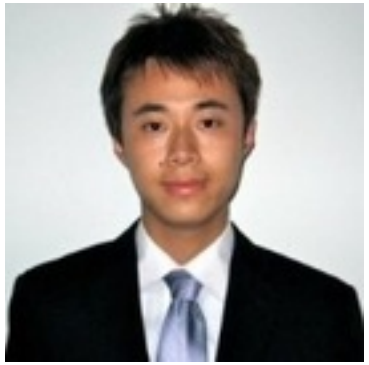





Computational Mesoscale Materials Problems

Dmitry Karpeyev

CI: UChicago & Argonne

Xujun Zhao, Xikai Jiang, Hong Zhang: MSD & MCS Argonne

with collaborators

Hanqi Guo, Carolyn Phillips, Todd Munson, Tom Peterka, MCS Argonne

Igor Aranson, George Crabtree, Andreas Glatz, Olle Heinonen, Ivan Sadovskyy, Alex Koshelev: MSD Argonne

Serge Nakhmanson, John Mangeri: U.Conn; Daniel Massatt, U. Minn.

the de Pablo Lab, IME U.Chicago

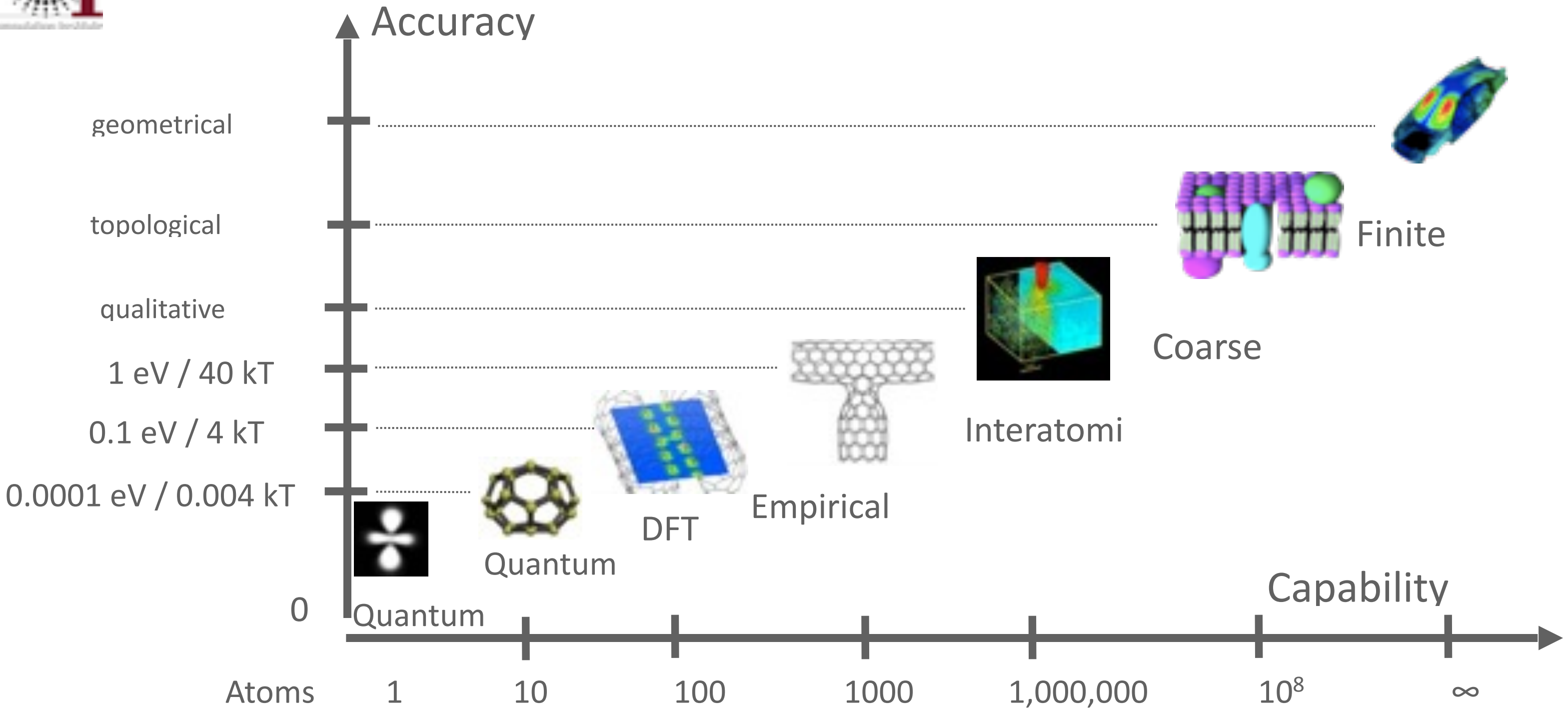
and many others!

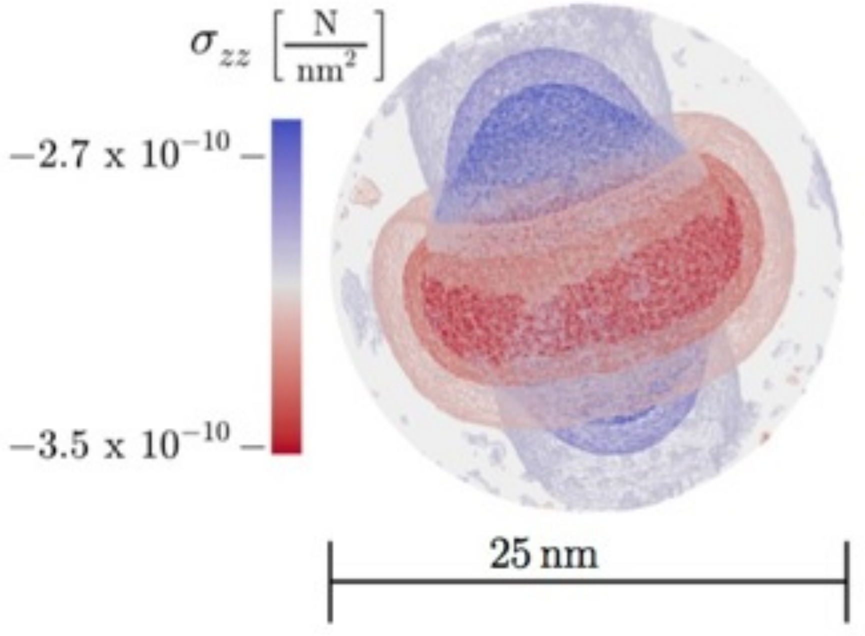


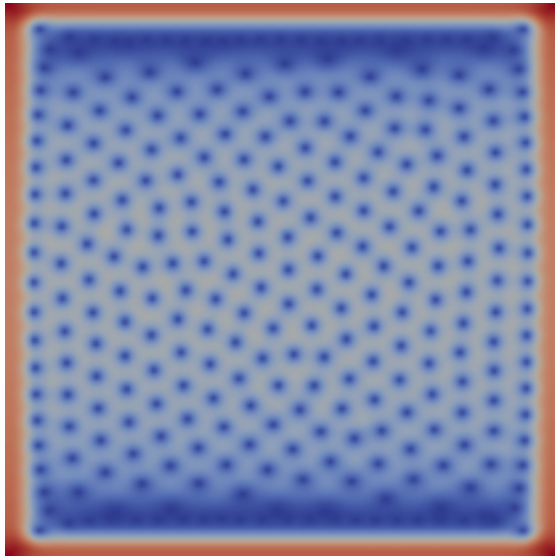




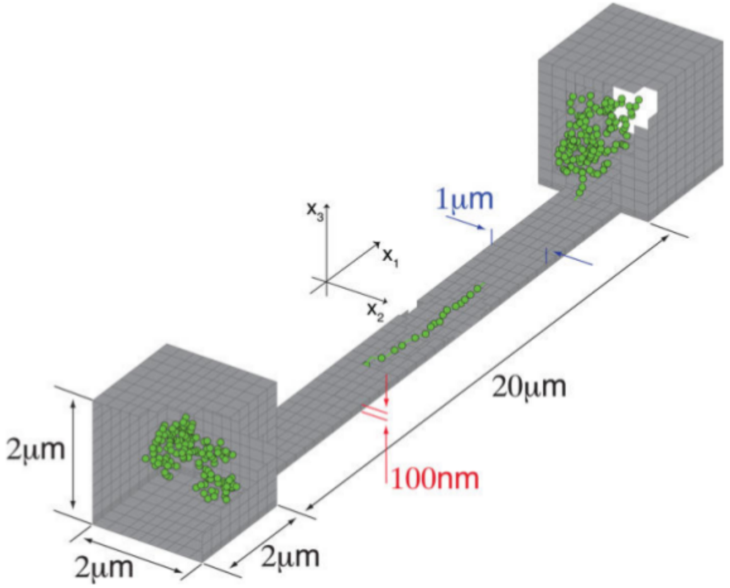
Discrete-continuum models of mesoscale phenomena







- Continuum and continuum-particle methods
- Methods/code development:
  - High performance simulation
  - Optimization/sensitivity
  - Visualization

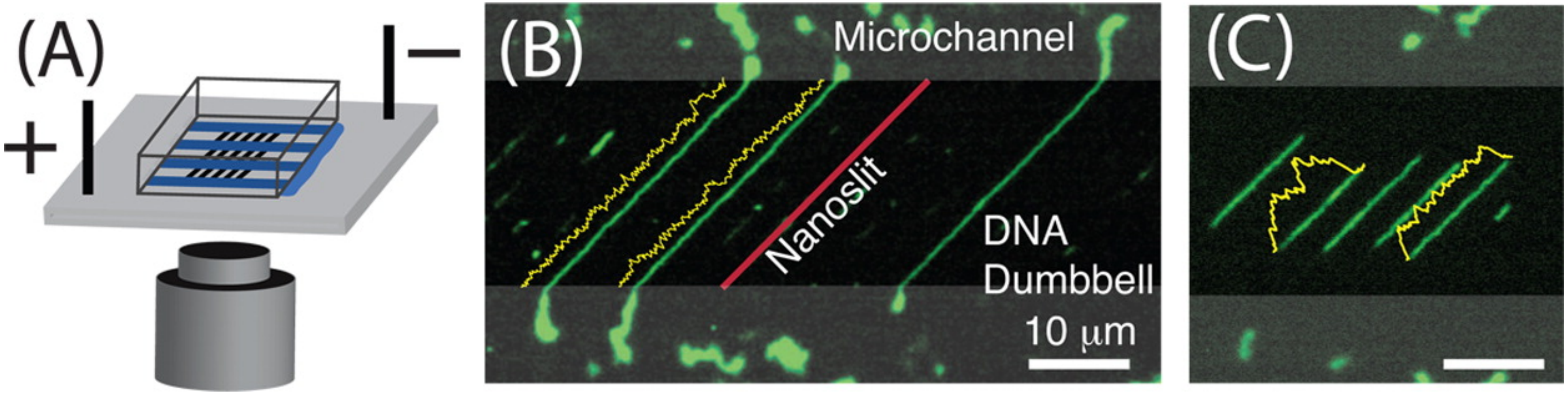




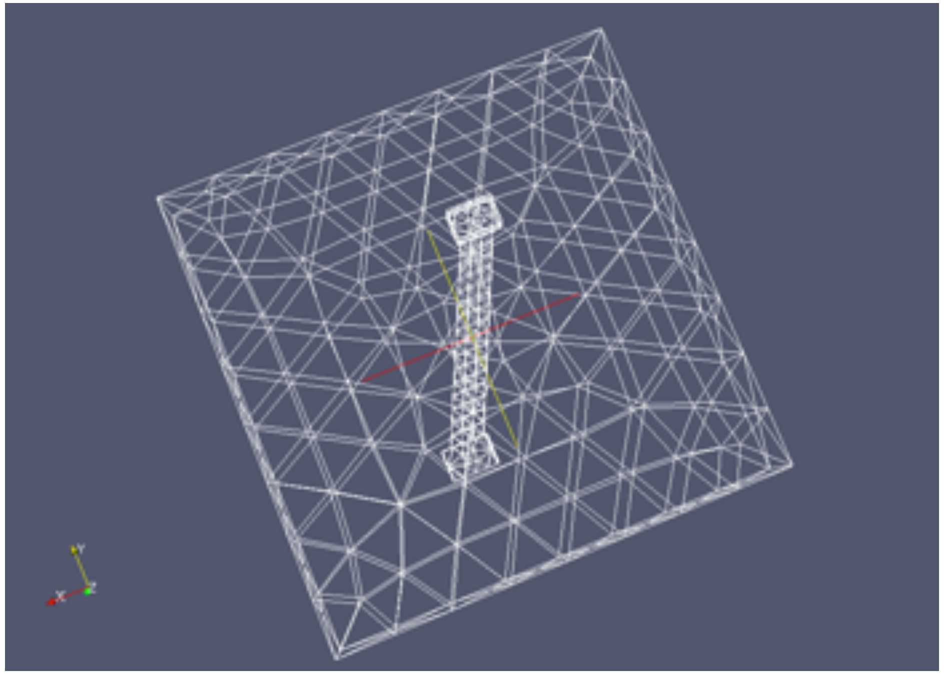
Soft matter: Institute for Molecular Engineering

THE INSTITUTE FOR
MOLECULAR
ENGINEERING

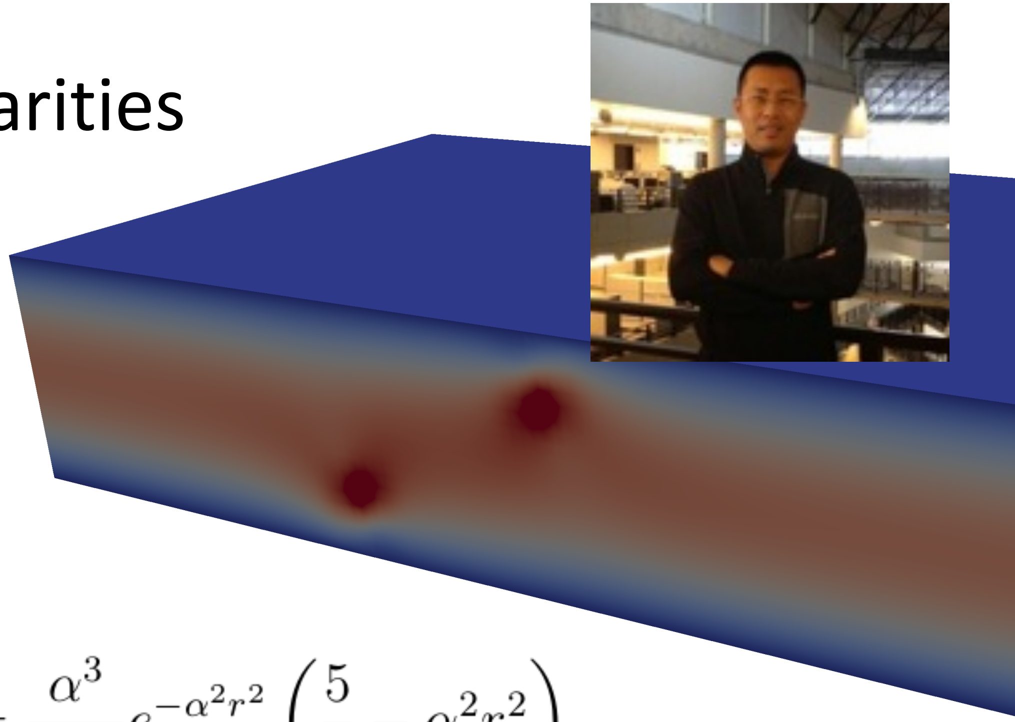
Translocation of DNA through nanochannels/nanoslits



