A Parallel Implicit Implementation of Smoothed Particle Hydrodynamics for Electrokinetic Applications

SAND2014-20488C

K. Kim, M. Maxey, W. Pan, M. Parks, M. Perego,

N. Trask, C. Siefert

MRS, 2014 Fall meeting & exhibit
December 1st, Boston (MA), 2014

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Smoothed particle hydrodynamics (SPH)

Navier Stokes equations in Lagrangian form

$$\begin{cases} \dfrac{d\mathbf{u}}{dt} = -\frac{1}{\rho}\nabla p + \nu\nabla^2\mathbf{u}, \\ \nabla \cdot \mathbf{u} = 0, \\ \dfrac{d\mathbf{x}}{dt} = \mathbf{u}, \end{cases}$$

SPH Pros/Cons of Standard explicit SPH scheme

PROS: - no need to track boundaries
- amenable to parallelization
- good performance in high Reynold©number regimes

CONS: - accuracy (does not converge when increasing number of particles)
- performs poorly in low Reynold©number regimes
- particles need to be ªwellº distributed

In mesoscale applications, flows typically feature low Reynold©numbers

Standard explicit implementations would require tiny time steps, therefore we consider an **implicit** implementation.

Navier Stokes time discretization:

Implicit 2nd order incremental pressure correction

(improve performance for low Reynold©number flows)

Helmoltz system:
find velocity estimate

$$\begin{cases} \dfrac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} = -\dfrac{1}{\rho}\nabla p^n + \dfrac{\nu}{2}\nabla^2\left(\mathbf{u}^* + \mathbf{u}^n\right) & x \in \Omega, \\ \mathbf{u}^* = 0 & x \in \partial\Omega, \end{cases}$$

Poisson system:
find pressure

$$\begin{cases} \dfrac{1}{\rho}\nabla^2 p^{n+1} = \dfrac{\nabla \cdot \mathbf{u}^*}{\Delta t} & x \in \Omega, \\ \partial_n p^{n+1} = 0 & x \in \partial\Omega. \end{cases}$$

Correct velocity
to make it
divergence free

$$\begin{cases} \dfrac{\mathbf{u}^{n+1} - \mathbf{u}^*}{\Delta t} = -\dfrac{1}{\rho}\nabla\left(p^{n+1} - p^n\right) & x \in \Omega, \\ \nabla \cdot \mathbf{u}^{n+1} = 0 & \\ \mathbf{u}^{n+1} \cdot \hat{n} = 0 & x \in \partial\Omega. \end{cases}$$

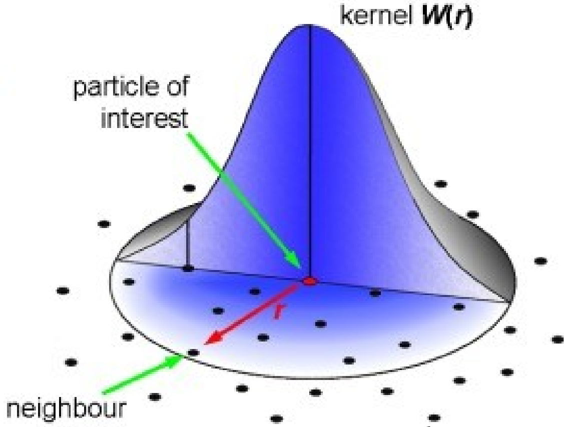
S. J. Cummins, M. Rudman, Journal of Computational Physics, 1999
S. M. Hosseini, J. J. Feng, 2011

Sandia National Laboratories

SPH discretization

Convolution with smooth kernel

$$\langle f \rangle(\mathbf{x}) = \int_{supp(W)} f(\mathbf{y}) W(\mathbf{x}-\mathbf{y}) d\mathbf{y}$$



www.nuigalway.ie

Compute derivatives differentiating the kernel

$$\langle \nabla f \rangle(\mathbf{x}) = \int_{supp(W)} f(\mathbf{y}) \nabla_{\mathbf{x}} W(\mathbf{x}-\mathbf{y}) d\mathbf{y}$$

Approximate integral as sum over particles

$$\widetilde{\nabla f}(\mathbf{x}_i) = \sum_{j \in supp(W)} f(\mathbf{x}_j) \nabla_{\mathbf{x}_i} W(\mathbf{x}_i - \mathbf{x}_j) V_j$$

volume fraction

Derive approximations for gradient operator and Lapalacian operator

$$\nabla_0 f_i = \sum_{j \in supp(W)} (f_j - f_i) \nabla_{\mathbf{x}_i} W_{ij} V_j \qquad \nabla_0^2 f_i = 2 \sum_{j \in supp(W)} \frac{f_i - f_j}{|\mathbf{r}_{ij}|} \mathbf{e}_{ij} \cdot \nabla_{\mathbf{x}_i} W_{ij} V_j$$

