Date: 03/07/23
User guide version number: development version
Corresponds with MuSSIC version number: development version

Multiscale Simulation Scattering Intensity Calculator (MuSSIC)

User guide

HB Kolli, T Youngs, G Jiménez-Serratos, and TF Headen

STFC, UK

February 2023

Contents

1. Description of the Code 1
2. Compilation and run 2
3. Input variables 2
4. INPUT files 3
4.1 input.dat for C10TAB system 3
4.2 model.dat for C10TAB in water 6
5. OUTPUT files 8
6. Gnuplot script 9

1. Description of the Code

MuSSIC is a computational tool for calculation of neutron scattering for coarse grain (CG) systems (detailed description is provided in the Documentation). It is based on the method described by Alan Soper in his journal article on coarse-grained Empirical Potential Structure Refinement (Biochimica et Biophysica Acta 1861, 6, 2017, Pages 1652-1660). The code computes the partial pair radial distribution function and the partial pair structure factors for the trajectory provided in xyz file format. The total structure factor is the weighted sum of all partial pair structure factors. The current version of the code is available on Github page (https://github.com/disorderedmaterials/MuSSIC). Any

developments or improvements in the code/method and related examples are updated on gitpage.

2. Compilation and run

MuSSIC is written in FORTRAN and requires gfortran compiler with openmp support for parallel implementation. After downloading the source code, follow the commands on terminal

>make

copy the executable 'scattcg' to the desired location. Execute with

>./scattcg

3. Input variables

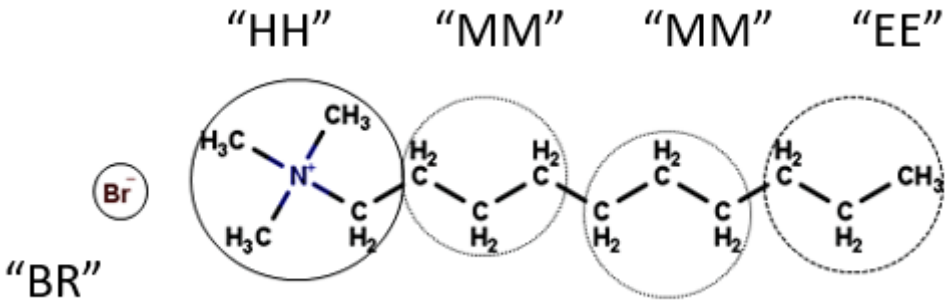
Input variables are separated into four categories. The following table shows the example variables of each category. Section 4 gives a detailed explanation of all variables by showing input files for C10TAB in water system.

Simulation parameters	**model parameters**
Title of the system number of frames box dimensions Ensemble	Number of species number of molecules in each species Number of atoms per molecule number of atom types
Scattering calculation parameters Binning of Q in F(Q) Binning of r in g(r) Binning regime: linear or logarithm Broadening parameters: OmegaDependentGaussian Isotopologue information	**Coarse grain representation** Number of CG bead types Label of bead beads_per_molecule Bead size

4. INPUT files

- input.dat - main input file
- model.dat - model parameters of the system
- trajectory - atomistic trajectory in xyz format
  (required only for the run type 'atomistic or 'compare_CG')
- CG_trajectory - coarse grain trajectory in xyz format
  (required only for the run type 'default')

  The inputs files are explained through an example, a system consisting of C10TAB surfactants in water. The coarse-grained representation is depicted below over the atomistic one.



The labels "HH", "MM", "EE", "BR", and "WW" are the names of the CG beads in the CG_trajectory file.

The aim is to use MuSSIC to obtain the total neutron scattering cross section for two isotopologues, described in the input file as "sample_1" and "sample_2":

- Sample 1: isotopic substitution in C10TAB + H2O
- Sample 2: C10TAB with natural hydrogen + D2O

Both outputs are achieved in a single calculation.

4.1 input.dat for C10TAB system

The simulation parameters and the scattering simulation parameters are given in the input.dat file with the following format (support comments added here):

variable description

```
run:
default          # 'default'  or 'atomisitc' or  'compare_CG'
                 # 'default'  -  computes F_CG(Q) for CG trajectory
                 'atomistic' - compute F(Q) for atomistic
                 trajectory
                 'compare_CG' - computes F(Q) for atomistic
                 trajectory provided and will compute F_CG(Q) for
                 the CG model provided in the model file.


nproc:           # number of the processors for openmp
20
title:           # title of the run (optional)
C10TAB+water
ntypes:          # number of atom types implicit in the system
6
box_length:      # box dimensions
159.6606d0  153.7097d0  155.2160d0
bin_width:       # bin width for rdf calculation
0.03d0
ensemble:        #Current version reads only 'NVT' ensemble
NVT
label_atom:      # labels for the implicit atom types
N
C
HC               # Label for H in the C10TAB surfactant
BR
OW
HW               # Label for H in the water molecule
scattering_lengths:   # scattering lengths of the atom types
9.36d0
6.646d0
-3.739d0
6.795d0
5.803d0
-3.739d0
6.671d0      #(ntypes+1) is the scattering length of deuterium
```

```
form_factor:   # density distribution within the CG bead
Gaussian
exchangeable_atoms:    # number of exchangeable atoms and the
                     label of the exchangeable atom types
1
HW
number_isotopologues: # number of isotopologues to be studied
2
isotopologues:    # ratio of the isotopes (isotope+natural=1.0)
                    in the same sequence of atom types as
                   defined above
sample_1   DH    # label for the isotopologue d-C10TAB + H2O
atom    natural   isotope
N         1.0       0.0
C         1.0       0.0
HC        0.0       1.0
BR        1.0       0.0
OW        1.0       0.0
HW        1.0       0.0
sample_2   HD    # label for the isotopologue C10TAB + D2O
atom    natural   isotope
N          1.0       0.0
C          1.0       0.0
HC         1.0       0.0
BR         1.0       0.0
OW         1.0       0.0
HW         0.0       1.0
Isotopic_substitution:
ratios       # 'manual' or 'ratios'. See notes below
qmin:         # minimum of Q value (usually 2*pi/box_length)
0.01d0
qdelta:       # delta in the Q scale
0.02d0
qmax:         # maximum Q value
5.0d0
binning:   # binning regime (either 'linear' or 'log_linear')
```

```
log_linear
broadening:    # type of broadening used
OmegaDependentGaussian
FWHM:          # full width at half maximum for broadening
0.02d0
windowfn:      # window function used in finding S(Q)
Lorch0
endinput
```

