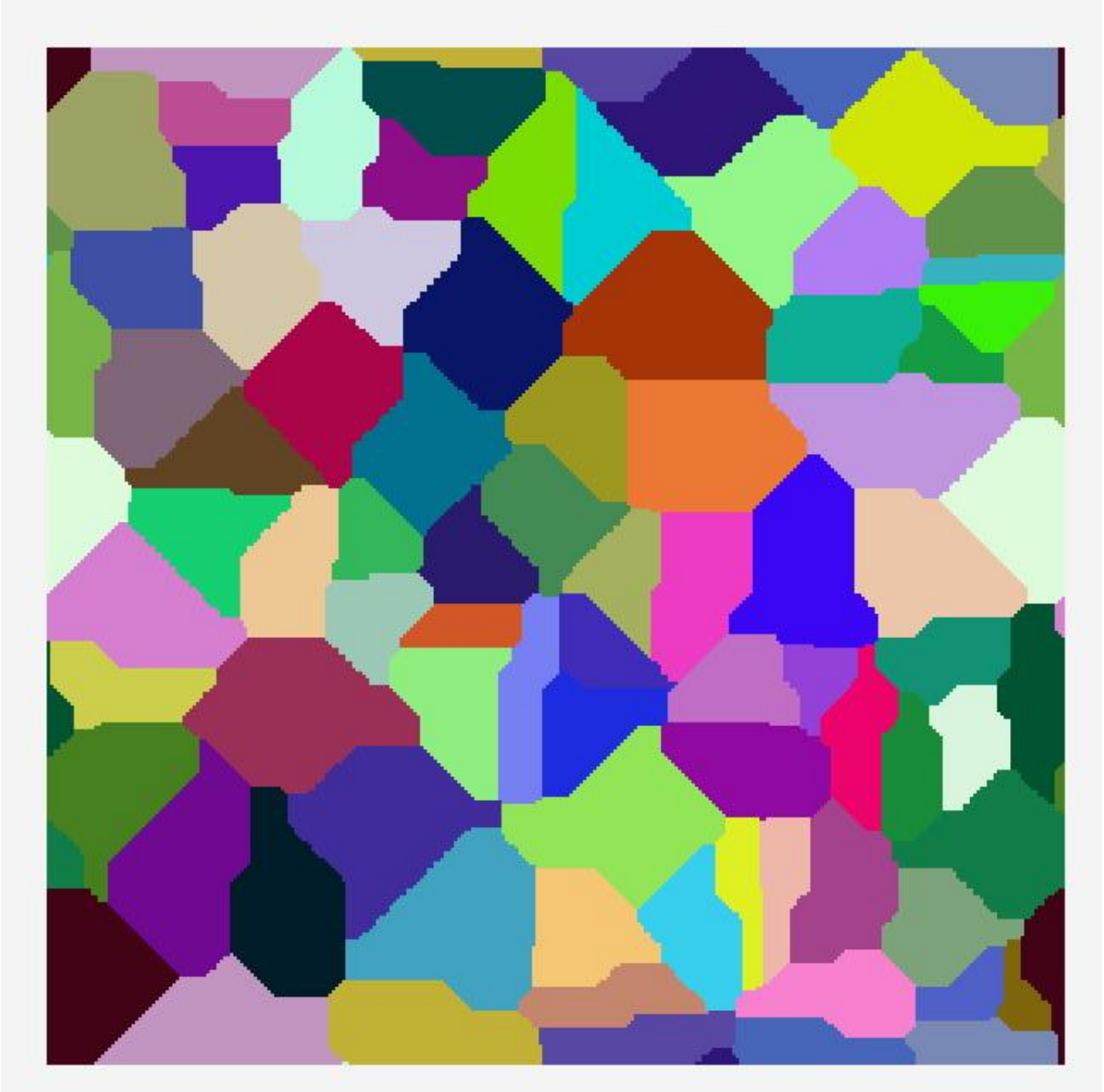
Multiscale modelling – 1st report

Author: Grzegorz Klimek, IS



Content

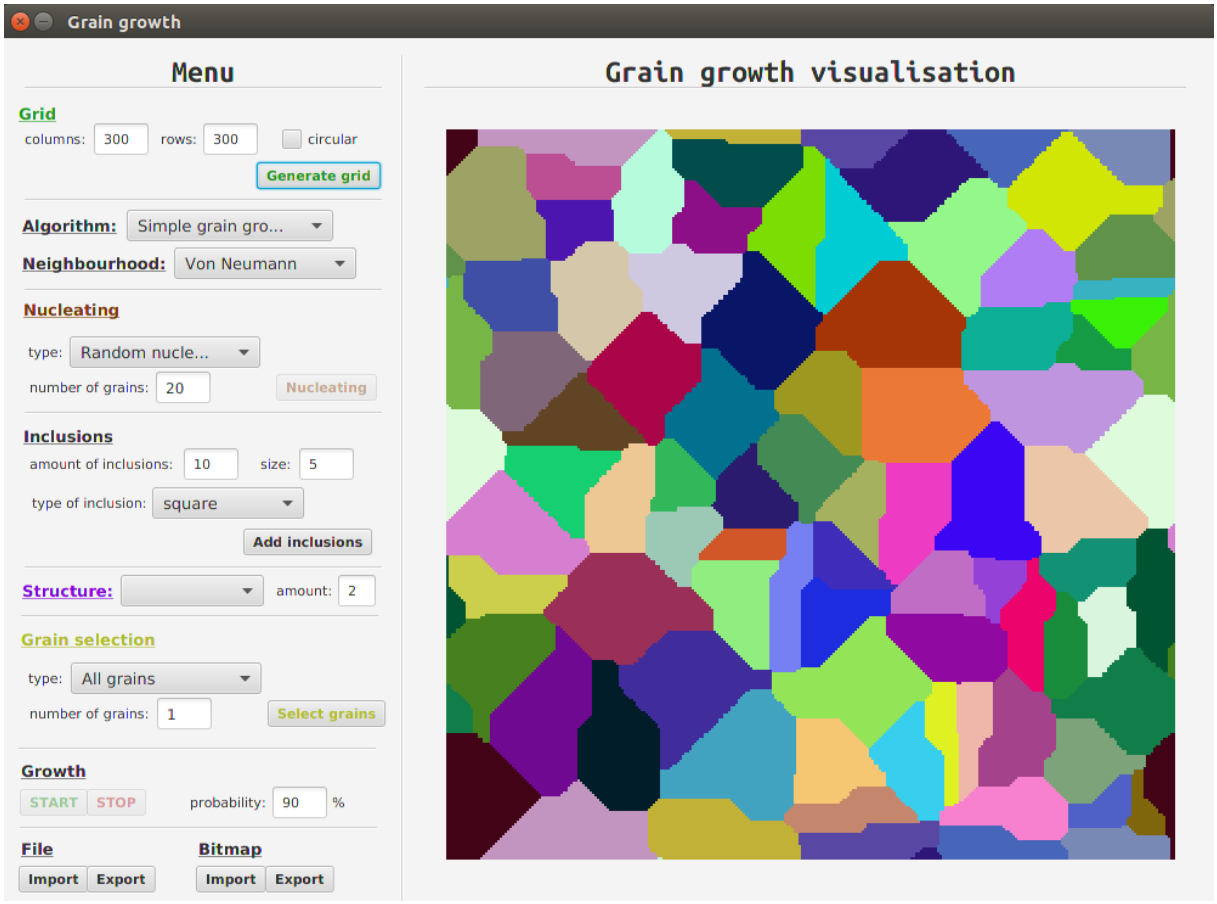
1. Technologies

- Java 8
- JavaFX (user interface)

I used Java because this language is very popular, have simply syntax and foremost is very universal. We can build application and run it from every OS without any trouble thanks to JVM.

