Braden Keiser
Complex simulation protocol for 7M1Z complexed with HMG-CoA and NADPH

2024/02/06

Software prerequisites:

Avogadro: Download Avogadro (sourceforge.net)
Rstudio and R: RStudio Desktop - Posit
Kate: Kate - Microsoft Apps
Pdb-tools (command-line, conda, ***your computer*** **and HPC**): pdb-tools | A swiss army knife for editing PDB files. (bonvinlab.org)
in conda: `conda install pdb-tools`
if conda doesn't work: `pip install pdb-tools`
Miniconda (your computer and HPC): below
VMD

Notes:

THIS PROTOCOL WAS INITIALLY DESIGNED TO RUN ON A TARA SUBMISSION SERVER AND WAS TAILORED TO A SPECIFIC HPC ENVIRONMENT. THE FOLLOWING METHOD IS FOR REFERENCE AND WILL LIKELY NEED SUBMISSION/DEPENDENCY MAINTENANCE WITHIN THE NESTED DIRECTORIES FOR RELEVANCE.

HOME = a placeholder name to censor identification content in the original protocol

Text in `Courier` *is emulation of code that can be run within a console terminal*

You can begin by running next_protein.sh (step 1) with the WT-complex_protonated.pdb and generating a 200ns simulation for the WT complex. It doesn't matter if your starting structure has ligand in it; the input parameterization step will remove everything and add in my pre-parameterized ligands. Moreover, **the WT-complex_protonated.pdb is already appropriately protonated, so step 0 is not needed**.

The only other issue I can see is getting RMSD, RMSF, and RoG. These require the use of R (bio3d) or python (MDAnalysis) to analyze the data locally. This can be done on the HPC, but it requires setting up your conda environment. Instructions for setting up miniconda on linux are here: How to Install Miniconda on Ubuntu 20.04 - VarHowto. Perform these steps in your own directory (i.e.,

```
cd ~/.
```

Then, begin the HowTo.

We can setup MDAnalysis in conda environment:

```
conda create -n mdsim
conda install -c conda-forge MDAnalysis
```

Setting up R in conda on the HOME HPC is much more difficult. We will use R on your local computer following the PBSA to generate some simple graphs of the PBSA data. You can also use excel for displaying some of the files (as mentioned in later section).

0. Structure Preparation

Ligand files have already been prepared on the HOME HPC in the HMG directory. All that needs to be done is that the mutated variant file be generated and protonated. Care must be taken to make sure the coordinates of the mutated PDB file match those with the WT PDB file obtained in the HMG directory. During input processing, the ligand coordinates will be deposited into the new structure based on their crystal structure geometries.

Not much is needed to do before beginning the Amber work; however, two important processes must be performed: protonation with PDB2PQR and manual protonation of the His741 (H381 in chain B). The HOME HPC HMG directory has a **new pt0**_protonation.sh that will protonate the structure at pH 6.5, but this does not protonate His741 like the Haines paper recommended for ideal active site interactions. Thus, the **overview of this section**:

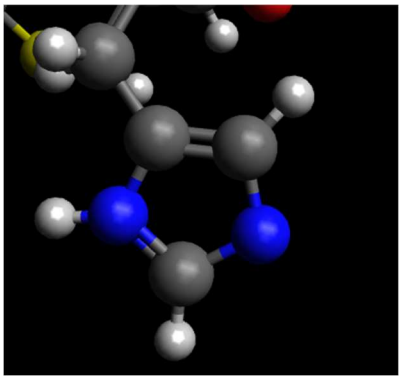
I. `bash pt0_protonation.sh [your_mutant].pdb`
II. protonate the histidine at ~381 or ~741 (depending on pdb numbering)
III. align new mutant to WT to make sure they are aligned

