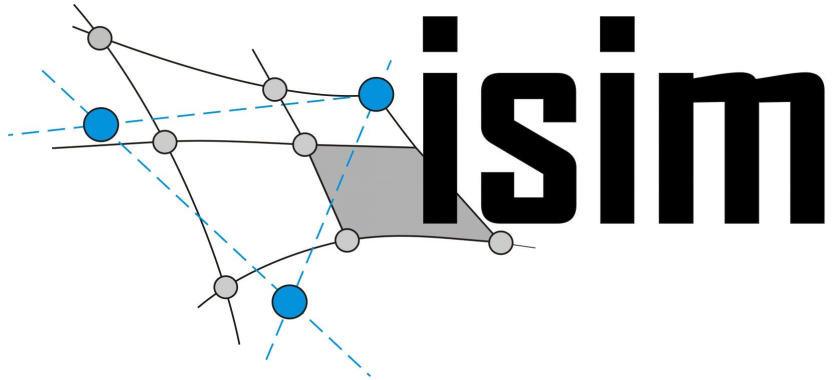
Multiscale Modeling

Dawid Zych

E-mail: dawid.zych@agh.edu.pl

Room : B5 607

Consultation: Sunday 1 pm - 2 pm
(please, contact me before)



Course credit terms and rules

- At the beginning of the classes your work from previous classes is rated
  - Working software, live demo
- One short test - announced beforehand
- Presence at the classes is obligatory
  - 1 absence without any after-effects
  - Medical leaves - right after absence period
  - Sum of absences cannot be > 50%
- Final grade is weighted arithmetic mean, but each grade has to be positive
  - Project progress - 5 x 0.8
  - Reports - 5 x 0.2
  - Test - 1 x 0.8
- Exam “0” from 4.0

Reports

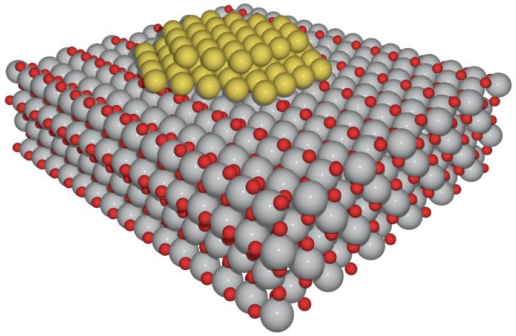
- Has to be short, concise
- What was the topic? What have I done?
- How the user interface looks like? Functionalities
- Sample results with the input parameters
- Conclusion / Afterthoughts / Discussion of some problem which you had and how you solved it?
- Link to GitHub / Bitbucket / GitLab / Any VCS provider
  - Repository is necessary
  - Commits should be meaningful, and not too big
- How the solution can be built and run, required dependencies?
- Mail to dawid.zych@agh.edu.pl with report saved as .pdf file - before next classes

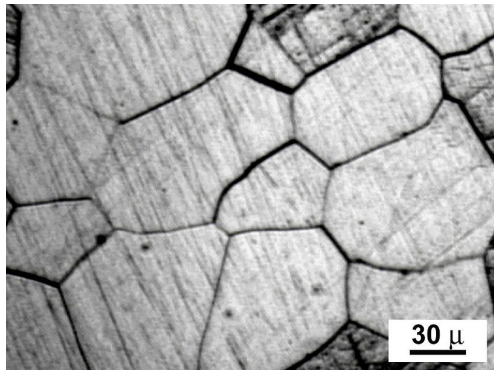
Classes Schedule

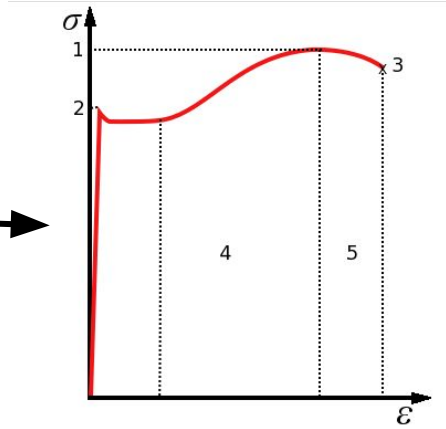
1. **Kickoff meeting. Introduction. Simple Grain Growth Cellular Automaton**
2. Inclusions feature
3. Consideration of grain curvature
4. Substructures
5. Boundaries detection
6. Leftovers, project submitting

