Large Scale FFT-Based Stress-Strain Simulations with Irregular Domain Decomposition

Anuva Kulkarni, Franz Franchetti (Advisor), Jelena Kovačević (Advisor)

Carnegie Mellon University
Electrical & Computer ENGINEERING

Approach overview

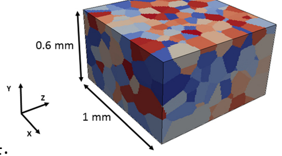
MSC-Basic Algorithm is an FFT-based algorithm [1] for calculating local stress and strain in composite materials.

3D Hooke's law

stress, strain

$$\sigma_{ij} = C_{ijkl} : \epsilon_{kl}$$

3x3x3x3 stiffness tensor

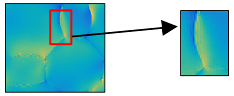


The algorithm solves a PDE:

$$C^0_{ijkl} u_{k,lj}(\mathbf{x}) + \tau_{ij,j}(\mathbf{x}) = 0$$

It requires **large amounts of memory** and has a high communication overhead which becomes a bottleneck.

Stress in crystals



Boundaries are critical regions of interest

Programmed in **FORTRAN**, difficult to run on accelerators due to memory requirement.

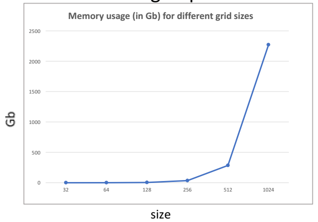
Increasing grid resolution is desirable. However, larger problem sizes must be run with parallelized code. This requires large parallel FFT computations which means **high memory usage and all-all communication**.

Problem scale:

- 3x3 stress and strain tensor at each grid point
- 9 FFTs of size N^3

Grid size: 32^3 to 1024^3

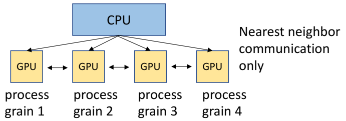
Memory requirement increases **32.4k** times!



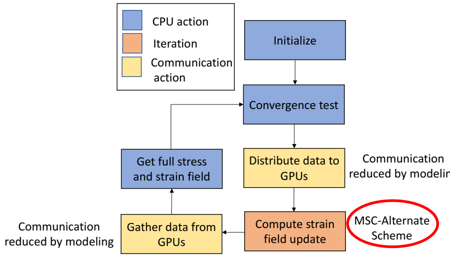
Solution:

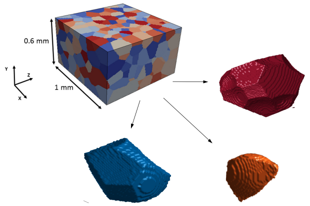
Decompose material into irregular domains, which are the grains

- memory requirement is reduced significantly
- all-all communication can be eliminated



I. MSC-Alternate Scheme





Each grain (domain) is assigned to a GPU. For small grains, single GPU can process multiple grains. Distribution will be done using appropriate load balance.

Original method by Moulinec and Suquet

Algorithm 1 MSC Basic Scheme

- **Initialize:**
  - $\epsilon^0 \leftarrow E$,
  - $\sigma^0_{mn}(\mathbf{x}) \leftarrow C_{mnkl}(\mathbf{x}) : \epsilon^0_{kl}(\mathbf{x})$
- **while** $e_s > e_{tol}$ **do**
  - $\hat{\sigma}^i_{mn}(\boldsymbol{\xi}) \leftarrow \mathrm{FFT}(\sigma^i_{mn}(\mathbf{x}))$
  - Check convergence
  - $\Delta\hat{\epsilon}^{i+1}_{kl}(\boldsymbol{\xi}) \leftarrow \hat{\Gamma}_{klmn}(\boldsymbol{\xi}) : \hat{\sigma}^i_{mn}(\boldsymbol{\xi})$
  - Update strain: $\hat{\epsilon}^{i+1}_{kl}(\boldsymbol{\xi}) \leftarrow \hat{\epsilon}^i_{kl}(\boldsymbol{\xi}) - \Delta\hat{\epsilon}^{i+1}_{kl}(\boldsymbol{\xi})$
  - $\epsilon^{i+1}_{kl}(\mathbf{x}) \leftarrow \mathrm{IFFT}(\hat{\epsilon}^{i+1}_{kl}(\boldsymbol{\xi}))$
  - Update stress: $\sigma^{i+1}_{mn}(\mathbf{x}) \leftarrow C_{mnkl}(\mathbf{x}) : \epsilon^{i+1}_{kl}(\mathbf{x})$

