CHEM-596 Practice Session 6: Molecular Dynamics simulations: system setup, execution, and trajectory analysis

Name: ________________________

Due: May 2, 2023 (11:59 pm)

In this practice, you will practice (i) setting up molecular dynamics (MD) simulations for a protein-ligand complex using AmberTools and (ii) the full process of MD simulation for a solute-solvent system (HBDI− in water) from parameterization and system setup to analysis of the MD trajectory. You will also practice using VMD for visualization and analysis. Note that your task in **Exercise 1** is identical to one the examples I demonstrated during **Lecture 19** (Tuesday April 18).

You will need the AmberTools MD software package to complete the two exercises below, which has been installed on both the ITCOTCK and DGX servers. If you haven't set up Amber before on these two servers, please do the following:

- On ITCOTCK:

```
echo "source /opt/amber22/amber.sh" >> $HOME/.bashrc
source $HOME/.bashrc
```

- On DGX:

```
echo "source /home/mao/Amber/amber22/amber.sh" >> $HOME/.bashrc
source $HOME/.bashrc
```

After you do these, you will get access to the programs under AmberTools (and you don't need to do it again next time you log in). You can check if you have successfully set up Amber by typing "`which sander`" on your terminal and hit Enter: on ITCOTCK you should see "`/opt/amber22/bin/sander`", and on DGX you should see "`/home/mao/Amber/amber22/bin/sander`".

During the practice session today, **please use the ITCOTCK server if your last name initial is A–J and use the DGX server otherwise**. You should upload the files provided In addition to AmberTools, you will also need to use the VMD software which should have been installed on your personal computers (see a brief instruction under "Pages" on Canvas).

You will need to complete the two exercises below step by step following instructions and submit the files (including images and movies) that are requested. There are also a few simple questions for you to complete on this handout. **8 points** are associated with the completion of this handout and the rest **2 points** are for attendance and completion of the in-class quiz.

Exercise 1: Setting up the MD simulation for a protein-ligand complex using AmberTools

In this exercise, you will use Ambertools to prepare the MD simulation for the complex between sustiva (a small-molecule drug serving as an inhibitor) and the protein structure of the HIV-1 reverse transcriptase (RT), which we refer to as the sustiva-RT complex below. From the provided PDB file `1FKO_trunc.pdb`, you can set up the MD simulation with the following steps

1. We start from processing the original PDB file using the `pdb4amber` script provided in AmberTools. This involves the following two commands:

```
pdb4amber -i 1FKO_trunc.pdb -o 1FKO_pro_dry.pdb -p --dry --add-missing-atoms
pdb4amber -i 1FKO_pro_dry.pdb -o 1FKO_pro_processed.pdb -y
```

In the first command, "`-p`" stands for "selecting protein only", "`--dry`" removes all the crystal waters, and "`--add-missing-atoms`" adds the missing heavy and H atoms. The second command then removes all the hydrogen atoms (through "`-y`") in the PDB file generated by the first command: by doing this, we leave the task of adding missing H's to the `tleap` program that will be invoked in a later step. The output PDB file of the second command, `1FKO_pro_processed.pdb`, will be used later for the MD setup.

2. A byproduct of the first `pdb4amber` command above is another PDB file `1FKO_pro_dry_nonprot.pdb`, which contains the non-protein part of the original PDB file. By investigating this file, you will find that it simply contains the small-molecule ligand (labeled as "HETATM" with resname "EFZ"). This small molecule needs to be separately parameterized within the Generalized Amber Force Field (GAFF) using the `antechamber` program. To make it ready for that, we first use the `reduce` command to add the missing H's:

   ```
   reduce 1FKO_pro_dry_nonprot.pdb > sustiva_h.pdb
   ```

   While not absolutely necessary, let's also update the resname for the ligand from "EFZ" to "SUS". This can be easily achieved using a Linux command:

   ```
   sed -i 's/EFZ/SUS/g' sustiva_h.pdb
   ```

3. Now we can use the `antechamber` program to parameterize the sustiva molecule:

   ```
   antechamber -i sustiva_h.pdb -fi pdb -o sustiva.mol2 -fo mol2 -c bcc -s 2
   ```

   Here "`-i`"/"`-fi`" and "`-o`"/"`-fo`" specify the input/output file name and format, respectively; "`-c`" specifies the scheme to generate the atomic charges and here we choose the semi-empirical method AM1-BCC; "`-s`" specifies the verbose level of the output, which is not too important. The output file `sustiva.mol2` is what we really need, which defines this small molecule. We can then use `parmchk2` to check the availability of the parameters in GAFF and generate the `.frcmod` molecule, which contains information about the force field parameters that the program is "less sure" about (typically found by analogy):

   ```
   parmchk2 -i sustiva.mol2 -f mol2 -o sustiva.frcmod
   ```

   You should inspect the `sustiva.frcmod` file generated. If it contains no attention information ("ATTN"), then we are good to go.