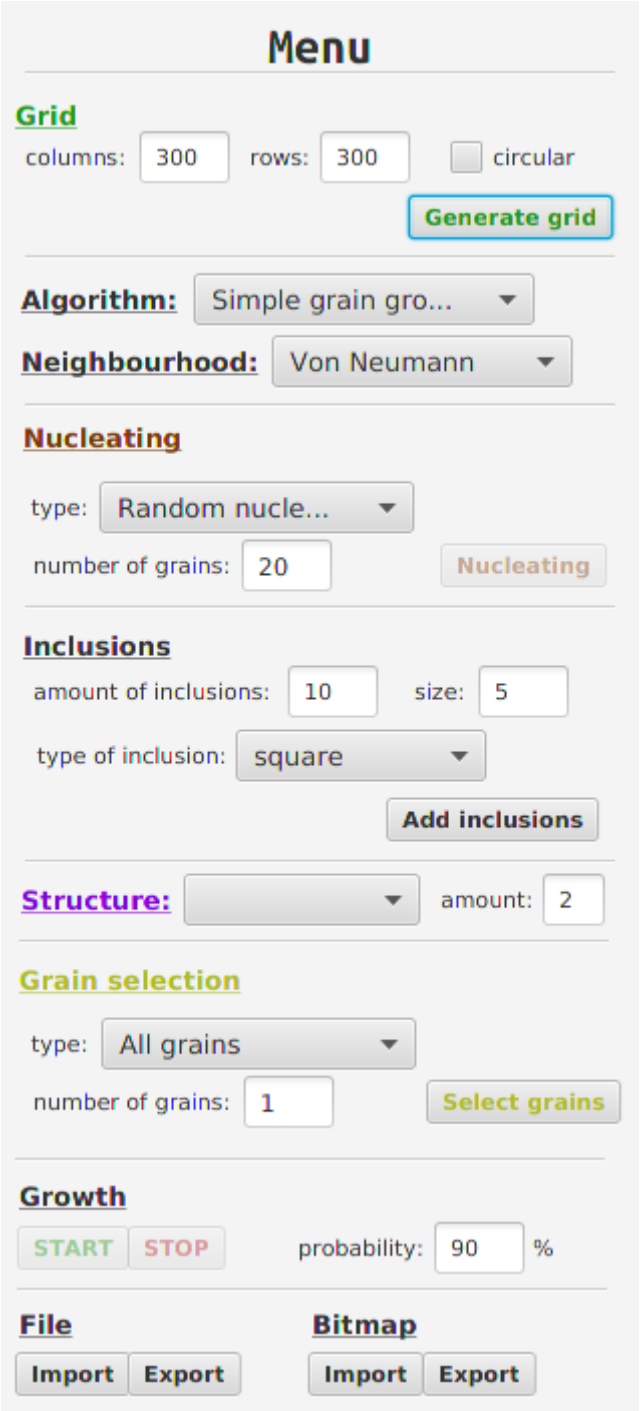
JavaFX used to build gui is flexible and easy to use. Thanks to FXML we can easily design all scenes. Very helpful tool is Java FX Scene builder which help us adding new elements to our application, change their properties, location etc. The key factor which persuade me to choose this library is mapping between gui elements and their object representations.

2. User interface



Img 1: Main app view

In the Img 1 we can see main app view. On the left, there is main menu where we can start simulation, import and export microstructure or set some options. On the right we have the main part of the application – microstructure visualization. We can start/stop simulation from every step also it's associated with import from file or bitmap.

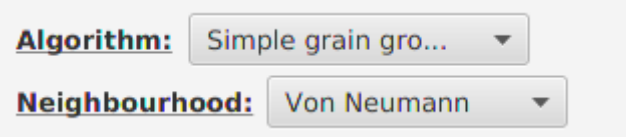


Img 2: Main menu

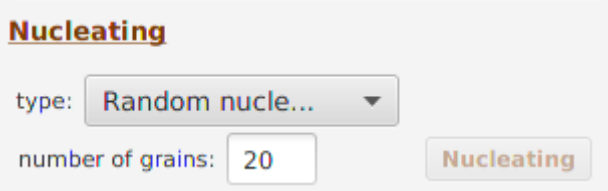
Img 2 shows main menu available in app. Menu was divided into sections to group it by functionalities. Some options are set when we generate grid for example algorithm or neighbourhood but not all of them. Structure or grain selection are available after microstructure is fully generated.

