**BIOS 10603 – Lab 5**

**Preparing your own protein for the simulation &**

**further ubiquitin simulation analysis**

**Name:**

Please insert all answers and figures into this Word document, and then save the completed lab as “Lastname\_Lab5.docx”. When you have completed the lab, submit it as an attachment to the Lab-5 assignment link on Canvas. The report is due by midnight the day after your lab period.

**Objectives:** In this week’s lab, you will run some analysis on your ubiquitin heat-up/ equilibration simulation. You will prepare your protein system for MD simulations, minimize it, and submit it for heating and equilibration. We will also introduce you to analysis of dihedral angles and Root Mean Square Fluctuation (RMSF) using VMD.

**Lab 5 Sections:**

1. Ubiquitin heat-up/ equilibration simulation analysis
2. Preparing your protein
3. Dihedral angle analysis using VMD
4. Root mean square fluctuation (RMSF) analysis
5. Equilibration submission for your protein

**Simulation setup steps**

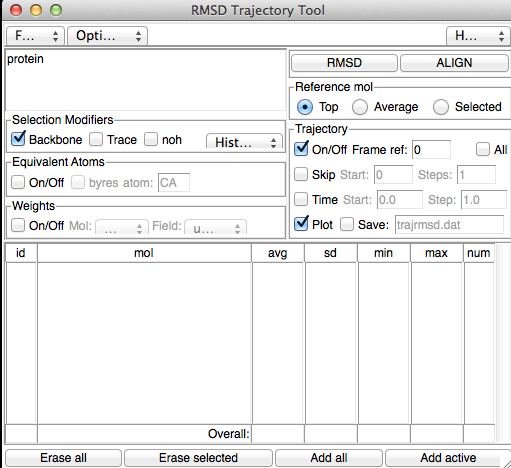
|  |  |  |  |
| --- | --- | --- | --- |
| **STEP** | | **DESCRIPTION** | **Lab** |
| **0** | Build PSF | Adding hydrogens to the protein structure | Lab 3 |
| Solvation | Putting the protein in a water box |
| Ionization | Adding salt at a specified concentration |
| Restraining | Adding restraints to the protein backbone |
| **1** | Energy Minimization | Using NAMD to move the protein into an energetically favorable conformation before kinetic energy is introduced to the system. |
| **2** | Heating up | Gradually warm up the system and apply atmospheric pressure. | Lab 3 & 4 |
| Equilibration | Gradually remove the restraints on the protein backbone to allow it to gradually adopts to the pressure and temperature applied without being deformed. Make the system as close as possible to physiological conditions. |
| **3** | Production Runs | Allowing the system to run undisturbed long enough to observe the wanted behavior. Usually broken up into smaller jobs to check for errors. | Lab 6 |
| **4** | Simulation Analysis | Energy, hydrogen bonds, pressure, temperature, volume NAMD Plot | Lab 3 & 5 |
| Density, RMSD, Rg, RMSF, Hydrogen bonds | Lab 4 & 5 |
| Secondary structures | Lab 6 |
| Timeline | Lab 7 |
| Other analyses | Lab 8 |

**Part-1: Ubiquitin simulation analysis**

Download your ubiquitin heat-up/equilibration **.dcd and .log** files from Midway to your computer.

**Q1.** Load your **solvate-ionize.psf** and the ubiquitin heat-up/equilibration trajectory .dcd into VMD (use the load data into molecule option to load the dcd INTO the psf).

**a.** Describe what you see happening to your protein as the simulation progresses. Hint: focus on the begining of the simulation and mainly on the protein by using the selection “not water”; show the backbone as new cartoon and add another representation using licorice to see the side chains.

**b.** Compare this simulation to the minimization trajectory. Explain your observation.

**c.** If you were to plot the Total Energy of both the minimization trajectory and the equilibration trajectory, what differences would you expect in the plot? (You can actually plot the energies but you do not need to attach a figure). What would account for the difference?

**Q2.** Use the RMSD trajectory tool (Extensions 🡪 Analysis 🡪 RMSD Trajectory Tool); make sure that “protein” is entered in the upper left-hand corner and that all **of the boxes in the picture (to the right) are checked.** **Click “ALIGN” and then “RMSD” and attach and interpret the figure. Explain the trends that you observe.**

**Q3.** Using **NAMD Plot**, attach a plot of the volume, pressure, and temperature for the ubiquitin equilibration trajectory and explain the trends you observe for each plot.