4. You are now ready to set up the MD simulation for the sustiva-RT complex, i.e., to generate the topology and input coordinates. While most of the times we execute `tleap` through an input file that contains a series of commands, as your very first practice it is better for you to go through the procedure step by step. First, type "`tleap`" on your command line and then hit Enter. The commands you need to set up the simulation are available below (note: the lines starting with a "#" are **comments** which should NOT be executed):

   ```
   # load the force field parameters
   source leaprc.protein.ff19SB
   source leaprc.gaff
   source leaprc.water.opc

   # load the sustiva ligand
   loadamberparams sustiva.frcmod
   SUS = loadmol2 sustiva.mol2
   check SUS

   # load the protein (RT)
   RT = loadpdb 1FKO_pro_processed.pdb
   check RT
   ```

```
# form the sustiva-RT complex
complex = combine {RT SUS}
# solvate, then add counterions
solvateOct complex OPCBOX 12.0
addIons complex Na+ 0
addIons complex Cl- 0
# save the coordinates and topology files
saveamberparm complex sustiva_RT_solvated.prmtop sustiva_RT_solvated.inpcrd
savepdb complex sustiva_RT_solvated.pdb
# we are done!
quit
```

The generated topology (`.prmtop`) and input coordinates (`.inpcrd`) files are what we need to proceed with Amber MD simulations, i.e., we are done with the setup now.

We are now going to use VMD to visualize the solvated sustiva-RT complex we just set up. Open up VMD, select "`File` → `New Molecule`" and then load the PDB file `sustiva_RT_solvated.pdb` saved by `tleap` (in this case VMD should be able to automatically detect the file type and change it to "PDB"). Let's first turn off the axis ("`Display` → `Axes` → `Off`") and change the view from "`Perspective`" to "`Orthographic`" (also under "`Display`"). Then we are going to create separate representations for protein, ligand (`resname SUS`), and Cl^- ions (`resname "Cl-"`) under `Graphics` → `Representations`. The details are specified in **Table 1**:

Table 1: Representations needed in the VMD visualization of the solvated sustiva-RT complex

Selection	Drawing Method	Coloring Method
protein	NewCartoon	Secondary Structure
resname SUS	CPK	Name
resname "Cl-"	VDW	Color ID 7 (green)

For the VDW representation for Cl^-, also increase the "`Sphere Resolution`" to 30. If you are unfamiliar with how to create these representations for different components of the system, please watch the video recording for the VMD demos in class. Now you can save the image in the OpenGL Display window via `File` → `Render`: the `.tga` format is OK for submission but ideally you should convert that to a `.pdf` or `.png` file.

Questions to answer:

1. Is the simulation we set up an explicit solvent or implicit solvent simulation? **(0.25 pt)** What will be the shape of the periodic box used in the simulation we set up? **(0.25 pt)**

2. How many missing atoms have been added to the system by `tleap`? **(0.25 pt)** What type of ions have been added and how many? **(0.25 pt)** (**Hint:** you should be able to find the answers from the `leap.log` file if you didn't pay much attention when executing the `tleap` commands)

Files to submit:

1. The input coordinates and topology files you generated using `tleap` **(0.5 pt)**

2. Visualization of `sustiva_RT_solvated.pdb` generated using VMD based on the given specifications in Table 1 **(1 pt)**

Exercise 2: MD simulation of HBDI− in water

HBDI (*p*-hydroxybenzylidene-2,3-dimethylimidazolinone) is a synthetic analog of the chromophore in green fluorescent protein (GFP), whose anionic form (HBDI−) is shown in Fig. 1. In this exercise, we will

1. Generate the force field parameters for HBDI− and set up the MD simulation for a single HBDI− solvated in water
2. Complete the energy minimization and heating steps with the input coordinates and topology files you generate and the provided input files for the MD program `sander`
3. Perform several trajectory analysis tasks based on a provided trajectory obtained from a production MD simulation