Main menu - options:

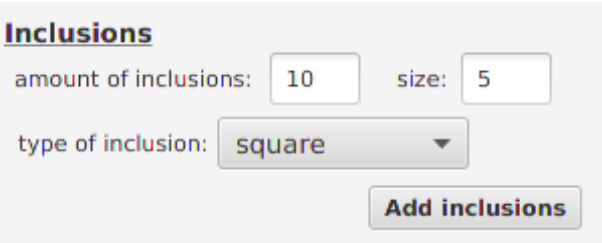
Grid – we can set columns and rows. Also we can generate circular grid. Button is used for generate grid with cells (initialize state – 0 and white color)



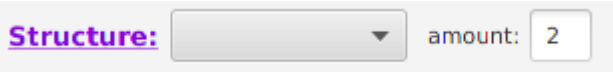
Algorithm and neighbourhood – we can select algorithm and neighbourhood type.



Nucleating – there is one type option – random nucleating which seed grains randomly on the grid. We can also specify how many grains generate.

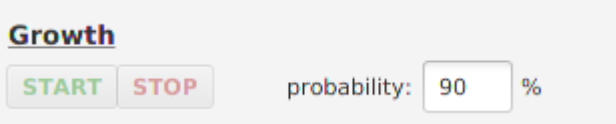


Inclusions – here we can generate inclusions, set their amount, type (square or circular) and size. There is possibility to add inclusion whenever we want. After simulation inclusions are added on grain edges.

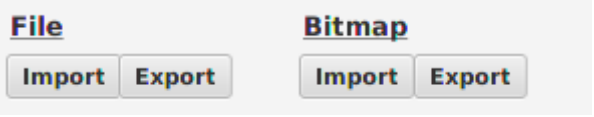


Structure – after simulation we can create grid with substructure or dual phase option

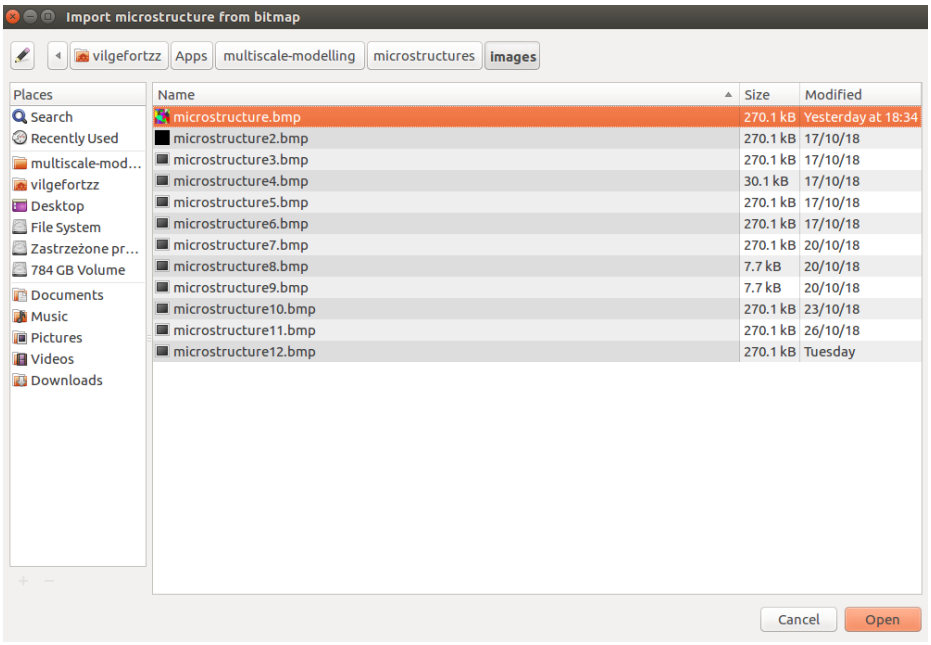
Grain selection – we can create inclusions, set amount, type (square or circular) and size. There is possibility to add inclusion whenever we want. After simulation inclusions are added on grain edges.



Growth – here is main control section where we can start/stop simulation. Additionally we can set probability percent.



