Kamil Wójcikowski
279052

Multiscale modeling – Project report II

1. General project description

In this part of project, we are focuses on Monte Carlo method and static recrystallization algorithm. On this approach our application has implemented some particular functionalities that will be describe below.

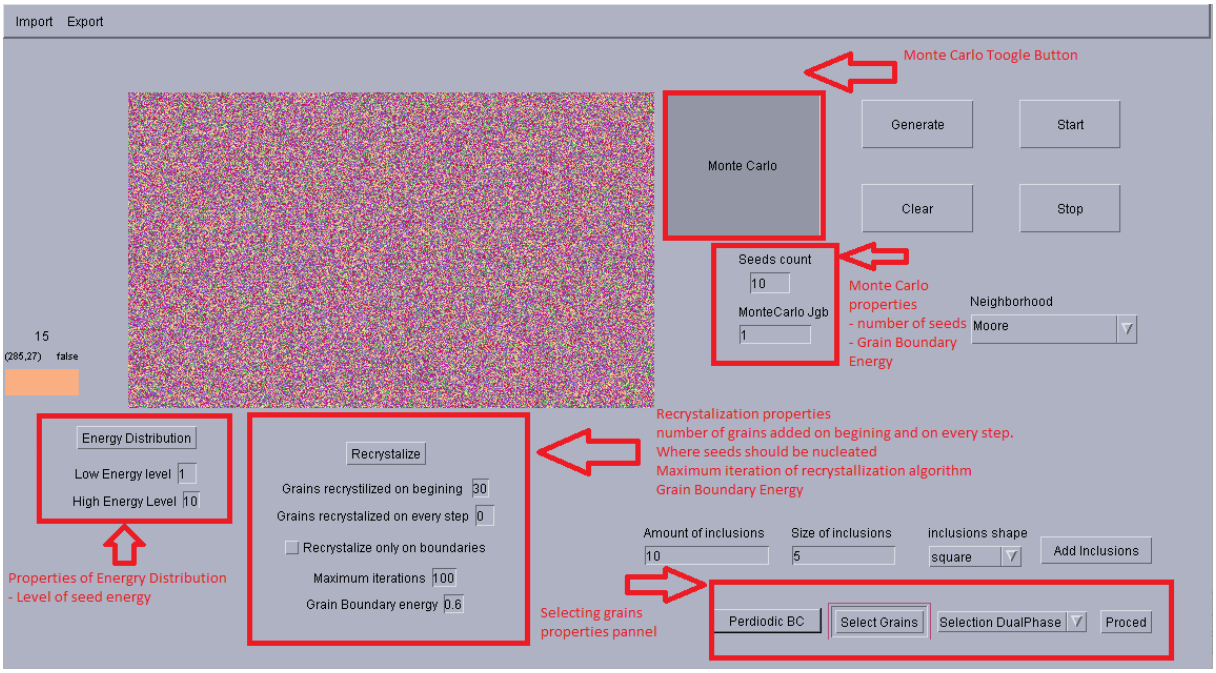

Fig 1. General interface view with description of functionality implemented in second part of project.

Monte Carlo grain growth algorithm

The Monte Carlo functionality user can enable by clicking in to “Monte Carlo” Toggle button (Fig. 1). The first step is to represent the material as a 2D for this step user have to generate microstructure by clicking “Generate”. After that the initial structure will be generate, every seed is chosen randomly (so there is no contiguous regions containing the same “id”) (Fig.2) .

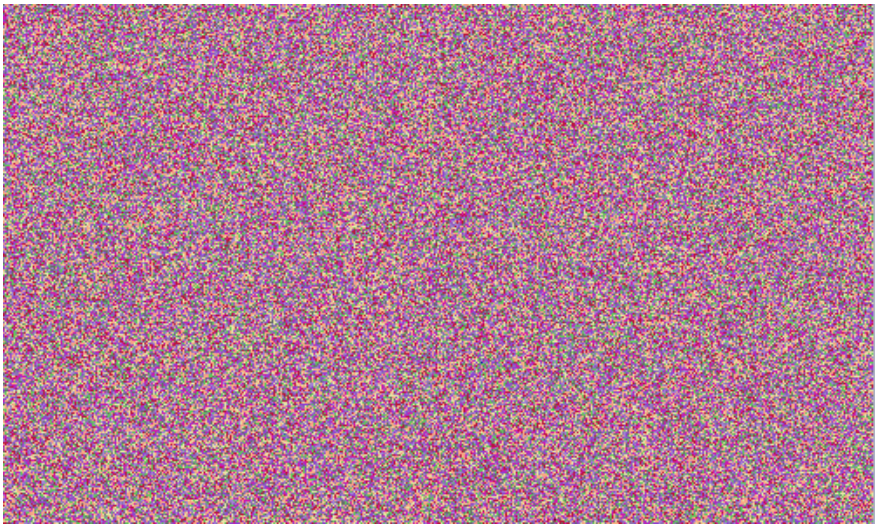

Fig 2. Initial microstructure prepared for Monte Carlo grain growth.

