

Setting up a protein simulation Session 2 - Adding water and ions

Practical session: setting up the full system using AMBER

Begin by copying two files from the directory `~/setting-up` into the directory `~/amber/session2`: the original PDB file, minus the ligand (**1kyn-noligand.pdb**) and the file produced by WHAT IF (**prepdock.pdb**).

1. “Cleaning up” the PDB file

Examine the PDB output file from WHAT IF (**prepdock.pdb**) in a text editor. WHAT IF will have made some suggestions regarding the tautomeric/ionization state of the histidines in 1kyn. Where it thinks the δ -tautomer is correct, it will have added an HD1 atom. Where it thinks the ϵ -tautomer is correct it will have added an HE2 atom. Where it thinks the histidine should be protonated, it will have added both HD1 and HE2 atoms. However *tleap* needs different “flavours” of histidine to have different residue names, you need to change the residue names of any δ -histidines from HIS to HID, any ϵ -histidines to HIE and any protonated histidines to HIP.

Now look at the top of the file. You will see three “SSBOND” lines. These tell you that the Cysteines in this structure are oxidized, and where the disulfide cross-links are. Again, *tleap* is not clever enough to work this out for itself, so you must work through the file, and change all the “CYS” residue names to “CYX” to indicate this.

Finally, WHAT IF has a different idea from AMBER about how N-terminal hydrogens should be named. So you need to rename atom “H” in the N-terminal ILE residue to “H1”.

Save this file as **1kyn-corrected.pdb**. You are now ready to use *tleap*

2. Load and complete the structure in *tleap*

During this part of the tutorial we will be using *tleap* from AmberTools (freely available online). There are other versions of leap (x-Leap, s-Leap) but *tleap* is the simplest to use.

- Start *tleap*

```
% tleap -f leaprc.ff03
```

You will notice a number of library files have been loaded, they contain the parameters for standard amino acids, nucleic acid bases, ions and solvents.

- Load the original PDB structure. Type:

```
> kyn_xtal = loadPdb 1kyn-noligand.pdb
```

You will notice that *tleap* tells you it has added missing atoms and hydrogens. You may wonder why we used WHAT IF earlier if *tleap* will do this for you. The reason

is that *tLeap* is not as clever as WHAT IF – for example, it will always assume that histidines are the ϵ -tautomers (HIE).

- So: load the structure prepared previously by the WHAT IF server into *tLeap*. Type:

```
> kyn = loadPdb 1kyn-corrected.pdb
```

and compare the output.

If you find that you DO get warning messages, this is most likely due to problems with the ionisation states of the histidines or some little error when you edited the file in the “cleaning up” stage – if you can’t spot the problem, ask for help!

This time you will notice that *tLeap* doesn’t moan about missing atoms, etc.

- Now we need to fix the disulfide bonds. Type:

```
>bond kyn.44.SG kyn.60.SG
>bond kyn.137.SG kyn.202.SG
>bond kyn.167.SG kyn.181.SG
```

3. Adding ions

1. Check the charge of the structure:

```
> charge kyn
```

You should find that the charge on your system is 24, this will need to be neutralised...

2. Add the appropriate ions to make the system neutral and save a PDB file:

```
> addIons kyn Cl- 0
> savePdb kyn 1kyn-ions.pdb
```

In a separate terminal window (don’t quit *tLeap*), load 1kyn.ions.pdb into VMD and check the position of counterions. (Hint: rename “Cl-“)

If you have time at the end of the session you may wish to try other modifiers of addIons, for example how can we set a given ionic strength? You will need to check the AmberTools manual to look this up.

4. Adding solvent

As was discussed in the presentation, there are different ways of solvating your system. This will often depend on the size of the system you are using and what areas of the system you are interested in. Below are two different ways to solvate your system.

1. Solvent Cubic Box

```
> kynBox = copy kyn
> solvateBox kynBox TIP3PBOX 10
> savePdb kynBox 1kyn-box.pdb
```

2. Solvent Octahedral Box

```
> kynOct = copy kyn
> solvateOct kynOct TIP3PBOX 10
> savePdb kynOct 1kyn-oct.pdb
```

Compare the *tLeap* output for cubic and octahedral boxes. Note that in both the examples above, the “10” refers to the distance (in angstroms) from the edge of the protein to the edge of the box.

Load the pdb files you have just generated into VMD and compare the amount of water.

5. Saving files for simulation

The present structures are ready for simulation. We will be using the octahedral water box. Using *tLeap* you can now create and output the necessary files using:

```
> saveAmberParm kynOct 1kyn.top 1kyn.crd
```

These Amber topology (top) and coordinate (crd) files can be viewed with VMD and can also be used in some other simulation packages, for example NAMD. If you are not familiar with AMBER data files, have a look at **1kyn.top** and **1kyn.crd** using a standard text editor. The topology file contains all the invariant information about the system – atom names, list of all the bonds, angles, etc, plus all the force field parameters. The coordinate file contains just what will be adjusted through the course of the equilibration procedure – the actual positions of the atoms.