Introduction to Molecular Dynamics - 2019

Gromacs tutorial for protein simulations.

Phil Biggin philip.biggin@bioch.ox.ac.uk

Table of Contents.

1. About this Practical

This practical is designed to demonstrate what sorts of questions and problems occur when simulating proteins. It is divided into two parts. The first is designed to show you just how easy it is to run a protein simulation and perform some basic analysis. The second involves a little more exploration into some typical problems. As this practical course stems from the Wellcome Trust Structural Biology programme, students come from an extremely wide and varied range of backgrounds, and I therefore make very little assumption as to previous knowledge. The practical is both open-ended and in the analysis part is fairly independent of previous steps which means you can opt out of certain exercises if you wish. However I would recommend proceeding sequentially through all of it. The instructions get progressively less detailed as this would result in a very lengthy document, but if you get stuck simply ask. Finally, I assume very little knowledge of gromacs and VMD so if you are familiar with those packages already you will find things very easy.

2. Practicalities.

In this practical session, it is assumed that the user is reasonably familiar with some basic Linux/Unix command line tools. See the Appendix I - Basic Unix Commands if you need some help in this respect. In the following the "%" symbol is used to indicate the command line prompt. Where indicated enter everything after this symbol. For visualization we will use the freely available molecular graphics package, VMD.

If you are worried you have done something wrong or are not sure what the actual result should look like, you can look at the prerun directory within each section which has example output.

We will be using the molecular dynamics simulation package GROMACS. Extensive help is available online there both in html and as complete downloadable pdfs of the manual. The manual is worth downloading if you wish to understand some of the implementations of the algorithms and if you need further background into the subject of molecular dynamics.

NOTE: The trajectories and the practical are all based on using gromacs version 2018. If you are using an older version then some things probably will not work. The VMD instructions are based on version 1.9.2.

First we need to obtain all the files for the practical. Open a terminal and go to your home directory:

```
% cd
```

Then, get a copy of the OxCompBio repository:

```
% git clone https://github.com/bigginlab/OxCompBio.git
```

This will create a directory called OxCompBio, which icludes subdirectories for each tutorial of this course. You should change directory to the one correponding to today's MD tutorial.

```
% cd OxCompBio/tutorials/MD/
```

List all the files and subdirectories included in this location.

```
% ls
```

From there, you can open these instructions with a browser if you find that more convenient:

```
% firefox md-prac.html &
```

Work from here, as the commands that follow assume you are starting from this location. Next, make sure you have the GROMACS environment setup.

```
% gmx help commands
```

should display a list of the in-built tools of GROMACS. If it does not, contact a practical demonstrator. For help on a particular command, you can type

```
% gmx [command] -h
```

and it will print more information on the specific tool. Indeed, the practical below is really only a route to show you what sorts of things are possible. You should explore the options available at each stage by giving the -h option.

3. Specifics to GROMACS

- Coordinates files have the extension .gro and the default name is conf.gro
- The topology file (default name topol.top) contains all the information about which atoms are bonded to which and what force-field parameters are applied etc.
- The trajectory files have the extension .xtc and .trr, the former does not contain velocity information and coordinates are held at a reduced precision, and so occupies less disk space. However you will need velocities if you want to continue a simulation.
- The .edr file contains the energy information from the trajectory.
- The .mdp file contains the information that was used to setup the actual simulation. Things related to temperature, pressure, how the electrostatics is calculated etc,etc. Although we will not use this today, the file is provided so you can see how this trajectory was made.
- The .ndx file allows you to specify atoms or groups of atoms for use in analysis, restraints, etc and is optional
- The .tpr file is a binary file that contains all the information needed to perform the actual run (this allows gromacs to do lots of self-consistency checks to minimize user errors)

4. Tutorial 1 - A simple protein simulation

In this section, we will obtain our protein coordinates and perform some routine Molecular Dynamics calculations on them. First of all we need to obtain our coordinate. For this we will use the HIV-1 protease structure (1HSG). It is a homo-dimer with two chains of 99 residues each. Start the web-browser (firefox or konqueror) and go to the protein data bank (https://www.rcsb.org/) . Enter the pdb code 1HSG in the site search box at the top and hit the site search button. The protein should come up. Select download from the top right hand menu and save the .pdb file to the md-prac/setup directory.

