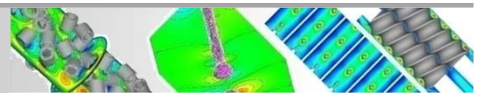
On FEM techniques for multiphase flow

Recent developments regarding Numerics and CFD Software

Stefan Turek, Otto Mierka, Raphael Münster
Institut für Angewandte Mathematik, LS III
Technische Universität Dortmund
ture@featflow.de

http://www.mathematik.tu-dortmund.de/LS3
http://www.featflow.de



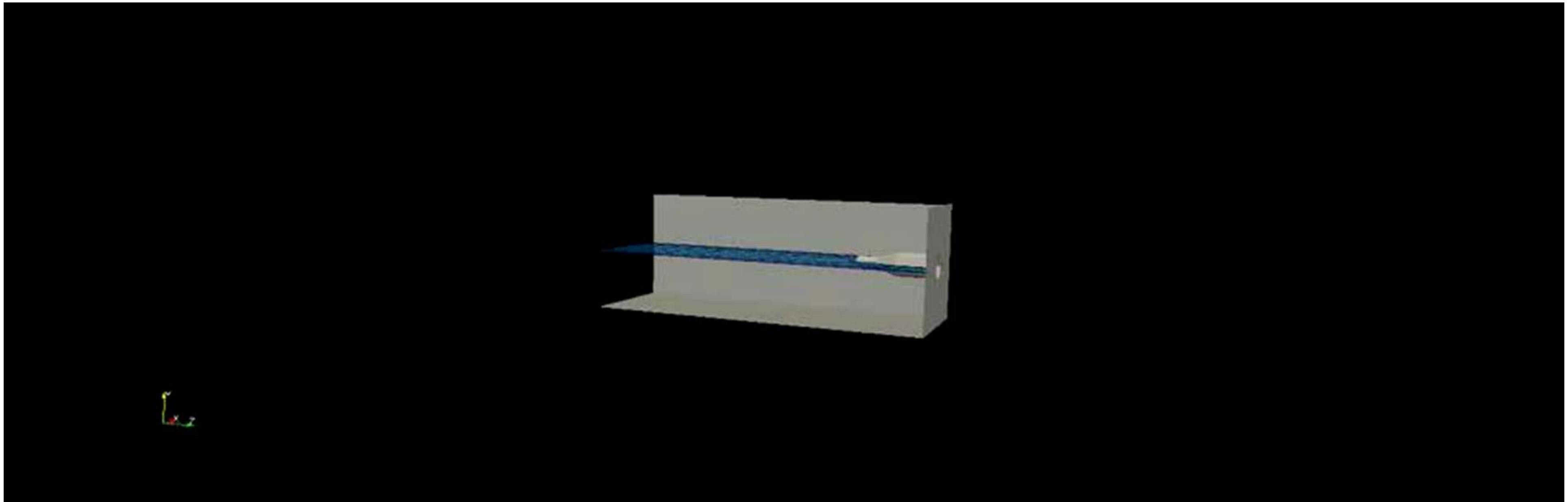
Overview & Motivation:

Accurate, robust, flexible and efficient simulation of multiphase problems with dynamic interfaces and complex geometries, particularly in 3D, is still a challenge!

- Mathematical Modelling
- Numerics / CFD Techniques
- Validation / Benchmarking
- HPC Techniques / Software

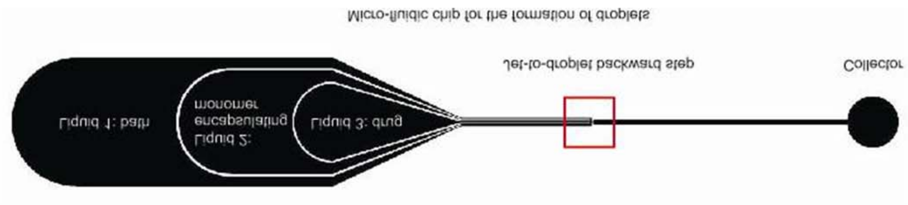
Vision: *Highly efficient, flexible and accurate „real life“ simulation tools based on modern Numerics and algorithms while exploiting modern hardware!*

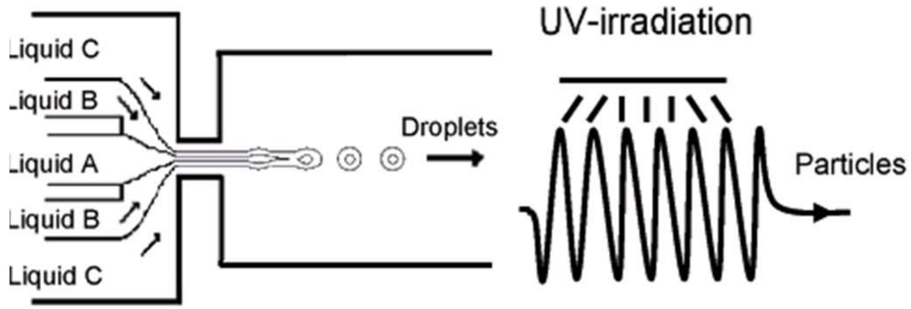
Realization: FEATFLOW

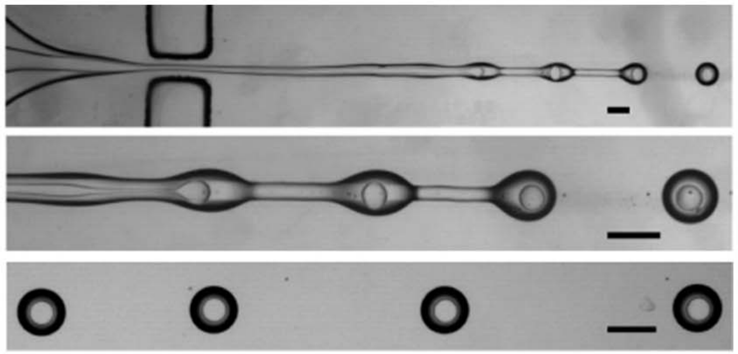


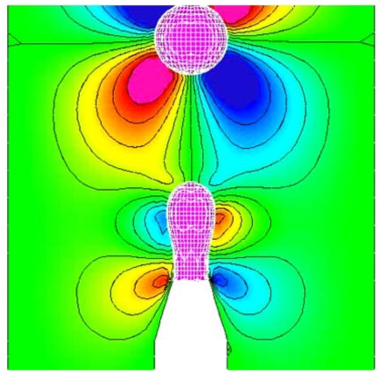
Exemplary Application

- Numerical simulation of *micro-fluidic drug encapsulation ("monodisperse compound droplets")* for application in lab-on-chip and bio-medical devices
- Polymeric "bio-degradable" outer fluid with viscoelastic effects
- Optimization of chip design w.r.t. boundary conditions, flow rates, droplet size, geometry

