



TREKIS-3: **Time Resolved Electron Kinetics in SHI-Irradiated Solids**

Current version: 3.3.0 (update 10.07.2024)

A transport Monte Carlo code of event-by-event simulation of swift heavy ion penetration in matter, secondary electrons, holes and photons cascades, and energy transfer to the lattice

N. Medvedev^{1,2,*}, R.A. Rymzhanov^{3,4}, A.E. Volkov⁵

with help of D. Zainutdinov, F. Akhmetov, P. Babaev, S. Gorbunov

- 1) Institute of Physics, Czech Academy of Sciences, Na Slovance 2, 182 21 Prague 8, Czech Republic
- 2) Institute of Plasma Physics, Czech Academy of Sciences, Za Slovankou 3, 182 00 Prague 8, Czech Republic
- 3) Joint Institute for Nuclear Research, Joliot-Curie 6, 141980, Dubna, Moscow Region, Russia
- 4) The Institute of Nuclear Physics, Ibragimov St. 1, 050032 Almaty, Kazakhstan
- 5) P.N. Lebedev Physical Institute of the Russian Academy of Sciences, Leninskij pr., 53, 119991 Moscow, Russia

* Email: nikita.medvedev@fzu.cz

Table of Contents

Table of Contents.....	2
I. Disclaimer, how to cite	3
II. Brief introduction into the model.....	4
III. Files of the code.....	5
1. Source files	5
2. Input data	5
1. File INPUT_PARAMETERS.txt.....	5
2. INPUT_CDF	9
3. INPUT_DOS.....	13
4. INPUT_EADL	13
5. Others.....	13
3. Compiling and executing	13
A. Compiling without parallelization	13
B. Compiling with OpenMP.....	14
C. Compiling and executing with MPI.....	15
IV. OUTPUT DATA.....	16
1) Error log file.....	16
2) Directory with differential cross sections	16
3) File with CDF coefficients.....	16
4) File with reconstructed optical CDF.....	16
5) Files with mean free paths	17
6) Files with ranges	17
7) File with density of states	17
8) Valence holes self-diffusion coefficient	18
9) SHI mean free paths, energy losses, and ranges	18
9) Directory with results of MC simulations.....	18
10) File !Parameters.txt.....	19
11) Total numbers and energy.....	19
12) Files with radial distribution of electrons and their energy	19
13) Files with radial distribution of photon and their energy.....	19
14) Files with holes radial distributions and their energy	19
15) File with radial distribution of the energy transferred to lattice	20
16) File with radial distribution of the total energy transferred to lattice	20
17) Files with electron and VB holes spectra.....	20
18) File with electron velocity distribution by azimuth angle (theta).....	20
19) File with valence-band holes velocity distribution by azimuth angle (theta).....	20
20) Gnuplot scripts and plots	20
V. Combining TREKIS-3 with MD via output-input.....	21
VI. Consistency checks and known problems	22
VII. Appendix.....	23
1. Comparison of automatic single-pole approximation with fitted CDF	23
2. Comparison of numerical integration vs. interpolation	24
VIII. References.....	25

I. Disclaimer, how to cite

Although we endeavour to ensure that the code TREKIS-3 and results delivered are correct, no warranty is given as to its accuracy. We assume no responsibility for possible errors or omissions. We shall not be liable for any damage arising from the use of this code or its parts or any results produced with it, or from any action or decision taken as a result of using this code or any related material.

This code is distributed *as is* for non-commercial peaceful purposes only (see GPL-3.0 license), such as research and education. Additionally, it is explicitly **prohibited** to use the code, its parts, its results or any related material for military-related and other than peaceful purposes.

By using this code or its materials, you agree with these terms and conditions.

The use of the code is at your own risk. Should you choose to use it, please use appropriate citations:

The code itself:

Medvedev, N., Rymzhanov, R., & Volkov, A. (2023). TREKIS-3 [Computer software].
<https://doi.org/10.5281/zenodo.8394462>

Literature:

1) N. A. Medvedev, R. A. Rymzhanov, A. E. Volkov, *J. Phys. D. Appl. Phys.* **2015**, 48, 355303

2) R. A. Rymzhanov, N. A. Medvedev, A. E. Volkov, *Nucl. Instrum. Methods B* **2016**, 388, 41

Should you use this code to create initial conditions for further molecular dynamics simulations of atomic response to the electronic excitation by a swift heavy ion (e.g. with LAMMPS), the following citation is required:

3) R. Rymzhanov, N. A. Medvedev, A. E. Volkov, *J. Phys. D. Appl. Phys.* **2017**, 50, 475301

In a publication, we recommend that at least the following parameters should be mentioned for reproducibility of the results: material, its structure, density, speed of sound, the used CDF coefficients, which processes were included (active) in the simulation, ion type, its energy, the model for SHI charge, number of MC iterations.

II. Brief introduction into the model

TREKIS-3 (Time-Resolved Electron Kinetics in SHI-Irradiated Solids, version 3) is the transport Monte Carlo code using event-by-event (analog) simulation of effects of swift heavy ion (SHI) impacts on matter, induced secondary electron and hole cascades, and energy transfer to the lattice/atomic system [1,2]. A swift heavy ion is typically defined as an ion with energy $E > 1$ MeV/amu. Electronic energy losses by such ions constitute over 95% of the energy deposition, whereas energy transfer to target atoms is almost negligible along the entire SHI path (except near the end of the range). Depending on the target, the ion energy and mass, the electronic stopping power may reach 5-50 keV/nm along the ion trajectory. For a review of typical effects of SHI on matter, the reader may have a look at the review [3]; for a review of modeling of SHI effects with help of TREKIS-3 see [4].

TREKIS-3 is capable of modelling ions from hydrogen (proton) to super-heavy elements, however, for protons, only energy above the Bragg-peak are calculated reliably. SHI energies are limited to the regime of nonrelativistic electronic energy losses (from ~ 1 MeV/amu up to ~ 100 GeV/amu).

The used scattering cross sections are based on the formalism of the dynamic structure factor proposed by L. van Hove [5], or, equivalently, the loss function (the inverse imaginary part of the complex dielectric function, CDF) in the linear response theory developed by Fano [6]. Ref.[1] contains CDF parameters for a number of materials, extracted from optical data following the Ritchie-Howie formalism [7], and provided as input files together with the code (see below). The procedure of reconstruction of the CDF is described in detail in Refs. [1,2,7].

Based on these cross sections, TREKIS-3 models the following processes: (a) penetration of a swift heavy projectile resulting in ionization of a target and appearance of primary electrons (δ -electrons) and holes; (b) scattering of δ -electrons on lattice atoms and target electrons as well as the kinetics of all secondary generations of electrons; (c) Auger decays of core holes, also resulting in production of secondary electrons; (d) radiative decays of core holes, following photon transport and photoabsorption exciting new electrons and holes; (e) valence holes transport and their interaction with target atoms. All details of the numerics used and corresponding cross sections for different processes can be found in Refs. [1,2].

III. Files of the code

The code consists of a few modules, using the following internal conventions:

- 1) All global variables start with "g_", e.g. g_numpar, and all defined in the module "Variables"
- 2) All modular variable names are defined starting as "m_", e.g. "m_number"
- 3) Subroutines and functions should be provided with a comment specifying in which module it can be found

1. Source files

All the f90-files are stored in the directory [Source_files](#):

[Analytical_IMFPs.f90](#) – The module contains subroutines for calculations and saving of cross sections, mean free paths, and ranges for all particles (SHI, electron, hole, photon)

[Cross_sections.f90](#) – The module contains subroutines for evaluation of cross sections

[Dealing_with_EADL.f90](#) – The module contains subroutines for accessing EPICS[†] databases EADL and EPDL, containing atomic parameters (Auger and radiative decay times, ionization potentials, kinetic energies of atomic shells)

[Gnuplotting_subs.f90](#) – The module contains subroutines for creating gnuplot scripts

[Monte_Carlo.f90](#) – The module contains all subroutines for Monte Carlo modelling of particles transport

[Objects.f90](#) – The module contains definitions of all objects and some subroutines to deal with them

[Reading_files_and_parameters.f90](#) – The module contains subroutines for reading input files

[Sorting_output_data.f90](#) – The module contains subroutines for creating output files

[Thermal_parameters.f90](#) – The unfinished module for calculation of thermal parameters (unused, almost)

[Universal_Constants.f90](#) – The module contains all universal constants as global variables

[Universal_MC_for_SHI_MAIN.f90](#) – This is the main file of the TREKIS-3, assembling the calculations, reading input and creating output files

