# UncorrelaTEd

## Solid-liquid thermoelectric systems with uncorrelated properties

# Deliverable 5.1

# Advanced simulator

H2020-EU.1.2.1. - FET Open

FETOPEN-01-2018-2019-2020 - FET-Open Challenging Current Thinking

Grant Management 863222

Type of Action: RIA

Start Date: 01 Jan 2020

Duration: 48 months

**Project partners**

| LOGO | Partner full name | Acronym |
|---|---|---|
| | Universitat Jaume I | UJI |
| | Institut de Recerca en Energia de Catalunya | IREC |
| | Kungliga Tekniska Högskolan | KTH |
| | University of Warwick | UW |
| | Solvionic | SOLV |
| | Specific Polymers | SP |

**Deliverable Name:** Advanced simulator

**Led by:** UW

**Partners:** UW, UJI

| Version | Date | Changes |
|---|---|---|
| 1.0 | 30/11/2021 | Original version |
| 2.0 | 13/10/2022 | Instructions and validation of results are added. |

## 1 Introduction

A code has been developed for the electron transport in a complex nanostructured material. Using Monte Carlo electron transport formalism, it solves for the electron conductivity in a two-dimensional porous structure domain.

## 2 Simulator

Taking few material parameters of interest, it gives the pristine material transport distribution function and the electrical conductivity. From the analytical Boltzmann Transport equation and from our Monte Carlo approach, we obtain a multiplication factor that optimises the further calculation when we introduce porosity in the domain. With this, electronic conductivity can be found for the porous material.
The simulator can be accessed here:
Project: UncorrelaTEd (warwick.ac.uk)

## 3 References

Dhritiman Chakraborty, Joshua Brooke, Nicholas C S Hulse, and Neophytos Neophytou, "Thermal rectification optimization in nanoporous Si using Monte Carlo simulations", Journal of Applied Physics 126, 184303 (2019) https://doi.org/10.1063/1.5119806

Chakraborty, Dhritiman and Foster, Samuel and Neophytou, Neophytos, Monte Carlo phonon transport simulations in hierarchically disordered silicon nanostructures, Phys. Rev. B, 98, 11, (2018) 10.1103/PhysRevB.98.115435

Monte Carlo code user instructions:

Pankaj Priyadarshi and Neophytos Neophytou

**Instructions and guidance on how to use the code:**

Download the code and unzip.

Open Matlab (if already installed on your machine)

The code is executed by running the main file, '*main.m*' by clicking on 'Run' from the main Matlab window.

The Matlab command window indicates the simulation progress and the user needs to wait until 100% is completed.

**Input parameter setting:**

There are two types of inputs that you need to specify for the simulation. They describe the material parameters and the material geometry. They need to be modified from the default values to the ones for the material under consideration. These files are labelled: *'input_parameters.m'* and *'input_geometry.m'*, respectively.

Specifics for the input variables in *'input_parameters.m':*

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%   Input material and simulation parameters for MC ray-tracing
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%----- Material parameters ------------------------------------------------
%----- Non-polar Acoustic Phonon Scattering (ADP) -------------------------
m=1*m0; % Effective mass (kg)
density=2329; % Mass Density (kg/m3)
vs=5.22e3; % Sound Velocity (m/sec)
D_ac=4.317; % Deformation Potential (eV), in Joule (eV*q)

