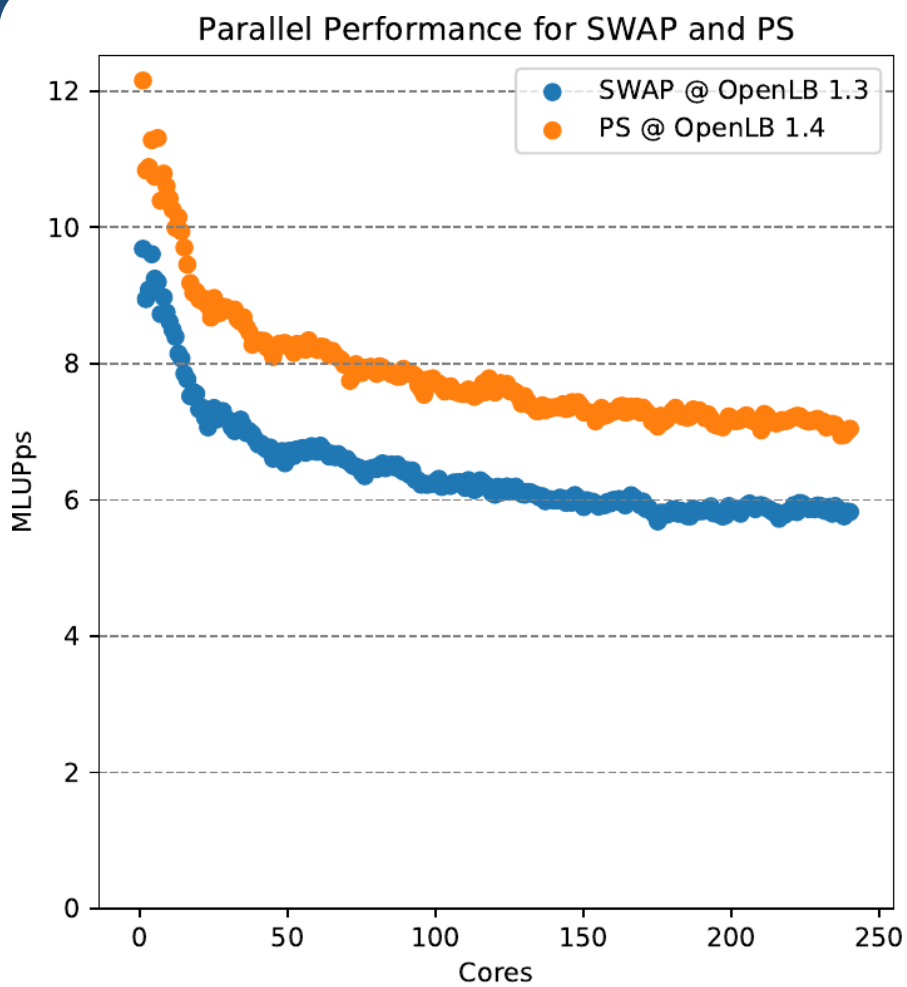
I. Facing Challenges in CFD with LBM and OpenLB

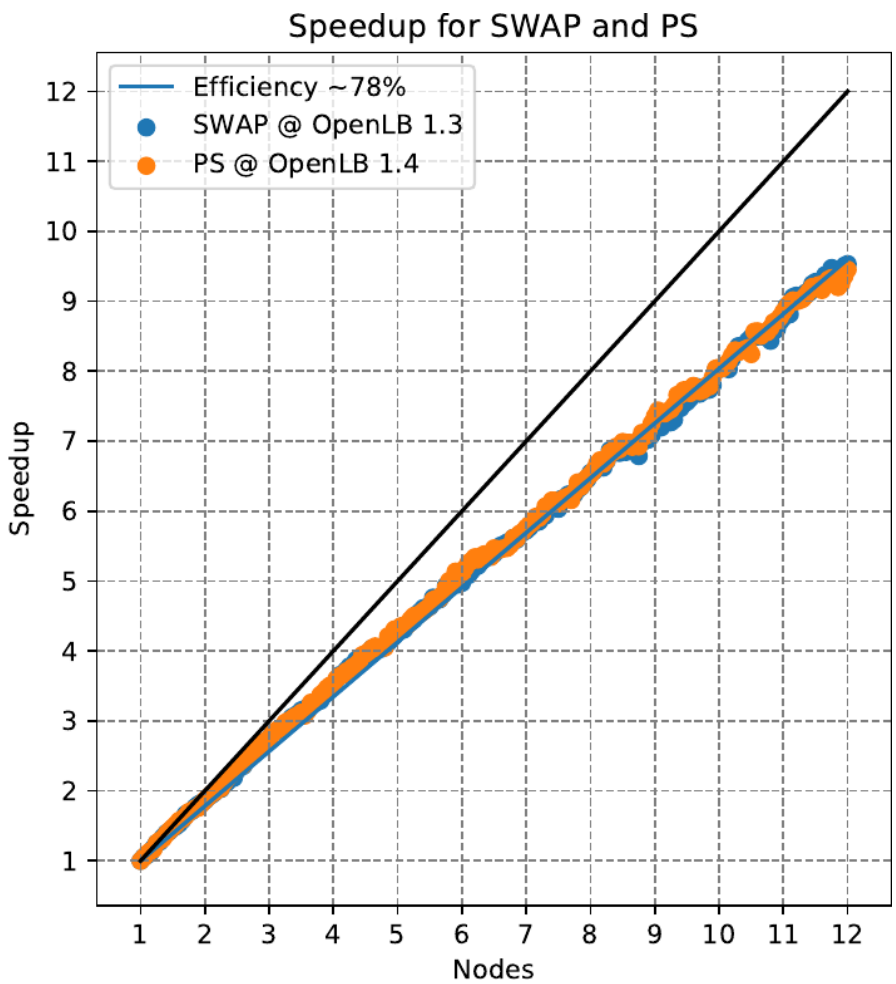
II. LBM on Heterogeneous HPC Platforms

1. OpenLB Concept
2. Parallelization Architecture and Data Structures
3. Performance Aspects: CSE and Propagation Pattern
4. **Parallel Performance of OpenLB**