[Variables.f90](#) – The module contains all global variables

2. Input data

1. File INPUT_PARAMETERS.txt

```
Au_test2      ! material name
54            ! SHI atomic number
200           ! [MeV] total SHI energy
-1            ! [proton mass] if mass of the SHI is NONSTANDARD isotope; if standard, put any NEGATIVE number
10.0          ! [fs] total time analyzed
10.0d0 1      ! [fs] timestep, how often the data are saved; kind of time-grid: 0=linear, 1=logarithmic
10.0d0        ! [eV] Cut-off energy (electrons with lower energies are excluded from calculation)
10.0          ! [Å] thickness of the layer analyzed
0.0           ! [K] temperature of the target
0 1.0         ! 0=Barkas, 1=Bohr, 2=Nikolaev-Dmitriev, 3=Schwiete-Grande, 4=fixed Zeff; Fixed value (only for case=4)
0            ! 0=point-like charge; 1=Brandt-Kitagawa ion
1 2          ! elastic cross sections: -1=disable; 0=Mott, 1=optical CDF; Target Zeff (0=Barkas-like; 1=(Z-1); 2=Z^2/CDF_e)
1 0          ! target dispersion (1=free el., 2=plasmon-pole, 3=Ritchie); m_eff [me] (0=effective mass from DOS of VB; -1=free-electron)
0            ! include plasmon integration limit
-1.0d30       ! [me] effective valence hole mass in units of electron mass
1            ! include radiative decay of deep holes? (0=no, 1=yes)
-1.0 10.0d0 6.18d0 ! [eV] Work function (<=0 - no emission); [Å] surface barrier length; [eV] barrier height for electron emission
1            ! number of MC iterations to be performed
-1           ! number of threads for parallel calculations with OpenMP (1=nonparallelized; -1=all available threads)

!-----
! Optional flags:

!verbose      ! To printout time of reaching each time-grid point
grid 1        ! E-grid index: -1=old; 0=new; 1=save-files

DOS Au.dos    ! File with DOS
print_CDF     ! printout cdf file

!CDF Au_test.cdf ! File with CDF
!print_optical_cdf ! printout optical CDF reconstructed from the fitted loss-function
!redo_MFP      ! recalculate mean free paths for all
!redo_MFP_SHI  ! recalculate mean free paths for SHI
!redo_IMFP     ! recalculate inelastic mean free paths for electrons and VB holes
!redo_EMFP     ! recalculate elastic mean free paths for electrons and VB holes
!gnuplot jpeg  ! gnuplot plots in format: jpeg; png; gif; eps; pdf; mo; (by default is it jpeg)
!very-verbose  ! To printout time of each collision (really A LOT of printout!)
40 !info       ! printout some basic info about the code
```

Figure 1. An example of the file Input_Parameters.txt

[†] <https://www-nds.iaea.org/epics/> This database is public domain and is freely distributed by its author at <http://redcullen1.net/HOMEPAGE.NEW/>

The main input file, where the user sets the material and SHI parameters, is the following:

[INPUT_PARAMETERS.txt](#) – the file containing the parameters of the calculations: material, ion, numerical parameters, etc., see an example in Figure 1.

It must contain the following lines, exactly in this order and format:

- 1) Material name (unless CDF flag is specified, this material name must match the name of the file `[material].cdf` with materials parameters, described below)
- 2) Swift heavy ion (SHI) atomic number according to the Periodic Table
- 3) SHI energy in [MeV]
- 4) Mass of the SHI in atomic units. Positive number sets the mass, negative number sets the mass according to the most abundant isotope. These masses are taken from the input file [INPUT_atomic_data.dat](#) in the directory [INPUT_EADL](#), see below.
- 5) Total time of the analysis in [fs], setting when the simulation stops.
Note: Typically, electron cascades are finished within ~100 fs.
- 6) Two numbers:
 - a) Time step in [fs], how often the data are printed out;
 - b) what kind of the time-grid to use for output printing:
0=linear grid;
1=logarithmic grid.
Note: too fine grid results in too long calculation times and/or memory overload. Logarithmic grid is recommended, or printing out only the final time instant of the simulation.
- 7) Cut off energy for stopping of electrons in [eV]. Electrons and valence holes with energies below this value are stopped in the simulation. To exclude this artificial stopping, set any negative number
- 8) Thickness of the materials to be simulated in [Å]. SHI penetration is modelled within this layer, while all secondary cascades assume periodic boundaries along the ion penetration.
Note: too large thickness results in too large numbers of secondary particles created, too long calculation times and/or memory overload. The recommended value is 10 Å.
- 9) Target temperature in [K]. The temperature is set in the temperature factor of the cross section, but not in the CDF itself, for details see [2].
- 10) Two numbers:
 - a) which model for SHI effective charge to be used [1]:
0=Barkas;
1=Bohr;
2=Nikolaev-Dmitriev;
3=Schiwietz-Grande,
4=fixed Zeff;
 - b) If the fixed value of the SHI charge is used (case=4), the second number set the charge in the units of electron charge.
Note: it is recommended to use option 0 (Barkas charge), or a specific fixed charge if known from experiments.
- 11) Index of the model of the SHI charge spatial distribution:
0=point-like charge;
1=Brandt-Kitagawa ion.
Note: option 0 is recommended as a faster one, whereas option 1 produces only a minor improvement around the Bragg-peak, see [8].
- 12) Two numbers:
 - a) Index of the model of electron and valence hole elastic scattering:

- 1 = disable elastic scattering;
- 0 = Mott (modified Molier-screened Rutherford),
- 1 = optical phonons CDF [1].
- 2 = DSF (*highly experimental option, requires additional input files; not provided*)

b) Index of which target atoms' effective charge to use:

- 0 = Barkas-like effective charge [9],
- 1 = fixed charge $Z=1$ (used in previous versions of TREKIS-3 and old calculations),
- 2 = dynamical screening of the nuclear charge via valence (conduction) band CDF and form-factors for core shells (similar to Chihara's formalism [10], but empirically adjusted to work for electron scattering).

Note: option 2 is the recommended choice, because it usually produces the best elastic scattering cross section across a wide range of energies (from phonon scattering at low energies to unscreened scattering on a nucleus at high energies).

13) Two numbers setting inelastic scattering cross section:

a) the first one sets the kind of target dispersion relation (recommended option is 1):

- 1=free electron,
- 2=plasmon-pole,
- 3=Ritchie [1,2];

b) effective mass of valence hole used in CDF scattering cross-section calculations:

- 0=from DOS of VB within effective one-band approximation [2];
- 1=free-electron;
- positive number sets a fixed value in free-electron masses.

Note: the following options are recommended: a) 1; b) 0.

14) Index, setting whether to include plasmon integration limit in the inelastic scattering cross section or not: 0=no, 1=yes [2].

Note: option 0 is recommended.

15) Effective valence hole mass in units of free-electron mass, used for transport of valence holes.

Note: setting any negative number means the valence hole mass is estimated from the valence-band DOS within the single-band approximation.

16) Index, whether to include radiative decay of deep holes or not: 0=no, 1=yes.

Note: option to exclude is only used for testing, however including radiative decays and ensuing photon transport does not significantly affect the results for nonrelativistic particles.

17) Three numbers setting a model for electron emission from the surface. If on, periodic boundaries are switched off in the simulation, and electrons are allowed to be emitted from the top and the bottom of the layer (along Z-axis, direction of SHI penetration). The three numbers set:

- a) Work function in [eV] (if negative number is set, ≤ 0 , no emission is used, and periodic boundaries are used instead for bulk simulation);
- b) surface barrier length in [\AA];
- c) barrier height for electron emission in [eV]. For details of the model, see [11].

18) Number of MC iterations to be performed.

Note: at least 1000 iterations is recommended for reliable statistics.

19) Number of threads for parallel calculations with OpenM.

Note #1: 1 sets nonparallelized calculations.

Note #2: any negative number sets the number of threads equal to the number of nodes on your computer.

Note #3: this parameter is only used if the code is compiled with OpenMP; it is ignored if TREKIS is compiled with MPI, or without parallelization

After the line #19, one may set optional flags:

- **redo_MFP** – forces TREKIS to recalculate all mean free paths (MFP), even if the files with them are already present (useful if material or CDF parameters have been changed).
- **redo_IMFP** – forces TREKIS to recalculate electron (and hole) inelastic mean free paths.
- **redo_EMFP** – forces TREKIS to recalculate electron (and hole) elastic mean free paths.
- **redo_MFP_SHI** – forces TREKIS to recalculate SHI's mean free paths.
- **CDF CDF_file** – specifies the name *CDF_file* of file containing the CDF-parameters of the material, to be found in the directory 'INPUT_CDF'; if not specified, the default name is used (see below).
- **DOS DOS_file** – specifies the name *DOS_file* of file containing VB DOS, to be found in the directory 'INPUT_DOS'; if not specified, the default name is used (see below).
- **grid ind** – flag to choose which type of integration of the CDF-based scattering cross section by energy is to be used (for elastic and inelastic scattering of electrons and VB holes): Parameters *ind* (integer) sets:

-1 is to use the old default grid (used in TREKIS-3 versions 3.1 and earlier; described in [1])

0 is to use an optimized integration grid (speeds up the calculation)

1 is to use tabulated differential cross sections; it saves the files in the directory diff_CS (see below) and uses more memory, but significantly speeds up the MC calculations.

The default option is 1, if the flag is unused.

Note: the difference between the numerical integration and interpolation may be on the order of a few percent, see Appendix with some examples.

- **print_CDF** – prints out the cdf-file with the CDF coefficients (especially useful for printing out automatically fitted single-pole CDF).
- **print_optical** – prints out the optical CDF reconstructed from the Ritchie-Howie fitted loss function.
- **units unit** – index for units used in gnuplot plots. The parameter *unit* sets the units in energy density in the plots to be:

0 sets the units to [eV/Å³] (old units used in TREKIS before update 22.05.2024)

1 sets the units to [eV/atom] (new, default)

Note: it only changes the units in the plot (gnuplot script), not in the output files.

- **verbose** – makes TREKIS-3 printout additional information on the screen, along the execution of the program (for example, timestamps after execution of various subroutines and during MC), which is useful for testing, code developing and debugging.
- **very_verbose** – prints out time of each collision of each particle (which is *a lot* of output, useful only for debugging, but not any productive calculations).
- **gnuplot ext** – flag to choose in what format to create plots with gnuplot (requires gnuplot[‡] to be installed in your system; most clusters have it). The second argument *ext* specifies the extension of the plots to be created. Currently, only the following extensions are supported: jpeg; png, gif, eps, pdf. If you do not want to create plots (e.g., if you don't have gnuplot installed), set *ext=no*.
E.g., gnuplot png - will create plots with extension '.png'.

gnuplot no – will switch off plotting option, no gnuplot will be called.

