Information about format_history.f90

M. A. Seaton and S. Chiacchiera

August 23, 2017

The module `format_history.f90` is a post-processing utility for DL_MESO_DPD, the Dissipative Particle Dynamics (DPD) code from the DL_MESO package [1, 2]. The module is to be used with DL_MESO in its last released version, version 2.6 (dating November 2015).

The HISTORY files are written by two modules of the DL_MESO_DPD code: `start_module.f90` and `statistics_module.f90`. They contain some basic information about the system and, most importantly, the particle trajectories.

The present module is meant to help the user to interpret and check the HISTORY files. Therefore, for sake of clarity, the notation here is kept as close as possible to that of DL_MESO main code (see Section 4).

1 How to run

The present module is compiled with the available Fortran90 compiler, e.g.:

```
gfortran -o format.exe format_history.f90
```

and the executable must be in the same directory of the HISTORY* files to be analyzed.

2 Output and options

A formatted version of each HISTORY* file is printed into a HISTORY*-F file, optionally including explicative comments. Some checks are printed to the standard output.

Two logical switches are present:

- `lcomm`: allows to add comments (lines of the type " `#...`") in the formatted version of the HISTORY* files.
- `lmcheck`: allows to do some checks on the molecules (number of molecules, beads and bonds per type) and defines the arrays `ltp, ltm, mole, bndtbl` which can be useful for a user-defined analysis of the trajectories. Also, these four arrays are printed to the standard output to allow for further checks.

To choose the preferred option, modify the value of the logical switches inside the module.

3 Test

To test the module, run the simulation with the "toy" input files given in the following. (Note that these files contain commented lines as suggestions for further tests.) For CONTROL:

```
-----------------------------------------
Simple test

volume 3.0  3.0  3.0
temperature 1.0
cutoff 1.0

timestep 0.01
steps 6
equilibration steps 2
traj  2  2  0
stats every 2
stack size 2
print every 2
job time 100.0
close time 10.0

#surface shear y
#surface frozen  x
#surface hard  x

ensemble nvt mdvv

finish
-----------------------------------------
```

For FIELD:

```
-----------------------------------------
Simple test

SPECIES 3
A    1.0  0.0   1   0
B    1.0  0.0   0   0
C    1.0  0.0   0   0

MOLECULES 2
AB
nummols 1
beads 2
A    0.0 0.0 0.0
B    0.1 0.0 0.0
bonds 1
harm 1 2 5.0 0.0
finish
AC
nummols 1
beads 2
A    0.0 0.0 0.0
C    0.1 0.0 0.0
bonds 1
harm 1 2 3.0 0.0
finish

INTERACTIONS 3
A    A    dpd 25.0 1.0 4.5
B    B    dpd 25.0 1.0 4.5
C    C    dpd 25.0 1.0 4.5

#EXTERNAL
#shear  3.0  0.0  0.0

CLOSE
-----------------------------------------
```

Then analyze the produced HISTORY file using `format_history.f90` (i.e., run `format.exe`).

In the case of a serial run (i.e., a single HISTORY file) and for both `lcomm` and `lmcheck` set to `.TRUE.`, the output printed to the screen should coincide with what shown below:

```
-------------------------------------------------
 Number of nodes used in calculations ?
 # Check of beads: i, ltp(i), ltm(i), mole(i)
           1           1           0           0
           2           1           1           1
           3           2           1           1
           4           1           2           2
           5           3           2           2
 # Check of molecules: nammol(i), nbdmol(i), nbomol(i), nmol(i)
 AB                 2           1           1
 AC                 2           1           1
 # Total number of molecules =            2
 # Check of bonds: bndbtl(i,1), bndbtl(i,2)
           2           3
           4           5
-------------------------------------------------
```