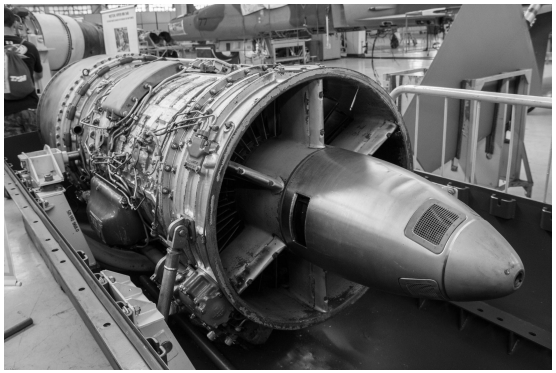
Multiscale modeling

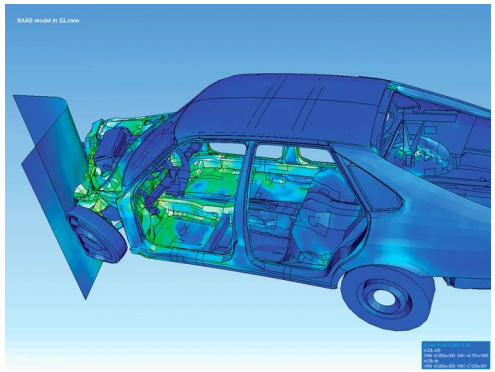
Problem of the scale



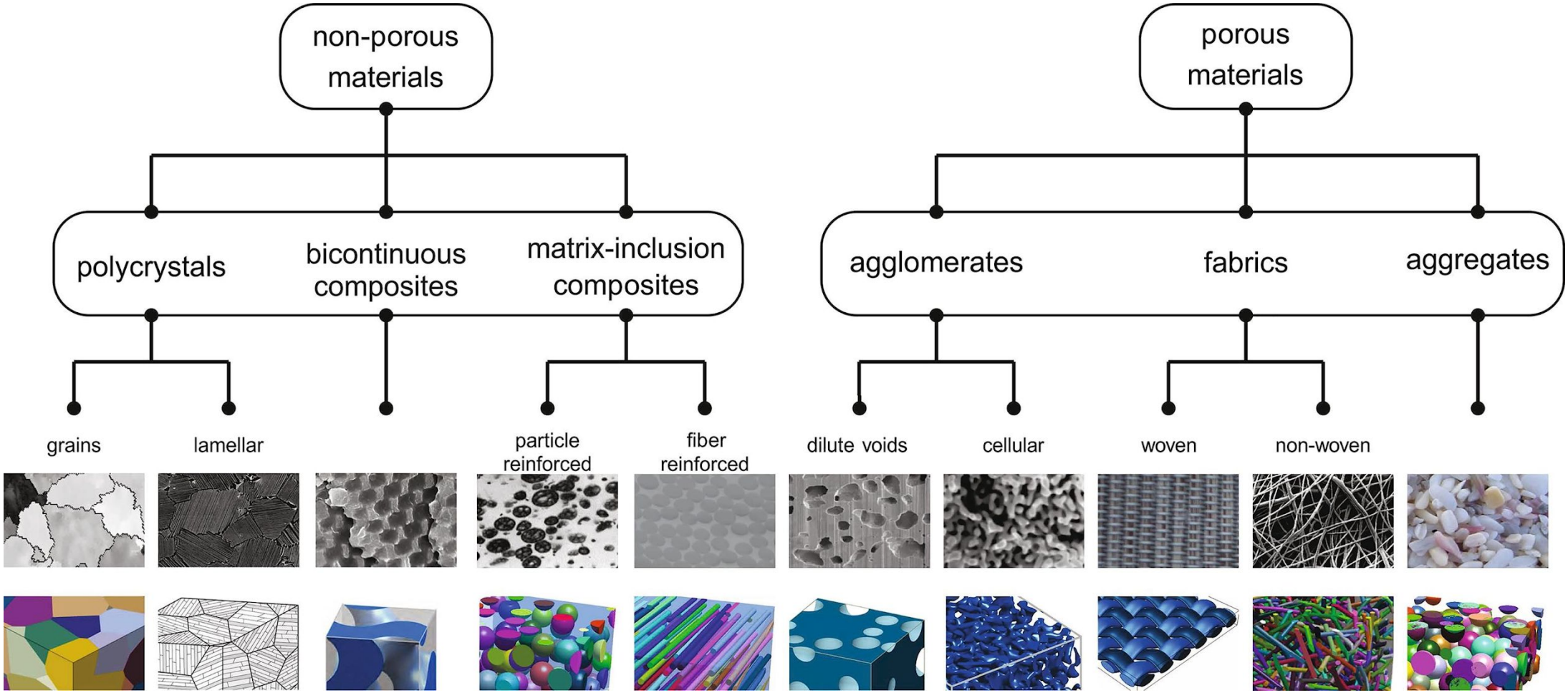




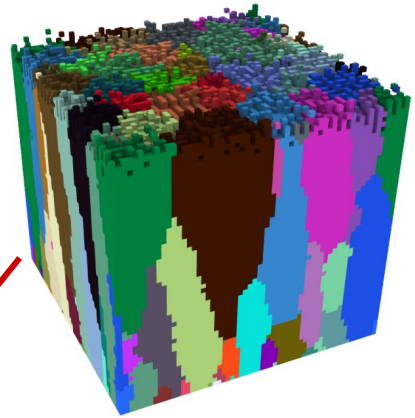




Digital Material Representation



How DMR can be obtained?



Microscopy Image Analysis (SEM, STM)