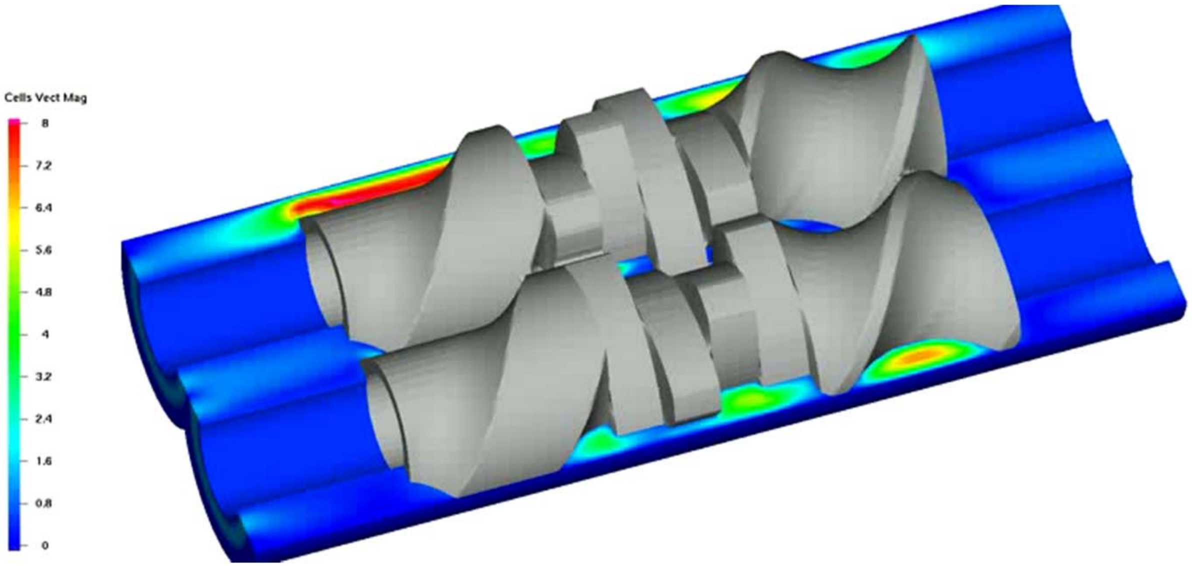


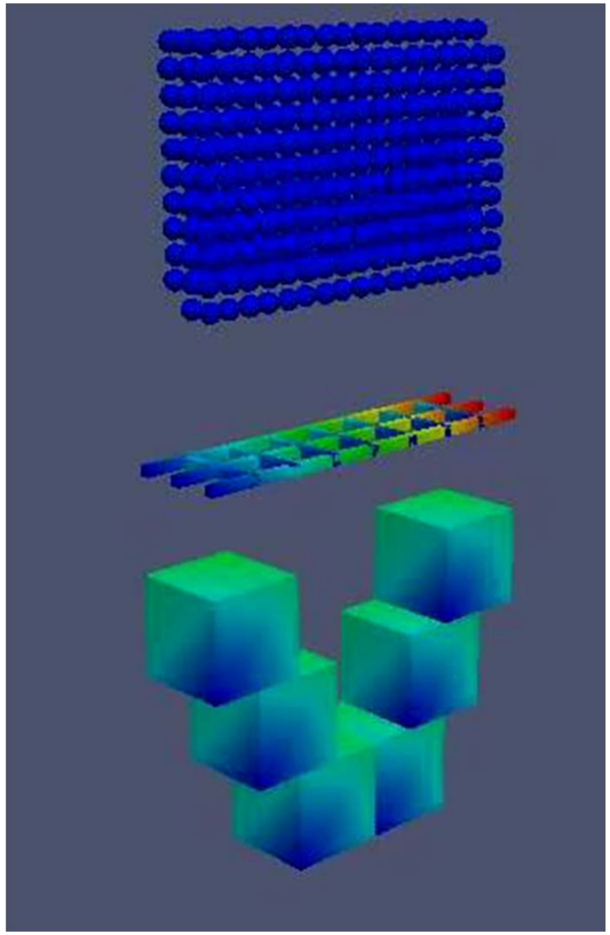






Typical applications require efficient basic flow solvers and techniques for liquid-liquid & liquid-solid interfaces in complex (time-dependent) domains

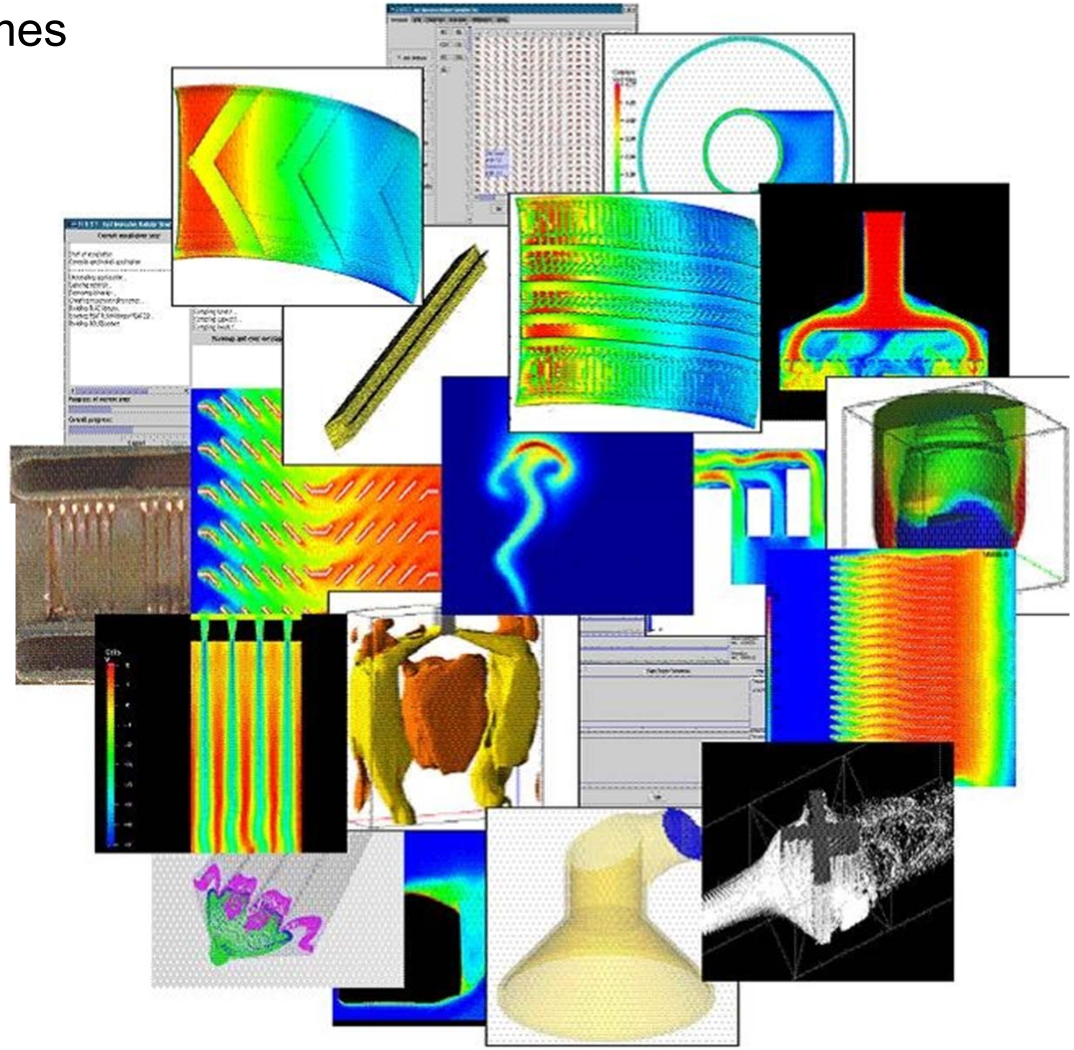




Basic Flow Solver: FeatFlow

Numerical features:

- High order FEM discretization schemes
- FCT & EO stabilization techniques
- Use of unstructured meshes
- Fictitious Boundary (FBM) methods
- Adaptive grid deformation
- Newton-Multigrid solvers



HPC features:

- Massive parallel
- GPU computing
- Open source

Hardware-oriented Numerics

Two phase flow (l-l) with resolved interfaces

The incompressible Navier Stokes equations

$$\rho\left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v}\cdot\nabla\mathbf{v}\right) - \nabla\cdot\left(\mu\left[\nabla\mathbf{v} + (\nabla\mathbf{v})^{\mathrm{T}}\right]\right) + \nabla p = \mathbf{f}_{\mathrm{ST}} + \rho\mathbf{g}$$

$$\nabla\cdot\mathbf{v} = 0$$

Interface tension force

$$\mathbf{f}_{ST} = \sigma\kappa\mathbf{n}, \quad \kappa = -\nabla\cdot\mathbf{n} \quad \text{on} \quad \Gamma$$

unknown interface location

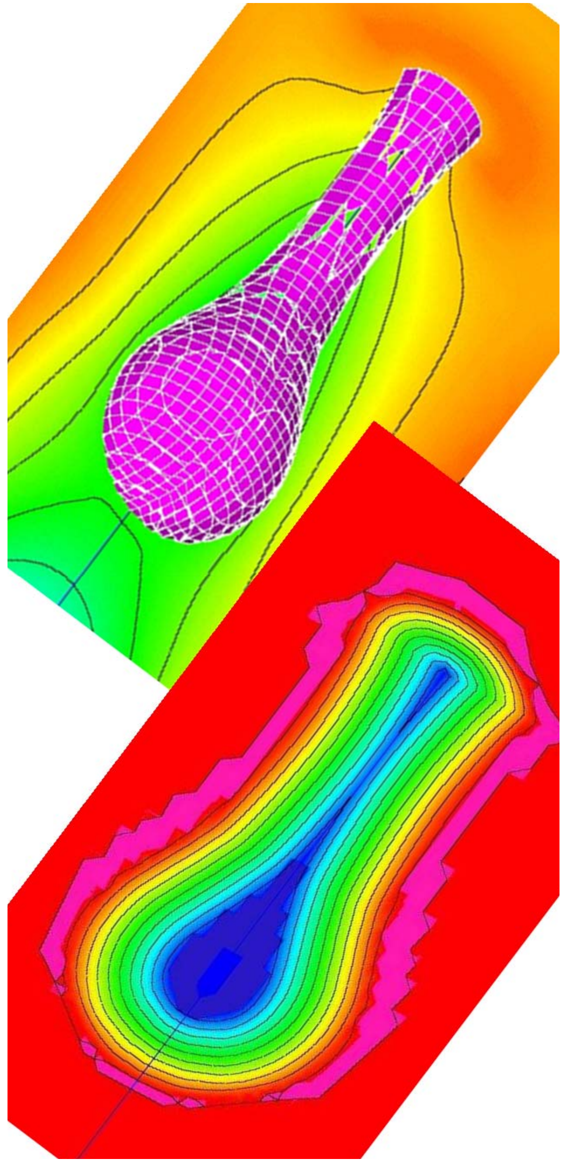
Dependency of physical quantities

$$\mu = \mu(D(\mathbf{v}), \Gamma), \quad \rho = \rho(\Gamma)$$

Interface capturing realized by **Level Set method**

$$\frac{\partial \phi}{\partial t} + \mathbf{v}\cdot\nabla\phi = 0$$

- Exact representation of the interface
- Natural treatment of topological changes
- Provides derived geometrical quantities (**n,** κ)



Two phase flow (l-l) with resolved interfaces

Problems and Challenges

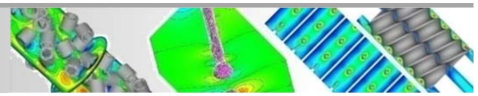
- **Steep gradients** of the velocity field and of other physical quantities near the interface (oscillations!)

