Lab #6: Molecular Modeling using TINKER Part 2

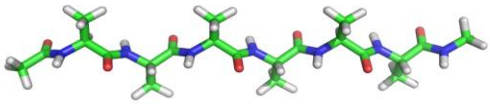
Submit the images of the structures and text output (the last 20 lines) for the lab report.

1. Convert PDB file to xyz files for molecular mechanics calculations

a. Start FFE, use the menu "File/Download from PDB" to fetch 1ejg. This is a small protein called Crambin.
b. Run ***pdbxyz*** command to convert it to 1ejg.xyz. Recall you specify commands in the ***Modeling Command*** tab. Choose amber99sb.prm when asked.
c. Select 1ejg.xyz on the left panel, Run ***analyze*** to compute energy. ***Minimize*** the refine the structure.
d. Note that the *iejg.xyz, 1ejg.key* etc. created by FFE will be located in *C:\Users\pren\MolecularTools\PDBdownloads*

FYI*: You could in principle set up any PDB using the above procedure. But some PDB files may contain ions or ligands and PDBXYZ may not handle them correctly. Also some proteins may have too many missing atoms/residues. You MAY need to edit the PDB file to delete any* ***heteroatoms (non-protein atoms)*** *if you have trouble converting and perform calculations on the resulting xyz file. To study these ligands, you have to come up with the force field parameters using other programs.*
**Swiss-PDB viewer is a free software that can add missing residues and loops for you.*
http://spdbv.vital-it.ch/loop_tut.html

2. Building your own peptide chain or protein from sequence



Strat FFE, use the "***Protein***" Modeling Command to build a 7-residue polyaniline (or any other peptide sequence of same length). There are number of options available under the "***protein***" command: change the filename to "pep"; select ACE and NME as the N and C terminals; for sequence, after select the secondary structures (sheet or helix…), click on ALA 7 times; "no" for cyclization, and hit GO. Select amber99sb.prm when asked which force field to use.

Now the FFE should have created the *pep.xyz* and *pep.key* files. Please use text editor to open and examine the files. They are located under MolecularTools\. For better organization, you can move them to MolecularTools\tinker\example (you may need to close them in FFE first). Now you can open them back and perform "analyze" and various other modeling commands.

3. Folding peptide using simulated annealing (*anneal*)

Use the peptide you build from above.
Edit the pep.key to include the following keywords using a text editor, save & exit.

```
PARAMETERS      C:\Users\pren\MolecularTools\ffe\..\tinker\params\amber99sb.prm

#implicit solvent enabled below
solvate gbsa

integrate  stochastic
rattle  bonds
archive
#parallel computing using 4 threads
openmp-threads 4
```

"#" in the key file means it is only a comment.
In the openmp-threads keyword, use a # of threads that is available on your computer.

Minimize the pep.xyz and then run the *"**anneal**"* command in FFE using the following suggested parameters:
Initial and Final Temperatures in Degrees K [1000,0] : 1000, 298
// The molecule will be heated up to 1000 K and cooled back to 298K

Enter the Number of Equilibration Steps [0] 1000
// Equilibrate 2 ps before the anneal

Enter the Number of Cooling Protocol Steps [2000] : 25000
// Cooling the temp from 1000 to 298 within 50 ps. This is not long enough to actually fold the peptide. A few ns is necessary even for a small peptide. The larger the molecule the longer it takes.

Use Linear, Sigmoidal or Exponential Cooling Protocol ([L], S or E): L
//This specifies how the temperature is varied during the cooling.

Time step in fs: 2
// in fs
Time between dump: 1
//in ps
Increase Atomic Weights by a Factor of 10^x [x=0.0]: 0
(Hit the enter without any input will set it to the default)

Optional information: If you wish to run this at the command line, you can put all the input parameters in a file and perform the simulation in one command:

```
cd  "directory where you store pep.xyz & pep.key"
%TINKER%\bin\nneal.exe  pep.xyz  <  input.txt  >  output.txt
```

In Windows OS, you can't really put the simulations in the background, i.e. you need to keep the FFE or *cmd* prompt window alive while simulation is running.

Watch the annealing movie after the job complete by loading the **pep.arc** in FFE. Capture the image of the final structure for lab report.