- Multiphysics
  - Continuum:
    - Electrostatics
    - Hydrodynamics (Stokes)
    - Counter-ions drift-diffusion (Nernst-Planck)
  - Discrete:
    - Excluded volume (Lennard-Jones)
    - Nonlinear spring
  - Continuum-discrete:
    - Capture singular charges/forces
- “Geometry”
  - Separation of spatial scales
  - “Irregular” boundary
- Outer-loop
  - Long-time noise-driven evolution
  - Shape optimization



Resolving point singularities



- GGEM: General Geometry Ewald-like Method
  - O(N) via alpha tuning
  - PRL 98, 140602 (2007), J. Hernandez-Ortiz, J. de Pablo, M. Graham
  - Serial workhorse of particle simulations
  - Slow: weeks to months for physically relevant runs
- Parallelization based on PETSc/libMesh (Xujun Zhao)
  - Particle-particle computation may be suitable for GPU/MIC

$$-\nu\Delta u + \nabla p = \sum_i f_i \delta(x - x_i), \quad \nabla \cdot u = 0, \quad u|_\Gamma = \overline{u}$$

$$\delta(x - x_i) = g_\alpha(x - x_i) + \underbrace{(\delta(x - x_i) - g_\alpha(x - x_i))}_{\hat{\delta}_\alpha(x - x_i)}$$

$$-\nu\Delta u_l + \nabla p_l = \sum_i f_i g_\alpha(x - x_i), \quad \nabla u_l = 0, \quad u_l|_\Gamma = \overline{u} - \hat{u}|_\Gamma$$

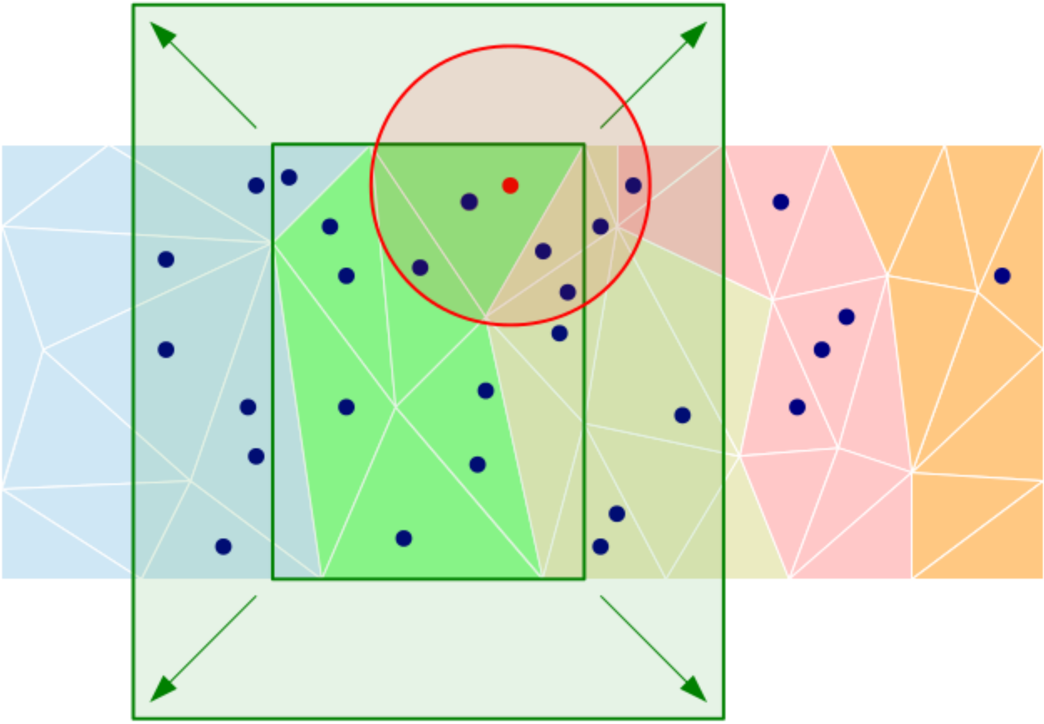
$$g_\alpha(x) = \frac{\alpha^3}{\pi^{3/2}} e^{-\alpha^2 r^2} \left(\frac{5}{2} - \alpha^2 r^2\right)$$

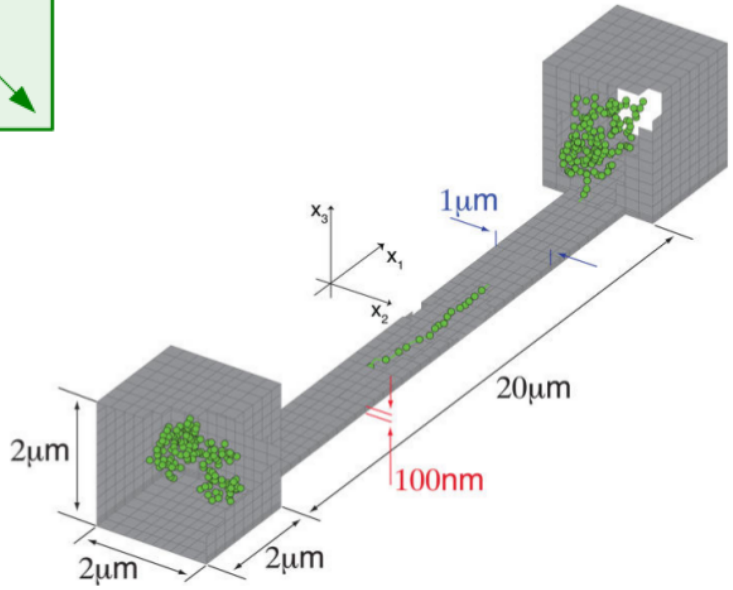
$$-\nu\Delta\hat{G}_\alpha + \nabla\hat{P}_\alpha = \sum_i f_i \hat{\delta}_\alpha(x - x_i)$$

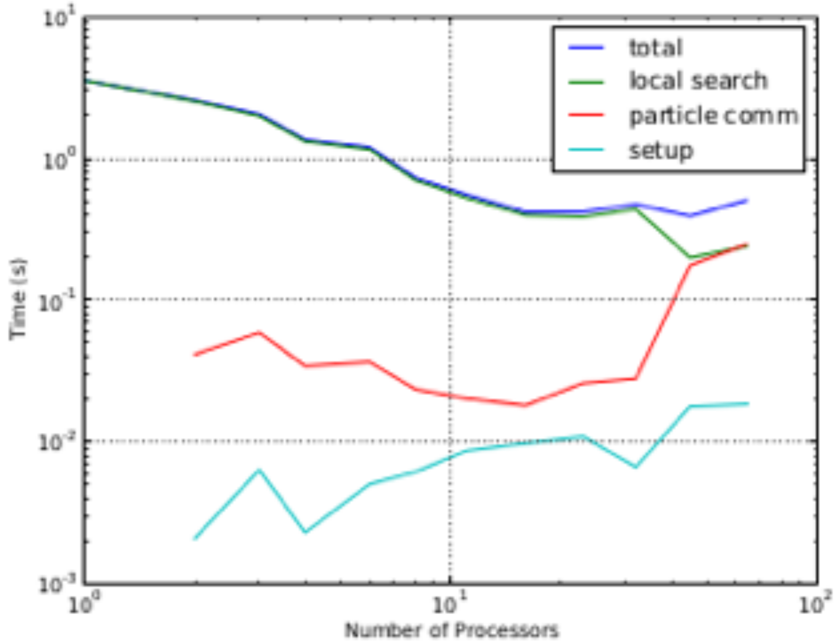
$$G_\alpha(x) = \frac{1}{8\pi\nu}\left(I - \frac{xx^T}{r}\right)\frac{1}{r} \times \underbrace{\mathrm{erfc}(\alpha r)}_{\mathrm{erfcx}(\alpha r)e^{-\alpha^2 r^2}} - \frac{1}{8\pi\nu}\left(I + \frac{xx^T}{r}\right)\frac{2\alpha}{\pi^{1/2}} e^{-\alpha^2 r^2}$$

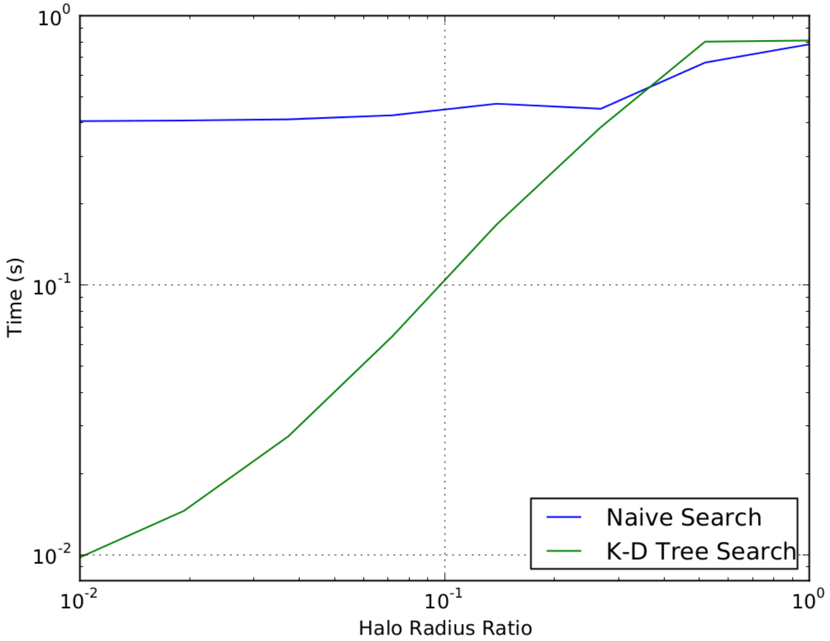
$$u_s(x) = \sum_i G_\alpha(x - x_i) f_i$$

Particles-mesh









Preconditioned Stokes Solver

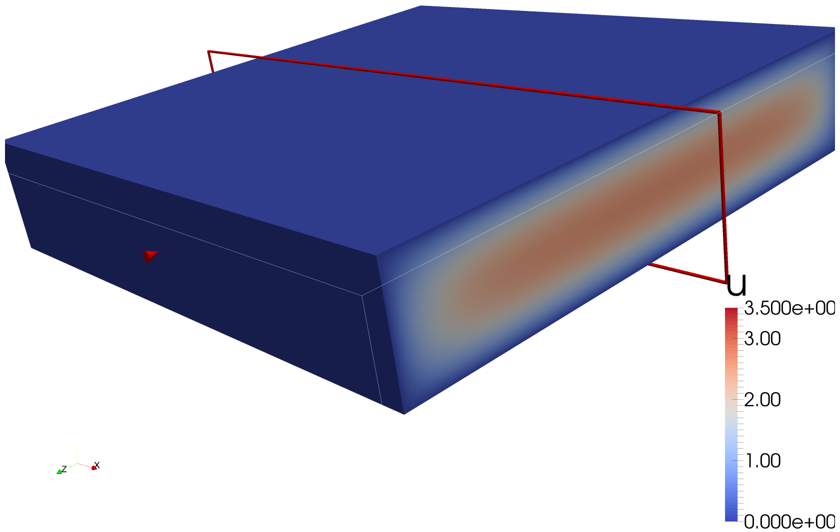
Stokes Equation:

$$-\nu\nabla^2 u + \nabla p = b$$
$$\nabla \cdot u = 0$$

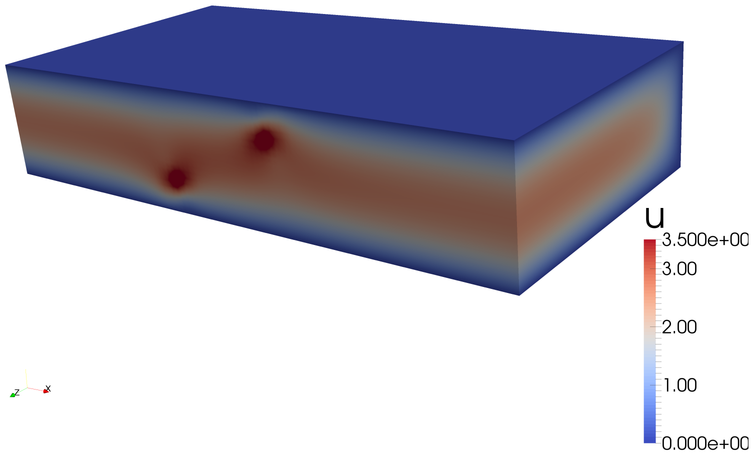
Mixed FEM

Discrete saddle point system:

$$\begin{pmatrix} \mathbf{A} & \mathbf{G} \\ \mathbf{G}^T & \mathbf{0} \end{pmatrix}\begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix}$$



Velocity in x-direction



Clip view at the cross section

Optimal FieldSplit configuration

	Direct Solver	Iterative Solver					
KSP	Super_LU (dist)	GMRES		TFQMR		GMRES	
PC		ASM		ASM		FIELDSPLIT(with user PC)	
Sub PC		ILU	ASM	ILU	ASM	multiplicative	Schur Complement
Iter #		377	377	219	219	56	43
time	2695.8s	125.8s	130.8	127.8	131.5	98.6	87.1s

- System size : 100 x 20 x 100 micrometers;
- Mesh: 50 x 10 x 50
- Element: Q2-Q1 mixed element
- Total DOF: 671,274
- Relative tol: 1E-9

- Pressure mass matrix in place of S

$$J = \begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} = \begin{pmatrix} I & 0 \\ B^T A^{-1} & I \end{pmatrix} \begin{pmatrix} A & 0 \\ 0 & -B^T A^{-1} B \end{pmatrix} \begin{pmatrix} I & A^{-1}B \\ 0 & I \end{pmatrix}$$

