

Information about `gen_dipoleaf.f90` and `gen_moldipaf.f90`

M. A. Seaton and S. Chiacchiera

August 23, 2017

The module `gen_dipoleaf.f90` is a generalization of the `dipoleaf.f90` post-processing utility of DL_MESO _DPD, the Dissipative Particle Dynamics (DPD) code from the DL_MESO package [1, 2]. The module `gen_moldipaf.f90` has a similar purpose, but applies to microscopic rather than macroscopic charge dipole moments. The modules are to be used with DL_MESO in its last released version, version 2.6 (dating November 2015).

They process the trajectory (HISTORY) files to obtain the charge dipole moments of all the (neutral) molecules in the system, and subsequently compute the dipole autocorrelation functions (DAFs) for each molecule type. As an output, `gen_dipoleaf.f90` (`gen_moldipaf.f90`) produces a file `DIPAFDAT` (`MDIPAFDAT`) containing both the un-normalized and normalized DAFs, and, optionally, a file `DIPAFFFT` (`MDIPAFFFT`) containing the Fourier transform of the latter.

The modules can be applied to systems including molecules with a generic charge structure, as long as each molecule is neutral (otherwise the charge dipole moment would be frame-dependent).

CAVEAT: the modules only analyze molecular trajectories. If a charged molecule is present, an error message will be given, while unbonded charges would not be detected and would lead to erroneous results. Therefore please keep in mind **to not apply** the modules to systems with unbonded charges.

1 Disambiguation on the concept of molecule

In DL_MESO a *defined molecule* is a set of beads, which can be bonded or not.

For the purpose of these modules it is *required* that each molecule is a connected cluster (via stretching bonds). In fact, this, together with the reasonable assumption that each stretching bond cannot be stretched to more than half the system linear size, allows to univocally define the charge dipole moment of each molecule.

2 Charge dipole moments

The charge dipole moment of a neutral molecule is

$$\vec{p}_{mol} = \sum_{i \in mol} q_i \vec{r}_i \quad (1)$$

where \vec{r}_i are the bead positions and q_i their charges. The total charge dipole moment of the simulated volume V is $\vec{P} = \sum_{mol \in V} \vec{p}_{mol}$. If two or more molecular species are present, one can split \vec{P} into the different species contributions:

$$\vec{P} = \sum_{j=1}^{N_{moldef}} \vec{P}^{(j)} = \sum_{j=1}^{N_{moldef}} \sum_{k=1}^{N_{mol}(j)} \vec{p}_k^{(j)}. \quad (2)$$

where N_{moldef} is the number of molecule types (definitions) and $N_{mol}(j)$ the number of molecules of type j .

3 Time autocorrelation functions

Given a scalar quantity A , its non-normalized time autocorrelation function is [3]

$$C_{AA}(t) = \langle A(0)A(t) \rangle \quad (3)$$

where $\langle \dots \rangle$ indicates an average over trajectories. The normalized one is [3]

$$c_{AA}(t) = \frac{\langle A(0)A(t) \rangle}{\langle A(0)A(0) \rangle} = \frac{C_{AA}(t)}{C_{AA}(0)} . \quad (4)$$

Averaging over trajectories means that different time origins will be considered, and different particles, if the quantity is a microscopic one.

Of course in the simulation the time variable is discretized (time step), and typically not all the time steps are recorded in the HISTORY* files, but only at certain intervals (called dt in this subroutine [5]).

Let us use τ to label successive time steps, $t = \tau dt$: the discretized version of the trajectory average is

$$C_{AA}(\tau) = \langle A(0)A(\tau) \rangle = \frac{1}{N_{samp}} \sum_{\tau_0=1}^{N_{samp}} A(\tau_0)A(\tau_0 + \tau) , \quad (5)$$

that is, we average over different time origins as mentioned above. On the other hand, the index τ varies between 1 and N_{af} , a value set by the user (see below).

To have the *same* statistics for any value of τ , we fix N_{samp} according to the larger time interval that is needed for the AFs: $N_{samp} = N_{traj} - N_{af} + 1$, where N_{traj} is the number of snapshots in the HISTORY file. In Figure 1, to clarify this point, we sketch a toy case with $N_{traj} = 6$ and $N_{af} = 4$: clearly, $N_{samp} = 3$.

