MRS Spring 2023

Tutorial SB06 Molecular Dynamics Simulations for Peptide Based Materials

Dhwanit R. Dave

ddave@gradcenter.cuny.edu

April 2023

This tutorial is divided into two major parts to cover both atomistic as well as coarse-grained molecular dynamics simulations. It is intended to help familiarize users with the basic aspects of multiscale MD, especially in the context of peptide self-assembly. In this tutorial, you will learn how to set up, run, analyze, and visualize molecular dynamics trajectories using two well understood and basic examples of the self-assembly amino acids and dipeptides such as Phenylalanine (F) and Diphenylalanine (FF).

Literature relevant to this tutorial can be found in these publications-

1. Uyaver, S.; Hernandez, H. W.; Habiboglu, M. G., Self-Assembly of Aromatic Amino Acids: a Molecular Dynamics Study. *Phys. Chem. Chem. Phys*. 2018, 20, 30525- 30536.
2. Frederix, P. W. J. M.; Ulijn, R. v.; Hunt, N. T.; Tuttle, T. Virtual Screening for Dipeptide Aggregation: Toward Predictive Tools for Peptide Self-Assembly. *Journal of Physical Chemistry Letters* **2011**, *2* (19), 2380–2384. https://doi.org/10.1021/JZ2010573/SUPPL_FILE/JZ2010573_SI_001.PDF.
3. Frederix, P. W. J. M.; Scott, G. G.; Abul-Haija, Y. M.; Kalafatovic, D.; Pappas, C. G.; Javid, N.; Hunt, N. T.; Ulijn, R. v.; Tuttle, T. Exploring the Sequence Space for (Tri-)Peptide Self-Assembly to Design and Discover New Hydrogels. *Nature Chemistry 2014 7:1* **2014**, *7* (1), 30–37. https://doi.org/10.1038/nchem.2122.

Part 1 – Atomistic MD Simulation to model the Self-Assembly of Phe (F) in Water

Phenylalanine is one of the three aromatic amino acids and is well known to undergo self-assembly into long fibrillar crystalline structures. Other than the interesting material properties of these structures such as water responsiveness, they have also been implicated to play a role in medical conditions such as Phenylketonuria where the concentration of F increases due to missing enzymes involved in its downstream metabolic processing.

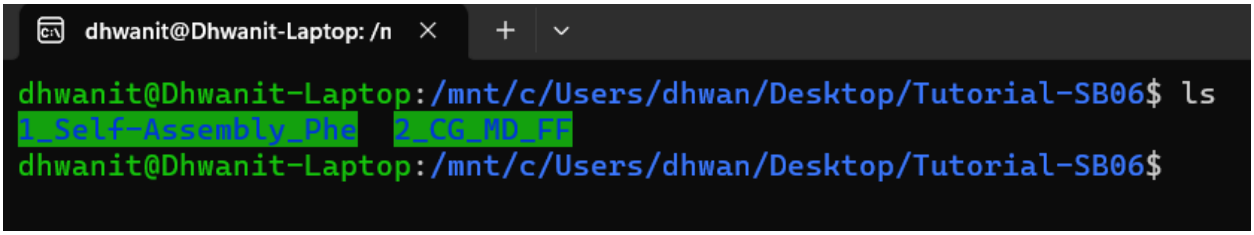
We will use an all-atom MD simulation using the OPLS-AA force field and TIP3P water model to study this self-assembly and gain insights into the non-covalent driving forces for the same.

