

# Stochastic simulation of spherical growth tessellation (SGT) models based on limited microstructure data

## Approach Using a Collective Rearrangement Algorithm

Nicolas Venkovic  
nvenkov1@jhu.edu

Kirubel Teferra  
ktferr1@jhu.edu

Lori Graham-Brady  
lori@jhu.edu

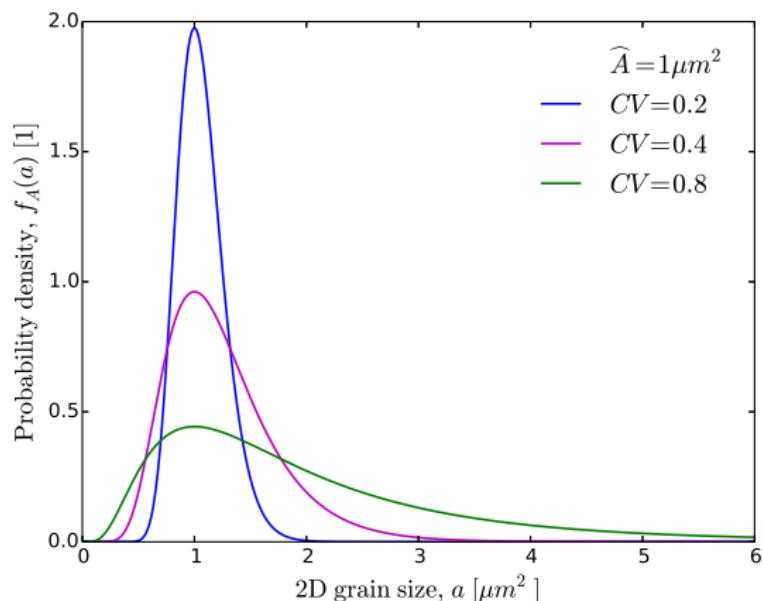
Johns Hopkins University

February 6, 2015

# Some target 2D grain size distributions

Log-normal probability density functions (PDF) of 2D grain size  $A$ :

$$f_A(a|\mu, \sigma) = \frac{1}{a\sqrt{2\pi}} \exp\left[-\left(\frac{\log(a) - \mu}{\sqrt{2\sigma^2}}\right)^2\right], \quad a \in ]0, +\infty[$$



Mean:

$$\mathbb{E}[A] = \exp[\mu + \sigma^2/2]$$

Variance:

$$\mathbb{V}[A] = [\exp(\sigma^2) - 1] \exp[2\mu + \sigma^2]$$

Mode:

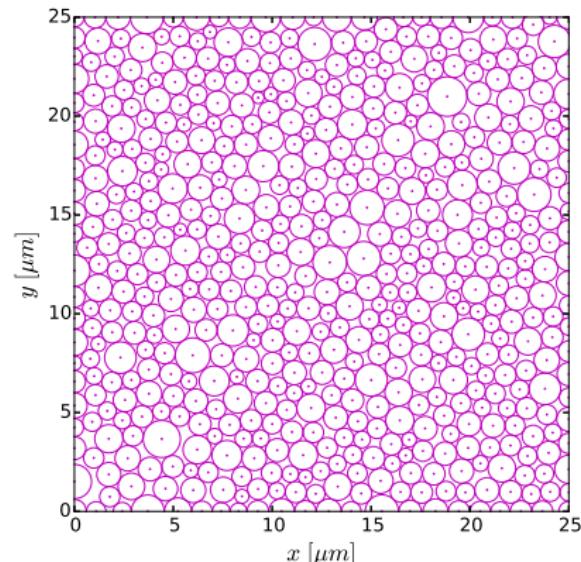
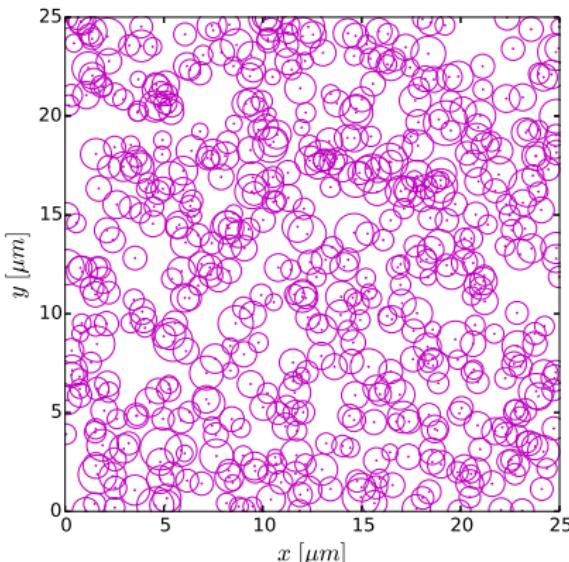
$$\widehat{A} = \exp[\mu - \sigma^2]$$

Coefficient of variation:

$$CV = \sqrt{\exp(\sigma^2) - 1}$$

## MPP simulation by collective rearrangement Force-bias algorithm, see Bezrukov et al. (2002):

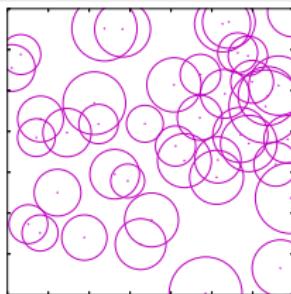
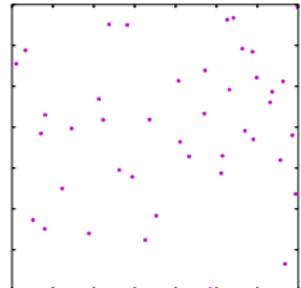
- ① Draw a realization of a Poisson point process with independent mark realizations drawn after  $f_A(a)$ .
- ② Apply these two steps until convergence:
  - ① uniformly scale all the marks to avoid any contact,
  - ② move every point after the effect of some prescribed pair potentials.



# MPP simulation by collective rearrangement

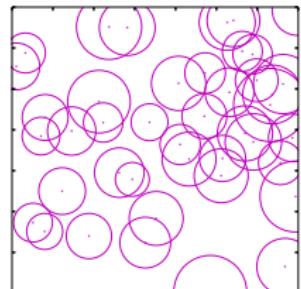
## Simulation of the point process

Simulate the point set  $\{\vec{x}_{i,0} \mid i = 1, N\}$  in the container of size  $A_{cont}$  after a Poisson point process with rate  $\lambda = 1/\mathbb{E}[A]$ .



## Simulation of the marks

Simulate the mark set  $\{r_{i,0} \mid i = 1, N\}$  independently of the points after the prescribed grain distribution  $f_A(a)$  and where  $r = \sqrt{a/\pi}$ .



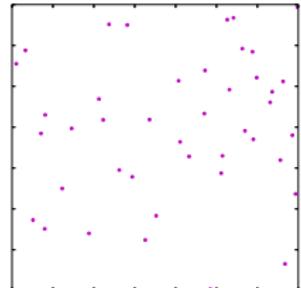
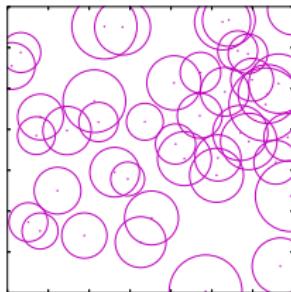
## Rearrangement of the marked point set

Assuming a pair potential  $p_{ij}$  between arbitrary marked points  $(\vec{x}_i; r_i)$  and  $(\vec{x}_j; r_j)$ , modify the marked point set using the force-biased algorithm.

# MPP simulation by collective rearrangement

## Simulation of the point process

Simulate the point set  $\{\vec{x}_{i,0} \mid i = 1, N\}$  in the container of size  $A_{cont}$  after a Poisson point process with rate  $\lambda = 1/\mathbb{E}[A]$ .

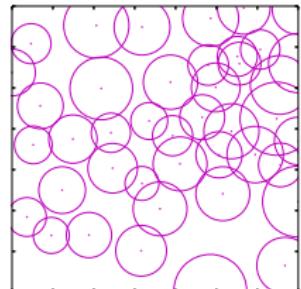


## Simulation of the marks

Simulate the mark set  $\{r_{i,0} \mid i = 1, N\}$  independently of the points after the prescribed grain distribution  $f_A(a)$  and where  $r = \sqrt{a/\pi}$ .

