Detailed hydrogen bond analysis of classical/ab initio molecular dynamics simulation using python

Marzieh Saeedimasine

November 2020

Department of Materials and Environmental Chemistry

Computer simulation methods are by now an established tool in many branches of science especially in studying the chemistry and physics of bio-nanomaterial [1]. Analyzing the trajectory of simulation is important to extract molecular information of interacting atoms. In this project, I was trying to analyze the trajectory of ab initio and classical molecular dynamics simulations using python [2]. Specifically, hydrogen bond formation was investigated for these trajectories and compared to other softwares results. In the following, the importance of hydrogen bond analysis in biological systems, different parts of the python code, and comparison of python code results with other softwares have been discussed.

Hydrogen bond analysis:

The presence of hydrogen bonds (Hbond) has been examined by two criterias: the distance between the donor and acceptor and the angle between the donor, hydrogen,and acceptor atoms. The cut-off values used for these parameters are 0.35 nm and 135°, respectively. The geometrical hydrogen bond criterion is shown in Figure1.

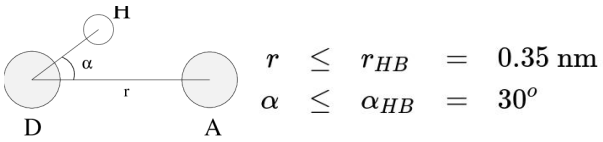

Figure 1: Geometrical hydrogen bond criterion (manual.gromacs.org).

In this project, one system composed of ZnO nanomaterial surface with 288 atoms interacting with 111 water molecules was simulated by ab initio molecular dynamics simulation using CP2K package [3]. Another larger system composed of graphene oxide surface with 600 atoms interacting with 4023 water molecules was also simulated by classical molecular dynamics simulation using GROMACS package [4].

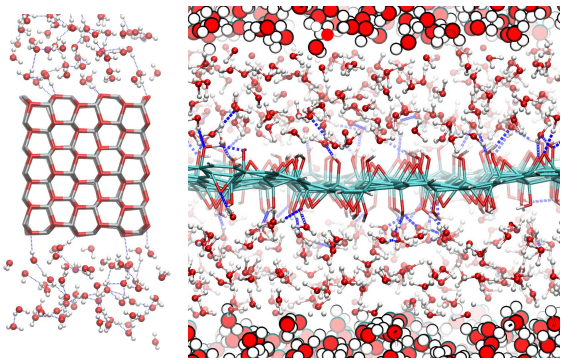

Figure2: Systematic representation of water molecules interacting with ZnO **(left)** and graphene oxide **(right)** nanomaterial surfaces after 100 frames of ab initio and classical molecular dynamics simulation, respectively. Hydrogen bonds are shown in dashed blue lines.

Python code for Hydrogen bond analysis:

Different parts implemented in the code is described as follow:

1- Importing libraries such as: *sys, pandas, numpy, itertools, math, matplotlib, time, threshing, and multiprocessing.*

2- Input that interacts with the user to get numbers of atoms, groups for hydrogen bond analysis, Donor-Acceptor distance and the angle cutoff.

3- Define functions:

Read_trajectory: reads each frame of trajectory with help of *pandas* and makes a list of atom names, and (x,y,z) cartesian coordinates of atoms.

Atom_groups: identify donor and acceptor atoms for selected groups by the user.

Angle: calculate the angle between three points using *numpy* library.

Hbond: finds Hbonds and prints tuples of donor, hydrogen, acceptor atoms numbers as well as Donor-Acceptor distance and the angle.

Thread: using *thread* for parallel computing of hydrogen bond for each frame of simulation with the help of *Queue* to return back the value from the thread.

Multiprocess: using *multiprocessing* for parallel computing hydrogen bond for each frame of simulation with the help of *Queue.*

Plot: plot the results using *matplotlib.*

In this project, **MDTraj**[1] [5], an open python library for analysis of molecular dynamics simulation, was also used to compare the result with my python code. **Mdtraj.load** was used to load the trajectory and coordination files, **topology.select** was used to select groups for Hbond calculation, **mdtraj.baker_hubbard** was used to find and print Hbond matrix. The criterias employed for finding Hbond is *theta* >120 and distance H-Acceptor <2.5 *A*.

Hydrogen bond analysis code for ab initio MD trajectory, classical MD trajectory, and MDTraj library are available in the github link:

https://github.com/marzieh-saeedimasine/Hbond-Computational-Python2020.git

[1]MDTraj webpage: https://mdtraj.org/1.9.4/index.html

Results and discussion:

Hydrogen bond analysis between the water molecules and ZnO surface as a function of frame of ab initio MD simulation is shown in Figure3. For comparison the hydrogen bond with the same criteria was analyzed with VMD software **[5]**. By increasing the simulation frames, water molecules are getting closer to the surface and make more Hbond with the surface. There are some differences between the results of my code and VMD software [6] that are labeled in the figure3 (right). It seems that my written code is performing better in finding the Hbond in the system.