If you look at this file (using the graphical editors vi or nedit for example):-

```
% cd setup
```

If you are not familiar with VMD, now is a good time to fire it up and get used to performing some routine manipulations. You should immediately see that it has two chains:- (the following file name might be uppercase depending on how you downloaded it).

```
%vmd 1hsg.pdb
```

If the internet is down or you cannot find 1hsg then you can use a pre-downloded copy of that which is to be found in the `data` directory. In fact, most of the files you need can be found in this directory so if you get confused/stuck/lost etc then you can always look here to check you did things correctly.

You should experiment with the menu system and try various representations of the protein such as trace, cartoon and ribbons for example.

Q. Can you find the indinavir drug? Give the protein the trace representation and then make the polar residues in vdw format as an additional representation. Repeat with the hydrophobic residues. What do you notice?

Exit vmd, either by clicking on File-->Quit or typing quit in the terminal box. Now we need to prepare our protein for simulation. First of all we will extract only the coordinates from the pdb file. To do this enter the following command:-

```
% grep ATOM 1hsg.pdb > protein.pdb
```

First of all we need to make sure that all the hydrogens are added to our protein. This process will also generate the parameter/topology file we need.

```
% gmx pdb2gmx -f protein.pdb -ignh -o protein_gmx.pdb
```

The program should run and present a list of force-fields from which to select. Select the GROMOS96 43A1 force field which should be option 9 in the list followed by 1 to select SPC water. If all goes well this should generate several files: 1. topol.top 2. topol_Protein_chain_A.itp 3. topol_Protein_chain_B.itp 3. protein_gmx.pdb and 4. posre_Protein_chain_A.itp.

5. posre_Protein_chain_B.itp Note that the protein has a net charge of +4e. You should see a line that says "`Total charge in system 4.000 e`".

Now we have hydrogens we can also ask questions about hydrogen bonding. Use `vmd` again to try and answer the following question.

Q. How many hydrogen bonds are there between aspartate (`resname ASP`) and arginines (`resname ARG`) residues?

Before we can add water we need to define a box in which to put the protein and the water:-

```
% gmx editconf -f protein_gmx.pdb -box 7 7 7 -c -o boxed.pdb
```

This puts the protein in the centre of the box that is 7 nm x 7 nm x 7 nm and creates the resulting file `boxed.pdb`. It is always a good idea to check that this operation produces the desired result.

```
% vmd boxed.pdb -e ../data/pbcbox.tcl
```

Next we need to add water to the system. We will also add ions to the system - enough to neutralize the system and to a concentration that is representative of the cell. Now we can add the water by repeatedly overlaying a small (216 molecules) box of water into the system.

```
% gmx solvate -cp boxed.pdb -cs -o solvated.pdb -p topol.top
```

You may have noticed in some of the output generated that the total system charge is +4. In order for us to use an Ewald method to calculate the electrostatic interactions we need to have a neutral system overall. Therefore we will add counterions (chloride ions, in this case) using the option `-neutral` and enough ions to make the solution up to 150 mM (`-conc 0.15`). This is done by replacing random water molecules (`SOL`) with `NA+` or `CL-` ions.

```
% gmx grompp -c solvated.pdb -p topol.top -f ../data/genion.mdp -o genion.tpr
% gmx genion -s genion.tpr -conc 0.15 -neutral -pname NA -nname CL -o system.gro -p topol.top
```

When prompted, enter the group that corresponds to `SOL` (should be 13 or thereabouts).