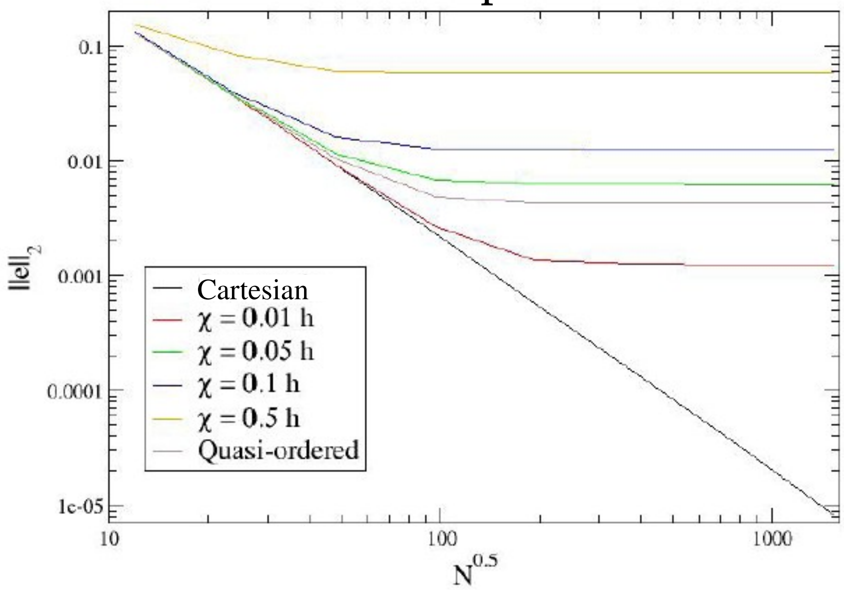
Zeroth order approximation

Sandia National Laboratories

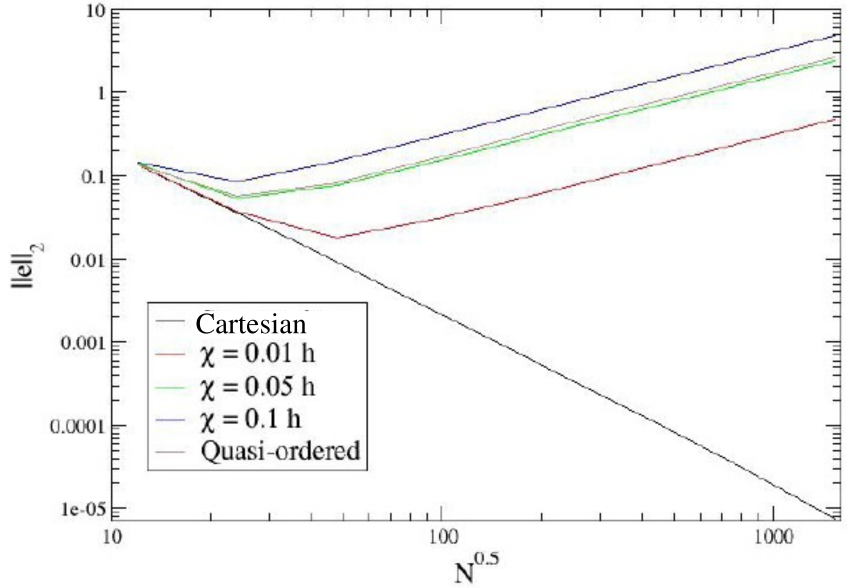
Zeroth order operators (in)accuracy

L2 error, for gradient and Laplacian operators applied on the function $u(x,y) = \sin(x)\sin(y)$, on different particles distributions (Cartesian, perturbed Cartesian, quasi-ordered).

Gradient operator



Lapalcian operator



χ : amount of uniformly distributed noise
h proportional $\sqrt{N}$ and to the kernel support radius

Convergence tests: N. Trask

Higher order SPH discretization

"standard" zeroth-order SPH discretization

$$\nabla_0 f_i = \sum_{j \in supp(W)} (f_j - f_i) \nabla_{\mathbf{x}_i} W_{ij} V_j$$

$$\nabla_0^2 f_i = 2 \sum_{j \in supp(W)} \frac{f_i - f_j}{|\mathbf{r}_{ij}|} \mathbf{e}_{ij} \cdot \nabla_{\mathbf{x}_i} W_{ij} V_j$$

"Corrected" first-order SPH discretization

$$\nabla_1 f_i = \sum_{j \in supp(W)} (f_j - f_i)\, \mathbf{G}_i \nabla_{\mathbf{x}_i} W_{ij} V_j$$

$$\nabla_1^2 f_i = 2 \sum_{j \in supp(W)} \left(\mathbf{L}_i : \mathbf{e}_{ij} \otimes \nabla_{\mathbf{x}_i} W_{ij}\right) \left(\frac{f_i - f_j}{|\mathbf{r}_{ij}|} - \mathbf{e}_{ij} \cdot \nabla_1 f_i\right) V_j$$

Correction tensors
(computed locally)

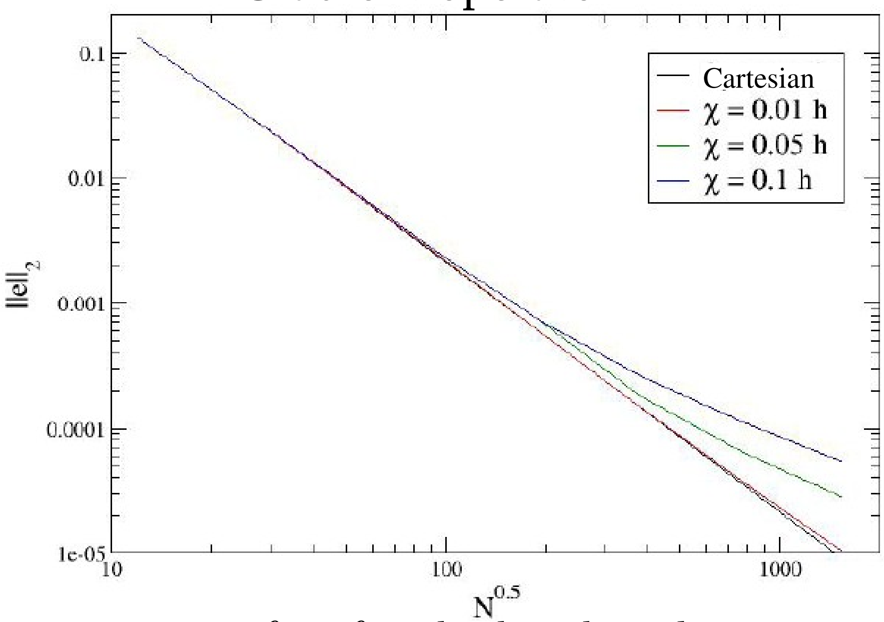
- "Almost" 2nd order accuracy
- Allows for fewer neighbors ➡ performance and scalability improvement