File/Bitmap – the last section applies to import and export microstructure from file/bitmap to file/bitmap.



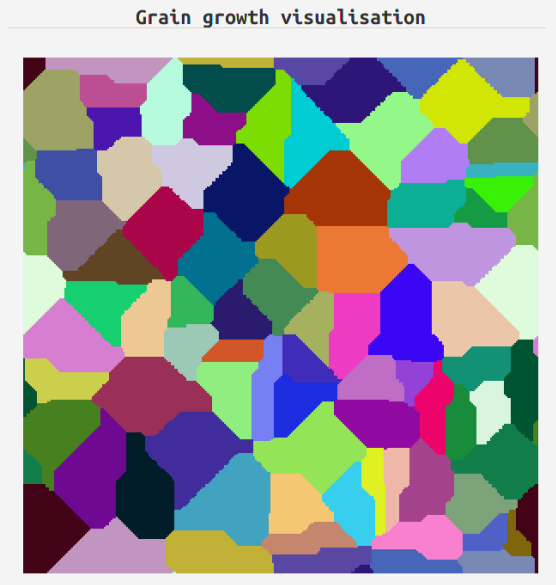
Img 3: File chooser for import/export microstructure

In the Img 3 there is export/import based on file chooser so easily we can choose location and name of file.

3. App operating

Below there is exemplary microstructure generated by application:

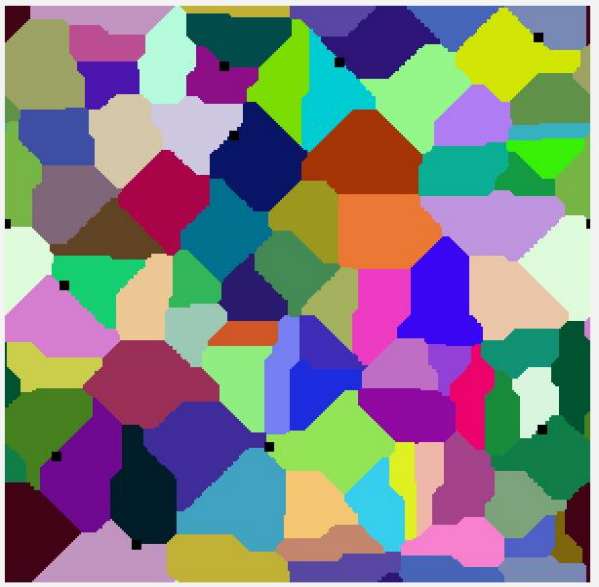
- Algorithm: Simple grain growth
- Neighbourhood: Von Neumann



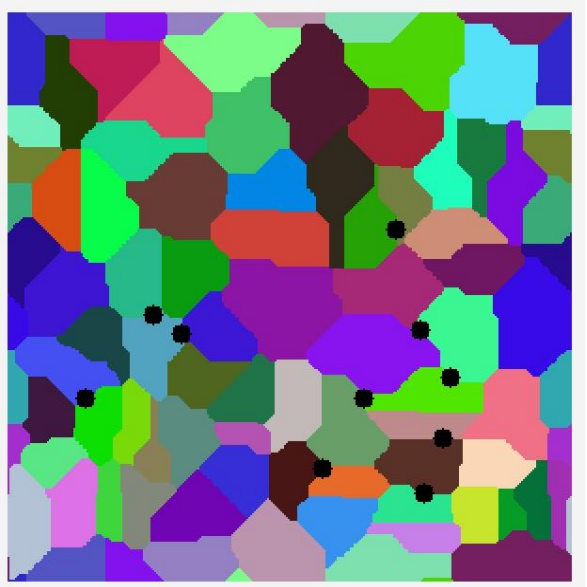
Img 4: Exemplary microstructure generated by application

Next example shows microstructure with inclusions:

- Amount of inclusions: 10
- Size: 5
- Type of inclusion: square or circular



Img 5: Square inclusions

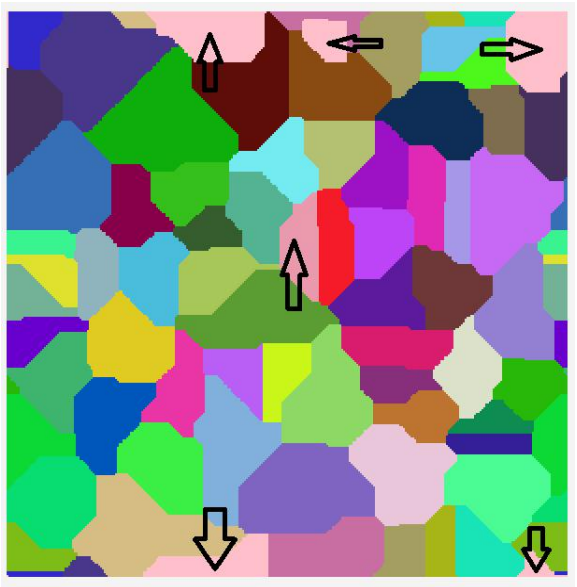


Img 6: Circular inclusions

This example shows new simulation with structure from previous one:

