Numerical simulation of three-dimensional grain growth in materials containing finely dispersed second-phase particles with the phase field method

Liesbeth Vanherpe[1] Nele Moelans[2] Bart Blanpain[2]
Stefan Vandewalle[1]

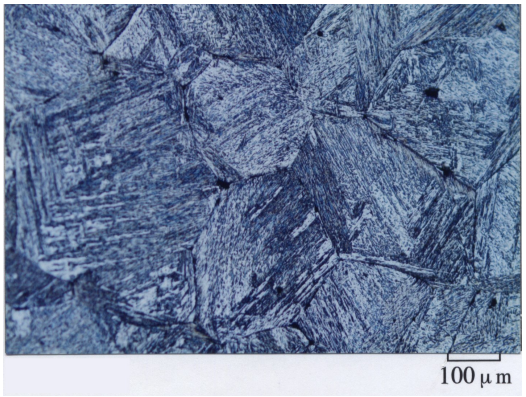
[1]Department of Computer Science, K.U.Leuven

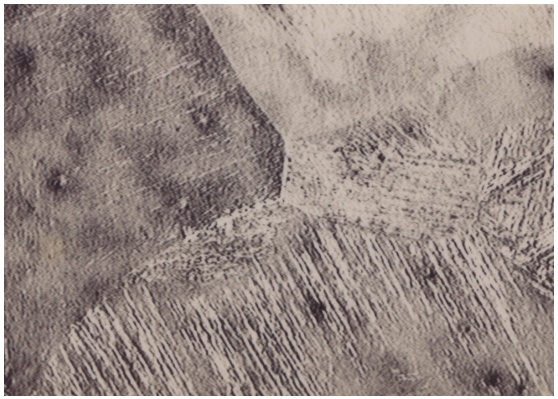
[2]Department of Metallurgy and Materials Engineering, K.U.Leuven

ICIAM 07, Zurich July 16 - 20, 2007

Outline

Polycrystalline materials

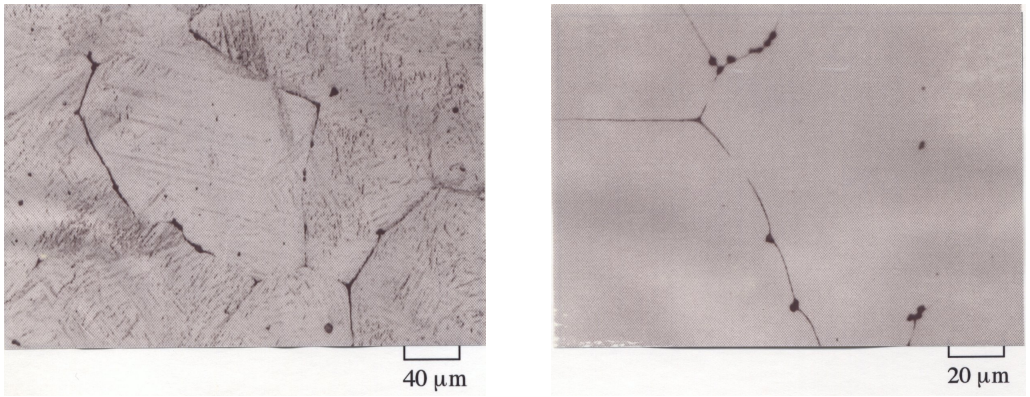




[M. Guo]

- microstructure of grains with different crystallographic orientations
- material properties depend on microstructure
- study of grain growth: increase of mean grain size

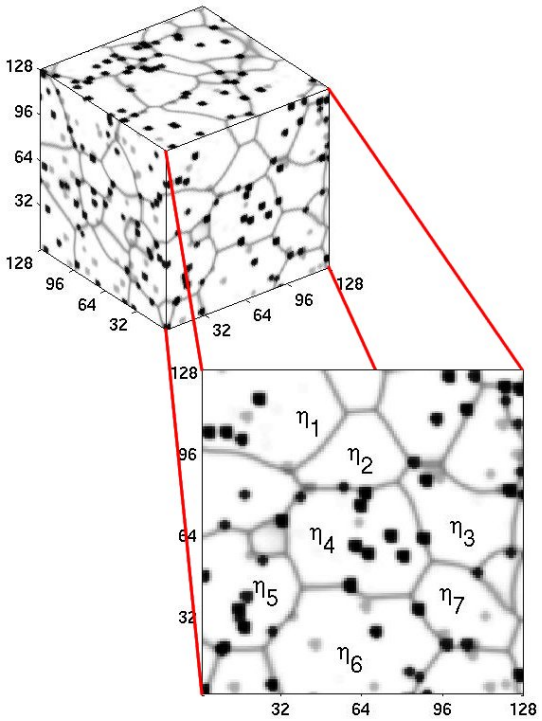
Grain growth in presence of second-phase particles