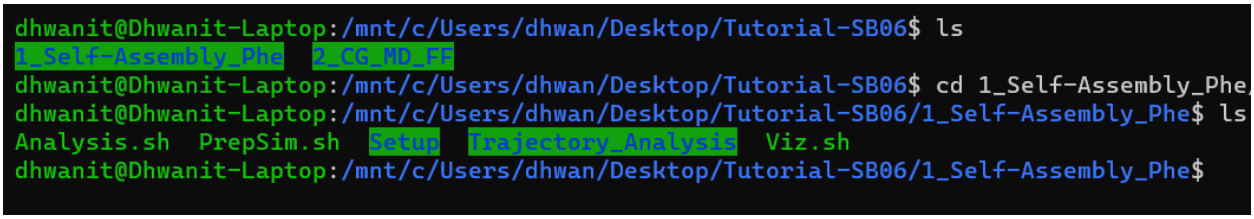
1 – Download the files for this tutorial from the Dropbox Link –

https://www.dropbox.com/scl/fo/vrrbd4g07ourdjlp3xsb2/h?dl=0&rlkey=kli54pm5wcbukdcvzsfs8f3lt

Once downloaded, unzip this compressed file to generate a directory containing the files and scripts for the tutorial.

2- Change to the first directory as shown below –

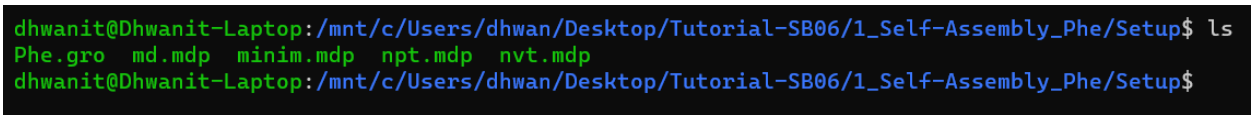




This directory 1_Self-Assembly_Phe contains two subdirectories called Setup and Trajectory_Analysis and three shell scripts that end in .sh.

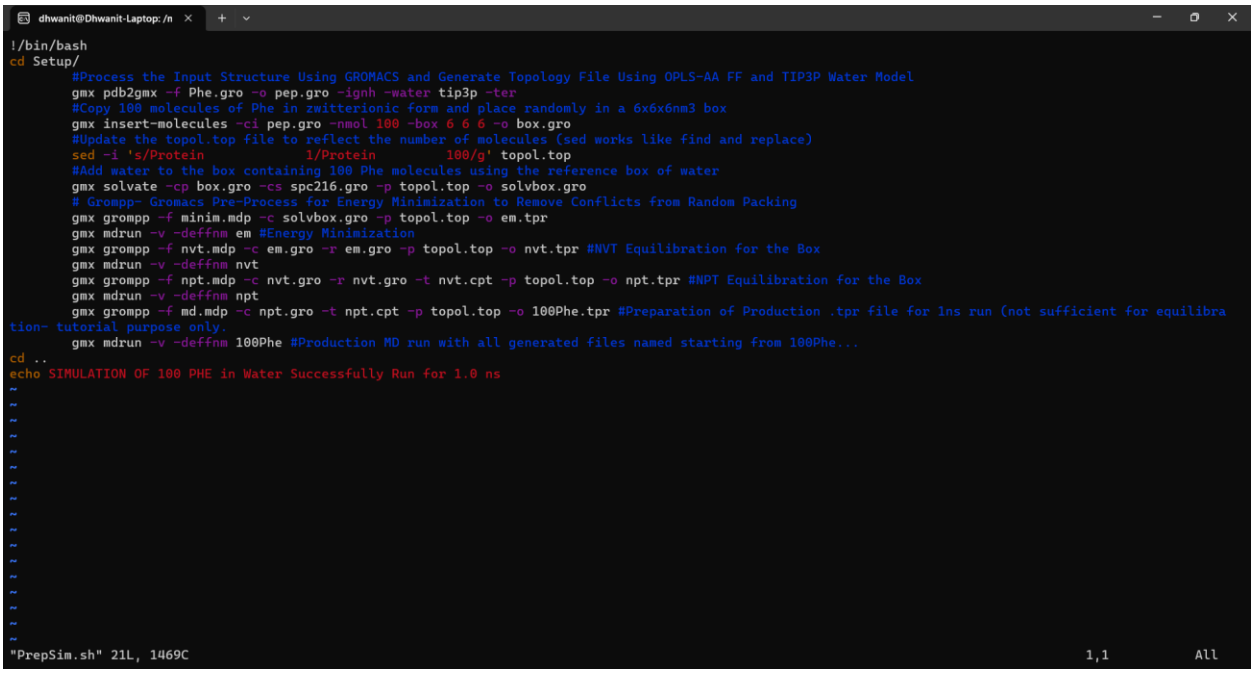
We are going to execute the PrepSim.sh first to setup and run a short MD simulation to model the self-assembly of Phe.