III. LBM for Turbulent Flows

Comparison between SWAP and PS

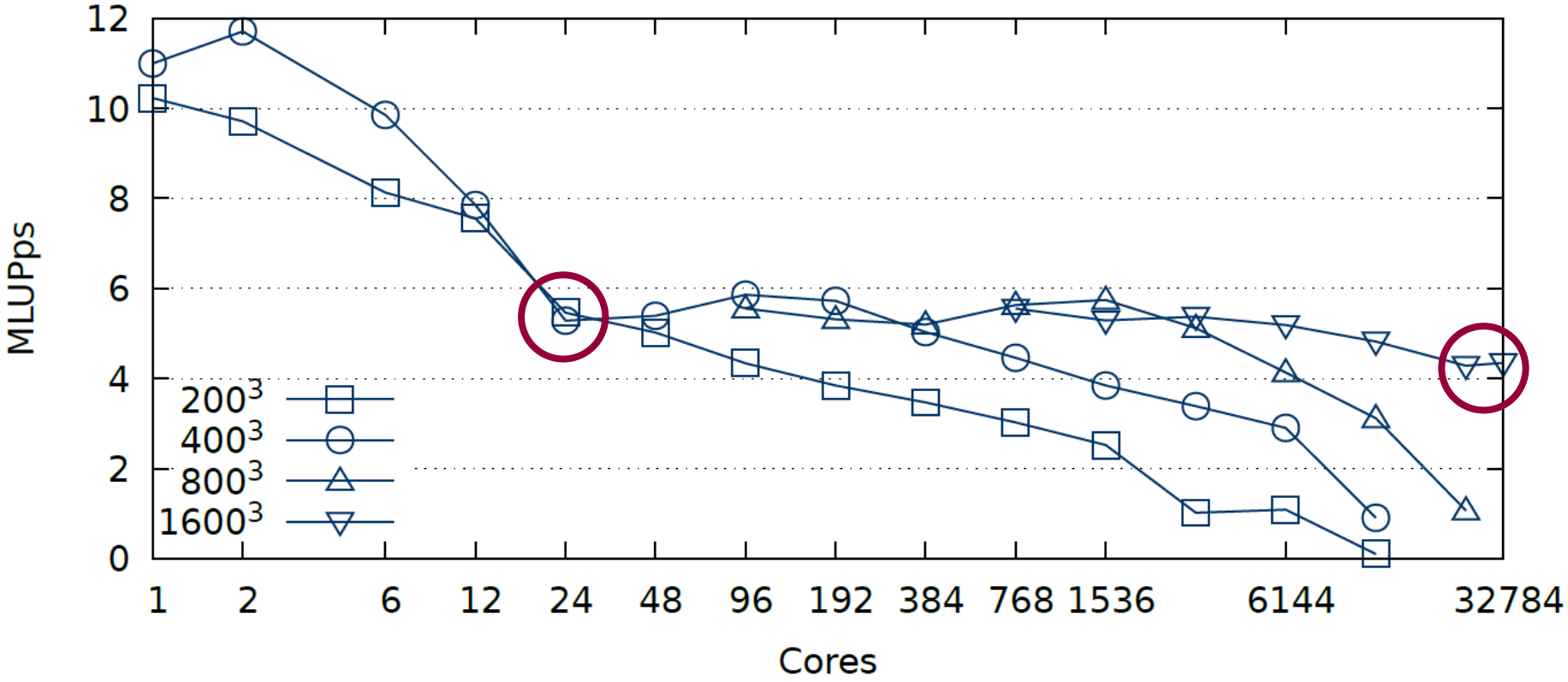




[1] Using 12 dual socket Intel Xeon Silver 4114 nodes

Parallel Performance @ Magnus, Curtin, Australia

Approximately 80% efficiency
1 node ~ 1 cluster (1366 nodes)
46 days ~ 1 hour



Agenda

I. Facing Challenges in CFD with LBM and OpenLB	Mathias J. Krause
II. LBM on Heterogeneous HPC Platforms	Adrian Kummerländer
III. LBM for Turbulent Flows	**Stephan Simonis**

Agenda

I. Facing Challenges in CFD with LBM and OpenLB

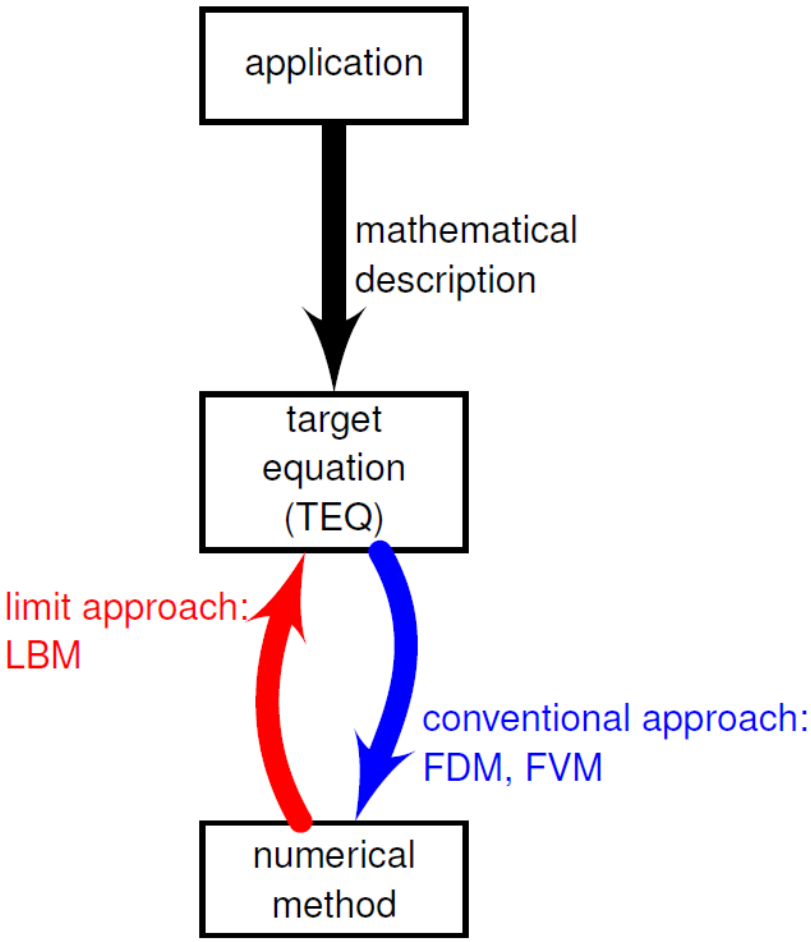
II. LBM on Heterogeneous HPC Platforms

III. LBM for Turbulent Flows

- **1. Approximating a PDE with LBM: A Constructive Approach**
- 2. LBM for Advection–Diffusion Equations
- 3. LBM DNS for Homogeneous Isotropic Turbulence
- 4. Turbulent Flows with LBM LES
- 5. Summary

Approximating a PDE with LBM

Aim:
Obtain computable discrete system of equations from given mathematical problem description in terms of partial differential equations (PDE)