- **Reinitialization** w.r.t. distance field (artificial movement of the interface → mass loss, how often to perform?)

- **Mass conservation** (during advection and reinitialization of the Level Set function)

- Representation of **surface tension**: CSF, Line Integral, Laplace-Beltrami, Phasefield, etc.

- **Explicit** or **implicit** treatment (→ *Capillary Time Step* restriction?)



Two phase flow (l-l) with resolved interfaces

<u>Steep changes of physical quantities:</u>

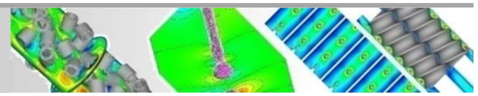
1) Elementwise averaging of the physical properties (prevents oscillations):

   $$\rho_e = x\rho_1 + (1-x)\rho_2, \quad \mu_e = x\mu_1 + (1-x)\mu_2$$

   x is the volume fraction

2) Flow part: Extension of nonlinear stabilization schemes (FCT, TVD, EO-FEM) for the momentum equation for LBB stable element pairs with discontinuous pressure (Q2/P1)

3) Interface tracking part with DG(1)-FEM: Flux limiters satisfying LED requirements



Two phase flow (l-l) with resolved interfaces

Reinitialization

- Mainly required in the vicinity of the interface
- How often to perform?
- Which realization to implement?
- Efficient parallelization (3D!)

Alternatives

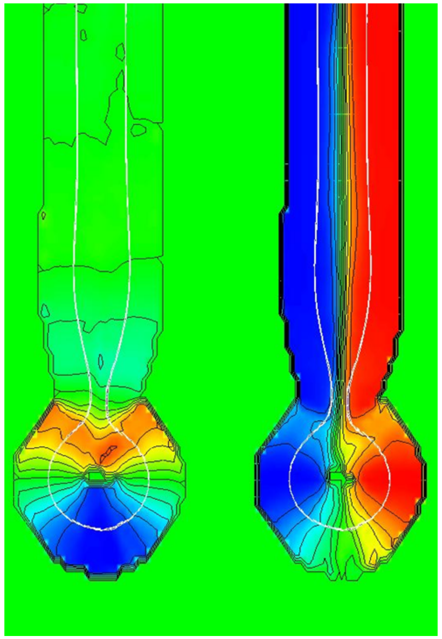
- Brute force (introducing new points at the zero level set)
- Fast sweeping („advancing front“ upwind type formulas)
- Fast marching
- Algebraic Newton method
- **Hyperbolic PDE approach**
- many more…..

Maintaining the signed distance function by PDE reinitialization

$$\frac{\partial \phi}{\partial \tau} + \mathbf{u} \cdot \nabla \phi = S(\phi) \qquad \mathbf{u} = S(\phi)\frac{\nabla \phi}{|\nabla \phi|} \quad \Leftrightarrow \quad |\nabla \phi| = 1$$

Problems:

- What to do with the sign function at the interface? (smoothing?)
- How to handle the underlying non-linearity?
- How often to perform? (expensive → steady state)



Globally defined normal vectors

Two phase flow (l-l) with resolved interfaces

Fine-tuned reinitialization

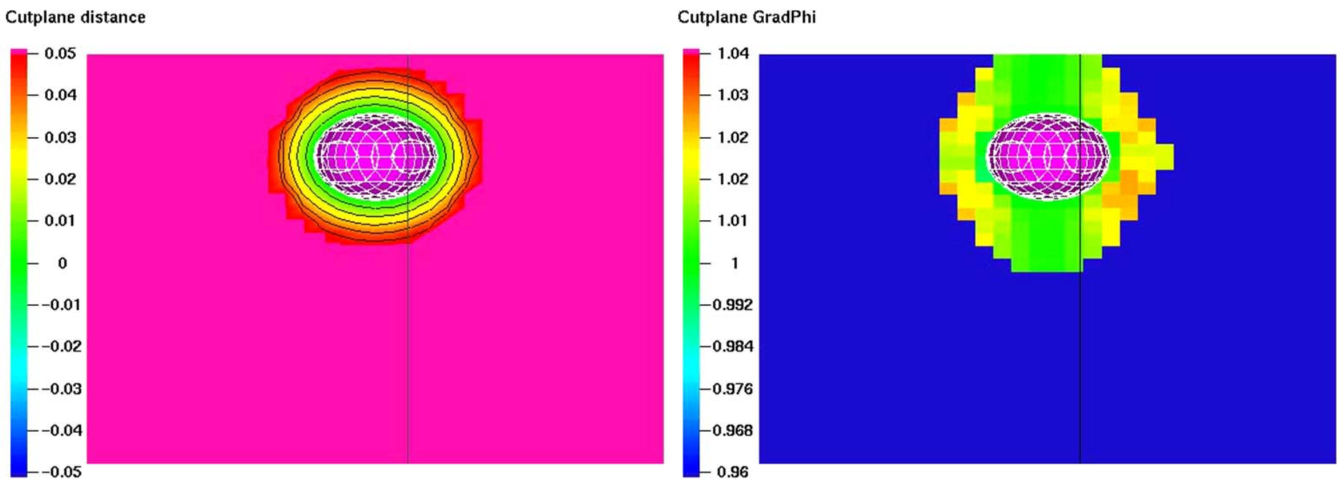
Our reinitialization is performed in combination of 2 ingredients:

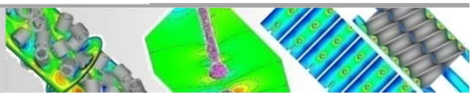
1) Elements intersected by the interface are modified w.r.t. the slope of the distance distribution („Parolini trick") such that $|\nabla \phi| = 1$

2) Far field reinitialization: realization is based on the PDE approach („FBM"), but it does not require smoothening of the distance function!

In addition: continuous projection of the interface (smoothening of the discontinuous P_1 based distance function)

$$\phi_{P_1} \xrightarrow{L_2 \text{ projection}} \phi_{Q_1} \xrightarrow{L_2 \text{ projection}} \phi_{P_1}$$





Two phase flow (l-l) with resolved interfaces

Continuum

Surface

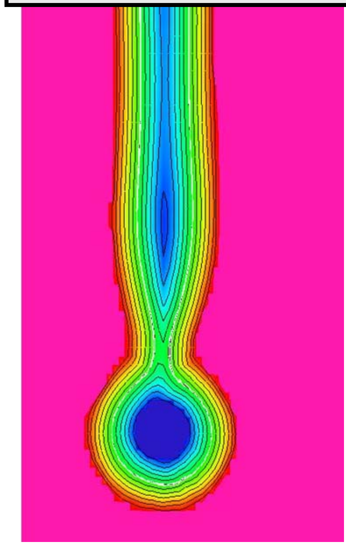
Force

- Transformation of the surface integrals to volume integrals with the help of a <u>regularized</u> Dirac delta function δ
- Requires globally defined normals and curvature
- Reduction of spurious oscillations

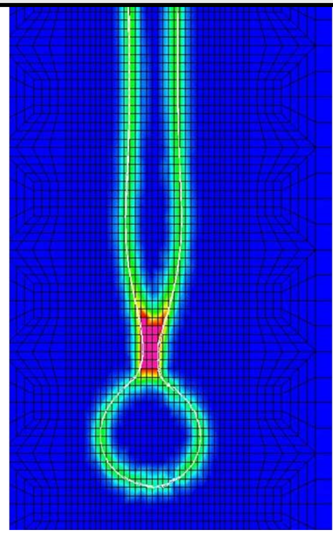
$$\mathbf{f}_{\mathrm{ST}} = \sigma\kappa\mathbf{n}\delta(x,\varepsilon)$$

$$\mathbf{n}_{P_1} \xrightarrow{\mathrm{L}_2 \text{ projection}} \mathbf{n}_{Q_1}$$ continuous normal field

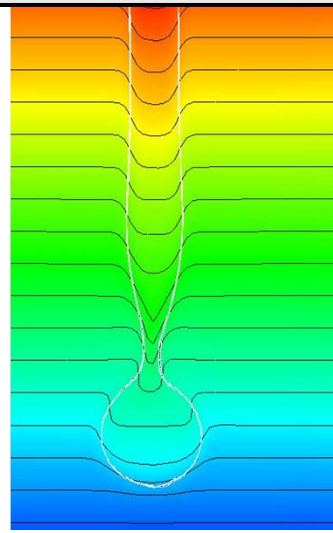
$$\kappa_{Q_1} = \frac{\int_\Omega \nabla \cdot \mathbf{n}_{Q_1} \, d\mathbf{x}}{\int_\Omega d\mathbf{x}}$$ continuous curvature field