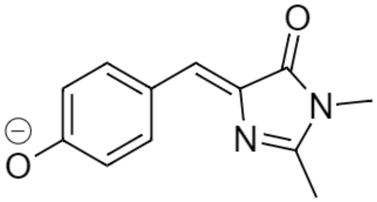


Figure 1: The molecular structure of HBDI−

Here we start from a DFT-optimized HBDI− in the XYZ format ("`HBDI-_optimized.xyz`"). You can open it up using IQmol, and then click "`File` → `Save as`". Make sure that you change the file format to "`.pdb`". Let's simply name it as "`HBDI-_iqmol.pdb`". We can then use the `pdb4amber` program to process this PDB file to make it fully compatible with Amber:

```
pdb4amber -i HBDI-_iqmol.pdb -o HBDI-_amb.pdb
```

You may have noticed that in the original PDB file the resname for HBDI− is "`UNL`", which stands for "unknown ligand". Let's update it to "CRO" using the same Linux command ("`sed`") as in Exercise 1:

```
sed -i 's/UNL/CRO/g' HBDI-_amb.pdb
```

Now you are ready to use `antechmaber` to generate the force field parameters:

```
antechamber -i HBDI-_amb.pdb -fi pdb -o HBDI-.mol2 -fo mol2 -c bcc -nc -1 -m 1 -s 2
```

Note that an additional keyword "`-nc -1`" is needed which specifies the charge of the small-molecule species ("`nc`" stands for "net charge"). Then, similar to the last example, we use `parmchk2` to check the availability of the force field parameters in GAFF:

```
parmchk2 -i HBDI-.mol2 -f mol2 -o HBDI-.frcmod
```

Examine the `HBDI-.frcmod` file obtained by opening it up using `vi`. You will notice that it contains more lines this time with a few items of pretty high *penalty score* (mostly for dihedrals), which renders a caveat that some of these force field parameters may *not* be very accurate. Nevertheless, for now we can just proceed without paying extra attention to these as long as the program is able to assign force field parameters from GAFF.

Now we are ready to use the `tleap` program to complete the MD setup. Instead of doing interactively, in this exercise we will run `tleap` with an input file, which is more convenient in practice since you can easily start from a template. Here a template for the `tleap` input file has been provided ("`HBDI-_leap.in`"). You need to complete the following sections based on the comment lines:

```
# Solvate in a 12.0 Angs TIP3P truncated octahedron box
[to be completed...]

# Add Na+ / Cl- to neutralize
[to be completed...]
```

In general the commands should be *very similar* to what we used in **Exercise 1**, except that the name for the water box should be changed to "TIP3PBOX" here. After you fill in these lines, simply execute

```
tleap -f HBDI-_leap.in
```

to complete the MD setup. As in the previous exercise, you are supposed to use VMD to visualize the MD setup, which can be achieved by loading the PDB file saved by `tleap` ("`HBDI-_water.pdb`"). The visualization should contain the representations as specified in **Table 2**:

Table 2: Representations needed in the VMD visualization of the solvated HBDI−

Selection	Drawing Method	Coloring Method
resname CRO	CPK	Name
resname "Na\+"	VDW	Color ID 27 (magenta)
water	line	Name

Now we are ready to proceed with MD simulations with the prepared input coordinates ("`HBDI-_water.inpcrd`") and topology ("`HBDI-_water.prmtop`") files. As we introduced in class, the complete procedure for an MD simulation typically involves (i) energy minimization, (ii) heating and equilibration, and (iii) production MD run. With AmberTools, all these steps will be completed using the `sander` program. In this exercise, **you only need to complete the minimization and heating steps** with the given input files for `sander` ("`min.in`" and "`heat.in`"); the output ("`prod_md.out`") and trajectory ("`prod_md.crd`") files obtained from the production MD run has been provided, based on which you will need to complete a few trajectory analysis tasks.

First, we minimize the energy of the system with command:

```
sander -O -i min.in -o min.out -p HBDI-_water.prmtop -c HBDI-_water.inpcrd -r min.rst &
```

Here "`-i`" and "`-o`" specify the names of the input and output files, respectively, "`-O`" means overwriting the output file ("`min.out`") if it already exists, "`-p`" and "`-c`" specify the names of the topology and input coordinates files, respectively, and "`-r`" specifies the name of the *restart* file, which in this case will just be the final structure obtained from the energy minimization.

From the input file "`min.in`", you should be able to tell that it is doing 3,000 minimization steps in total, including 1,000 steepest descent steps followed by 2,000 steps of conjugate gradient optimization. For the full explanation of the keywords in "`min.in`", please refer to the alanine dipeptide tutorial on the Amber website (under "**Minimization input**").