**Q4.** What may happen if you were to remove the backbone restraints too quickly?

**Part 2. Preparing Your Protein**

**The instructions provided in lab are for a general system. If your protein contains specific parts, such as ligands or multiple subunits (chains), please check with Dr. Haddadian or your TA about how to proceed. Create a directory on your laptop and download the PDB file for your protein and load it into VMD**

First, we need to check for missing residues. Open the PDB file with a text editor and look at the section of the header denoted by “REMARK”. Towards the end of this section, there may be a section denoted “MISSING RESIDUES”, with a list of missing residues underneath. If the missing residues listed are in the middle of the protein, you need to have Dr. Haddadian look at your protein before you continue. **The new AI database** [**Alpha Fold**](https://alphafold.com/) **may be useful if your structure has any kind of missing atoms. Please consult Dr. Haddadian for more information.**

Now, check if your protein has duplicate subunits included in the PDB file. To do this, create a graphical representation that colors the protein based on the selection “Chain”. If there are duplicate chains, select one of the chains using the command “chain A” for example. We want to remove any duplicate subunits because our simulation will run faster without them, allowing us to get more simulation time. **If the different chains in your protein are not identical, then you should not delete any chains.** You want to make sure that you use the "Biological Assembly" of your protein for modeling (Hint: refer to the PDB homework).

Then, we need to remove anything that might cause an error in the system preparation (ligands or modified amino acids that are not part of the regular CHARMM force field).Check if there are any ligands on your protein. Type “not protein” into the selection on VMD, and if anything is visible (use VDW as drawing method), these are most likely ligands or water molecules. We want to remove these from the protein, so create a new representation with just “protein” selected. If you also removed a subunit, you would want to use the selection “chain A and protein”, for example.

To save these changes, **highlight your protein name in the VMD Main window**. Go to File 🡪 Save Coordinates and create a new PDB file with the exact selection as above (i.e.: *Protein and chain A*). **You will use this PDB file for your simulations, instead of the original protein.**

**Q5**. Run your PDB through AutoPSF generator in VMD as you have done before. Are there any unparameterized components or portions that caused errors? If there are, look through your protein’s PDB page to find out what they are and describe them. Let Dr. Haddadian or your TA know about these error messages.

Solvate (by padding with 10Å of water on each side; remember to use the “rotate to minimize volume” option), ionize (neutralize the total charge with 0.1 mol/L salt concentration), center, and restrain your protein and submit it for 10,000 steps of minimization. Remember, the command to restrain your protein is

set sel [atomselect top “protein and backbone”]

$sel set beta 10

set sel [atomselect top all]

$sel writepdb XXXX-restrain.pdb

Open the XXXX-restrain.pdb file and check the beta column to make sure that the restraining is done correctly.

Copy the minimization config file from in the lab folder and change the following lines to match that of your protein

1. Coordinates
2. Structure
3. Output name
4. Parameter file name
5. Constraints (“on”)
6. Consref
7. ConsKFile
8. CellBasisVector1
9. CellBasisVector2
10. CellBasisVector3

**Create a new directory in Midway-3 scratch space with your protein name**, move all the needed files to this directory, and submit the minimization job (**ask for 1 hour and 2 nodes**). Please note that the minimization time will vary depending on the total number of atoms in your system. Throughout the next section, check to make sure your minimization finishes successfully – if it does not, let your TA or Dr. Haddadian know immediately.

**As you are waiting for the minimization to finish, complete the following two sections.**

**Part 3. Dihedral angle measurement using VMD**

As you may recall from lecture, the Ramachandran plot is very useful for visualizing the dihedral angles of a protein. We can use the Ramachandran Plot tool in VMD to analyze how the dihedral angles in our Ubiquitin protein change over the course of a simulation.

**Q6.** In VMD, go to Extension🡪Analysis🡪Ramachandran Plot. In the window that opens, select the molecule number corresponding to your simulation. Go to the first frame and play the trajectory while looking at the Ramachandran window.

**a. Describe your observation.**

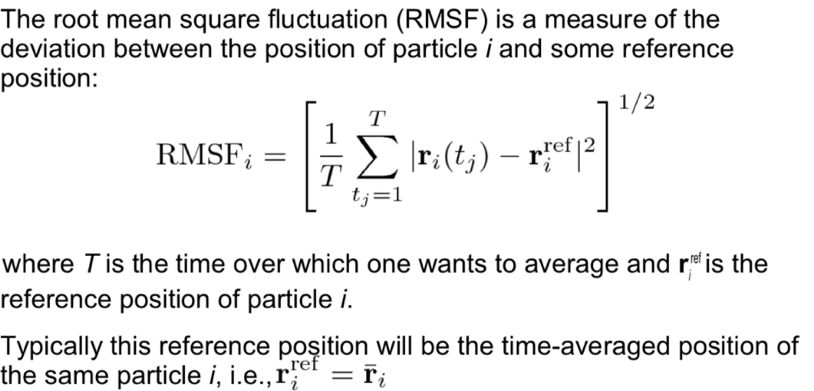
**b.** What residue most frequently goes outside the allowed regions? Why? (Hint: clicking on the yellow dots in the Ramachandran plot shows their names).

**Q7.** Generate a 3D histogram for one of the Alanine residues in Ubiquitin; take a screenshot and attach it to your report.

**a.** What does this plot represent (Hint: check the help option in the Ramachandran window).

**b.** In what secondary structure region is the peak of the histogram located? What does the peak represent in the context of the alanine residues within ubiquitin?