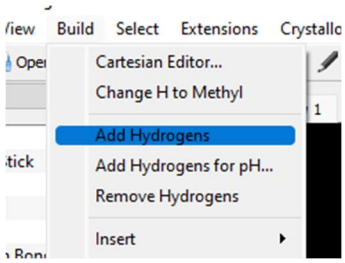
Once you have your mutant file, you can align it to the WT complex (WT-complex_protonated.pdb) in pymol with:

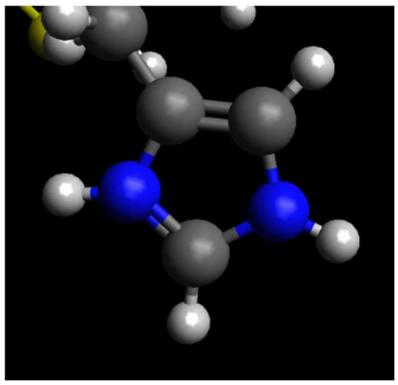
```
align [new_mutant], WT-complex_protonated
```

When both of them are open in the same window. After that, save the new protein and open it alone. Within chain B's ~381 region or at 740/741 in a renumbered format, there should be a '**QKGHMA'** motif, and once selected, these will be revealed to be in the active site.

Select '**GHM'** of the motif and on the right-hand side of pymol, press A → copy-to-object → new. Now, click the main protein so it becomes hidden (grey to black nameplate) and only the new object is shown as a small alpha helix curve. We can save this as a PDB 'h741.pdb' and open it in Avogadro.







Now, we save this PDB and open both this and the protonated mutant PDB with **Kate**.

Copy all atoms from **only** the HIS residue in the H741.pdb and copy it to the proper location in the mutant profile which will be around residue 740/741 if the residues have renumbered from 1, or it will be around 381 if each chain is respectively numbered in your PDB file. Copy over **all atoms of HIS in the mutant**. The motif will be three-letter AA code now: **GLN-LYS-GLY-HIS-MET-ALA**.

```
392  ATOM  11389  HA2 GLY   739      33.165  50.378  42.783  1.00  0.00           H
393  ATOM  11390  HA3 GLY   739      34.676  49.746  43.590  1.00  0.00           H
394  ATOM  11391  C   GLY   739      34.938  51.663  42.758  1.00  0.00           C
395  ATOM  11392  O   GLY   739      34.269  52.694  42.938  1.00  0.00           O
396  ATOM      8  N   HIP   740      36.285  51.673  42.439  1.00  0.00           N
397  ATOM      9  CA  HIP   740      37.002  52.897  42.220  1.00  0.00           C
398  ATOM     10  C   HIP   740      36.352  53.868  41.248  1.00  0.00           C
399  ATOM     11  O   HIP   740      36.230  55.046  41.593  1.00  0.00           O
400  ATOM     12  CB  HIP   740      38.490  52.581  41.949  1.00  0.00           C
401  ATOM     13  CG  HIP   740      39.459  53.724  42.048  1.00  0.00           C
402  ATOM     14  CD2 HIP   740      39.335  54.943  42.608  1.00  0.00           C
403  ATOM     15  ND1 HIP   740      40.624  53.617  41.375  1.00  0.00           N1+
404  ATOM     16  CE1 HIP   740      41.215  54.809  41.435  1.00  0.00           C
405  ATOM     17  NE2 HIP   740      40.448  55.644  42.240  1.00  0.00           N
406  ATOM     18  H   HIP   740      36.829  50.825  42.369  1.00  0.00           H
407  ATOM     19  HA  HIP   740      37.022  53.482  43.139  1.00  0.00           H
408  ATOM     20  HB2 HIP   740      38.890  51.794  42.588  1.00  0.00           H
409  ATOM     21  HB3 HIP   740      38.610  52.126  40.965  1.00  0.00           H
410  ATOM     22  HD1 HIP   740      40.810  52.814  40.791  1.00  0.00           H
411  ATOM     23  HD2 HIP   740      38.455  55.360  43.076  1.00  0.00           H
412  ATOM     24  HE1 HIP   740      42.137  55.111  40.961  1.00  0.00           H
```