Note #1: the default option is 'gnuplot jpeg', which will be used if this optional argument is not present in the input file or if an unsupported *ext* is specified.

Note #2: TREKIS-3 will create a set of shell scripts (cmd in Windows, sh in Linux) calling gnuplot to plot some values such as mean free paths, ion stopping power and range, radial distributions of

[‡] <http://www.gnuplot.info/>

various quantities calculated (many, but not all – core-shell holes distributions will not be plotted but may be created by analogy from the shell-scripts plotting valence-band distributions).

- **info** – prints out additional basic information about the code, including citations.

2. INPUT_CDF

Directory **INPUT_CDF** contains files with material parameters. If the user did not specify the name of the CDF-file in the input file (optional flag ‘CDF’), the default name is used **[material].cdf**. Files with extension cdf (they are just text files) contain the description of material parameters and complex dielectric function (CDF), see an example in Figure 2. The default file names **[material]** in this directory must exactly coincide with the name given above in the file **Input_Parameters.txt**.

To create a new material, a cdf-file must be set according to the format described below. All lines must be exactly in the described format and order. For materials, whose CDF parameters in the Ritchie-Howie approximations are not present in TREKIS-3, two options could be used: create a new CDF parameterization, following the algorithm described, e.g., in Ref. [12]; or use single-pole approximation to automatically construct an approximate CDF (see details in [9]).

To use the single-pole approximation with automatic construction of the CDF parameters, the CDF file does not need to contain any information on the CDF (those lines may be omitted), but it needs to have the material bandgap (for valence band CDF), column #4 in the line #5 in the example below.

Additionally, one may use optional markers to set the CDF for only the valence band (and/or phonons), whereas all the core shells will use single-pole CDF.

```
Lithium fluoride
2          ! how many elements are in this molecule of solid
9          ! atomic number, contribution of 1st element into compound
3          ! atomic number, contribution of 2d element into compound
2.635 4200.0 7.3 ! density [g/cm^3], speed of sound [m/s], fermi energy [eV]
2          ! number of shells of the first element: F
1          1      689.1e0 2      3.6e0 ! number of CDF functions, shell-designator, ionization potential, number of electrons, Auger-time
400      350      450      ! E0, A, Gamma coefficients
7          63      14.6e0 8      1.0e23 ! number of CDF functions, shell-designator, ionization potential, number of electrons, Auger-time
13.5     4.3      0.45     ! E0, A, Gamma coefficients
15.5     8.5      1.3
18.0     4.9      1.2
24.5     99.0     1.5
26.0     43.0     3.0
29.0     15.0     1.5
37.0     500.0    40.0
1          ! number of shells of the second element: Li
2          1      61.0e0 2      1.0e23 ! number of CDF functions, shell-designator, ionization potential, number of electrons, Auger-time
85.0     120.0    20.0     ! E0, A, Gamma coefficients
85.0     120.0    40.0
2          ! phonon peaks:
0.083    0.0022   0.0035 ! E0, A, Gamma coefficients
0.0635   0.00009  0.01
```

```
1 SiC          ! Material name
2 SiC          ! Chemical formula
3 3.03 9800.0 3.53 2.77 ! density [g/cm^3], speed of sound [m/s], fermi energy [eV], bandgap [eV]
```

```
1 Gold
2 Au
3 19.32 2030.0 5.53 0.0d0 ! density [g/cm^3], speed of sound [m/s], Fermi energy [eV]
4
5 VALENCE
6 9 63 0.0d0 11 1.0d23 ! number of CDF functions, shell-designator, ionization potential, number of electrons, Auger-time
7 1.0 5.0e-4 0.9
8 1.9 -0.14 2.5
9 2.0 -0.042 0.9
10 2.8 0.23 0.8
11 9.0 10.0 9.0
12 16.2 62.0 9.0
13 25.7 55.0 5.0
14 35.0 380.0 14
15 46.0 360.0 15.0 ! E0, A, Gamma coefficients
```

Figure 2. An example of cdf files: (Top panel) LiF.cdf featuring a full description of CDF coefficients; (Middle panel) SiC.cdf showing a file for usage of the single-pole approximation; and (Bottom panel) Au_test2.cdf showing how only valence-band CDF is defined, while for the core shells the single-pole CDF is to be used.

The cdf format is defined as follows:

Line 1: Material name. May be arbitrary name within 100 symbols length.

Line 2: Line specifying the atomic composition of the target materials. It may be specified in two ways:

Option #1: Chemical formula of the target (character). Each element must start with a capital letter (that's how the code identifies the element names), and after it, the relative number of this element may be specified, e.g. SiO₂ (for quartz), Al₂O₃ (for aluminum oxide).

Option #2: Number of atoms in a molecule of the target (integer). E.g. For LiF, it must be 2; for MgFeSiO₄ it is 4, etc. This number defines the number of following lines, since each atom type must be set in a separate line.

Line 3 (if option #2 is used in line 2, number of elements, example in the top of Figure 2): Sets two numbers:

- 1) Atomic number of the first kind of atoms of the compound according to the Periodic Table (integer)
- 2) Stoichiometric contribution of this element into the molecule of the target (real)

Note #1: elements must be sorted according to their atomic number in descending order. E.g., for LiF, the first line must define F, the second line Li, not the other way around.

E.g., for LiF, the first line must be:

9 1 – element 9, one per molecule.

For Si₃N₄, the line looks like:

14 3 - First element is 14 (Si), 3 of them per molecule (Si₃N₄)

Note #2: skip this line if chemical formula was specified in line 2 (example in the bottom of Figure 2).

Line 4 (if option #2 is used in line 2, number of elements): In case of a compound, the line type is identical to the previous one, setting the parameters of the second element. In LiF, it is

3 1 - Element 3 (Li), contribution 1

In case of Si₃N₄, that is

7 4 - element 7 (N), contribution 4 in a molecule (Si₃N₄).

Note: skip this line if chemical formula was specified in line 2.

Line 5 (or another number, depending on the number of elements in the compound): Three (or four) numbers:

- 1) Material density in [g/cm³]
- 2) Speed of sound [m/s]
- 3) Fermi energy in [eV]
- 4) Bandgap [eV]

Note: the bandgap value only needs to be provided here if the single-pole approximation for the CDF is used (lines #6 and further are absent in this file). Otherwise, the bandgap specified in the description of the valence band is used, see below, line #9.

Note #2: Fermi level is used only in metals (materials with zero bandgap specified); for nonmetals, any value may be given (but some real number should be specified).

Line 6: Number of atomic shells in the first element of the compound

Note #1: valence band is set only among the shells of the first element of the target; for all other elements, valent energy levels, contributing into the valence band, must be omitted to exclude double counting.

For example, in LiF, this line contains 2, corresponding to two shells: K-shell and the valence band.

Note #2: all shells must be sorted according to ionization potential in descending order.

Note #3: if these lines (#6 and further) are absent in the file, TREKIS-3 will use a single-pole approximation to automatically construct the CDF [9].

In this case, it sets the parameters empirically as follows.

For the valence/conduction band:

$$E_0 = \hbar\omega_{pl} \quad (\omega_{pl} \text{ is the plasmon frequency for valence band})$$

$$\gamma = E_0$$

For the core shells:

$$E_0 = I_p + 10 \text{ eV} \quad (I_p \text{ is the ionization potential for core shell})$$

$$\gamma = E_0$$

For phonons (elastic scattering):

$$E_0 = 2 * E_{ein} \quad (E_{ein} \text{ is the Einstein energy of the materials})$$

$$\gamma = E_0/2$$

And the A coefficient of the CDF is defined by the normalization according to the k-sum rule [9].

Line 7: This line sets the parameters of the first atomic shell (in case of LiF, K-shell of F):

It contains 5 numbers (see Figure 2):

1 1 689.1e0 2 3.6e0

1) Number of oscillator functions in the CDF for this shell (integer)

2) Shell designator, according to the EPDL convention[§] (see example in Table below, Figure 3).

Note: for the valence band, the number 63 must be used; this addition to the EPDL convention allows TREKIS to identify the valence band.

Note 2: setting here a negative number will force TREKIS to use atomic BEB cross-sections [13], instead of CDF, for electrons and holes (whereas for SHI only CDF model is available).

3) Ionization potential of the shell in [eV] (real).

Note: setting here a negative number, a default value from EADL database will be used. To find a correct value in the EPICS database, it requires a correct setting of the shell designator.

4) Number of electrons in this atomic shell (per atom) or in the valence band (per molecule) (real).

Note: setting here a negative number, a default value from EADL database will be used. It cannot be used for the valence band, since it is an atomic database.

5) Auger decay time of this shell in [fs] (real). If decay is not possible, an “infinite” number must be set here (e.g. 1e20).

Note: setting here a negative number, a default value from EADL database will be used. It cannot be used for the valence band, since it is an atomic database.

Line 8: This and following lines (number of lines is set by the first number in the previous line) sets the CDF oscillators coefficients according to the Ritchie-Howie formalism [7,14]: E0, A, Gamma (real)

For example, for K-shell of F, we set 1 CDF oscillator, so there will be only one line with the CDF coefficients.