Optimal FieldSplit configuration

$$J = \begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix} = \begin{pmatrix} I & 0 \\ B^T A^{-1} & I \end{pmatrix} \begin{pmatrix} A & 0 \\ 0 & -B^T A^{-1} B \end{pmatrix} \begin{pmatrix} I & A^{-1}B \\ 0 & I \end{pmatrix}$$

- Can we find this optimal configuration automatically?
  - KSP(A):tol1, KSP_INNER(A): tol2,KSP(S):tol3
  - Uses derivative-free optimization (POUNDERS) over tol1,tol2,tol3
    - Limit: **500** total evaluations (Stokes solves), **17 hours**
    - 212 points over 7 local optimization runs and 288 points randomly sampled over the domain.
    - time-to-evaluate the starting points for the 6 completed local optimization:
    - 282.6, 291.5, 276.0, 271.5, 288.5, 294.7
    - These are 6 best randomly sampled points, the corresponding minima had solve time
    - 235.2, 282.2, 271.3, 256.0, 286.8, 270.6
    - So the improvement percentages are

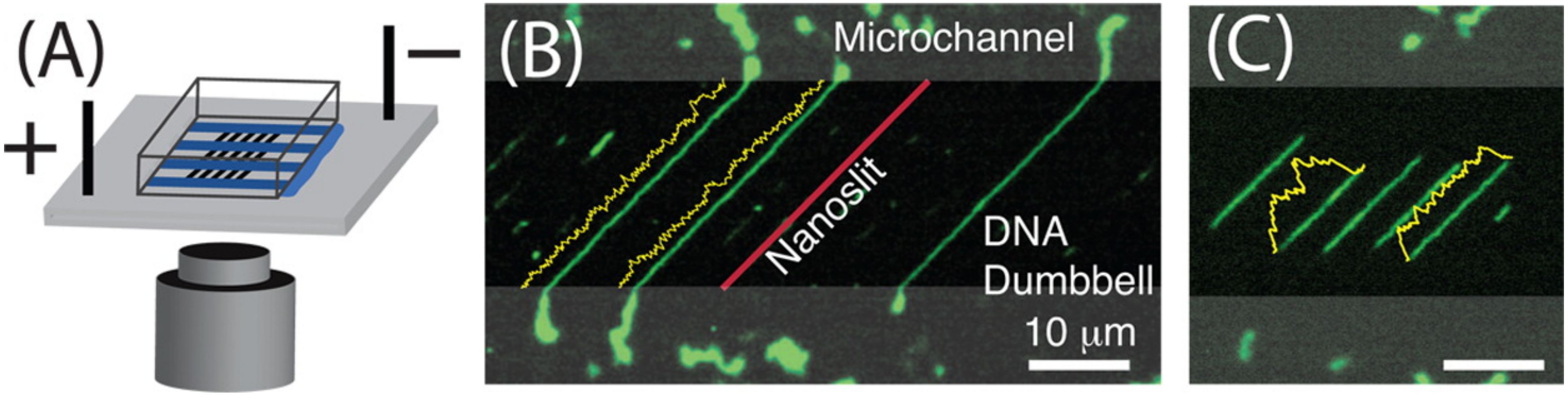
      16.8%, 3.2%, 1.7%, 5.7%, 0.6%, 8.2%, 0.4%

    Mean evaluation times for the 288 sample points: 407.2. Minimum found is 42% better.

Discrete choice (e.g., replacing S by Mp) requires more work

Work with Jeff Larson and Stefan Wild

Correlation matrix



- Very long-time simulation
- BdW by far most expensive
- Computed by Chebyshev approximation
- SLEPc spectral estimate, lagged
- Can we do better?
- Use Krylov space of M?
- H-matrix representation of M?

$$dx_i = u(x_i) + B_{ij} dW_j$$

$$u_i = M_{ij} f_j$$

$$M : f_i \to f(x) = \sum_i f_i \delta(x - x_i) \to \text{Stokes} \to u(x) \to u(x_i)$$

$$B = \sqrt{k_B T M}$$

Future (nascent) directions

- Extended particles
- Singular interfaces/boundaries
  - forces
  - charges
- GGEM not always applicable:
  - Boundary integral operator/equation formulations
  - Accelerated by FMM
  - In parallel
  - Take advantage of accelerators (GPU, etc.)?
  - Same for particles?

Thermoelastic Contact

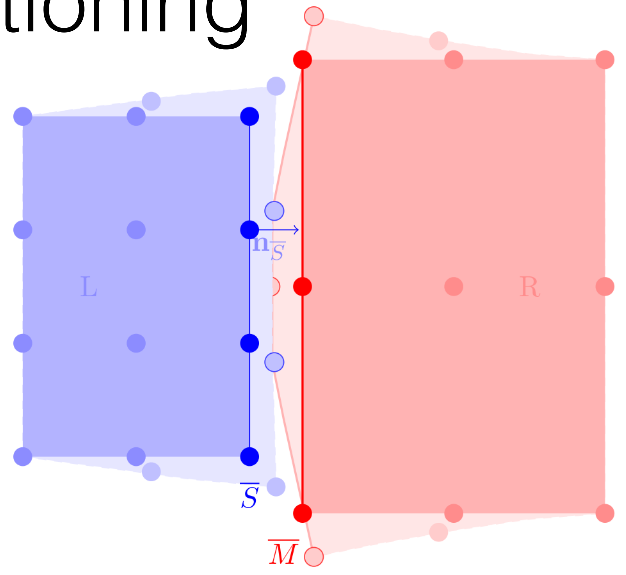
libMesh

MOOSE

Contact: FieldSplit preconditioning

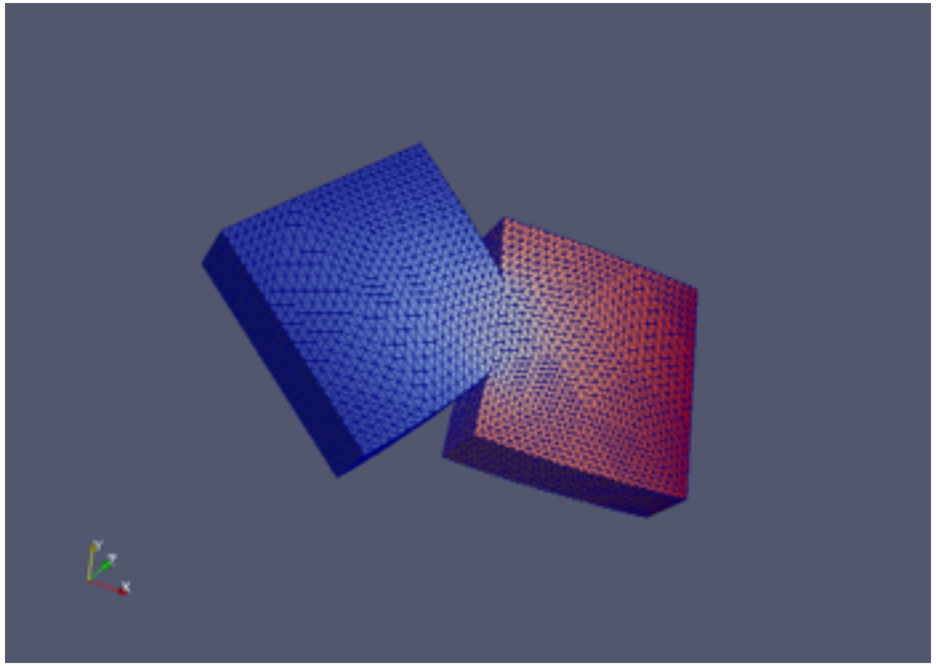
$$
\begin{array}{c} \\ \mathbf{u}_L \\ \mathbf{u}_S \\ \mathbf{u}_M \\ \mathbf{u}_R \\ \lambda \end{array}
\begin{array}{c}
\begin{array}{ccccc} \mathbf{u}_L & \mathbf{u}_S & \mathbf{u}_M & \mathbf{u}_R & \lambda \end{array} \\
\begin{pmatrix}
K_{LL} & K_{LS} & 0 & 0 & 0 \\
K_{SL} & K_{SS} & 0 & 0 & I \\
0 & 0 & K_{MM} & K_{MR} & -I \\
0 & 0 & K_{RM} & K_{RR} & 0 \\
0 & I & -I & 0 & 0
\end{pmatrix}
\end{array}
\begin{bmatrix} \delta\mathbf{u}_L \\ \delta\mathbf{u}_S \\ \delta\mathbf{u}_M \\ \delta\mathbf{u}_R \\ \delta\lambda \end{bmatrix} =
$$

$$
\begin{bmatrix}
\mathbf{f}_L - K_{LL}\mathbf{u}_L^0 - K_{LS}\mathbf{u}_S^0 \\
\mathbf{f}_S - K_{SL}\mathbf{u}_L^0 - K_{SS}\mathbf{u}_S^0 - \lambda^0 \\
\mathbf{f}_M - K_{MM}\mathbf{u}_M^0 - K_{MR}\mathbf{u}_R^0 + \lambda^0 \\
\mathbf{f}_R - K_{RM}\mathbf{u}_M^0 - K_{RR}\mathbf{u}_R^0 \\
\mathbf{x}_M - \mathbf{x}_S + (\mathbf{u}_M^0 - \mathbf{u}_S^0)
\end{bmatrix}
$$

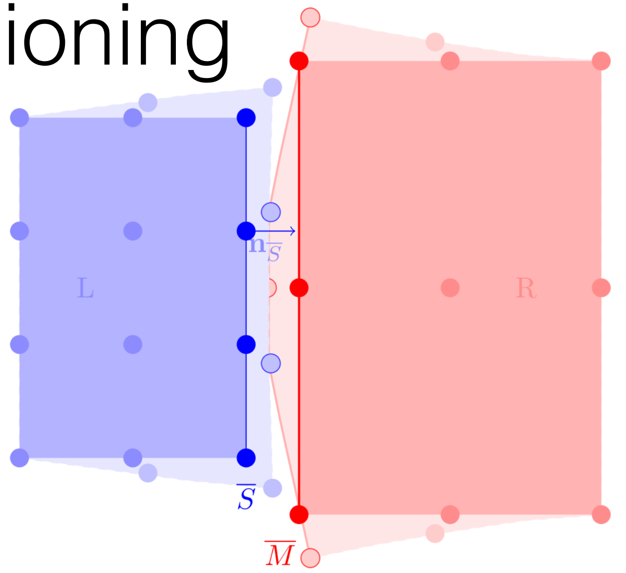


$$
\begin{gathered}
f(x) + \lambda^T B(x) = 0 \\
0 \le \lambda \perp g(x) \ge 0 \\
B(x) = \nabla g(x)
\end{gathered}
$$

$$
\begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix}
$$



Contact: FieldSplit preconditioning



$$
\begin{array}{c} \\ \mathbf{u}_S \\ \lambda \\ \cdots \\ \mathbf{u}_L \\ \mathbf{u}_M \\ \mathbf{u}_R \end{array}
\begin{array}{c}
\begin{array}{ccccc} \lambda & \mathbf{u}_S & \mathbf{u}_L & \mathbf{u}_M & \mathbf{u}_R \end{array} \\
\left(\begin{array}{cc:ccc}
I & K_{SS} & K_{SL} & 0 & 0 \\
0 & I & 0 & -I & 0 \\
\hdashline
0 & K_{LS} & K_{LL} & 0 & 0 \\
-I & 0 & 0 & K_{MM} & K_{MR} \\
0 & 0 & 0 & K_{RM} & K_{RR}
\end{array}\right)
\end{array}
\begin{bmatrix} \delta\lambda \\ \delta\mathbf{u}_S \\ \cdots \\ \delta\mathbf{u}_L \\ \delta\mathbf{u}_M \\ \delta\mathbf{u}_R \end{bmatrix}
$$

- General VIs

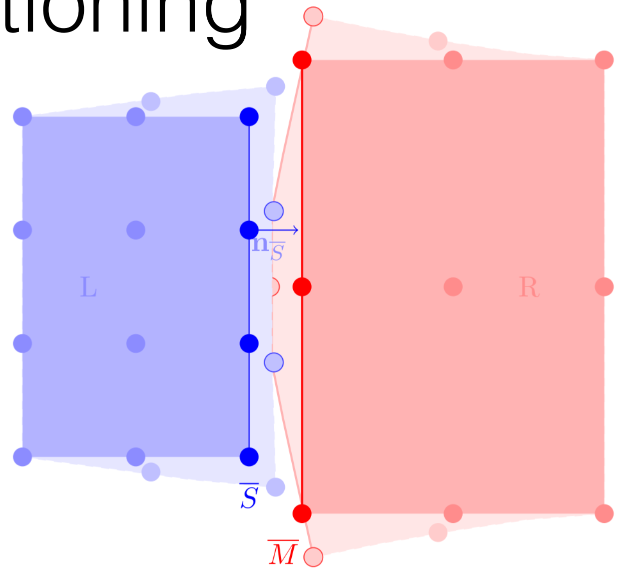
$$
\begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix}
$$

$$
\begin{array}{c} \\ \mathbf{u}_S \\ \lambda \\ \cdots \\ \mathbf{u}_L \\ \mathbf{u}_M \\ \mathbf{u}_R \end{array}
\begin{array}{c}
\begin{array}{cc:ccc} \lambda & \mathbf{u}_S & \mathbf{u}_L & \mathbf{u}_M & \mathbf{u}_R \end{array} \\
\left(\begin{array}{cc:ccc}
I & K_{SS} & K_{SL} & 0 & 0 \\
0 & I & 0 & -I & 0 \\
\hdashline
0 & 0 & K_{LL} & K_{LS} & 0 \\
0 & 0 & K_{SL} & K_{MM} + K_{SS} & K_{MR} \\
0 & 0 & 0 & K_{RM} & K_{RR}
\end{array}\right)
\end{array}
\begin{bmatrix} \delta\lambda \\ \delta\mathbf{u}_S \\ \cdots \\ \delta\mathbf{u}_L \\ \delta\mathbf{u}_M \\ \delta\mathbf{u}_R \end{bmatrix}
$$

- Primal reduced system
- Also phasefield models (volume fraction constraint)
- On-going work with Todd Munson, Jason Sarich, Fande Kong