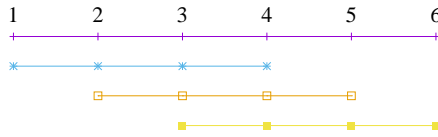


Figure 1: Sketch of the samples involved in AFs.

Of course, to have good statistics for the AFs, the simulation must be much longer than the maximum time interval for which we would like to analyze the correlations ($N_{traj} \gg N_{af}$). Given a certain HISTORY file, the larger N_{naf} the smaller N_{samp} (i.e, the worse the statistics).

Finally, a comment on the case in which A is a microscopic quantity: in this case Eq. (5) involves also an average over microscopic degrees of freedom. If, say, a is molecular quantity, and there are N_{mol} molecules, then

$$C_{aa}(\tau) = \langle a(0)a(\tau) \rangle = \frac{1}{N_{samp}N_{mol}} \sum_{\tau_0=1}^{N_{samp}} \sum_{i=1}^{N_{mol}} a_i(\tau_0)a_i(\tau_0 + \tau) . \quad (6)$$

4 Installing

The present modules require the library FFTW (version 3.x) [4] to be installed.

5 Compiling

The present modules, `gen_dipoleaf.f90` and `gen_moldipaf.f90`, are compiled with the available Fortran90 compiler, e.g.:

```
gfortran -lfftw3 -lm -o gen_dipoleaf.exe gen_dipoleaf.f90
gfortran -lfftw3 -lm -o gen_moldipaf.exe gen_moldipaf.f90
```

and the executable must be in the same directory of the HISTORY* files to be analyzed. In case the file `fftw3.f`, containing constants that are necessary for the Fourier transform, is not found by the compiler, a simple way out is to copy it in the same directory where the module is run.

6 Input

The user is asked to provide the number of nodes used to run the simulation, the electric charges for all the types of beads and `naf`, that sets the maximum time difference that is required (in terms of number of snapshots) for the AFs. If the trajectories are stored at intervals dt [5], the AFs are computed up to a maximum time $t_{max}=(naf-1)*dt$. The last input is a switch for the FT: 1 for *yes*, 0 (or any other integer) for *no*.

To input these parameters one can enter them from the keyboard or write them into a text file (say, `input.txt`), one per line (in the right order) and run the program in this way:

```
gen_dipoleaf.exe < input.txt
gen_moldipaf.exe < input.txt
```

7 Hints for a good usage

For the modules to produce clean results for the AFs it is necessary that

1. the trajectory sampling in HISTORY files is frequent enough (this is set when running the simulation)
2. the simulation is much longer than the time interval over which correlations are analyzed ($N_{traj} \gg N_{af}$) (this is set both when running the simulation and when choosing `naf` during the analysis).

8 Important variables

<code>numtraj</code>	computed	integer	number of snapshots in each HISTORY* file
<code>dt</code>	derived	real(KIND=dp)	time interval between consecutive snapshots in HISTORY*
<code>naf</code>	input	integer	maximum number of snapshots contained in a sample for AFs
<code>nsamp</code>	derived	integer	number of samples

9 Fourier transform

The Fourier transform (FT) of the autocorrelation function can be optionally computed. The continuum FT of a time function $A(t)$ is a function of the (angular) frequency ω

$$\hat{A}(\omega) = \int_{-\infty}^{\infty} dt A(t)e^{-i\omega t} \quad (7)$$

and the inverse FT is¹

$$A(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} d\omega \hat{A}(\omega)e^{i\omega t} . \quad (8)$$

¹Notice that a number of slightly different definitions are possible.

Consider now a function defined at n points in time, at intervals δt . The discrete Fourier transform (DFT) and its inverse are:

$$\hat{A}_k = \sum_{m=0}^{n-1} A_m e^{-i\omega_k t_m} = \sum_{m=0}^{n-1} A_m e^{-i2\pi km/n} \quad (9)$$

$$A_m = \frac{1}{n} \sum_{k=0}^{n-1} \hat{A}_k e^{i\omega_k t_m} = \frac{1}{n} \sum_{k=0}^{n-1} \hat{A}_k e^{i2\pi km/n} \quad (10)$$

where $t_m = m\delta t$, $\omega_k = k\delta\omega$ and $\delta\omega = \frac{2\pi}{n\delta t}$. Also, we use $\hat{A}_k \equiv \hat{A}(\omega_k)$ and $A_m \equiv A(t_m)$.