and the HISTORY-F file should be

```
-----------------------------------------------------------------------------------------------------
 # nspe, nmoldef, nusyst, nsyst, nbeads, nbonds
           3           2           1           5           5           2
 # dimx, dimy, dimz, volm
     3.000        3.000        3.000       27.000
 # keytrj, srfx, srfy, srfz
           0           0           0           0
 # SPECIES:
 # namspe, amass, rcii, lfrzn
 A               1.000        1.000    0
 B               1.000        1.000    0
 C               1.000        1.000    0
 # MOLECULES:
 # nammol
 AB
 AC
 # Simulation name:
 Simple test
 # BEADS:
 # global, species, molecule, chain
           1           1           0           0
           2           1           1           1
           3           2           1           1
           4           1           2           2
           5           3           2           2
 # BONDS:
 # extremes of the bond
           2           3
           4           5
 # --- TRAJECTORIES --- (key =           0 )
 # mglobal, x, y, z
 # time, mbeads, dimx, dimy, dimz, shrdx, shrdy, shrdz
     0.000        5.000        3.000        3.000        3.000        0.000        0.000        0.000
 # snapshot number:           1
       1.0   1.471873E+00    1.525203E+00    1.507395E+00
       2.0   1.364570E+00    2.228593E+00    2.475293E+00
       3.0   1.300679E+00    2.256132E+00    2.340013E+00
       4.0   2.306000E+00    2.535871E+00    2.718987E-01
       5.0   2.292338E+00    2.576789E+00    2.972781E-01
 # time, mbeads, dimx, dimy, dimz, shrdx, shrdy, shrdz
     0.020        5.000        3.000        3.000        3.000        0.000        0.000        0.000
 # snapshot number:           2
       1.0   1.443747E+00    1.550407E+00    1.514791E+00
       2.0   1.403396E+00    2.234155E+00    2.504685E+00
       3.0   1.294301E+00    2.264003E+00    2.309210E+00
       4.0   2.296760E+00    2.511000E+00    2.881257E-01
       5.0   2.297256E+00    2.563023E+00    2.750665E-01
 # time, mbeads, dimx, dimy, dimz, shrdx, shrdy, shrdz
     0.040        5.000        3.000        3.000        3.000        0.000        0.000        0.000
 # snapshot number:           3
       1.0   1.415620E+00    1.575610E+00    1.522186E+00
       2.0   1.444185E+00    2.239136E+00    2.537687E+00
       3.0   1.285960E+00    2.272457E+00    2.274797E+00
       4.0   2.287877E+00    2.466031E+00    3.080055E-01
       5.0   2.301818E+00    2.569355E+00    2.492021E-01
```

```
# time, mbeads, dimx, dimy, dimz, shrdx, shrdy, shrdz
    0.040        5.000        3.000        3.000        3.000        0.000        0.000        0.000
-----------------------------------------------------------------------------------------------------
```

4 Variables

Here is a list of the variables used, where the notation [1] of the user manual [3] (USRMAN.pdf) is followed. See also Tables 10.1, 10.2, 10.3 and 10.4 of the manual [3], from which most of the information contained in the following table is extracted.

parameter	meaning	data type	notation
`nspe`	number of particle species	integer	≥ 1
`nmoldef`	number of defined molecule types	integer	
`nusyst`	total number of unbonded particles in system	integer	
`nsyst`	total number of particles in system	integer	
`numbond`	total number of bonds in system	integer	
`nbeads`	number of particles in domain cell	integer	
`nbonds`	number of bonds in domain cell	integer	
`dimx`	size of system in x-dimension	real (KIND=dp)	
`dimy`	size of system in y-dimension	real (KIND=dp)	
`dimz`	size of system in z-dimension	real (KIND=dp)	
`volm`	system volume	real (KIND=dp)	
`keytrj`	data key for HISTORY* files	integer	
`srfx`	surface switch for boundary normal to x-axis	integer	
`srfy`	surface switch for boundary normal to y-axis	integer	
`srfz`	surface switch for boundary normal to z-axis	integer	
`srftype`	surface type	integer	
`nammol`	molecule name	character(8)*	A `0:nmoldef`
`nmol`	molecule type population	integer*	A `nmoldef`
`nbdmol`	bead numbers in molecule types	integer	A `nmoldef`
`namspe`	species name	character(8)*	A `nspe`
`text`	name of DL_MESO_DPD calculation	character(80)	
`shrdx`	x component of Lees-Edwards shearing displacement	real (KIND=dp)	
`shrdy`	y component of Lees-Edwards shearing displacement	real (KIND=dp)	
`shrdz`	z component of Lees-Edwards shearing displacement	real (KIND=dp)	
`time`	DPD time starting from equilibration	real (KIND=dp)	
`amass`	species particle mass	real(KIND=dp)*	A `nspe` / scalar
`lfrzn`	species frozen status	integer*	A `nspe` / scalar
`ltp`	particle species number	integer*	A `maxdim` / A `nsyst`
`ltm`	particle molecule type number	integer*	A `maxdim` / A `nsyst`
`bndtbl`	bond table	integer*	A `numbond`,3 / A `numbond`,2
`rcii`	length scale for like-like conservative interactions	real (KIND=dp)	
`mole`	molecule index	integer	A `nsyst`
`bead1, bead2`	extremes of a bond	integer	
`nbdmol`	number of beads per molecule of given type	integer	A `nmoldef`
`nbomol`	number of bonds per molecule of given type	integer	A `nmoldef`

Table 1: Variables needed for HISTORY files in DL_MESO and in the present module.

