A Molecular Dynamics Simulation study of lipid bilayer self assembly and helix dimerisation

Syma Khalid, Astrid Brandner and Dheeraj Prakaash.

General introduction

In this practical class you will learn how to run molecular dynamics (MD) simulations, to explore interactions between water, lipid molecules and transmembrane helical proteins. We will be doing coarse-grained molecular dynamics simulations, but the principles we cover will also apply to the more fine-grained atomistic simulations. This is a two-part practical class.

PART 1

The first part of the MD class is focused upon the self-assembly of phospholipids into bilayers.

Phospholipids

Lipids are components of membranes that surround and protect cells (as well as creating sub-cellular compartments in eukaryotic cell). Lipids are composed of polar headgroups and hydrophobic tails and thus in water, they can self-assemble into bilayers. In the first part of this class you will simulate molecules of the phospholipid, dilauroylglycerophosphocholine (shortened to DLPC) randomly placed into a box of water to simulate the process of bilayer self-assembly (Figure 1).

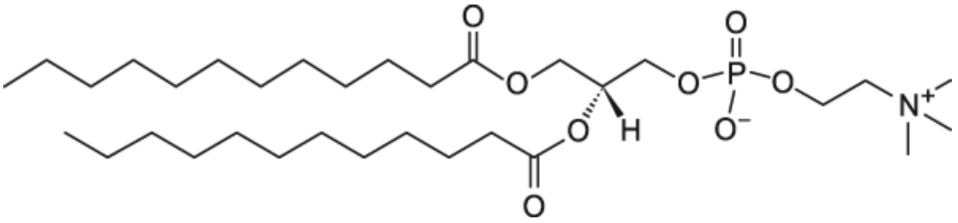

Figure 1. The structure of DLPC.

Computational techniques

You will be doing molecular dynamics simulations and energy minimisation in this class; these are classical simulation techniques (i.e. no electrons). In classical molecular simulation methods, the potential energy is calculated from a 'force-field', which is a set of parameters containing atom types (with mass & charge), bond lengths, angles, dihedral angles, charges and other terms. The general form of a force field is given by Equation 1:

$$U_{pot} = \Sigma_{bonded} + \Sigma_{non\text{-}bonded} \qquad \text{Equation 1}$$

where U_{pot} is the total potential energy of the system, the bonded terms are the bond lengths, bond angles and dihedral angles, and the non-bonded terms are the permanent electrostatics, and repulsion and dispersion (polarisation is usually neglected).

Molecular Dynamics

Molecular Dynamics is a simulation technique in which classical force-fields are used to calculate thermodynamic properties of a 'system' composed of numerous particles. These particles may represent atoms or groups of atoms. The particles move according to Newtonian mechanics. In other words, equation 1 is integrated many times to determine the motion of the atoms using Equation 2:

$$F_i = m\, a_i \qquad \text{Equation 2}$$

where F is the force, M is the mass and a is the acceleration. The link between the forcefield and Newton's second law is that the force can be expressed as the negative gradient of the potential energy (Equation 3):

$$F = -dU_{pot}/dr$$

Equation 3

This provides us with the crucial link between the force field (which gives us the potential energy) and classical (Newtonian) mechanics from which we can establish the acceleration of the atoms. The full details of the mathematics are beyond the remit of this practical. Don't panic, the software will do the integration for you! However, variables such as the temperature, pressure, length of simulation are set by the user (in this case, you!).

You can learn more about molecular dynamics here:
https://doi.org/10.1016/j.ymeth.2020.02.007

Coarse-grained models
You will use coarse-grained models for this practical (inclusion of atomistic detail would be far too computationally demanding for the purposes of this practical, allowing you to only simulate a few steps). Coarse-grained models reduce the dimensionality of the systems they represent by combining groups of atoms together into larger particles that can be thought of as pseudo atoms. Care is taken to ensure that the characteristic chemistry of the system is retained and reproduced.