After the minimization is done (which should only take ∼1 min), open up the "`min.out`" file and report the energy for the initial and final steps **(0.5 pt)**:

- Initial energy (`NSTEP = 1`):
- Final energy (`NSTEP = 3000`) :

By skimming through the energy values printed out every 100 steps, is the system energy changing along the right direction during the *energy minimization* process? **(0.25 pt)**
Your answer:

We then perform a relatively short MD simulation to (i) bring the system from 0 K to 300 K and (ii) let the system equilibrate at 300 K. From the last 3 lines of the input file "`heat.in`", you can tell that in the first 10,000 MD steps (20 ps with a time step of 0.002 ps) the temperature will be elevated from 0 to 300 K, and then the system will equilibrate at 300 K for another 15,000 time steps (30 ps). For the rest of the input keywords please refer to the tutorial on the Amber website again (under "**Heating input**").

The command to execute for the heating step:

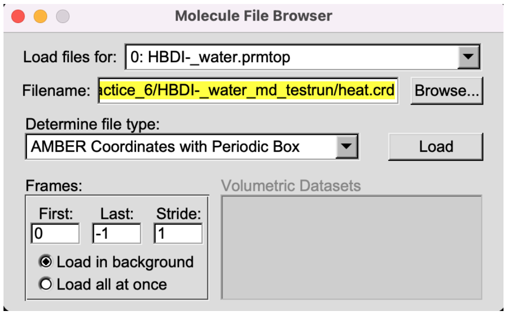
```
sander -O -i heat.in -o heat.out -p HBDI-_water.prmtop -c min.rst -r heat.rst -x heat.nc &
```

Here we set "`-c min.rst`" so that the input coordinates come from the restart file generated by the energy minimization step, and the new keyword "`-x`" specify the output trajectory file: with 25,000 time steps in total and "`ntwx = 250`" (saving coordinates every 250 steps) in `heat.in`, the resulting trajecotry file `heat.nc` should contain 100 MD snapshots.

The trajectory file `heat.nc` is a binary file which cannot be read directly using `vi` or other text editors and may not be supported by VMD (at least on Mac that is the case). In order to visualize and analyze the heating trajectory using VMD, let's first convert `heat.nc` into a text trajectory file. This can be done using the following `cpptraj` command:

```
cpptraj -p HBDI-_water.prmtop -y heat.nc -x heat.crd
```

You can now copy `heat.crd` back to your computer. To visualize it using VMD, under "`File` → `New Molecule`" you need to first load the topology file `HBDI-_water.prmtop` (make sure that the file type is shown as "`AMBER7 Parm`"). Now the top line in the window should change to "`Load files for 0:HBDI-_water.prmtop`": click "`Browse`", select "`heat.crd`" from the location you save this file, and make sure to **change the file type manually** to "`AMBER Coordinates with Periodic Box`" (**important**). The file loading window should now look like in **Fig. 2**:


Molecule File Browser

Load files for: 0: HBDI-_water.prmtop

Filename: actice_6/HBDI-_water_md_testrun/heat.crd

Determine file type: AMBER Coordinates with Periodic Box

Frames: First: 0 Last: -1 Stride: 1

● Load in background

○ Load all at once

Figure 2: Loading Amber trajectory to VMD

Now you can click "`Load`" to load the trajectory to VMD. You may notice that there is an extra bond connecting the two H's of each water, which is because VMD regards TIP3P as a rigid water model. We don't need to deal with that right now. Instead, let's create the representations for **only** the HBDI− chromophore (`resname CRO`) and the Na$^+$ cation (`resname "Na\+"`) with the same specifications in **Table 2**, then choose "`Mouse` → `Label` → `Bonds`" and click on the Na$^+$ cation and the oxygen atom on the **phenol ring** of HBDI−. You should be able to see two atom labels and a "bond" showing up on the OpenGL Display. Play the trajectory with a slower speed and check out how this distance changes over the entire trajectory.

To get a quantitative measure, go to "`Graphics` → `Labels`" and select "Bonds" from the dropdown menu; then click on the bond between "`CRO1:O`" and "`Na+2:Na`" and switch to "`Graph`" tab, and click "Graph". This will produce a plot of this distance over time. Save this plot which needs to be submitted with your work.