After this step the simulation itself begins. These are the four main steps of the algorithm:

a) Calculation of the free energy of an element of the matrix with its present crystallographic orientation based on its neighborhood.

b) Random choice of a new crystallographic orientation for that element

c) New calculation of the free energy of the same element, but with the new crystallographic orientation

d) Comparison of the two values. The orientation that minimizes the energy is chosen.

Presented steps are applied for every seed on the microstructure. Such number of activities is known as one iteration of Mc algorithm. Results of algorithm could be visible on Fig. 3 and Fig.4

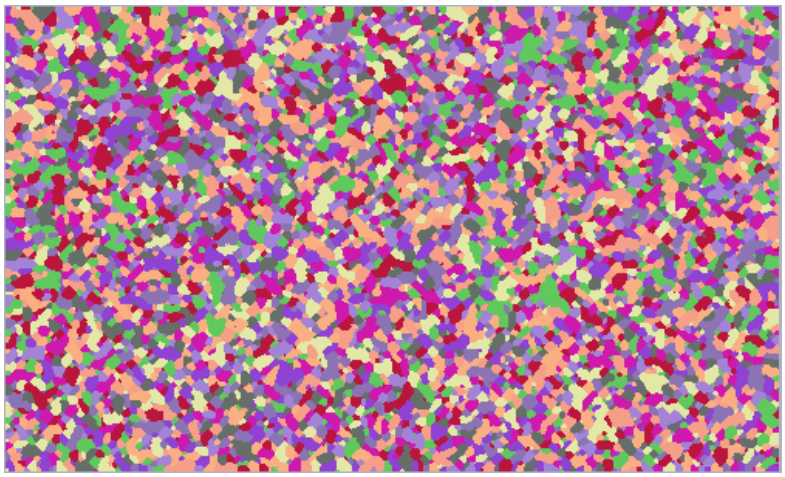

Fig 3. Microstructure after 10 Monte Carlo grain growth iterations.

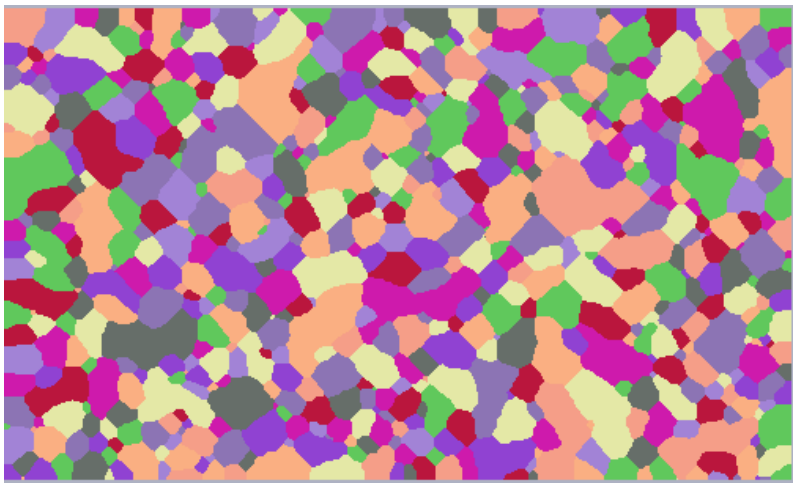

Fig 3. Microstructure after 100 Monte Carlo grain growth iterations.

As it is visible, grains are going to some order, smaller contiguous regions are connecting to bigger. The final result of MC method would be fully Heterogenous microstructure.

Dual phase microstructure: – (CA) -> (CA or MC) – (MC) -> (CA or MC).

Functionality allow user to choose some grains from generated microstructure after MC method or simple grain growth and remove others, then user have possibility to start nucleation process one more time and growth new generated seed but earlier chosen grains will still have same size and shape. Chosen grains could be treated as grains with different cellular ids (Dual Phase selection), or selected Grains could have same new selected id (New phase selection). User is choosing grains by clicking on the canvas. (Fig.4, Fig. 5)