Here we will use the MARTINI force-field, this is the most popular coarse-grained force-field for biological molecules (Figure 2). You can read more about MARTINI here: http://cgmartini.nl/, and about coarse-grained force-fields more generally here: https://doi.org/10.1007/978-1-0716-1394-8_14

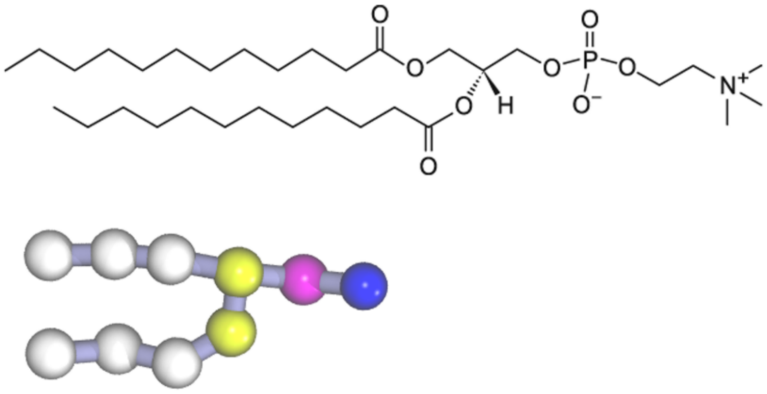

Figure 2. Atomistic structure of DLPC (top). MARTINI, coarse-grained structure of DLPC model (bottom).

Q. Which coarse-grained bead corresponds to which groups of atoms? I.e. blue is equivalent to _____ and pink is _____. If you are unsure, have a look at the MARTINI website or talk to a demonstrator.

Energy minimisation
This is a method used to remove any steric clashes between molecules before the simulation is performed. The potential energy of the molecular system is minimised in this process. Any particles that may accidentally have gotten too close to each other during the setup will be moved to more energetically favourable positions. It is essential to perform energy minimisation before attempting to run an MD simulation.

Software

- VMD
  This is a visualisation and analysis tool that you will all have encountered in your year 2 computational chemistry practical. Please refer to the link below for further information including online tutorials:
  http://www.ks.uiuc.edu/Research/vmd/

- Gromacs
  This is a free molecular simulation software package. We will be using it to set up and run our simulations. Further information and an online help forum can be accessed via:
  http://www.gromacs.org/

We will be performing two types of calculations; energy minimisation and molecular dynamics (explained above). The file types summarised in Table 1 will be required (you will be expected to edit some of these files). A number of preparatory steps are required to create the input file for a molecular dynamics simulation within Gromacs. The grompp routine processes the various files to create a binary file which is then used by the mdrun routine to perform the molecular dynamics simulation.

File types

File extension	Contents	Input/Output
pdb	Coordinates of all the particles and information about box size and shape	Input for grompp
gro	Coordinates of all the particles and information about box size and shape	Input for grompp. Output from mdrun
mdp	Simulation parameters e.g. length of simulation, temperature, pressure and others. Also type of simulation (MD or minimisation)	Input for grompp
itp	This is the force-field file. It contains the atom types, masses, charges, details of bonds, angles etc	Input file, not directly called from the 'grompp' command but referenced within the top file.
top	Links to the relevant itp files and information about how many molecules of each types are in the system (they must be listed in the same order in which they appear in the pdb/gro file)	Input for grompp
tpr	This is a binary file containing a summary of the information from all the above files and is required to run the simulation	Output from grompp. Input for mdrun

Enter the directory, self-assembly:
cd self-assembly <enter>

You have been provided with a file called: emptybox.pdb
You will now add DLPC molecules to this box.
Visualise the file DLPC.pdb using VMD.
vmd DLPC.pdb <enter>

Go to **Graphics > Representations** and from the **Drawing Method** menu select: VDW. You should now see the DLPC molecule (you can rotate the molecule with the mouse). Now take a look at the file called 'commands'. You can do this with:
cat commands <enter>

Or alternatively open the file with a text editor (such as 'gedit' on Ubuntu operating systems).
gedit commands & <enter>

Type the first command in that file into the terminal:
gmx insert-molecules -f emptybox.pdb -ci DLPC.pdb -nmol 224 -o lipids.pdb <enter>

So, using the '*gmx insert-molecules'* command, you have instructed the Gromacs to add DLPC (x 224) into the file called 'emptybox.pdb'. The output file called 'lipids.pdb' contains the box now with DLPC molecules.

Now use VMD to visualise your new file called lipids.pdb. Once again, use the 'VDW' Drawing Method within VMD. (Hint, from the Display menu, you may wish to switch to Orthographic view). You should see something that resembles Figure 3.
To close VMD, close the VMD main window.