Contact: FieldSplit preconditioning

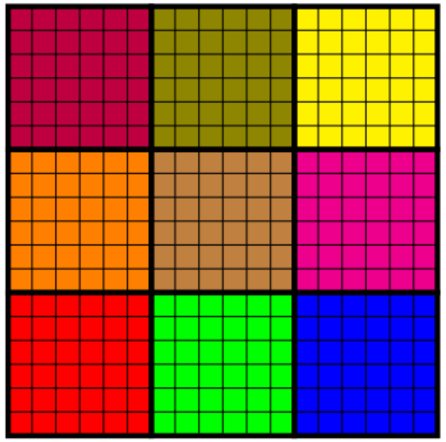
$$
\begin{array}{c} \\ \mathbf{u}_S \\ \cdots \\ \lambda \\ \mathbf{u}_L \\ \mathbf{u}_M \\ \mathbf{u}_R \end{array}
\begin{array}{c}
\begin{array}{ccccc} \lambda & \mathbf{u}_S & \mathbf{u}_L & \mathbf{u}_M & \mathbf{u}_R \end{array} \\
\left(\begin{array}{c:cccc}
I & K_{SS} & K_{SL} & 0 & 0 \\
\hdashline
0 & I & 0 & -I & 0 \\
0 & K_{LS} & K_{LL} & 0 & 0 \\
-I & 0 & 0 & K_{MM} & K_{MR} \\
0 & 0 & 0 & K_{RM} & K_{RR}
\end{array}\right)
\end{array}
\begin{bmatrix} \delta\lambda \\ \cdots \\ \delta\mathbf{u}_S \\ \delta\mathbf{u}_L \\ \delta\mathbf{u}_M \\ \delta\mathbf{u}_R \end{bmatrix}
$$

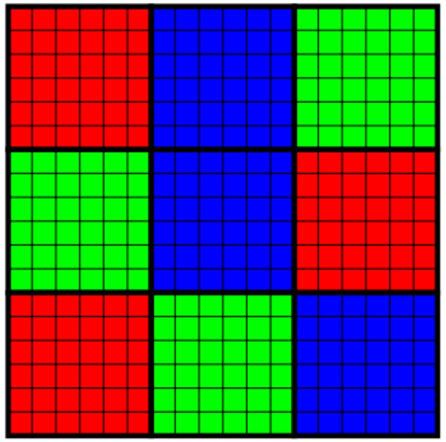


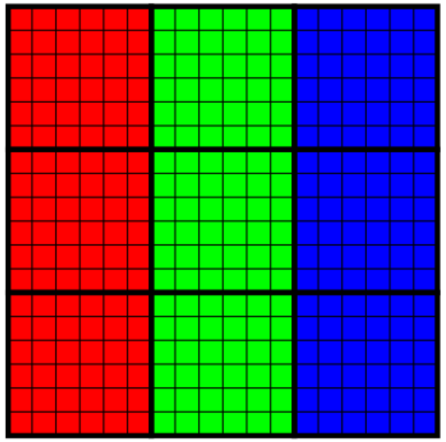
$$
\begin{array}{c} \\ \mathbf{u}_S \\ \cdots \\ \lambda \\ \mathbf{u}_L \\ \mathbf{u}_M \\ \mathbf{u}_R \end{array}
\begin{array}{c}
\begin{array}{ccccc} \lambda & \mathbf{u}_S & \mathbf{u}_L & \mathbf{u}_M & \mathbf{u}_R \end{array} \\
\left(\begin{array}{c:cccc}
I & K_{SS} & K_{SL} & 0 & 0 \\
\hdashline
0 & I & 0 & -I & 0 \\
0 & K_{LS} & K_{LL} & 0 & 0 \\
0 & K_{SS} & K_{SL} & K_{MM} & K_{MR} \\
0 & 0 & 0 & K_{RM} & K_{RR}
\end{array}\right)
\end{array}
\begin{bmatrix} \delta\lambda \\ \cdots \\ \delta\mathbf{u}_S \\ \delta\mathbf{u}_L \\ \delta\mathbf{u}_M \\ \delta\mathbf{u}_R \end{bmatrix}
$$

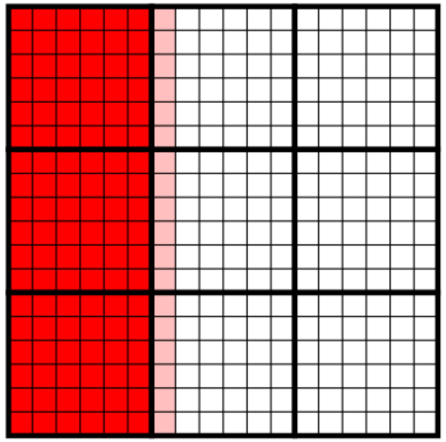
- Preconditioner?
- PCASM is remarkably robust
- Limited to small subdomains

PCGASM









- Multirank subdomains
- Hierarchical partitioning
- Multirank MatIncreaseOverlap()
  - On-going work with Fande Kong

Geometric multigrid support for libMesh

Phasefield crystal

- Phasefield Crystal (PFC) is used in problems where atomic effects are needed, but on a larger time scale, typically microseconds.
- PFC is a type of Density Functional Theory, which requires minimizing the energy functional:

$$\frac{\beta \triangle F}{\rho_0} = \int dr\big([1+n(r)]\ln[1+n(r)] - n(r)\big) - \frac{\rho_0}{2}\int\int dr_1 dr_2 n(r_1) c^{(2)}(|r_1 - r_2|) n(r_2)$$

INL LDRD: M. Tonks, Y. Zhang
U.Michigan: K. Thornton's group
D. Massatt: 2014 Argonne Givens Fellow

- The Fourier Transform of $c^{(2)}$ can be approximated by a Rational Function Fit, $\rho_0 \hat{c}^{(2)}_{RFF} = \sum_{j=1}^{m} \left[\frac{A_j}{k^2+\alpha_j} + \frac{A_j^*}{k^2+\alpha_j^*} \right]$
- Taking the inverse Fourier Transform, one finds:

$$\rho_0 \int c^{(2)}(|r_1 - r_2|) n(r_2) dr_2 = \sum_j [L_j(r) + L_j^*(r)]$$

Where L_j defined to be the solution to $-\triangle L_j(r) + \alpha_j L_j(r) = A_j n(r)$, and $-\triangle L_j^*(r) + \alpha_j^* L_j^*(r) = A_j^* n(r)$.

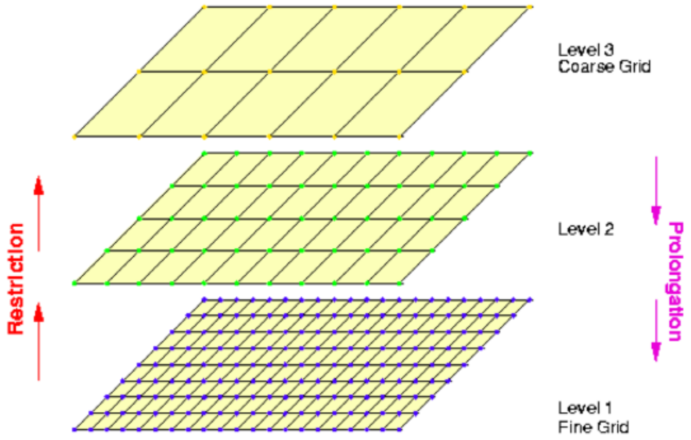
Helmholtz Equation

$$-\triangle u + \gamma u = f$$

$$A^h u^h = f^h$$

Difficulties of Solving Helmholtz

1. GMRES Block Jacobi or Additive Schwarz Method (ASM) preconditioning have poor convergence rates
2. Geometric Multigrid diverges
3. Algebraic Multigrid (AMG) using Hypre BoomerAMG has too expensive a setup time



[Luksch]

Helmholtz Eigenvalues

$$-\frac{d^2}{dx^2}u - k^2 u = f, \qquad A^h u^h = f^h$$

Eigenvalues: $\lambda_i = \frac{4}{h^2}\sin^2\left(\frac{\pi i h}{2}\right) - k^2$

- Prolongation generates error dependent on $(1 - \frac{\lambda^h}{\lambda^H})$, which makes eigenvalue sign changes problematic, so use GMRES as an outer iteration
- For $\pi/5 \leq kh \leq 2\cos(\pi h/2)$, damped Jacobi smoothers have poor convergence, so use GMRES as a smoother on these intermediate levels.

	256 Elements				512 Elements			
	k = 4 π		k = 8 π		k = 4 π		k = 8 π	
# levels	MG	GMRS	MG	GMRS	MG	GMRS	MG	GMRS
2	6	3	11	4	7	3	6	4
3	25	5	-	6	10	6	-	5
4	-	6	-	8	-	6	-	7
5	-	7	-	12	-	7	-	8
6	-	10	-	16	-	8	-	12
7	-	11	-	19	-	10	-	17
8	-	12	-	20	-	11	-	19
9	-	12	-	20	-	12	-	19
10					-	12	-	19

For the 3D, we compare using Multigrid with Damped Jacobi Smoothers to adding FGMRES as an outer iteration, and then adding GMRES smoothers to the appropriate intermediate and coarse levels.

Domain: 65 x 65 x 65, 6 levels of Multigrid			
γ	MG	FGMRES outer	Elman smoothing
0	5	4	N/A
-.0606 +.746i	5	4	N/A
-3.062 +.7919i	5	4	N/A
-10 + i	5	4	4
-25 + i	5	4	4
-27 + i	6	5	4
-28.5 + i	24	6	6
-30 + i	-	6	6
-50 + i	-	6	4
-100 + i	-	14	8
-200 + i	-	54	9
-300 + i	-	391	11
-400 + i	-	2000+	19

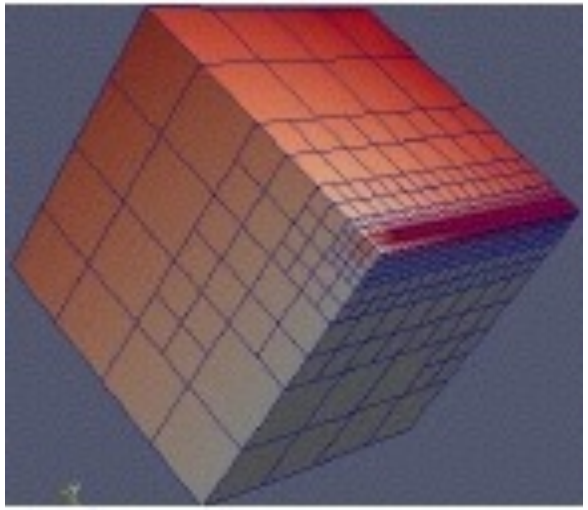
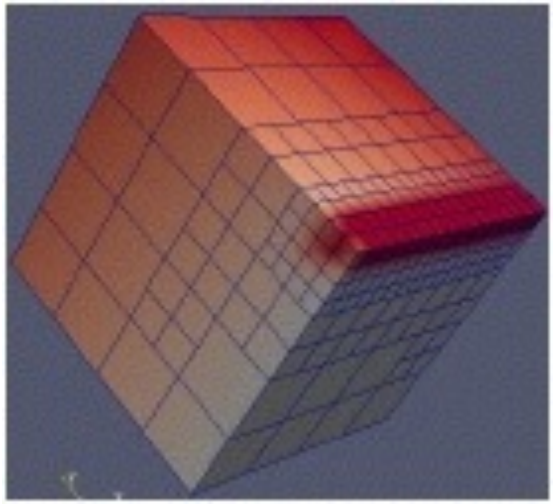
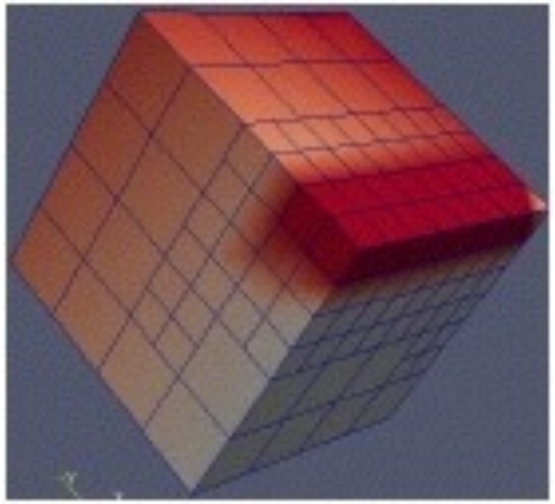
Here we compare Elman's method to GMRES with ASM preconditioner, and to AMG.

Domain: 401 x 401 x 401, 5 levels of Multigrid				
γ	FGMRES, PC Multigrid		ASM	
-.0606 +.746i	3	176s	555	2080s
-3.239 +.472i	3	165s	595	2180s
-1.568 + .601i	3	174s	556	2046s
-1.734 + 1.074i	3	168s	574	2113s
-3.062 + .7919i	3	181s	593	2169s
-1.554 - 1.394i	3	170s	572	2087s

Domain: 201 x 201 x 201, 4 levels of Multigrid				
γ	FGMRES, PC Multigrid		Hypre, BoomerAMG	
-.0606 +.746i	4	23s	4	357s
-3.239 +.472i	4	24s	4	352s
-1.568 + .601i	4	26s	4	362s
-1.734 + 1.074i	4	27s	4	352s
-3.062 + .7919i	4	28s	4	356s
-1.554 - 1.394i	4	22s	4	351s

Multigrid on AMR meshes

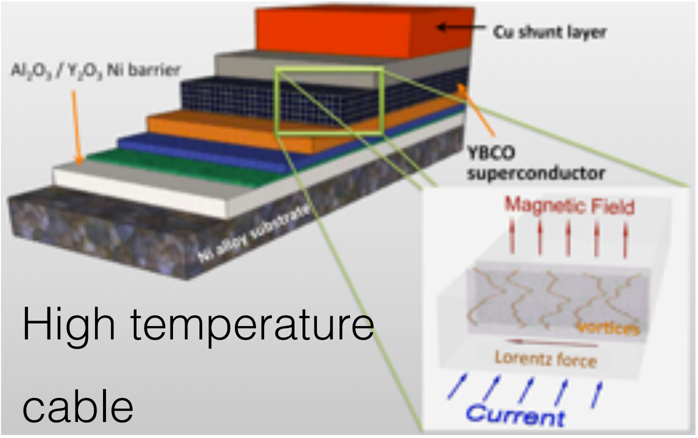
- Most interesting problems are not on uniform grids, so we move to unstructured grids.
- We are using Fast Adaptive Composite (FAC) grid refinement since it is simpler to setup in libMesh.
- The Multi-level Adaptive Technique (MLAT) is faster, but harder to implement.