G. Oger et al, Journal of Computational Physics, 2007
R. Fatehi and M. T. Manzari, Computer and Mathematics with Application, 2011

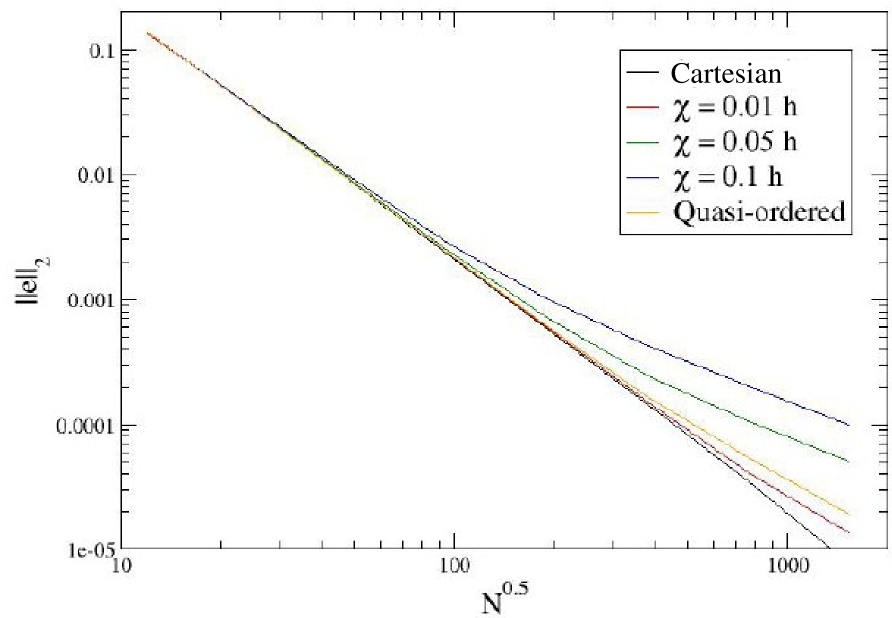
First-order operators accuracy

L2 error, for gradient and Laplacian operators applied on the function $u(x, y) = \sin(x)\sin(y)$ on different particles distributions (Cartesian, perturbed Cartesian, quasi-ordered).

Gradient operator



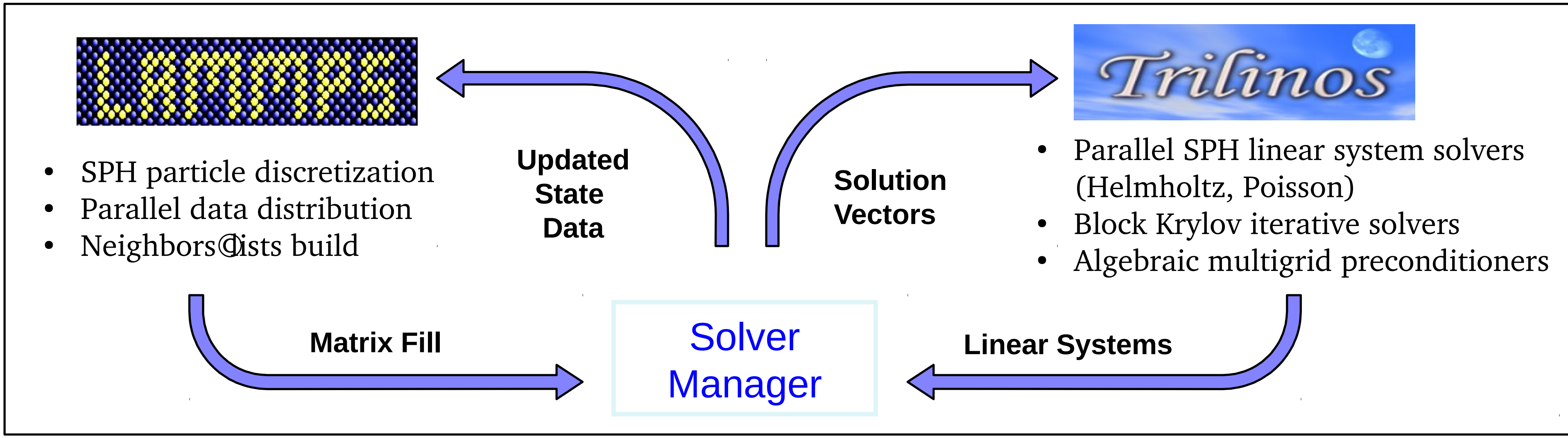
Lapalcian operator



χ : amount of uniformly distributed noise
h proportional $\sqrt{N}$ and to the kernel support radius

Convergence tests: N. Trask

SPH implementation



Fuse LAMMPS and Trilinos for massively parallel 3D SPH computational framework

Let each code handle what it was designed to do well

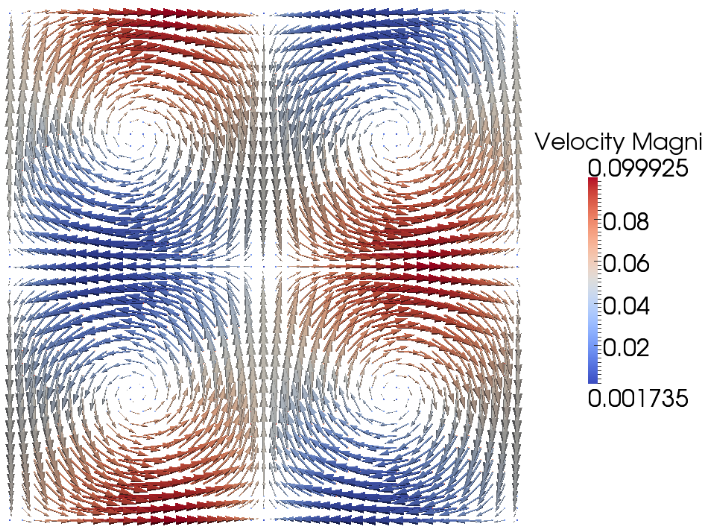
- LAMMPS handles particle data, parallel data distribution, neighboring
- Trilinos handles distributed memory linear/nonlinear solvers, preconditioners, etc.