Visualise the system using `vmd` and convince yourself that the system has been solvated and ions have been placed at random locations

```
% vmd system.gro
```

Before we can run the actual dynamics, we need to first minimize the system. Ideally you would minimize down until the forces were below a certain level (tolerance), but we will just give a quick burst of 200 steps here.

```
% cd ../run
% gmx grompp -c ../setup/system.gro -p ../setup/topol.top -f ../data/em.mdp -o em.tpr
% gmx mdrun -deffnm em -v
```

We can examine the minimization in terms of the potential energy versus the number of minimization steps:

```
% gmx energy -s em.tpr -f em.edr -o em_potential_energy.xvg
```

type in 10 when prompted (should correspond to potential energy from the list of options presented), then a zero to finish. Next we can draw the graph/plot using xmgrace:-

```
% xmgrace em_potential_energy.xvg
```

At this stage, we would normally run a short simulation where the protein atoms are restrained while the water molecules and ions are allowed to freely move around and equilibrate around the protein. For this tutorial, we will skip this bit due to limited time. Now finally let us perform some molecular dynamics:-

```
% gmx grompp -c em.gro -p ../setup/topol.top -f ../data/md.mdp -maxwarn 1 -o md.tpr
% gmx mdrun -deffnm md &
```

At the moment it is set up to run for 1000 ps. This will take several minutes to complete depending on the speed of your system - time for lunch! You don't have to wait for it to finish completely though, although now might be a good time for a break to allow at least some data to appear. The analysis can be done on the files as they appear or you can always use the "one I made earlier" in the directory `prerun/run` (This is 1000 ps simulation of the same system).

5. Analysis of the Trajectory

The simplest and easiest type of analysis you should always do is to look at it with your eyes! Your eye will tell if you something strange is happening immediately. A numerical analysis may not. Let us look at the simulations in VMD.

```
% vmd
```

When it has finished placing all the windows on the screen. Click on "File" in the VMD main menu window and select "new molecule". The Molecule File Browser window should appear. Click on "browse" then select setup and finally select em.gro

(ie the file you made that has protein system energy minimized). Click "OK" and then click "Load". It should load up the starting coordinates into the main window. Then click browse in Molecule File Browser window. Select md.xtc. Select "OK" and then hit "load". Alternatively locate the longer pre-run xtc file made in the backup directory and load that. The trajectory should start loading into the main VMD window. Although things will be moving, you can see that its quite difficult to visualize the individual components. That is one of the problems with simulating such large a complicated systems. VMD makes it quite easy to look at individual components of a system. For example, let us consider the protein only. On the main menu, left click on Graphics and select "Representations". A new menu will appear (Graphical Representations). In the box entitled "Selected Atoms" type protein and hit enter. Only those atoms that form part of the protein are now selected. Various other selections and drawing methods will help to visualize different aspects of the simulation. More help on the selections is available here.

Now that we are sure the simulation is not doing anything ridiculous, we can start to ask questions about the simulation. The first thing to establish is whether the simulation has equilibrated to some state. So what are some measures of the system being equilibrated? and what can we use to test the reliability of the simulation?

1. Analysis of System Properties

View the simulation

Observe the protein, water and ions move around using `vmd`.

```
% vmd -f ../prerun/run/em.gro ../prerun/run/md.xtc &
```

Q. Are there any regions that move a lot?

Temperature fluctuation

There are various so-called ensembles that are used for protein simulations - probably the most common is a system where the number of particles, the pressure and the temperature are held constant (NPT). This is usually achieved by means of a heat-bath. Nevertheless, it is usually a good idea to check these as a function of time through the trajectory just to make sure nothing unexpected happened. First let us check the temperature of our simulation.

```
% cd ../analysis
% gmx energy -f ../prerun/run/md.edr -s ../prerun/run/md.tpr -o 1hsg_temperature.xvg
```

The program will then present you with a large table of all the values recorded in the energy (.edr) file. We want to examine temperature so type 13, press enter and then 0 and press enter again. The program will then analyse the temperature and present some statistics of the analysis at the end. We can look at the fluctuations in more detail by using `xmgrace` (or xmgr).

```
% xmgrace 1hsg_temperature.xvg
```

will display the temperature fluctuations.