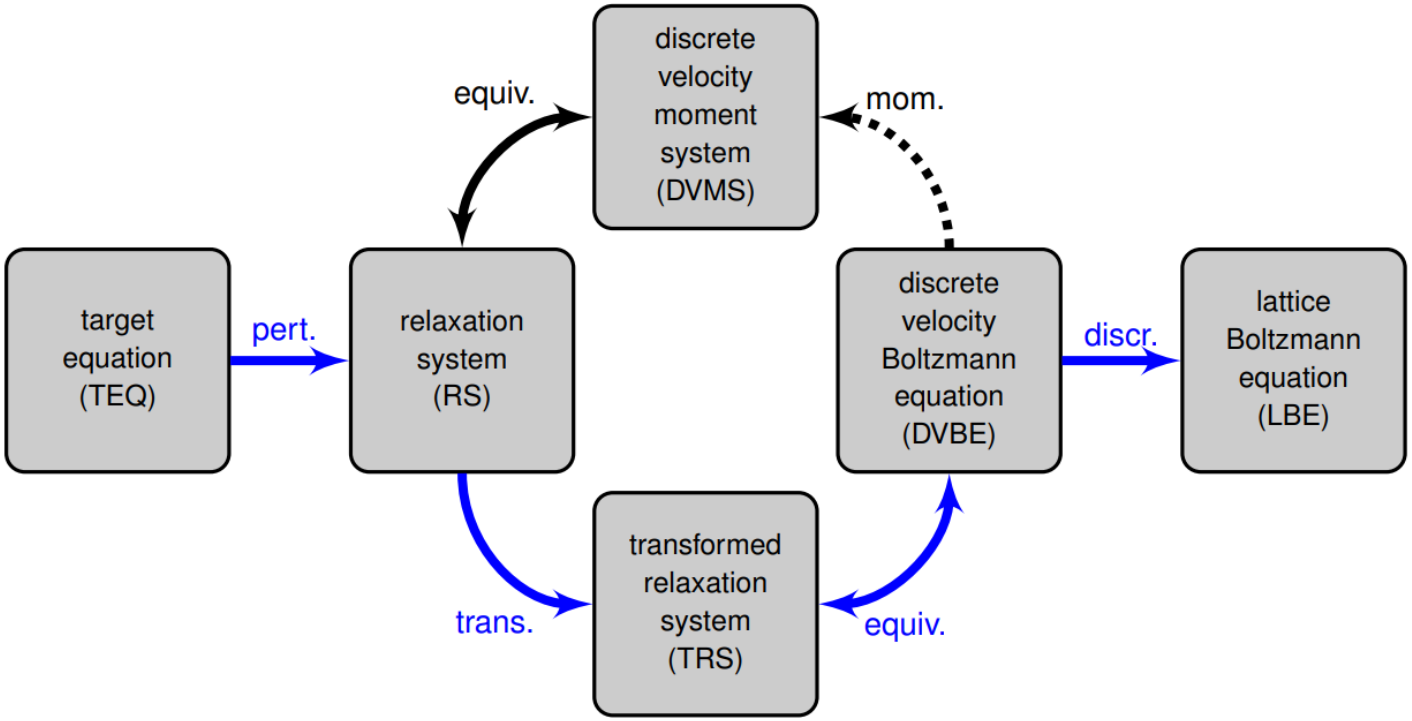


How does LBM work?

- **"Bottom-up"** approach
- Discretization **and** Relaxation limit towards TEQ
- Stability estimates complicated
- Constructive approach?

Relaxation Systems and Discrete Velocity Models

- Formal constructive approach:
  Construct a relaxation system starting from the TEQ
- E.g. via:



Simonis, S., Frank, M., Krause, M. J. (2020). Philos. Trans. Royal Soc. A, 378: 201904.

Agenda

I. Facing Challenges in CFD with LBM and OpenLB

II. LBM on Heterogeneous HPC Platforms

III. LBM for Turbulent Flows

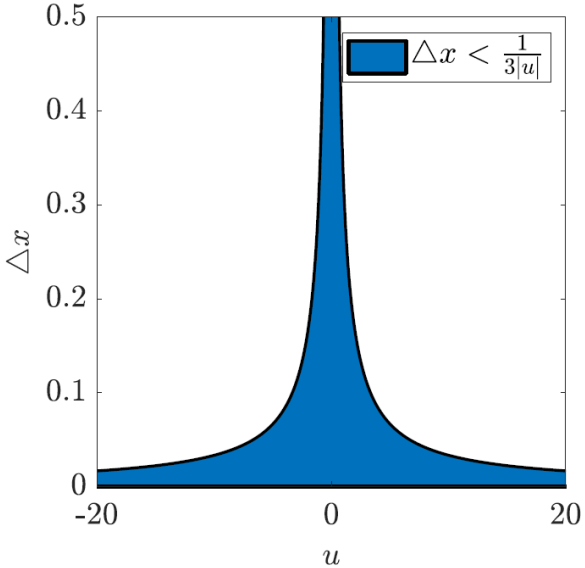
1. Solving a PDE with LBM: A Constructive Approach
2. **LBM for Advection–Diffusion Equations**
3. LBM DNS for Homogeneous Isotropic Turbulence
4. Turbulent Flows with LBM LES
5. Summary

LBM for Advection–Diffusion Equations

TEQ: Scalar linear, one-dimensional advection–diffusion equation (ADE)

$$\partial_t \rho + u \partial_x \rho - \mu \partial_{xx} \rho = 0$$

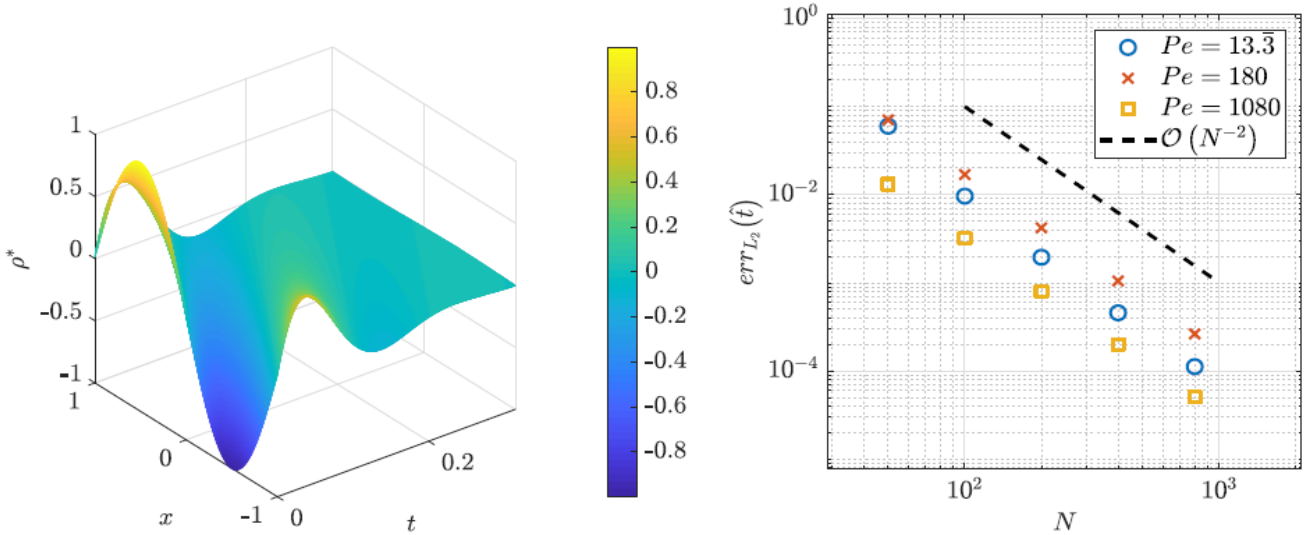
- **Several types of TEQ**, scaling dependent (parabolic/hyperbolic limit, or both)
- Use **constructive approach** to build moment system for initial value problem
- **Construct equilibrium and BGK collision**
- **Discretize** (typically second order in space)
- Along the way: **stability, consistency, convergence**



Simonis, S., Frank, M., Krause, M. J. (2020). Philos. Trans. Royal Soc. A, 378: 201904.