N. Trask et al, CMAME, submitted, 2014

Scalability results for 3D Taylor-Green Vortex

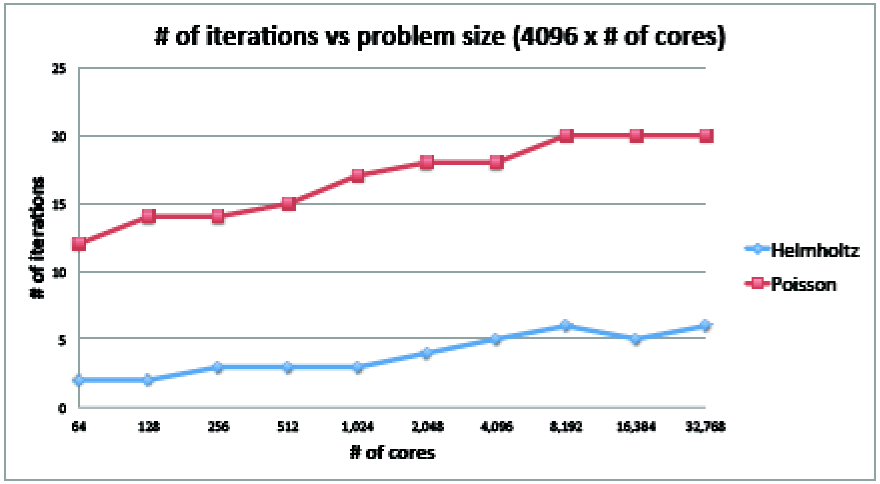
Synthetic problem
manufactured solution:

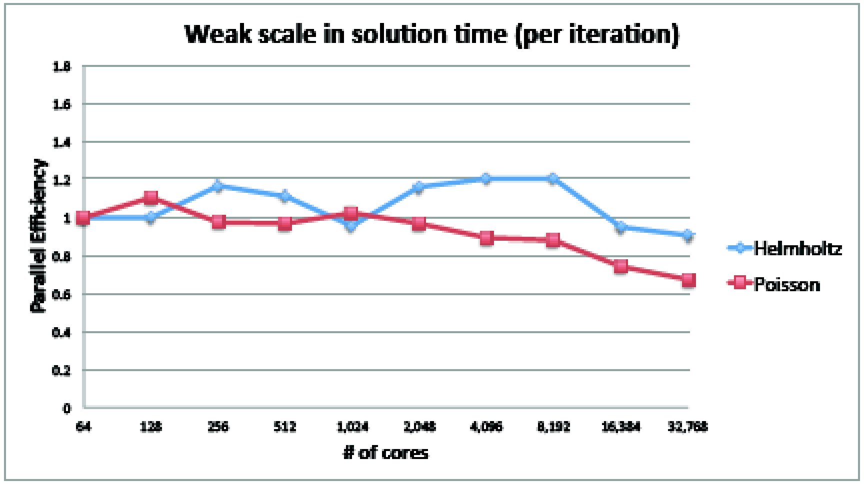
$$u_x = U_0 e^{-2\nu\pi^2 t}\sin(\pi x)\cos(\pi y)$$
$$u_y = -U_0 e^{-2\nu\pi^2 t}\cos(\pi x)\sin(\pi y)$$
$$p = \frac{U_0^2}{4} e^{-4\nu\pi^2 t}(\cos(2\pi x) + \cos(2\pi y) + 2)$$



t = 0.000

Scalability results using up to 134 millions particles and 32K cores





To our knowledge, the largest implicit SPH simulation ever

Sandia National Laboratories

Verification with Finite Volume for BCC lattice

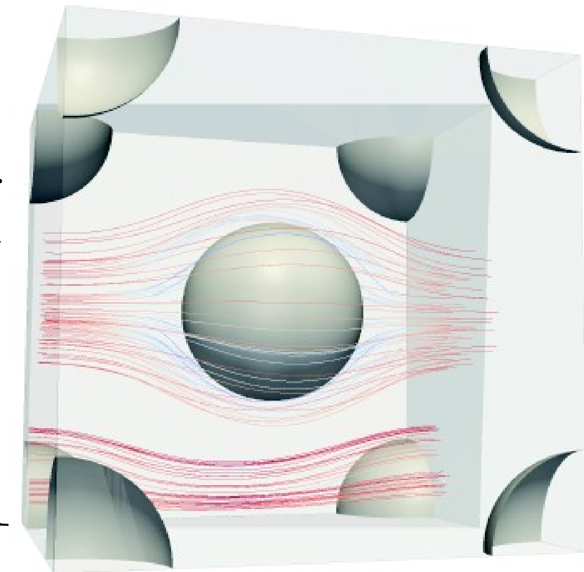
Problem details:

No slip boundary conditions:
Morris/Holmes Dirichlet boundary conditions to provide $^{a}2^{nd^{o}}$ accuracy