This script acts on the contents of the Setup/ directory whose contents are shows below-



The contents of this directory basically contain the requirements for setting up and running an MD simulation. The structure of the starting amino-acid/peptide or molecule is typically supplied as a .pdb or .gro file. These can be generated using VMD, PyMOL or Avogadro. The folder also contains .mdp files, these specify the parameters to be used for energy minimization (em.mdp), number, volume, temperature equilibration (nvt.mdp), number, pressure, temperature equilibration (npt.mdp) and the final md.mdp which is also an *NPT* type equilibration in this case with no restraints on the solute molecules to allow self-assembly. These .mdp files are force-field specific and sometimes ions.mdp is also required in case ions need to be added to neutralize the box etc.

The script to perform a short setup and run of this self-assembly of 100 Phe molecules in a 6x6x6 nm^3 box with water is shown below-

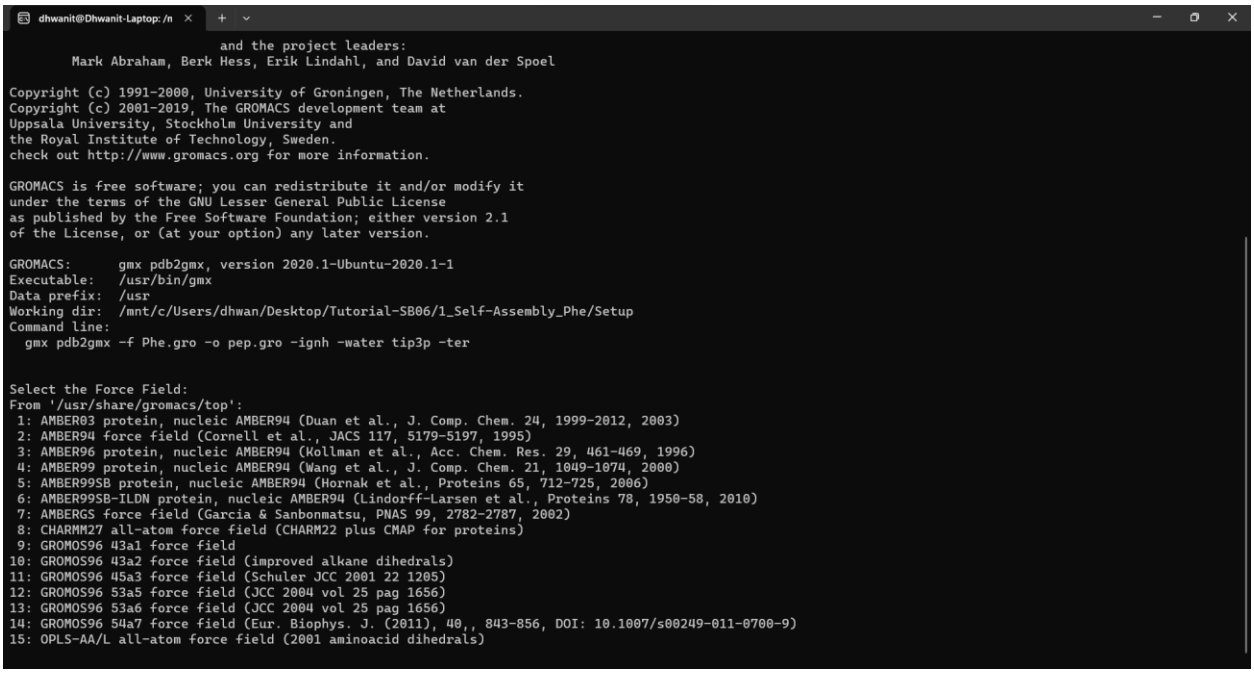


The commented-out lines in blue that start with a # are not read by BASH.

