A GPU-enabled smoothed particle hydrodynamics (SPH) method for micro- and nanofluidic simulations

Daniel Gaudlitz

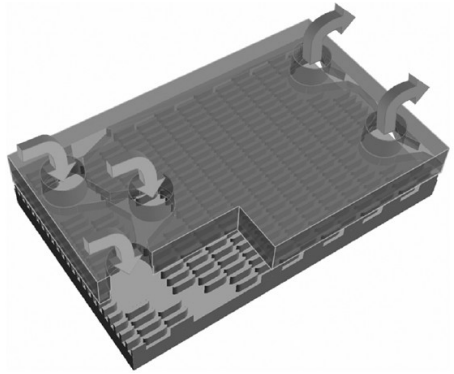
Institute of Aerodynamics and Fluid Mechanics
Technische Universität München

Objective

Analysis of micro- and nanofluidic flows by numerical simulation

- Complex geometries of flow domain boundaries
- Multiphase flows
- Heat transfer
- Phase changes
- Focus on laminar flows
- Analyzing flows problems in reasonable time

→ **Implementation of a SPH method featuring all relevant physical models and making extensive use of GPUs**



Colgan et al., 2007

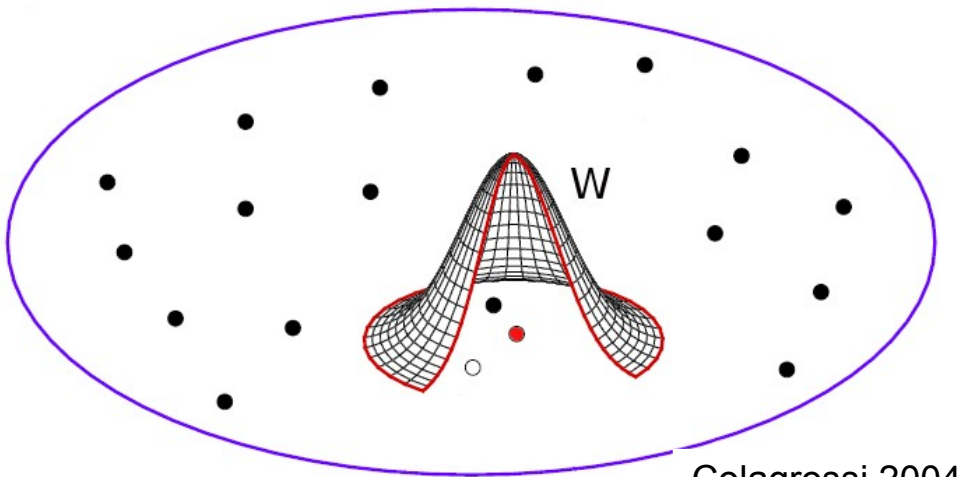
Lutz et al., 2010

SPH method

- Approximation of spatial function by integral interpolation based on disordered elements
- Particles associated with certain properties $\vec{x}, m, \vec{u}, T$
- Particle properties change due to particle interaction
- Determination of value at location *r* of variable *A*