[M. Guo]

- control grain size by addition of impurities
- applications
  - design of high-strength steels for construction applications
  - design of thin metallic films in microelectronic devices
- importance of computational research

Phase field model



- set of phase field variables

$$\eta_i(\mathbf{r}, t), \qquad i = 1, \ldots, p$$

- inside grain i

$$(\eta_1, \ldots, \eta_i, \ldots, \eta_p) = (0, \ldots, \pm 1, \ldots, 0)$$

- parameter ϕ for the second-phase particles
  - $\phi = 1$ inside particles
  - $\phi = 0$ else

Phase field model

- kinetic equations

$$\frac{\partial \eta_i}{\partial t} = -L\frac{\partial F}{\partial \eta_i}, \qquad i = 1, \ldots, p$$

- free energy

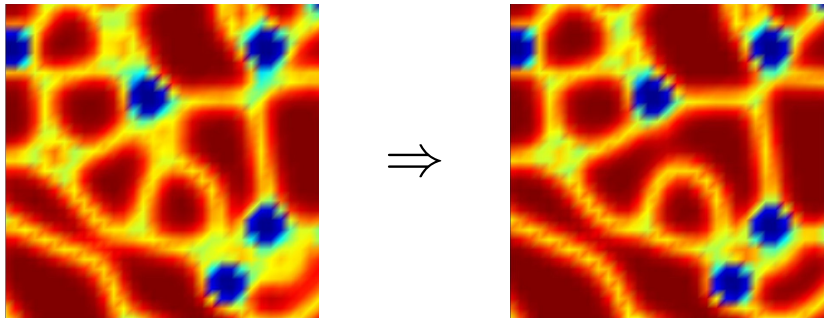
$$F = \int_V \left(\sum_{i=1}^{p} \left(\frac{\eta_i^4}{4} - \frac{\eta_i^2}{2}\right) + \sum_{i=1}^{p}\sum_{j\neq i}^{p} \eta_i^2\eta_j^2 + \phi^2 \sum_{i=1}^{p} \eta_i^2 + \sum_{i=1}^{p} \frac{\kappa}{2}(\nabla \eta_i)^2 \right) dV$$

- set reaction-diffusion partial differential equations

$$\frac{\partial \eta_i}{\partial t} = L\kappa\nabla^2\eta_i - L\left(\eta_i^3 - \eta_i + 2\eta_i \sum_{j\neq i}^{p} \eta_j^2 + 2\eta_i\phi^2\right), \qquad i = 1, \ldots, p$$

Points of attention

- prevent grain coalescence:



solution large amount of phase field variables

- grains, boundaries and particles

  solution fine resolution of grid

- statistically relevant results

  solution large amount of grains

Parallel computing

$$\frac{\eta_i^{n+1} - \eta_i^n}{\partial t} = L\kappa\nabla^2\eta_i^{n+1} - L\left(\eta_i^3 - \eta_i + 2\eta_i\left(\sum_{j\neq i}^{p}\eta_j^2 + \phi^2\right)\right)^n,\ i = 1,\ldots,p$$

- semi-implicit discretisation
  - \+ p smaller systems to solve
  - \+ allows for large time step
- parallel computing
  - \+ assign equations to different processors
  - \+ compute solutions in parallel
  - – communication step: large message

Characteristics of the solutions

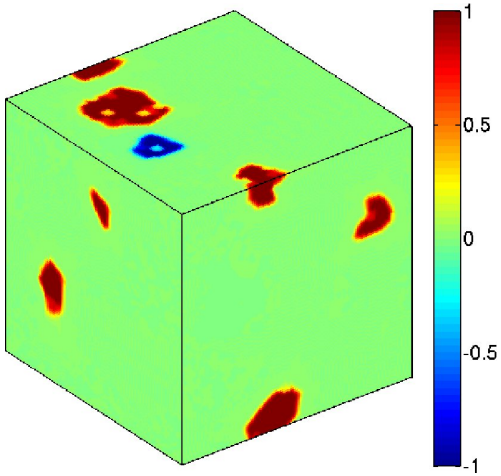
Definition

A phase field variable η_i is active at a grid point $\mathbf{r}$ *when* $|\eta_i(\mathbf{r})| > \epsilon$*, with* ϵ *a small positive threshold value.*