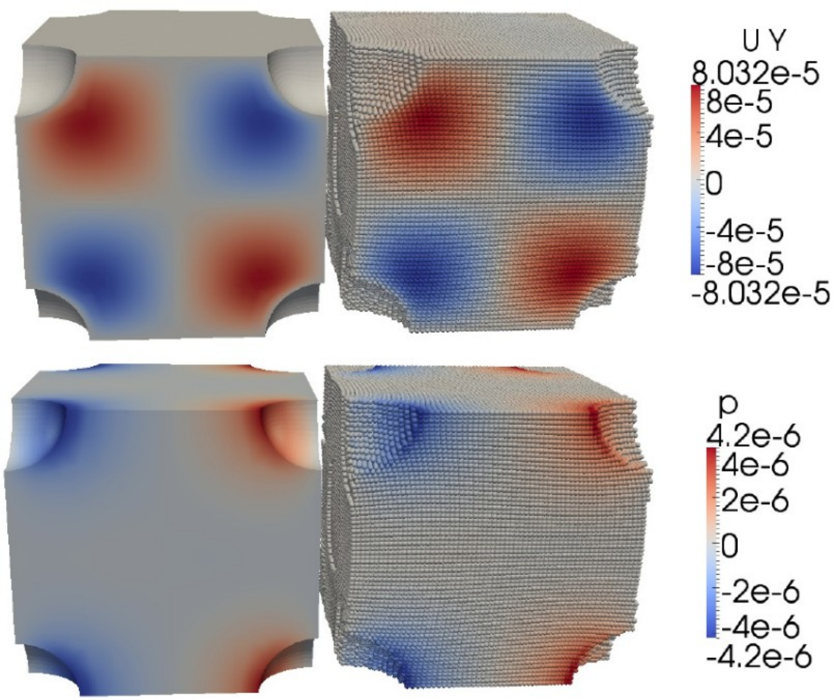
OK for complex geometries
(no need to have analytic representation of boundaries)