For the production MD, the input, output, topology, and text coordinates files have been provided. While you are NOT supposed to run the production MD by yourself (since it is going to take 2-3 hours), the command for that looks like the following:

```
sander -O -i prod_md.in -o prod_md.out -p HBDI-_water.prmtop -c heat.rst -r prod_md.rst -x
prod_md.nc &
```

From the input file you should be able to tell that the production run is an **NPT simulation at 300 K**. The full explanation for the keywords in `prod_md.in` can be found in the same tutorial online (under "**Production input**"). Based on the total number of time steps (given by "`nstlim`") and the size of the time step ("`dt`"), **what is the total time duration of this simulation**? Show your math below and report your result **in ns** (1 ns = 10^3 ps = 10^6 fs): **(0.25 pt)**

Your answer:

Now let's jump directly to the trajectory analysis part. **Start a new VMD session** and load the trajectory "`prod_md.crd`" following the same procedure as introduced above (**Note: the topology file to load here should be the provided** "`prod_md.prmtop`" **rather than your own** "`HBDI-_water.prmtop`"). Then use the same procedure to generate the plot for the O(phenol)$\cdots$Na$^+$ over the entire trajectory and

save that. **Describe what you observe from this plot and its difference from the previous case: (0.5 pt)**
Your answer:

Let's now investigate the root-mean-square displacement of the HBDI− solute (excluding H's) over the production MD run. Go to "`Extensions` → `Analysis` → `RMSD Trajectory Tool`", change the selection in the first text box from "`protein`" (which we don't have here) to "`resname CRO`" and leave the "`noh`" (no hydrogen) option selected. On the right of the same window, make sure that "`Frame ref:`" is 0 and select "`Plot`" as well. To generate the so-called "best-fit" RMSDs, you should first click "`ALIGN`" first to let the program try it best to overlay the selected part on top of the reference frame (note that this will change the visualization of the trajectory as well), and then click "`RMSD`". A plot of the RMSD against the MD frame number will be generated. Save the plot which will need to be submitted with your work.

From the RMSD plot, you should be able to notice that there is a sudden "burst" roughly starting from frame 200. This most likely corresponds to some unusual event in the MD simulation. To further investigate that, first let's delete all the labels we have created so far under "`Graphics` → `Labels` → `Atoms`" and "`Graphics` → `Labels` → `Bonds`". Then we create a new label for one of the two H's at the *meta*-position of the phenolate group (using "`Mouse` → `Label` → `Atoms`"). Play the trajectory from frame 190 with a slower speed and watch how the position of the labeled H atom changes.

Based on your observation and the definition of RMSD, try to **give an explanation for this sudden increase of RMSD after frame 200 (0.5 pt)**. Based on the molecule structure of HBDI−, discuss how likely/unlikely it is for this "event" to occur and its implication for the force field parameters we generated for this molecule. **(bonus 0.5 pt)**
Your answer:

As the final task, let's create a movie for the production. First make sure that you delete the axes and the atomic label you just created. Then you should center the HBDI− molecule on the OpenGL Display and then add back the water molecules. Since the TIP3P water has the weird H−H connection, we should use the representations in **Table 3** for water: The OpenGL Display should now look like **Fig. 3**. Now go to "`Extensions` → `Visualization` → `Movie Maker`" to create a movie for this trajectory. If you are unfamiliar with the procedure, please refer to the recording of Lecture 18. Two things to note:

Table 3: Representations for TIP3P water in VMD

Selection	Drawing Method	Coloring Method
name O H1	Line	Name
name O H2	Line	Name

- Remember to set the working directory (otherwise you may have trouble finding the movie file generated)
- Under "`Movie Settings`", you should select "`Trajectory`"

Files to submit:

1. The completed `tleap` input file "`HBDI-_leap.in`" **(0.5 pt)** and the prepared input coordinates and topology files (`HBDI-_water.inpcrd` and `HBDI-_water.prmtop`) **(0.5 pt)**

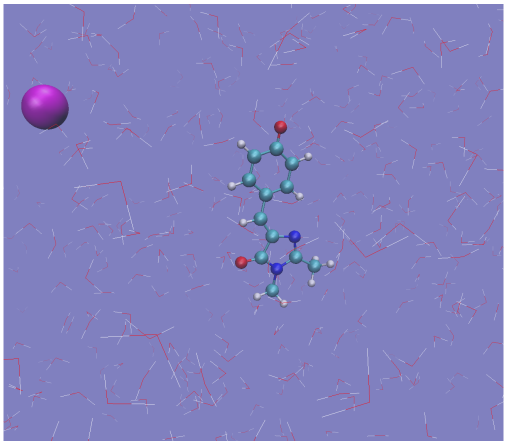


Figure 3: VMD view for the movie generation

2. VMD visualization of the MD setup (the "`HBDI-_water.pdb`" file generated by `tleap`) with the given specifications **(0.5 pt)**
3. VMD-generated plots of the O$\cdots$Na$^+$ distance over (i) the heating trajectory and (ii) the production MD run. **(1 pt)**
4. VMD-generated plot for the RMSD of the heavy atoms on HBDI$-$ along the production MD simulation **(0.5 pt)**
5. VMD-generated movie for the production MD run **(0.5 pt)**