Observations:

1. at each grid point, only a few η_i active
2. values of η_i only evolve at the grain boundaries

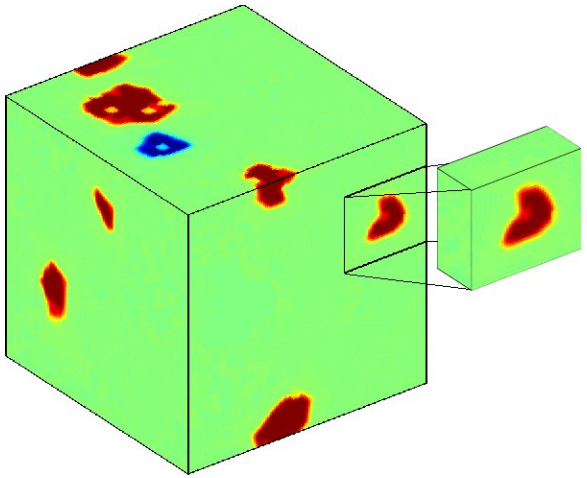
$\Rightarrow$ solve only locally



Bounding box algorithm

Preprocessing step

1. apply threshold ϵ
2. locate grains: connected grid points $\mathbf{r}$ where $|\eta_i(\mathbf{r})| > \epsilon$
3. determine bounding boxes
   - bounding box: smallest cuboid containing grain
   - grain numbering procedure to prevent coalescence



Computation step

Bounding box algorithm

Preprocessing step

Computation step

- solve the equations only locally:

$$\frac{\eta_i^{n+1} - \eta_i^n}{\partial t} = L\kappa\nabla^2\eta_i^{n+1} - L\left(\eta_i^3 - \eta_i + 2\eta_i(\sum_{j\neq i}^{p}\eta_j^2 + \phi^2)\right)^n$$

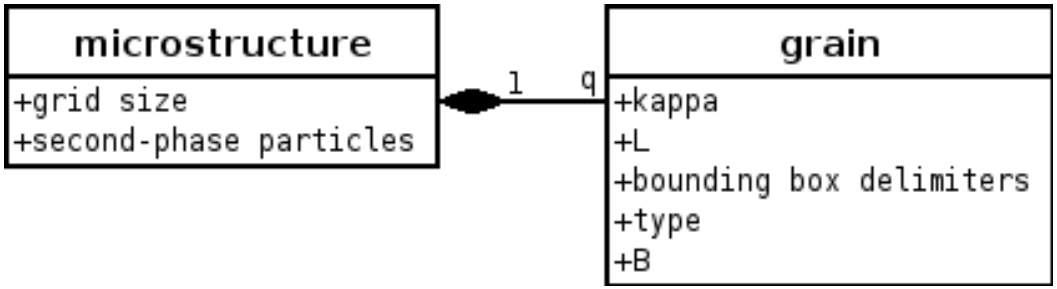
for time steps $t = 1, \ldots, n$

1. for all grid points: compute $\sum \eta_j^2 + \phi^2$
2. for all bounding boxes: solve equations

- only values inside boxes are kept in memory

Bounding box algorithm

Data structure



- object-oriented data structure
- advantages in post-processing
  - location of grains known
- extensible to more complex models
  - anisotropy
  - evolving second-phase particles

Computational requirements

Table 1: Number of phase field values per grid point on a $256 \times 256 \times 256$ grid with $p = 100$.

	threshold value ϵ				
simulation time t	10^{-3}	10^{-4}	10^{-5}	10^{-6}	conventional
t_s	5.75	6.48	7.26	8.10	100
$t_s + 200$	6.71	6.74	6.90	6.90	100
$t_s + 400$	6.72	6.73	6.81	6.81	100
$t_s + 600$	6.68	6.69	6.73	6.73	100
$t_s + 800$	6.64	6.65	6.68	6.68	100

- approx. 7 instead of p phase field values per grid point
- scales with system size, not with p

Computational requirements

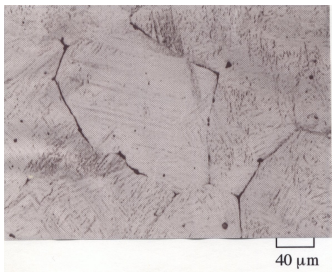
Table 2: Requirements of 5000 time steps with $\epsilon = 10^{-5}$ on a $256 \times 256 \times 256$ grid with $p = 100$.