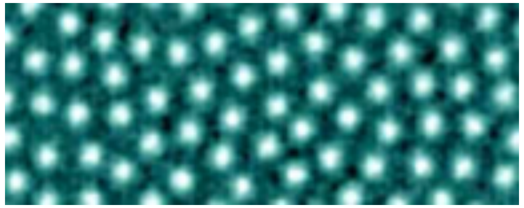




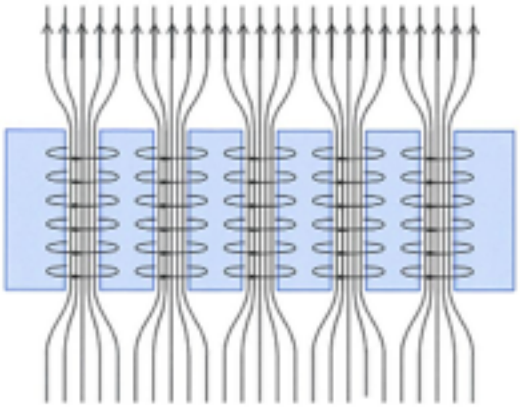
High temperature type-II superconductors (zero electrical resistance material)

oscon-scidac.org

- Magnetic field penetrates the superconductor as quantized fluxes – magnetic vortices
- Vortices are flexible tubes that move, twist, repel, merge.
- Vortices determine *all* the electrodynamic responses of superconductors to electric and magnetic fields
- Vortex moving leads to power dissipation. Vortices can be pinning on non-superconducting defects





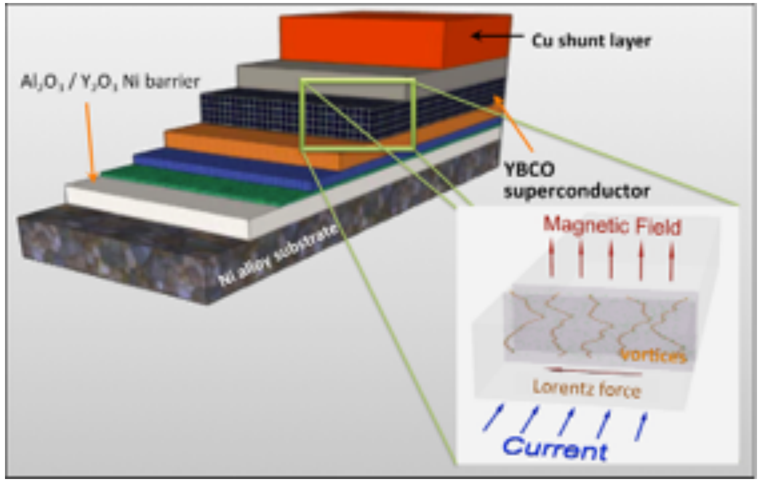


Magnetic vortices

Lossless energy transport in through superconducting cables

oscon-scidac.org

1st generation cable including insulation & cooling



high-current transmission (in urban areas, here NY)

2nd generation cable with illustration of vortex motion

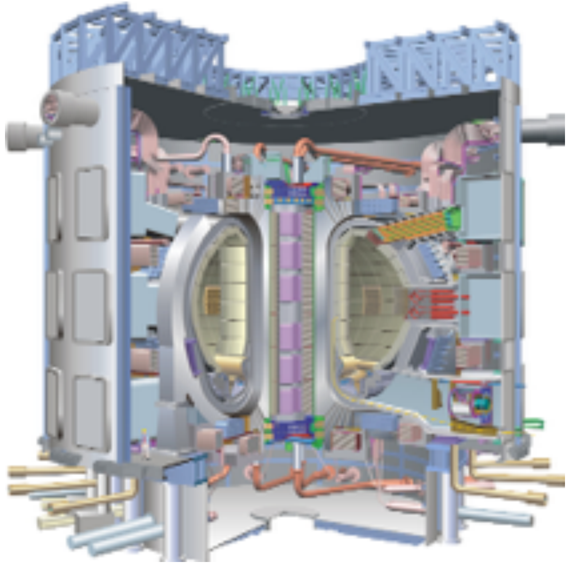
compact generators & motors

Other applications

oscon-scidac.org

LHC magnets

Maglev trains



ITER magnets

Diagnostic applications (MNR, MRI, …)

Superconducting cables

oscon-scidac.org

- 5x power capacity of copper in same cross-sectional area
  - Relieve urban power bottleneck in cities and suburbs
- Cables operating at 77 K are technically ready
  - in-grid demonstrations at Copenhagen DK, Albany NY, Long Island NY, Columbus OH, New Orleans LA, Amsterdam

Barriers to grid penetration

- Reduce cost by factor 10 - 100 to compete with copper
- Demonstrate reliable multiyear operation

Ginzburg-Landau equations

oscon-scidac.org

Time dependent Ginzburg-Landau equations

$$\frac{\partial \Psi}{\partial t} = -\frac{\delta \mathcal{F}_{\mathrm{GL}}}{\delta \Psi^*}\,, \quad \frac{\delta \mathcal{F}_{\mathrm{GL}}}{\delta \mathbf{A}} = 0$$

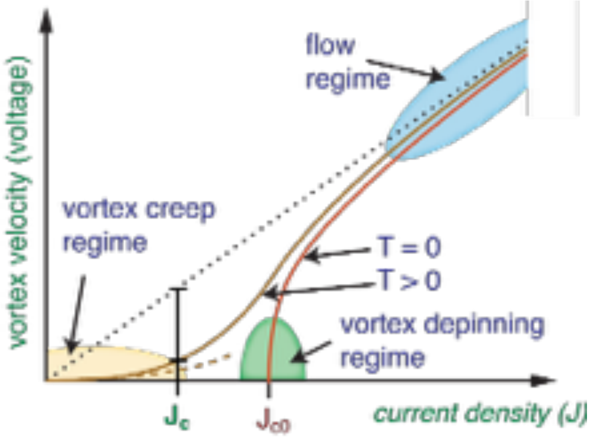
$$u(\partial_t + i\mu)\psi = \epsilon(\mathbf{r})\psi - |\psi|^2\psi + (\nabla - i\mathbf{A})^2\,\psi + \zeta(\mathbf{r}, t)$$

$$\kappa^2 \nabla \times (\nabla \times \mathbf{A}) = \mathbf{J}_n + \mathbf{J}_s + \mathcal{I},$$

Total current **J** = $\mathbf{J}_s + \mathbf{J}_n$

$$\mathbf{J} = \mathrm{Im}\left[\psi^*(\nabla - i\mathbf{A})\psi\right] - (\nabla\mu + \partial_t \mathbf{A})$$

Critical current J_C (maximal possible non-dissipative current) is usually defined when voltage V is a small fraction (e.g., 1%) of the free flow value V_{ff}.



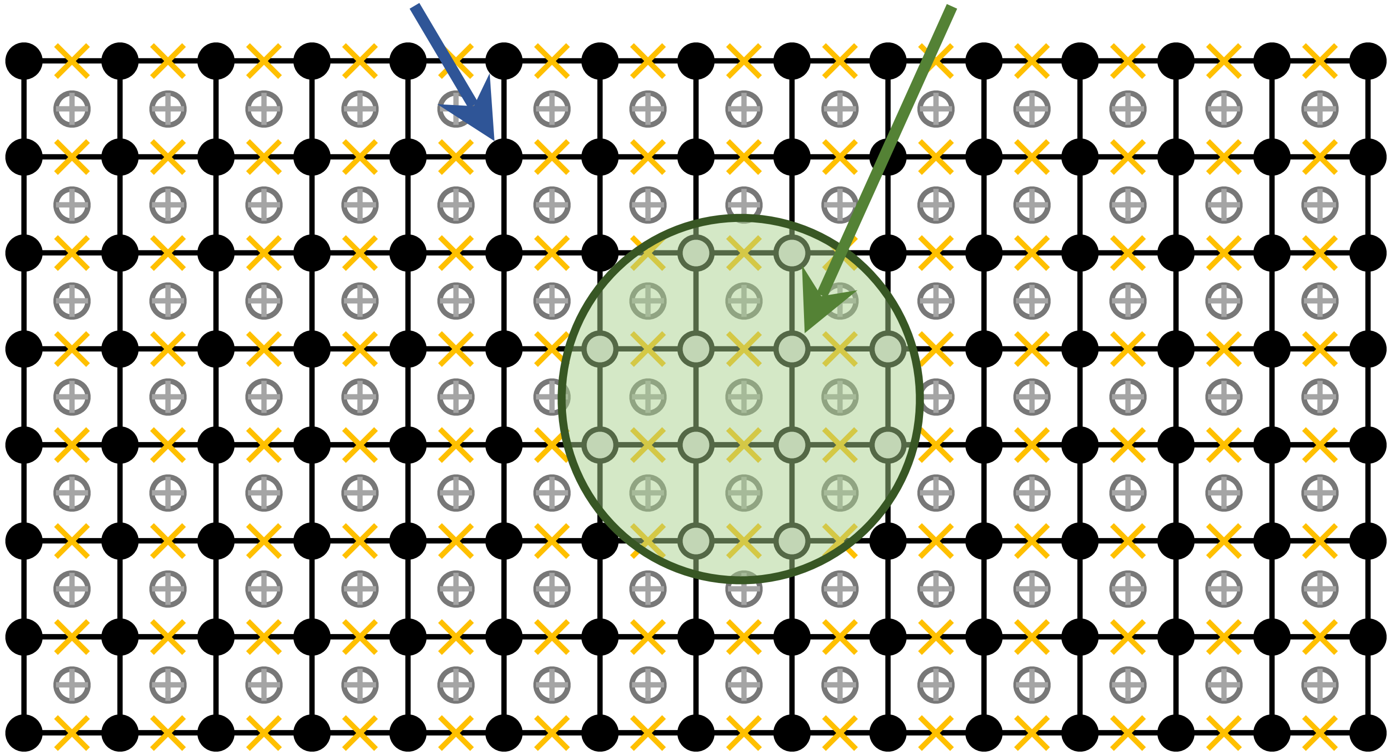
Modelling of the inclusions

oscon-scidac.org

$$u(\partial_t + i\mu)\psi = \epsilon(\mathbf{r})\psi - |\psi|^2\psi + (\nabla - i\mathbf{A})^2\psi + \zeta(\mathbf{r}, t)$$