## Rearrangement of the marked point set

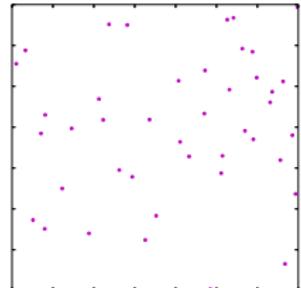
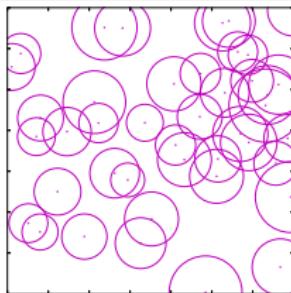
Assuming a pair potential  $p_{ij}$  between arbitrary marked points  $(\vec{x}_i; r_i)$  and  $(\vec{x}_j; r_j)$ , modify the marked point set using the force-biased algorithm.



# MPP simulation by collective rearrangement

## Simulation of the point process

Simulate the point set  $\{\vec{x}_{i,0} \mid i = 1, N\}$  in the container of size  $A_{cont}$  after a Poisson point process with rate  $\lambda = 1/\mathbb{E}[A]$ .

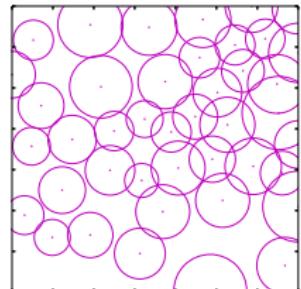


## Simulation of the marks

Simulate the mark set  $\{r_{i,0} \mid i = 1, N\}$  independently of the points after the prescribed grain distribution  $f_A(a)$  and where  $r = \sqrt{a/\pi}$ .

## Rearrangement of the marked point set

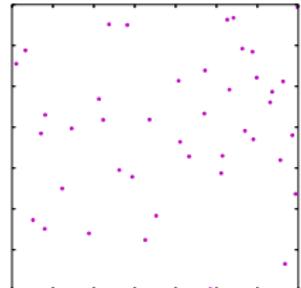
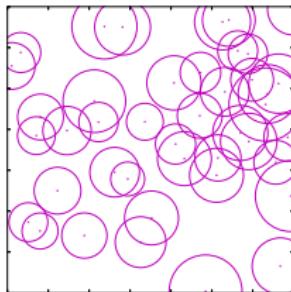
Assuming a pair potential  $p_{ij}$  between arbitrary marked points  $(\vec{x}_i; r_i)$  and  $(\vec{x}_j; r_j)$ , modify the marked point set using the force-biased algorithm.



# MPP simulation by collective rearrangement

## Simulation of the point process

Simulate the point set  $\{\vec{x}_{i,0} \mid i = 1, N\}$  in the container of size  $A_{cont}$  after a Poisson point process with rate  $\lambda = 1/\mathbb{E}[A]$ .

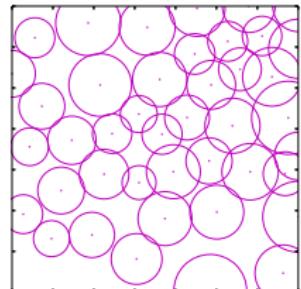


## Simulation of the marks

Simulate the mark set  $\{r_{i,0} \mid i = 1, N\}$  independently of the points after the prescribed grain distribution  $f_A(a)$  and where  $r = \sqrt{a/\pi}$ .

## Rearrangement of the marked point set

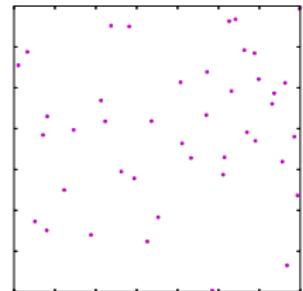
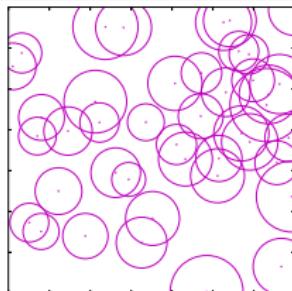
Assuming a pair potential  $p_{ij}$  between arbitrary marked points  $(\vec{x}_i; r_i)$  and  $(\vec{x}_j; r_j)$ , modify the marked point set using the force-biased algorithm.



# MPP simulation by collective rearrangement

## Simulation of the point process

Simulate the point set  $\{\vec{x}_{i,0} \mid i = 1, N\}$  in the container of size  $A_{cont}$  after a Poisson point process with rate  $\lambda = 1/\mathbb{E}[A]$ .

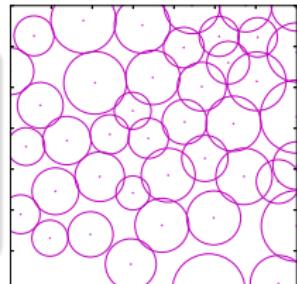


## Simulation of the marks

Simulate the mark set  $\{r_{i,0} \mid i = 1, N\}$  independently of the points after the prescribed grain distribution  $f_A(a)$  and where  $r = \sqrt{a/\pi}$ .

## Rearrangement of the marked point set

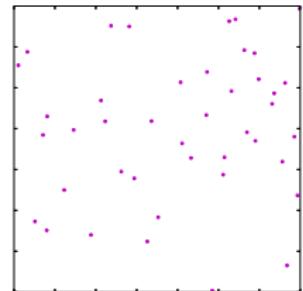
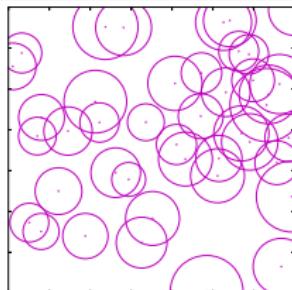
Assuming a pair potential  $p_{ij}$  between arbitrary marked points  $(\vec{x}_i; r_i)$  and  $(\vec{x}_j; r_j)$ , modify the marked point set using the force-biased algorithm.



# MPP simulation by collective rearrangement

## Simulation of the point process

Simulate the point set  $\{\vec{x}_{i,0} \mid i = 1, N\}$  in the container of size  $A_{cont}$  after a Poisson point process with rate  $\lambda = 1/\mathbb{E}[A]$ .

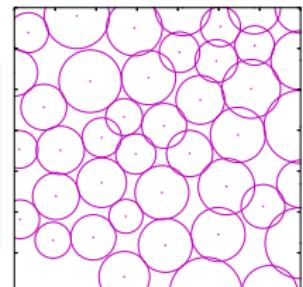


## Simulation of the marks

Simulate the mark set  $\{r_{i,0} \mid i = 1, N\}$  independently of the points after the prescribed grain distribution  $f_A(a)$  and where  $r = \sqrt{a/\pi}$ .

## Rearrangement of the marked point set

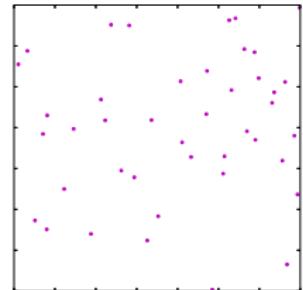
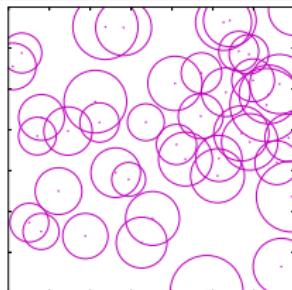
Assuming a pair potential  $p_{ij}$  between arbitrary marked points  $(\vec{x}_i; r_i)$  and  $(\vec{x}_j; r_j)$ , modify the marked point set using the force-biased algorithm.



# MPP simulation by collective rearrangement

## Simulation of the point process

Simulate the point set  $\{\vec{x}_{i,0} \mid i = 1, N\}$  in the container of size  $A_{cont}$  after a Poisson point process with rate  $\lambda = 1/\mathbb{E}[A]$ .

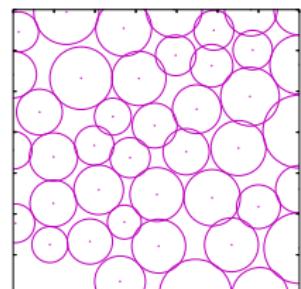


## Simulation of the marks

Simulate the mark set  $\{r_{i,0} \mid i = 1, N\}$  independently of the points after the prescribed grain distribution  $f_A(a)$  and where  $r = \sqrt{a/\pi}$ .

## Rearrangement of the marked point set

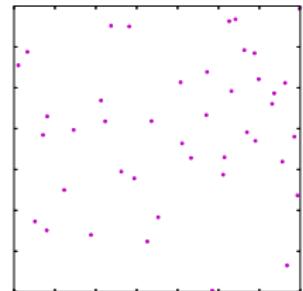
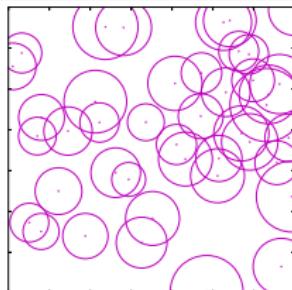
Assuming a pair potential  $p_{ij}$  between arbitrary marked points  $(\vec{x}_i; r_i)$  and  $(\vec{x}_j; r_j)$ , modify the marked point set using the force-biased algorithm.