Level Set distribution



Distribution of the smoothed surface tension force $(\sigma\kappa\delta)_{Q_1}$



Resulting pressure distribution

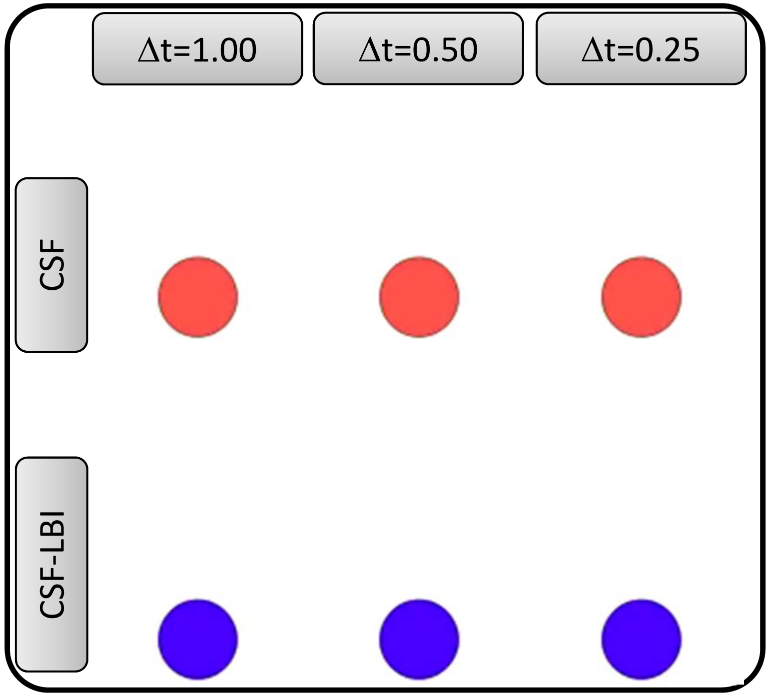
Two phase flow (l-l) with resolved interfaces

<u>Surface Tension</u>: Semi-implicit CSF formulation based on Laplace-Beltrami

$$\mathbf{f}_{\mathrm{ST}} = \int_\Omega \sigma\kappa\hat{\mathbf{n}}\cdot\mathbf{v}\,\delta(\Gamma,\mathbf{x})\,d\mathbf{x} \qquad = \int_\Omega \sigma\left(\underline{\Delta}\mathbf{x}\big|_\Gamma\right)\cdot\left(\mathbf{v}\,\delta(\Gamma,\mathbf{x})\right)d\mathbf{x}$$

$$= -\int_\Omega \sigma\underline{\nabla}\mathbf{x}\big|_\Gamma\cdot\underline{\nabla}\left(\mathbf{v}\,\delta(\Gamma,\mathbf{x})\right)d\mathbf{x} = -\int_\Omega \sigma\underline{\nabla}\mathbf{x}\big|_\Gamma\cdot\underline{\nabla}\mathbf{v}\,\delta(\Gamma,\mathbf{x})\,d\mathbf{x}$$

Application of the semi-implicit time integration yields $\mathbf{x}\big|_{\Gamma^{n+1}} = \mathbf{x}\big|_{\Gamma^n} + \Delta t\,\mathbf{u}^{n+1}$



$$\Downarrow$$

$$\mathbf{f}_{\mathrm{ST}} = -\int_\Omega \sigma\,\delta_\varepsilon\left(dist(\Gamma^n,\mathbf{x})\right)\underline{\nabla}\tilde{\mathbf{x}}\big|_\Gamma^n\cdot\underline{\nabla}\mathbf{v}\,d\mathbf{x}$$

$$-\Delta t^{n+1}\int_\Omega \sigma\,\delta_\varepsilon\left(dist(\Gamma^n,\mathbf{x})\right)\underline{\nabla}\mathbf{u}^{n+1}\cdot\underline{\nabla}\mathbf{v}\,d\mathbf{x}$$

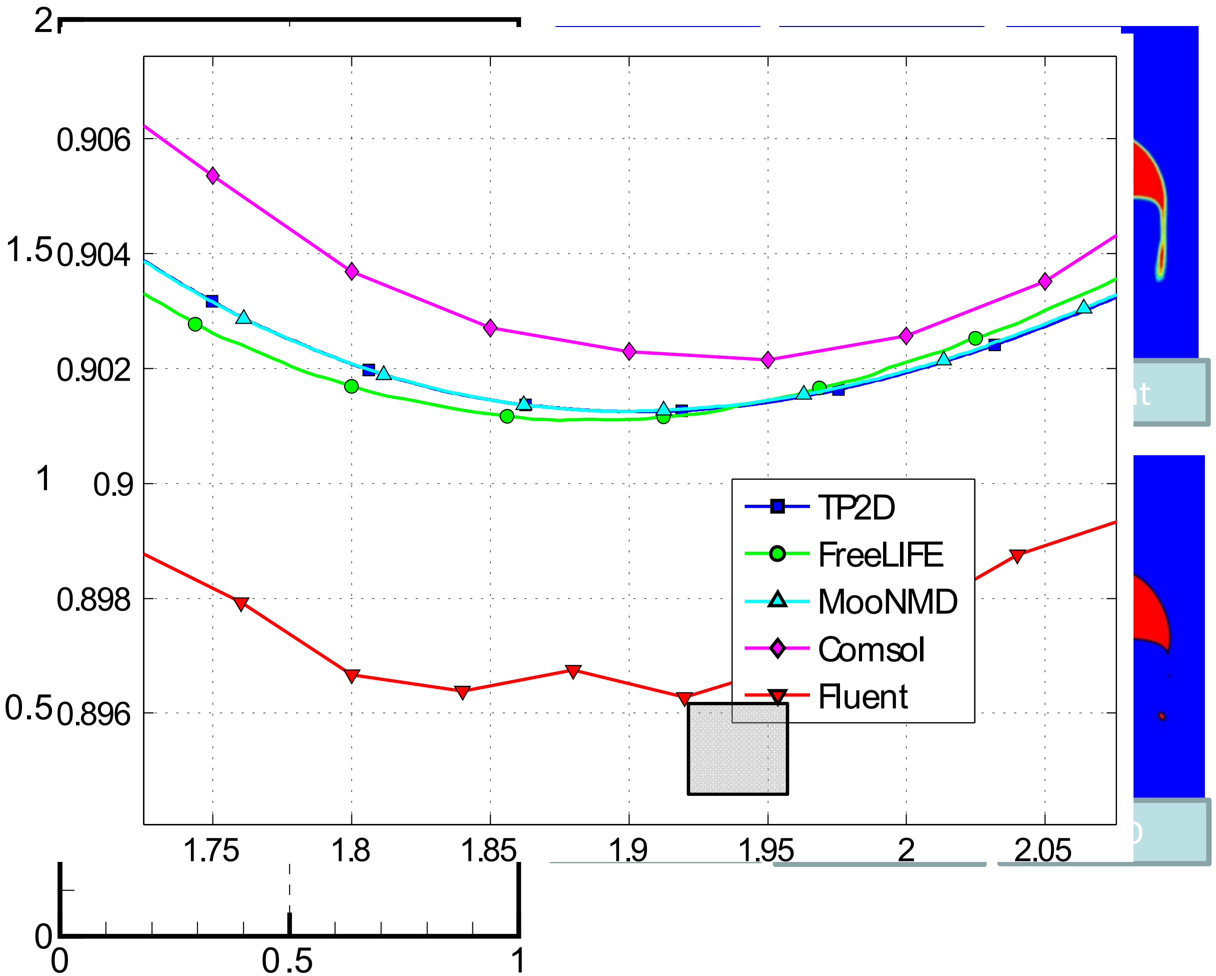
Advantages

- Relaxes *Capillary Time Step* restriction
- „Optimal“ for FEM-Level Set approach due to global information

Benchmarking

2D Bubble Benchmarks

http://www.featflow.de/beta/en/benchmarks/



Benchmark quantities

Center of mass $\mathbf{x}_c = \dfrac{\int_{\Omega_2} \mathbf{x}\, dx}{\int_{\Omega_2} 1\, dx}$

Mean rise velocity $\mathbf{U}_c = \dfrac{\int_{\Omega_2} \mathbf{u}\, dx}{\int_{\Omega_2} 1\, dx}$