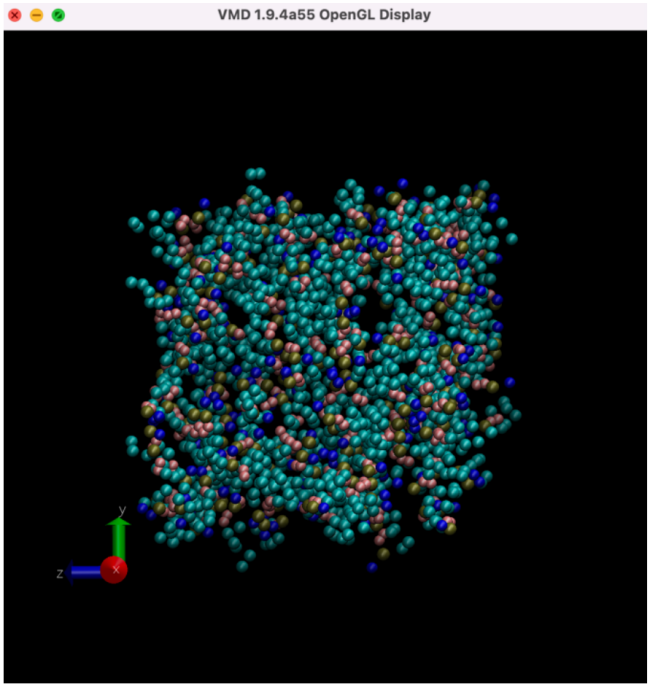

Figure 3. VDW representation of the file 'lipids.pdb' created by adding 224 x DLPC.pdb into emptybox.pdb

Now let's perform an energy minimisation calculation to remove any steric clashes between the DLPC molecules (to achieve a local minimum point).

Now let's perform an energy minimisation calculation to remove any steric clashes between the DLPC molecules (to achieve a local minimum point).

First, edit the system.top file.
You will see:

```
[ molecules ]
; name number
DLPC x
;W  x
```

For DLPC, replace the 'x' with the number of DLPC molecules you have added. 'W' is the name for water molecules. The ';' in front of the line tells the code to ignore this line because we do not currently have any water molecules in our box, we only have DLPC.

Now enter the second and third commands from the commands file:
gmx grompp -c lipids.pdb -p system.top -f minimisation.mdp -r lipids.pdb -o min.tpr <enter>

Here you have told the routine ‘grompp’ to create an input for MD called ‘min.tpr’ by using coordinates from ‘lipids.pdb’ and simulation parameters from ‘minimisation.mdp’ and system information from ‘system.top’.

Check that the min.tpr has been created. To do this, you can list all the files in the current directory by entering:
ls <enter>

Then run the energy minimisation using the ‘mdrun’ routine as follows:
gmx mdrun -s min.tpr -v -deffnm minimised <enter>

The -v (verbose) option tells the code to provide information about the minimisation process as it is running, thus you will see the potential energy at each step. Once the minimisation has finished, you will see a file called ‘minimised.gro’ produced. This contains the coordinates of the minimised system.

Now visualise the resulting file (solvated.gro) in VMD. Create separate representations for the water and DLPC and colour each differently so you can clearly see both types of molecules as shown in Figure 4.