# MPP simulation by collective rearrangement

## Simulation of the point process

Simulate the point set  $\{\vec{x}_{i,0} \mid i = 1, N\}$  in the container of size  $A_{cont}$  after a Poisson point process with rate  $\lambda = 1/\mathbb{E}[A]$ .

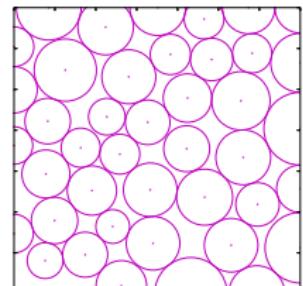


## Simulation of the marks

Simulate the mark set  $\{r_{i,0} \mid i = 1, N\}$  independently of the points after the prescribed grain distribution  $f_A(a)$  and where  $r = \sqrt{a/\pi}$ .

## Rearrangement of the marked point set

Assuming a pair potential  $p_{ij}$  between arbitrary marked points  $(\vec{x}_i; r_i)$  and  $(\vec{x}_j; r_j)$ , modify the marked point set using the force-biased algorithm.



## 2D adaptation of the force-biased algorithm

1. for  $i = 1, N$ :
2.      $(\vec{x}_i; r_i) = (\vec{x}_{i,0}; r_{i,0})$
3.  $\eta = \min_{i \neq j} \{ \| \vec{x}_j - \vec{x}_i \| / (r_i + r_j) \}$
4.  $\rho = 1$
5. while  $\eta < 1$ :
6.     for  $i = 1, N$ :
7.         for  $j \neq i$ :
8.              $\vec{n}_{ij} = (\vec{x}_j - \vec{x}_i) / \| \vec{x}_j - \vec{x}_i \|$
9.              $p_{ij} = r_i r_j \left[ \frac{\| \vec{x}_j - \vec{x}_i \|^2}{(r_i + r_j)^2} - 1 \right]$
10.             $\vec{F}_{ij} = \varphi \mathbf{1}_{ij} p_{ij} \vec{n}_{ij}$
11.             $\vec{x}_i = \vec{x}_i + \frac{1}{r_i} \sum_{j \neq i} \vec{F}_{ij}$
12.      $A_{nom} = \sum_{i=1}^N \pi r_i^2 / A_{cont}$
13.      $\delta = -\log_{10} \lfloor (1 - \eta^2) A_{nom} \rfloor$
14.      $\rho = \rho - 2^{-\delta} / \tau$
15.     for  $i = 1, N$ :
16.          $r_i = \rho r_{i,0}$
17.      $\eta = \min_{i \neq j} \{ \| \vec{x}_j - \vec{x}_i \| / (r_i + r_j) \}$

Size of the container:

$$A_{cont}$$

Indicator function:

$$\mathbf{1}_{ij} = \begin{cases} 1 & \text{if } b(\vec{x}_i, r_i) \cap b(\vec{x}_j, r_j) \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$$

Contraction rate:

$$\tau \approx 3000 - 100000$$

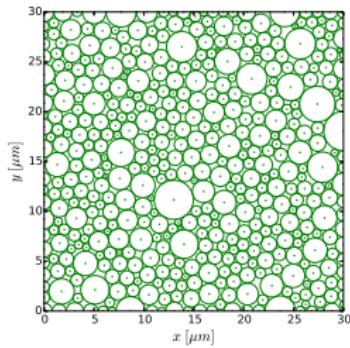
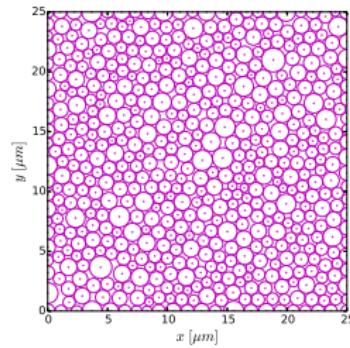
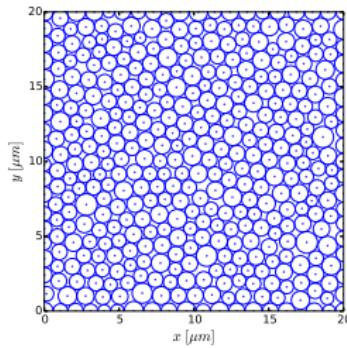
Force parameter:

$$\varphi \approx 0.3 - 0.6$$

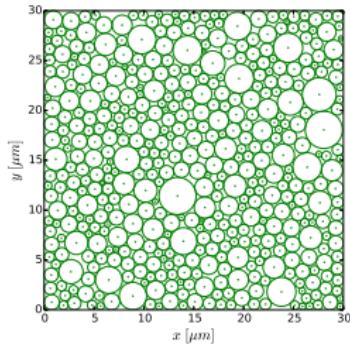
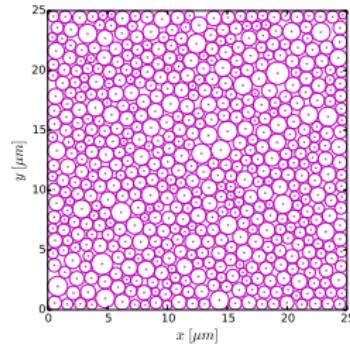
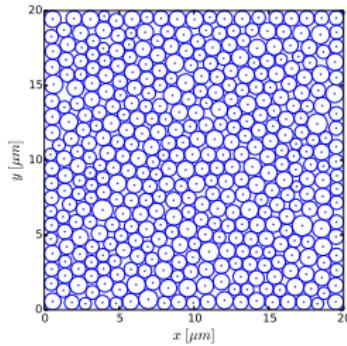
NB: Because  $\vec{F}_{ij} \propto \mathbf{1}_{ij} p_{ij} \vec{n}_{ij}$ , only repulsive potentials, i.e.  $p_{ij} < 0$ , affect the configuration of the system.

# Different ways to handle boundaries

Enforce points to be within the domain:

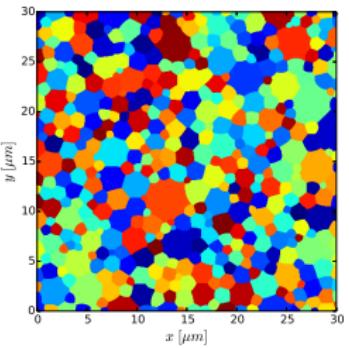
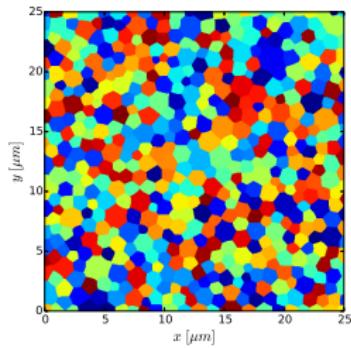
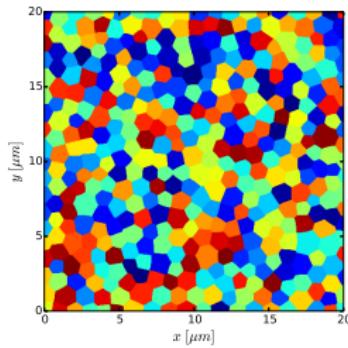


Enforce particles to be within the domain:

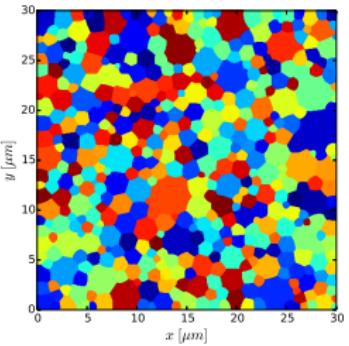
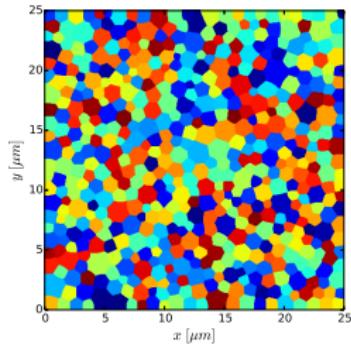
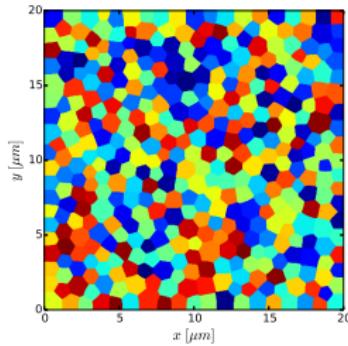


# Different ways to handle boundaries – resulting tessellations

Enforce points to be within the domain:

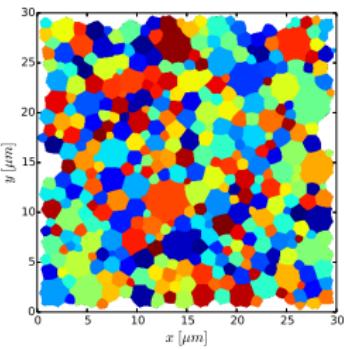
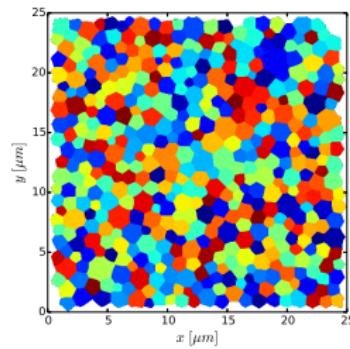
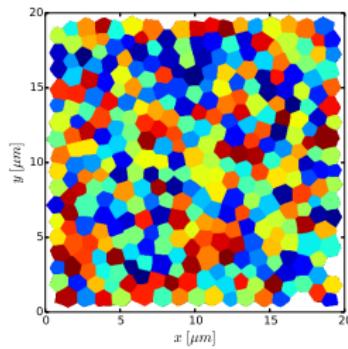


Enforce particles to be within the domain:

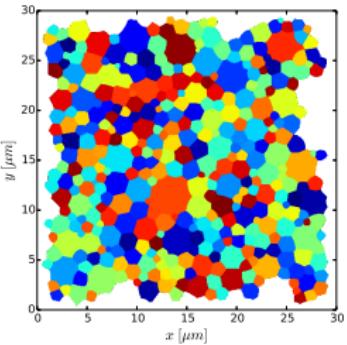
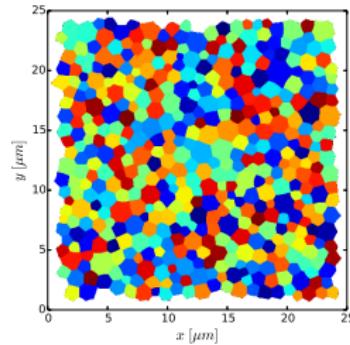
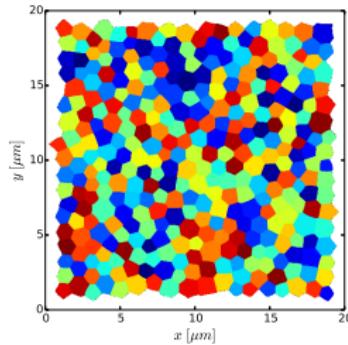


# Different ways to handle boundaries – resulting tessellations

Enforce points to be within the domain:

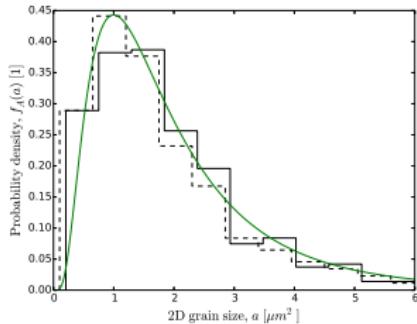
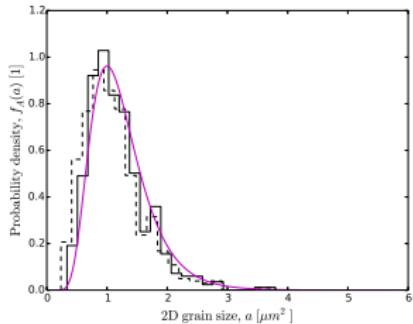
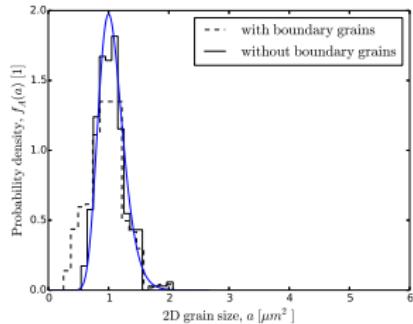


Enforce particles to be within the domain:

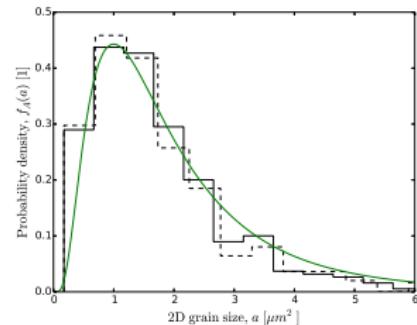
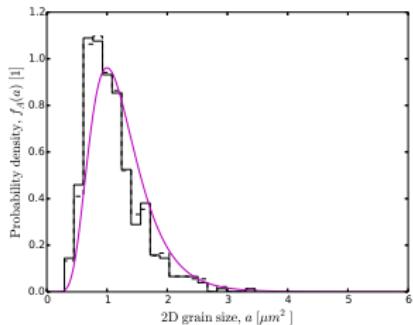
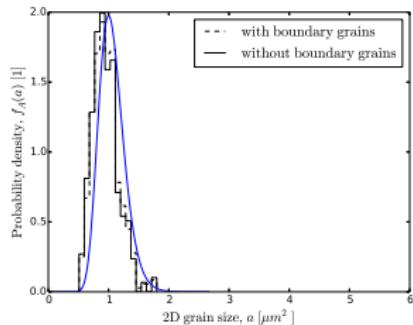


# Effect of boundary grains on the recovered grain size PDF

Enforce points to be within the domain:

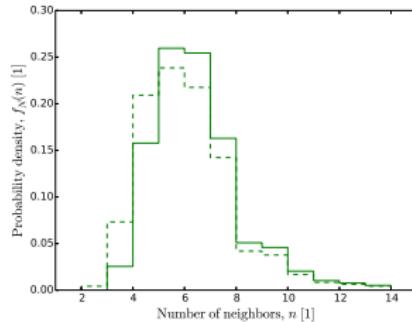
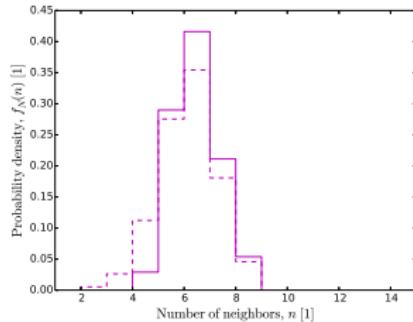
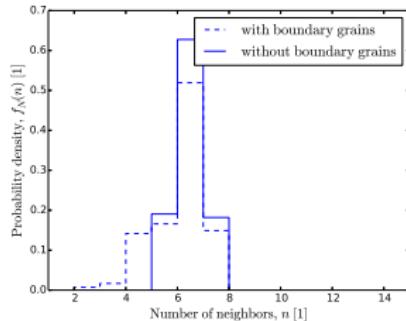


Enforce particles to be within the domain:

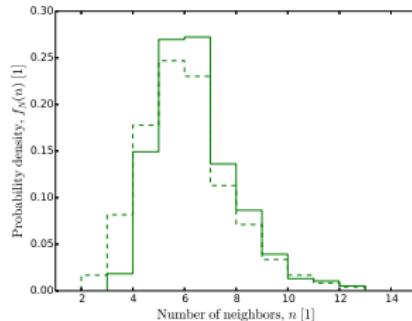
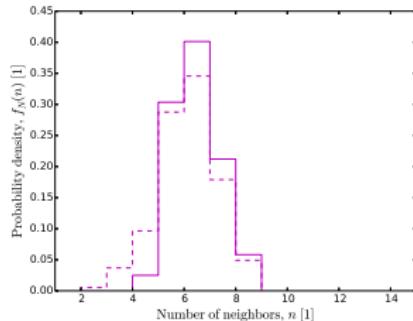
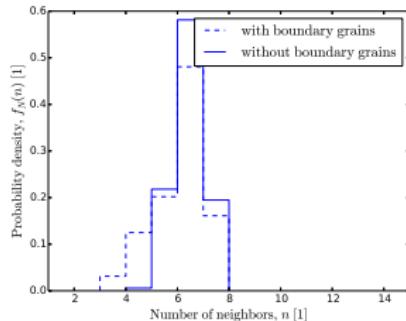


# Effect of boundary grains on the number of neighbors

Enforce points to be within the domain:



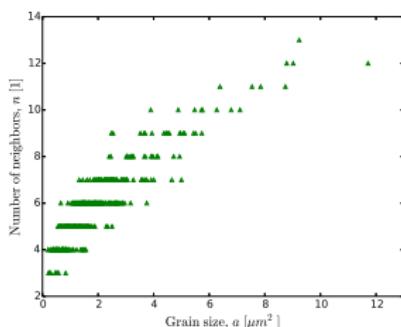
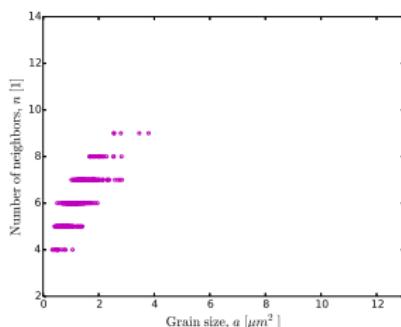
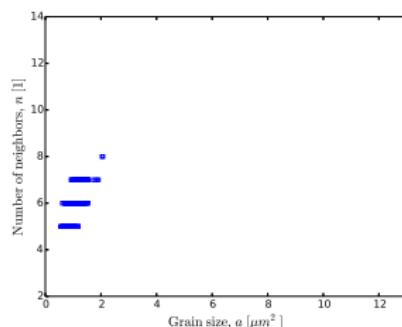
Enforce particles to be within the domain:



# Correlation between number of neighbors and grain size

Yet to investigate:

- How do these scatters evolve as a function of  $\varphi$  and  $\tau$ ?
- Can we relate the random number of neighbors to the random grain size for a prescribed  $f_A(a)$ ?



