

# POST-PROMD: an open-source POST-PROcessing software for Molecular Dynamics simulations

Ashkan Shekaari<sup>1</sup>

<sup>1</sup>Department of Physics, K. N. Toosi University of Technology, Tehran 15875-4416, Iran

## Summary

In this work, a new software named POST-PROMD has been developed, which is capable of computing three main phase transition quantities used in the area of computational condensed matter physics, including the Lindemann index, mean square displacement (MSD) of atoms, and radial distribution function (RDF) of systems of particles only via post-processing molecular dynamics (MD) simulations outputs. POST-PROMD runs independently (is not a module patched to any other software), is length-scale-independent (consistent with both quantum mechanical and classical simulations), is released for the first time under the GNU GPL (Version 3) as a free and open-source package, has originally been written to post-process the atomic-positions output file generated during Car-Parrinello MD simulations of QUANTUM ESPRESSO, and is capable of post-processing MD output files of any other software as well.

## Statement of Need

One of the most important capabilities of QUANTUM ESPRESSO [1] is to perform density functional molecular dynamics (DFMD) simulations within both Car-Parrinello (CP) [2] and Born-Oppenheimer (BO) [3, 4] frameworks. However, there is still a lack of required codes/modules to post-process raw outputs of such MD simulations generated by the CPV package. As an illustration, the output file "`cp.pos`" of CPMD simulations—including atomic positions of the system at every time-step—contains important information about phase transition/finite-temperature behavior/dynamics of that system, extracting information from which is left to the user.

To this end, i.e., to extract such important information contained in "`cp.pos`", we have written a post-processing package for MD simulations, called POST-PROMD (acronym for "a POST-PROcessor for Molecular Dynamics simulations"), which computes three main phase transition indicators, including (i) the Lindemann index of the whole system, (ii) mean square displacement (MSD) of every atom/particle of the system, and (iii) radial distribution function (RDF) of the whole system.

The present package has been written both in Fortran 95 [5] and Linux Bash [6, 7], and is released as a free and open-source package, with assigned version `v.1.0.0`, under the GNU General Public License (Version 3, 29 June 2007) [8].

## State of the Field

There is no software tools to compute the aforementioned phase transition quantities for QUANTUM ESPRESSO users, and this is the main reason why the CPV package of QUANTUM ESPRESSO

has not yet become a standard in quantum MD simulations. As a result, the POST-PROMD package fills the gap.

## Phase Transition in Solids

Investigating phase transition/finite-temperature behavior of the condensed matter at the computational level of research using MD simulations is carried out via calculating a number of indicators the three important of which are the Lindemann index, MSD, and RDF, as mentioned before. These quantities are prevalently used to probe phase transition in solid materials, namely solid-to-liquid transition; however, they can also be applied to so-called few-body systems of atoms/molecules whose bulk phases are not solid. The point at which phase transition takes place is the same as the melting temperature ( $T_m$ ) of the solid; at this point, temperature dependence of the three phase transition indicators aforementioned undergo a sudden, substantial increase compared to their former behaviors. The facts aforementioned have clearly been demonstrated in an investigation on phase transition of B<sub>36</sub> nanocluster [9]. In the following, we describe the three phase transition quantities.

### The Lindemann Index

The Lindemann index  $\Delta_{rms}$ , also known as global Lindemann index, or root-mean-square bond length fluctuation [10], is indeed the extent of fluctuations in bond lengths between atoms/particles of the system, averaged both over all those atoms, and the total time span of the MD simulation. This quantity is defined as

$$\Delta_{rms} = \frac{2}{N(N-1)} \sum_{i>j} \frac{\left( \langle r_{ij}^2 \rangle_t - \langle r_{ij} \rangle_t^2 \right)^{1/2}}{\langle r_{ij} \rangle_t}, \quad (1)$$

where  $N$  is total number of atoms,  $r_{ij}$  is distance between atoms  $i$  and  $j$ , and  $\langle \dots \rangle_t$  is time ( $t$ ) average over the entire trajectory/simulation.  $2/N(N-1)$  also means average over total number of atomic pairs/bonds  $N(N-1)/2$ . The two modules of POST-PROMD, namely "**lind.sh**" and "**ln.source**" are used to compute the Lindemann index of the system according to Eq. 1.