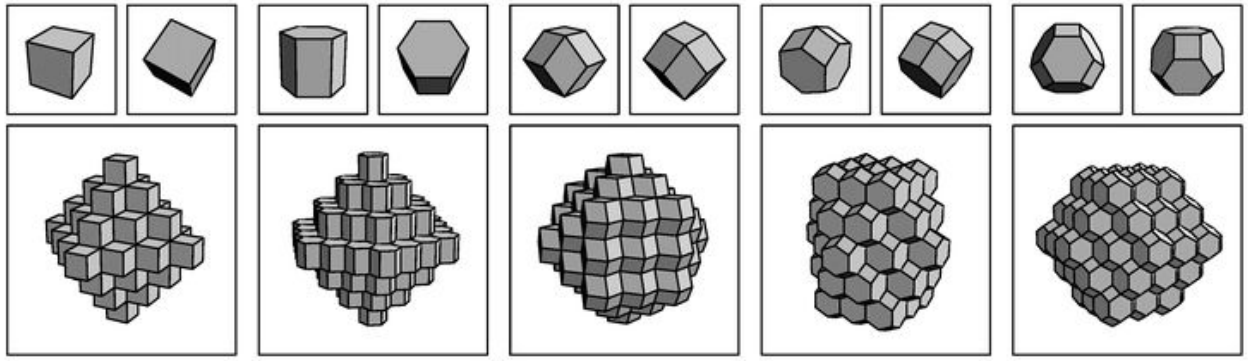
Methods statistically representative

- **Cellular Automata**
- Monte Carlo
- Voronoi diagrams

Cellular Automata Model

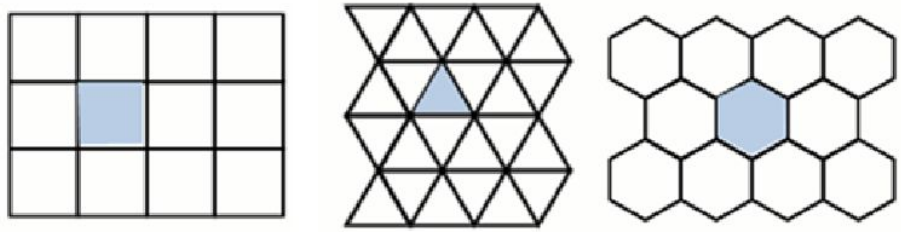
Fundamental definitions

Idea of the model : Divide domain into finite set of cells, each cell has the state which can evolve in time.