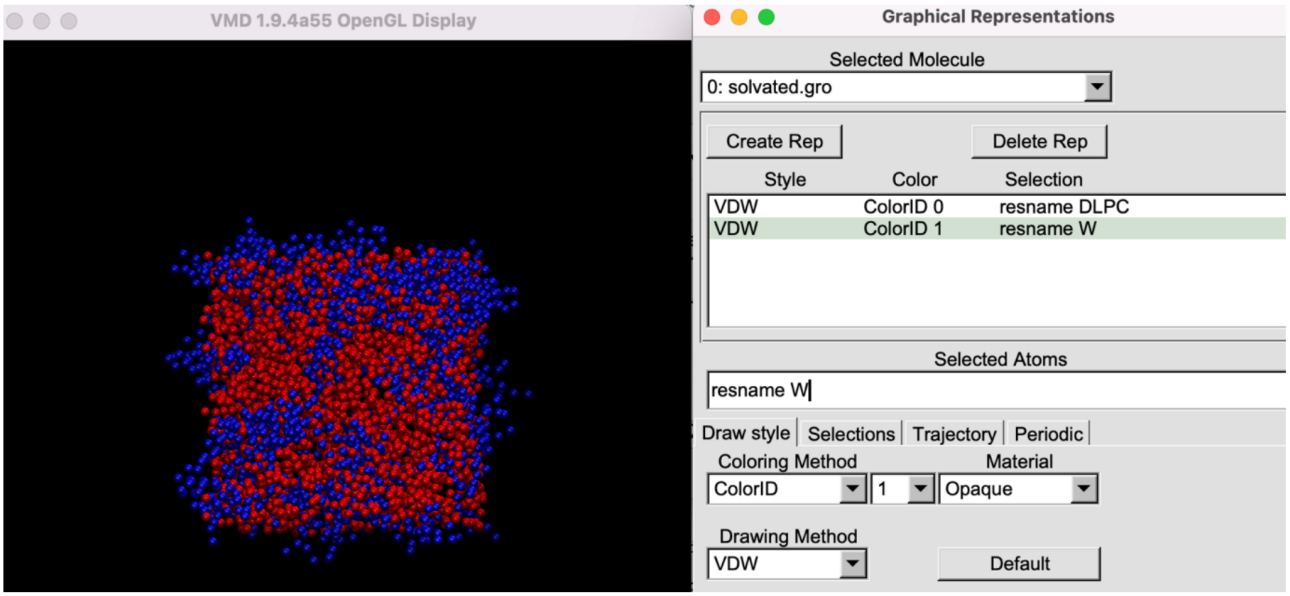

Figure 4. Water (resname W) and DLPC (resname DLPC) shown using Drawing Method:VDW and Colouring Method: ColorID.

Now edit the system.top file to remove the ‘;’ from the line listing the number of water (W) molecules and replace the ‘x’ with the number of water molecules you have added. Now back to your commands file.

Energy Minimisation
To create the ‘solvated_min.tpr’ input file for energy minimisation:
gmx grompp -c solvated.gro -f minimisation.mdp -r solvated.gro -p system.top -o solvated_min.tpr <enter>

To run the minimisation:
gmx mdrun -s solvated_min.tpr -v -deffnm minimised_solvated <enter>

Next, we will run a three-step equilibration procedure. Take a look at file ‘equil1.mdp’.

You will see that the lipid headgroups have positional restraints applied to them as follows:
define = -DBILAYER_LIPIDHEAD_FC=200 (units are kJ/(mol × nm^2)

You will see that the integration timestep (dt) is set to:
dt = 0.002 (units are ps).

Q. How many steps is the simulation going to perform? What will be the total simulation time? What is the target temperature for the simulation?

Equilibration stage 1
Now back to your commands file. Lets run the first step of the equilibration using our minimised coordinates:
gmx grompp -c minimised_solvated.gro -r minimised_solvated.gro -p system.top -f equil1.mdp -o equil1.tpr <enter>
gmx mdrun -s equil1.tpr -v -deffnm equil1 & <enter>

Now check the equil2.mdp and equil3.mdp files. What are the force constants and the integration timesteps for each of these files? Now run each simulation (obviously waiting for each stage to complete before starting the next).

Equilibration stage 2
gmx grompp -c equil1.gro -p system.top -r equil1.gro -f equil2.mdp -o equil2.tpr <enter>
gmx mdrun -s equil2.tpr -v -deffnm equil2 & <enter>

Equilibration stage 3
gmx grompp -c equil2.gro -p system.top -r equil2.gro -f equil3.mdp -o equil3.tpr <enter>
gmx mdrun -s equil3.tpr -v -deffnm equil3 & <enter>

We are limited in terms of time in this practical, but given more time, we should check that the equilibration procedure is sufficient. We can do this by monitoring various systems properties, e.g. system volume, temperature, pressure etc. You can have a quick go by comparing the average system volume from each of your equilibration stages:

gmx energy -f equil1.edr <enter>
15 <enter>
0 <enter>
gmx energy -f equil2.edr <enter>
15 <enter>
0 <enter>
gmx energy -f equil3.edr <enter>
15 <enter>
0 <enter>

Q. Is the estimated error increasing or decreasing as you progress with the equilibration procedure?

Production simulation
We will now run the production molecular dynamics simulation (this will take 20 mins or so to run; Gromacs will show you an estimate of when the simulation will finish once it starts).