Numerical Example: ADE

- Construct relaxation system and transform (eigendecomposition)
- Obtain discrete velocity system (here with velocity stencil: D1Q3)
- Fully discretized: $f_i(x + c_i \triangle t, t + \triangle t) = f_i(x,t) - \frac{\triangle t}{\tau - \frac{\triangle t}{2}}[f_i(x,t) - f_i^{eq}(x,t)]$
- Advection and diffusion (ADE):



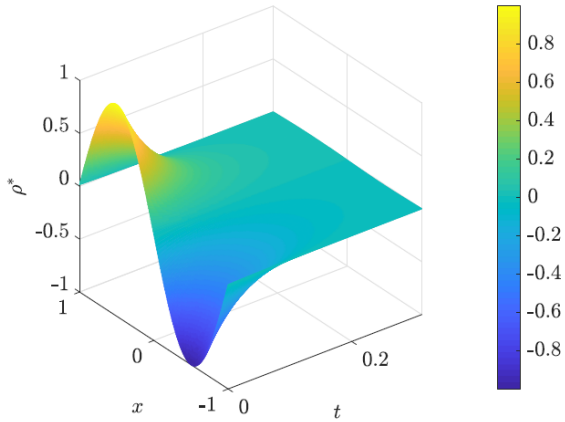
(a) Analytical solution ($Pe = 13.\bar{3}$)

(b) L_2-error at time $\hat{t} = 0.3$

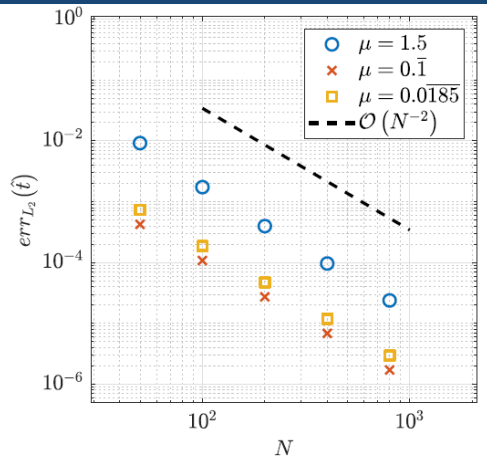
Simonis, S., Frank, M., Krause, M. J. (2020). Philos. Trans. Royal Soc. A, 378: 201904.

Numerical Example: Diffusion/Advection

- Pure diffusion (Heat equation):

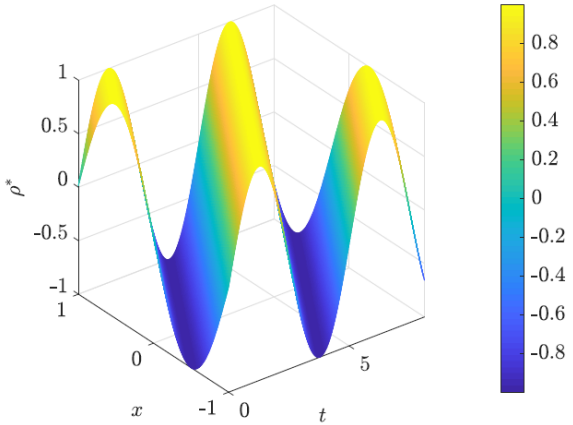


(c) Analytical solution ($u = 0$, $\mu = 1.5$)

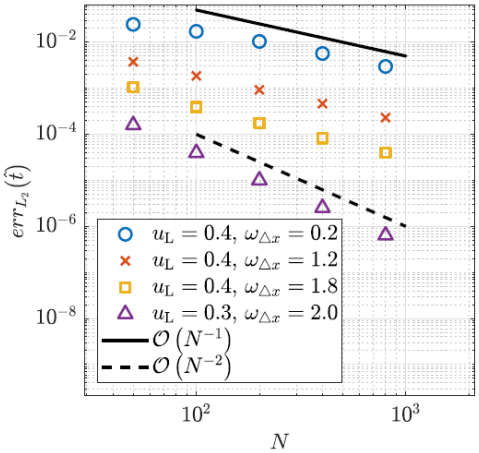


(d) L_2-error at time $\hat{t} = 0.3$

- Pure Convection (Hyperbolic conservation law):



(e) Analytical solution ($u = 10$, $\mu = 0$)



(f) L_2-error at time $\hat{t} = 0.3$

Simonis, S., Frank, M., Krause, M. J. (2020). Philos. Trans. Royal Soc. A, 378: 201904.

Agenda

I. Facing Challenges in CFD with LBM and OpenLB

II. LBM on Heterogeneous HPC Platforms

III. LBM for Turbulent Flows

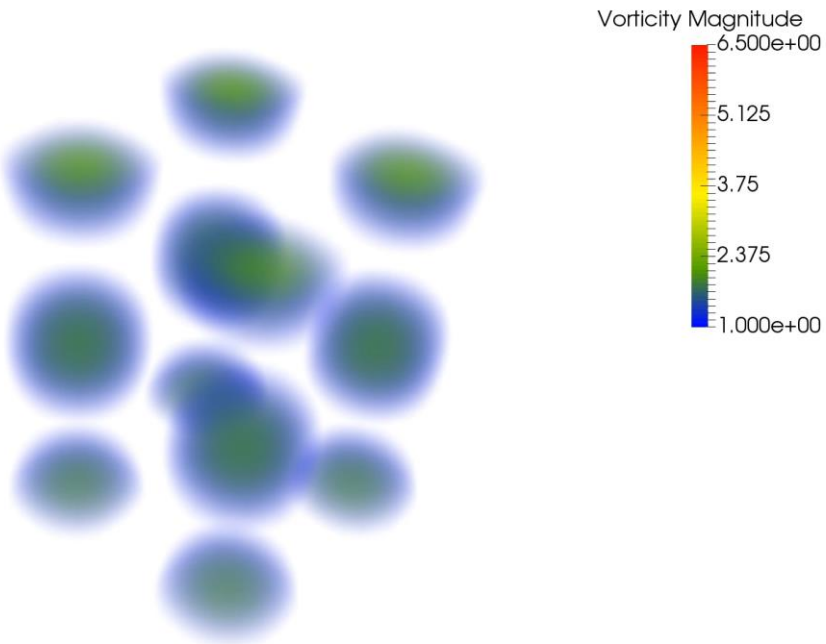
1. Solving a PDE with LBM: A Constructive Approach
2. LBM for Advection–Diffusion Equations
3. **LBM DNS for Homogeneous Isotropic Turbulence**
4. Turbulent Flows with LBM LES
5. Summary

LBM DNS for Homogeneous Isotropic Turbulence (HIT)

TEQ: Incompressible Navier–Stokes equations (NSE)

$$\partial_t \boldsymbol{u} + (\boldsymbol{u} \cdot \boldsymbol{\nabla})\boldsymbol{u} + \boldsymbol{\nabla} p = \nu \boldsymbol{\Delta} \boldsymbol{u}$$
$$\boldsymbol{\nabla} \cdot \boldsymbol{u} = 0$$

- **Direct numerical simulation** with distinct collision schemes: **BGK (SRT), TRT, MRT, RLB, ELB**
- **Taylor–Green vortex** flow $Re = 800, 1600, 3000$
- **Acoustic scaling** ("near CFL" + known error sources) vs. **diffusive scaling** (errors diminished in relaxation/discretization limit)?