Different cell shapes, different lattices. Space 3D

- **CA Space** - finite set of cells, where each cell is described by a set of internal variables describing the state of a cell.

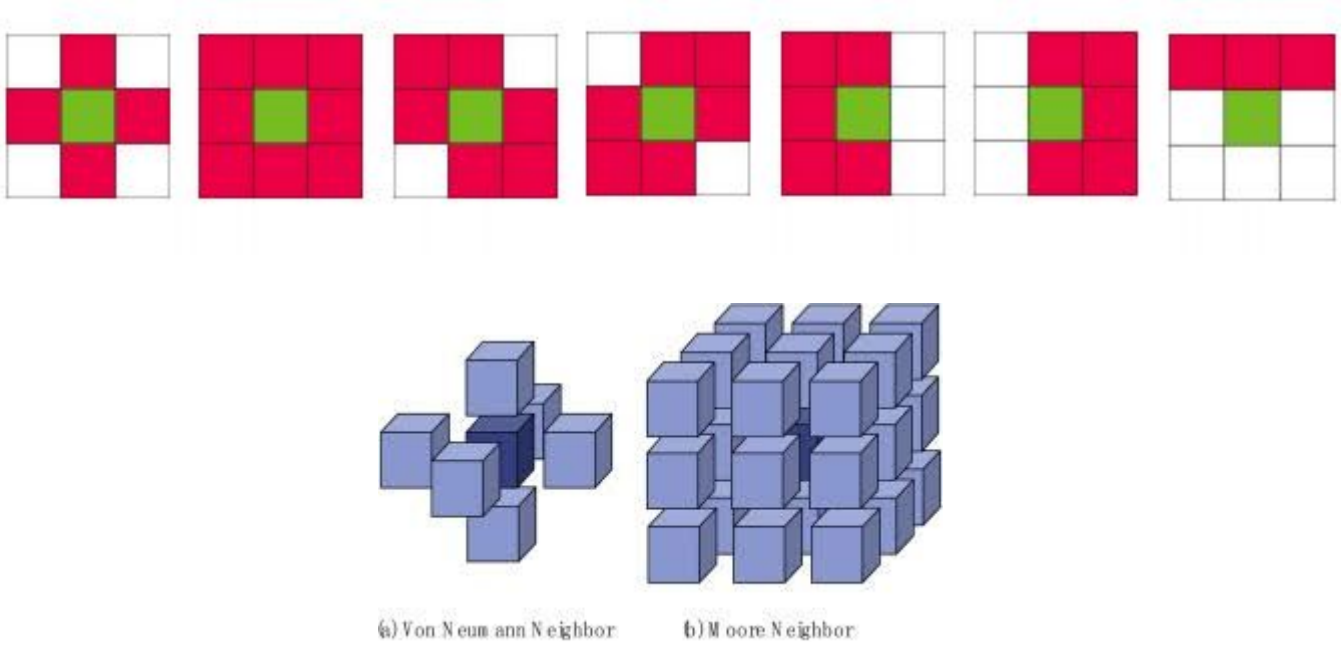


Space 2D

Cellular Automata Model

Fundamental definitions

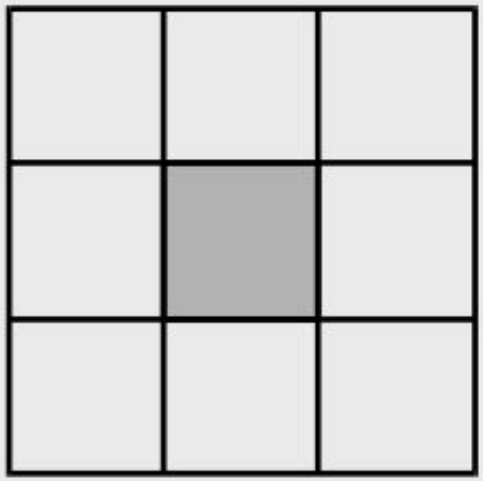
- **Neighbourhood** - set of cells which are located in particular, close order to considered cell