### Mean Square Displacement (MSD)

In contrast to the Lindemann index and radial distribution function, which are quantities related to the entire system, MSD is a single-particle quantity, and POST-PROMD consequently computes it for every individual atom of the system according to

$$\langle \mathbf{r}_I^2(t) \rangle = \frac{1}{n} \sum_{i=1}^n \left[ \mathbf{R}_I(t_{0i} + t) - \mathbf{R}_I(t_{0i}) \right]^2, \quad (2)$$

where we average over  $n$  different time origins  $t_{0i}$  distributed over the entire trajectory with the interval  $\text{dt0}$  (included as an input in the file "**par.input**", described later on in section 1) between any two consecutive  $t_{0i}$  for the average. The  $I$  runs over number  $N$  of atoms, and  $\mathbf{R}_I$  is the position vector of the  $I$ th atom with respect to the center of mass (COM) of the system. The two modules "**msd.sh**" and "**msd.source**" compute MSD according to Eq. 2.

## Radial Distribution Function (RDF)

The quantity RDF [ $g(r)$ ], being calculated both in two and three spatial dimensions, computes the average number  $\langle N \rangle$  of atoms within the shell volume  $V_s$  ( $= 2\pi r dr$  in two, and  $4\pi r^2 dr$  in three dimensions) of radius  $r$  around COM of the system, between  $r - dr/2$  and  $r + dr/2$ , according to

$$g(r) = \frac{\langle N \rangle}{N} / V_s, \quad (3)$$

where  $dr$  is shell width. Indeed, POST-PROMD computes the average of atoms within each shell over the entire simulation, and then repeats the same for the next shell until reaching the outermost that encompasses the entire system. This is why RDF calculation takes a longer time compared to the Lindemann index or MSD. The two modules `"rdf.sh"` and `"rdf.source"` compute RDF according to Eq. 3.

### 1 The Input

So far, we have demonstrated POST-PROMD modules that perform calculations (placed in `<POST-PROMD_ROOT_DIRECTORY/src>`). The Bash scripts suffixed with `".sh"` use the same input files `"cp.pos"` and `"par.input"` to do calculation. The former (`"cp.pos"`) is indeed an output file, formerly generated by the CPV package of QUANTUM ESPRESSO; the latter (`"par.input"`), on the other hand, is completely user-defined, containing a number of numerical parameters some of which have already been defined in MD input file of QUANTUM ESPRESSO. These input parameters are described in detail, as follows:

- `n0` = Total number of particles (also included in MD input file of QUANTUM ESPRESSO).
- `nstep0` = Total number of steps (also included in MD input file of QUANTUM ESPRESSO).
- `iprint0` = Number of steps between successive output writings (also included in MD input file of QUANTUM ESPRESSO).
- `ign0` = Number of steps for thermalization/thermal equilibration, to have reliable statistical averages. In fact, whether or not the initial atomic positions are random, it takes a time for the entire system to reach thermal equilibrium according to the target temperature defined in the simulation. Over this time interval, statistical averages, such as total energy, are not valid because they are only meaningful at thermal equilibrium according to theory (thermodynamics/statistical mechanics). Usually, integer values between 50 and 500 steps are suitable. Evidently, systems with larger numbers of particles take longer times/steps to reach thermal equilibrium (to be more accurate, the user has to check the diagram of total energy vs. simulation step/index to determine the time/step at which total energy begins to get converged).
- `dt0` = Time-step, included in (1st row, 2nd column) of the MD output file `"cp.pos"`.
- `dt00` = Time-step in MD input file of QUANTUM ESPRESSO.
- `delta_r0` = Increment value in radius  $r$  ranging from zero (COM of the system) to the outermost shell encompassing the whole system. It is a very sensitive parameter; the value 0.0005 Å works well.

- `rmax0` = Distance from COM to the outermost shell of the system. For example, the value 5 Å is suitable for B<sub>36</sub> nanocluster [9].

Note that in "`par.input`", there must not be any blank space before and after the equal signs (=). The sample input files ("`cp.pos`" and "`par.input`") have also been placed in `<POST-PROMD_ROOT_DIRECTORY/input>`.