Energy of the system

Another set of properties that is quite useful to examine is the various energetic contributions to the energy. The total energy should be constant. but the various contributions can change and this can sometimes indicate something interesting or strange happening in your simulation. Let us look at some energetic properties of the simulation.

```
%  gmx energy -s ../prerun/run/md.tpr -f ../prerun/run/md.edr -o 1hsg_energies.xvg
```

We shall select short-range lennard-jones (6), short range coulombic (7) and the potential energy (9). End your selection with a zero. (Note: If you are unfamiliar with xmgrace you may find it easier to make these as three separate files instead of saving everything in `1hsg_energies.xvg`.)

Look at the resulting graph with xmgrace again.

```
% xmgrace -nxy 1hsg_energies.xvg
```

Q. Is the total energy stable in this simulation? Does it make a difference how long the simulation is? (you could compare what you have generated so far with the one I made earlier)

Q. What is the dominant contribution to the potential energy?

2. Analysis of Protein

Root mean square deviation (RMSD) of 1HSG

Now that we are happy that the system is OK, let us now consider the root mean square deviation (RMSD). We shall use the tool gmx rms

```
% gmx rms -s ../prerun/run/em.gro -f ../prerun/run/md.xtc  -o 1hsg_rms.xvg
```

We will use the Cα coordinates to do the fitting to, so select group 3 when prompted. Then enter 3, to look at the RMSD of all Cα atoms. Have a look at the resulting graphs with xmgrace.

Q. What does this tell you about the stability of the protein? Is it in a state of equilibrium and if so why and at what time?

Q. Can you think of a situation where this approach might not be a very good indication of stability?

Root mean square fluctuation (RMSF)

A similar property that is particularly useful is the root mean square fluctuation (RMSF). This can be related back to the b-factor value which is given in X-ray crystallography and hence provides an easy way to compare your simulation back to experiment. But first, we need to renumber the residues in the .gro file, so that residues of the two chains will not have the same numbers:

```
%  gmx editconf -f ../prerun/run/em.gro -o em_renumbered.gro -resnr 1
```

Now, let us first obtain the rmsf for 1hsg according to the 1000 ps of simulation you ran.

```
% gmx rmsf -s em_renumbered.gro -f ../prerun/run/md.xtc -b 200 -oq 1hsg_bfac.pdb -o 1hsg_rmsf.xvg
```

select 3 when prompted to obtain b-factors for Cα atoms. NOTE: This will omit the first 200 ps (where the simulation is less stable). For this part you'll probably want to use the pre-run data. This will produce a pdb file (1hsg_bfac.pdb) that has b-factors in the 10th column. We can extract this data ready for plotting in xmgrace as followings:-

```
% grep CA 1hsg_bfac.pdb | awk '{print $5, $10}' > 1hsg_bfac_ca.dat
```

Look at this data in xmgrace.

Q. Can you identify structural regions alone from this plot and does that fit in with the structure??

We can compare this simulation data directly with experimental b-factors from the crystal structure of protease. Read in the following ascii data file into xmgrace:-

```
../data/1hsg_bfac_ca_expt.dat
```

Q. Does the simulation agree with experiment?

Q. Where does it not agree and can you suggest reasons why it does not?

Q. Residues 43-58 form part of the flexible flap that covers the binding site. How does this region behave in the simulation?

Hydrogen bond (HB) formation

We can also use the simulation to monitor the formation of hydrogen bonds between the ASP and ARG residues that you visually observed at the beginning of the tutorial with VMD. First edit the index.ndx file to include one group that contains all the ASP residues and another that contains the ARG residues:-

```
% gmx make_ndx -f em_renumbered.gro -o index.ndx
```

When prompted, type "r ASP", press Enter and then type "r ARG" and press enter again. Press Enter once more to make sure that the new groups have appeared and type "q" to quit. For the hydrogen bond calculation type:

```
% gmx hbond -f ../prerun/run/md.xtc -s ../prerun/run/md.tpr -n index.ndx -num ASP_ARG_hbnum.xvg
```

Select the number of the group that corresponds to ASP first and ARG next (it should be 19 and 20, respectively). Plot the number of hydrogen bonds with xmgrace and observe how much it fluctuates with time. Close xmgrace and calculate the average number of hydrogen bonds between aspartates and arginines.

```
gmx analyze -f ASP_ARG_hbnum.xvg -av aver_ASP_ARG_hbnum.xvg
```

The number printed in the first line shows the average number of HBs, followed by the standard deviation.

