Multiscale Simulation Scattering Intensity Calculator (MuSSIC)

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 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 a FORTRAN code and is built using gfortran compiler with openmp for parallel implementation. After downloading the sourcecode, follow the commands on terminal

>cd source_code

>make

copy the executable 'scattcg' to the desired locati>./scattcg

3. Input variables

Input variables are separated into four categories. 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 connectivity in the molecule

3. INPUT files

In the input file 'input.dat', one can choose the run type either 'default' or 'compare_CG'. The default runtype performs the CG scattering calculation on the CG_trajectory which is provided as an input file. The run type 'compare_CG' calculates the scattering using the atomistic trajectory and generates the CG_trajectory using CG model given in model.dat to calculate the CG scattering.

- input.dat - main input file
- model.dat - model parameters of the system
- trajectory - atomistic trajectory in xyz format (file needed in case of 'compare_CG' run type)
- CG_trajectory - coarse grain trajectory in xyz format (file needed in case of 'default' runtype; code generates this file if run type is 'compare_CG')

4.1 input.dat for C10TAB system

variable description

run: # type of the run either 'CG_compare' or 'default'

default

nproc: # number of the processors available for openmp

20

title: # title of the run (optional)

C10TAB+water

ntypes: # number of atomtypes of system

6

nframes: # number of frames in the trajectory

10

box_length: # box dimensions (Incase of NPT ensemble, box dimensions are taken for each frame from trajectory file'

159.6606d0 153.7097d0 155.2160d0

bin_width: # bin width for rdf calculation

0.03d0

ensemble: # type of ensemble either 'NVT' or 'NPT'

NPT

label_atom: # label of the atom types

N
C
HC
BR
OW
HW

scattering_lengths: # scattering lengths of the atom types and additional input

(ntypes+1) is the scattering length of deuterium

9.36d0
6.646d0
-3.739d0
6.795d0
5.803d0
-3.739d0
6.671d0

form_factor: # scattering length density distribution with in the bead

Gaussian

exchangeable_atoms: # number of exchangeable atoms and the label of the exchangeable atom types

0

number_isotopologues: # number of isotopologues to be studied

2

isotopologues: # ratio of the isotopes in each isotopologue (isotope + natural = 1.0) in the same sequence of atom types as defined

sample_1 DH # label of the isotopologue (first letter indicating hydrogen in CTAB molecule and second one indicating hydrogen in water)

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_1 HD

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
qmin: # minimum of Q value (usually 2*pi/box_length)

0.01d0

Isotopic_substitution:

manual # isotopic substitution can be implemented either by choosing

'manual' or 'ratios'

'manual' option is be 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: 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)

qdelta: # delta in the Q scale

0.02d0

qmax: # maximum Q value

5.0d0

binning: # choice of 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: # windown function used in finding S(Q)

Lorch0

endinput

4.2 model.dat for C10TAB in water

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

6

label_bead # label of CG bead types

1 N
2 C3
3 C2
4 CT
5 BR
6 W
beads_per_molecule # number of beads in each molecule
14 # number of beads in C10TA molecule
1 # number of beads in Br- ion
1 # number of beads in water molecule
bead_pairs_molecule # bead connectivity matrix of bead_types x bead_types;
1 being 'connected' and 0 being 'not connected' by a bond
0 1 1 0 0 0 # N C3 C2 CT BR W

```
1   0   0  0  0  0          # C3
1   0   1  1  0  0          # C2
0   0   1  0  0  0          # CT
0   0   0  0  0  0         # BR
0   0   0  0  0  0         # W
bead_radius      #  radii of the CG beads
0.707              (note: if the radii are provided, there is no need to provide
2.343               bond connectivity matrix. Either one of the two is required for
1.199               the radius of the CG bead)
0.707
0.707
0.676
num_atom_types_CGbead    # description of the CG bead (atom types and their
number)

1                    # number of atom types in CG bead  'N'

1    1              # 'N' atom type number (refer to the atom type numbers in
                    input.dat) and number of atoms of 'N'
2                   # number of atom types in CG bead 'C3'
2    1           # 'C' atom type number and number of atoms of 'C'  in CG bead 'C3'
3    3         # 'H' atom type number and number of atoms of 'HC'  in CG bead 'C3'
2                   # number of atom types in CG bead 'C2'
2    1           # 'C' atom type number and number of atoms of 'C'  in CG bead 'C2'
3    2         # 'H' atom type number and number of atoms of 'HC'  in CG bead 'C2'
2                   # number of atom types in CG bead 'CT'
2    1           # 'C' atom type number and number of atoms of 'C'  in CG bead 'CT'
3    3         # 'H' atom type number and number of atoms of 'HC'  in CG bead 'CT'
1                 #  number of atoms in CG bead 'BR'
4    1        # 'BR' atom type number and number of atoms of 'BR' in CG bead 'BR'
2             # number of atom types in CG bead 'W'
5    1     # 'OW' atom type number and number of atoms of 'OW' in CG bead 'W'
6    2      # 'HW' atom type number and numbe rof atoms of 'HW' in CG bead 'W'
endinput
```

5. OUTPUT files

#filename description

fort.7 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)

Output files for atomisitc scattering are the same filenames without 'CG'

File name	description
unweighted-*-*.gr	partial pair g_{ij}(r) (* - atom types in the pair)
unweighted-*-*.sq	partial pair S_{ij}(Q) (* - atom types in the pair)
unweigthed-total.gr	unweighted S(Q)
unweighted-total.sq	unweighted 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)

5. Gnuplot script

A gnu plot script is available for the test systems which can be used plot the scattering curves. One has to change the file names according to the systems that is being tested.