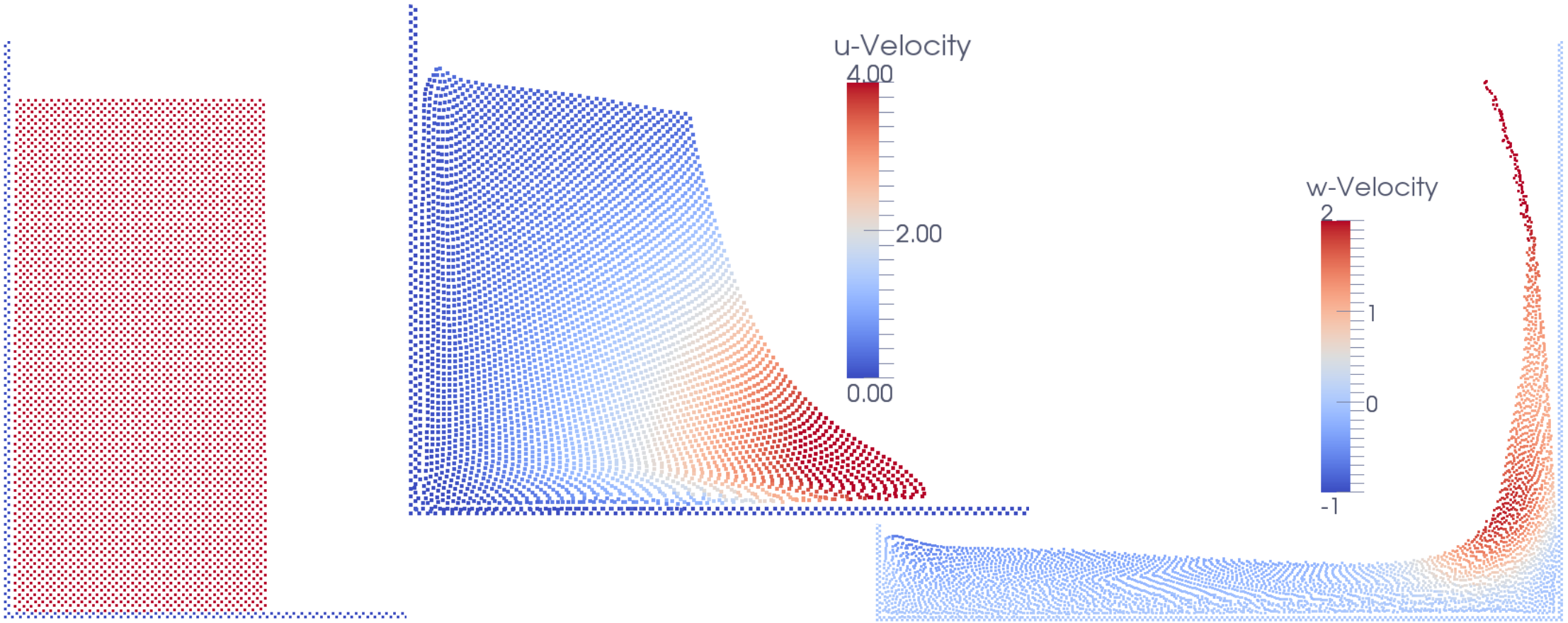
$$A(\vec{r}) = \sum \frac{m_b}{\rho_b} A(\vec{r}_b) W(\vec{r} - \vec{r}_b, h)$$

- Interpolation functions W (Kernel)
  - Gaussian
  - Quadratic
  - Cubic spline
  - Quintic