Line 9: The same 5 numbers setting the next atomic shell parameters (see Figure 2):

7 63 14.6e0 8 1.0e23

In this case, the valence band (note designator 63) has 7 CDF oscillators coefficients. This indicates that the next 7 lines define the CDF coefficients of the valence band in LiF, in the same order as described below: E0, A, Gamma.

[§] <http://www-nds.iaea.org/epdl97/>

Line 17: Sets the number of shells for the next element in the compound. In case of LiF, it sets Li parameters, which has only 1 shell: K-shell; valence band was already set when the first element (F) was defined.

Line 18: Sets 5 numbers, defining the shell parameters in the same way as line 9 above, for K-shell of Li:

2 1 61.0e0 2 1.0e23

Lines 19-20: Sets the CDF coefficients for this shell: E0, A, Gamma.

After defining all elements in the compound, it is optional to define phonon-CDF for elastic scattering, following the Kuhr-Fitting method (analogous to Ritchie-Howie formalism for inelastic scattering) [15].

Line №21: Number of CDF-oscillators for phonon peaks (in case of LiF, it is 2).

Lines №22-23: CDF coefficients E0, A, Gamma for phonon peaks.

Note: if phonon CDF is absent, the elastic scattering cross sections for electron and valence hole scattering, TREKIS will use Rutherford cross section with Molier screening parameter [16], or the single-pole automatically fitted CDF, if set in the input file.

Optional markers:

‘VALENCE’ flags that after that line, the valence/conduction-band CDF will be defined, in the same format as the CDF for the shells described above (from Line 7 on). This optional marker can only be used *instead* of the CDF for each shell – meaning, the single-pole approximation will be used for all the core shells, and only the valence band CDF will be read from this file.

‘PHONON’ flags to set phonon CDF, in the same format as above (from Line 21 on). This optional marker can only be used instead of the CDF for each shell (can be used together with ‘valence’).

Table VI. Atomic Subshell Designators

Designator	Subshell	Designator	Subshell	Designator	Subshell
1.	K (1s1/2)	21.	N4 (4d3/2)	41.	Pl (6s1/2)
2.	L (2)	22.	N5 (4d5/2)	42.	P23 (6p)
3.	L1 (2s1/2)	23.	N67 (4f)	43.	P2 (6p1/2)
4.	L23 (2p)	24.	N6 (4f5/2)	44.	P3 (6p3/2)
5.	L2 (2p1/2)	25.	N7 (4f7/2)	45.	P45 (6d)
6.	L3 (2p3/2)	26.	O (5)	46.	P4 (6d3/2)
7.	M (3)	27.	O1 (5s1/2)	47.	P5 (6d5/2)
8.	M1 (3s1/2)	28.	O23 (5p)	48.	P67 (6f)
9.	M23 (3p)	29.	O2 (5p1/2)	49.	P6 (6f5/2)
10.	M2 (3p1/2)	30.	O3 (5p3/2)	50.	P7 (6f7/2)
11.	M3 (3p3/2)	31.	O45 (5d)	51.	P89 (6g)
12.	M45 (3d)	32.	O4 (5d3/2)	52.	P8 (6g7/2)
13.	M4 (3d3/2)	33.	O5 (5d5/2)	53.	P9 (6g9/2)
14.	M5 (3d5/2)	34.	O67 (5f)	54.	P1011 (6h)
15.	N (4)	35.	O6 (5f5/2)	55.	P10 (6h9/2)
16.	N1 (4s1/2)	36.	O7 (5f7/2)	56.	P11 (6h11/2)
17.	N23 (4p)	37.	O89 (5g)	57.	Q (7)
18.	N2 (4p1/2)	38.	O8 (5g7/2)	58.	Q1 (7s1/2)
19.	N3 (4p3/2)	39.	O9 (5g9/2)	59.	Q23 (7p)
20.	N45 (4d)	40.	P (6)	60.	Q2 (7p1/2)
				61.	Q3 (7p3/2)

Figure 3 EPDL-format atomic subshell designators.

3. INPUT_DOS

Directory **INPUT_DOS** contains files with the density of states of the valence band of the material. If the filename with DOS was not specified in the input file (optional), than the default name *[material].dos* is used; in this case, the name of the material must coincide with the name provided in the file **INPUT_PARAMETERS.txt**. Files with extension dos are just text files, containing two columns: energy in [eV], and electronic DOS in arbitrary units (DOS is renormalized in TREKIS to match the number of valence electrons).

Note: the energy scale must start from the bottom of the valence band, and end at the top of the valence band, *strictly* in the ascending order: non-monotonously increasing energy grid may produce errors in TREKIS calculations.

Note #2: in the materials with gaps inside of the valence band, the gap must be separated by exact zeros at the top and bottom of subbands (see example in file LiF.dos).

4. INPUT_EADL

Directory **INPUT_EADL** contains three files with the databases of atomic parameters:

INPUT_atomic_data.dat – Parameters of the Periodic Table

EPDL2023.ALL – EPDL parameters from EPIS2023 database, containing photon parameters for scattering on atoms

EADL2023.ALL – EADL parameters from EPIS2023 database, containing atomic parameters

Atomic_form_factors.dat – database of atomic form factors from [17], used for dynamical-screening elastic cross section calculations.

5. Others

!READ_ME_TREKIS_3.doc or **!READ_ME_TREKIS_3.pdf** – this very manual.

3. Compiling and executing

Simple calculations may be done without any parallelization; for productive runs, it is better to use parallelized version. For small number of processors, typically OpenMP version is slightly faster than MPI-version. OpenMP uses shared-memory parallelization, meaning it is only suitable to be run on a single machine – all the processors must be physically within the same computer and share access to the same memory. To use parallelization between a few nodes, MPI version must be used, which does not require the same memory but instead relies on the message-passing interface (MPI).

All paths to compilers and libraries (OpenMP, MPI) must be written in the system variables to be found from your command line. In Windows, the easiest way to use it is to install Microsoft Visual Studio Community^{**}, and oneAPI HPC Toolkit^{††} containing freely distributed Intel Fortran compiler and OpenMP and MPI (and other) libraries. The provided with it “Intel oneAPI command prompt for Intel 64 for Visual Studio 2022” (of similar command line program) has all the paths to intel-compilers and libraries written in.

A. Compiling without parallelization

^{**} <https://visualstudio.microsoft.com/vs/community/>

^{††} <https://www.intel.com/content/www/us/en/developer/tools/oneapi/hpc-toolkit-download.html>

Compilation:

- **Makefile** – make-file for compilation of the TREKIS-3 code in Unix system with help of make^{††} program. Requires ifort2011 or a later version (change the compiler name in the file accordingly) To compile the code without parallelization, use **make non=y** which creates the executable **TREKIS.x**.
- **Make.bat** – cmd-file for compilation for TREKIS-3 code in Windows system, where direct calls of ifx.exe (former ifort.exe). Compilation with option “**db no**” includes debug options and switches OpenMP parallelization off: use **make db no** which creates the executable **TREKIS_DEBUG.exe**.
 - Compiling with option “**fast no**” makes fast compilation and switches OpenMP parallelization off: **make fast no**. It creates executable **TREKIS_no_OMP.exe** without debug option, no parallelization, and without optimizations.

Execution:

To run the code, call **TREKIS_[option].exe** (or **./TREKIS.x** in Linux).

The code also accepts additional options in the command line: ‘verbose’, ‘very-verbose’, and ‘info’ described above (on page 6) – for that, call the code, e.g., as **TREKIS.exe verbose info**.

Note #1: It may also be useful to redirect the program’s printout from the screen into a file, e.g. as **TREKIS.exe > OUTPUT_screen.txt**

B. Compiling with OpenMP

Compilation:

- **Makefile** – make-file for compilation of the TREKIS-3 code in Unix system. Requires ifort2011 or a later version (change the compiler name in the file accordingly) and OpenMP installed or modify to match an existing Fortran compiler and OpenMP settings on your computer. Calling **make** creates the executable **TREKIS.x**.
- **Make.bat** – cmd-file for compilation for TREKIS-3 code in Windows system, where direct calls of ifx.exe (former ifort.exe) and /Qopenmp are made.
 - Calling **make** (or **Make.bat**) in the Intel oneAPI command prompt creates the executable **TREKIS.exe** that can be run in the same command line. This is the default compilation option. For developing and debugging, the following options may also be used:
 - Compilation with option “**db**” includes debug options (currently only available for Windows) is **make db** or **make DEBUGOMP**. Compiling with this option creates the executable **TREKIS_DEBUG_OMP.exe**.
 - Compiling with option “**fast**” makes fast compilation (convenient for debugging): **make fast**. It creates executable **TREKIS_OMP.exe** without debug option and without optimizations.

Execution:

To run the code, call **TREKIS.exe** (or **./TREKIS.x** in Linux), or corresponding **TREKIS_[option].exe** if compiled with an additional option.

The code also accepts additional options in the command line: ‘verbose’, ‘very-verbose’, and ‘info’ described above (on page 6) – for that, call the code, e.g., as **TREKIS.exe verbose info**.

Note #1: It may also be useful to redirect the program’s printout from the screen into a file, e.g. as **TREKIS.exe > OUTPUT_screen.txt**

^{††} [https://en.wikipedia.org/wiki/Make_\(software\)](https://en.wikipedia.org/wiki/Make_(software))

C. Compiling and executing with MPI

Compilation:

To compile TREKIS-3 with MPI-parallelization, use the same make-files with the additional option:

- **Makefile** – makefile for compiling in Linux/Unix-based system; requires OpenMPI and mpif90 compiler installed (or modify to match an existing Fortran compiler and MPI library on your computer; e.g. Intel OneAPI uses mpiifort compiler).