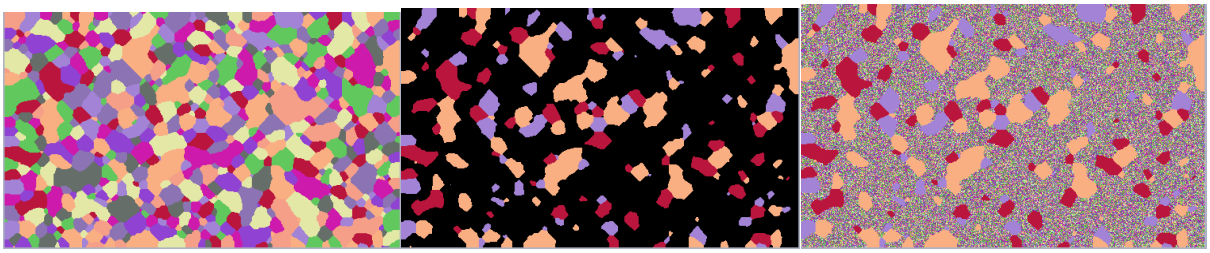

Fig 4. Selection of Dual phase section, nucleation and growth with MC algorithm usage

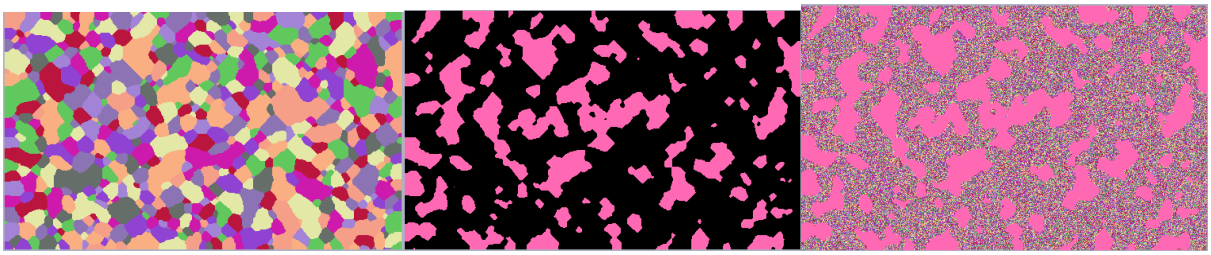

Fig 5. Selection of new phase section, nucleation and growth with MC algorithm usage

MC static recrystallization algorithm – energy distribution

Functionality allow user to take a look on how energy is distributed each cell. Cell that is on the boundary of grains have bigger energy than cell from contiguous regions. User could define level of energy assigned to cells by providing data in “Properties of energy distribution Panel” (Fig. 1). Energy have influence on static recrystallization algorithm work. Visualization windows could be open by clicking on the “Energy Distribution Button”

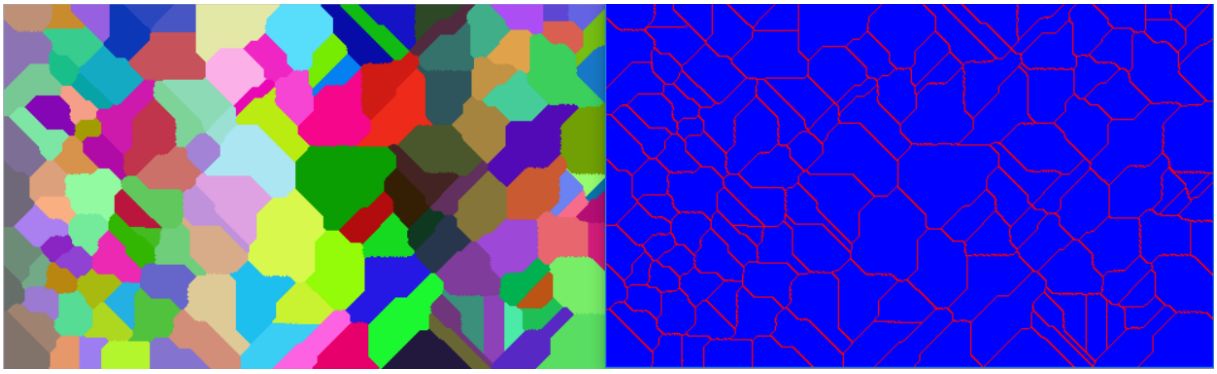

Fig. 6 Visualization of energy distributed over Heterogenous microstructure.