gmx grompp -c equil3.gro -p system.top -f production.mdp -o md.tpr <enter>
gmx mdrun -s md.tpr -v -deffnm md & <enter>

Visualise the resulting simulation in VMD. Load in the equil3.gro file and the new trajectory you have just calculated (md.xtc).
vmd equil3.gro md.xtc <enter>

To make it easier to see what is happening, create a representation that just shows the phosphate groups of the phospholipids:
Graphics > Representations > Selected Atoms: **name PO4**
Once again, use Drawing Method: VDW

It is highly likely that you will see the lipids forming a structure that is not quite a complete bilayer yet. We will now run the production simulation again at a higher temperature. Edit the 'production.mdp' file to change the temperature (value of ref_t) to 330 K. Create a new input file using the same equilibration file, but the new mdp with higher temp:
gmx grompp -c equil3.gro -p system.top -f production.mdp -o md_330.tpr <enter>

and run the simulation:
gmx mdrun -s md.tpr -v -deffnm md_330 & <enter>

Upon visualisation in VMD, you should now see a fully formed DLPC bilayer. The extra kinetic energy provided by heating up the simulation to 330 K was enough to allow the DLPC molecules to form a bilayer in the same number of steps in which they were unable to do so at 310 K. Your bilayer should look something like the one shown in Figure 5 (looking at just the phosphate groups of the lipids).

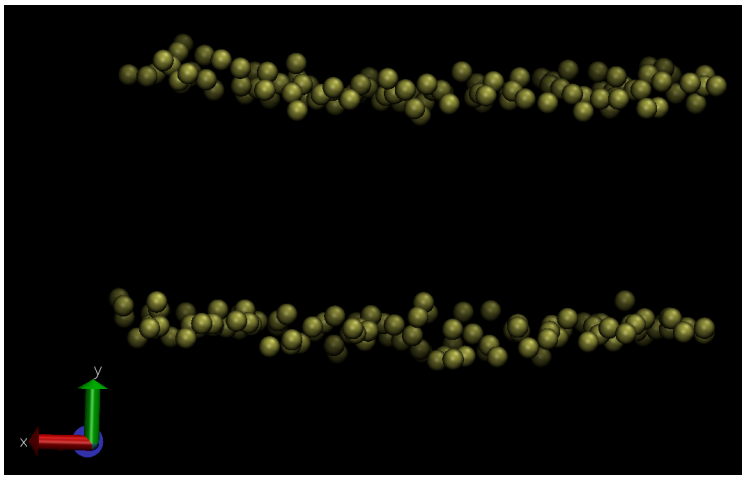

Figure 5. Phosphate (name PO4) particles of DLPC showing formation of a bilayer at 330 K

You can self-assemble bilayers of other lipids in the same manner as long as you have the coordinate files (pdb or gro) and itp files (these are available from the MARTINI website for most common lipids). You can even use this approach to self-assemble a bilayer around a membrane protein.

CONGRATULATIONS, YOU HAVE NOW COMPLETED PART 1 OF THE MD PRACTICAL.

PART 2

We will now explore the dynamics of helix dimers in two different environments. We will focus on Glycophorin A.

Glycophorin A (GpA) is a relatively simple erythrocyte membrane protein. GpA is the cellular receptor for the malarial parasite *Plasmodium falciparum*, via binding of parasite ligands to sialic acid on GpA. It is composed of 131 amino acid residues found in three domains: a short hydrophilic cytosolic domain, a hydrophobic transmembrane (TM) domain of ~25 residues, and a larger extracellular domain. GpA forms a symmetrical homodimer via specific interactions between the TM helices. The structure of the TM domain of the dimer has been determined by solution NMR in dodecyl phosphocholine (DPC) micelles and by solid-state NMR in dimyristoyl phosphatidylcholine (DMPC) and palmitoyl oleoyl phosphatidylcholine (POPC) bilayers. The helices pack together in a right-handed manner.

Here you will run a simulation of two copies of the GpA TM domain in a DPPC phospholipid bilayer to explore the process of dimerisation.

Enter the directory 'GpA_dimerisation'. Visualise the file 'initial_system.pdb' in VMD. We recommend viewing the backbone of the two helices and the phosphate groups of the lipids using:
Graphics > Representations >
Selected Atoms: name BB PO4
You should see something that looks like Figure 6.