Cellular Automata Model

Fundamental definitions

- **Transition rule** - a function which returns new state of the cell based on:
  - state of considered cell in previous time step
  - state of neighbourhood of considered cell in previous time step



$$\gamma_{i,j}^{t+\Delta t} = f\begin{pmatrix} \gamma_{i-1,j-1}^{t}, & \gamma_{i-1,j}^{t}, & \gamma_{i-1,j+1}^{t}, \\ \gamma_{i,j-1}^{t}, & \gamma_{i,j}^{t}, & \gamma_{i,j+1}^{t}, \\ \gamma_{i+1,j-1}^{t}, & \gamma_{i+1,j}^{t}, & \gamma_{i+1,j+1}^{t} \end{pmatrix}$$

Boundary conditions

- How we can fetch the neighbourhood when considered cell is on the edge of the lattice?

9	3	6	9	3
7	1	4	7	1
8	2	5	8	2
9	3	6	9	3
7	1	4	7	1

Periodic boundary condition
(snakelike)

0	0	0	0	0
0	1	4	7	0
0	2	5	8	0
0	3	6	9	0
0	0	0	0	0

Non-periodic boundary condition
(absorbing)

Simple Grain Growth CA Definition

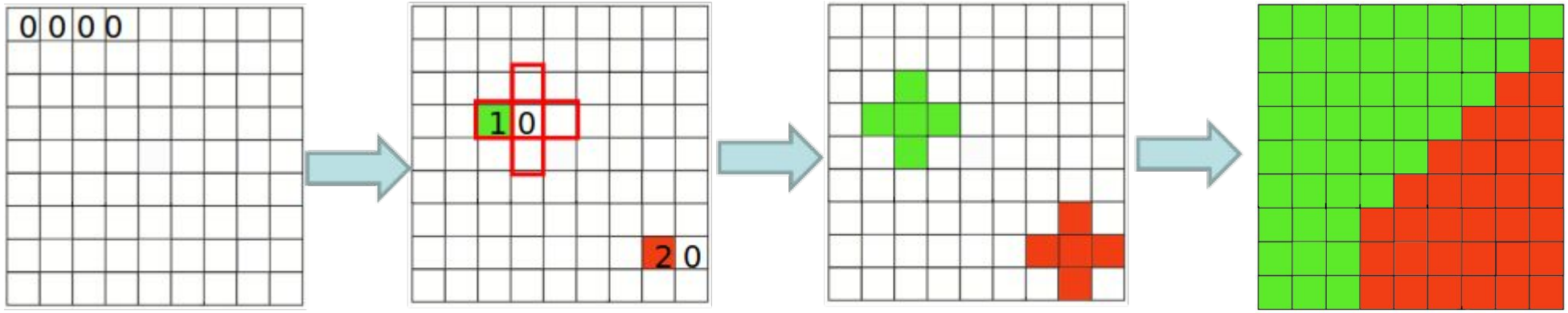
1. Space:
    a. Dimension: 2D
    b. Size: 500x500
    c. Lattice: regular, square
    d. Number of possible states of the cell : 2 (Grain, Empty)
    e. Internal Variable : GrainId
2. Neighbourhood:
    a. von Neumanna
    b. Moore'a
    c. Hexagonal left / right / random
    d. Pentagonal left / right / random
3. Transition rules