Streamlines colored with magnitude of velocity

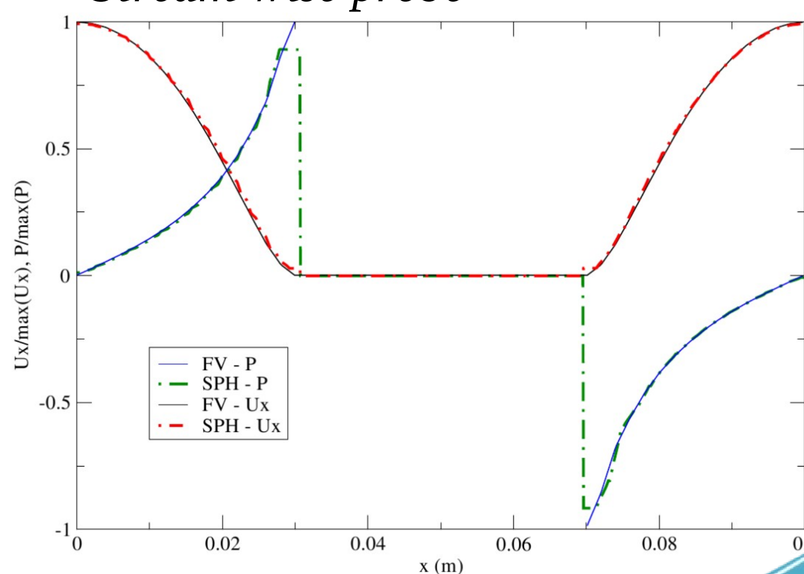
$$Re \simeq 0.1$$



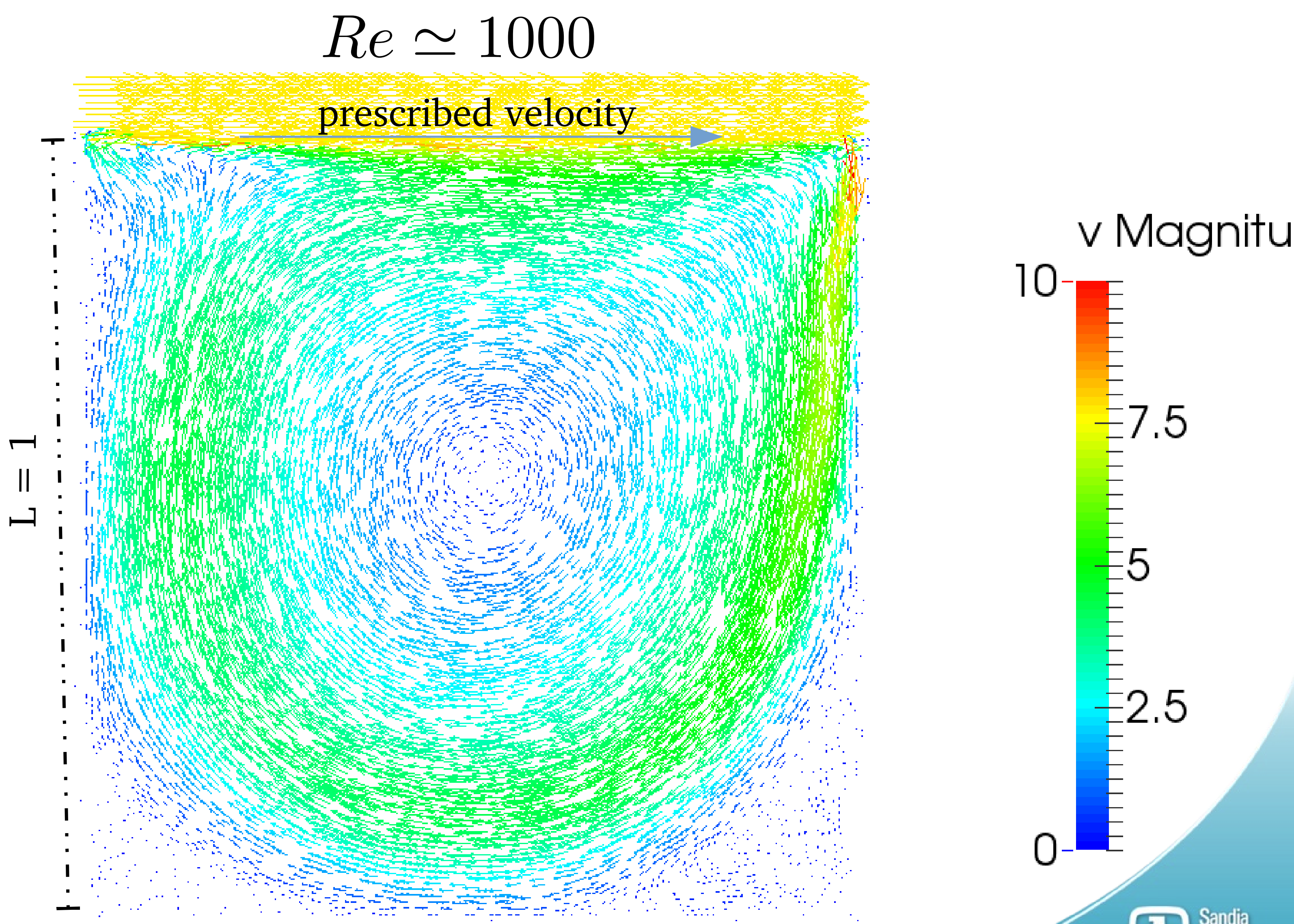
Comparison with finite volumes



Stream-wise probe

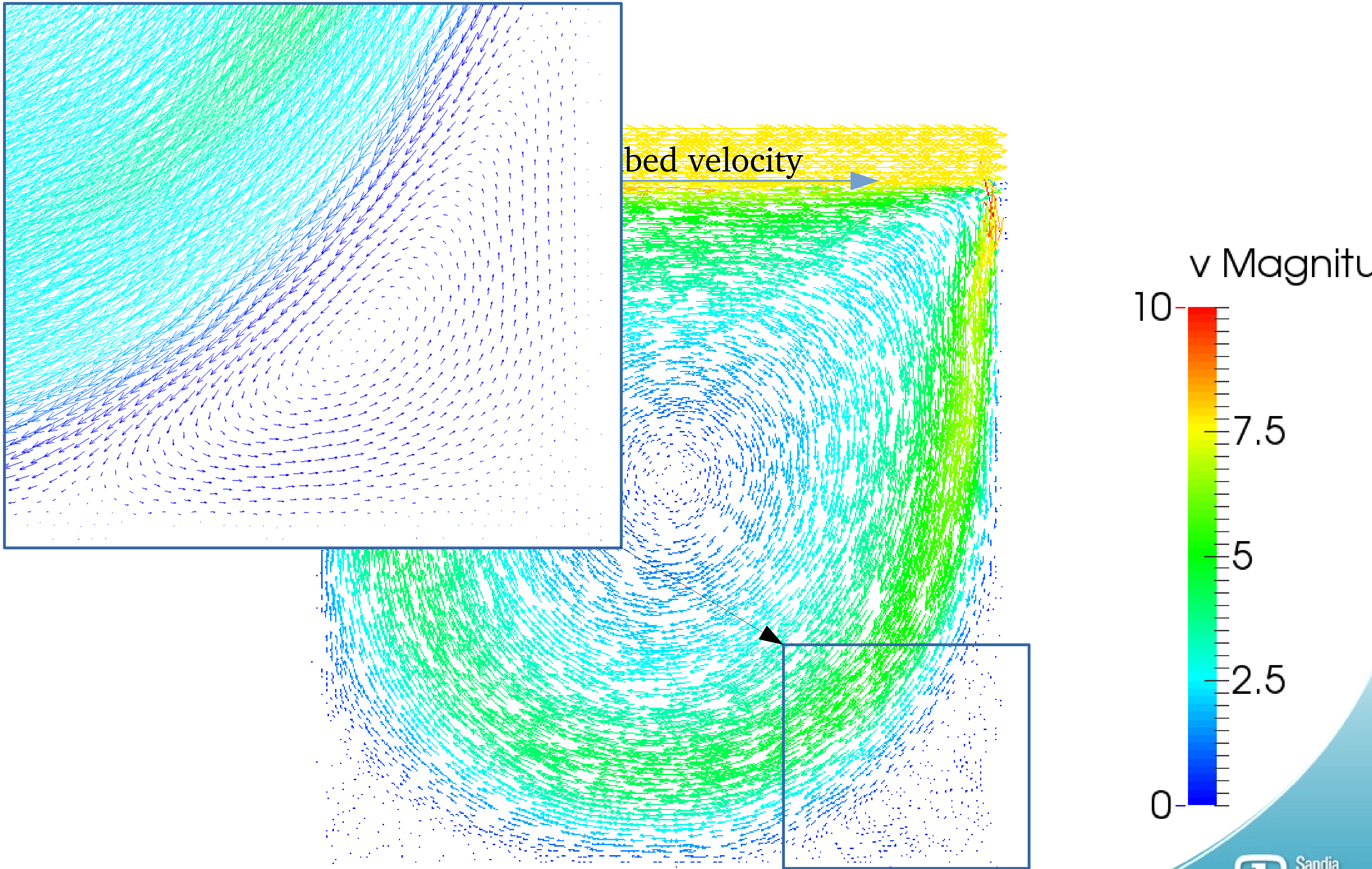


Navier Stokes numerical benchmarks
lid driven cavity



simulation: K. Kim

Navier Stokes numerical benchmarks
lid driven cavity



simulation: K. Kim

Towards electrokinetic problems: Poisson Boltzmann equation

Boltzmann charge distribution for a **two-ion monovalent** system: $\boxed{c^{\pm} = \pm c_b \exp(\mp\tilde{\psi})}$

Nonlinear Poisson-Boltzmann equation:

$$\boxed{-\nabla^2\tilde{\psi} = -\kappa^2 \sinh\tilde{\psi}, \qquad \text{with } \kappa^2 := 2\frac{c_b e^2}{\epsilon K_B T}}$$

Elctrokinetic ***forcing term*** appearing in the Navier Stokes equation

$$\boxed{\rho\,\mathbf{f} = 2\,e\,c_b\,\psi_{th}\,\sinh(\tilde{\psi})\,\nabla\tilde{\psi}}$$