Haussmann, M., Simonis, S., Nirschl, H. *et al.* (2019). Int. J. Mod. Phys. C, 30(09):1–29. 2019.

LBM DNS for HIT: Collision Schemes

[dimensionless, shifted notation]

BGK collision operator:

Collide: $f_i^\star(\boldsymbol{x},t) = f_i(\boldsymbol{x},t) + \Omega_i(f_i)$

Stream: $f_i(\boldsymbol{x}+\boldsymbol{c}_i, t+1) = f_i^\star(\boldsymbol{x},t)$

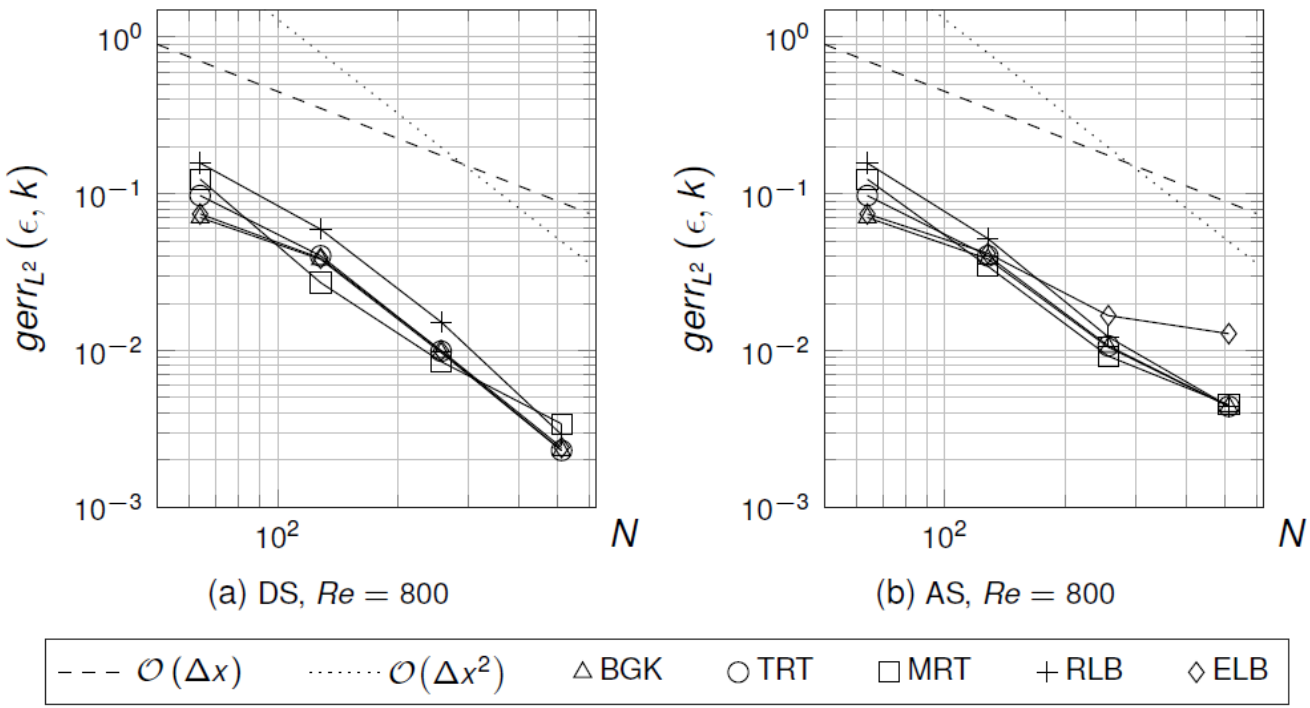
- Single-relaxation-time (SRT): $f_i^\star(\boldsymbol{x},t) = f_i(\boldsymbol{x},t) - \frac{1}{\tau}[f_i(\boldsymbol{x},t) - f_i^{eq}(\rho,\boldsymbol{u})]$
- Two-relaxation-time (TRT): $f_i^\star(\boldsymbol{x},t) = f_i(\boldsymbol{x},t) - \sum_{\circ\in\{+,-\}} \omega^\circ[f_i^\circ(\boldsymbol{x},t) - f_i^{eq,\circ}(\boldsymbol{x},t)]$
- Multiple-relaxation-time (MRT): $\boldsymbol{f}^\star(\boldsymbol{x},t) = \boldsymbol{f}(\boldsymbol{x},t) - M^{-1}SM[\boldsymbol{f}(\boldsymbol{x},t) - \boldsymbol{f}^{eq}(\boldsymbol{x},t)]$
- Regularized LB (RLB): $f_i^\star(\boldsymbol{x},t) = f_i^{eq}(\boldsymbol{x},t) + \left(1 - \frac{1}{\tau}\right) f_i^{(1)}(\boldsymbol{x},t)$
- Entropic LB (ELB): $f^\star(\boldsymbol{x},t) = f(\boldsymbol{x},t) - \alpha\beta[\check{f}_i^{eq}(\boldsymbol{x},t) - f_i(\boldsymbol{x},t)]$

Calculate error err_{L^2} of dissipation rate $\epsilon(t)$ and total kinetic energy $k(t)$, and average:

Haussmann, M., Simonis, S., Nirschl, H. *et al.* (2019). Int. J. Mod. Phys. C, 30(09):1–29. 2019.

LBM DNS for HIT: Diffusive vs Acoustic Scaling

- $Re = 800$
- **Diffusive scaling (DS):** $Ma \xrightarrow{N \to \infty} 0$
- **Acoustic scaling (AS):** $Ma = const.$



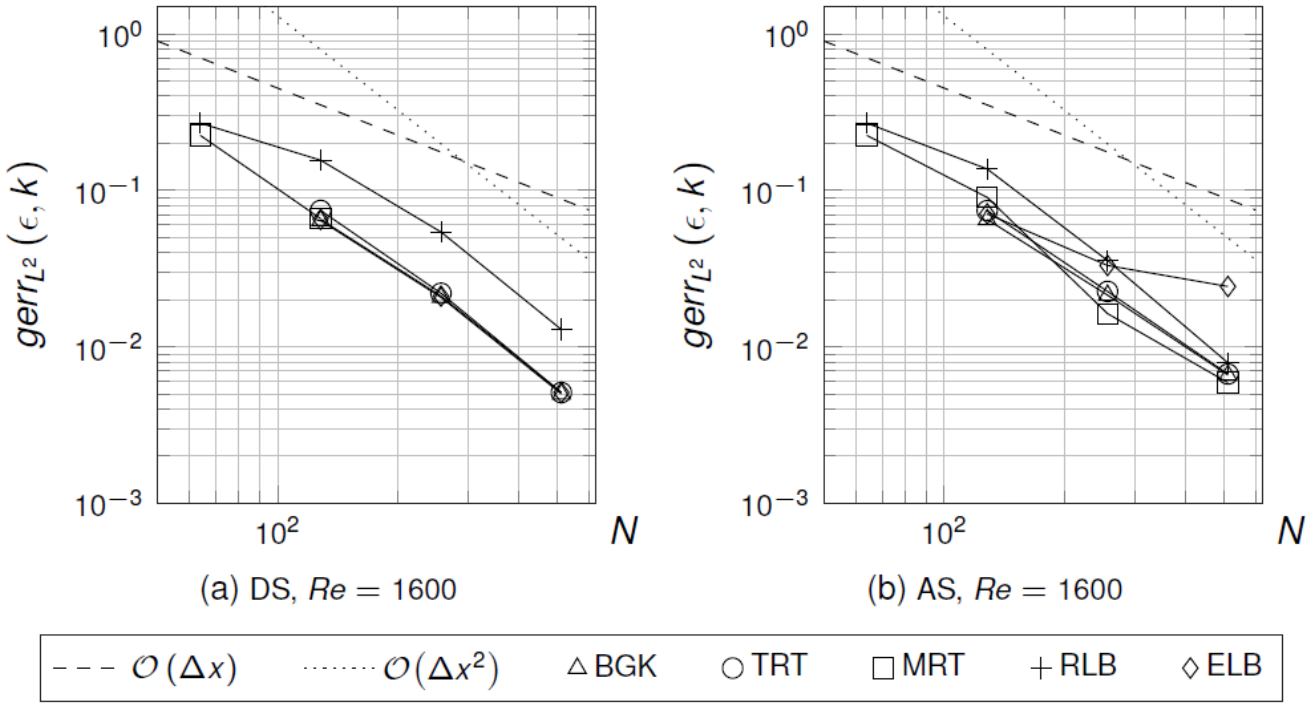
Conclusions:

- Fully resolved scales for highest resolution, no instabilities
- DS: 2nd order convergence
- AS: Mach number plateau visible for $N > N_\kappa \approx 313$

Haussmann, M., Simonis, S., Nirschl, H. *et al.* (2019). Int. J. Mod. Phys. C, 30(09):1–29. 2019.

LBM DNS for HIT: Diffusive vs Acoustic Scaling

- $Re = 1600$
- **DS vs AS**



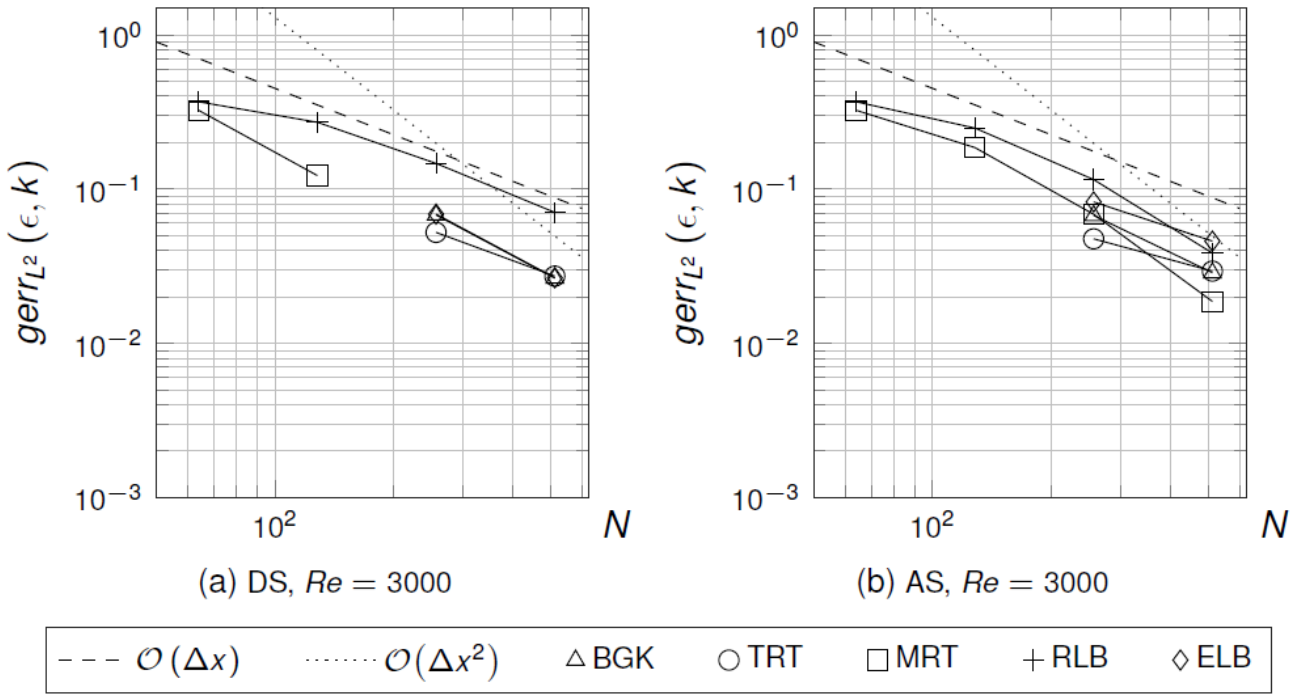
Conclusions:

- AS and DS: BGK, TRT, ELB show coarse resolution instabilities.
- DS: MRT shows Mach number dependent instabilities

Haussmann, M., Simonis, S., Nirschl, H. *et al.* (2019). Int. J. Mod. Phys. C, 30(09):1–29. 2019.

LBM DNS for HIT: Diffusive vs Acoustic Scaling

- $Re = 3000$
- **DS vs AS**



Conclusions:

- AS: Kolmogorov length barely resolved (no plateau visible)
- Earlier instability onset: MRT for DS: Mach dependent;
  BGK, TRT, ELB for AS and DS: resolution dependent

Haussmann, M., Simonis, S., Nirschl, H. *et al.* (2019). Int. J. Mod. Phys. C, 30(09):1–29. 2019.

LBM DNS for HIT: Outlook

- Parameter dependent instabilities of BGK and MRT
- Parameter dependent advantages of BGK and MRT

Choose relaxation frequencies for MRT which prevent instabilities?

[implicit LES?]

Simonis, S., Oberle, D., Gaedtke, M. *et al.* (2020). Submitted to J. Comput. Phys.

Agenda

I. Facing Challenges in CFD with LBM and OpenLB

II. LBM on Heterogeneous HPC Platforms

III. LBM for Turbulent Flows

1. Solving a PDE with LBM: A Constructive Approach
2. LBM for Advection–Diffusion Equations
3. LBM DNS for Homogeneous Isotropic Turbulence
4. **Turbulent Flows with LBM LES**
5. Summary

Turbulent Flows with LBM LES

TEQ: Filtered incompressible Navier–Stokes equations (FNSE)

$$\partial_t \overline{\boldsymbol{u}} + (\overline{\boldsymbol{u}} \cdot \boldsymbol{\nabla})\overline{\boldsymbol{u}} + \boldsymbol{\nabla}\bar{p} = \nu \boldsymbol{\Delta}\overline{\boldsymbol{u}} - \boldsymbol{\nabla} \cdot T$$
$$\boldsymbol{\nabla} \cdot \overline{\boldsymbol{u}} = 0$$

where $T = \overline{\boldsymbol{u}\boldsymbol{u}} - \overline{\boldsymbol{u}}\,\overline{\boldsymbol{u}}$ is the subgrid/residual stress tensor.