Transition Rules

1. If the cell is in state *Grain*, nothing happens

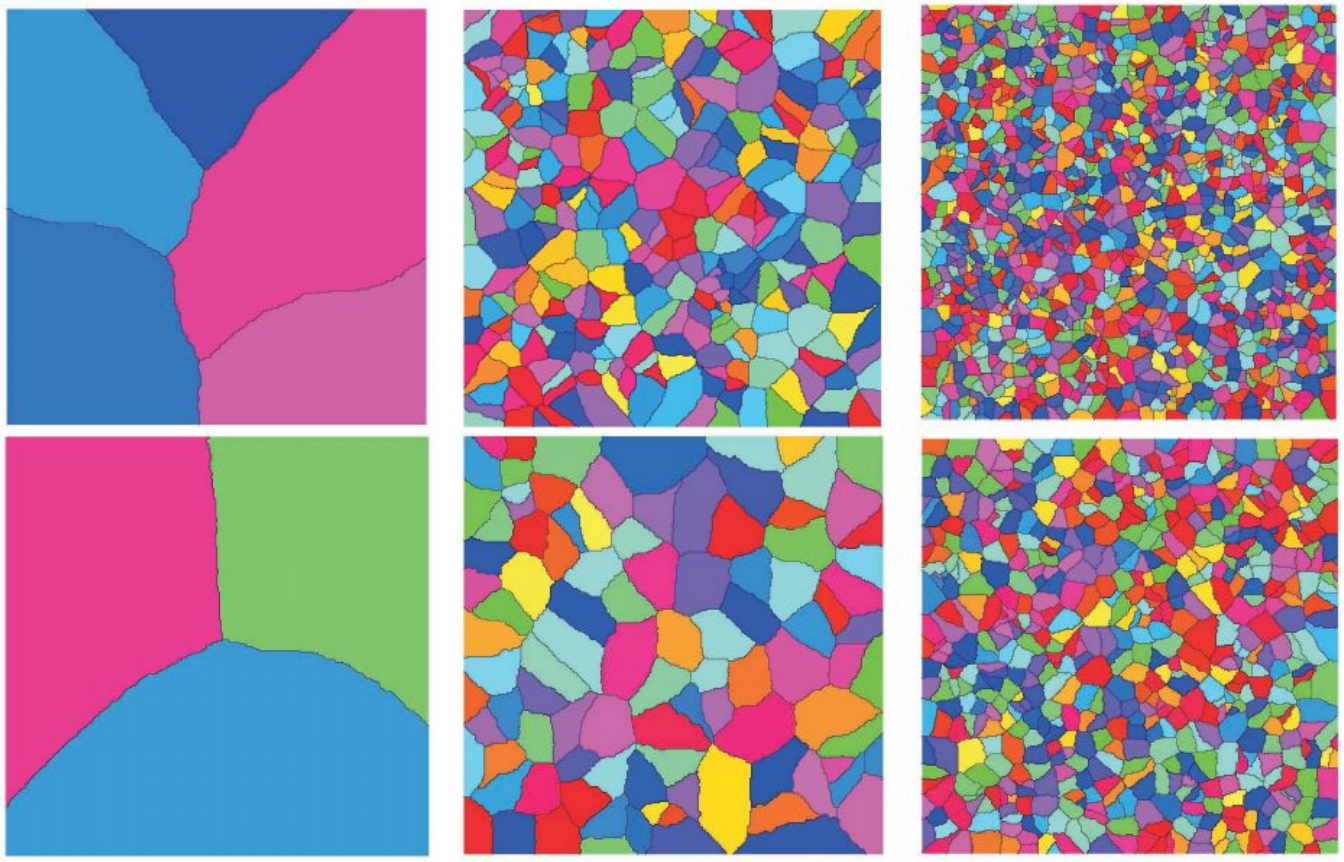
2. If the cell is in state *Empty*, and there is any non-empty cell in the neighbourhood, then the cell will become *Grain*
    a. The cell is going to belong to the grain (GrainId) which is most numerous in the neighbourhood. In case of tie, the choice should be random across most numerous cases.

Progression



(von Neumann neighbourhood)

Expected results



Madej, Łukasz & Rauch, Łukasz & Yang, Richard (Chunhui. (2009).
Strain distribution analysis based on the digital material representation. Archives of Metallurgy and Materials. 54. 499-507.

Implementation

- GUI - any language, any framework, any platform, any OS
- Visual representation of the current space state
- Number of grain seeds
- Size of the space
- Control of the execution
  - Reset - clears the space
  - Next step - performs one CA step
  - Play / Pause - performing in the loop
- Neighbourhood selection
- Boundary condition selection
- Possibility of Export as
  - Dump of the current state (e.g. CSV file: X, Y, Z, GrainId)
  - Bitmap or another raster format
- Possibility of Import and continuation of the calculation from:
  - Dump of state file