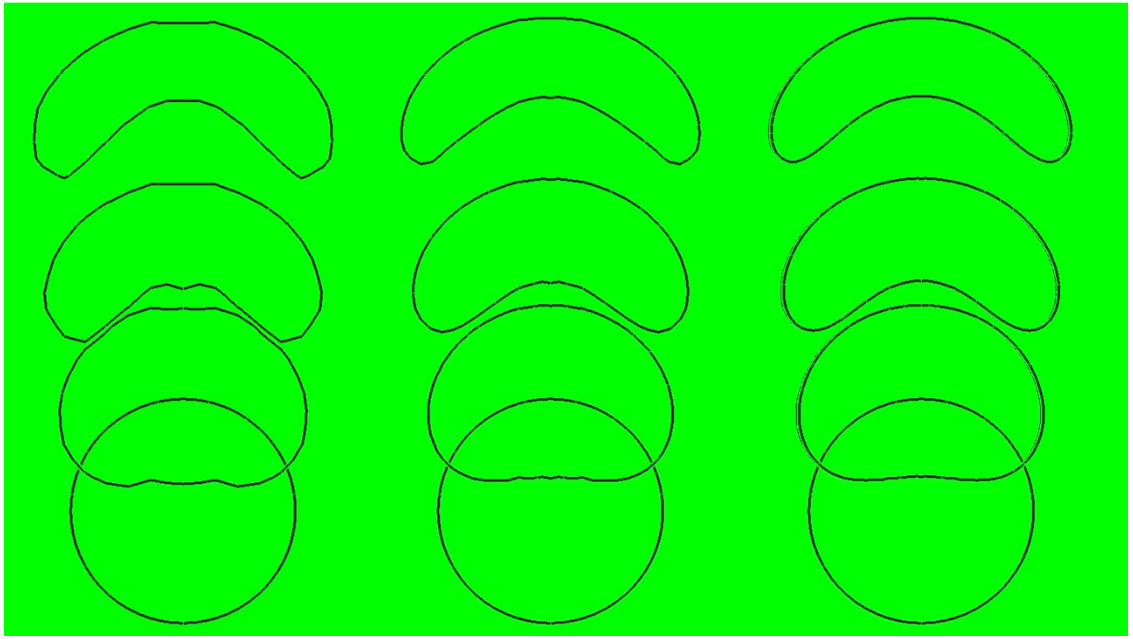
Circularity $¢ = \dfrac{P_a}{P_b} = \dfrac{\pi d_a}{P_b}$

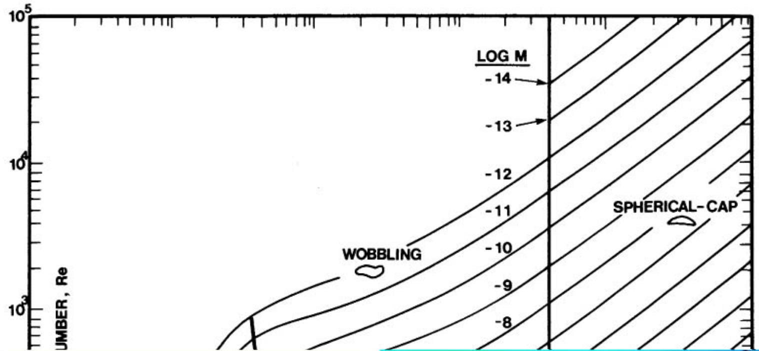
Hysing, S.; Turek, S.; Kuzmin, D.; Parolini, N.; Burman, E.; Ganesan, S.; Tobiska, L.: Quantitative benchmark computations of two-dimensional bubble dynamics, International Journal for Numerical Methods in Fluids, 2009

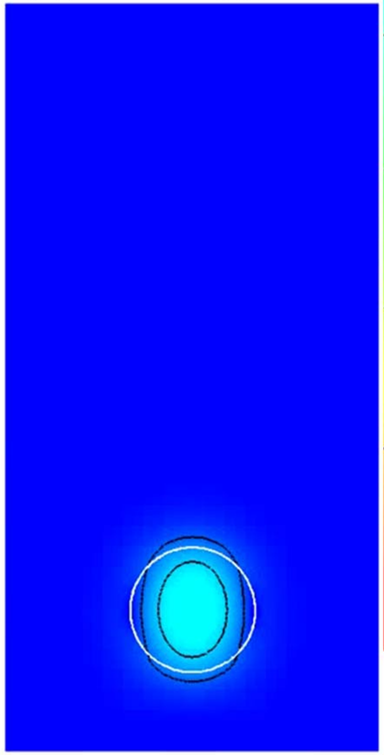
3D convergence analysis for large density jumps

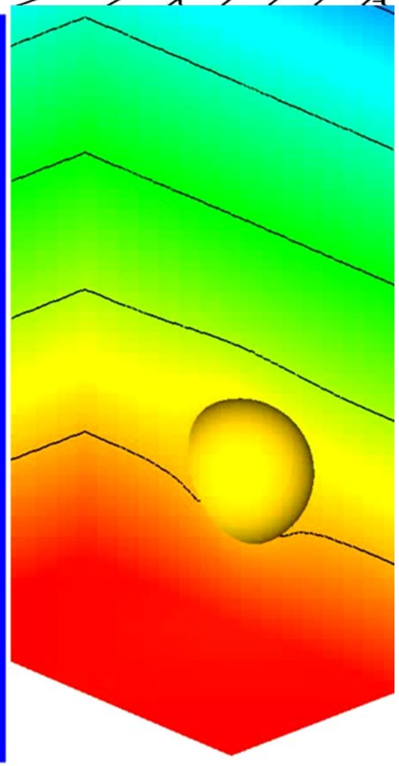
Rising bubble problem for Eo = 60, Re = 34
Density jump 1:100

Level 2 Level 3 Level 4









Benchmarking with experimental results

Continuous phase:

Glucose-Water mixture

$\mu_D = 500\,mPa\,s$

$\rho_D = 972\,kg\,m^{-3}$

$\dot{V}_D = 3{,}64\,ml\,min^{-1}$

$\sigma_{CD} = 0{,}034\,N\,m^{-1}$

Silicon oil

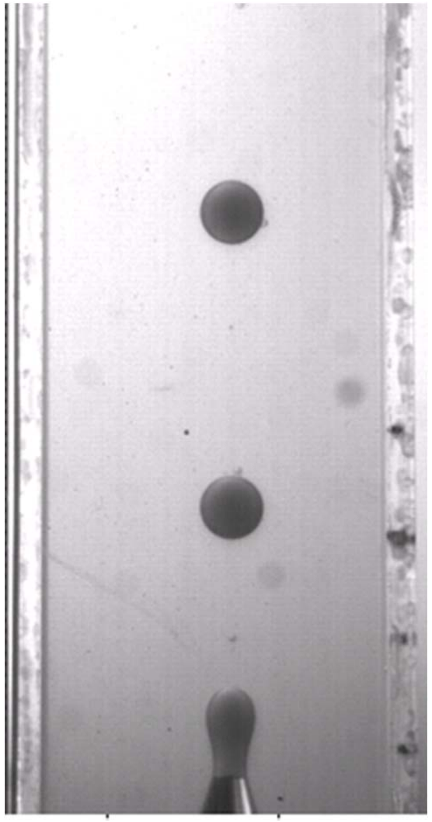
$\mu_C = 500\,mPa\,s$

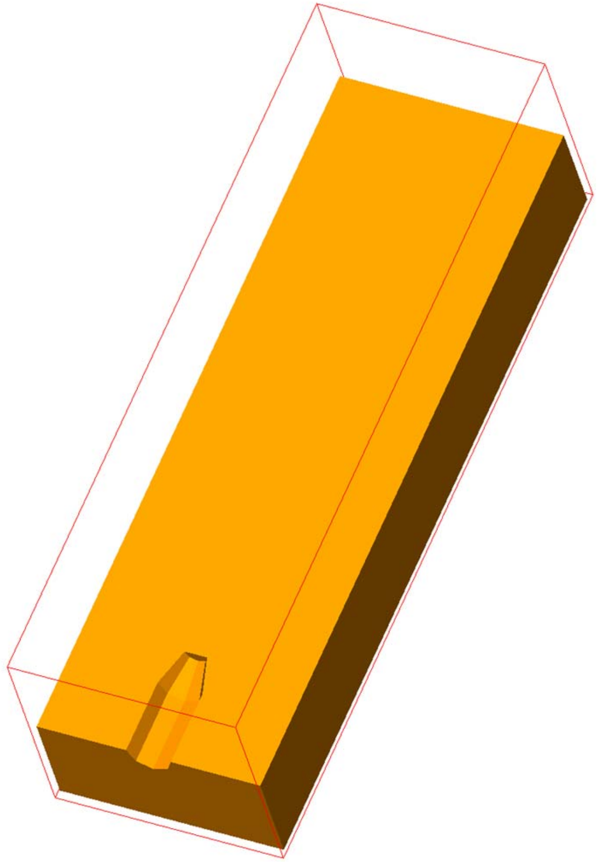
$\rho_C = 1340\,kg\,m^{-3}$

$\dot{V}_C = 99{,}04\,ml\,min^{-1}$

Dispersed phase:

