GPU-Accelerated String Method for Defect Annealing in Copolymer Self-Assembly

Michael Carilli[1,2], Bongkeun Kim[1], Ken-ichi Izumi[1,3], Tatsuhiro Iwama[1], Nabil Laachi[1], Kris T. Delaney[1], Glenn H. Fredrickson[1,4]
[1]Materials Research Laboratory, University of California Santa Barbara, Santa Barbara, CA, USA
[2]Department of Physics, University of California Santa Barbara, Santa Barbara, CA, USA
[3]JSR Corporation, 100 Kawajiri-cho, Yokkaichi, Mie Prefecture, Japan
[4]Department of Chemical Engineering, University of California Santa Barbara, Santa Barbara, CA, USA

Copolymer self-assembly

Diblock copolymer:
2 tethered chains of chemically distinct monomers

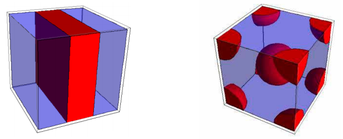
AAAAABBBBB

Block fractions f $1-f$

AA attract, BB attract, AB repel
Effective interaction parameter χ_{AB}

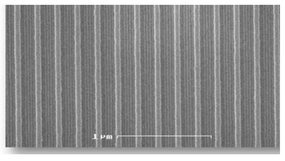
Self-assemble into mesoscale structures

Lamellae BCC Spheres



Cylinders Gyroid

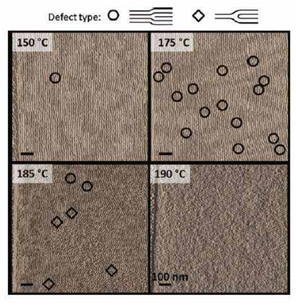
Significant, growing industrial interest



Cylindrical DSA- Courtesy of The Dow Chemical Company

- Nanolithography
- Filtration membranes

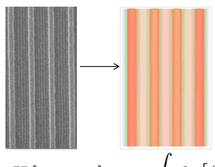
Problem: Metastable defects



Experimental cylindrical DSA
Dislocation (circle) and disclination defect (diamond) in PS-P2VP (f=0.23)
V. Mishra, G. H. Fredrickson, and E. J. Kramer, *ACS Nano* **6**, 2629 (2012)

Tool #1: Self-consistent field theory (SCFT)

Field-based model of monomer densities



$$\omega_A = i\omega_+ - \omega_- \quad , \quad \omega_B = i\omega_+ + \omega_-$$

Integrate along chain contour to compute propagator q

$$\frac{\partial}{\partial s} q(\mathbf{r}, s; [\omega_A, \omega_B]) = \frac{[b(s)]^2}{6} \nabla^2 q(\mathbf{r}, s) - \omega(\mathbf{r}, \mathbf{s}) q(\mathbf{r}, s)$$

Integrate q over space for single-chain partition fn Q

$$Q[\omega_A, \omega_B] = \frac{1}{V} \int d\mathbf{r}\, q(\mathbf{r}, N; [\omega_A, \omega_B])$$

$$H[\omega_+, \omega_-] = \rho_0 \int d\mathbf{r} \left[(1/\chi_{AB})\omega_-^2 - i\omega_+ \right] - n \ln Q[\omega_A, \omega_B]$$

Update step

$$\frac{\partial \omega_-}{\partial t} = \frac{\delta H}{\delta \omega_-}$$

$$\frac{\partial \omega_+}{\partial t} = \frac{\delta H}{\delta \omega_+}$$

Relaxes to saddle point

$$\frac{\delta H[\omega_+^*, \omega_-^*]}{\delta \omega_-} = 0$$

$$\frac{\delta H[\omega_+^*, \omega_-^*]}{\delta \omega_+} = 0$$

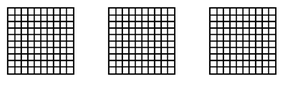
Saddle point approximation: $Z \approx Z_0\, e^{\left(-H[\omega_+^*, \omega_-^*]\right)}$

$$\rho_A(\mathbf{r}; [\omega_A, \omega_B]) = -n \frac{\delta \ln Q[\omega_A^*, \omega_B^*]}{\delta \omega_A}$$

$$\rho_B(\mathbf{r}; [\omega_A, \omega_B]) = -n \frac{\delta \ln Q[\omega_A^*, \omega_B^*]}{\delta \omega_B}$$