## How to Run

Needless to install, POST-PROMD would be ready to run only via unzipping the source file `post-promd-v.1.0.0.tar.gz`. To this end, the user needs to pursue the following simple steps:

1. Unzip the source file.
2. Place "`cp.pos`" in `<POST-PROMD_ROOT_DIRECTORY/src>`
3. Edit "`par.input`" according to your MD setup, then place it in `<POST-PROMD_ROOT_DIRECTORY/src>`
4. Open a terminal in `<POST-PROMD_ROOT_DIRECTORY/src>`
5. To compute the Lindemann index, run in terminal: `bash lind.sh`
6. To compute MSD, run in terminal: `bash msd.sh`
7. To compute RDF, run in terminal: `bash rdf.sh`

Note that the Fortran part of POST-PROMD, namely the files suffixed by "`.source`", are compatible with the Intel Fortran Compiler `ifort`; therefore, the user either has to use `ifort`, or modify the "`.source`" files simply via replacing the keyword "`ifort`" with other available Fortran compilers, such as `gfortran`, `mpif90`, etc. The output files generated by POST-PROMD are also described as follows:

1. The output of `bash lind.sh` is displayed in terminal.
2. Those of `bash msd.sh` will be in the directory `<msd_output_files>`, each of which belongs to one atom of the system. Each of these files, named with the template "`msd***.out`", contain two columns of numerical data, which, in order, are: time-step, and MSD. The user has to plot these files.
3. The output of `bash rdf.sh`, namely "`rdf5.out`", will be in the directory `<rdf_output_file>`; the number '5' stands for the five columns of numerical data contained in that file, being, in order: `r` (radius), `rdf` (RDF values) on `xy` plane, `rdf` on `xz` plane, `rdf` on `yz` plane, and `rdf` in three dimensions. The user also has to plot "`rdf5.out`".

## Example Files

The present distribution of POST-PROMD also contains example files, placed in the directory <POST-PROMD\_ROOT\_DIRECTORY/example>, which are related to the CPMD simulation of the unit cell of SLSiN (single-layer silicon nitride; chemical formula:  $\text{Si}_3\text{N}_4$ ; containing 14 atoms) [11], at  $T = 5$  K. The related outputs have also been placed in the directory <POST-PROMD\_ROOT\_DIRECTORY/example/reference>.

## References

- [1] P. Giannozzi, S. Baroni, N. Bonini, M. Calandra, R. Car, C. Cavazzoni, D. Ceresoli, G.L. Chiarotti, M. Cococcioni, I. Dabo, A.D. Corso, S. de Gironcoli, S. Fabris, G. Fratesi, R. Gebauer, U. Gerstmann, C. Gougoussis, A. Kokalj, M. Lazzeri, L. Martin-Samos, N. Marzari, F. Mauri, R. Mazzarello, S. Paolini, A. Pasquarello, L. Paulatto, C. Sbraccia, S. Scandolo, G. Sclauzero, A.P. Seitsonen, A. Smogunov, P. Umari, R.M. Wentzcovitch, QUANTUM ESPRESSO: a modular and open-source software project for quantum simulations of materials, *J. Phys.: Cond. Matter* 21 (2009) 395502.
- [2] R. Car, M. Parrinello, The unified approach to density functional and molecular dynamics in real space, *Solid State Commun.* 62 (1987) 403.
- [3] D. Marx, J. Hutter, *Ab Initio Molecular Dynamics: Basic Theory and Advanced Methods*, 1st edn., Cambridge University Press, Cambridge, 2009.
- [4] M. Born, R. Oppenheimer, Zur quantentheorie der molekeln, *Ann. Phys.* 389 (1927) 457.
- [5] S.J. Chapman, *Fortran for Scientists and Engineers*, 4th edn., McGraw-Hill Education, 2 Penn Plaza, New York, NY 10121, 2018.
- [6] L. Torvalds, The Linux edge, *Commun. ACM* 42 (1999) 38.
- [7] <https://www.gnu.org/software/bash/>.
- [8] <https://www.gnu.org/licenses/gpl-3.0.txt>.
- [9] A. Shekaari, M. Jafari, Finite temperature properties and phase transition behavior of quasi-planar  $\text{B}_{36}$  nanocluster from first principles, *Mater. Res. Express* 6 (2019) 025014.
- [10] F.A. Lindemann, The calculation of molecular vibration frequencies, *Phys. Z.* 11 (1910) 609.
- [11] A. Shekaari, M. Jafari, Unveiling the first post-graphene member of silicon nitrides: A novel 2D material, *Comput. Mater. Sci.* 180 (2020) 109693.