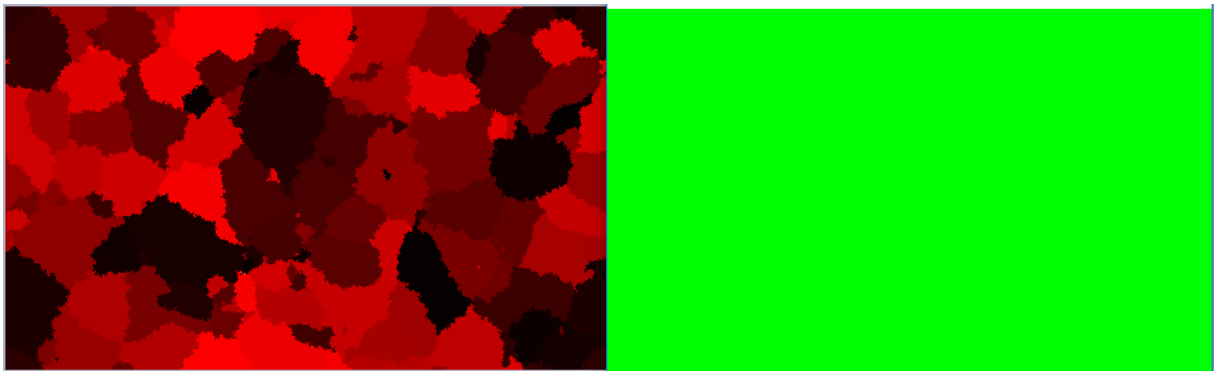

Fig. 7 Visualization of energy distributed over Homogenous microstructure.

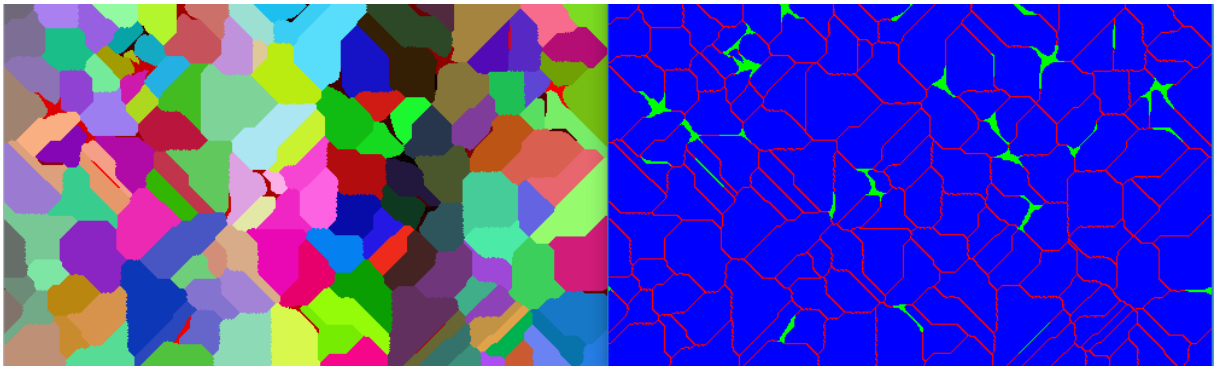

Fig. 8 Visualization of energy distributed over microstructure during static recrystallization algorithm.

MC static recrystallization algorithm – growth.

Whole procedure is divided in two steps, nucleation and growth:

- In the nucleation step new seeds appears on the microstructure. Seeds might be added on the borders of grains or on the continues regions of grains (Choose is make by user who could mark check box “Recrystallize only on boundaries” There are two types of nucleation, continues where on every iteration of Monte Carlo algorithm new recrystallized seed will appear. Second option is static nucleation where only on the beginning of algorithm recrystallized seeds will appear. Both of the types could be mixed.

- Second step is growth of recrystallized grains. This step could be divided on 4 more points
    - Calculation of the free energy of an element of the matrix with its present crystallographic orientation based on its neighborhood and distributed energy.
    - Random choice of a new crystallographic orientation for that element chosen from neighborhood that already are recrystallized (if there are not so points, we are choosing next element from matrix)
    - New calculation of the free energy of the same element, but with the new crystallographic orientation
    - Comparison of the two values. The orientation that minimizes the energy is chosen.

From the interface point of view, user have to first of all generate new microstructure that will be subjected to the process of static recrystallization. Next step for user will be to insert properties of algorithm as grain boundary energy, value of cells energy, number of recrystallized seeds on beginning and on each of step and so one. After that user could start computing by clicking into “Recrystallize” button. The results of algorithm are shown on the Fig 8 and Fig 9.