Calling **make mpi=y** creates the executable **TREKIS_MPI.x**.

- **Make.bat** – cmd-file for compilation of TREKIS-3 code in Windows system, where direct calls of mpiifx (former mpiifort) is made.

Calling the script as **make mpi** creates the executable **TREKIS_MPI.exe**.

Execution:

To run the code, call the program via MPI library of your choice (consult the rules on your cluster), e.g.:

mpiexec -np [n] ./TREKIS_MPI.exe (if Intel OneApi is used; the syntax of the call is the same in both, Windows and Linux)

or

mpirun -np [n] ./TREKIS_MPI.exe (if OpenMPI is used)

where [n] specifies the number of processes to run it on. For example, to execute the code on 64 processes, call

mpiexec -np 64 ./TREKIS_MPI.exe

Note #1: It may also be useful to redirect the program's printout from the screen into a file, e.g. as **mpiexec -n 32 ./TREKIS_MPI.exe > OUTPUT_screen.txt**

Note #2: it usually does not make sense to use the number of processes larger than the number of threads/CPU's in the machine.

IV. OUTPUT DATA

TREKIS creates a directory with output files named `OUTPUT_[material]`, containing a number of output files, produced in simulation runs using this material target.

1) Error log file

File with errors that TREKIS can identify will be named `[SHI]_in_[material]_error_log.txt`. If there were any errors that TREKIS identifies, it will save their description in the file, and save the file after the simulation. Otherwise, an empty file will be deleted at the end of simulation. So, if the file is present after the simulation ends, it makes sense to check what's in the file.

Note #1: in MPI version of the code, not all the errors are printed in the file, some of them are only shown on screen. So, use a dump file to save the information from the screen, if needed. I.e., run the program redirecting its printout into a file, e.g.

```
mpirun -n 32 ./TREKIS_MPI.exe > OUTPUT_screen.txt
```

2) Directory with differential cross sections

Note: A working directory, can be skipped, no meaningful output inside.

Unless specified in the input file not to save tabulated differential cross sections, all the files with integrated differential cross sections that TREKIS-3 requires will be stored in the directory `diff_CS`. This directory and the files are created at the first run of the program for the given material and reused in all subsequent runs to significantly speed up the calculations. Files inside are named as follows:

`[particle]_[type]_[CS]_[E].dat`

Where `[particle]` is either Electron or Hole

`[type]` is either IMFP (inelastic mean free path) or EMFP (elastic)

`[CS]` is the model of the cross section, also specifying at what temperature of the target, etc.

`[E]` is the energy of particle, for which the integrated differential cross section is stored.

The files contain 2 columns: 1) transferred energy [eV]; 2) integral of the differential mean free path [$\text{\AA}/\text{eV}$].

No plots are created there, as the files are mainly needed for further calculations, not data analysis.

3) File with CDF coefficients

File in the cdf-format (see above) is printed named: `[material]_[CDF].cdf`. Where

`[CDF]` marks the model used for the loss function (and CDF):

‘CDF’ if the coefficients used were taken from a cdf file,

‘spCDF’ if (if single-pole approximation is used).

Note: This file may be used as a starting point for further refinements of the CDF coefficients, and then placed in the input directory `INPUT_CDF`.

4) File with reconstructed optical CDF

File with the optical coefficients of the complex dielectric function, restored from the fitted Ritchie-Howie loss function, is printed (if required) in the file named: `OUTPUT_Optical_CDF_from_[CDF].cdf`

Where [CDF] marks the used CDF: Ritchie_CDF (if coefficients of the loss function were taken from a CDF file) or single_pole_CDF (if single-pole approximation is used to automatically construct the coefficients).

In this file, the following columns are printed (first line comments the values in the columns):

Column 1 is energy of photon [eV]

Column 2 is real part of the CDF

Column 3 is imaginary part of the CDF

Column 4 is real part of the complex refraction coefficient n (defined as $n_c=n+i*k$)

Column 5 is imaginary part of the complex refraction coefficient k

5) Files with mean free paths

If files with calculated mean free paths (of electrons, valence holes, and photons) do not exist (e.g., it is the first run of the code with the given SHI), or if the file with the material parameters (cdf-file described above) was modified later than the MFP-files, TREKIS will create (overwrite) files with the mean free paths in this material, to reuse them in all future runs without recalculating them. Electrons and valence holes elastic and inelastic mean free paths are saved in the following files: `OUTPUT_[particle]_[type]_[CS].dat` where:

[particle] is either Electron or Hole

[type] is either IMFP (inelastic mean free path) or EMFP (elastic)

[CS] is the model of the cross section, also specifying at what temperature of the target, etc.

The marker CS='CDF' corresponds to Ritchie-Howie CDF provided by the user in the file (and 'spCDF' for single-pole automatic fitting of the loss function), which is augmented with the name of the charge model used in elastic cross section calculation:

- 1) 'CDF_Z=1' is for the fixed charge $Z=1$;
- 2) 'CDF_Zeff' for Barkas-like effective charge;
- 3) 'CDF_FF' for dynamical screening using form-factors for screening by core-shells;

'BEB' is the BEB cross section for electron scattering;

'EADL' is for the photon absorption cross section taken from the EPICS database.

Those files contain a few columns: particle energy in [eV], mean free paths in [\AA]. Number of columns with mean free paths depends on the particular material: for each shell of each element, as set in the cdf-file (described in the input files), and the last column is the total mean free path according to the Matthiessen rule.

6) Files with ranges

Electron and valence holes ranges, calculated within the continuous slowing down approximation, are saved in files `OUTPUT_[particle]_Range_[CS].dat` where

[particle] is either Electron or Hole

[CS] is the model of the cross section, such as CDF or BEB, at what temperature of the target, etc.

Those files contain 3 columns:

Column 1 is energy of electron or hole in [eV]

Column 2 is energy loss in [eV/\AA]

Column 3 is the residual range of the particle in [\AA]

7) File with density of states

File `OUTPUT_[material]_DOS_analysis.dat` contains the data with the DOS, as used in the TREKIS (after renormalization and extracted parameters required in the code). It contains 5 columns:

Column 1 is energy of the valence band counted down from the Fermi energy (top of the valence band) in [eV]

Column 2 is the corresponding wave-vector within the effective one-band approximation [2] in [1/m]

Column 3 is the valence DOS, interpolated on the grid used by TREKIS

Column 4 is the integral of the DOS

Column 5 is the effective valence hole mass, according to the effective one-band approximation in [free electron mass units]

8) Valence holes self-diffusion coefficient

In case if the valence holes transport is allowed, file `Hole_selfdiffusion_coeff_in_[material].txt` will contain a self-diffusion coefficients of valence holes, calculated via its mean free path and velocity.

9) SHI mean free paths, energy losses, and ranges

If the file with SHI mean free path does not exist, or the cdf-file with material parameters was modified later than the MFP-file, TREKIS will calculate the inelastic MFP of the given SHI and save it in the file. SHI mean free paths, energy losses, and ranges, are saved in the files analogous to those for electrons and holes (see above). They will be saved in an ion-specific directory.

Files `OUTPUT_[SHI]_[charge]_[charge_model]_[output].dat` where `[SHI]` is the SHI elements name

`[charge]` is the marker of the model of the effective ion charge used:

- 1) Barkas [18]
- 2) Bohr [19]
- 3) ND stands for Nikolaev-Dmitriev approximation [20]
- 4) SG stands for Schiwietz-Grande approximation [21]
- 5) Fixed marks the user-defined constant charge

`[charge_model]` identifies the model of the shape of the SHI:

- 1) P stands for point-like charge
- 2) BK stands for the Brandt-Kitagawa approximation [22]

`[output]` marks what exactly is in the file:

- 1) IMFP is for the mean free path
- 2) dEdx is for the energy loss of SHI
- 3) Range is for the SHI residual range
- 4) effective_charges stands for the effective charge

In those files, the first column is always the ion energy in [MeV], the second one is Mean free path in [\AA]; or energy loss in [$\text{eV}/\text{\AA}$]; or residual range in [\AA]; or calculated effective charge in [units of electron charge].

9) Directory with results of MC simulations

A directory with all results of the Monte Carlo simulations is created with the name

`[SHI]_E_[Energy]_MeV_[simulation_time]_fs`

For example, for a gold ion with energy of 2187 MeV and total simulation time of 100 fs, the directory will be called `Au_E_2187.00_MeV_10.0_fs`.

In case identical parameters were used in previous simulations, the new directory will have a number at the end, e.g., `Au_E_2187.00_MeV_10.0_fs_1`, and then `Au_E_2187.00_MeV_10.0_fs_2` etc. TREKIS will not overwrite existing results.

The following files are created in this directory:

10) File !Parameters.txt

This file contains the information about the used input parameters, and parameters extracted from the EPICS database, such as atomic energy levels, decay times etc.

11) Total numbers and energy

The total numbers of excited electrons and their energy are saved in the file

Total_numbers.txt where

Column 1 is time in [fs],

Column 2 is the total number of electrons

Column 3 is the total energy in [eV]

The total energy should conserve in simulations without electron emission after the SHI passage.

The file Total_energies.txt contains the total energies of the electrons, holes, and atoms.

12) Files with radial distribution of electrons and their energy

Radial distributions of electron density and energy density are printed in the files:

Radial_electron_density[1_cm^-3].txt and

Radial_electron_energy[eV_A^-3].txt, respectively.

The units are shown in the file names: electron density is in [1/cm³], and energy density is in [eV/Å³].