	b. box (on 1 proc)		conv. (on 20 procs)	
simulation time	time (h)	mem (GB)	time (h)	mem (GB)
$200 \rightarrow 400$	6.3	2.8	6.8	30.3
$400 \rightarrow 600$	4.7	2.2	6.8	30.3
$600 \rightarrow 800$	3.8	1.9	6.8	30.3
$800 \rightarrow 1200$	3.5	1.7	6.8	30.3
$1200 \rightarrow 1400$	3.1	1.6	6.8	30.3

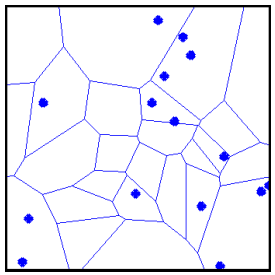
- 1 instead of 20 processors and faster
- efficiency increases for coarser topology

Initial microstructure

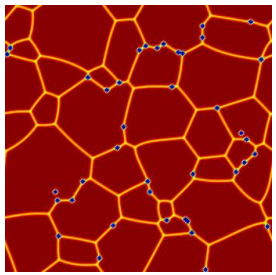
- start from initial polycrystalline microstructure



microscopic images

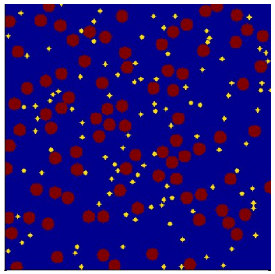


Voronoi calculation

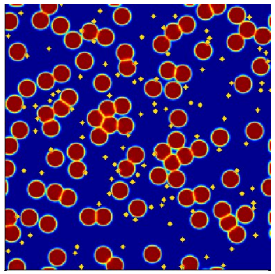


phase field simulation

- generate nuclei according to some distribution



$\Rightarrow$

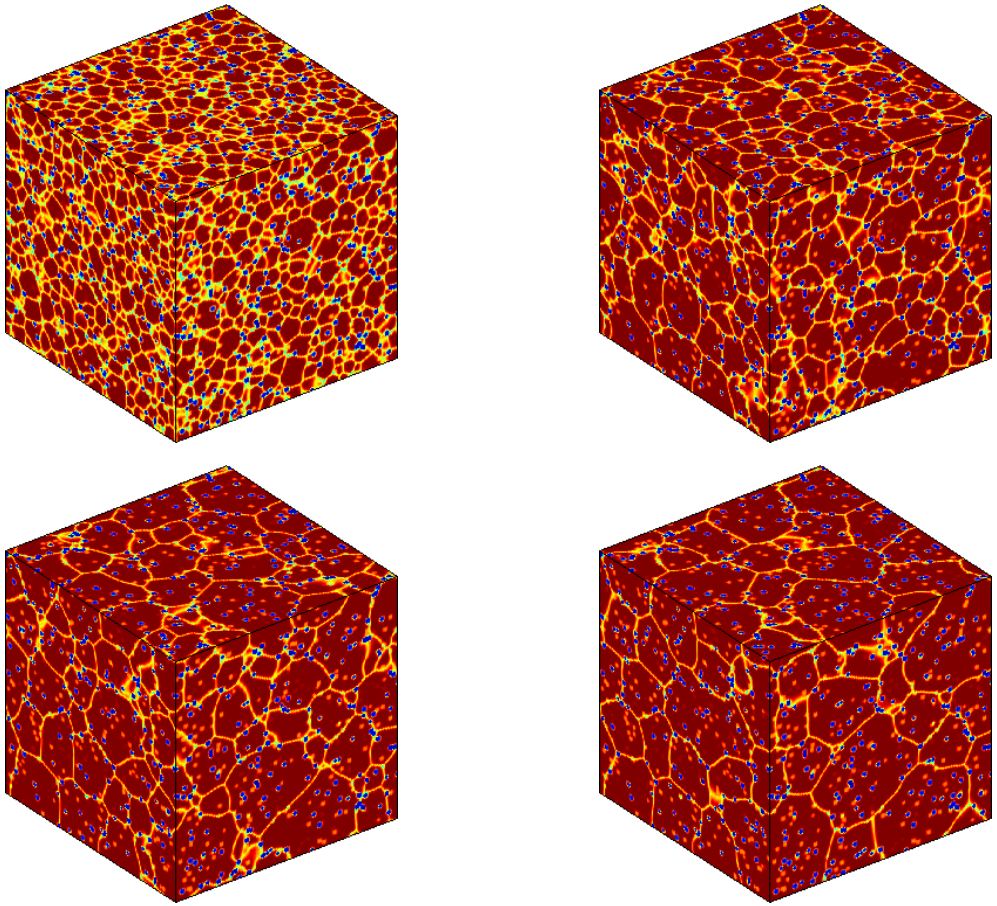


Application

Grain growth in presence of second-phase particles

- initial polycrystalline microstructure obtained by 'conventional' phase field simulation