- **Consistent filtering** of DVBE
- Discretization results in

$$\bar{f}_i^{\star}(\boldsymbol{x}, t) = \bar{f}_i(\boldsymbol{x}, t) - \frac{1}{\tau}\left[\bar{f}_i(\boldsymbol{x}, t) - f_i^{eq}(\bar{\rho}, \overline{\boldsymbol{u}})\right] + R_i,$$

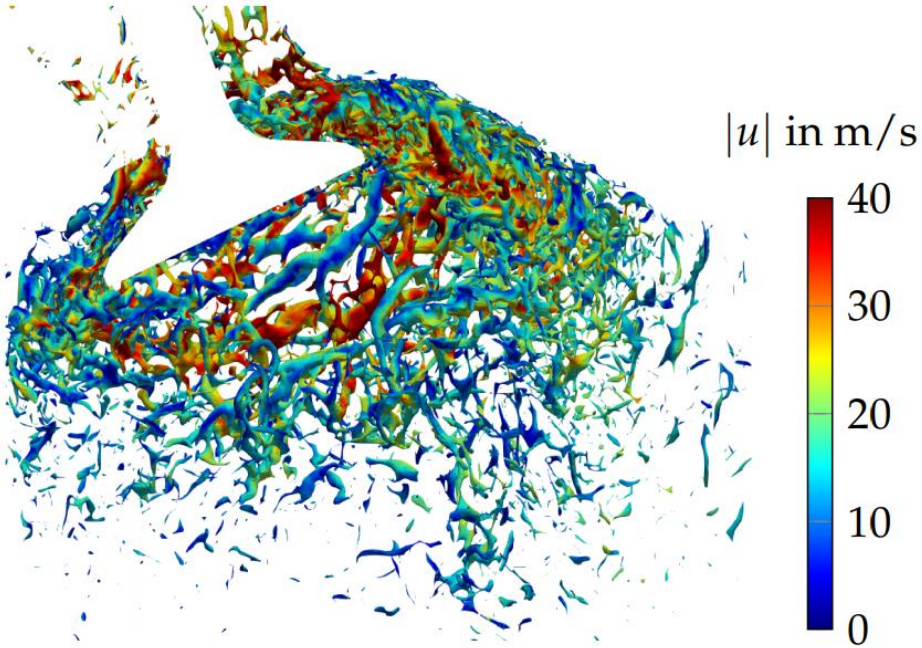
  where $\bar{\rho}$ and $\overline{\boldsymbol{u}}$ are filtered moments.
- **Subgrid scale (SGS) closures** via order by order expansion of residual R_i.

Malaspinas, A., Sagaut, P. (2012). J. Fluid Mech. 700, 514-542.

Turbulent Flows with LBM LES: Applications

- **Wall models** for LBM LES, e.g.:
  Near-wall-models (NWM)
  with SRT LBM

- **NWM-LES LBM for**
  Complex turbulent flows
  relevant to
  internal combustion engines

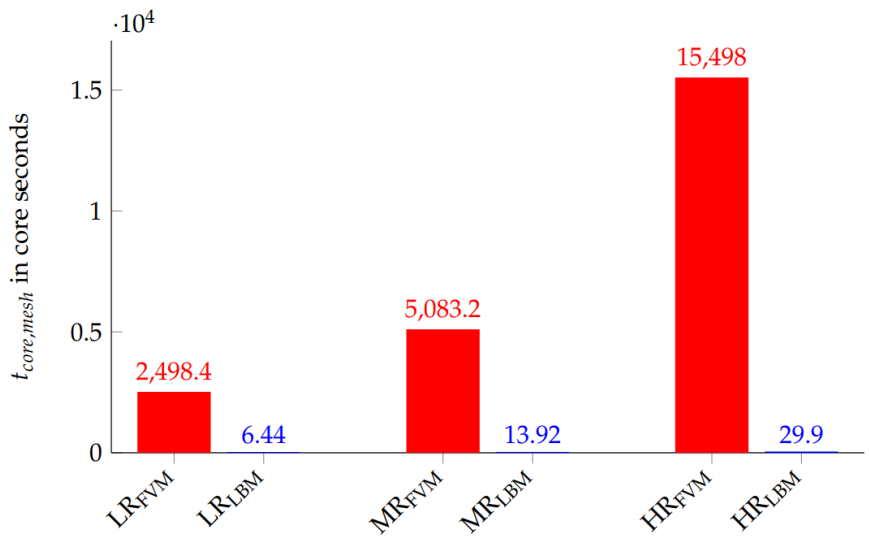


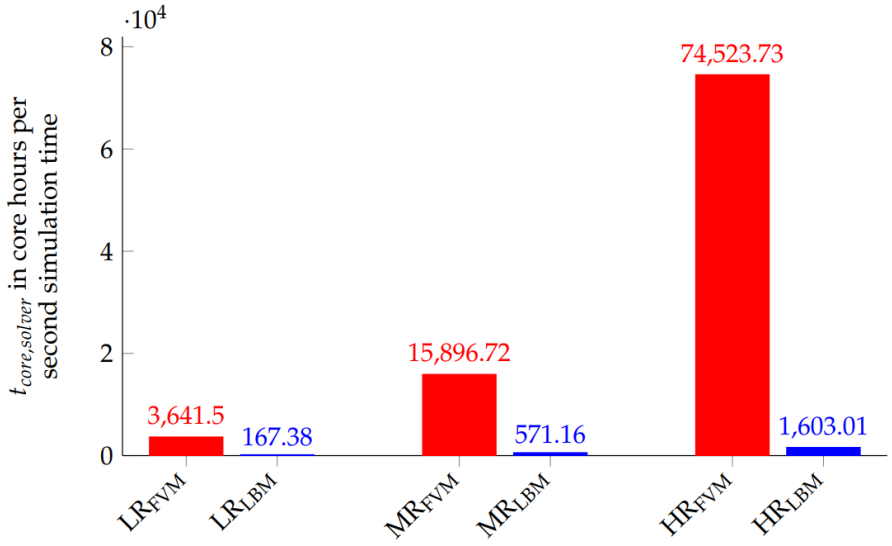
- **Comparison of OpenFOAM and OpenLB** w.r.t.:
  capability of **prediction accuracy, computational cost, ease of use**.

Haussmann, M., Barreto, A. C., Kouyi, G. L. *et al.* (2019). Comput. Math. App. 78, 3285–3302.

Haussmann, M., Ries, F., Jeppener-Haltenhoff, J. B. *et al.* (2020). Computation 2020, 8(2), 43.