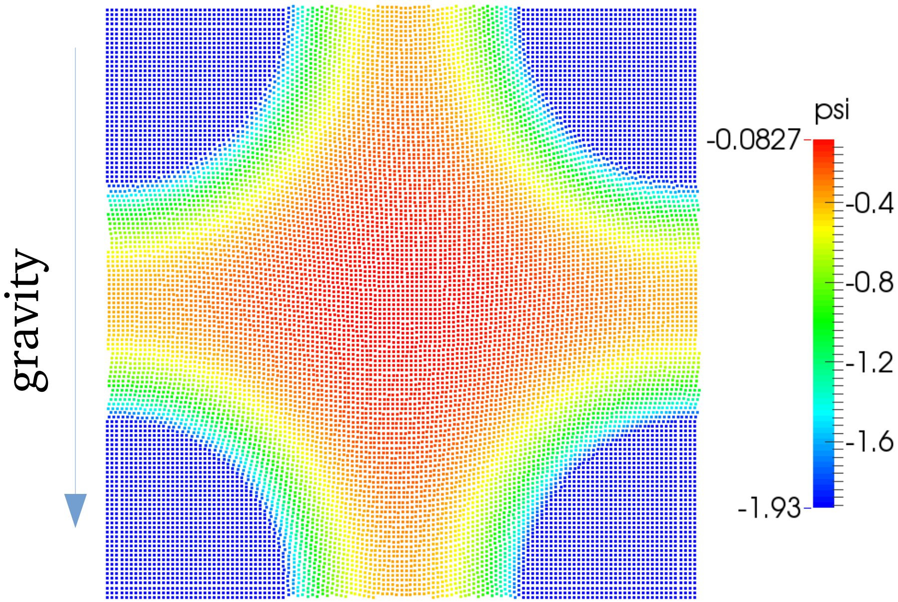
Polarized fluid is driven by electric potential

Implementation uses SPH Laplacian operator introduced before
Nonlinearity solved with Newton method (NOX package in Trilinos)

Sandia National Laboratories

Benchmarks on electrokinetic effect

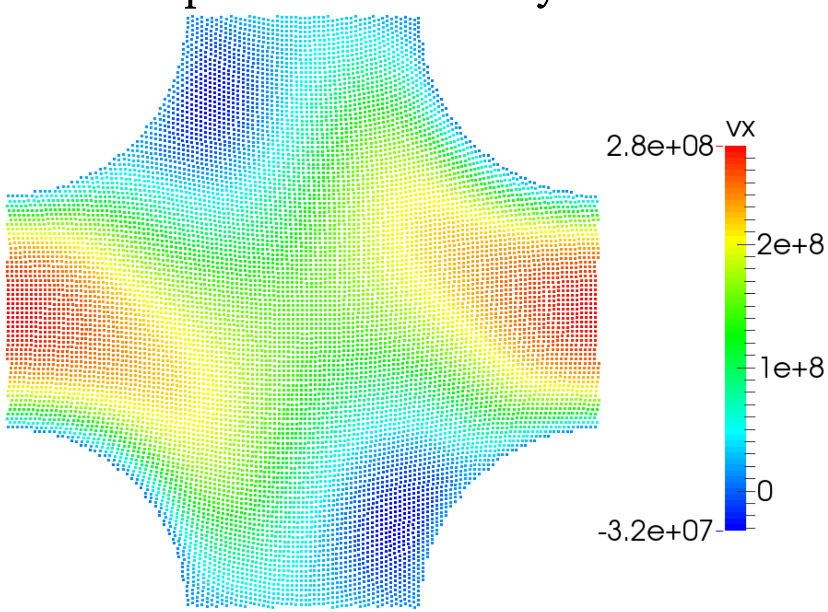
potential field



Four corners has fixed negative potential
Gravity field is prescribed

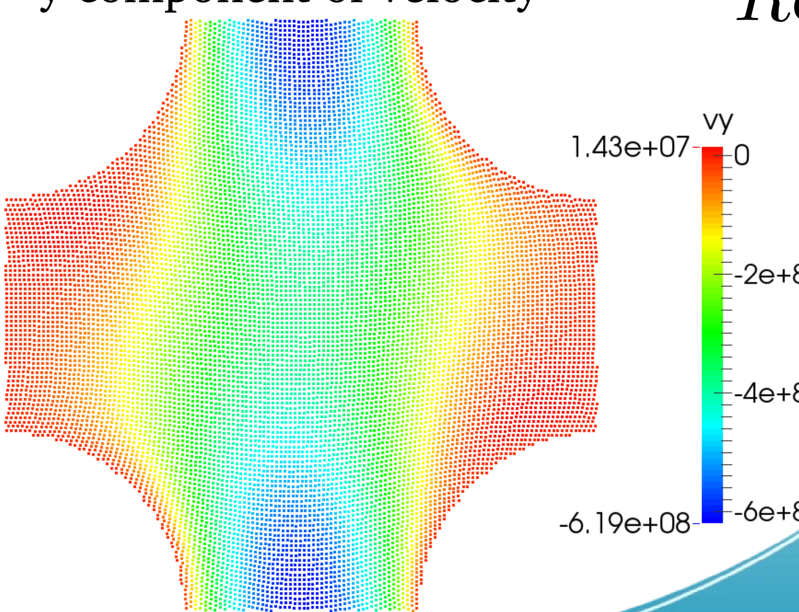
The electric potential clearly perturb the gravity driven flow