To execute this script, in your terminal, type –

bash PrepSim.sh

This will run the script using BASH and the first step will call GROMACS (gmx) to read the input structure and generate a topology. The *gmx pdb2gmx* runs interactively and will ask for a choice of forcefield-



Choose OPLS-AA/L which is option 15 by typing 15 and pressing the enter key.

It will then ask you to assign termini protonation states, at pH 7, we expect these to be ionized –

```
Select start terminus type for PHE-1
 0: NH3+
 1: ZWITTERION_NH3+ (only use with zwitterions containing exactly one residue)
 2: NH2
 3: None
1
Start terminus PHE-1: ZWITTERION_NH3+
Select end terminus type for PHE-1
 0: COO-
 1: ZWITTERION_COO- (only use with zwitterions containing exactly one residue)
 2: COOH
 3: None
1
```

Choose option 1 for both termini.

The rest of the script will automatically prepare a box, solvate it with water, and do the equilibration steps and a short 1ns MD run. This trajectory generated will not be converged in terms of self-assembly. Therefore we use the second directory to perform the analysis where an existing longer trajectory is available.

Analysis of the Self-Assembly of 100 F from a 500ns all-atom trajectory

To do the analysis, we will use the analysis.sh script also in the same directory 1_Self-Assembly_Phe –

```
dhwanit@Dhwanit-Laptop: /n
!/bin/bash
cd Trajectory_Analysis/
        gmx make_ndx -f mdrun.gro -o index.ndx #Making an index file to hold atom numbers of N and C of Phe
        #Solvent-Accessible Surface Area (SASA) Analysis
        gmx sasa -f mdrun.xtc -s mdrun.tpr -n index.ndx -tu ns -o sasa_100F.xvg
        #Hydrogen Bonding Analysis
        gmx hbond -f mdrun.xtc -s mdrun.tpr -tu ns -n index.ndx -num hbnum_Phe-Phe.xvg
        gmx hbond -f mdrun.xtc -s mdrun.tpr -tu ns -n index.ndx -num hbnum_Phe-Water.xvg
        #Radial Distribution of N around C of Phe
        gmx rdf -f mdrun.xtc -s mdrun.tpr -n index.ndx -tu ns -b 250 -o rdf_N-C_100F.xvg
cd ..
echo Successful Analysis of Self-Assembly of 100Phe molecules in a box of water
~
~
~
~
~
~
```

This script is in 3 parts, part 1 will make an index file that holds a list of atoms of interest from our simulation box. The second part measures hydrogen bonds between Phe residues and Phe residues and water. The third part measures a radial distribution function between the N atom of the N-terminus and the C atom of the C-terminus.

Run this script using-

Bash Analysis.sh

```
 nr : group      '!': not  'name' nr name   'splitch' nr    Enter: list groups
 'a': atom       '&': and  'del' nr         'splitres' nr   'l': list residues
 't': atom type  '|': or   'keep' nr        'splitat' nr    'h': help
 'r': residue              'res' nr         'chain' char
 "name": group             'case': case sensitive           'q': save and quit
 'ri': residue index

> a N & r PHE

Found 100 atoms with name N
Found 2300 atoms with residue name PHE
Merged two groups with AND: 100 2300 -> 100

 15 N_&_PHE             :   100 atoms

> a C & r PHE

Found 100 atoms with name C
Found 2300 atoms with residue name PHE
Merged two groups with AND: 100 2300 -> 100

 16 C_&_PHE             :   100 atoms
```

Specify the N and C atoms of interest as shown above and press enter to add new entries. Once these have been added, type q and press enter.