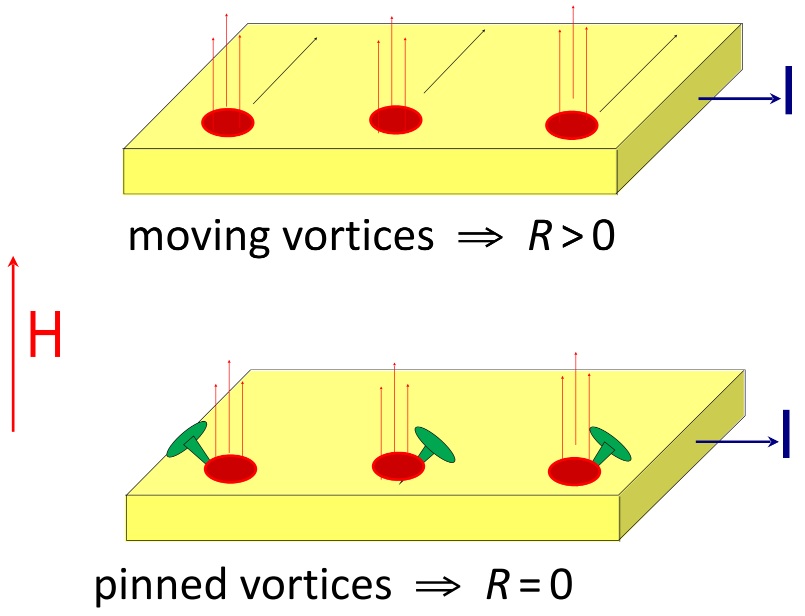
ε > 0 in superconductor

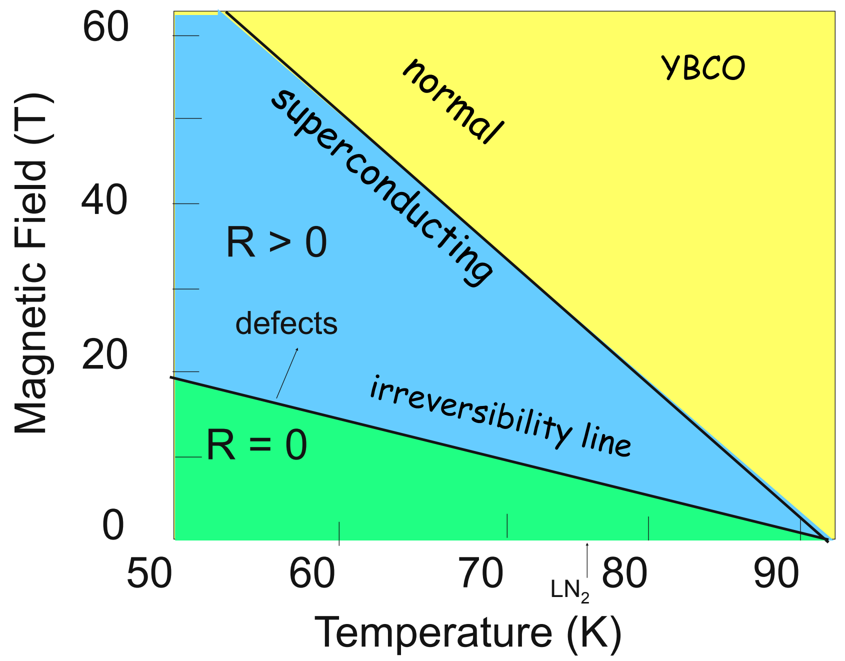
ε < 0 in inclusion



Vortex motion and dissipation

oscon-scidac.org





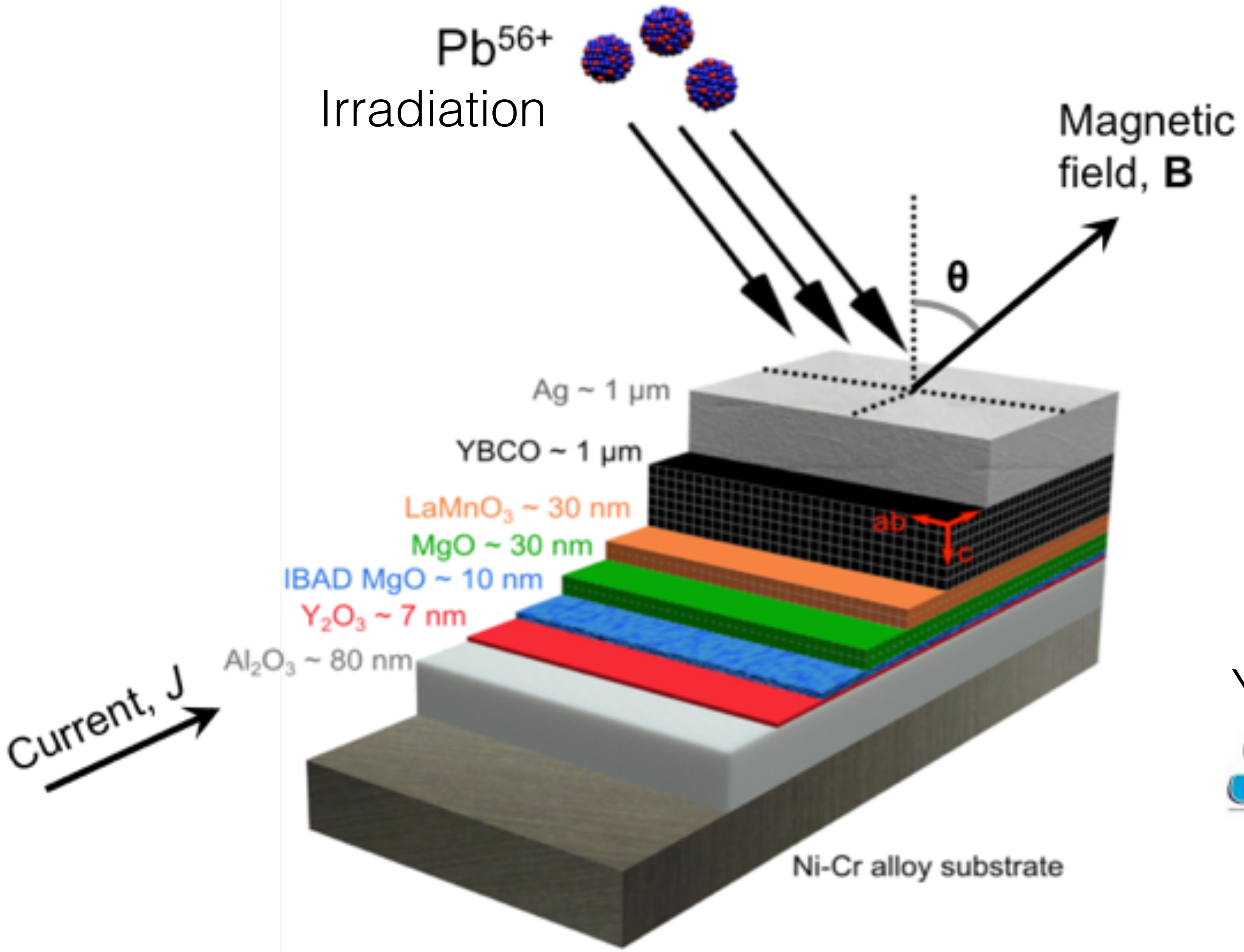
Pinning defects: nanodots, disorder, 2nd phases, dislocations, intergrowths, etc

Higher transition temperature ⇒ new materials

Higher currents ⇒ control “vortex matter”

Materials by design

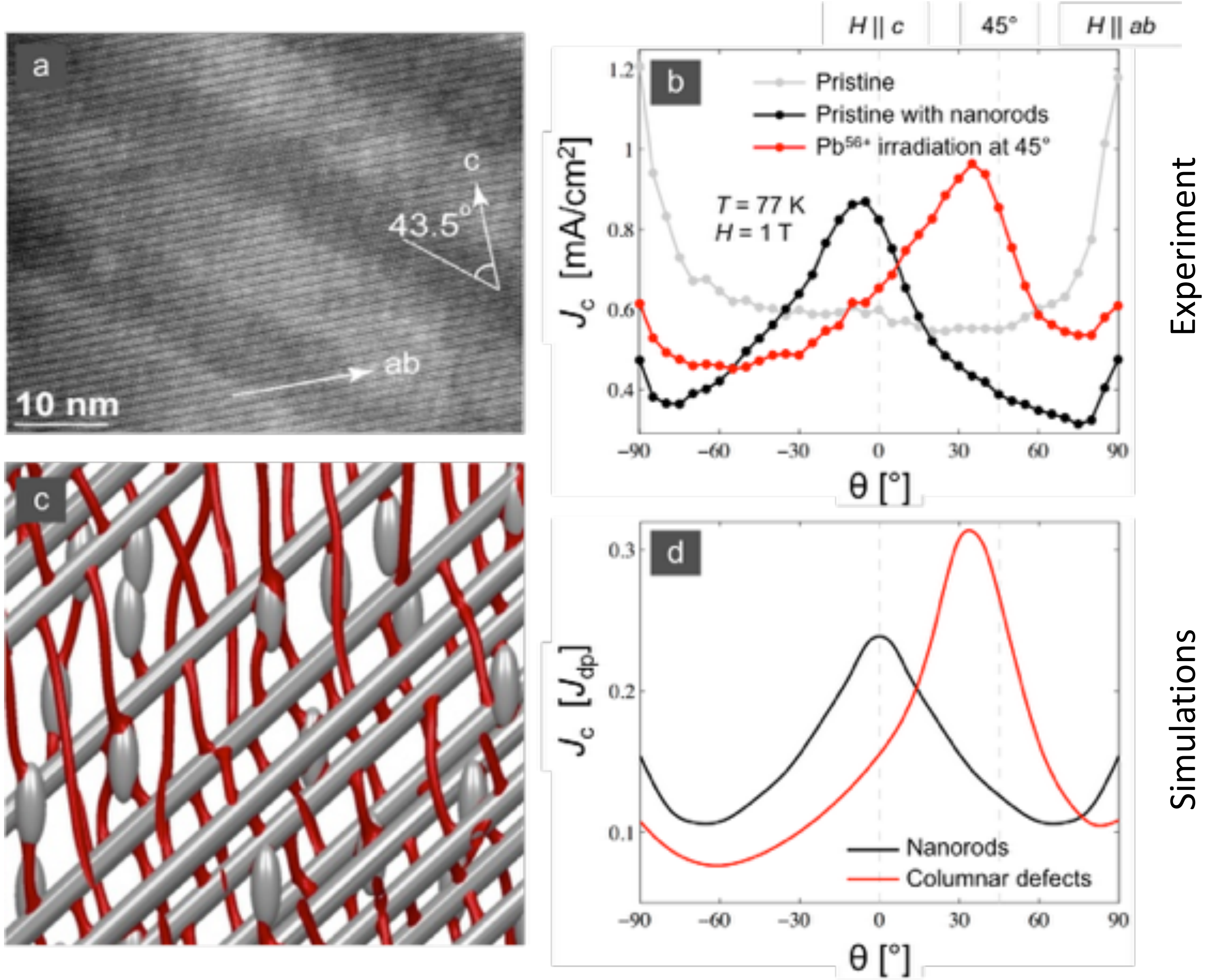
oscon-scidac.org



YBCO tape made by SuperPower Inc.
A Furukawa Company

Materials by design: Angular dependence

oscon-scidac.org

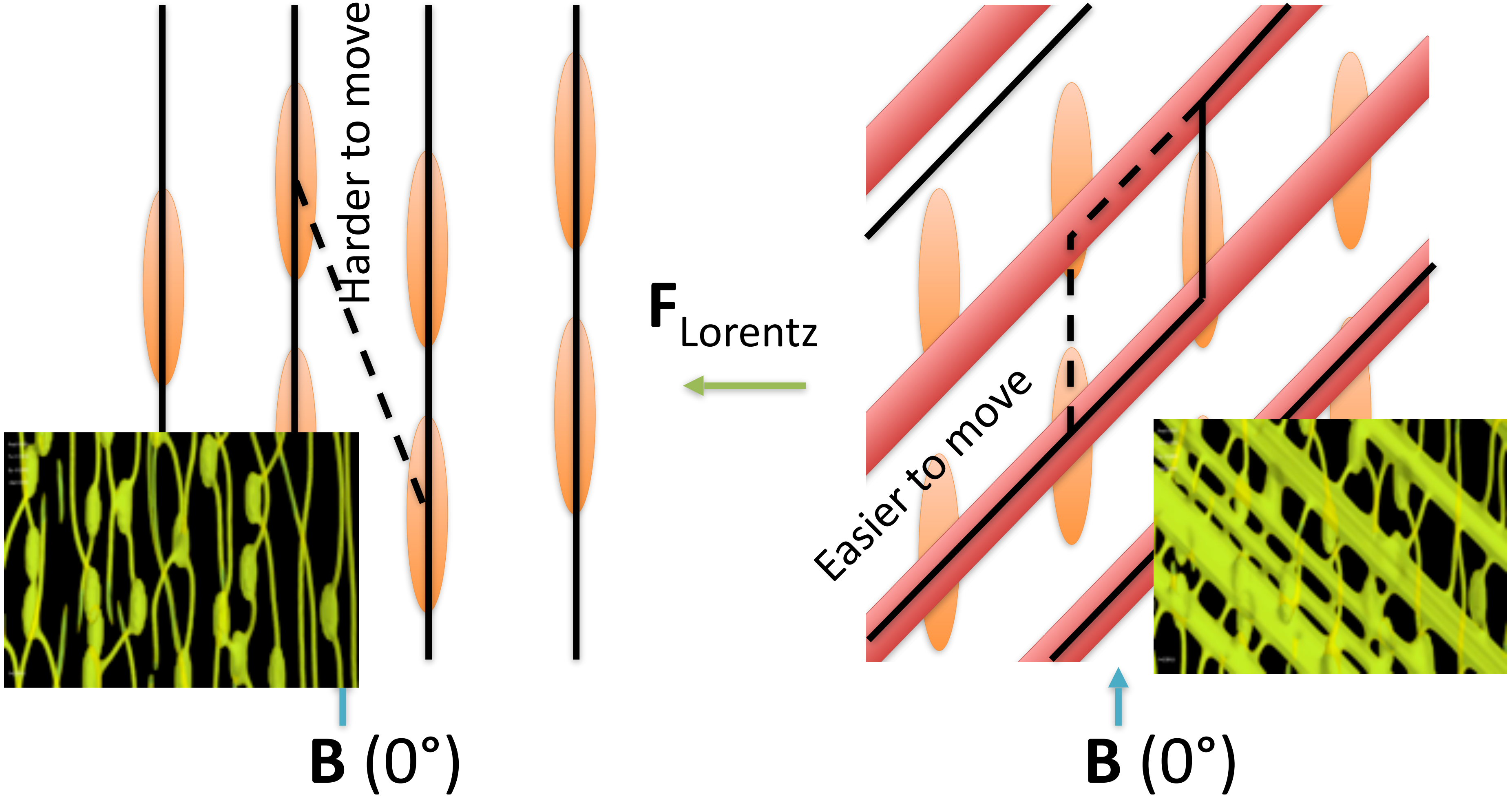


Experiment

Simulations

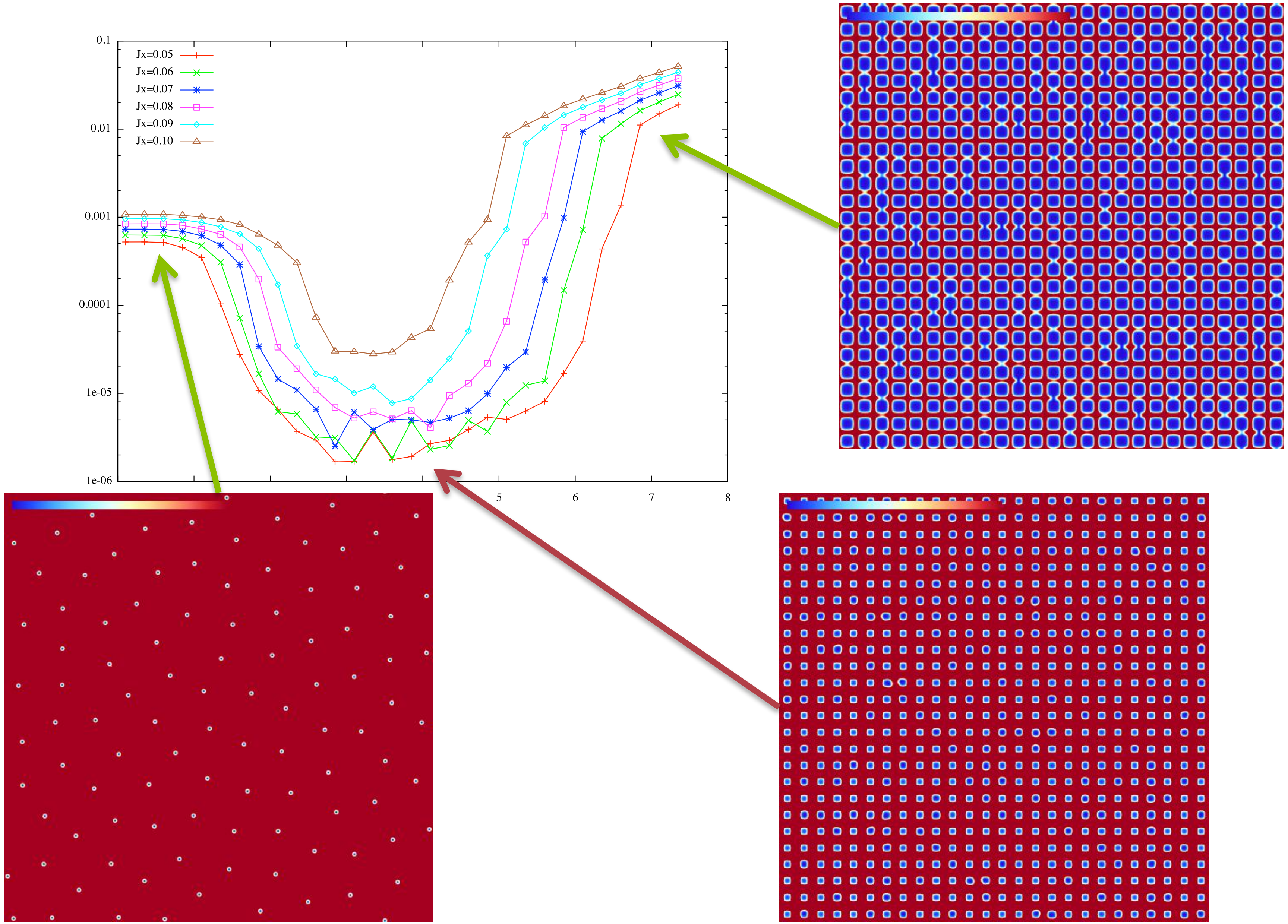
Materials by design:
Strong non-additivity of defects

