Manual for the script MonteCarlo_run.m

Example: Ar/N2 (30%/70%) mixture at E/N=100 Td. The script "/Ar_N2_mixture/MonteCarlo_single_run" can be tested without any modifications.

For new gases, the following steps have to be followed to **perform a single simulation** (one E/N value):

1. Download the cross section files from LXcat: http://fr.lxcat.net/data/set_type.php and **save each cross section .txt-file for each gas separately in a directory**. E.g. for the Ar/N2 mixture: / N2_Biagi/*.txt and/ Ar_Biagi/*.txt. The name of the .txt-file is not important.
2. create a directory where to perform the simulation, e.g. "/Ar_N2_mixture"
3. copy the script MonteCarlo_run.m into the directory and open it
4. set paths for the following two directories correctly:
    a. **functionsDir**: path of the directory "_functions" (should be already correct).
    b. **gasDir**: directories of the cross section input *.txt files of the gases, here:

    gasDir = {'.._Xsection\Ar_Biagi','.._Xsection\N2_Biagi'}

5. set the following parameters for the simulation:
    a. **gas**: sum formulas of the gas components as a cell array, here:

    gas = {'Ar','N2'}

    b. **mix**: mixing ratio of the gas components as a vector, here:

    mix = [0.3 0.7]

    c. **EN**: reduced field E/N in Townsend, here:

    EN = 100;

    d. **energy**: this is the vector of the energy range in eV (here: [0:0.01:100]) in that cross sections are considered. The speed of the simulation can increase when lowering the maximum energy. For high E/N-values, when electron energies exceed the maximum energy, the latter might need to be increased. The user will be informed if that happens in the command window and in the temporal.txt file by the messages:" E_max might be too small!"
    e. The variable **interactive** plots the temporal data if set to 1. If it is set to zero no plots will be performed.
    f. There are a few other settings, which in general do not have to be changed to obtain steady-state transport data. For more information, type "doc MonteCarlo" into the command window.
6. **run** the script **MonteCarlo_run.m**
7. Check cross sections in plot and **press the button "continue"** if ok

The results are saved during the simulation in the file **results.mat.** The development of the results (mean energy, drift velocities, diffusion constants and effective ionization rate constants) during the simulation can be followed in the command window as well as in the file **temporal.txt**. The simulation can be stopped at any time without losing the results.

The **results.mat** file is a matlab-structure and contains:

1. the input parameters **E/N**, **gas**, **mix** and the **dir** (directories of the cross sections)
2. **energy data E**: mean energy **E_mean**, electron energy distribution function **EEDF** and electron energy probability function **EEPF** with the corresponding **energy**
3. **transport data bulk:** drift velocities **w** in x-,y- and z-direction, density reduced diffusion constants **DN** in x,y and z-direction
4. **transport data flux:** drift velocities **w** in x-,y- and z-direction, density reduced diffusion constants **DN** in x,y and z-direction
5. **reaction rate constants**:
    a. calculated via counting of ionization/attachment events: **count**
    b. calculated via convolution of cross sections with EEDF: **conv**

    Both methods contain the total ionization rate constant **ion_tot**, total attachment rate constant **att_tot** and effective ionization rate constant **eff**