- Structure: substructure or dual phase

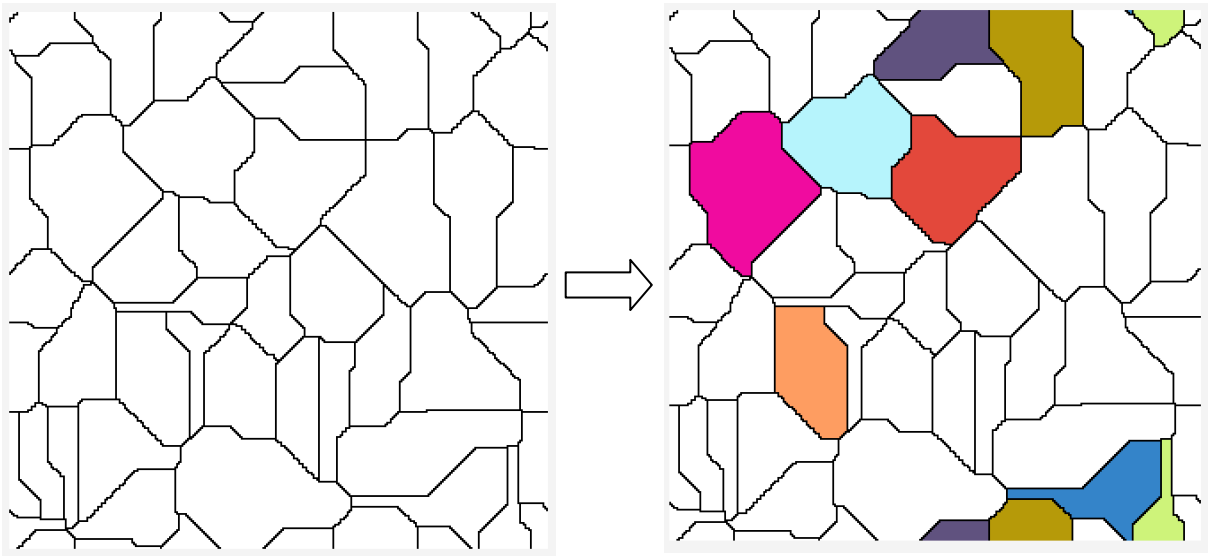
Img 7: Substructure



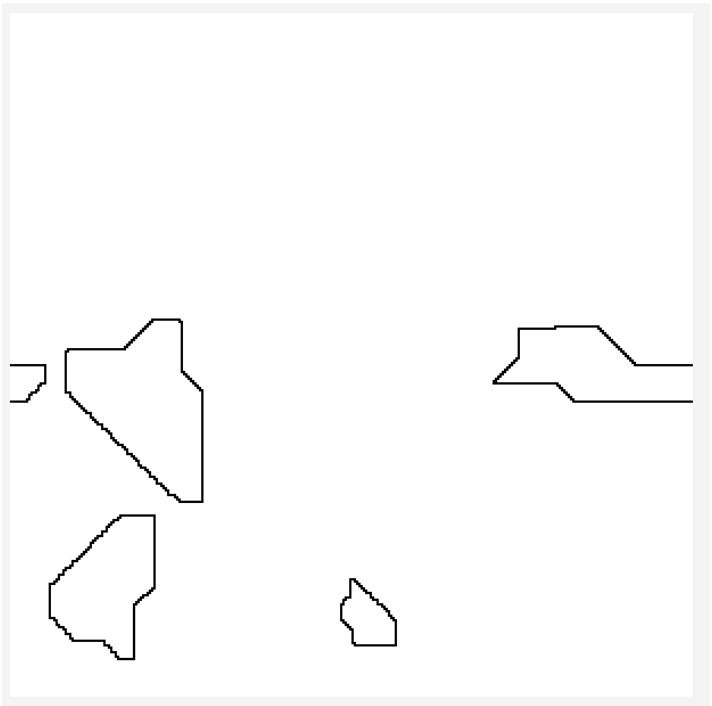
Img 8: Dual phase

Finally, we have grain selection (boundaries coloring):

- Grain selection: all grains or n grains
- Number of grains: 4



Img 9: All grains selected -> nucleating

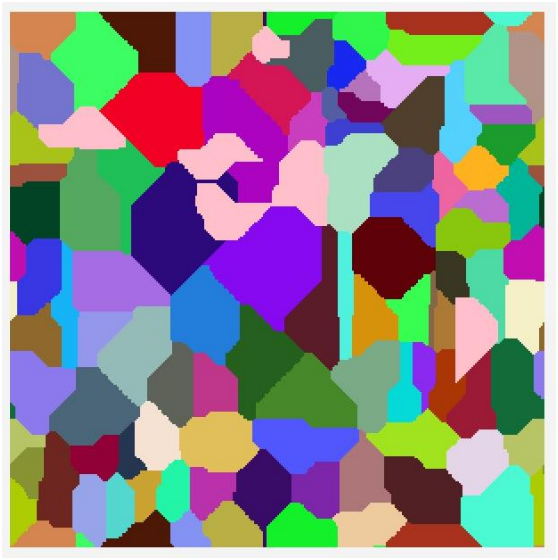


Img 10: 4 grains selected

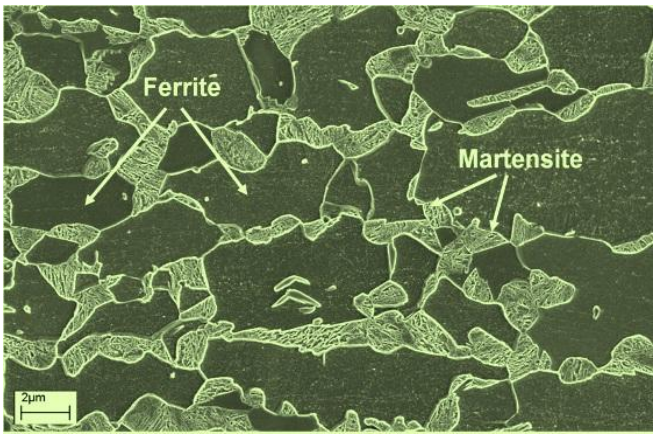
In the Img 10 we can see 4 grains actually their boundaries. After that we can nucleate and start simulation with grains selected.

4. Microstructure comparison

I took two microstructures to compare them. First microstructure comes from my application, second microstructure is dual phase steel. Both of them are shown below.



Img 11: Dual phase from app



Img 12: Dual phase steel

Conclusions from the comparison:

- The shape of grains are similar, maybe there are too many grains in my app so some grains are a bit smaller,
- In app there is dual phase structure chosen, so we can see pink grains in microstructure (Img 11),
- In application the pink grains simulate and behave like martensite and other grains like ferrite (similar to real dual phase steel),
- Difference is between ferrite representation, in app all grains except from pink grains should have one state. Thanks for that these two microstructures could look very similar.

5. Conclusions

- Simulations give results which are similar to real microstructures
- Application can be used to solve real problems, is expandable if new functionalities will be needed
- Export and import give opportunity to generate microstructures from one computer and send for example as image to the other, where simulation can be completed.