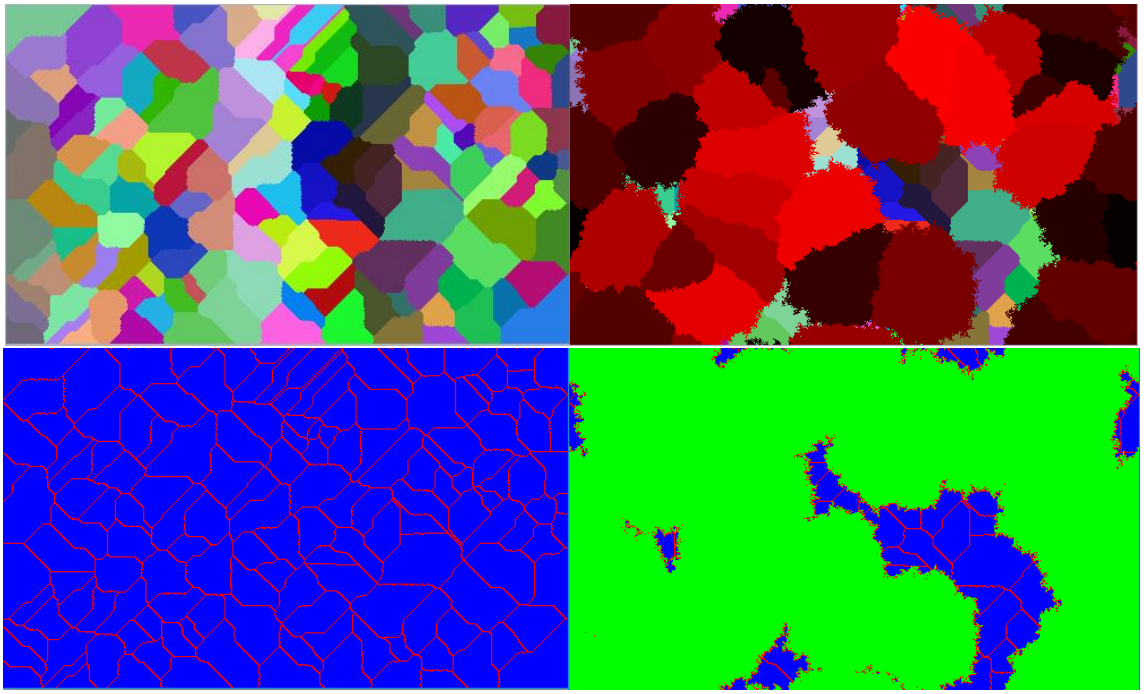

Fig. 8 Visualization of energy distributed over microstructure during recrystallization algorithm with static nucleation, Grain boundaries energy = 0.6, low energy value = 5, high energy value =7, Monte Carlo iterations = 10, Seeds recrystallized on beginning = 30 and during algorithm = 0

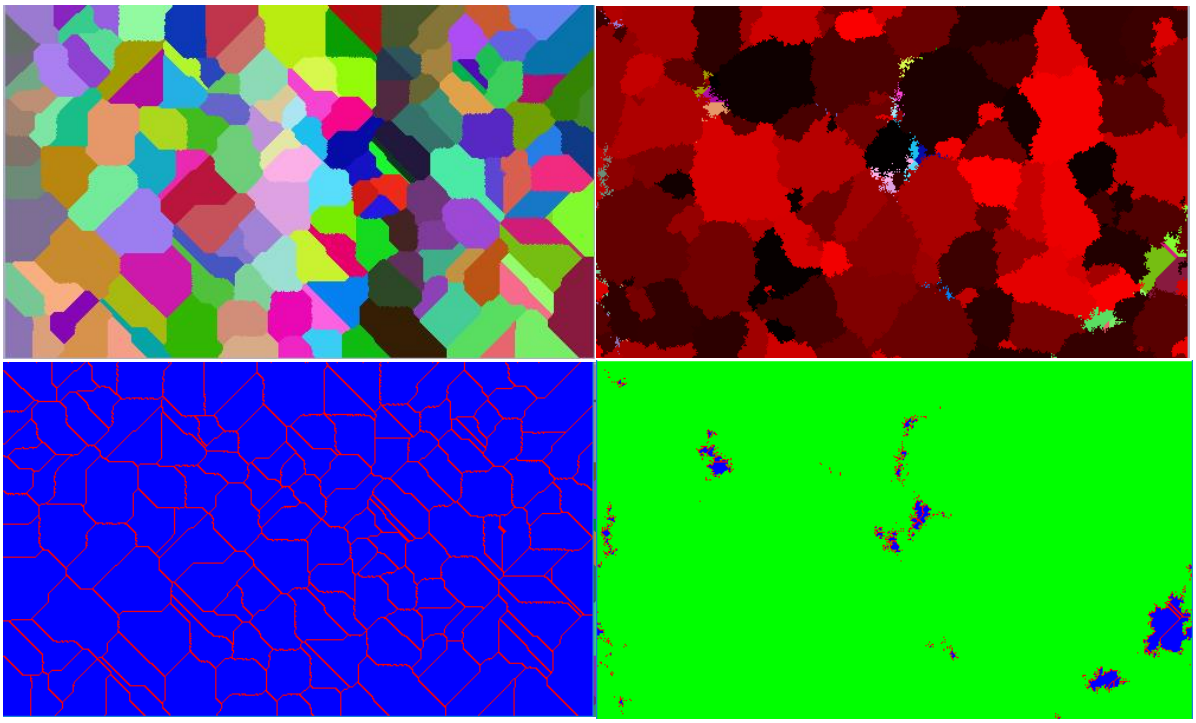

Fig. 9 Visualization of energy distributed over microstructure during recrystallization algorithm with static nucleation, Grain boundaries energy = 0.6, low energy value = 5, high energy value =7, Monte Carlo iterations = 10, Seeds recrystallized on beginning = 10 and during algorithm = 10