**NB:** The results presented are for the bulk grains in simulations where the points only are constrained to be within the domain.

# Computation time

## MPP simulation

- serial implementation
- convergence at  $\eta = 1$
- $\varphi = 0.5, \tau = 40000$

## SGT model resolution

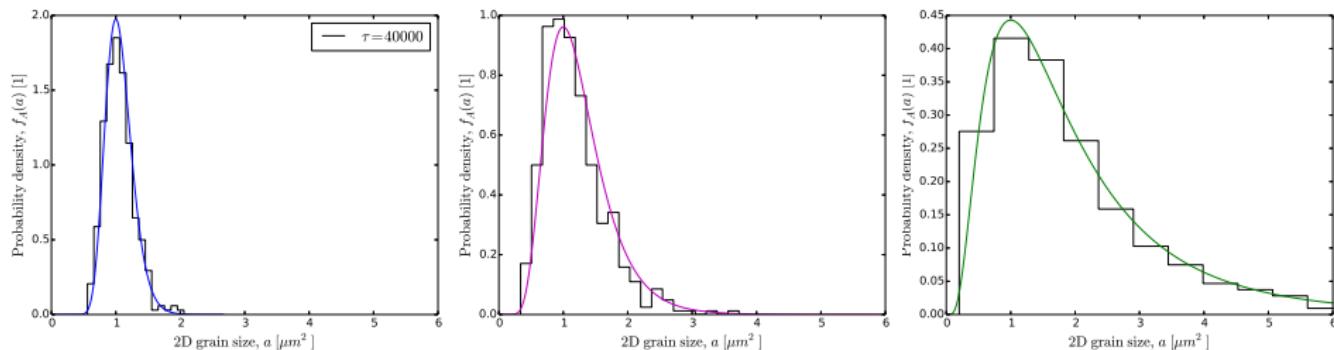
- parallel implementation: 6 threads
- $\Delta x = 0.01 \mu m$

Target, $f_A(a \hat{A} = 1 \mu m^2)$	MPP	SGT	Total
$CV = 0.1, A_{cont} = 20 \times 20 \mu m^2$	47 s (5950 it.)	29 s	76 s
$CV = 0.4, A_{cont} = 25 \times 25 \mu m^2$	98 s (6172 it.)	63 s	161 s
$CV = 0.8, A_{cont} = 30 \times 30 \mu m^2$	65 s (5596 it.)	90 s	155 s

**NB:** The cpu time of MPP simulations is very sensitive to the contraction parameter  $\tau$ . For  $CV = 0.4$ ,  $\tau = 40000$  yields convergence after 98 s,  $\tau = 4000$  after 12 s and  $\tau = 400$  after 2 s.

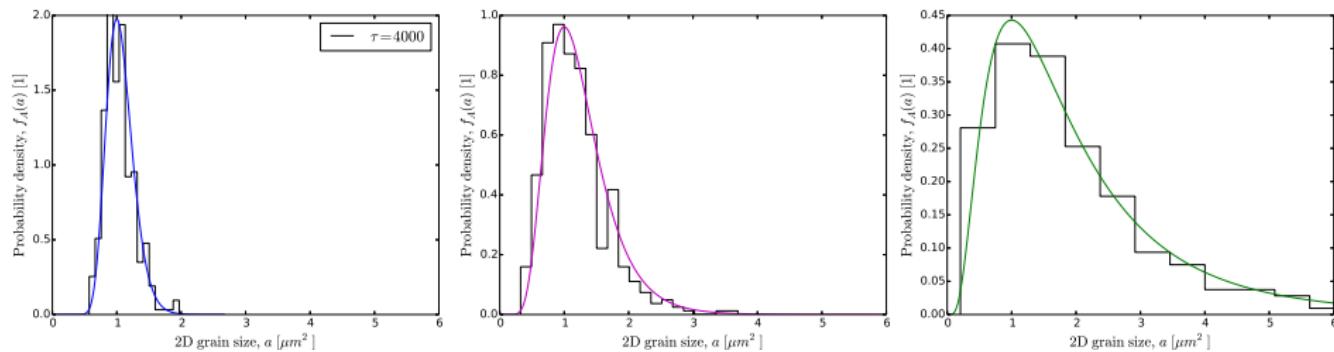
# Effect of the contraction parameter parameter on tessellation summary statistics

2D grain size distribution:



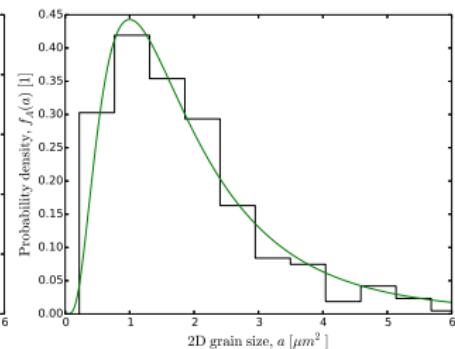
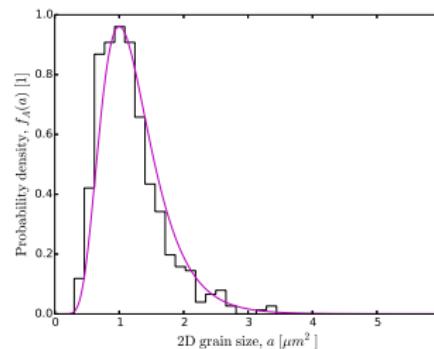
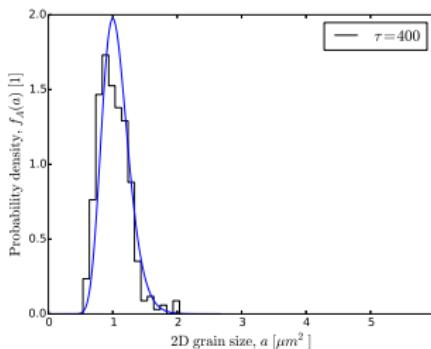
# Effect of the contraction parameter parameter on tessellation summary statistics

2D grain size distribution:



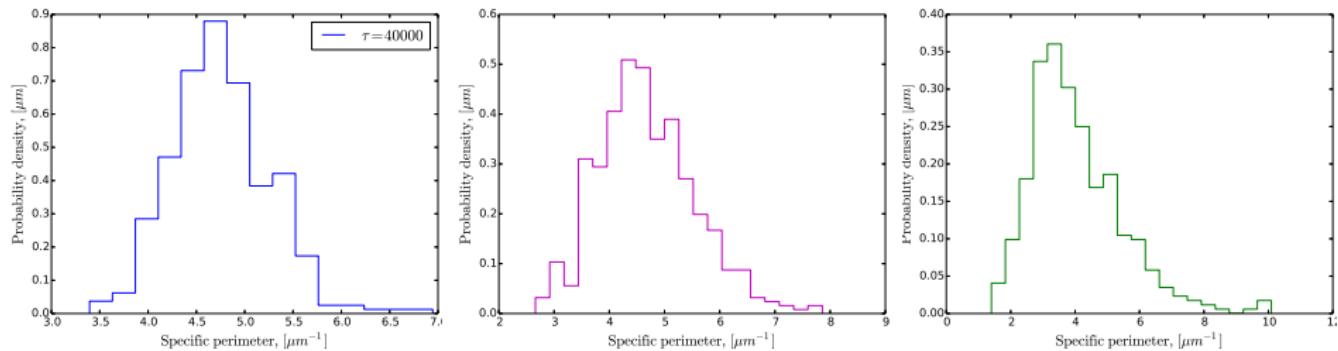
# Effect of the contraction parameter parameter on tessellation summary statistics

2D grain size distribution:



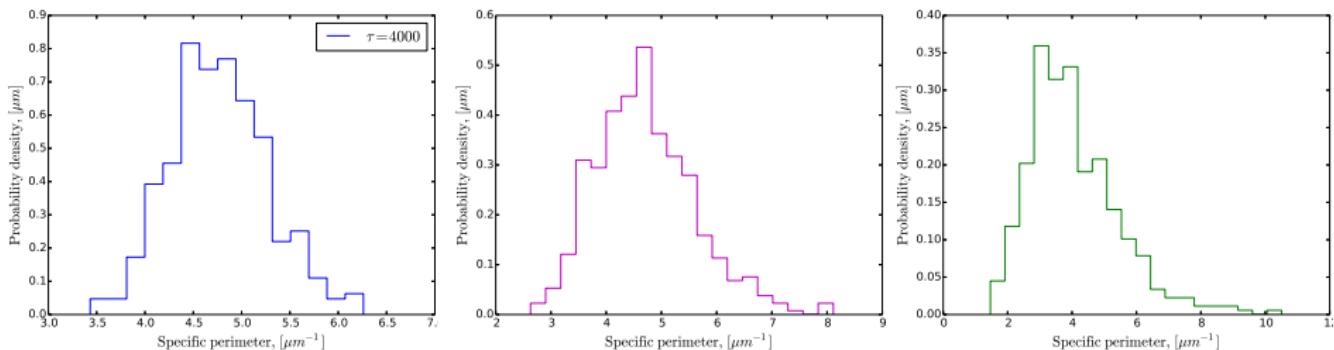
# Effect of the contraction parameter parameter on tessellation summary statistics

Specific perimeter:



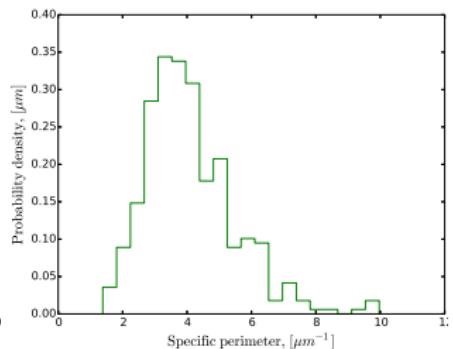
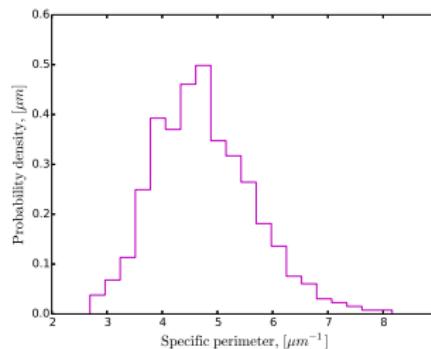
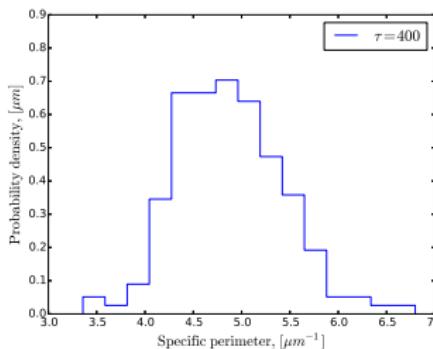
# Effect of the contraction parameter parameter on tessellation summary statistics

Specific perimeter:



# Effect of the contraction parameter parameter on tessellation summary statistics

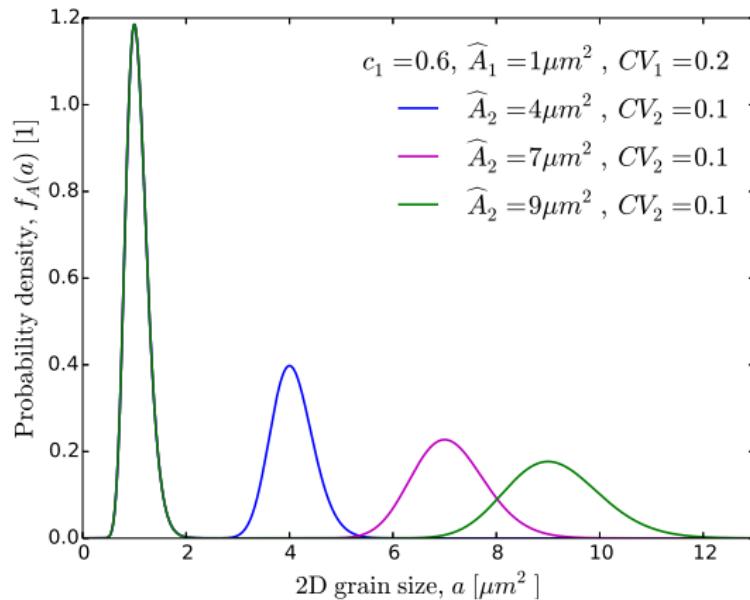
Specific perimeter:



# Some bimodal target 2D grain size distributions

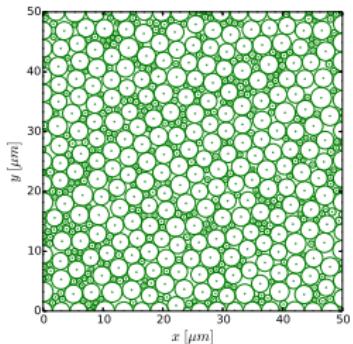
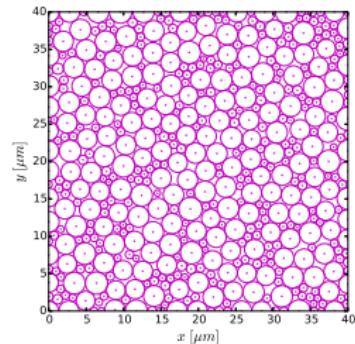
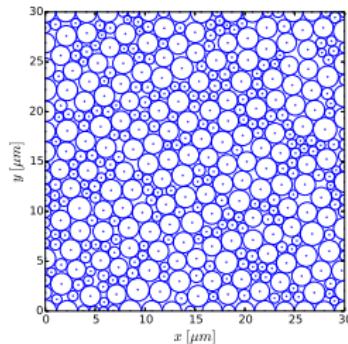
Bimodal log-normal PDF of 2D grain size  $A$ :

$$f_A(a|\mu_1, \sigma_1, \mu_2, \sigma_2, c_1) = c_1 f_A(a|\mu_1, \sigma_1) + (1 - c_1) f_A(a|\mu_2, \sigma_2), \quad a \in ]0, +\infty[$$

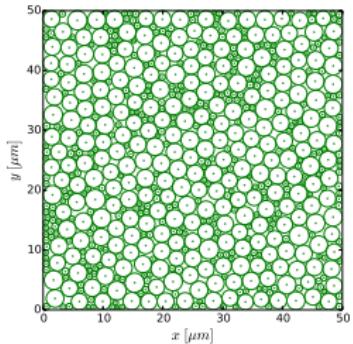
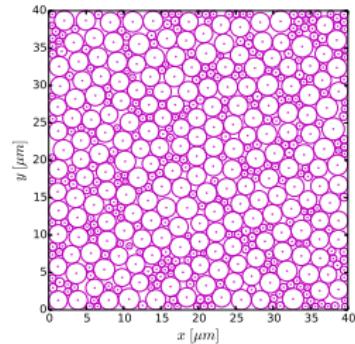
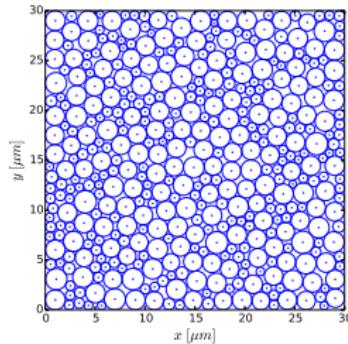


# Different ways to handle boundaries

Enforce points to be within the domain:

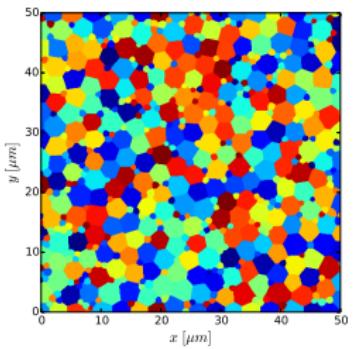
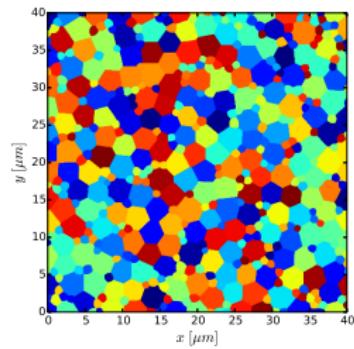
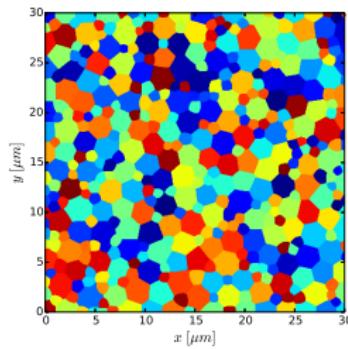