Don't worry about residue name or number errors, here. We will first automatically renumber. In **conda**, navigate to the folder with the protein name stored and run pdb_reres:

```
pdb_reres -1 [mutant_protein_protonated].pdb > [mutant_protein_protonated]_renum.pdb
```

Now, your whole molecule should be renumbered from 1 to ~784. Copy this to:

```
cp  [mutant_protein_protonated]_renum.pdb [mutant_protein]_protonated.pdb
```

Of course, you will want to open this in pymol and check the area, making sure everything is correct.

Upload this to the HOME HPC in the main folder of the HMG directory. We can now begin.

1. **Folder setup**

   After logging into the HOME HPC, edit your .bashrc by typing:

   ```
   nano ~/.bashrc
   ```

   add the line to the bottom:

   ```
   export hmg=/path/to/your/hmg
   ```

   ctrl+x then press y and enter.

   Now, we can directly access the hmg directory by typing:

   ```
   cd $hmg
   ```

   Upload your finished pdb to this folder and use the next_protein.sh script to setup your folder:

   ```
   bash next_protein.sh [mutant]_protonated.pdb [folder_name]
   cd [folder_name]
   e.g.,
   bash next_protein.sh WT-complex_protonated.pdb WT
   cd WT
   ```

2. **Input parameterization**

Perform input parameterization by running the pt1 script **WITHOUT '.PDB'**:

```
bash pt1_input-parameterization_2024.sh [mutant]_protonated y
bash pt1_input-parameterization_2024.sh WT-complex_protonated y
```

3. **Minimization**

Run minimization via:

```
sbatch run_pt2_water-min.slurm
```

4. **Heat and water equilibration**

Run equilibration via:
(input mutant name is one you give to yourself – it won't be searching for a file name, but will name subsequent files with the prefix)

```
sbatch run_pt3-NVT_EQ.slurm [mutant] [test_num] [temp] [mpi]
```

e.g., `sbatch run_pt3-NVT_EQ.slurm WT 1 300 32`

5. **Production**

Production is initiated with:

```
sbatch run_pt4_production.slurm [folder_name] [test_num]
e.g,
sbatch run_pt4_production.slurm tested_wt 1
sbatch run_pt4_production.slurm WT-test 2
```

(These are the parent folders we created with next_protein.sh and are in the HMG directory)

6. **Get analysis:**

To get ligand information, **first we need to run the get_results.sh** script to create our results directory and perform some stripping of water and ions and renumbering residues for good analysis.

```
bash get_results.sh [test_num] [mutant]
```

To get **RMSD of ligand**, you can run the ligand_analysis.sh script following the completed of the production run and the running of get_results.sh:

```
bash ligand_analysis.sh [test_num] [mutant]
```

A location will be the output here:

/path/to/your/……./XXXXXX.tar.gz

Copy this and you can use SCP to download it directly to your local machine in another tab of MobaXTerm:

e.g.,

scp bkeiser@tara.nstda.or.th:/tarafs/...../XXXXX.tar.gz ./

tar xfvz XXXXX.tar.gz

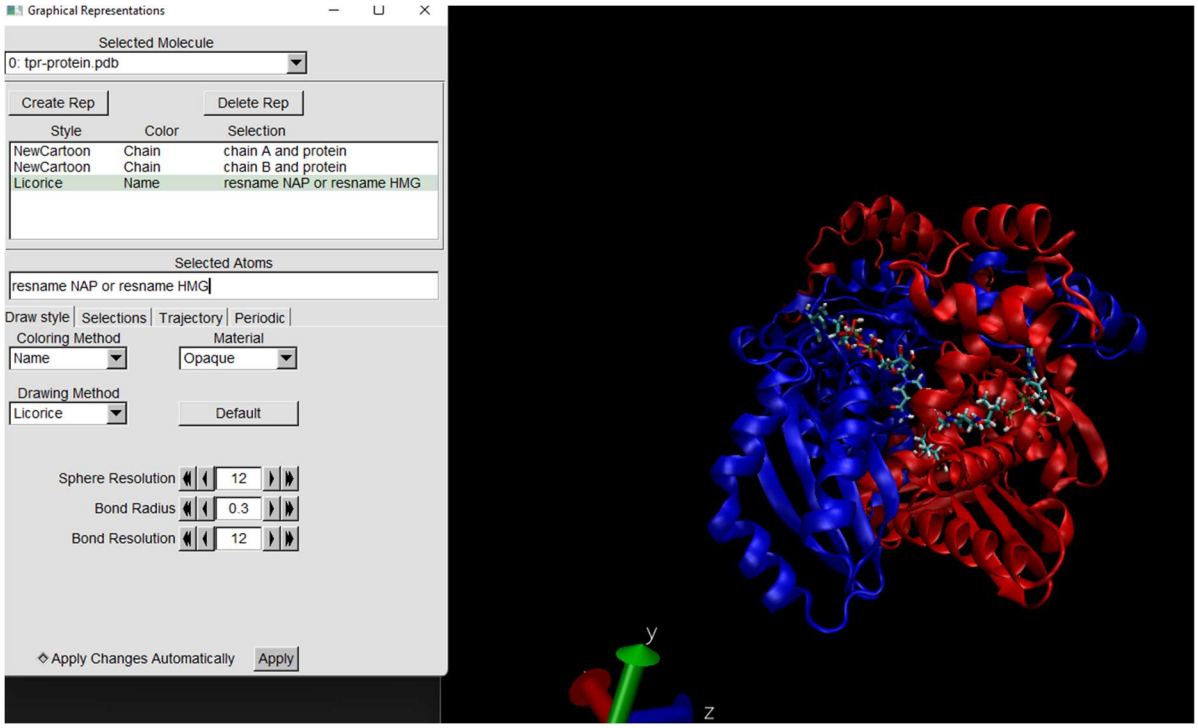
and then your ligand information will be available.

You can get the graphical information of the rmsd information by running the Rscript for ligand analysis (hmg_ligandanalysis.R) or you can view it in **Excel**, but first we need to extract only the relevant information into a new file using awk in **MobaXTerm**:

```
awk '{print $2}' hmg_rmsd.xvg > hmg_rmsd.csv
```

You will now have just a 1-column file. As Amber doesn't output comma-separated files, this is the easiest approach. The other column in the RMSD file is just for frames 1-[total_num_frames]. This is equivalent to the number of cells in Excel.

We can also **view the trajectory with VMD** by loading the tpr-protein.pdb and trj-small.xtc:



After reviewing the PBSA, we can make a distinction of where we want our 5ns to be for the PBSA calculation. Ideally, this should be the last 5ns of the simulation. But, this is not always doable if a mutation appears to disrupt ligand binding; therefore, the last 5ns of most stable RMSD can be used.

7. **PBSA analysis**

PBSA analysis is first prepped by:

```
bash prep_pt5_pbsa.sh [test_num] [mutant] [pbsa_start_frame] [pbsa_end_frame]

e.g.,

bash prep_pt5_pbsa.sh 1 WT 1500 2000
```

This will perform PBSA from 15ns-20ns. Often, we can extract every other frame, so only 250 frames are used here. This can be negated by removing “offset 2” in hmg/qm_protocol/cpptraj_inputs/extract.in

Following this, a new folder will appear in your mutant main folder: pbsa_t[test_number]. This is where the PBSA data will be stored.

In the main folder still, run the PBSA:

sbatch run_pt5_pbsa.sh [test_num] [mutant] [ligand_name]

e.g.,

sbatch run_pt5_pbsa.sh 1 WT HMG