Q. How does it compare to the hydrogen bods you visualised with VMD?

Some things to think about:-

Q. How much variation is there in the number of hydrogen bonds?

Q. Do any break and not reform? Do you get salt bridges formed that were not present in the crystal? Why is this? Using VMD, can you observe the HB formation and breakage throughout the simulation?

In the last part of the practical you should explore other types of analysis. It is worth looking at some of the other standard gromacs analysis tools. If you type "gmx help commands", it should list all the gromacs analysis tools. Typing any of these program names followed by the -h flag will tell you what that program does and that should help you decide whether it is the tool for the job.

As a follow up, how might you calculate/show how water molecules behave in the vicinity of the protein? Are they always tightly bound? etc.. etc..

That concludes this practical. I welcome all comments (including negative ones) and/or suggestions - please feel free to email philip.biggin@bioch.ox.ac.uk.

6. Further Reading.

The texts recommended here are the same as those mentioned in the lecture:-

- "Molecular Modelling. Principles and Applications". Andrew Leach. Publisher: Prentice Hall. ISBN: 0582382106. *This book has rapidly become the defacto introductory text for all aspects of simulation.*
- "Computer simulation of liquids". Allen, Michael P., and Dominic J. Tildesley. Oxford university press, 2017.
- "Molecular Dynamics Simulation: Elementary Methods". J.M. Haile. Publisher: Wiley. ISBN: 047118439X. *This text provides a more focus but slightly more old-fashioned view of simulation. It has some nice simple examples of how to code (in fortran) some of the algorithms though*

7. Appendix I - Basic Unix/Linux Commands.

ls -lrt	provides a "long" list of all files in the current directory in reverse order of time.
cd dir	change directory to the directory 'dir'
pwd	print the current working directory on the screen
rm file	delete (remove) 'file'
mv file newfile	rename file to newfile
cat file	print the contents of file to the screen
more file	print the contents of file to the screen but with more navigation possible.

8. Appendix II - Trajectory Manipulations.

If your computer is less powerful than a pentium 4 with 1Gb of memory, then you will certainly have problems visualizing the trajectories in VMD. This is a problem for most membrane simulations, where the capacity to simulate long timescales has exceeded the visualization capability. There are however a number of ways to circumvent this and still be able to follow the practical :-

- Break the whole trajectory down into a number of small components.
- Make a complete trajectory but with fewer frames
- Make a trajectory of the protein (or bits you are interested in) only (ie reduce the number of atoms per frame).

All of these manipulations can be accomplished very easily with the trjconv program. For example

```
gmx trjconv -f file.xtc -s file.tpr -b 500 -o file_last_500ps_only.xtc
```

will write out the last five hundred picoseconds from this 1ns trajectory.

To write out less frequently, use the skip option. For example to write out the 1ns trajectory but every 20ps instead of every 10:

```
gmx trjconv -f file.xtc -s file.tpr -o file_less_sample.xtc -skip 2
```

This option is probably the best one as it gives you visualization across the whole trajectory (although obviously you should be aware that you might miss some very short timescale motions).

The program will prompt you selections so writing out just protein for example is very straightforward.

If your selection is not there, then you should include an index file.

9. Appendix IIII - Useful Packages.

- VMD The best free tool for looking at dynamics
- Pymol Another opensource project. Has good graphics
- MolMol Designed from an NMR point of view, and has some nice nove l ways of displaying information (eg. sausage plots)
- swiss-pdb viewer Another free viewer
- gromacs The only GPL molecular simulation package.
- xmgrace The free data plotting program.

10. Acknowledgements.

I thank Syma Khalid for some of the initial files used in this practical and Ranjit Vijayan and Ben Hall for additional tweaks over the years

I also thank the Wellcome Trust and the Oxford Supercomputing Centre.