The DFT in the present modules is computed via the FFTW package [4], version 3.x, in its implementation for Legacy Fortran. The package computes un-normalized DTFs for arbitrary sizes (n) of the input array². Please refer to the documentation of the FFTW package for further information.

10 Output

10.1 Output file(s) for gen_dipoleaf.f90

The header of the file DIPAFDAT contains the simulation title and a line with the number of snapshots in HISTORY and of those used for the AFs (**naf**). Then a block follows for each molecule type, started by the *molecule name*, then three columns of data,

$$t, C_{\vec{p}\vec{p}}, c_{\vec{p}\vec{p}}. \quad (11)$$

It is intended that in any block only the contribution to \vec{P} from a given species is used. The last block is called *all species* and refers to the full system charge dipole moment. The header of the file DIPAFFFT is as for DIPAFDAT (notice that the number of points for the FT is also set equal to **naf**). Then a block follows for each molecule type, started by the molecule name, then three columns of data

$$\omega, \Re[\hat{c}_{\vec{p}\vec{p}}(\omega)], \Im[\hat{c}_{\vec{p}\vec{p}}(\omega)], \quad (12)$$

where \hat{c} is the discrete FT of $c(t)$.

10.2 Output file(s) for gen_moldipaf.f90

The files have the same structure as above, but the macroscopic \vec{P} is replaced by microscopic \vec{p} :

$$t, C_{\vec{p}\vec{p}}, c_{\vec{p}\vec{p}}. \quad (13)$$

and

$$\omega, \Re[\hat{c}_{\vec{p}\vec{p}}(\omega)], \Im[\hat{c}_{\vec{p}\vec{p}}(\omega)]. \quad (14)$$

Please notice that the averages over trajectories in this case involve an average over particles too (beside the average over different time origins).

11 Checks

The present modules perform the same checks on the molecules (labelling, connectivity, charge neutrality, content) described in the the manual for gen_dipole.f90. Please refer to Section 7 of that document for more details.

²In this case un-normalized means that the DFTs are computed according to Eqs. (9) and (10), a part from the factor $1/n$ in the second one. The algorithm works for generic n , however sizes that are products of small factors are transformed most efficiently.

12 Test: water in oil

As a test, we suggest to consider a fluid made of harmonically bonded dimers ($+q, -q$). Fixing appropriately the partial charge q and the Bjerrum length l_B this system mimics water in an oil background, as long as the dielectric properties are concerned. For more details about this model, please see Section 13.

Run DL_MESO_DPD using for the CONTROL file

```
-----  
DL_MESO charged harmonic dimers with dpd repulsion
```

```
volume 64.0  
temperature 1.0  
cutoff 1.0  
  
timestep 0.01  
steps 70000  
equilibration steps 20000  
traj 20000 10  
stats every 100  
stack size 100  
print every 100  
job time 7200.0  
close time 10.0  
  
ensemble nvt mdv  
  
ewald sum 1.0 5 5 5  
bjerrum 42.0  
smear gauss equal
```

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

and for the FIELD file

```
-----  
DL_MESO charged harmonic dimers with dpd repulsion
```

```
SPECIES 2  
solp 1.0 0.46 0  
solm 1.0 -0.46 0  
  
MOLECULES 1  
DIMER  
nummols 96  
beads 2  
solp 0.0 0.0 0.0  
solm 0.1 0.0 0.0  
bonds 1  
harm 1 2 5.0 0.0
```

```
finish
```

```
INTERACTIONS 3  
solp solp dpd 25.0 1.0 4.5  
solm solm dpd 25.0 1.0 4.5  
solp solm dpd 25.0 1.0 4.5
```

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

Analyzing the HISTORY file with `gen_dipoleaf.exe` choosing `naf=100`, i.e. using this `input.txt` (which assumes a serial run)

```
-----  
1  
0.46  
-0.46  
100  
1  
-----
```

this output is printed on the standard output

```

-----
Number of nodes used in calculations ?
nchist:      0      96      0      0      0      0      0      0      0      0
Charges for SPECIES type solp   :
Charges for SPECIES type solm   :
chg=      0.4600      -0.4600
Number of time steps in autocorrelation profile?
switch for FFT computation? (1=yes, 0 or any other integer=no)
-----

```

The first line shows the histogram of cluster sizes: in this case, it correctly gives 96 molecules of two beads. Since internally the module checks that each molecule is a connected cluster, this line should always give a histogram with the molecule sizes (by default, shown up to ten beads).