The rest is FYI only, not required.

Command line operations

1. Convert PDB file to xyz files for molecular mechanics calculations

To study any proteins using TINKER, the first step is to convert the PDB to xyz format using the ***"..\bin\pdbxyz.exe xxx.pdb***" command. This will assign force field atom types for each atom and add missing hydrogen atoms.

First read the instruction in the textbox below, then complete the *tasks a through c* on next page.

> ***How to convert xxx.pdb you've downloaded to xxx.xyz to use with TINKER:***
> In your working directory (*%TINKER% \example\)*, **create** a file xxx.key (where xxx is the same prefix as the PDB file). Make sure **do not** save it as xxx.key.txt (Windows may hide the extension *txt*).
>
> In the key file, specify oplsaa.prm (which is lcaoted in the *"%TINKER%\params"* directory) as the force field by including the following as the first line of the xxx.key file:
>
> *parameters ..\params\oplsaa.prm*
>
> "../" means going up one level in the directory tree to look for the "*params*" folder. You may also give the full path ("C:\...\params\") if your files are located somewhere else. You may specify other force field parameters in *params* folder as well (e.g. mm3, amber…)
>
> You can now convert the xxx.pdb to xxx.xyz by using "*pdbxyz*".

a. In the *example* directory, you should see a crambin.pdb. Delete crambin.xyz and crambin.key if they already exist. Create the crambin.key following the instruction above, and then convert the *crambin.pdb* to xyz by
   "..\bin\pdbxyz.exe crambin.pdb"
   Now you have the new crambin.key and crambin.xyz to work with.

b. Run *analyze* first to make sure xyz and key files work, and then *minimize*
   "..\bin\analyze.exe" xxx.xyz e
   "..\bin\minimize.exe" xxx.xyz 0.1

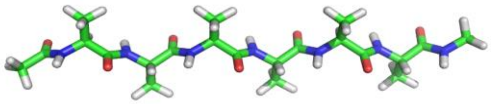
c. Download your favorite protein from www.rcsb.org and convert to xyz file and run *analyze* and *minimize* (see warning below).

Build a 7 residue polyalanine capped by ACE and NME terminal groups.

a. Start "*cmd*", "cd" into your "%TINKER%*example"* directory, type "***notepad pep.key***" to create a "pep.key" that add the following lines (in the textbox). If you use Mac, use any text editor to do the same but replace any "\" with "/" in the key file.

```
parameters ..\params\OPLSAAL.prm
#implicit solvent enabled below
solvate gbsa

integrate  stochastic
rattle  bonds
archive
#parallel computing using 4 threads
openmp-threads 4
```



"#" in the key file means it is only a comment.
You may need to change "..*params\OPLSAAL.prm"* to absolute path (*C:\Users…*) if you work in different folders.
In the *openmp-threads* keyword, use a # of threads available on your computer.

b. With the pep.key, run "..***\bin\protein.exe pep.xyz***" or ***"%TINKER\bin\protein.exe pep.xyz***" to build a peptide of any sequence. You can run it interactively by typing "*../bin/protein.exe*" at the command prompt, and enter filenames (name it "pep") and residues (one residue a line). You may built a specific configuration (helix or sheet) by giving the phi and psi angles following the residue name: ***ala -60 -60.*** Choose *No* for the last cyclic peptide option. Hit enter again to exit.

Now build a ***8mer polypeptide in alpha-helix conformation*** (meaning 7 residues + two caps) interactively. After executing the command, pep.xyz and pep.seq (sequence) files will be generated. Keep these files for future lab.

```
filename
title/molecule name
ACE
ala -60 -60
…
NME
Blank line
N
Blank line
```

<u>Instead of running the *protein* command interactively,</u> you may also put all the inputs in a file (e.g. named ***input.txt, with contents as shown in the textbox on the left***)
Next you can run the script to build a peptide (be sure to have the xyzfile.key or tinker.key present) by type:
"..\bin\protein.exe" < input.txt

You can convert the resulting .xyz file back to PDB by using "*xyzpdb.exe*".

<u>Similarly you can build NA using the "nucleic.exe" command.</u>