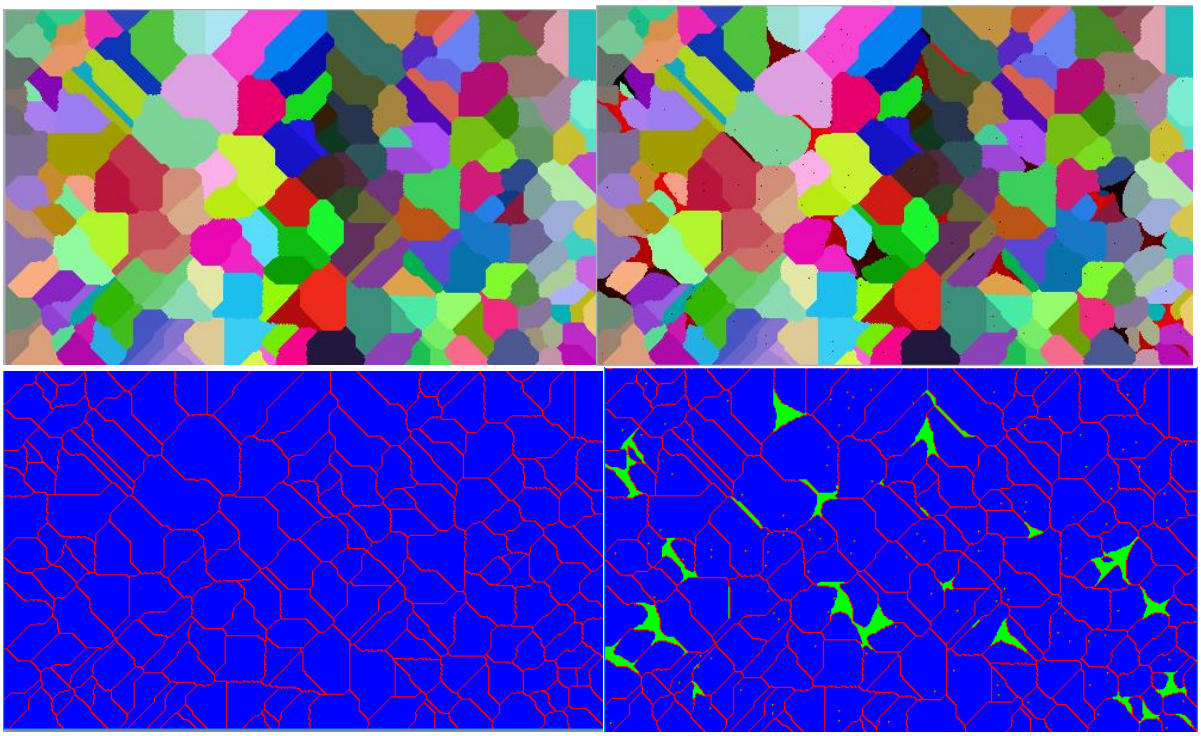

Fig. 10 Visualization of energy distributed over microstructure during recrystallization algorithm with static nucleation, Grain boundaries energy = 1, low energy value = 2, high energy value =3, Monte Carlo iterations = 10, Seeds recrystallized on beginning = 10 and during algorithm = 10

2.Technology used in project.

Project is written in Java. Only requirement to run program is Java Runtime Environment. To develop new functionalities Java SDK minimum version 8 is required. Project is cross platform software could be run on Windows OS and Linux as well. To make front end interface I used NetBeans which give easy and effective plugin to create Swing GUI.