%----- Simulation parameters ----------------------------------------------
T=300; % Temperature in Kelvin
E_min=0.0001; % Minimum energy point (in eV)
E_max=0.2; % Maximum energy point (in eV)
NE=100; % # of energy points
Ef_min=-0.20; % Minimum Fermi energy level (in eV)
Ef_max=0.20; % Maximum Fermi energy level (in eV)
NEf=20; % # of Fermi energy points
Nele=10; % # of electrons per energy point
```

The first four parameters describe the material, which should be changed by the user to specify the material of interest. Above you see the default values. The second set of parameters are simulation physics specifics, and can remain as is.

Specifics for the input variables in *'input_geometry.m':*

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%     2D input geometry details for MC ray-tracing
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%----- Domain geometry ----------------------------------------------------
Lx=1000e-9; % Length in x-direction (length)
Ly=500e-9; % Length in y-direction (width)
dLx=2e-9; % Step size in x-direction
dLy=2e-9; % Step size in y-direction
%% ----- Details of nanocrystalline grains --------------------------------
include_gb='yes'; % Include grain boundaries ? TYPE 'yes' or 'no'
if strcmp(include_gb,'yes')
    n_grain=100; % # grain seeds
end
%% ----- Details of porous geometry ---------------------------------------
include_pores='no'; % Include pores ? TYPE 'yes' or 'no'
if strcmp(include_pores,'yes')
%----- Select only one type of pores arrangement -------------------------
    ordered_pores='yes'; % TYPE 'yes' or 'no'
    staggered_pores='no'; % TYPE 'yes' or 'no'
    nonoverlap_random_pores='no'; % TYPE 'yes' or 'no'
    nonoverlap_random_size_pores='no'; % TYPE 'yes' or 'no'
    overlap_random_size_pores='no'; % TYPE 'yes' or 'no'
    nonoverlap_random_oval_pores='no'; % TYPE 'yes' or 'no'
end
%-------------------------------------------------------------------------
%----- Number of pores as an input ---------------------------------------
%..... For orderd and staggered pores ....................................
n_pore_x=10; % # of pores in x-direction
n_pore_y=5; % # of pores in y-direction
%..... For randomly distributed pores ....................................
n_pore=50; % For random pores
%----- Radius of pores as an input ---------------------------------------
%..... For equal size of pores ...........................................
r=10e-9; % Radius of pore
%..... For unequal size of pores .........................................
r_min=1e-9; % Minimum radius of a pore
r_max=3e-9; % Maximum radius of a pore
%% ----- Include both GB and pores in the geometry ----------------------
include_gb_pores='no'; % Include combination of GB and pores ? TYPE 'yes' or
'no'
```

The first four parameters are the domain dimension. The flag '`include_gb`' allows for the inclusion of grain boundaries, and asks for the number of seed points. The flag '`include_pores`' allows the insertion of pores and is followed by specifics on the type and density of pores. Comments in the code are explaining the details.

Typical examples of a nanocrystalline geometry, a porous geometry and a porous + crystalline geometry are shown below:

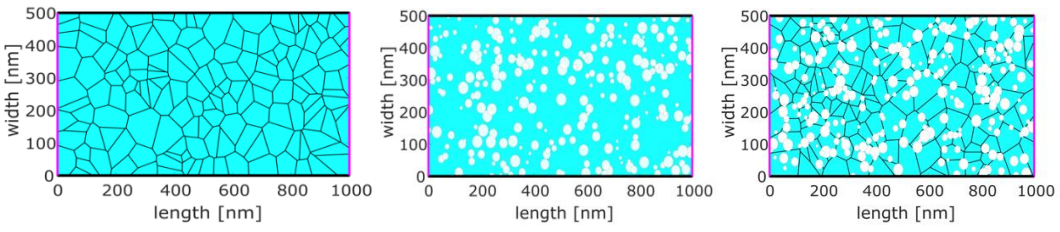


Fig. 1: Schematic of complex nanostructured material in 2-dimensional domain populated with (a) grain boundary, (b) mesopores, and (c) combination of pores & grain boundary.

**Results and output illustration of the simulation:**

To see the geometry simulated, run the 'plot_domain_meshgrid.m' file. Typical figures are as shown above.

The main output of the code is saved in the variable 'flux' and can be plotted versus energy 'E'. The 'flux' is then used in post-processing to extract the conductivity (σ), Seebeck coefficient (S), and Power Factor (PF). For this, run the *'plot_TE_coefficient.m'* file. Typical outputs are shown below for the flux, σ, S, PF, plotted versus the Fermi energy *'Ef'*.