In each file, the first line is a comment declaring the columns. The first column is always radius in [Å], counted from the SHI trajectory. Next columns are the electron density or energy density at different time instants defined by the user.

Energy is also printed out in terms of average kinetic temperature in the classical limit ($T=2/3E$) in the file:

Radial_electron_temperature[K].txt

13) Files with radial distribution of photon and their energy

Analogously to the electronic radial distributions, photons distributions are printed out in the files

Radial_photon_energy[eV_A^-3].txt and

Radial_photon_density[1_cm^-3].txt,

only if the radiative decays, producing photons, are included in the code.

14) Files with holes radial distributions and their energy

Analogously to electrons radial distributions, distributions of holes densities and energy densities are saved in a number of files

Radial_Valence_holes_density[1_A^-3].txt,

Radial_Valence_holes_pot_energy[eV_A^-3].txt and

Radial_Valence_holes_kin_energy[eV_A^-3].txt

For the number of valence holes, and their kinetic and potential energies. The valence energy is counted from the top of the valence band. The potential energy is the band gap of the material, counted for each valence hole.

For atomic deep shells (core shells), analogous files are created, but without the kinetic energy, with their potential energy only. Files are marked by the element and shell names, e.g., Radial_Li_K-shell_holes_density[1_cm^-3].txt stands for the K-shell of Li atoms in our example of LiF.

Energy is also printed out in terms of average kinetic temperature in the classical limit ($T=2/3E$) in the file: [Radial_Valence_holes_temperature\[K\].txt](#)

15) File with radial distribution of the energy transferred to lattice

Energy transferred to the lattice in elastic scattering of electrons and valence holes is printed in the file [Radial_Lattice_energy\[eV_Å⁻³\].txt](#). Energy is also printed out in terms of average kinetic temperature in the classical limit ($T=2/3E$) in the file: [Radial_Lattice_temperature\[K\].txt](#)

16) File with radial distribution of the total energy transferred to lattice

Total energy transferred to the lattice (via elastic scattering of electrons and valence holes summed with potential and kinetic energy of valence holes) is printed in the file [Radial_Track_energy\[eV_Å⁻³\].txt](#). This file may be used for setting initial conditions for MD simulations, as discussed below in Section V.

17) Files with electron and VB holes spectra

File [Electron_distribution_vs_E\[1_eV\].txt](#) contains electronic distribution in energy space (electronic spectrum). The distribution is normalized to the total number of electrons. Each column is printed for the chosen time instant.

File [VB_holes_distribution_vs_E\[1_eV\].txt](#) is the same but for valence-band electrons.

18) File with electron velocity distribution by azimuth angle (theta)

File [Electron_theta_distribution.txt](#) contains distribution of electron velocities by the azimuth angle theta, allowing to trace how electron transport loses preferential direction. The first column is the angle from 0 to 180 degrees. Others are electron distributions at different time instants.

19) File with valence-band holes velocity distribution by azimuth angle (theta)

File [VB_holes_theta_distribution.txt](#) contains distribution of VB holes velocities by the azimuth angle theta, allowing to trace how holes transport loses preferential direction. The first column is the angle from 0 to 180 degrees. Others are VB holes distributions at different time instants.

20) Gnuplot scripts and plots

TREKIS-3 also creates a set of gnuplot scripts and plots them by calling gnuplot for a number of key-parameters such as radial distributions of densities and energies, spectra, and total numbers and energies.

Note: the units in the plots of energy density (dose) are [eV/atom], which is set by a multiplier in the gnuplot-script file; the units in the files are [eV/Å], which must be taken into account when using the TREKIS output for MD input.

V. Combining TREKIS-3 with MD via output-input

Following the methodology developed in [23], one can model formation of the track after an SHI impact with atomic resolution using classical molecular dynamics (MD) simulations, such as, e.g., LAMMPS [24]. The electronic kinetics, providing energy to the atomic system of the target, may be calculated with TREKIS-3. It is important to take into account that there are three distinct channels of energy transfer from excited electrons to atoms of the target:

- electron elastic scattering,
- valence holes elastic scattering,
- nonthermal modification of the interatomic potential [25].

The first two channels are accounted for in TREKIS-3, and the energy provided to lattice atoms is printed out in the file #15 (`Radial_Lattice_energy[eV_A^-3].txt`) described above. The last one is not directly included, however, may be approximately accounted for *via* the approximation of the instantaneous conversion of the potential energy to the kinetic energy of the atoms [9]. In this case, the potential energy associated with the nonthermal modifications of the interatomic potential is approximated as the band gap energy, assigned to each hole [23]. Thus, the energies of holes from the point #14 above, must be added to the atomic energy.

The sum of all these energies is printed out in the file #16 above. It can be used at the time instant when electronic cascades are finished, can then be used to set the initial conditions for further atomic dynamics. The time instant for that is typically ~ 100 fs [23]. TREKIS-3 output at 100 fs defines the initial atomic energy in cylindrical layers to be used in LAMMPS (or other MD simulations). The kinetic energies of atoms in the corresponding cylindric layers should be set according to the calculated energy transfer in the layer, whereas momentum direction can be sampled randomly.

The justification for it is discussed in Ref. [9]. Validation of the proposed methodology was presented in Ref. [4] and references therein.

VI. Consistency checks and known problems

Prior to productive simulation runs, the following checks must be performed.

- 1) Make sure that the calculated electron, valence hole, and photon mean free paths are correct. They are printed in the output files #2-3. Electron inelastic mean free paths for many materials may be checked against databases such as, e.g., NIST [26], electron ranges may be found, e.g., in X-ray-data booklet [27]. If the agreement is poor, there may be a few reasons for it: model CDF parameters are incorrect and CDF oscillators need an adjustment; other model parameters such as effective masses set in the input file may be incorrect. Elastic scattering cross sections may be checked against NIST database in the high-energy limit (note that typically agreement is not perfect) [28].
- 2) Make sure that the interpolated DOS in the file #4 looks reasonable. If it has several sub-bands in the valence band, the gaps between them must be exactly zero. Atomic energy levels must not overlap with the DOS of the valence band (if they do, you probably have double-counting, and some atomic levels must be excluded and transferred into the valence band).
- 3) Make sure that calculated SHI energy loss function S_e printed in the file #6 is correct. It may be compared to known databases such as SRIM^{§§} [29], IAEA^{***}, or NIST^{†††}. Note, however, that the residual ranges may have a rather large mismatch by the absolute value, but qualitatively the same if shifted to match the Bragg peaks.
- 4) Make sure that the values of the atomic energy levels and decay time printed out in the file #8 make sense. Those provided by the user in cdf-file may contain errors and would need to be adjusted.
- 5) Make sure that the total energy in the MC calculations printed in the files #9 is conserved (if electron emission is not allowed in the simulation). If it is not, it may indicate some bug in the code we missed.
- 6) If the program stops without producing output data, most probably it is due to memory overflow. In such a case, try reducing the number of time grid point in the output – you may try to use temporal log-scale with only a few points to print out. It may also indicate that the thickness of the chosen simulated layer is too large, which results in a too large number of secondary particles created.
- 7) Jitter instead of smooth curves in energy loss or MFPs may indicate insufficient number of grid points in the cross section integration (may be present in materials with very sharp peaks in the loss function); this may be changed in the code directly (module parameters in the module “Cross_sections”), which needs to be recompiled then.

^{§§} www.srim.org

^{***} <https://www.iaea.org/resources/databases/stopping-power>

^{†††} <https://www.nist.gov/pml/stopping-power-range-tables-electrons-protons-and-helium-ions>

VII. Appendix

1. Comparison of automatic single-pole approximation with fitted CDF

Figure 4 shows comparisons of the electronic inelastic mean free paths in a few materials calculated with TREKIS-3 using the fitted CDF (provided with the code) with the automatically fitted single-pole approximation. Wherever available, the results are compared with other calculations and experimental data. It can be seen that the single-pole approximation in most cases produces reasonable approximation at electron energies above some ~ 50 eV, but may overestimate the cross section (underestimate the mean free path) at lower energies.

Stopping power of SHI has similar quality, see Figure 5. A comparison with SRIM [29] and PASS [30] codes shows that the agreement is reasonable, and may be used in practice.