Experimental setup with **AG Walzel** **(BCI/Dortmund)**

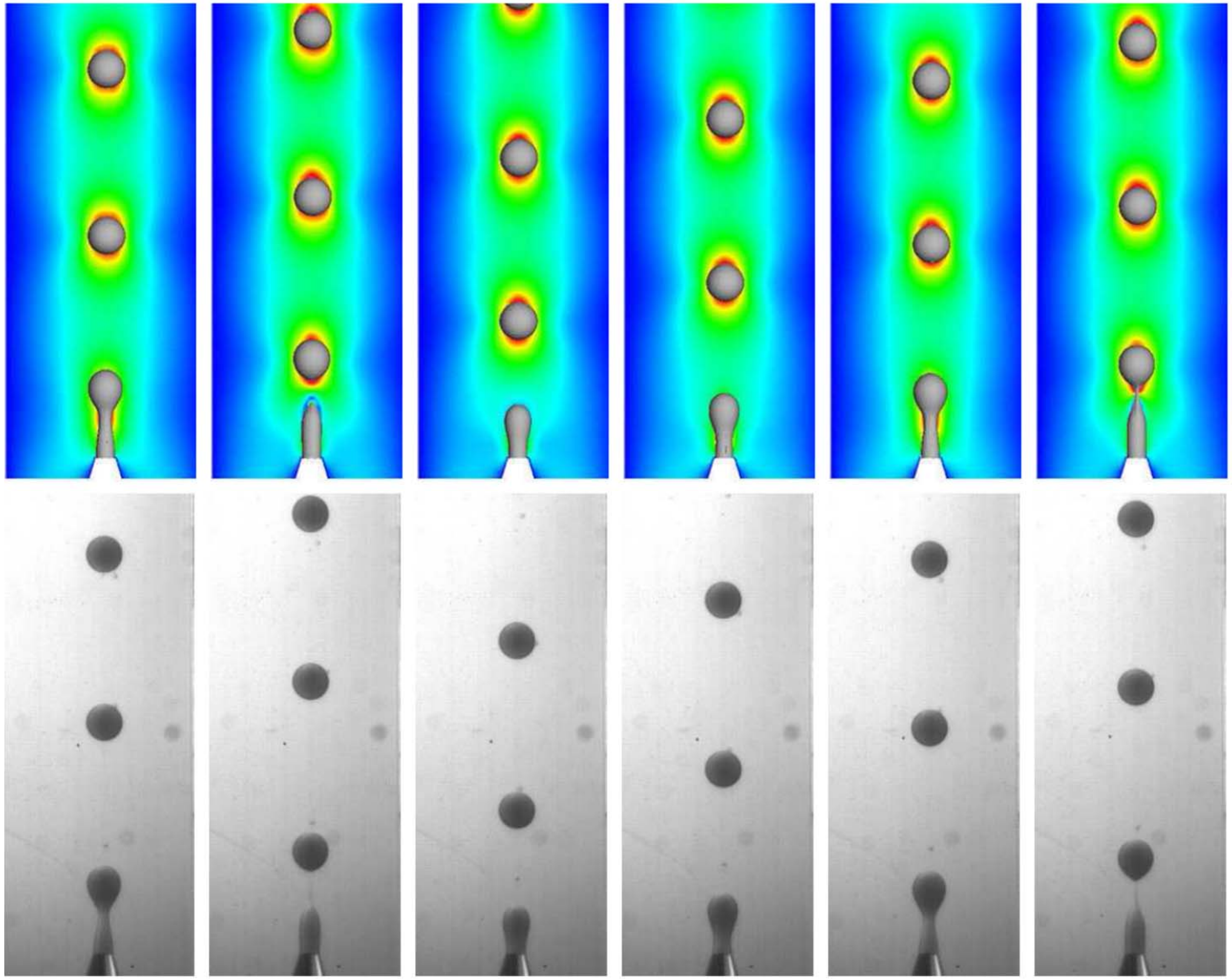


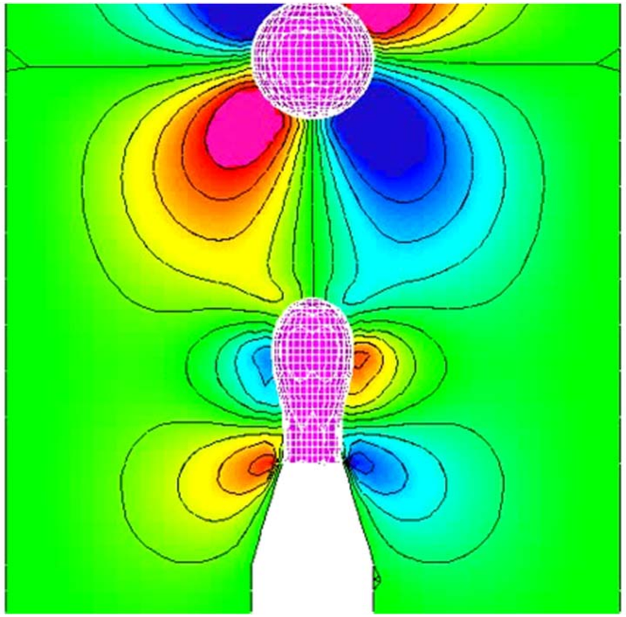


Validation parameters:

- frequency of droplet generation
- droplet size
- stream length

Benchmarking with experimental results





	Separation frequency [Hz]	Droplet size [dm]	Stream Length [dm]
Exp	0,58	0,062	0,122
Sim	0,6	0,058	0,102

Tailored monodisperse droplets via modulation

In case of monodisperse droplet generation:

$$\dot{V}_D = f V_{\text{droplet}}$$

Influencable variables

On the level of the process:

- Flowrates
- Modulation frequency
- Modulation amplitude

Geometrical changes:

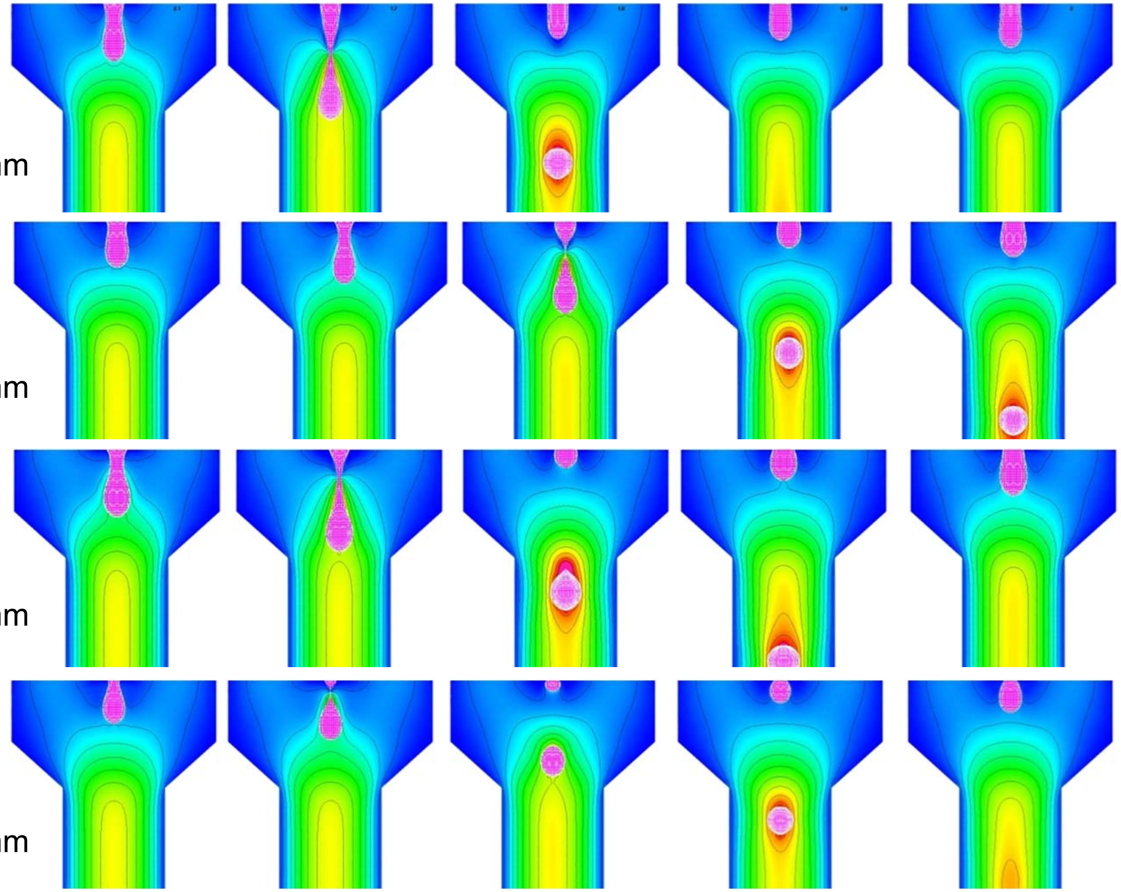
- Capillary size
- Contraction angle
- Contraction ratio