SASA is typically measured for the Protein group 1, in this case the 100 Phe molecules.

For hydrogen bonding, specify the correct index groups for Phe-Phe and Phe-Water (SOL) hydrogen bonds.

Finally, for measuring the radial distribution function of the N-termini around the C-termini, specify the newly created index groups of 15 and 16.

The .xvg files generated can be plotted using xmgrace, a command line plotting software simply using-

cd Trajectory_Analysis/

xmgrace nameofxvgfile.xvg

Alternatively, since these are text files, they can also be plotted using gnuplot or any other graph plotting tool such as Excel, Matlab, Origin etc.

Plot the SASA to see the trend in solvent accessibility and compute the aggregation propensity ($SASA_{initial}/SASA_{final}$).

Plot the two hydrogen bonding curves. What non-covalent interaction is driving this trend? Comment on the changes in Hydrogen bonding between Phe and Water.

Does the intense peak in the RDF of N-C support the non-covalent interaction previously discussed?

Visualization of the Self-Assembly of Phe all-atom trajectory using VMD

To visualize the trajectory, simply run the BASH script -

bash Viz.sh

Choose the Protein index 1 for centering and output of the .xtc (trajectory) and .gro (structure) files.

Change representations of the amino acids to Licorice and color them by atom. Apply trajectory smoothening and comment on the correlation between the analysis plots and visualized trajectory.

Part 2 – Coarse-Grained MD Simulation to model the self-assembly of FF in Water

Change to the 2_CG_MD_FF/ directory whose contents are shown below –

```
dhwanit@Dhwanit-Laptop:/mnt/c/Users/dhwan/Desktop/Tutorial-SB06/2_CG_MD_FF$ ls
AnalysisTrajectory          CG_SelfAssembly_Viz.sh  SimulationSetup  sim_FF_selfassembly.sh
CG_SelfAssembly_Analysis.sh  CG_Viz.sh              martinize.py
dhwanit@Dhwanit-Laptop:/mnt/c/Users/dhwan/Desktop/Tutorial-SB06/2_CG_MD_FF$ cat sim_FF_selfassembly.sh
!/bin/bash
cd SimulationSetup
        python3 ../martinize.py -f FF.pdb -x FF_CG.pdb -o topol.top -p Backbone -ff martini21 -ss EE
        sed -i 's/Protein_A      1/Protein_A     300/g' topol.top
        sed -i -e '$a\' topol.top
        gmx insert-molecules -ci FF_CG.pdb -nmol 300 -box 13 13 13 -o box.gro
        gmx solvate -cp box.gro -cs water.gro -radius 0.21 -p topol.top -o solvbox.gro
        gmx grompp -f minim.mdp -c solvbox.gro -p topol.top -o em.tpr
        gmx mdrun -v -deffnm em
        gmx grompp -f npt.mdp -c em.gro -r em.gro -p topol.top -o npt.tpr
        gmx mdrun -v -deffnm npt
        gmx grompp -f md.mdp -c npt.gro -t npt.cpt -p topol.top -o 300FF_CG.tpr
        gmx mdrun -v -deffnm 300FF_CG
cd ..
        echo Coarse-Grained FF Self-Assembly Simulation Successfully Setup and Run
dhwanit@Dhwanit-Laptop:/mnt/c/Users/dhwan/Desktop/Tutorial-SB06/2_CG_MD_FF$
```

To run a CG MD Simulation, we first convert the atomistic FF.pdb structure to a CG representation using the martinize.py script. The rest of the workflow is similar to the atomistic equilibration, expect that the .mdp files are for the MARTINI 2.1 Forcefield with a large timestep compared to all-atom ones.

On executing the sim_FF_selfassembly.sh script, you will generate a 13x13x13 nm^3 box containing coarse-grained FF and MARTINI water. This will be equilibrated and simulated for 5ns.

The other directory in this part of the tutorial contains a 2.5 μs trajectory of the self-assembly of 1600 FF peptide molecules and will be used for the analysis and visualization part.