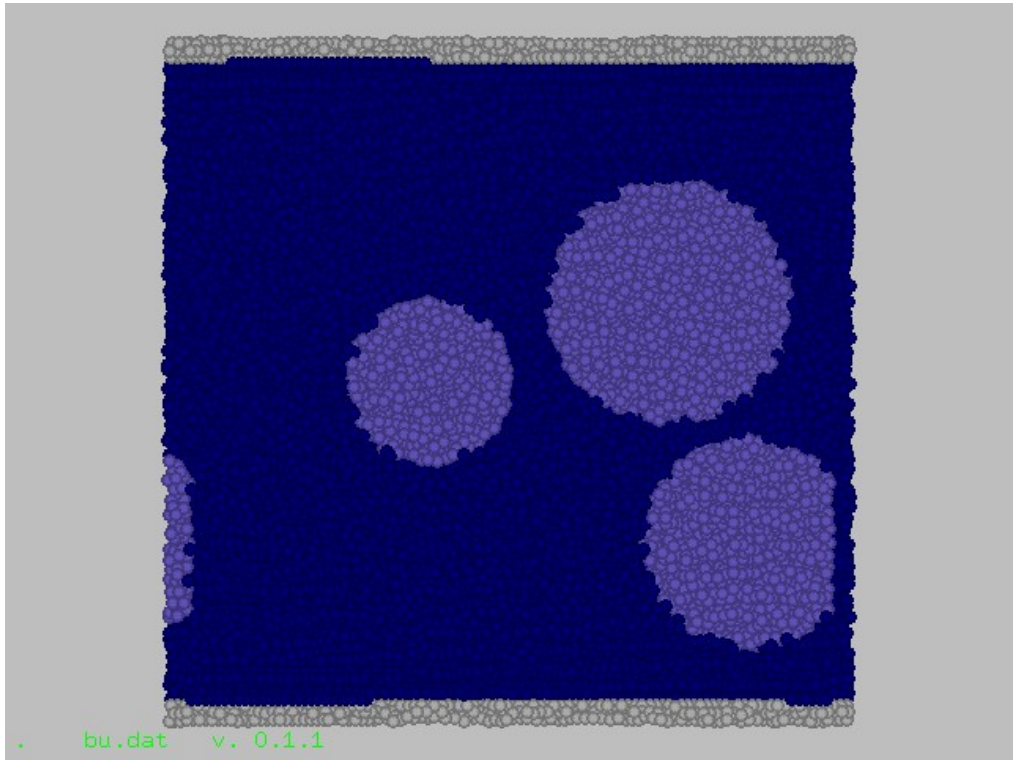


Colagrossi 2004

Flow field with SPH methods



Two-phase flows with SPH methods



Microchannels for cooling electronic devices

Liquid in channel with heated walls

↓

Boiling at walls, hot spots

↓

Phase change at walls

↓

Multiphase flow with different flow regimes

Concept for a SPH method on GPUs

- Porting of the majority of the code to GPU
  - Data transfer CPU – GPU via PCI only during initialization and for data output

- Grouping of computational elements
  - Tree-like structure of cells for mapping of computational domain
  - Filling of cells with reasonable amount of SPH particles
  - Allocation of neighboring cells on neighboring hardware

- Buffered operations: Accumulation of tasks
  - E.g. insertion of particles only if sufficient amount of new particles are requested