The DIPAFDAT file is (only the first fifteen lines are shown)

```

-----
DL_MESO charged harmonic dimers with dpd repulsion
      5001      100
-----

```

```

DIMER
0.000000E+00  1.500096E+01  1.000000E+00
1.000000E-01  1.443052E+01  9.619731E-01
2.000000E-01  1.301975E+01  8.679278E-01
3.000000E-01  1.123656E+01  7.490557E-01
4.000000E-01  9.404209E+00  6.269071E-01
5.000000E-01  7.679286E+00  5.119196E-01
6.000000E-01  6.134042E+00  4.089099E-01
7.000000E-01  4.803145E+00  3.201892E-01
8.000000E-01  3.692590E+00  2.461569E-01
9.000000E-01  2.781081E+00  1.853935E-01
-----

```

and the DIPAFFFT file is (only the first fifteen lines are shown)

```

-----
DL_MESO charged harmonic dimers with dpd repulsion
      5001      100
-----

```

```

DIMER
0.000000E+00  6.233649E+00  0.000000E+00
6.283185E-01  5.405816E+00 -1.412683E+00
1.256637E+00  6.257941E+00 -2.520388E+00
1.884956E+00  4.150463E+00 -2.387533E+00
2.513274E+00  2.693334E+00 -3.912521E+00
3.141593E+00  2.131104E+00 -3.246574E+00
3.769911E+00  1.650412E+00 -2.860556E+00
4.398230E+00  1.278705E+00 -2.801992E+00
5.026548E+00  1.124330E+00 -2.102817E+00
5.654867E+00  9.385175E-01 -1.890993E+00
-----

```

Analyzing the HISTORY file with `gen_moldipaf.exe` choosing `naf=100`, i.e. using the `input.txt` given above, this output is printed on the standard output

```

-----
Number of nodes used in calculations ?
nchist:      0      96      0      0      0      0      0      0      0      0
Charges for SPECIES type solp   :
Charges for SPECIES type solm   :
chg=      0.4600      -0.4600
Number of time steps in autocorrelation profile?
switch for FFT computation? (1=yes, 0 or any other integer=no)
-----

```

The MDIPAFDAT file is (only the first fifteen lines are shown)

```

-----
DL_MESO charged harmonic dimers with dpd repulsion
      5001      100
-----

```

```

DIMER

```

```

0.000000E+00  1.419184E-01  1.000000E+00
1.000000E-01  1.360256E-01  9.584780E-01
2.000000E-01  1.217816E-01  8.581103E-01
3.000000E-01  1.045570E-01  7.367406E-01
4.000000E-01  8.750295E-02  6.165724E-01
5.000000E-01  7.161709E-02  5.046358E-01
6.000000E-01  5.723649E-02  4.033057E-01
7.000000E-01  4.466587E-02  3.147293E-01
8.000000E-01  3.406310E-02  2.400189E-01
9.000000E-01  2.537519E-02  1.788013E-01

```

and the MDIPAFFFT file is (only the first fifteen lines are shown)

```

DL_MESO charged harmonic dimers with dpd repulsion
5001      100

```

```

DIMER
0.000000E+00  6.013063E+00  0.000000E+00
6.283185E-01  5.915405E+00 -1.378369E+00
1.256637E+00  5.283007E+00 -2.365590E+00
1.884956E+00  4.077657E+00 -3.150028E+00
2.513274E+00  3.076368E+00 -3.326590E+00
3.141593E+00  2.244706E+00 -3.195932E+00
3.769911E+00  1.631808E+00 -2.873471E+00
4.398230E+00  1.273571E+00 -2.521504E+00
5.026548E+00  1.024723E+00 -2.219809E+00
5.654867E+00  8.805952E-01 -1.950491E+00

```

In Figure 2 we show, on the left, a plot of the normalized AF $\frac{\langle \bar{P}(0)\bar{P}(t) \rangle}{\langle \bar{P}(0)\bar{P}(0) \rangle}$ (obtained using the first and