Note 1: ‘manual’ option in the isotopic substitution is being used to generate manually deuterated trajectories. Test case 4 shown in the documentation explains the results with manual deuteration for PA66. While the test case 5 shows the results obtained using ‘ratios’ option for C_{10}TAB in water (refer to verification tests in the documentation)

Note 2: the current version of the code supports only one isotope of the atoms and only one isotopologue can be tested per run if using ‘manual’ substitution

Note3: In addition to the ‘default’ run type, the current version can take two other run types which are ‘atomistic’ and ‘compare_CG’.

default - computes F_CG(Q) for the provided CG trajectory named ‘CG_trajectory’

atomistic - computes F(Q) for the provided atomistic trajectory named ‘trajectory’

compare_CG - computes F(Q) for the provided atomistic trajectory named ‘trajectory’ and performs CG mapping using model file parameters to compute F_CG(Q) for the same atomistic ‘trajectory’

Note4: There is no need to provide ‘nframes’ as input in the current version. The code automatically reads the number of frames from the trajectory

4.2 model.dat for C10TAB in water

The model and CG representation parameters are given in the model.dat file:

variable description

```
num_species  # number of species/molecule types in the system
3
nmol_species    # number of molecules for each molecule type
810             # C10TA molecule
810             # Br- ions
115177          # water molecules
size_per_species     # number of atoms in each molecule
44                   # atoms in C10TA molecule
1                    # number of atoms Br- ion
3                    # number of atoms in water molecule
num_CGbead_types  # number of CG bead types
5
label_bead        # label of CG bead types
1       HH
2       MM
3       EE
4       BR
5       WW
beads_per_molecule    # number of beads in each molecule
4                     # number of beads for C10TA molecule
1                     # number of beads for Br- ion
1                     # number of beads for water molecule
bead_radius       # radii of the CG beads. See notes below
6.091
2.932
3.166
0.707
0.676
num_atom_types_CGbead # Implicit atom contents in the CG beads
3      # number of atom types in CG bead 1, 'HH' (list above)
1     1    # atom type number 1, 'N' (refer to the atom type
           numbers in input.dat) and number of atoms, 1
2     4    # 4 atoms with type number 2, 'C'
3    11    # 11 atoms with type number 3, 'HC'
2           # number of atom types in CG bead 'MM'
2     3    # 3 atoms with type number 2, 'C'
3     6    # 6 atoms with type number 3, 'HC'
2           # number of atom types in CG bead 'EE'
2     3    # 3 atoms with type number 2, 'C'
3     7    # 7 atoms with type number 3, 'HC'
1           # number of atom types in CG bead 'BR'
```

```
4      1     # 1 atom with type number 4, 'BR'
2            # number of atom types in CG bead 'WW'
5      1     # 1 atom with type number 5, 'OW'
6      2     # 2 atoms with type number 6, 'HW'
Endinput
```

Note3: If the radii of beads not available, one can provide bead connectivity matrix as shown below.

```
bead_connectivity
    HH   MM   EE   BR   WW
HH  0    1    1    0    0     # rows and columns  equal to the
MM  1    0    0    0    0     number of bead types. '1' being

EE  1    0    1    1    0     the bead pair connected and '0'

BR  0    0    0    0    0     being not connected

WW  0    0    0    0    0
```

5. OUTPUT files

filename	description
out.log	log file printing the details of the run
CG-unweighted-*-*.gr	partial pair g_{ij}(r) (* - bead types in the pair)
CG-unweighted-*-*.sq	partial pair S_{ij}(Q) (* - bead types in the pair)
CG-unweigthed-total.gr	unweighted total G(r)
CG-unweighted-total.sq	unweighted total S(Q)
%%-weighted-*-*.sq	weighted partial pair S_{ij}(Q) (%% -isotpologue name, * - bead types in the pair)
%%-weighted-*-*.gr	weighted partial pair g_{ij}(r)

(%% -isotpologue name, * - bead types in the pair)

%%-CG-weighted-total.sq	weighted S(Q) (%%-isotopologue name)
%%-weighted-total.gr	weighted G(r) (%%-isotopologue name)

6. Plotting

A gnuplot script is available for the test systems, which can be used to plot the scattering curves. Please change the file names according to the system that is being tested.

Usage:

>gnuplot plot.gnu