No Regulation
Flowrate: 100%
Capillary: STD
Droplet size: 5.2mm

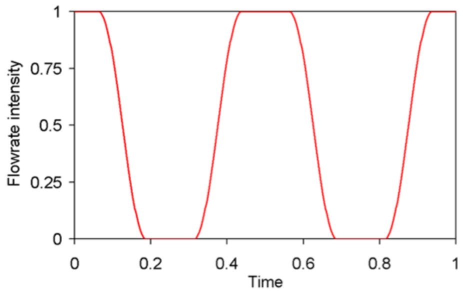
Regulated
Flowrate: 100%
Capillary: STD
Droplet size: 5.0mm

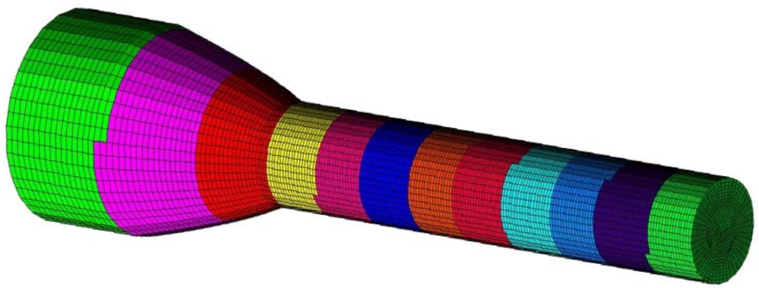
Regulated
Flowrate: 150%
Capillary: STD
Droplet size: 5.7mm

Regulated
Flowrate: 75%
Capillary: STD
Droplet size: 4.5mm



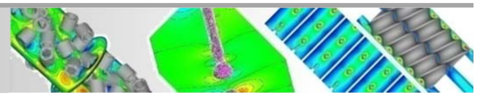
Modulation function





Resulting operation envelope:

- Size: 4.5 mm – 5.7 mm
- Volume: 0.38 cm^3 – 0.77 cm^3



Current Status of (Multiphase) Simulation Tools

- **Numerical efficiency?**
  → OK

- **Parallel efficiency?**
  → OK (tested up to appr. 1000 CPUs)
  → More than 10.000 CPUs???

- **Single processor efficiency?**
  → OK (for CPU)

- **'Peak' efficiency?**
  → NO

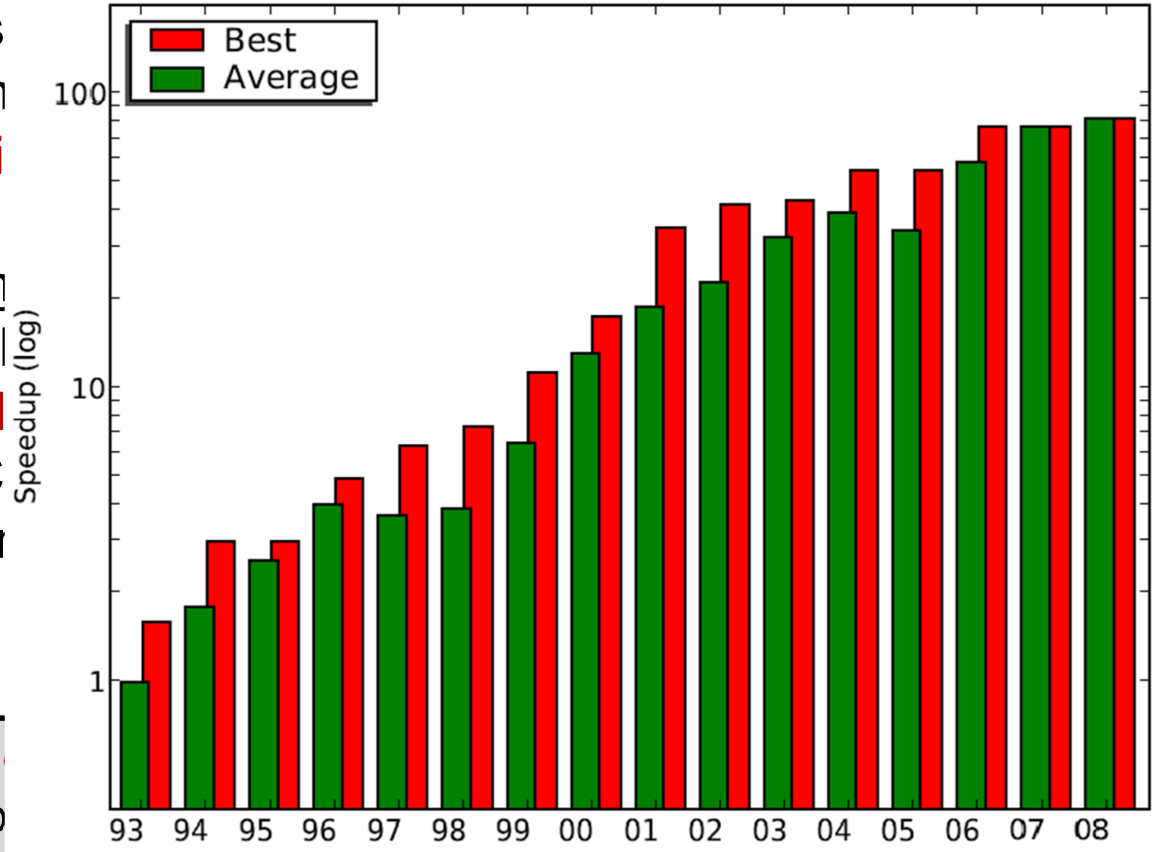
Next: Special HPC Techniques

The 'free ride' is over, paradigm shift in HPC:

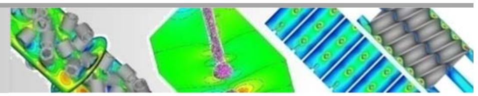
- phys ... ltage)
- mem ... oblems)
- appli ... er hardware

Heterogen ... rs

- grapl
- ARM
- HPC
- recoı



Finite Elem...
solvers: mo...
simulation tools for PDEs nowadays.



Extensive Tests show.....

- It is (almost) impossible to come close to **Single Processor Peak** Performance with modern (= high numerical efficiency) simulation tools

- **Parallel Peak Performance** with modern Numerics even harder, already for moderate processor numbers

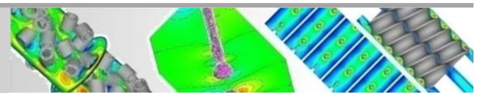
Hardware-oriented Numerics (HwoN)

+

UnConventional Hardware (UCHPC)

=

FEAST Project



Unconventional Hardware

- CELL multicore processor (PS3),
  7 synergistic processing units @ 3.2 GHz,
  Memory @ 3.2 GHz
  ≈ 218 GFLOP/s

- GPU (NVIDIA GTX 285):
  240 cores @ 1.476 GHz,
  1.242 GHz memory bus (160 GB/s)
  ≈ 1.06 TFLOP/s

UnConventional High Performance Computing (UCHPC)

Design Goals

Include GPUs into FEAST

- without
  - changes to application codes FEA(S)TFLOW
  - fundamental re-design of FEAST
  - sacrificing either functionality or accuracy

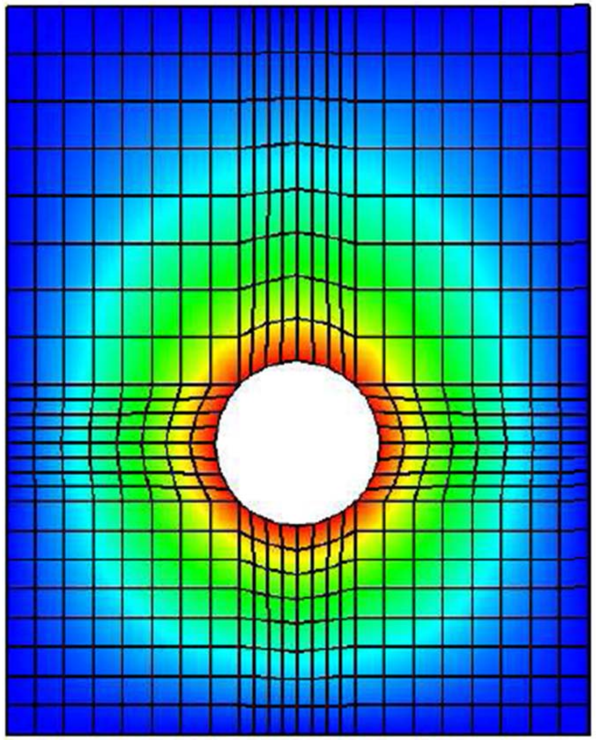
- but with
  - noteworthy speedups
  - a reasonable amount of generality w.r.t. other co-processors
  - and additional benefits in terms of space/power/etc.