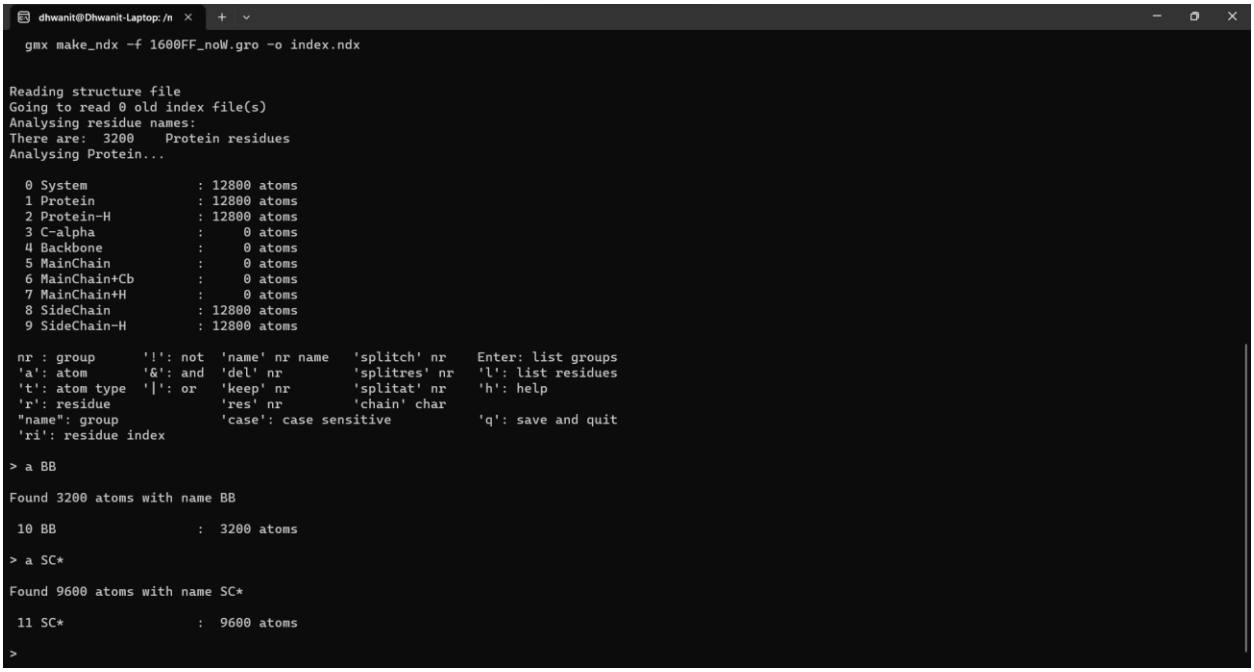
First let us visualize the difference between all-atom and CG structure of FF by overlying them in VMD by executing the CG_Viz.sh script. Represent the CG structure using the vdW transparent beads. Identify the backbone and sidechain beads.

SASA Analysis

Execute the analysis script –

```
!/bin/bash
cd AnalysisTrajectory/
gmx make_ndx -f 1600FF_noW.gro -o index.ndx
#SASA Analysis
gmx sasa -f 1600FF_noW.xtc -s 1600FF_noW.gro -n index.ndx -tu ns -probe 0.4 -surface 1 -output 10 11 -o sasa_1600FF.xvg
cd ..
echo ANALYSIS DONE
~
~
```

In this index file preparation, create entries for the Backbone and Sidechain beads as shown below –



10, 11 are index groups for BB beads and SC beads and will be used in the SASA computation to understand the contribution of each to the self-assembly.

Plot the SASA xvg graph and comment on the differential solvent exposure change for the SC and BB beads.

Visualize these differences in the overall trajectory of the self-assembly and identify if the changes in SASA for the individual bead index (SC vs. BB) agree with the self-assembled nanostructures observed.