  - $256 \times 256 \times 256$ grid
  - 100 phase field variables
  - $f_v = 5\%$, 8% and 12% of particles, radius $r = 3$
  - performed on 20 processors
- bounding box algorithm
  - threshold $\epsilon = 10^{-5}$
  - only 1 processor
  - approx. three times faster

Application
Microstructural evolution



Application
Results

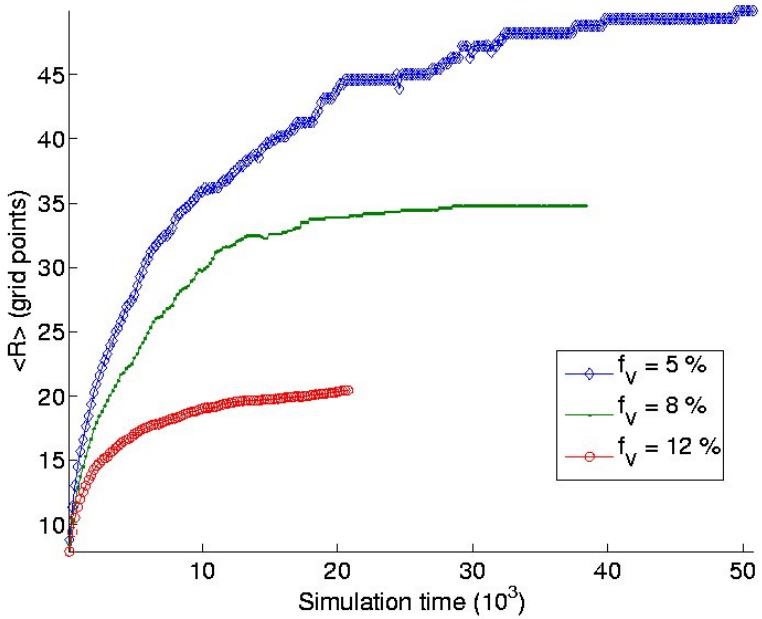


Fig 1: Evolution of mean grain size for different f_V.

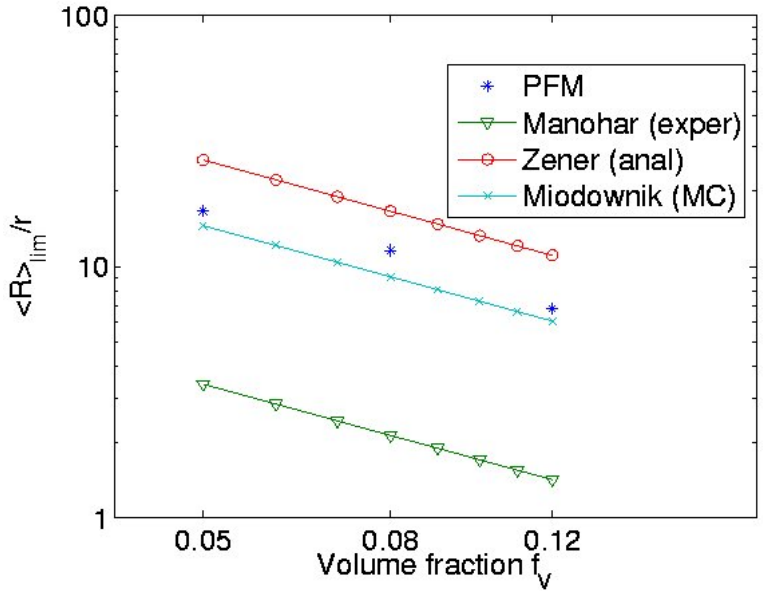


Fig 2: Comparison with other results.

Conclusion

- phase field model for grain growth in presence of second-phase particles
  - computationally intensive
- three-dimensional simulation
  - parallel computing
    - semi-implicit discretisation
    - larger systems, but large communication message
  - bounding box algorithm
    - less computing time, less memory
    - no grain coalescence
- application

Thank you

Thank you! Questions?