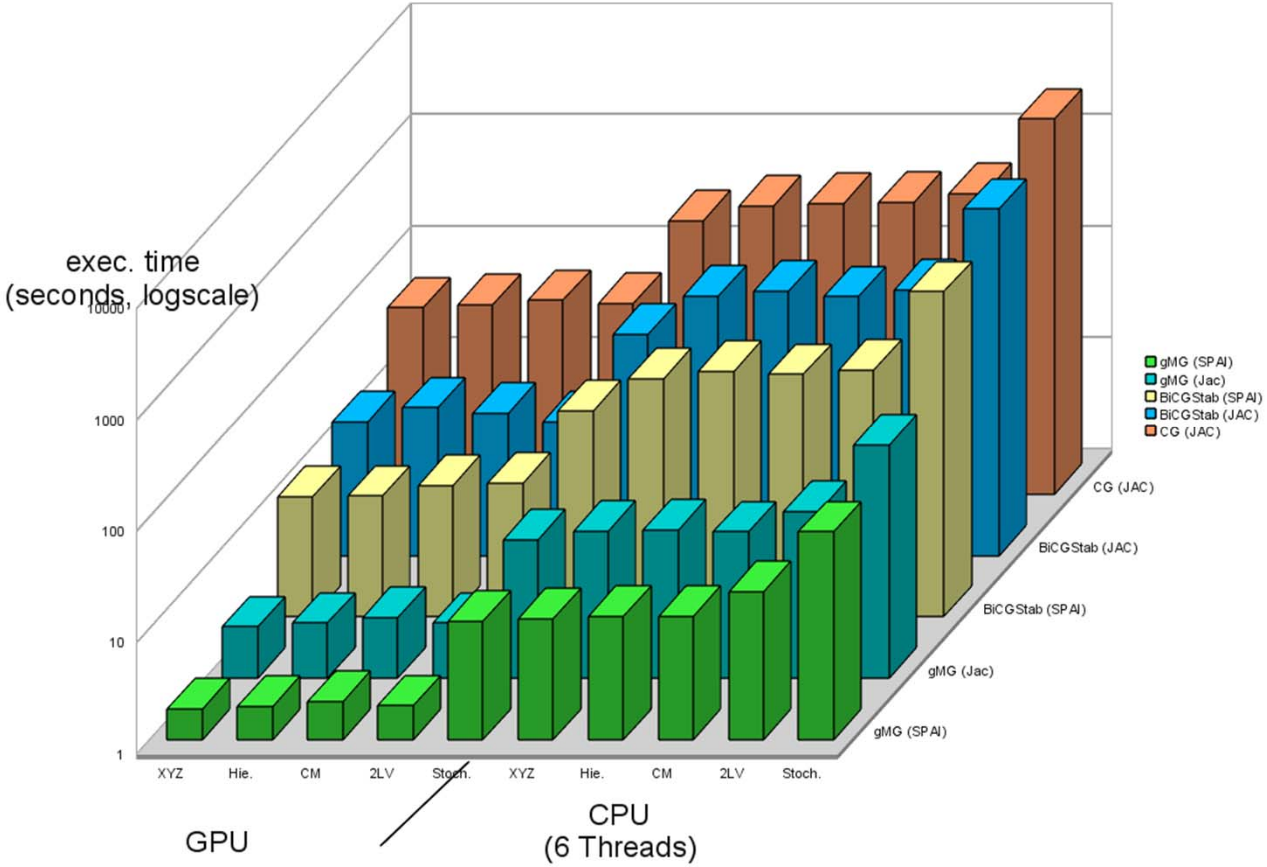
But: no --march=gpu/cell compiler switch

Poisson Solver Tests

$$-\Delta u = 0 \quad \text{in } \Omega,$$
$$u = 0 \quad \text{on } \Gamma_1$$
$$u = 1 \quad \text{on } \Gamma_2$$

	Q_1		Q_2	
L	N	non-zeros	N	non-zeros
4	576	4552	2176	32192
5	2176	18208	8448	128768
6	8448	72832	33280	515072
7	33280	291328	132096	2078720
8	132096	1172480	526336	8351744
9	526336	4704256	2101248	33480704
10	2101248	18845696	-	-





Poisson Solver Tests

Identical solution, but differences of more than a

factor 1000x

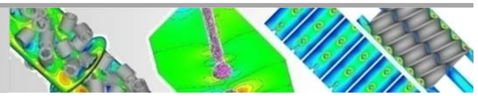
regarding the CPU time for one „simple“ (small) subproblem

after „optimization“ on all levels!

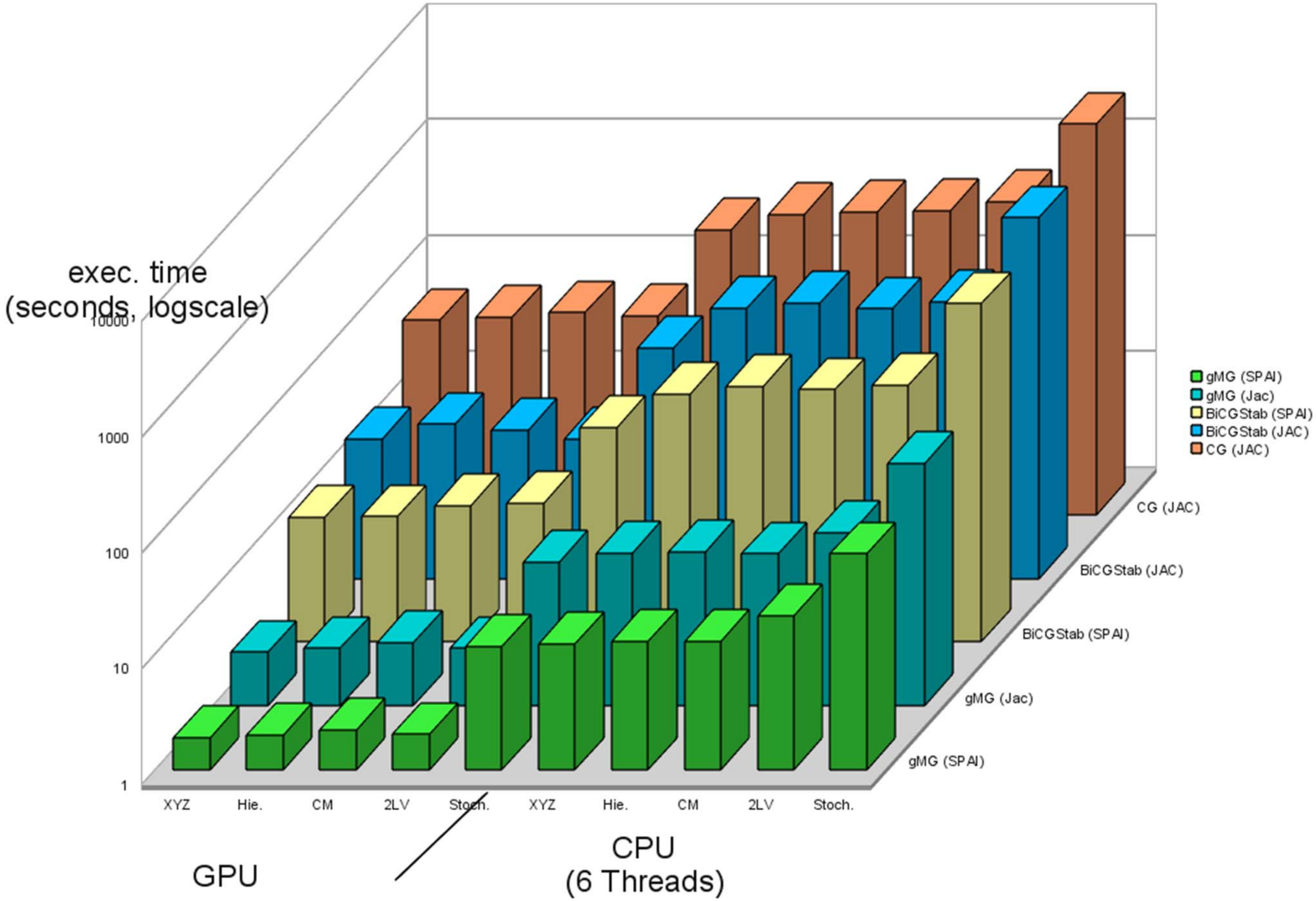
Huge Potential for the Future …

- **Numerical Simulation & High Performance Computing** have to consider recent and future **hardware trends**, particularly for heterogeneous multicore architectures and massively parallel systems!
- More research in the combination of **‘Hardware-oriented Numerics’** and **‘Unconventional Hardware’** is necessary!
- (Still) much more powerful CFD tools are possible if **modern Numerics meets modern Hardware**!

…or most of existing (academic/commercial) CFD software will be ‘worthless’ in a few years!



Poisson Solver Tests



Solver Benchmark (unstructured mesh)

$$
\begin{aligned}
-\Delta u &= 0 \quad \text{in } \Omega,\\
u &= 0 \quad \text{on } \Gamma_1\\
u &= 1 \quad \text{on } \Gamma_2
\end{aligned}
$$

L	Q_1 N	Q_1 non-zeros	Q_2 N	Q_2 non-zeros
4	576	4552	2176	32192
5	2176	18208	8448	128768
6	8448	72832	33280	515072
7	33280	291328	132096	2078720
8	132096	1172480	526336	8351744
9	526336	4704256	2101248	33480704
10	2101248	18845696	-	-