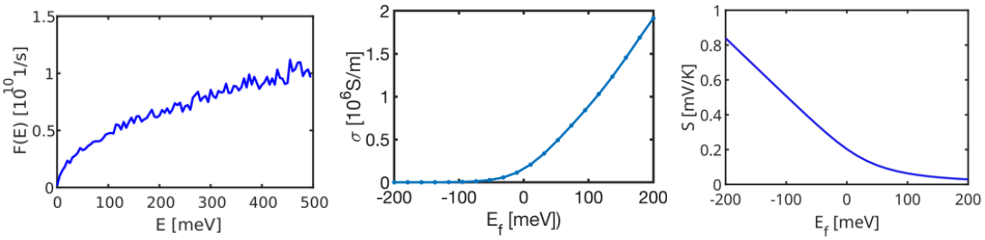


Fig. 2: (a) An example flux vs. energy plot. (b) Calculated MC conductivity and Seebeck coefficients as a function of Fermi energy Ef.

**Validation results:**

To validate the code, we extract the TDF, σ, S, PF, from analytical BTE, and compare these with the Monte Carlo simulated results for a pristine Si material. The results are matching extremely well, as seen from the figures below:

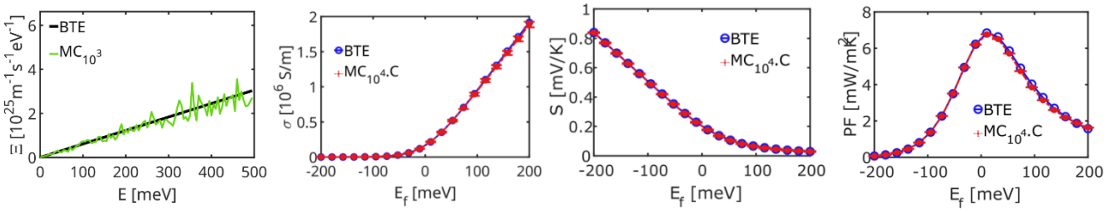


Fig. 3: (a) Electrical conductivity, (b) Seebeck coefficient and (c) power factor for Si (as an example) in pristine domain as a function of Fermi energy Ef.

We also perform simulations for nanostructures in for the geometries shown above and compare the pristine material performance with the nanostructured ones. A crucial test for the validity of the code is how much the results from the different individual nano-features (grain boundaries and pores) can be combined using the analytical Matthiessen's rule, and whether Monte Carlo simulations provide something close to that. As seen from the conductivity of the structure combining both pores and grain boundaries, it is very close to the analytical one, which validates the behavior of the advanced Monte Carlo simulations for highly nanostructured materials.

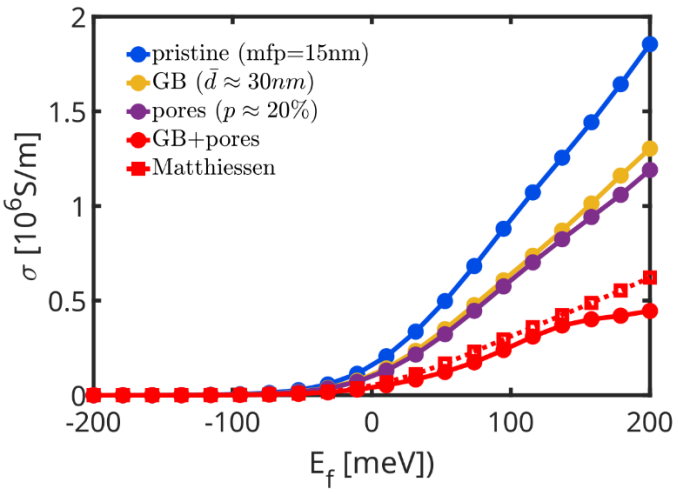


Fig. 4: Electrical conductivity calculated from MC algorithm in nanostructured material domain populated with grain boundaries, pores and combination of both as a function of Fermi energy. The red dotted line shows the calculated Matthiessen's conductivity that found a good agreement with the simulated result.