**Part 4. Root mean square fluctuation (RMSF) Analysis of Ubiquitin**

**

We are going to calculate a new metric for the ubiquitin trajectory called Root-Mean-Square-Fluctuation (**RMSF**). RMSF is a concept related to RMSD, but with several important differences. For RMSF, instead of plotting the average distance for **all atoms** (with respect to a reference structure) **against time**, we plot the average deviation (in Å) for **each** amino acid’s alpha-carbon (CA) in the protein **against the amino acid number**. Thus, for RMSD, the x-axis is time, while for RMSF, the **x-axis is residue number**. RMSF can essentially be thought of as a time-averaged RMSD per residue. This quantity gives us information about the **local changes** in the protein (in contrast to RMSD, which describes the protein’s global changes).

**Q8.** There is a script on Canvas called “RMSF.tcl”, which we want you to complete. Once you have done this, copy and paste the changed lines below (Hint: recall the TcL command lindex).

LINE 1:

LINE 2:

LINE 3:

LINE 4:

LINE 5:

LINE 6:

Two lines of sample R code useful for making the plot in the next question are:

rmsf = read.table("rmsf.dat-file-location/RMSF.dat",header=TRUE, sep="\t")

plot(rmsf$Residue, rmsf$RMSF, type="l", col="RED", lwd=3, xlab="Residue Number", ylab="RMSF")

**Q9.** **Align your ubiquitin simulation to the first frame using the RMSD tool as you did last week.** Next, type source RMSF.tcl to run this script.

1. Include a plot of the RMSF with labeled axes and explain any sections (residue range) with high RMSF values (peaks in the RMSF plot). Review the motion of those segments in VMD.
2. Why do the N and C termini fluctuate more than the other residues in the protein? Knowing this, what different selection could you have made that would have yielded a lower RMSF value, on average?
3. Why did we only consider the C atoms in the RMSF calculation?
4. Why is it vital to align the protein trajectory to a reference frame before RMSF calculations (hint: run the RMSF script on the unaligned trajectory and compare your plot to what you obtained in part A).

**Part 5: Equilibration submission for your protein**

Download **your protein’s minimization log file** from Midway to your laptop.

**Q10**.

**a.** Using “NAMD Plot”, attach a plot and **explain any trends you observe** for the total energy of your protein. **As in lab 4, export the data (to ASCII Matrix) and plot it in R.**

**b.** What does the total energy represent? How would you expect the total energy of your system to change during minimization? Does the total energy of your system change as expected?

**c.** Why are these checks important and necessary?

If your minimization was successful, we now want to submit it for heating up and equilibration. As a reminder, the steps are as follows (substituting “XXXX” for the code name of your protein):

**1.** You need to save the last frame of your minimization as XXXX-mini-LF.pdb

1. Delete all molecules in VMD, and then load the XXXX-mini-LF.pdb file. Run the following commands to restrain the last frame:

set sel [atomselect top “protein and backbone”]

$sel set beta 10

set sel [atomselect top all]

$sel writepdb XXXX-mini-LF-restrain.pdb

This will create the file XXXX-mini-LF-restrain.pdb in your VMD current directory. Open the created file and check the beta column to make sure that the restraining is done correctly.

1. Transfer the following files to Midway, in the same directory where you performed your own protein minimization: XXXX-mini-LF.pdb and XXXX-mini-LF-restrain.pdb.
2. Copy the heatup-equil configfile from the lab folder and rename it as   
   XXXX-mini-heatup-equil.conf. Open this file, and edit the following locations:

* Coordinates
* Structure
* Input name
* Output name
* Restart name
* Parameter file name
* Constraints (Make sure they are set to “on”)
* Consref (Use XXXX-mini-LF-restrain.pdb)
* ConsKFile (Use XXXX-mini-LF-restrain.pdb)
* CellBasisVector1
* CellBasisVector2
* CellBasisVector3

**Q11.** List the changes you made below.

1. When you have made all configfile edits, edit your job submit script. Ask for **1** node and **10** minutes. Then, change the config and log file names to XXXX-mini-heatup-equil.conf and XXXX-mini-heatup-equil.log
2. Finally, submit the job when you have verified that all of your edits are correct.

**After the job starts to run, check the log file for any possible errors (do not wait until next week’s Lab). If you notice any error messages, contact Dr. Haddadian or your TA immediately. You need to make sure that your job is running correctly before you leave the lab. It is difficult for Dr. Haddadian and the TAs to resolve your issues remotely.**

1. After the initial 10 minutes, if your job ran successfully, resubmit it asking for the full 4 nodes and 6 hours.

**Q12**. Please **briefly** answer the following questions about job submission. (You don’t need to write a long paragraph. We expect one or two sentences that concisely provide a reasonable answer.)

1. How do you decide how much time and how many nodes you are going to request for the job? Is there anything in the config file that may help you do so? What are the tradeoffs between asking more and asking for less? What are the tradeoffs between asking for more nodes and asking for more time?
2. Why do we ask you to do a 10-minute test run? Why is a short period of time enough to decide if the job can go through smoothly?
3. If the job is terminated before it finished due to the fact that you did not ask for enough computational power, how can you proceed by making use of the part we’ve already run, thus avoiding repeated computation?
4. After the job finishes, how can you check if your job has run correctly?