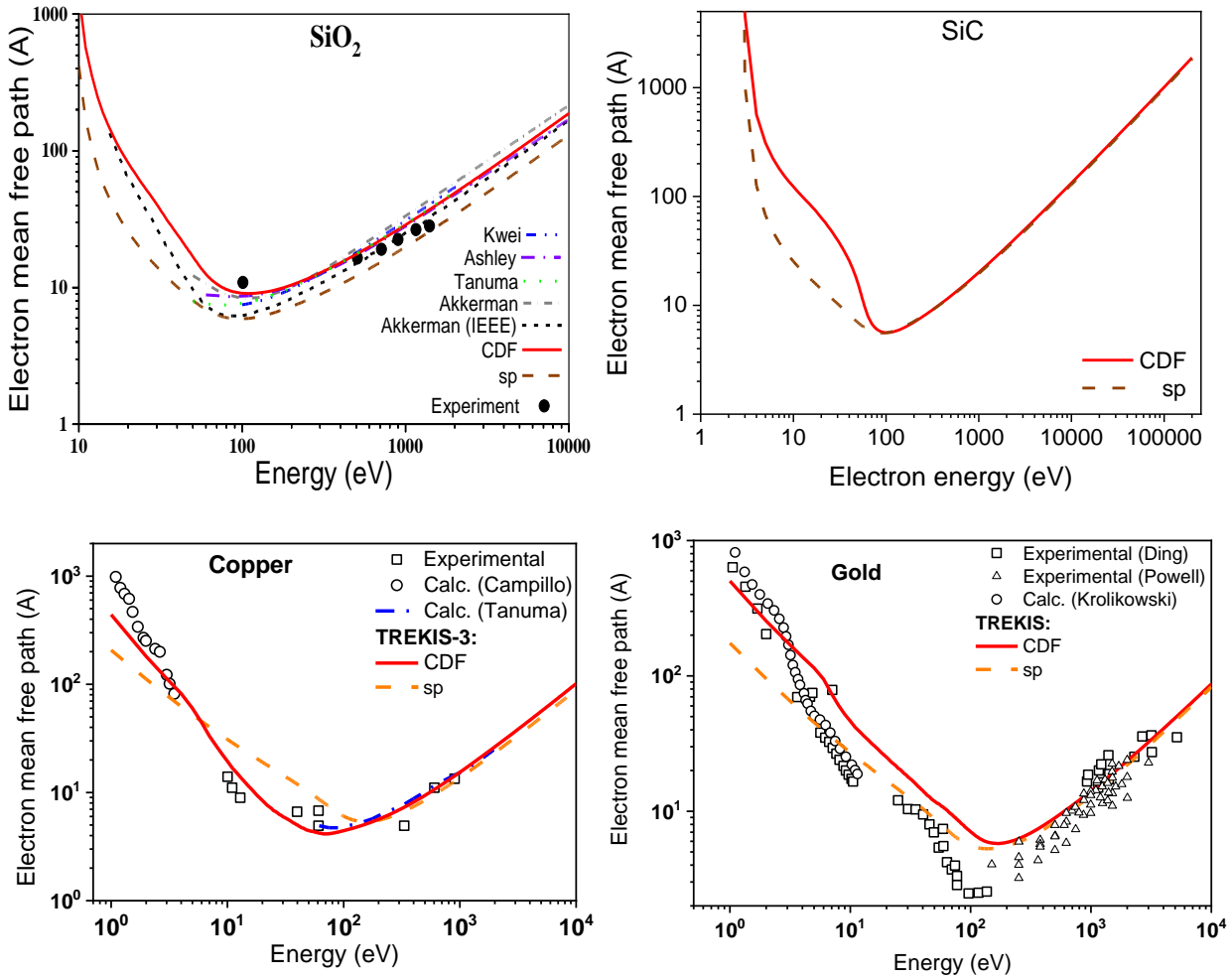


Figure 4. Calculated electronic mean free path with fitted CDF (marked as ‘CDF’, provided with TREKIS-3) and with the automatically created single-pole approximation (marked as ‘sp’ in the plot), wherever available compared with other data.

(Top left panel) in solid SiO₂, the experimental data from Ref. [40] and other theoretical calculations (data from Ref. [40] denoted as “Akkerman (IEEE)”, the other data are from Ref. [41]), figure adapted from [12].

(Top right panel) in SiC.

(Bottom left panel) in copper, the experimental data (open squares) were taken from [31,32], the calculated data from Campillo *et al.* (circles, Ref. [33]), and Tanuma *et al.* (blue dots, Ref. [26]); figure adapted from [2].

(Bottom right panel) in gold, the experimental data were taken from the papers of Ding *et al.* [34] (open squares) and Powell *et al.* [35] (open triangles). The calculated data (circles) are from Krolikowski [36]; figure adapted from [2].

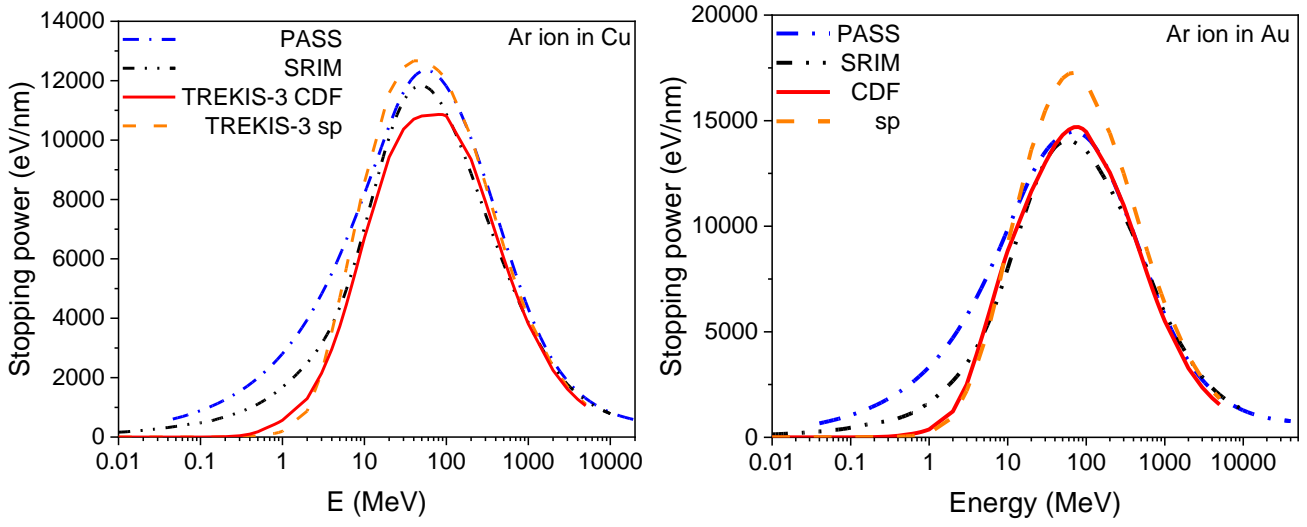


Figure 5. Stopping power of Ar ion in copper (left panel) and in gold (right panel), compared with the two standard codes: SRIM [29] and PASS [30].

2. Comparison of numerical integration vs. interpolation

Since TREKIS-3 version 3.2.0 update 30.05.2024, the numerical integration of CDF-based cross-sections for energy transfer in scattering events is replaced with pre-calculation of the integrated differential cross section, saving the tables, and using interpolation in simulation run. It significantly sped up the simulation, only marginally increasing the used memory and disk-space. The interpolation method produces results very close to the exact integration, typically within a few percent, see example in Figure 6.

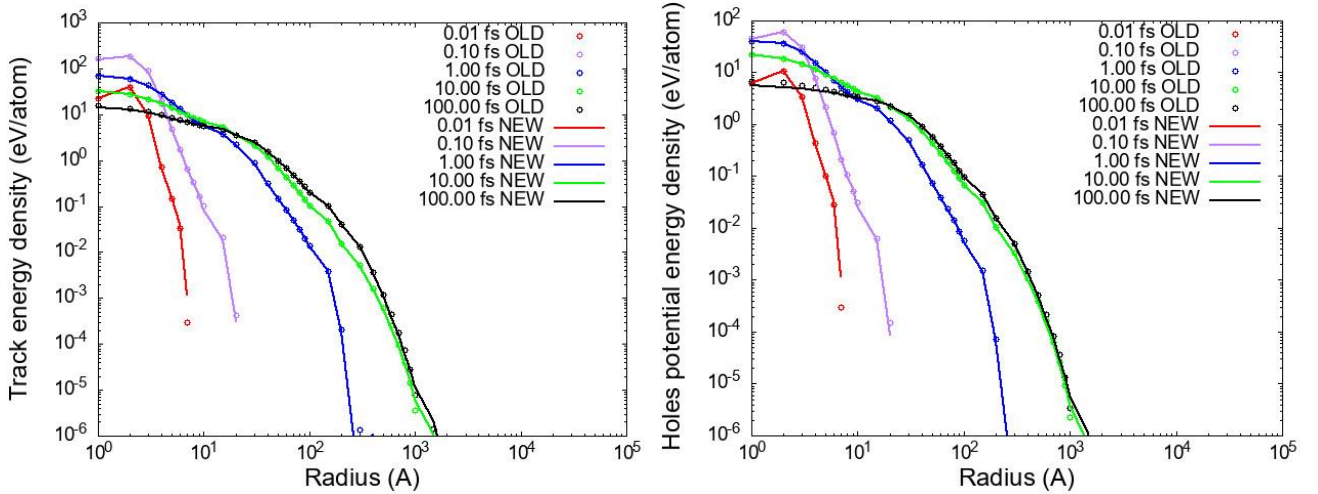


Figure 6. Track energy (left panel) and holes potential energy (right panel) radial distribution in CeO_2 at different times after irradiation with 200-MeV Xe ion, calculated *via* numerical integration (OLD) and interpolation method (NEW).

