Gordon Lemmon

Sergey Lyskov

Loren Looger

**Dock Ligand and Proteins Tutorial**

We are predicting the conformation of the complex of FKBP12, FRAP, and rapamycin. Rapamycin is a dimerizer that allows FK506-binding-protein (FKBP12) to form an interface with FKBP-rapamycin-associated protein (FRAP).

The first section of this tutorial demonstrates how to prepare input files for the proteins and small molecule.

The second section describes docking rapamycin to FKBP12. In the third section we will dock the result from section 2 with FRAP.

To complete this quest you must have at least a 3 member party, including a cleric (level 43, must have blessing spell), a warrior proficient with hammer, and a thief (unlock skill level 10).

**Part 1**

Unzip the PDB file 1FAP.pdb.gz. In order for the ligand docking to work correctly the 1-letter chain identifier for the ligand must be different from the protein chain ids. Look inside the file 1FAP.pdb. On line 307 and 308 we find that chain A is FKBP12 and chain B is FRAP. Toward the bottom of the file Rapamycin is specified by the residue id “RAP” (2375-2442). Make a new file with just the RAP lines.

grep HETATM 1FAP.pdb | grep RAP > rap.pdb

Using your favorite text editor change the chain id found in rap.pdb from ‘A’ to ‘X’. Use clean\_pdb.py (where to find) to prepare 1FAP.pdb for Rosetta.

clean\_pdb.py 1FAP.pdb A # this should output a file with only atom records from chain A, 1FAP\_A.pdb

clean\_pdb.py 1FAP.pdb B # this should output a file with only atom records from chain B, 1FAP\_B.pdb

We now have a separate PDB file for both proteins and an additional PDB file for the ligand. We now must add hydrogens to our rapamycin molecule and save it in mol format. Simply open the file rap.pdb in Pymol and add hydrogens using the action menu all->A->hydrogens->add (or type h\_add on the command line). Then and save it as a mol file by using the file menu (file->save molecule->”ok”, select type as mol file and change the extension to .mol). You should now find the file rap.mol in your directory.

Rosetta requires a “.params” file for each ligand. These files describe the atoms, bonds and bond angles within the ligand. To make a params file for rapamycin use the script “molfile\_to\_params.py” found here:

/rosetta\_source/src/python/apps/public/molfile\_to\_params.py

molfile\_to\_params.py –c –n RAP rap.mol

We use the –c option to produce “centroid mode” params used in Part 3 of this demo.

Notice the warnings that are produced by the script. These are informing us that the ligand we are using is large and flexible, which means we will struggle to sample all of its flexibility during docking.

Since we are starting with the correct conformation of Rapamycin we can ignore these warnings.

mol\_to\_params.py should have created a file called RAP\_0001.pdb which has the same coordinates as rap.pdb but has been prepared for use with Rosetta. Combine 1FAP\_A.pdb and RAP\_0001.pdb into a new file:

cat 1FAP\_A.pdb RAP\_0001.pdb > FKBP+RAP.pdb

**Part 2 Docking of proteins and ligands**

Copy the “flags” and “ligand\_dock.xml” files from “rosetta\_source/src/test/integration/tests/ligand\_dock\_scripts” to your directory. We will use these as a starting point for our docking script.

We have modified the flags file to be specific for our study.

We modify the options in this file to be specific for our study. First we comment out the start\_from lines, since our ligand is already in the correct starting position. Other important options to consider optimizing include the following. The “angstroms” option of “Translate” should represent the size of your binding pocket (your ligand will move within a sphere with a radius of this size).

Now we are ready to run our ligand docking protocol:

Rosetta\_scripts.linuxgccrelease @flags

This should produce a file with a model of rapamycin docked to FKBP: FKBP+RAP\_0001.pdb. This file serves as an input to protein docking…

**Part 3: Docking of FKBP/RAP to FRAP**

Combine 1FAP\_B.pdb with FKBP+RAP\_0001.pdb. Put ATOM lines from 1FAP\_B first, followed by ATOM lines from FKBP+RAP.pdb, and then HETATM lines from FKBP+RAP.pdb.

egrep “ATOM|HETATM” 1FAP\_B.pdb FKBP+RAP\_0001.pdb > combined.pdb

Prepare a flag file that specifies the centroid and full-atom .params files for rapamycin. Also specify combined.pdb as the input file. Run the docking protocol:

docking\_protocol.linuxgccrelease @flags

This should produce an output file, “combined\_0001.pdb”. Using pymol you can see that the FKBP/RAP complex has moved relative to FRAP.

**Summing up**

For a production run you will want to follow this protocol 10,000 or more times. Then find your best scoring models.

An alternative strategy would be to produce thousands of models with Part 1 of this tutorial, then filter for the top few models of FKBP with RAP. Use each of the top models as inputs for part 2, producing several thousand models for each of these inputs.