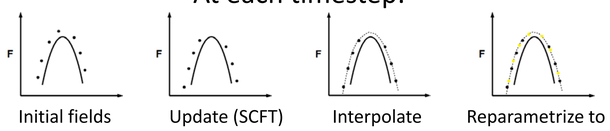
Tool #2: String Method

For barrier crossing in high-dimensional systems
Compatible with SCFT
Operates on sequence ("string") of models ("images")

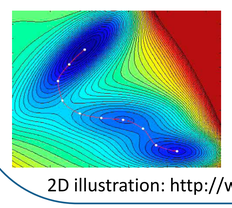


… x ~30

At each timestep:



Initial fields | Update (SCFT) | Interpolate (spline) | Reparametrize to maintain spacing



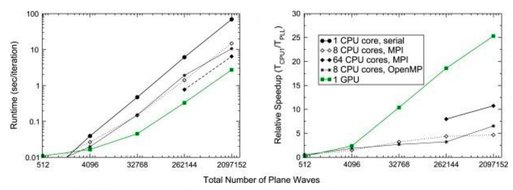
Converges to minimum energy path (MEP)
"Mountain pass:" Good estimate of transition path

2D illustration: http://www.cims.nyu.edu/~eve2/string.htm

Implementation on single GPU

SCFT Update

Relax fields with spectral methods
Involves FFTs (use cuFFT) and elementwise field arithmetic. Good for GPU!



K. T. Delaney, G. H. Fredrickson, *Computer Physics Comm.* **184**, 2102-2110 (2013)

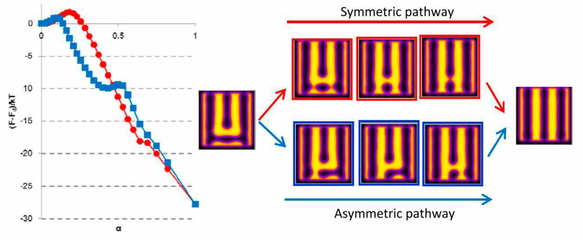
Interpolate and Reparametrize

Parallel over approx. 10^6 grid points

Each core constructs and evaluates spline for 1 grid point, compute-heavy kernel hides latency

Caveat: Update step parallel over *images,* suggests multi-GPU treatment. Stay tuned…

Results



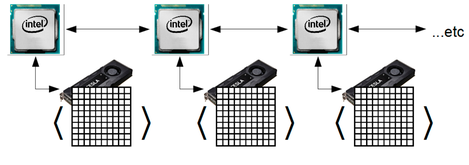
Example calculation: annealing of disclination defect for cylinders in cylindrical confinement. 2 annealing pathways found: symmetric and asymmetric.

T. Iwama, N. Laachi, K. T. Delaney, B. Kim, M. Carilli, G. H. Fredrickson, *Macromolecules* 2015, submitted

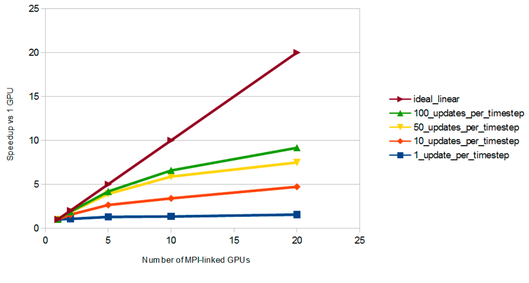
Top view:

Current work: MPI-GPU Finite Temperature String

Finds *free energy* barrier (fluctuations included)
Requires thermal average at each timestep
(100+ updates per image per reinterpolate step)



Parallelize averaging step over multiple GPUs; scaling improves as number of update steps used in thermal average increases



Strong scaling attained around 100 thermal average steps per reinterpolate step

(Scaling data collected on the TACC's Maverick)