oscon-scidac.org



Columnar defects works like a shortcuts for magnetic vortices

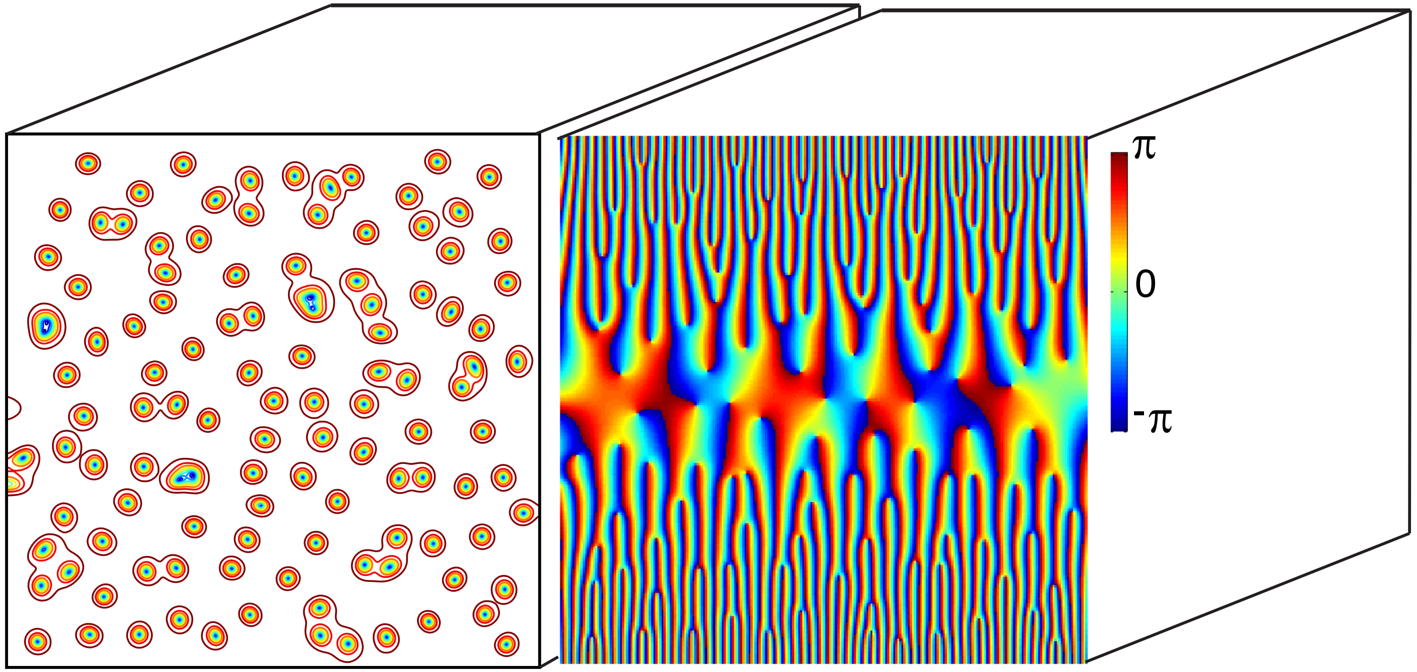
Interpreting voltage curves



Identifying and visualizing vortices

Output of Ginzburg-Landau Simulation:

complex scalar defined over mesh

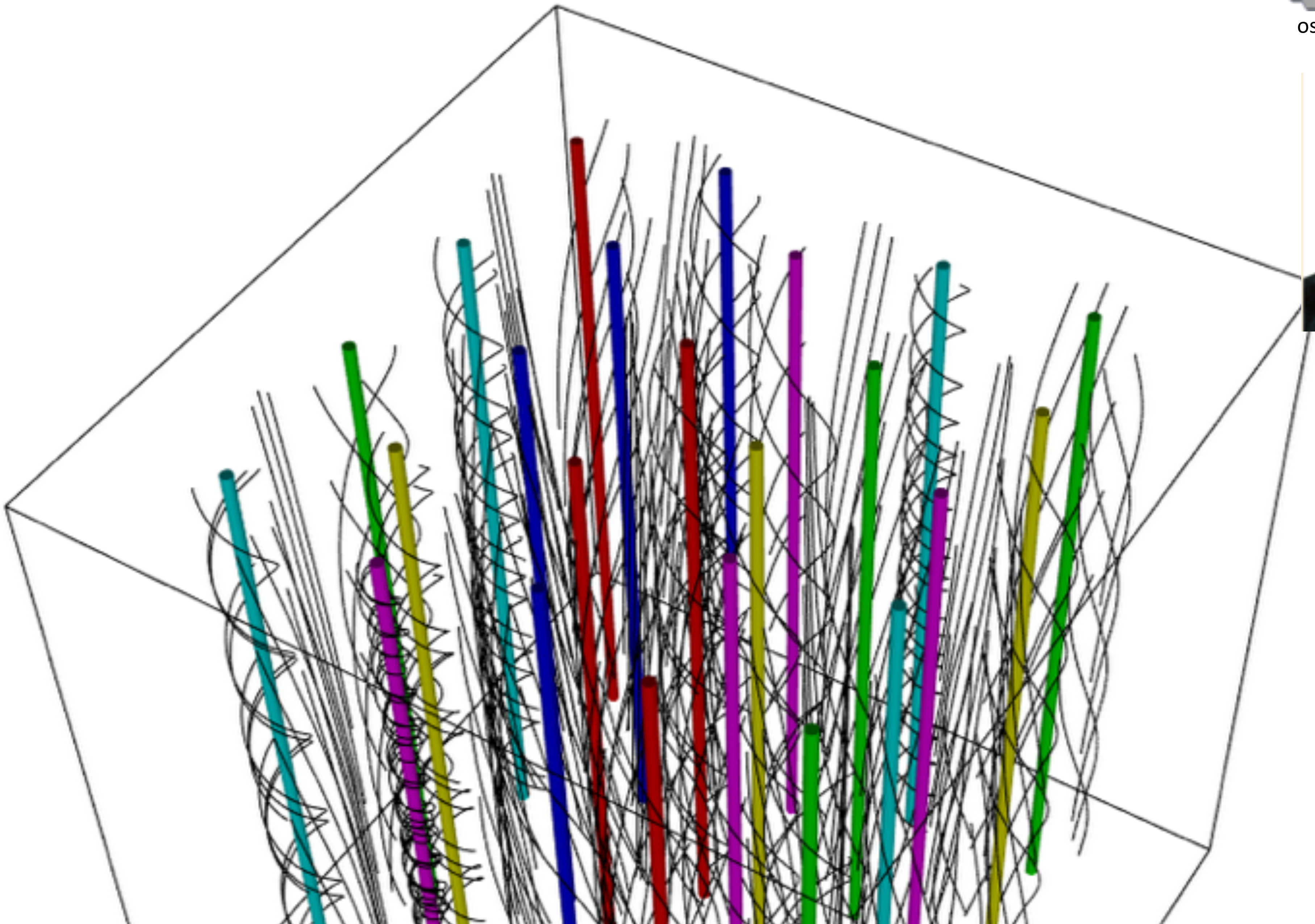


Amplitude

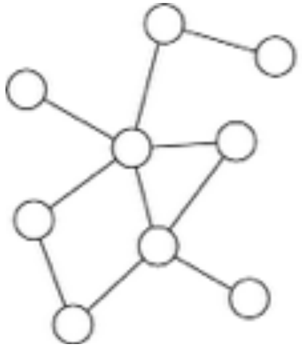
Phase Field

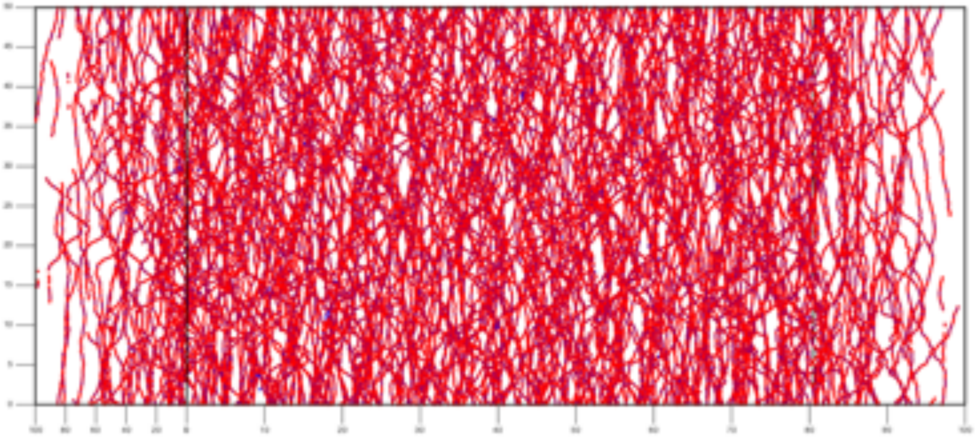
Determining how super currents flow through the material