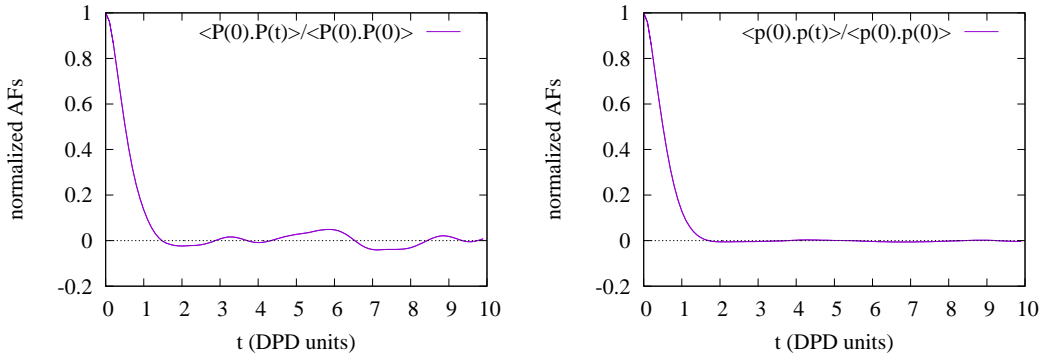
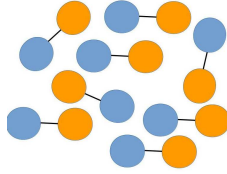


Figure 2: Macroscopic (left) and microscopic (right) AFs.

third columns of DIPAFDAT) and, on the right, a plot of the normalized AF $\frac{\langle \bar{p}(0)\bar{p}(t) \rangle}{\langle \bar{p}(0)\bar{p}(0) \rangle}$ (obtained using the first and third columns of MDIPAFDAT).

13 Test case: a dimer solvent



Here we describe a physical system in which the software modules dealing with charge dipole moments in DPD simulations can be tested. It is a *polarizable fluid* made of harmonically bonded dimers $(+q, -q)$, pictorially represented above (not in scale). Fixing appropriately the partial charge q and the Bjerrum length l_B , this system mimics *water in an oil background*, as long as the dielectric properties are concerned.

We recall that the electric permittivity is ϵ_0 for vacuum, and $\epsilon < \epsilon_0$ for a medium. The medium effect can be split into a background and a relative term $\epsilon/\epsilon_0 = \epsilon_b \epsilon_r$. The background is constant and uniform, whereas the explicit term is due to dynamic microscopic objects (dimers in this case) which carry a charge dipole moment. The strength of electrostatic interactions in a background is set by the bare Bjerrum length $l_B = e^2/(4\pi\epsilon_0\epsilon_b k_B T)$. On the other hand, from linear response theory, the bulk value of the relative permittivity is $\epsilon_r = 1 + \frac{\langle \vec{P}^2 \rangle_{\vec{E}=\vec{0}}}{3\epsilon_0\epsilon_b k_B T V}$, where tin-foil boundary conditions are assumed.

Two types of beads are present in the simulation, `solp` and `solm`, the solvent positive and negative partial charges, respectively. We fix the bare Bjerrum length $l_B = 42$ (appropriate for oil³), the repulsion parameter $A = 25$, the harmonic spring constant $k = 5$, the bead density $\rho = 3$, the partial charges $|q| = 0.46$ and the Gaussian smearing length $\sigma = 0.5$. All quantities are given in DPD units, where $k_B T = 1$, $r_c = 1$ and $m = 1$. This fluid has a relative permittivity $\epsilon_r \simeq 40$, as can be checked using the `gen_dipole.f90` utility. This value is compatible with the ratio of water and oil permittivities $\epsilon^{water}/\epsilon^{oil} \simeq 40$.

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. P. Allen and D. J. Tildesley, “Computer simulation of liquids”, Oxford University Press, Oxford (1987).
- [4] <http://www.fftw.org/>
- [5] In the main code dt is the simulation time step, but here only the time interval between consecutive snapshots is present, therefore there is no risk of confusion.

³Notice that the physical length scale is set choosing r_c : if we choose $r_c = 0.646nm$ (appropriate to match water density at room temperature if $N_m = 3$, i.e., one bead represents three water molecules), the Bjerrum length of oil in DPD units is $l_B = 27nm \simeq 42r_c$, hence the value given above for the oil background.