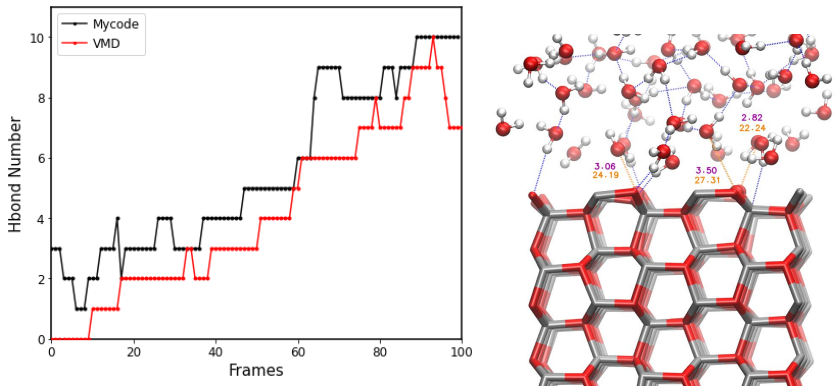

Figure 3: Number of hydrogen bonds as function of ab initio MD frames **(left)**. Hbond representations for interaction of water molecules with ZnO surface(right). Hbond differences are labeled by orange.

To study how concurrency can help to save time in python calculation, I have implemented threading and multiprocessing to the code and compare the wall time of Hbond analysis for ab initio MD simulation (Table 1). As can be seen, multiprocessing can improve the calculation performance while thread is not useful for this intensive calculation.

Table1: Comparison of the time needed for calculation of Hbond from ab initio MD for different concurrency methods in python

Method/ Wall time (s)	1 Frame	10 Frame	100 Frame
No Concurrency	1.57	15.1	138
Threading	1.38	15.1	463
Multiprocessing	1.44	2.58	22.3

The python code was also used to calculate Hbond of the trajectory (.pdb format) from classical MD simulation of water molecules interacting with graphene oxide surface which is shown in Figure 4. Comparison between the result of mycode and other software is showing that Hbond

analysis with Gromacs overestimates the number of Hbond while VMD shows similar results to my code. The number of Hbond calculated by MDTraj shows good agreement with my code.

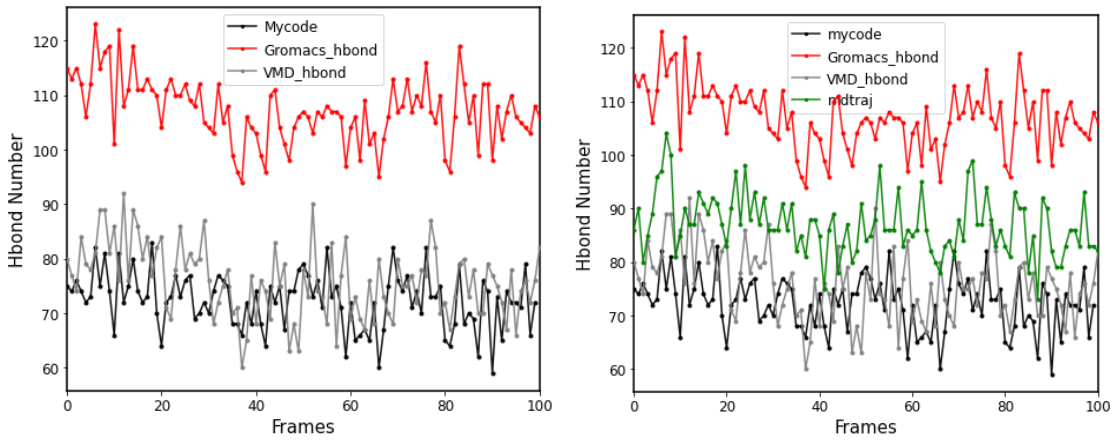

Figure 4: Comparison of number of Hbond between water molecules and graphene oxide surface as function of MD frames computed by python code and other software in the field.

Conclusion:

Analyzing the trajectory of ab initio/classical molecular dynamics simulation is important to extract molecular information of interacting atoms. Python can play a great role for data analysis in this field by the help of various libraries. In this project, python was implemented to analyze hydrogen bonds formed between water molecules and two different nanomaterial surfaces. Comparison between this result and other software calculations has also been made.

Hydrogen bond analysis codes of this project are available in the github link:
https://github.com/marzieh-saeedimasine/Hbond-Computational-Python2020.git

References:

[1] Allen, M. P., & Tildesley, D. J. (2017). *Computer simulation of liquids*. Oxford university press.

[2] Van Rossum, G., & Drake, F. L. (2009). *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace.

[3] Hutter, J., Iannuzzi, M., Schiffmann, F., & VandeVondele, J. (2014). cp2k: atomistic simulations of condensed matter systems. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, *4*(1), 15-25.

[4] Abraham, M. J., Murtola, T., Schulz, R., Páll, S., Smith, J. C., Hess, B., & Lindahl, E. (2015). GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX*, *1*, 19-25.

[5] McGibbon, R. T., Beauchamp, K. A., Harrigan, M. P., Klein, C., Swails, J. M., Hernández, C. X., ... & Pande, V. S. (2015). MDTraj: a modern open library for the analysis of molecular dynamics trajectories. *Biophysical journal*, *109*(8), 1528-1532.

[6] Humphrey, W., Dalke, A., & Schulten, K. (1996). VMD: visual molecular dynamics. *Journal of molecular graphics*, *14*(1), 33-38.