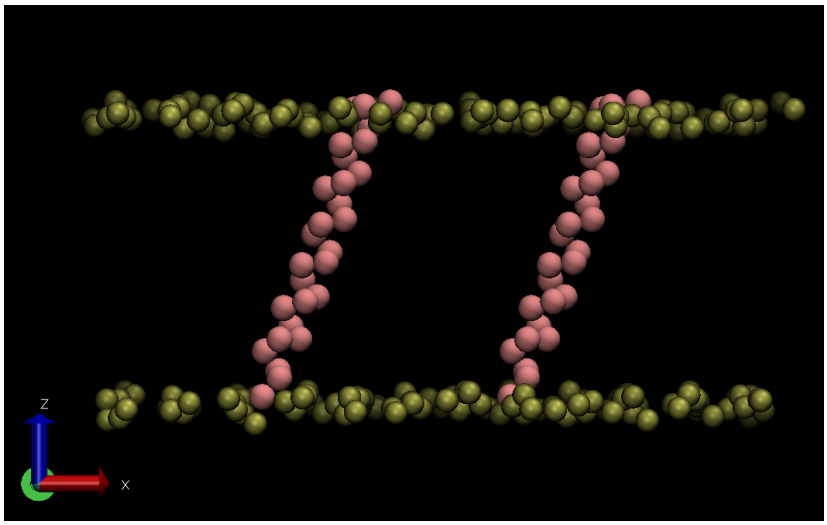

Figure 6. The file 'initial_system.pdb' viewed in VMD using selected atoms 'name BB PO4' to visualise the protein backbone and lipid phosphate particles.

Q. What is the distance between the central Glycine residues of the two helices?

To measure this distance, create a new representation using selected atoms 'resname GLY and name BB'.

Now apply a different colour to this than the rest of the protein to make visualisation easier (Coloring Method > ColorID > Yellow). You should see something that looks like Figure 7. To measure the distance between two glycine residues, on the VMD Main window, go to Mouse > Label > Bonds. Now, click on a glycine residue belonging to each of the two helices as shown in Figure 7.

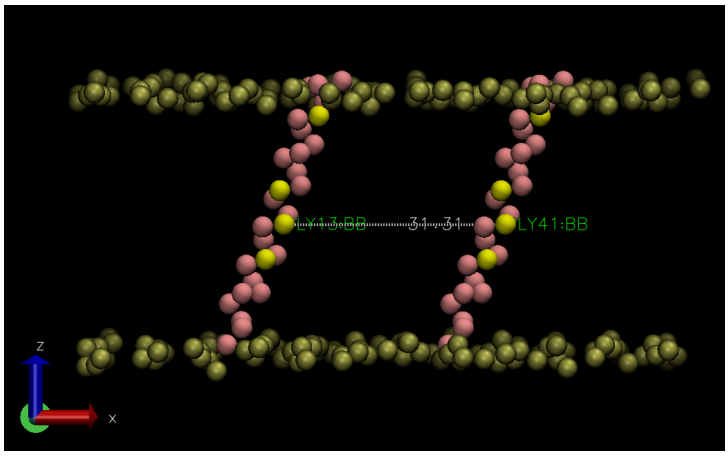

Figure 7. The file 'initial_system.pdb' with an extra representation showing glycine residues in yellow

Note down the distance between them. The distance will be presented in the graphical window and also in the terminal.

Energy minimisation, equilibration, and production simulation.
Use the 'commands' file to run the energy minimisation and equilibration procedure. You will need to add in the integrator to the 'minimisation.mdp' file (use your previous simulations to work this out), and the temperature (323 K) into the 'production.mdp' file.

For each equilibration stage, you need to include (1) '-n index.ndx' and (2) '-r x.gro' at the end of each grompp command, where 'x' is the name of the gro file produced by the previous step. The index file is used to define groups of atoms corresponding to 'tc-grps in the mdp files, where 'tc' stands for temperature coupling. Temperature coupling is briefly explained in the following paragraph. The gro file from the previous step is provided while using position restraints. In this case, we are position-restraining the lipid headgroups during the equilibration stage and slowly releasing the restraints (with decreasing forces) in each step of equilibration.