OpenLB (LBM) vs OpenFOAM (FVM)



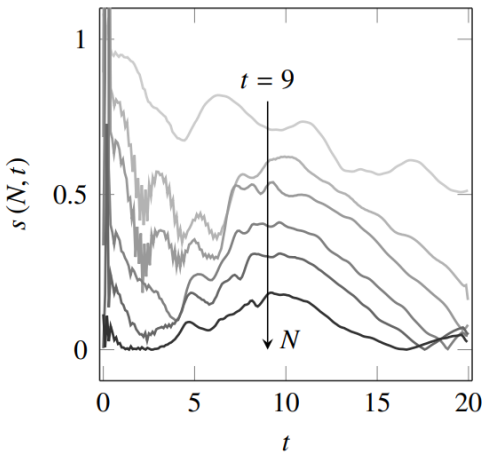


- Similar **NWM-LES**:
  Smagorinsky–Lilly, van Driest damping, Musker wall function
- Similar **prediction accuracy**
- **computational cost for the present setup**:
  meshing with OpenLB is 424x faster than with OpenFOAM
  simulation with OpenLB is 32x faster than with OpenFOAM

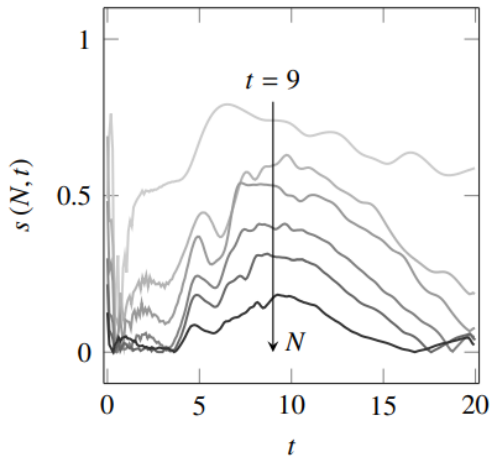
Haussmann, M., Ries, F., Jeppener-Haltenhoff, J. B. *et al.* (2020). Computation 2020, 8(2), 43.

Turbulent Flows with LBM LES: Outlook

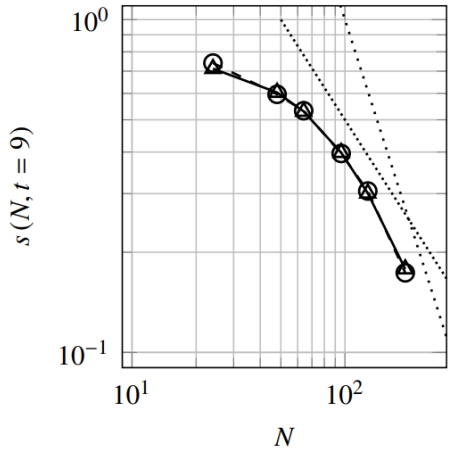
- **LBM for novel LES techniques**:
  Temporal LES (TLES) via temporal direct deconvolution method (TDDM)
- MRT TLES retains high parallelizability of LBM (discretized TDDM **local in space**) and **consistency** up to known sources of errors.



(a) Subgrid activity for LBM TLES with various resolutions: —— $N = 24$, —— $N = 48$, —— $N = 64$, —— $N = 96$, —— $N = 128$, —— $N = 192$.



(b) Subgrid activity for LBM UDNS with various resolutions: —— $N = 24$, —— $N = 48$, —— $N = 64$, —— $N = 96$, —— $N = 128$, —— $N = 192$.



(c) Subgrid activity of UDNS (- ⊖-) and TLES (—△—) with LBM. Reference slopes: ····· $O(N^{-1})$, ····· $O(N^{-2})$.

Oberle, D., Pruett, C. D., Jenny, P. (2020). Phys. Fluids 32, 065112.

Simonis, S., Oberle, D., Gaedtke, M. *et al.* (2020). Submitted to J. Comput. Phys.

Agenda

I. Facing Challenges in CFD with LBM and OpenLB

II. LBM on Heterogeneous HPC Platforms

III. LBM for Turbulent Flows

1. Solving a PDE with LBM: A Constructive Approach
2. LBM for Advection–Diffusion Equations
3. LBM DNS for Homogeneous Isotropic Turbulence
4. Turbulent Flows with LBM LES
5. **Summary**

Summary

Facing challenges in CFD:

Turbulence

DNS/LES instead of RANS

Suspensions

Resolve particles (shape, force, …)

Optimal Control/ Optimization

Algorithmic differentiation, adjoints, CFD-MRI

LBM & *OpenLB*: meshing and high performance at your fingertips!

5th Spring School: LBM with OpenLB Software Lab

5th Spring School

Lattice Boltzmann Methods with OpenLB Software Lab

Greenwich, England, UK, 31. May - 4. June 2021

- for scientists and industry, beginners level
- comprehensive **theoretical lectures on LBM**
- **mentored training** on case studies using *OpenLB*, **bring your own problem**
- knowledge exchange, networking at poster session, coffee breaks and excursion

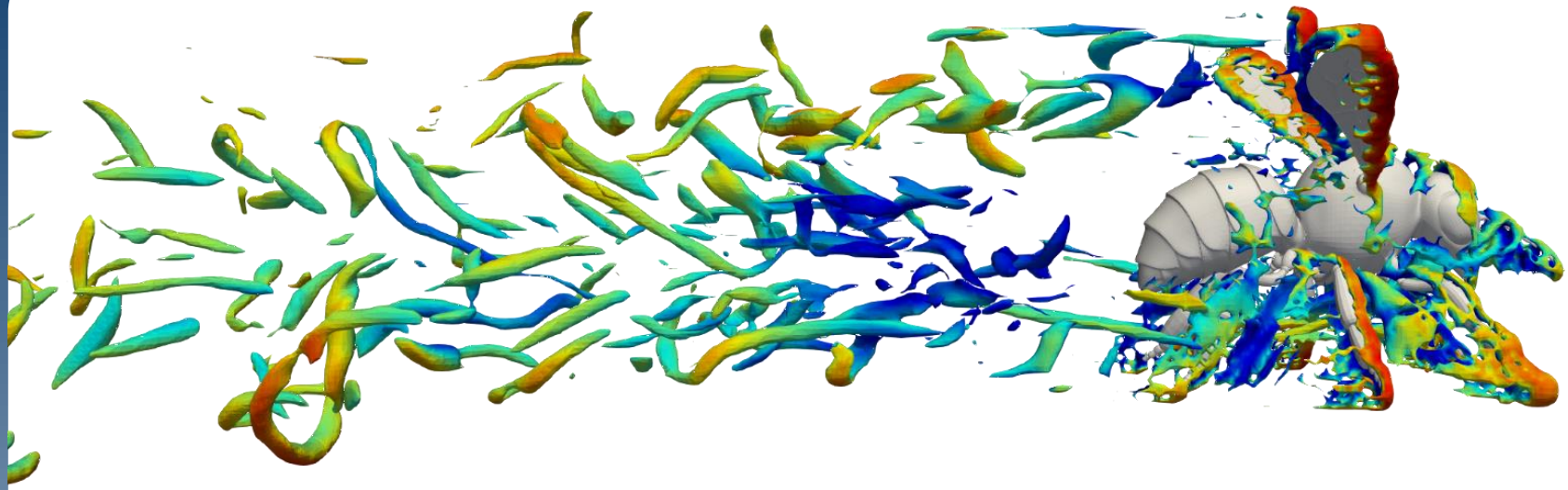
300 £ academia/1.300 £ industry for 5 days course including course material, 5x lunch, 2x dinner, coffee breaks and excursion

Executive committee
Nicolas Hafen, **Mathias J. Krause**, Jan E. Marquardt, **Timothy Reis**, Choi-Hong Lai, Tao Gao

Invited speakers
Timm Krüger, Halim Kusumaatmaja, Francois Dubois

Karlsruhe Institute of Technology

UNIVERSITY of GREENWICH



Questions?

Download, try, participate…

Download | User Guide | Forum