3. Comparison with real microstructure photo

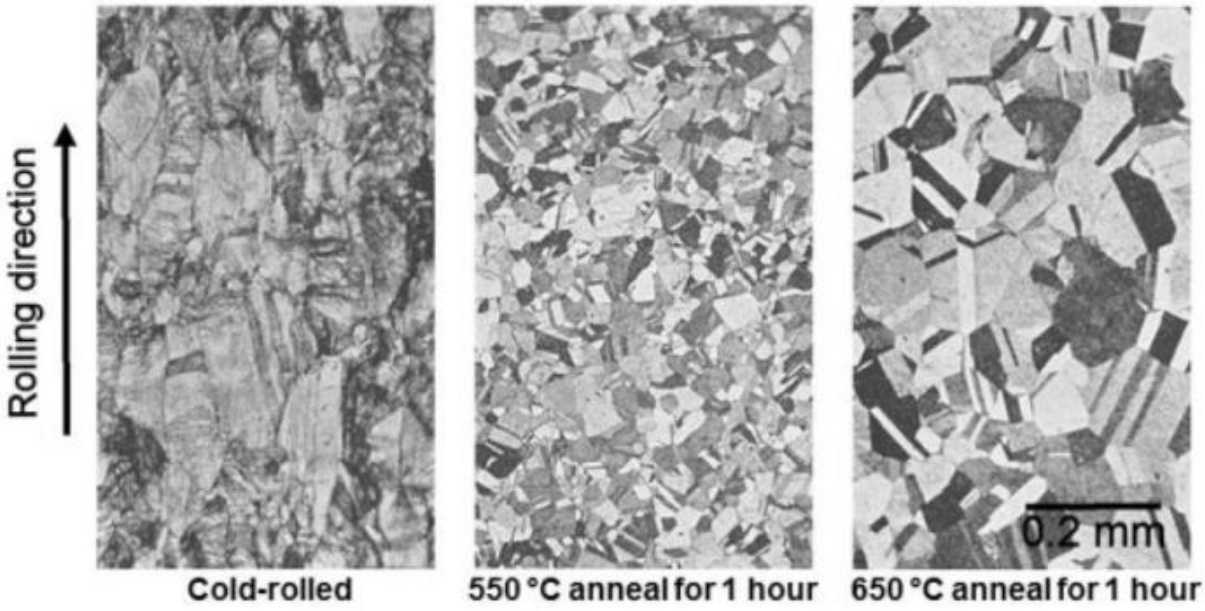

Fig.11 “Metallurgical effects of recrystallization anneal”
https://www.imetllc.com/training-article/recrystallization-anneal/

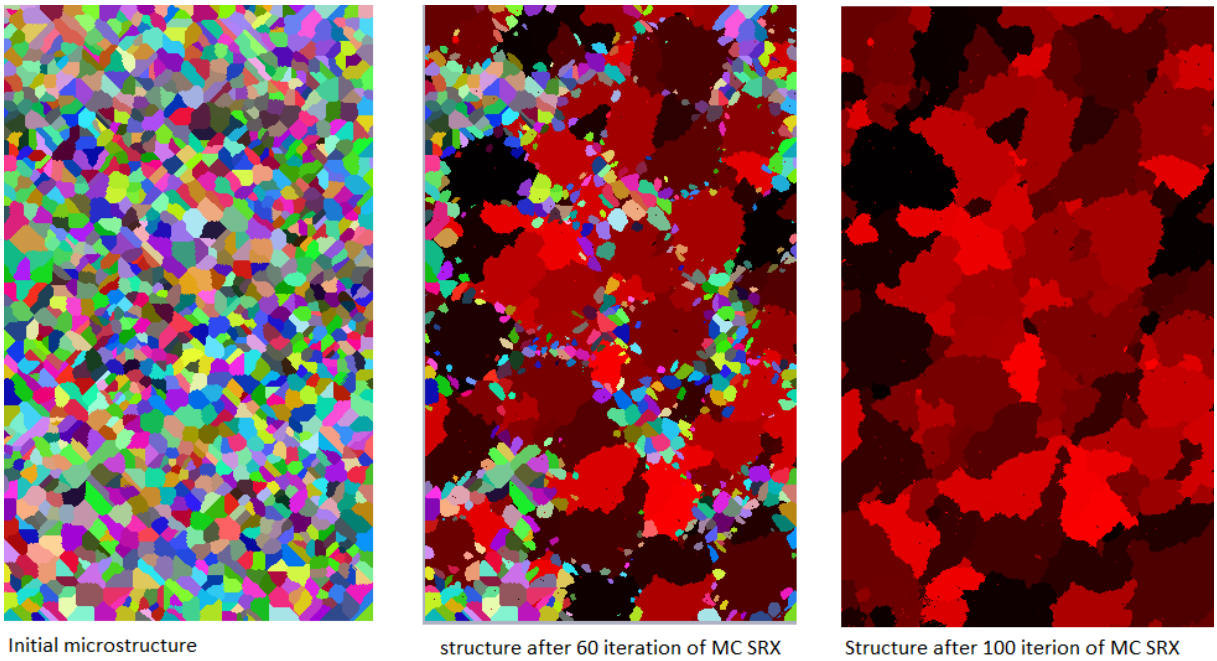

Fig.12 Simulation of recrystallization Grain boundaries energy = 1, low energy value = 2, high energy value =4, Monte Carlo iterations = 100, Seeds recrystallized on beginning = 30 during algorithm = 10