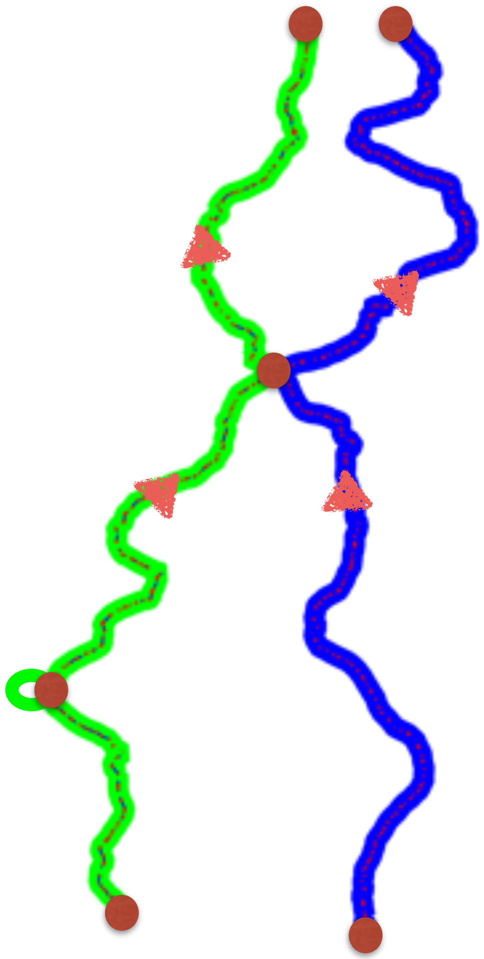
oscon-scidac.org



Graph Analysis

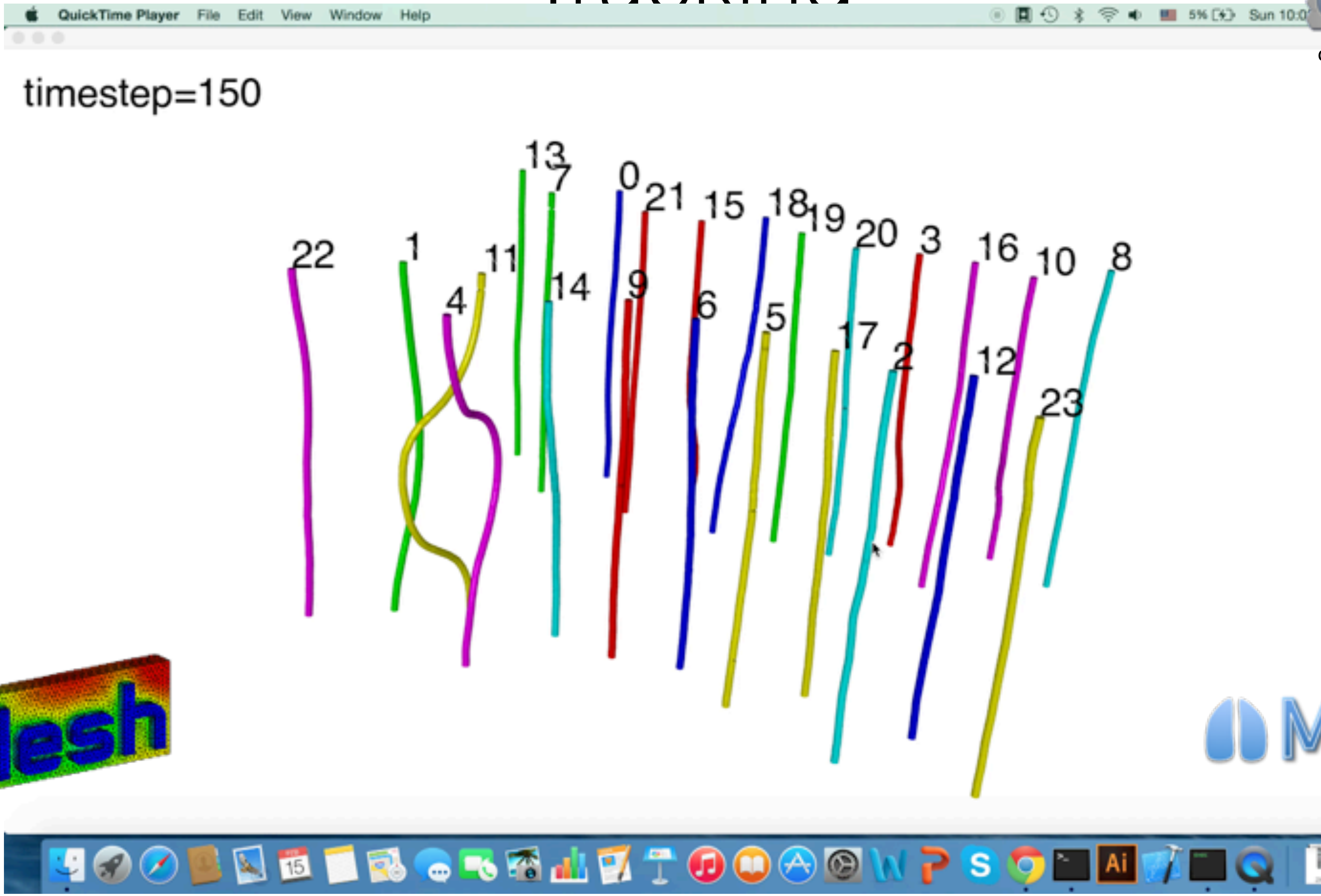






- Disentangle vortices
- Remove tiny (unstable) loops

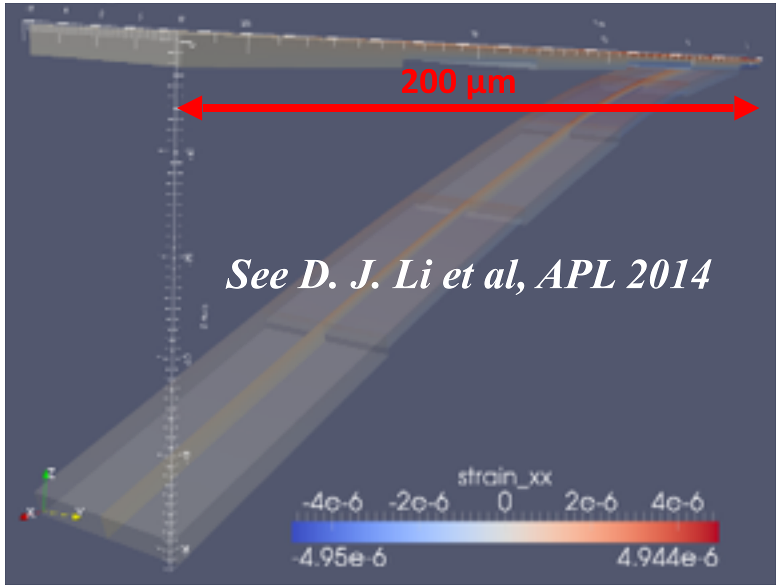
Tracking



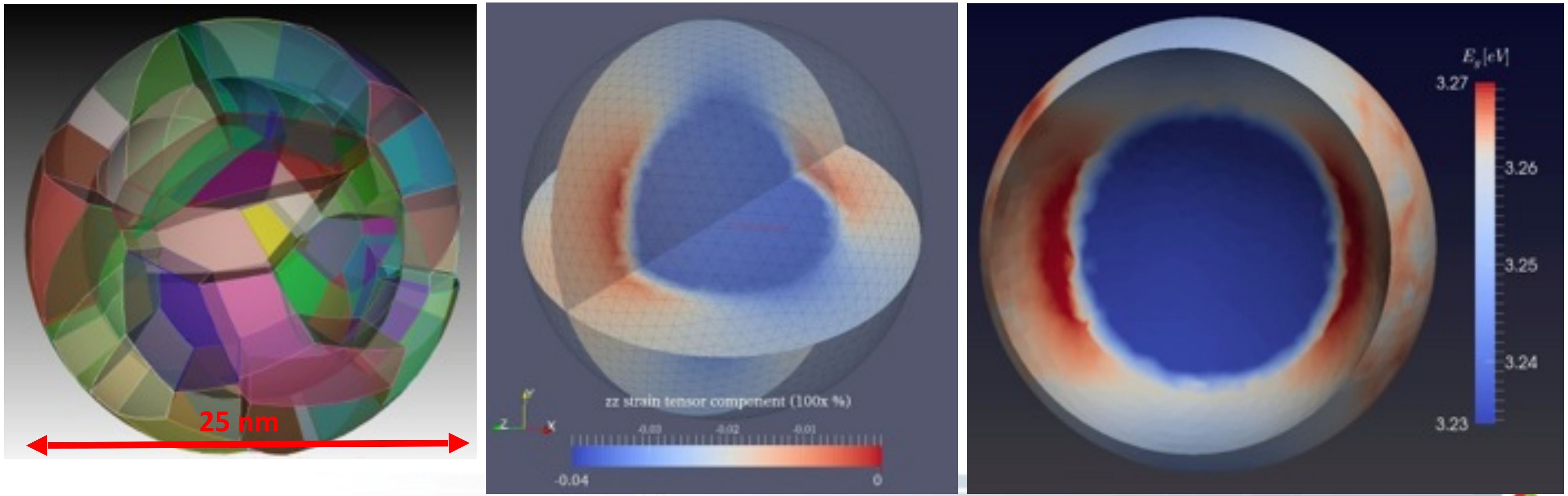
oscon-scidac.org

Coupled phasefield models of solid state materials

Energy Harvesters
(with Seungbum Hong, ANL)

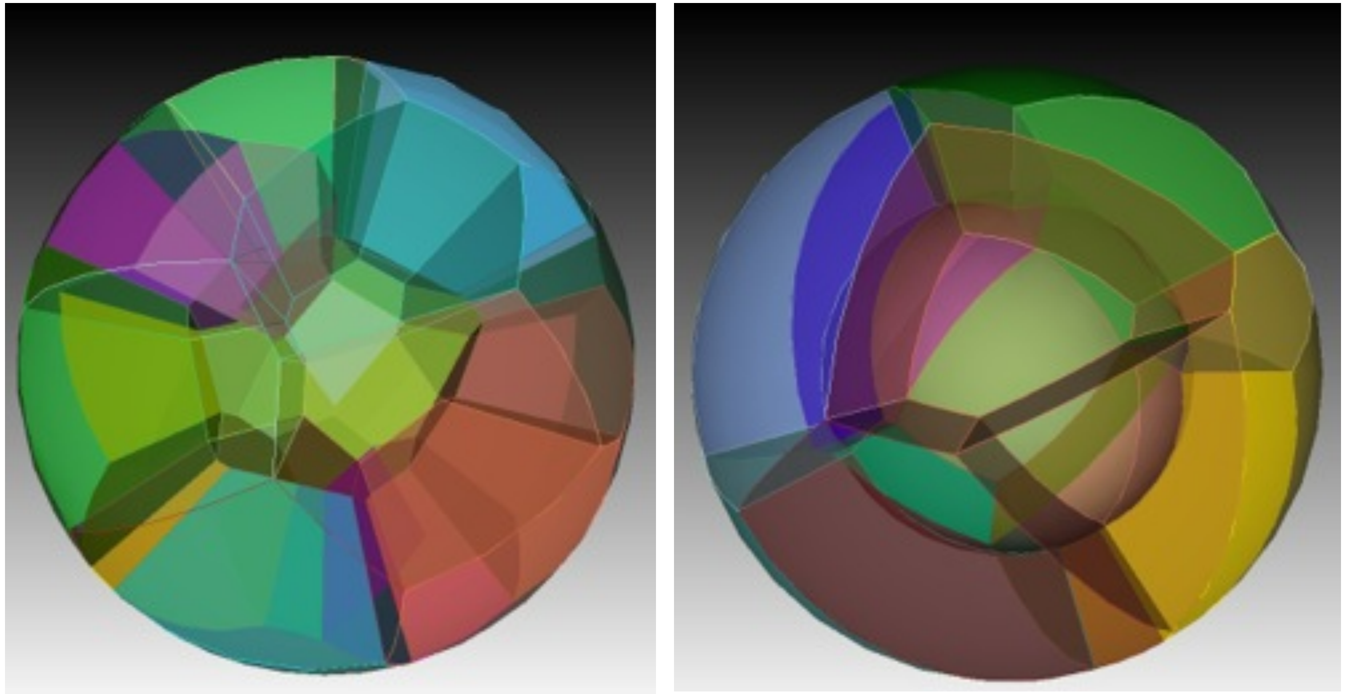


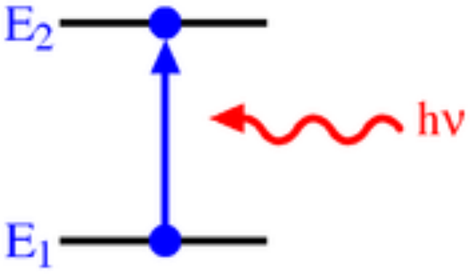
Core-Shell Nanoparticles: from Structure to Elastic Fields to Optical Properties



Core-Shell Nanoparticles: Structure

- Composite nanoparticles (metal-semiconductor, semiconductor-semiconductor)
- Here, **ZnO**/**TiO$_2$** and **Zn**/**ZnO** ~25 nm outer diameter
- Potentially useful for photovoltaics (solar absorption)



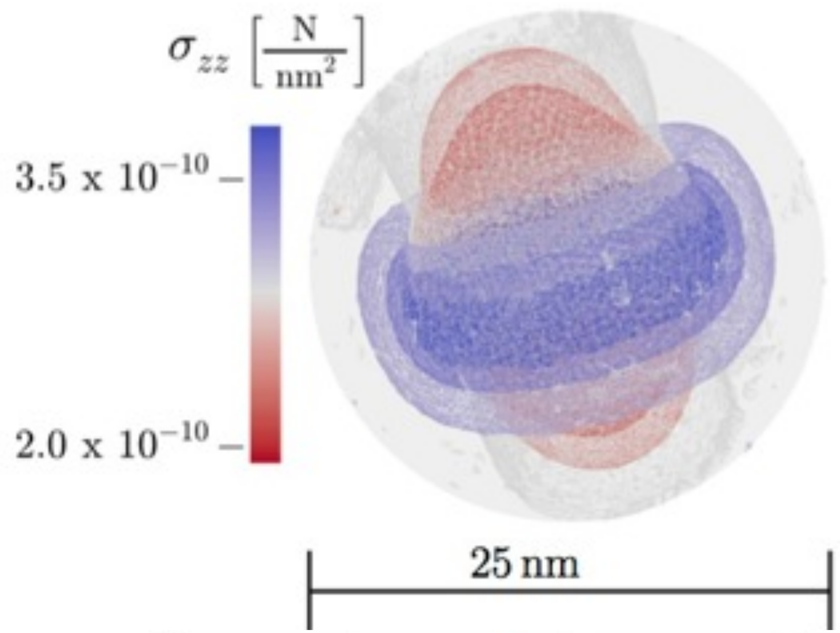


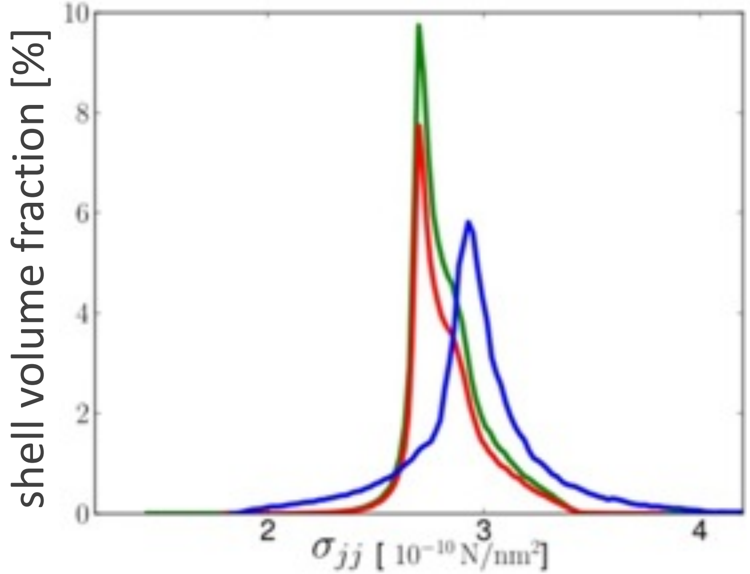
Bandgap = E_2 - E_1 depends on strain

- *What is the strain/stress in a core-shell nanoparticle (bulk and surface)?*
- *How do we relate the stress to the band gap and absorption spectrum?*
- *Can we tune the absorption spectrum by tuning the stress?*

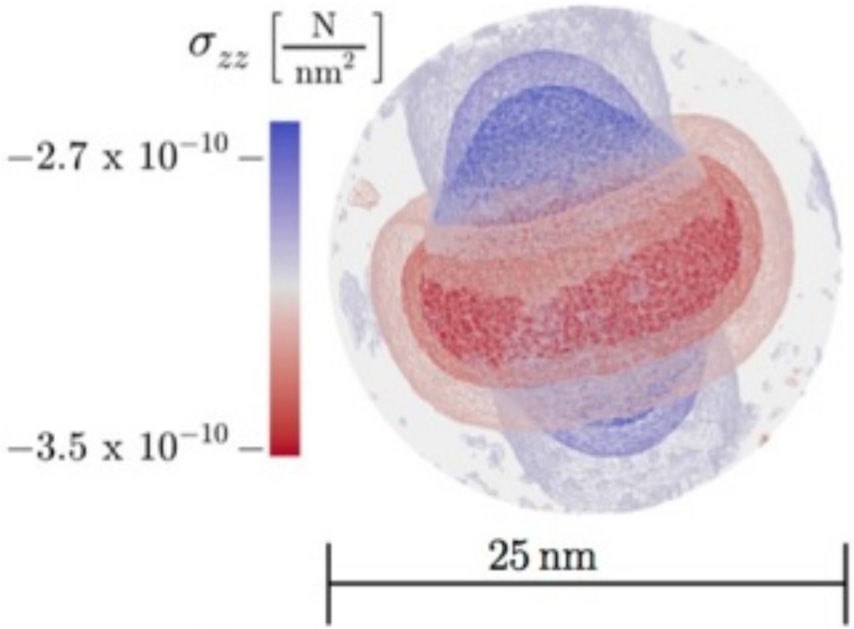
Stress Fields in Core-Shell Nanoparticles

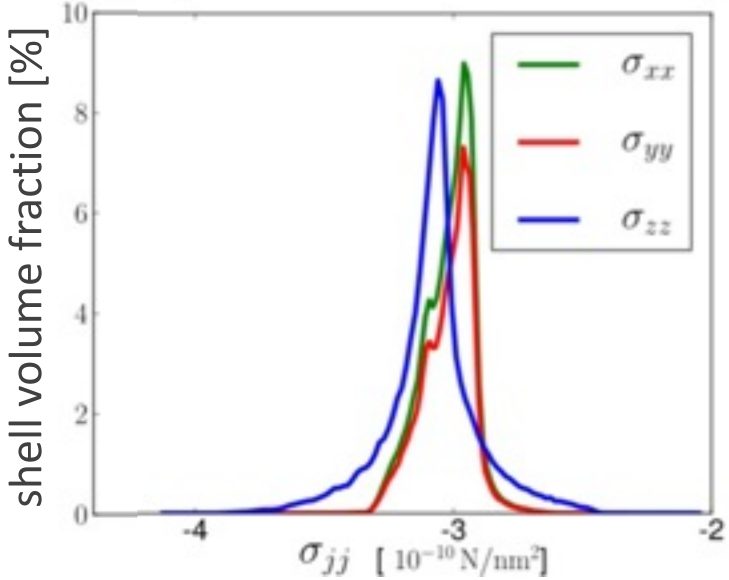
Spherical Zn core (hexagonal)
Monocrystalline ZnO shell (almost isotropic)





Spherical ZnO core (almost isotropic)
Monocrystalline TiO_2 shell (rutile, tetragonal)





- Hides software stack
- Multiple hardware resources
    - Amazon
    - Remote clusters
    - *Mira*
    - *GPU clusters*
- Integrate multiple tools
    - Workflows
    - Batch runs
    - Parameter sweeps
    - Sensitivity/UQ
    - Postprocessing
    - Visualization
- Data management
    - Archive
    - Share
    - Publish
- Focus on configuration of kernels into a usable app