Performance of GPU implementation

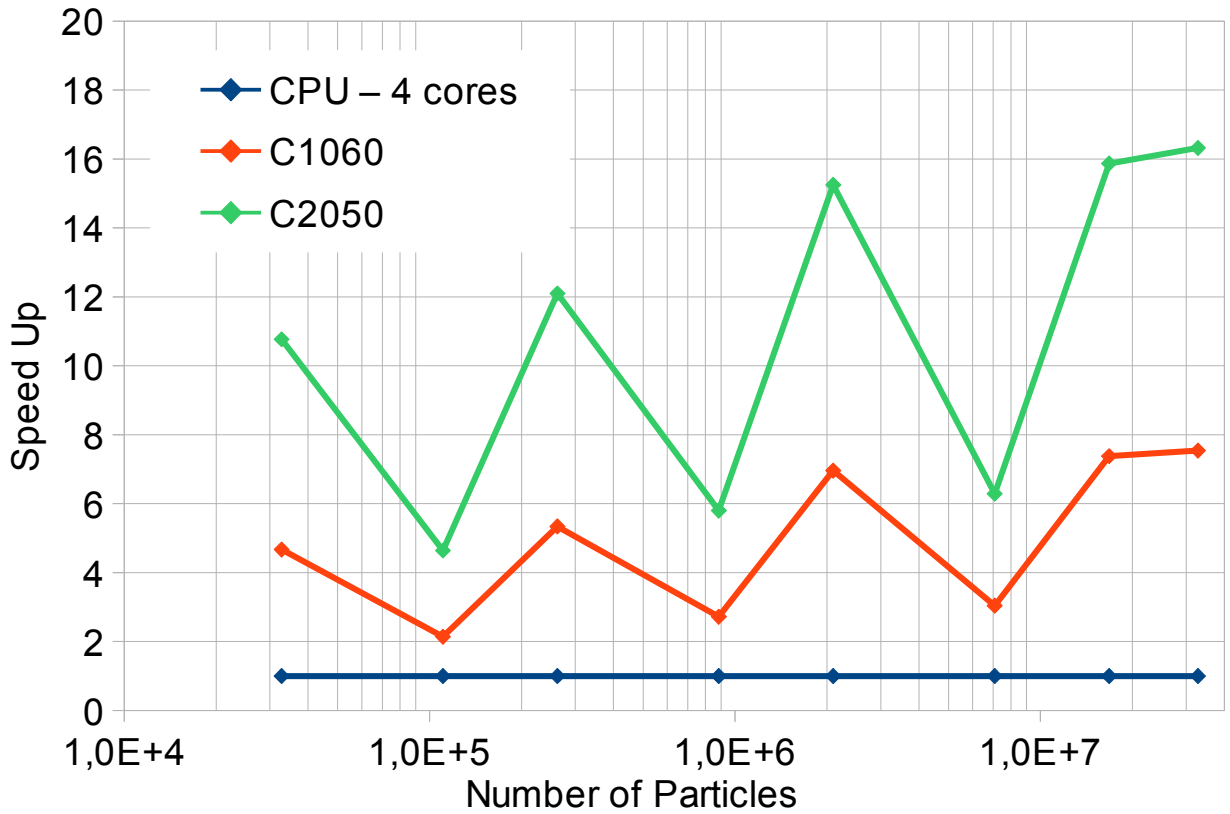
- Operation

  Compute particle density via

$$\rho_P = \sum m_b W\left(\vec{r}_P - \vec{r}_b, h\right)$$

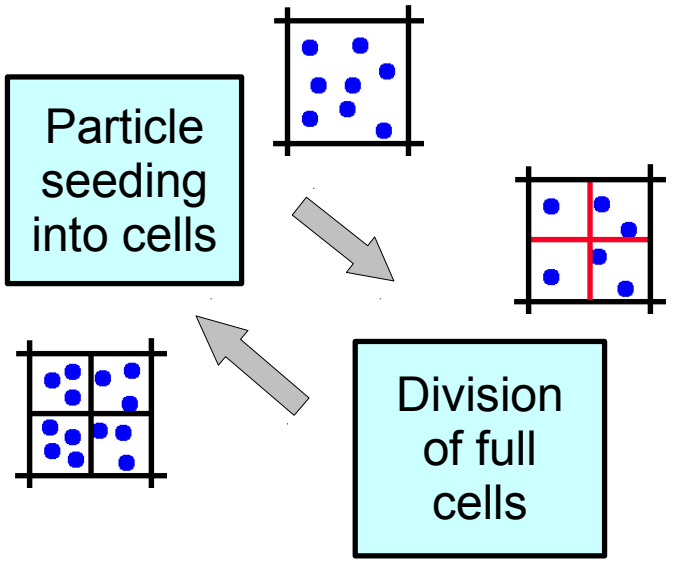
  for periodic domain

- Hardware
  - AMD Phenom 9950 2.6 GHz
  - Nvidia Tesla C1060
  - Nvidia Tesla C2050



Analysis of GPU performance

- Generation of SPH particles



# Cells	Min. particles	Max. particles
1	0	256
8	257	2.048
64	2.049	16.384
512	16.385	131.072
4.096	131.073	1.048.576
32.768	1.048.577	8.388.608
262.144	8.388.609	67.108.864

Analysis of GPU performance

GPU data structure			Case setup			
# Cells	Min. particles	Max. particles	Employed resolution		Particles per Cell	Warps per Cell
1	0	256				
..	..	..				
512	16.385	131.072	32x32x32	32.768	64	2
			48x48x48	110.592	216	6,75 (!)
4.096	131.073	1.048.576	64x64x64	262.144	64	2
			96x96x96	884.736	216	6,75 (!)
32.768	1.048.577	8.388.608	128x18x128	2.097.152	64	2
			192x192x192	7.077.888	216	6,75 (!)
262.144	8.388.609	67.108.864	256x256x256	16.777.216	64	2
			320x320x320	32.768.000	125	3,91 (!)

Summary

- SPH code for non-isothermal, multiphase microfluidic flow simulations
- Parallel CPU and GPU versions being developed
- Key features of the GPU code
  - Maximizing GPU usage, CPU for data output only
  - Mapping of particles to cells for efficient execution on GPU
  - Buffered operations
- Ongoing work
  - Extension of physical models within GPU version
  - Adaptivity of particle mapping to cells (local coarsening / refinement)

Financial support by BMBF is gratefully acknowledged.