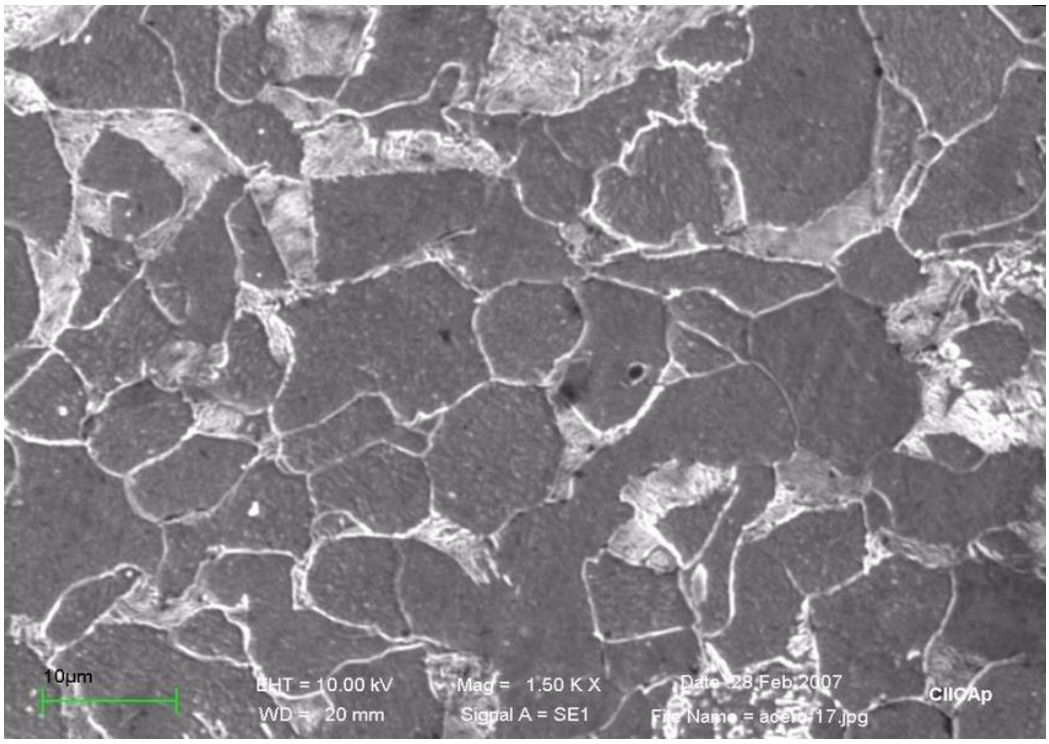

Fig. 14 Optical images of the microstructure of carbon steel. https://www.researchgate.net/figure/Optical-images-of-the-microstructure-of-carbon-steel_fig1_266258854

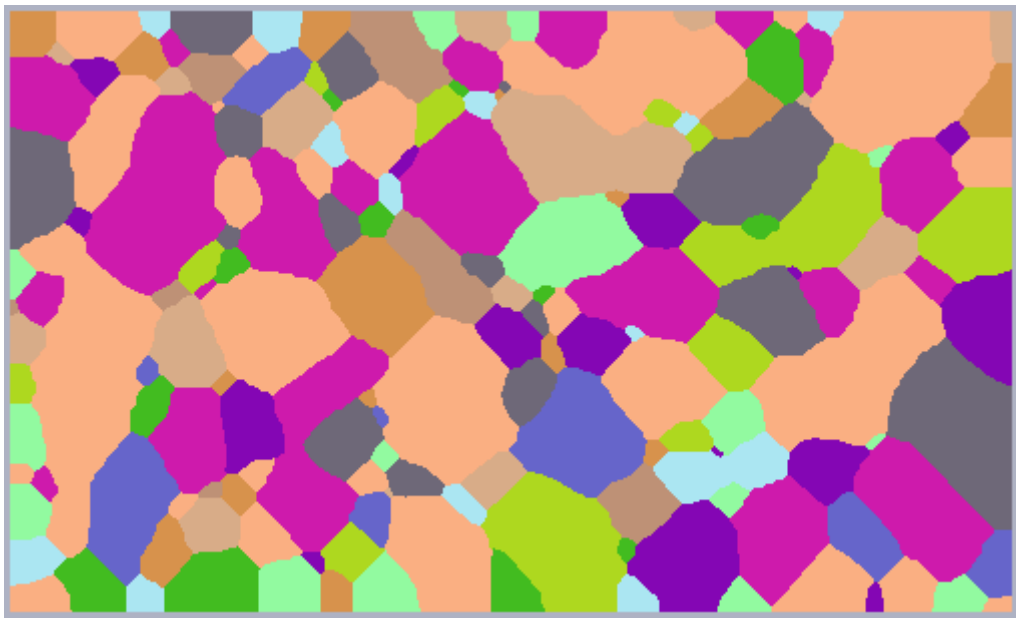

Fig. 15 Microstructure generated by use Monte Carlo algorithm (400 iterations)