VIII. References

- [1] N.A. Medvedev, R.A. Rymzhanov, A.E. Volkov, Time-resolved electron kinetics in swift heavy ion irradiated solids, *J. Phys. D. Appl. Phys.* 48 (2015) 355303. <https://doi.org/10.1088/0022-3727/48/35/355303>.
- [2] R.A. Rymzhanov, N.A. Medvedev, A.E. Volkov, Effects of model approximations for electron, hole, and photon transport in swift heavy ion tracks, *Nucl. Instruments Methods Phys. Res. Sect. B Beam Interact. with Mater. Atoms* 388 (2016) 41–52. <https://doi.org/10.1016/j.nimb.2016.11.002>.
- [3] M. Lang, F. Djurabekova, N. Medvedev, M. Toulemonde, C. Trautmann, Fundamental phenomena and applications of swift heavy ion irradiations, in: *Compr. Nucl. Mater.*, Elsevier, 2020: pp. 485–516. <https://doi.org/10.1016/B978-0-12-803581-8.11644-3>.
- [4] R.A. Rymzhanov, N. Medvedev, J.H. O’Connell, V.A. Skuratov, A. Janse van Vuuren, S.A. Gorbunov, A.E. Volkov, Insights into different stages of formation of swift heavy ion tracks, *Nucl. Instruments Methods Phys. Res. Sect. B Beam Interact. with Mater. Atoms* 473 (2020) 27–42. <https://doi.org/10.1016/j.nimb.2020.04.005>.
- [5] L. Van Hove, Correlations in Space and Time and Born Approximation Scattering in Systems of Interacting Particles, *Phys. Rev.* 95 (1954) 249–262. <https://doi.org/10.1103/PhysRev.95.249>.
- [6] U. Fano, Penetration of Protons, Alpha Particles, and Mesons, *Annu. Rev. Nucl. Sci.* 13 (1963) 1–66. <https://doi.org/10.1146/annurev.ns.13.120163.000245>.
- [7] R.H. Ritchie, A. Howie, Electron excitation and the optical potential in electron microscopy, *Philos. Mag.* 36 (1977) 463–481. <https://doi.org/10.1080/14786437708244948>.
- [8] N.A. Medvedev, R.A. Rymzhanov, A.E. Volkov, Complex dielectric function formalism for description of the electron kinetics in swift heavy ion tracks in LiF and Y2O3, *Nucl. Instruments Methods Phys. Res. Sect. B Beam Interact. with Mater. Atoms* 315 (2013) 85–89. <https://doi.org/10.1016/j.nimb.2013.04.090>.
- [9] N. Medvedev, F. Akhmetov, R.A. Rymzhanov, R. Voronkov, A.E. Volkov, Modeling Time-Resolved Kinetics in Solids Induced by Extreme Electronic Excitation, *Adv. Theory Simulations* 5 (2022) 2200091. <https://doi.org/10.1002/adts.202200091>.
- [10] J. Chihara, Interaction of photons with plasmas and liquid metals - photoabsorption and scattering, *J. Phys. Condens. Matter* 12 (2000) 231–247. <https://doi.org/10.1088/0953-8984/12/3/303>.
- [11] R.A. Rymzhanov, N.A. Medvedev, A.E. Volkov, Electron emission from silicon and germanium after swift heavy ion impact, *Phys. Status Solidi B* 252 (2015) 159–164. <https://doi.org/10.1002/pssb.201400130>.
- [12] N. Medvedev, Modeling ultrafast electronic processes in solids excited by femtosecond VUV-XUV laser Pulse, *AIP Conf. Proc.* 582 (2012) 582–592. <https://doi.org/10.1063/1.4739911>.
- [13] Y.-K. Kim, M. Rudd, Binary-encounter-dipole model for electron-impact ionization, *Phys. Rev. A* 50 (1994) 3954–3967. <https://doi.org/10.1103/PhysRevA.50.3954>.
- [14] A. Akkerman, T. Boutboul, A. Breskin, R. Chechik, A. Gibrekhterman, Y. Lifshitz, Inelastic Electron Interactions in the Energy Range 50 eV to 10 keV in Insulators: Alkali Halides and Metal Oxides, *Phys. Status Solidi* 198 (1996) 769–784. <https://doi.org/10.1002/pssb.2221980222>.
- [15] J.-C. Kuhr, H.-J. Fitting, Monte Carlo simulation of electron emission from solids, *J. Electron Spectros. Relat. Phenomena* 105 (1999) 257–273. [https://doi.org/10.1016/S0368-2048\(99\)00082-1](https://doi.org/10.1016/S0368-2048(99)00082-1).
- [16] T.M. Jenkins, W.R. Nelson, A. Rindi, Monte Carlo Transport of Electrons and Photons, Springer US, Boston, MA, 1988. <https://doi.org/10.1007/978-1-4613-1059-4>.
- [17] F. Salvat, M. Fern, PENELOPE-2014 – A code system for Monte Carlo simulation of electron and photon transport, 2015th ed., NUCLEAR ENERGY AGENCY, Organisation for Economic Co-operation and Development, Barcelona, Spain, 2015. https://www.oecd-neo.org/jcms/pl_19590/penelope-2014-a-code-system-for-monte-carlo-simulation-of-electron-and-photon-transport.
- [18] W.H. Barkas, Nuclear research emulsions., New York, 1963. <http://hdl.handle.net/2027/uc1.b3533401> (accessed November 20, 2013).
- [19] H.-D. Betz, Charge States and Charge-Changing Cross Sections of Fast Heavy Ions Penetrating Through Gaseous and Solid Media, *Rev. Mod. Phys.* 44 (1972) 465–539.

<https://doi.org/10.1103/RevModPhys.44.465>.

- [20] V.S. Nikolaev, I.S. Dmitriev, On the equilibrium charge distribution in heavy element ion beams, *Phys. Lett.* 28 (1968) 277–278.
- [21] G. Schiwietz, K. Czerski, M. Roth, F. Staufienbiel, P.L. Grande, Femtosecond dynamics – snapshots of the early ion-track evolution, *Nucl. Instruments Methods Phys. Res. Sect. B Beam Interact. with Mater. Atoms* 226 (2004) 683–704. <https://doi.org/10.1016/j.nimb.2004.05.043>.
- [22] W. Brandt, M. Kitagawa, Effective stopping-power charges of swift ions in condensed matter, *Phys. Rev. B* 25 (1982) 5631–5637. <https://doi.org/10.1103/PhysRevB.25.5631>.
- [23] R. Rymzhanov, N.A. Medvedev, A.E. Volkov, Damage threshold and structure of swift heavy ion tracks in Al₂O₃, *J. Phys. D. Appl. Phys.* 50 (2017) 475301. <https://doi.org/10.1088/1361-6463/aa8ff5>.
- [24] A.P. Thompson, H.M. Aktulga, R. Berger, D.S. Bolintineanu, W.M. Brown, P.S. Crozier, P.J. in 't Veld, A. Kohlmeyer, S.G. Moore, T.D. Nguyen, R. Shan, M.J. Stevens, J. Tranchida, C. Trott, S.J. Plimpton, LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales, *Comput. Phys. Commun.* 271 (2022) 108171. <https://doi.org/10.1016/J.CPC.2021.108171>.
- [25] N. Medvedev, A.E. Volkov, Nonthermal acceleration of atoms as a mechanism of fast lattice heating in ion tracks, *J. Appl. Phys.* 131 (2022) 225903. <https://doi.org/10.1063/5.0095724>.
- [26] C.J. Powell, A. Jablonsky, NIST Electron Inelastic-Mean-Free-Path Database: Version 1.2, (2014) <http://www.nist.gov/srd/nist71.cfm>. <http://www.nist.gov/srd/nist71.cfm>.
- [27] A. Thompson, D. Vaughan, J. Kirz, D. Attwood, E. Gullikson, M. Howells, K.-J. Kim, J. Kortright, I. Lindau, P. Pianetta, A. Robinson, J. Underwood, G. Williams, H. Winick, X-Ray Data Booklet, Center for X-ray Optics and Advanced Light Source, Lawrence Berkeley National Laboratory, Berkeley, CA, USA, 2009. <http://xdb.lbl.gov/> (accessed August 26, 2015).
- [28] A. Jablonski, F. Salvat, C.J. Powell, A. Y. Lee, NIST Electron Elastic-Scattering Cross-Section Database, NIST Electron Elastic-Scattering Cross-Section Database Version 4.0, NIST Stand. Ref. Database Number 64, Natl. Inst. Stand. Technol. Gaithersbg. MD, 20899 (2016). <https://srdata.nist.gov/SRD64/Elastic> (accessed October 19, 2023).
- [29] J.F. Littmark, J.P. Ziegler, U. Biersack, *The Stopping and Range of Ions in Solids*, Pergamon Press, New York, 1985.
- [30] A. Schinner, P. Sigmund, Expanded PASS stopping code, *Nucl. Instruments Methods Phys. Res. Sect. B Beam Interact. with Mater. Atoms* 460 (2019) 19–26. <https://doi.org/10.1016/j.nimb.2018.10.047>.
- [31] B. Ziaja, R.A. London, J. Hajdu, Ionization by impact electrons in solids: Electron mean free path fitted over a wide energy range, *J. Appl. Phys.* 99 (2006) 033514. <https://doi.org/10.1063/1.2161821>.
- [32] D.R. Penn, Electron mean-free-path calculations using a model dielectric function, *Phys. Rev. B* 35 (1987) 482–486. <https://doi.org/10.1103/PhysRevB.35.482>.
- [33] I. Campillo, V.M. Silkin, J.M. Pitarke, E. V Chulkov, A. Rubio, P.M. Echenique, First-principles calculations of hot-electron lifetimes in metals, *Phys. Rev. B* 61 (2000) 13484. <https://doi.org/10.1103/PhysRevB.61.13484>.
- [34] Z.-J. Ding, R. Shimizu, Inelastic collisions of kV electrons in solids, *Surf. Sci.* 222 (1989) 313–331.
- [35] C.J. Powell, A. Jablonski, Evaluation of Calculated and Measured Electron Inelastic Mean Free Paths Near Solid Surfaces, *J. Phys. Chem. Ref. Data* 28 (1999) 19. <https://doi.org/10.1063/1.556035>.
- [36] W. Krolikowski, W. Spicer, Photoemission Studies of the Noble Metals. I. Copper, *Phys. Rev.* 185 (1969) 882–900. <http://link.aps.org/doi/10.1103/PhysRev.185.882>.