Our method

Algorithm 2 MSC Alternate Scheme

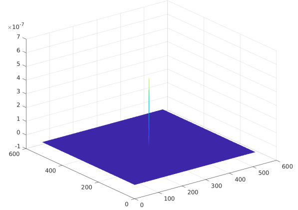
- **Initialize:**
  - $\epsilon^0 \leftarrow E$,
  - $\sigma^0_{mn}(\mathbf{x}) \leftarrow C_{mnkl}(\mathbf{x}) : \epsilon^0_{kl}(\mathbf{x})$
- **while** $e_s > e_{tol}$ **do**
  - **for** each grain $j \in G$ **do**
    - $(\hat{\sigma}^i_{mn}(\boldsymbol{\xi}))_j \leftarrow \mathrm{FFT}((\sigma^i_{mn}(\mathbf{x}))_j)$
    - Check convergence
    - Update strain: $\Delta(\hat{\epsilon}^{i+1}_{kl}(\boldsymbol{\xi}))_j \leftarrow \hat{\Gamma}_{klmn}(\boldsymbol{\xi}) : (\hat{\sigma}^i_{mn}(\boldsymbol{\xi}))_j$
    - $\Delta(\epsilon^{i+1}_{kl}(\mathbf{x}))_j \leftarrow \mathrm{IFFT}(\Delta(\hat{\epsilon}^{i+1}_{kl}(\mathbf{x}))_j)$
  - Gather step: $\Delta\epsilon^{i+1}_{kl} \leftarrow \sum_j \Delta(\epsilon^{i+1}_{kl})_j$
  - Update strain: $\epsilon^{i+1}_{kl}(\mathbf{x}) \leftarrow \epsilon^i_{kl}(\mathbf{x}) - \Delta\epsilon^{i+1}_{kl}(\mathbf{x})$
  - Update stress: $\sigma^{i+1}_{mn}(\mathbf{x}) \leftarrow C_{mnkl}(\mathbf{x}) : \epsilon^{i+1}_{kl}(\mathbf{x})$

More on the nature of Green's functions:

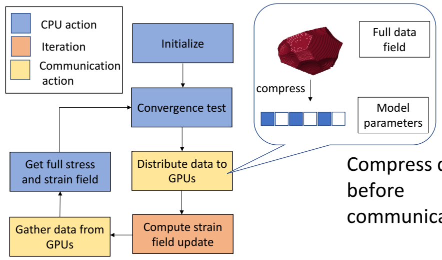
We observe that 99% energy of the space-domain Green's function is concentrated at central peak, in a n^3 volume, n<<N. Hence, Green's function can be truncated before convolution.

An Ewald-type method may be employed to avoid the 1% error, since errors accumulate in the iterative PDE solver.

Right:
N = 512
Slice of 3D component of space-domain Green's function



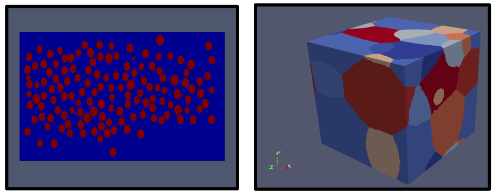
II. Data models



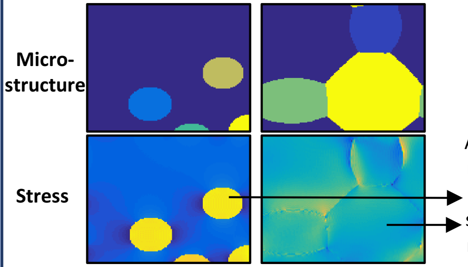
Compress data before communication

Example datasets: regular and irregular shaped grains

Dataset 1 Dataset 2

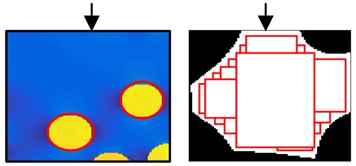


Preliminary experiment: Cropped microstructure and model in 2D plane

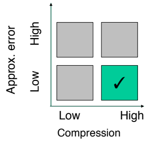


Approx. models for smooth regions

Identify regular shapes for compact representation



Ideal model characteristics



Model type I: Polynomial

$$z_k = a_0 + a_1 x_i + a_2 x_i^2 + a_3 y_j + a_4 y_j^2 + a_5 x_i y_j$$

Model type II: Spline

- Piecewise polynomial
- Cubic spline

Result highlights

Stress and strain fields calculated by MSC – Basic Scheme and MSC– Alternate Scheme are in agreement

Iter. #	% error in stress 32^3	% error in stress 64^3	% error in stress 128^3	% error in strain 32^3	% error in strain 64^3	% error in strain 128^3
1	0.013 %	0.004 %	0.002%	0.012 %	0.003 %	0.003 %
5	0.972 %	0.066 %	0.180 %	1.020 %	0.096 %	0.130 %

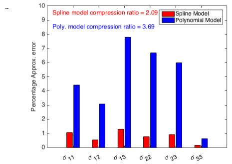
Data compression used to model grain interior can reduce communication.

Dataset 1

	Stress Component σ_{11}	σ_{22}	σ_{33}
Approx. Error %	0.099	0.1039	0.0599

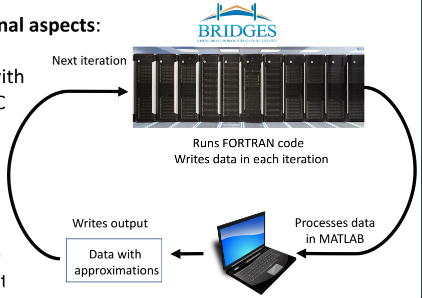
Compression ratio $C = 4.65$

Dataset 2



Computational aspects:

FORTRAN interfaced with MATLAB or C



Next Phase

- Algorit
  - Energy-preserving truncation of the Green's operator in space domain
  - Grain boundary interactions
- Performance test: GPU implementation and quantify savings in communication

Acknowledgements

The authors would like to thank Dr. Anthony Rollett, Dr. Vahid Tari and the staff at the Pittsburgh Supercomputing Center for all their assistance and collaboration with this project.

References

[1] H. Moulinec and P. Suquet. 1998. A numerical method for computing the over- all response of nonlinear composites with complex microstructure. Computer methods in applied mechanics and engineering 157, 1-2 (1998), 69–94.
[2] R. A. Lebensohn. 2001. N-site modeling of a 3D viscoplastic polycrystal using fast Fourier transform. Acta Materialia 49, 14 (2001), 2723–2737.