The Table 1 contains the variables used in `format_history.f90` and is organized in three blocks:

- The first block contains those variables that maintain in `format_history.f90` exactly (name and notation) with their meaning in DL_MESO.
- The second block contains variables that are present both in DL_MESO and `format_history.f90`, but with a different dimensionality. Both shapes are indicated in the last column, separated by a "/". For example, some variables are arrays in DL_MESO and are used in `format_history.f90` as scalars, one component at a time. Or, in the case of `bndtbl`, its dimension in the second index is lowered from 3 (DL_MESO) to 2, because the type of bond is not recorded in the HISTORY files.
- The third block contains newly defined variables.

[1] The "notation" column indicates "the restrictions applicable on the parameters. 'A' indicates an array of data, followed by the number of elements in the array. For example, 'A maxdim' means the parameter is actually an array with maxdim elements (numbered from 1 to maxdim). 'P' indicates a pointer array, followed by the number of elements. '≥ 1' means the number must be greater or equal to one, while for a Boolean parameter 'T or F' means its value can either be .true. or .false.. An asterisk in the data type for the array indicates that it is allocatable and defined during the run." (from Ref. [3], page 170).

Caveat:

- `srfx, srfy, srfz` are actually written by DL_MESO as `srftype*srfx,srftype*srfy, srftype*srfz`, where `srftype` has values $0, 1, 2, 3$ and `srfx,sfry,sfrz` have values $0, 1$. For simplicity, the same name is used when reading the product of the two.
- Concerning `bndtbl`: no re-arrangement is done on it, the table is filled with the bonds in the order they are read. It means there is *no ordering* of the labels (i.e., the bond $4-5$ can appear before the bond $2-3$).

About the newly defined variables:

- `rcii` is written by DL_MESO as the appropriate component of the table `vvv(:,:)`. Depending on the potential type, it is σ_{ii} or $R_{c,ii}$ (see table 12.4 of Ref. [3]).
- `bead1, bead2` are written by DL_MESO as appropriate components of the table `bndtbl(:,:)`.

Other:

- The variable `mbeads` is a real version of the variable `nbeads`.

5 Parallel runs: multiple HISTORY* files

If DL_MESO_DPD is run using openMPI, then multiple HISTORY* files are produced, one per process. The simulated volume is decomposed into domain cells, and each process deals with one of them (and therefore with a subset `nbeads` of the total number of beads). Notice that the number of beads in a domain is not constant in time, it may vary from one snapshot to the other. Concerning bonds, two options are possible: they can be dealt with 1) locally (default) or 2) globally (using the directive `global bonds` in the CONTROL file, see [3]). At the level of HISTORY* files, this implies that either 1) the information about bonds is *all* in the HISTORY000000 file, or 2) it is split into the various files.

The general structure of all the HISTORY* files is, in any case, as follows:

- (1 line): nspe, nmoldef, nusyst, nsyst, nbeads ($\#_1$), nbonds ($\#_2$)
- (1 line): dimx, dimy, dimz, volm
- (1 line): keytrj, srfx, srfy, srfz
- (`nspe` lines): namspe, amass, rcii, lfrzn
- (`nmoldef` lines): nammol
- (1 lines): Simulation name
- ($\#_1$ lines): global, species, molecule, chain
- ($\#_2$ lines): bead1, bead2
- (for each snapshot "k"):
  - (1 line): time, mbeads ($\#_3(k)$), dimx, dimy, dimz, shrdx, shrdy, shrdz
  - ($\#_3(k)$ lines): global, x, y, z
- (1 line): time, mbeads ($\#_3(N_{av})$), dimx, dimy, dimz, shrdx, shrdy, shrdz

Therefore, the total number of lines of each file is:

$$N_{tot}^{lin} = 3 + \texttt{nspe} + \texttt{nmoldef} + 1 + \#_1 + \#_2 + \sum_{k=1}^{N_{av}} (1 + \#_3(k)) + 1 \tag{1}$$

where N_{av} is the total number of snapshots.

The colors above put in evidence the similarities between different HISTORY* files: in the first block of lines ending with the simulation name, all the files coincide, except for the two integers in red; in the rest of the file, after the simulation name, they all differ, except for the quantities in green.

In the case of multiple HISTORY* files, `format_history.f90` checks that the various files agree on the values of the following quantities: `nspe`, `nmoldef`, `nusyst`, `nsyst`, `dimx`, `dimy`, `dimz`, `volm`, `keytrj`, `srfx`, `srfy`, `srfz`, `namspe`, `nammol`, and the simulation name. If this is not the case, an error message is printed to the standard output.

The formula above is also valid in the case of a single HISTORY file. In this case it simplifies since $\#_3(k) = \#_1, \forall k$: in fact, during the simulation the (only available) process deals with all the beads in the system, at any time step.

References

[1] `www.ccp5.ac.uk/DL_MESO`

[2] M. A. Seaton, R. L. Anderson, S. Metz and W. Smith, Mol. Sim. **39** (10), 796 (2013).

[3] M. A. Seaton and W. Smith, *DL_MESO User Manual* - Version 2.6, November 2015.