Visualise the system after the final equilibration stage (equil3.gro). Assuming everything looks ok, proceed to the production stage. Set the temperature to 323 K in the 'production.mdp' file. You will note that three values are required for the temperature. The reason for this is that the temperature is coupled to separate baths for (1) protein, (2) membrane, and (3) water+ions. This is done to prevent some parts of the system being heated up more than others, which can happen if the molecular types present in a system are very different from each other.

Now proceed to running grompp and mdrun. This mdrun process may take about 20 mins or so. Once the simulation is complete, visualise the trajectory:
vmd equil3.gro md.xtc <enter>

Q. Have the helices moved closer together? Have they formed a dimer?
Measure the distance between the glycine residues as done previously. This time, we can monitor this distance throughout the trajectory, rather than just from one static structure. This can be done via VMD Main > Graphics > Labels > Dropdown menu: Bonds > select a pair of residues > Graph panel. Then you can save the data (by clicking on Save) and also visualise a preview (by clicking on Graph).

Q. Looking at the time distance vs data you have just saved, are the helices moving closer to each other, further away or have they stopped moving by the end of the simulation? By 'move' we mean translation, because of course these molecules are not static, they are always jiggling about.

Q. What is the closest distance between the backbones of the two glycine residues?

Now take a look at the structure of the dimer solved by NMR in a DLPC detergent micelle. Download this from the protein databank: https://www.rcsb.org (PDB code: 2KPE).

Q. Does your dimer look the same as the structure? Consider the crossing angle of the two helices or the distance between the tips (at both ends of the helices).

It likely does not look the same, this is because while the simulation is long enough for the two TM helices to approach each other and dimerise, it requires a bit longer to form the specific interactions that will lead to the crossing angle seen in the structure. Because we are nice, we have supplied you

with a simulation of the NMR structure in a DLPC bilayer. This is available in the directory, 'GpA_pre_bilayer'. Visualise the trajectory in VMD.

Q. Is the dimeric arrangement of the helices retained throughout the simulation?

Let's measure some distances to characterise the inter-helical interactions. Save the data corresponding to the distances between the residues at both termini of the helix (GLU-GLU and LEU-LEU) and also the central GLY-GLY. Once again, use VMD Main > Mouse > Label > Dropdown menu: Bonds > select a pair of residues > Graph panel. Save each dataset to new files called GLU-GLU.dat, LEU-LEU.dat, and GLY-GLY.dat respectively. Plot them using xmgrace:
xmgrace GLU-GLU.dat LEU-LEU.dat GLY-GLY.dat <enter>

Note that, the first, second, and third datasets are shown as black, red, and green lines by default in xmgrace.

Q. What do you notice about the distances? Are the inter-helix interactions longer-lived in the centre of the helix or at the termini? Is this what you would have expected and why?

Q. Given one terminus of both helices have a GLU residue, we have two acidic residues in close proximity. How do you think such an arrangement could be stabilised?

Use your trajectory to justify your answer. (Hint: look at your system.top file to see what other molecules in addition to helices, lipids and water are in your simulation system and where they are located during the simulation.
You may wish to use the 'within' selection criterion in the 'Selected Atoms' menu to help to refine your visualisation (use the VMD manual & Google to work out how to use this before asking a demonstrator).

The final part of this practical is to analyse a trajectory of the GpA dimer in a DPC detergent micelle.

Enter the directory: 'GpA_pre_micelle'. Visualise the structure file: 'initial_system.pdb' using VMD. To get a good view of the micelle and the dimer make two representations in which Selected Atoms is (1) name BB and (2) resname DPC.

Q. Would you expect the environment of the detergent micelle to allow greater or lesser helical dynamics than the phospholipid bilayer and why?

Use the helical measurements as previously (tip to tip and central GLY to GLY distances) to characterise the dynamics of your dimer in the micelle.

Q. Was your prediction correct?

CONGRATULATIONS, YOU HAVE NOW COMPLETED PART 2 OF THE MD PRACTICAL.

To set up other coarse-grained or atomistic systems, (assuming you have an atomistic protein or peptide structure) you can use the CHARMM-GUI web-server: https://www.charmm-gui.org

To construct a perfectly helical atomistic peptide (with a sequence of your choice) we have provided you with a jupyter notebook (which assumes you have Modeller installed) called 'helix_modeller.ipynb'.