Enforce particles to be within the domain:

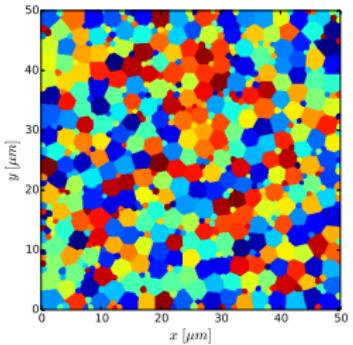
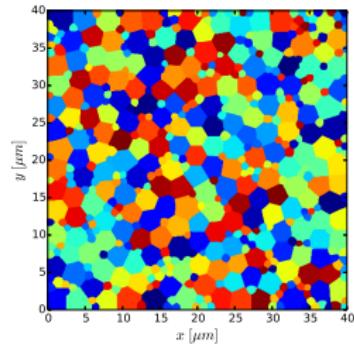
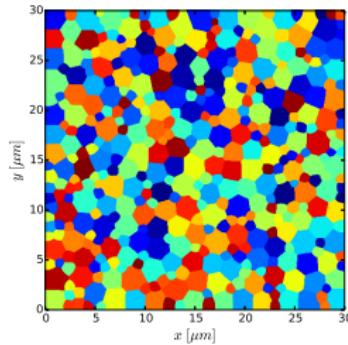


# Different ways to handle boundaries – resulting tessellations

Enforce points to be within the domain:

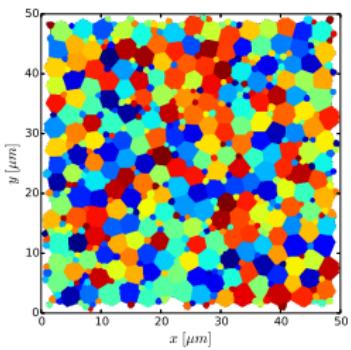
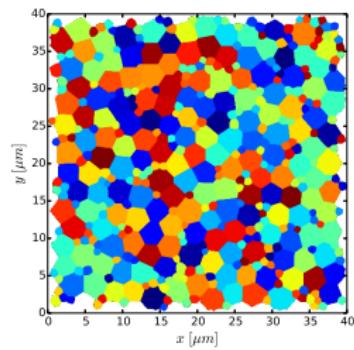
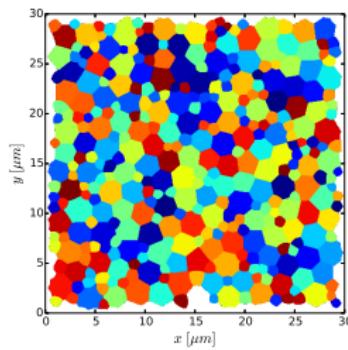


Enforce particles to be within the domain:

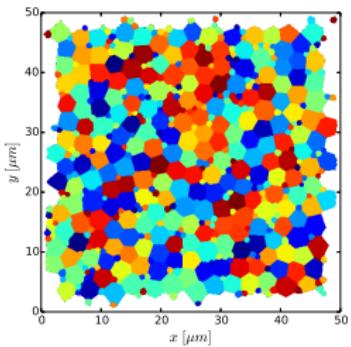
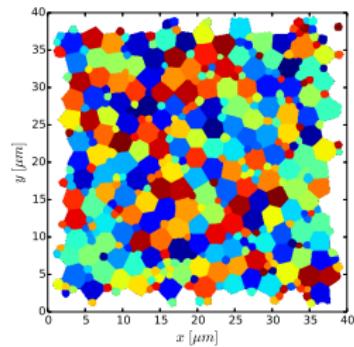
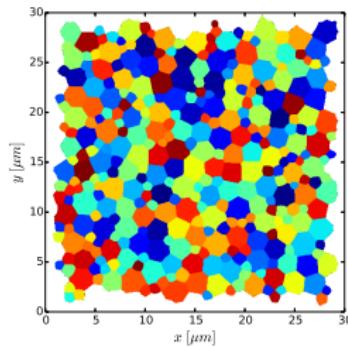


# Different ways to handle boundaries – resulting tessellations

Enforce points to be within the domain:

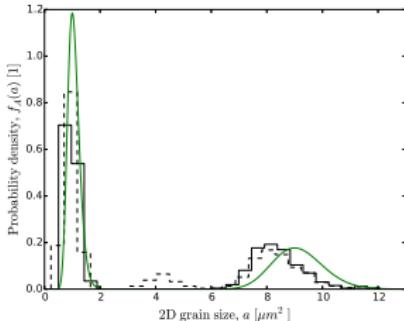
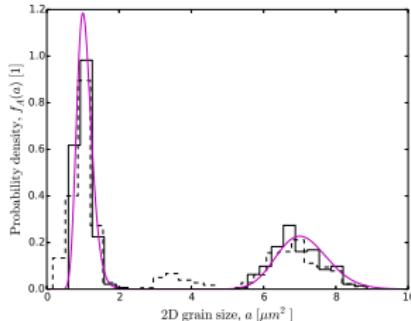
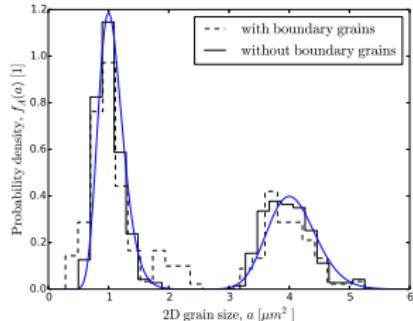


Enforce particles to be within the domain:

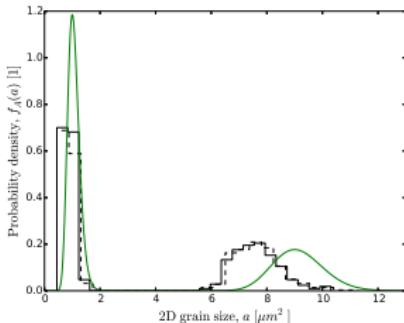
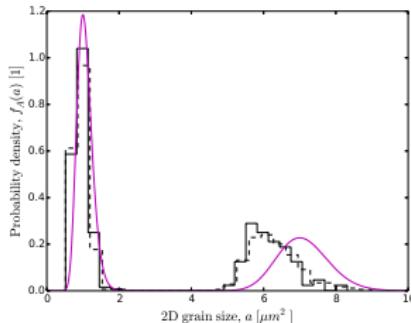
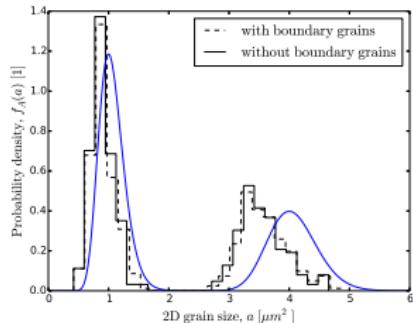


# Effect of boundary grains on the recovered grain size PDF

Enforce points to be within the domain:

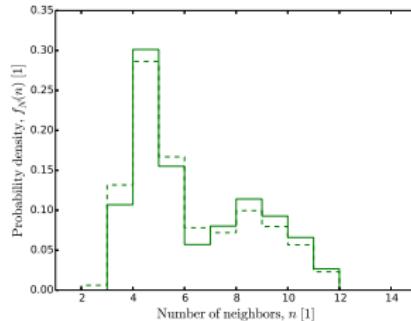
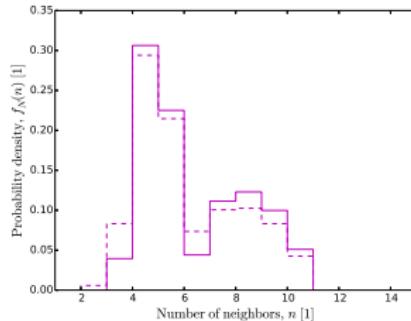
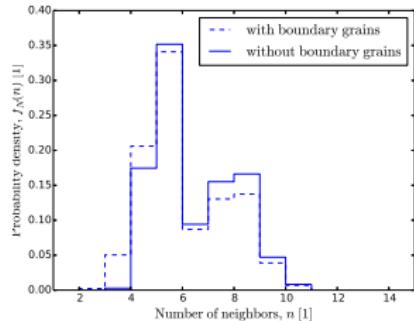


Enforce particles to be within the domain:

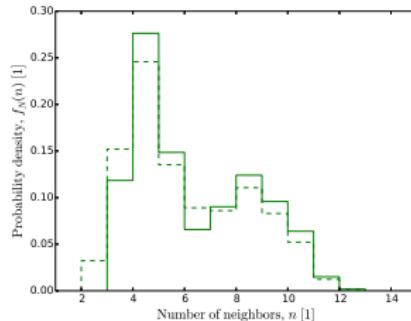
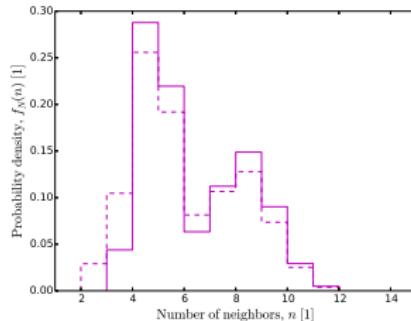
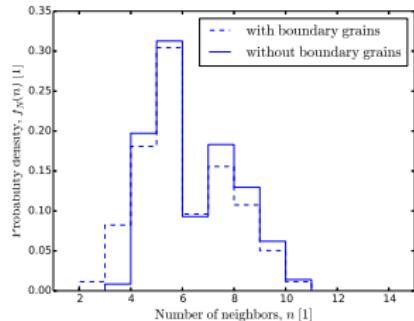


# Effect of boundary grains on the number of neighbors

Enforce points to be within the domain:



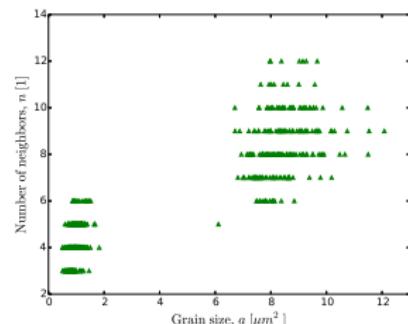
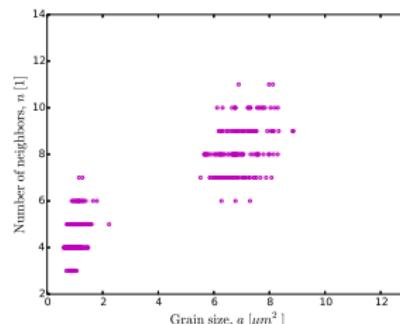
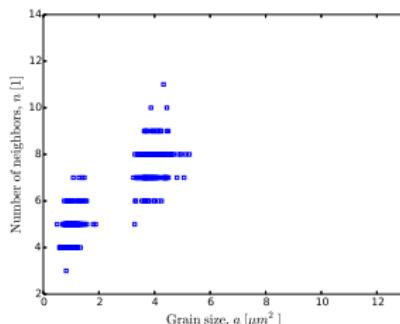
Enforce particles to be within the domain:



# Correllation between number of neighbors and grain size

Yet to investigate:

- How would these scatters change if convergence of the FBA was stated for a prescribed value of  $\eta < 1$ ?
- How do these scatters evolve as a function of  $\varphi$  and  $\tau$ ?
- Can we relate the random number of neighbors to the random grain size for a prescribed  $f_A(a)$ ?



**NB:** The results presented are for the bulk grains in simulations where the points only are constrained to be within the domain.

# Computation time

## MPP simulation

- serial implementation
- convergence at  $\eta = 1$
- $\varphi = 0.5$ ,  $\tau = 40000$

## SGT model resolution

- parallel implementation: 6 threads
- $\Delta x = 0.02 \mu m$

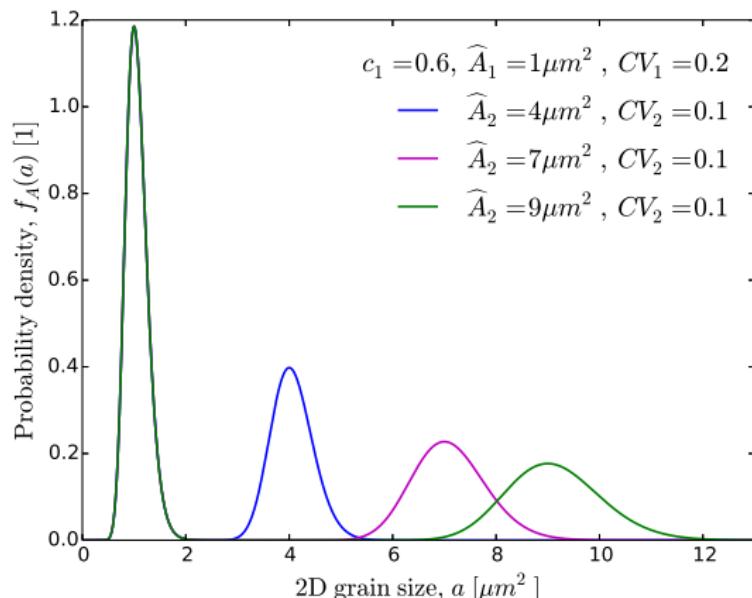
Target, $f_A(a \hat{A} = 1 \mu m^2)$	MPP	SGT	Total
$\hat{A}_2 = 4 \mu m^2$ , $A_{cont} = 30 \times 30 \mu m^2$	66 s (5265 it.)	25 s	91 s
$\hat{A}_2 = 7 \mu m^2$ , $A_{cont} = 40 \times 40 \mu m^2$	85 s (4826 it.)	57 s	142 s
$\hat{A}_2 = 9 \mu m^2$ , $A_{cont} = 50 \times 50 \mu m^2$	157 s (5596 it.)	109 s	266 s

**NB:** The results presented are only for points constrained to be within the domain. MPP simulations enforcing particles to be within the domain are more time-consuming. The simulation times for SGT might be underestimated.

# A target 2D grain size PDF of nanoengineered material

Bimodal log-normal PDF of 2D grain size  $A$ :

$$f_A(a|\mu_1, \sigma_1, \mu_2, \sigma_2, c_1) = c_1 f_A(a|\mu_1, \sigma_1) + (1 - c_1) f_A(a|\mu_2, \sigma_2), \quad a \in ]0, +\infty[$$



Volume fractions:

$$\phi_1 = \frac{c_1 \hat{A}_1}{c_1 \hat{A}_1 + (1 - c_1) \hat{A}_2 \exp(CV_2^2 - CV_1^2)}$$

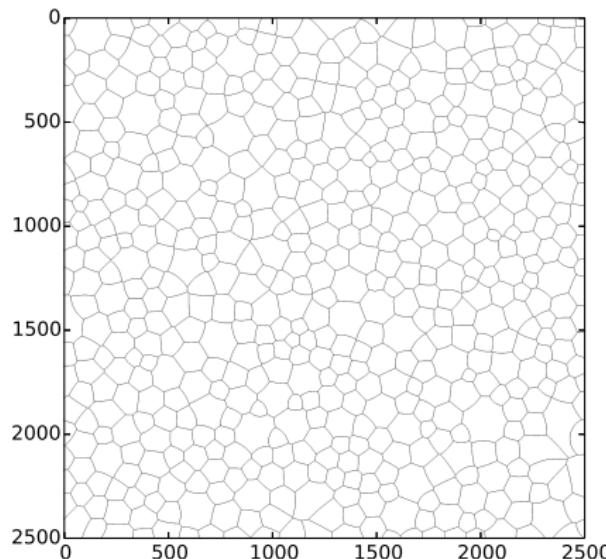
$$\phi_2 = 1 - \phi_1$$

Ratio of modes:

# A target 2D grain size PDF of nanoengineered material

Bimodal log-normal PDF of 2D grain size  $A$ :

$$f_A(a|\mu_1, \sigma_1, \mu_2, \sigma_2, c_1) = c_1 f_A(a|\mu_1, \sigma_1) + (1 - c_1) f_A(a|\mu_2, \sigma_2), \quad a \in ]0, +\infty[$$



Volume fractions:

$$\phi_1 = \frac{c_1 \hat{A}_1}{c_1 \hat{A}_1 + (1 - c_1) \hat{A}_2 \exp(CV_2^2 - CV_1^2)}$$

$$\phi_2 = 1 - \phi_1$$

Ratio of modes:

## Points and questions to address

- ① Can we fit a inhomogeneous rate of Poisson process onto such results? If so, what about the marks?
- ② Improve the efficiency of the current force-biased algorithm implementation

## References |

- Bezrukov, A., Bargiel, M., and Stoyan, D. (2002). Statistical analysis of simulated random packings of spheres. *Particle and Particle Systems Characterization*, 19(2):111–118.
- Teferra, K. and Graham-Brady, L. (Under review). Grain growth tessellation models for polycrystalline microstructures. *Computational Materials Science*.