x component of velocity



y component of velocity

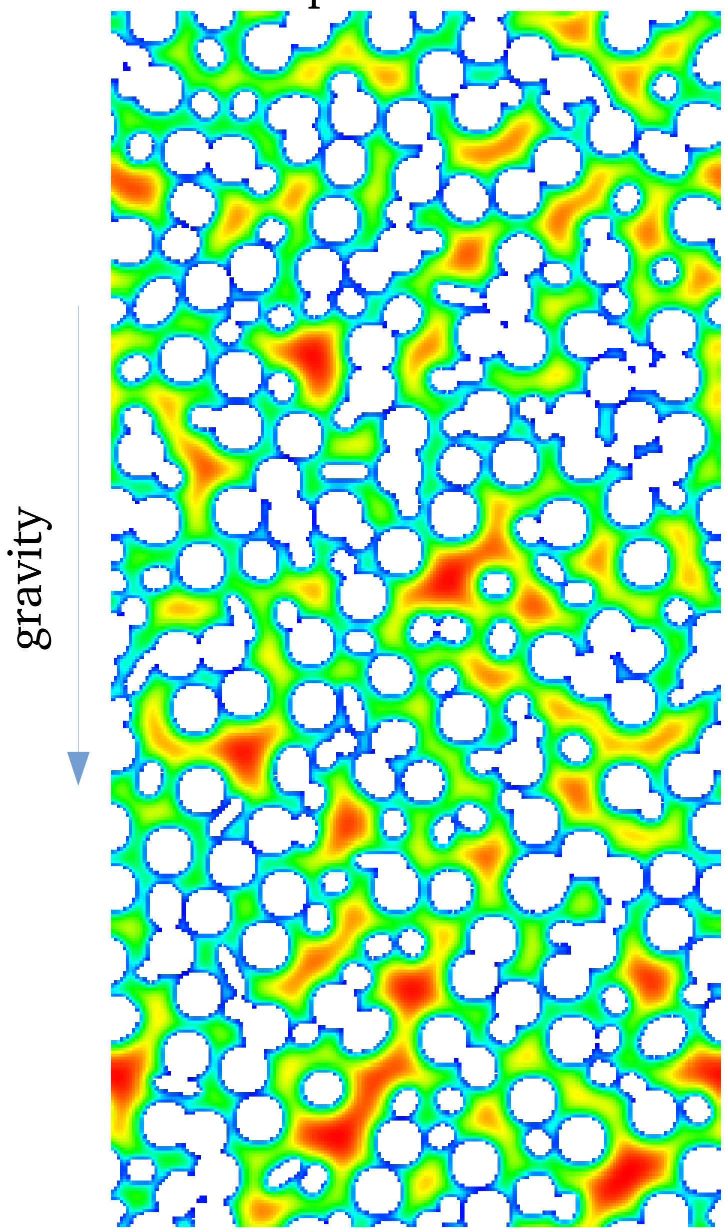
$$Re \simeq 0.02$$



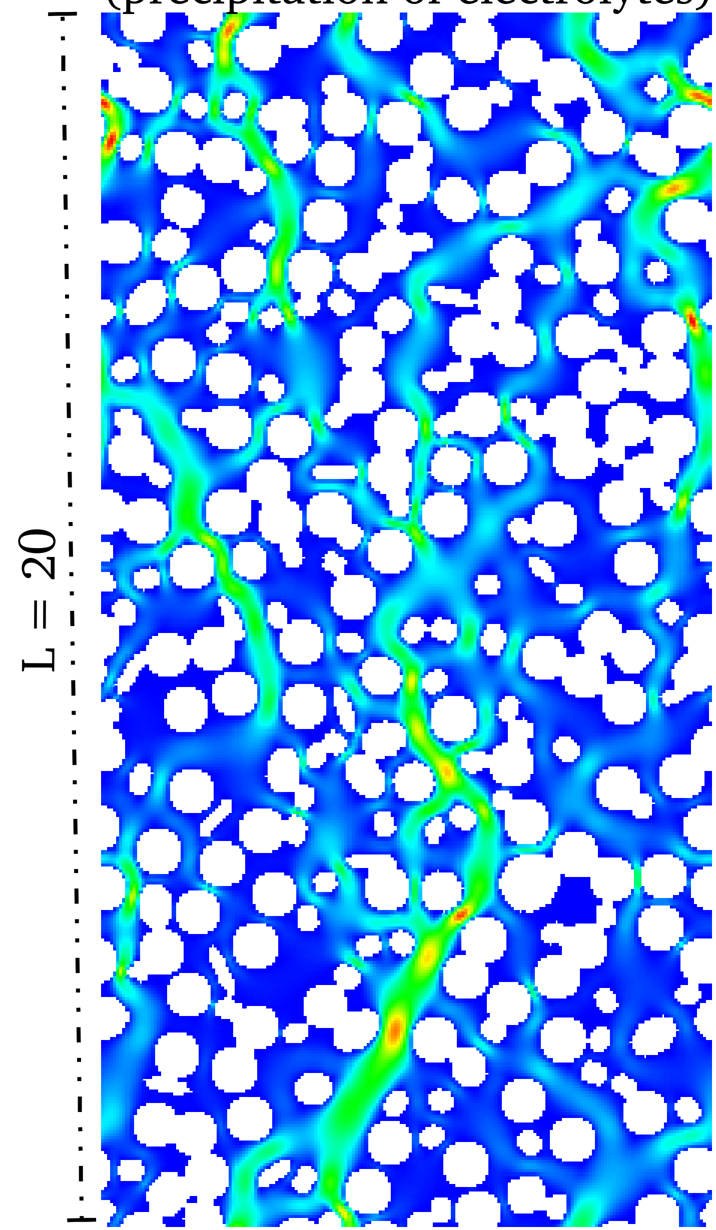
Problem setting: W. Pan, simulation: K. Kim

Li-air batteries application

potential



velocity magnitude
(precipitation of electrolytes)



Used first order time scheme to prevent oscillations

$$Re \simeq 10^{-7}$$

Problem setting: W. Pan, simulations: K. Kim

On-going / future developments

- Improve scheme stability for very low Reynolds numbers
- Use high-order mesh-free schemes (e.g. MLS)
- Design high order conservative schemes
- Improve model fidelity using Poisson-Nerst-Planck equations and/or use DFT (density functional theory)
- Solve new problems, e.g. microfluidic mixing

Sandia National Laboratories