

Quick-start guide to structure refinement/relaxation using Rosetta 3.1

This is one of a series of tutorials designed to get you started with protein structure relaxation using Rosetta 3.1. It was produced to accompany Kaufmann et. al. (2010) Biochemistry, and the latest version can be found at <http://meilerlab.org/>. The purpose of Rosetta's -relax mode is to lower the energy of a structure by making small adjustments to the backbone and sidechain torsion angles. In addition, it can be used to add full-atom side chains to a low-resolution Rosetta model produced in centroid mode. The algorithm is described in Simons et al. (1997) J Mol Biol 268, 209-225. It is assumed that you have installed the Rosetta suite from rosettacommons.org, and that you are comfortable working in Linux. You will also want to become familiar with the documentation that can be found in the /manual/ and /demos/ subdirectories, as well as the online Rosetta 3 User Manual, FAQ, and forums at <http://www.rosettacommons.org/tiki/>.

Steps to execute the relax run

1. You will start with an input PDBfile, with or without side chains. Any decoy produced by Rosetta will work as input, but structures downloaded from the PDB may need to be simplified significantly using perl scripts found in the /BioTools/ directory of the Rosetta installation. There is a sample file 1ubi.pdb in the /relax/ directory included with this tutorial. This file has no header, only a single protein chain with no missing residues, and contains no solvent or heteroatoms.
2. Create a flags file to specify the parameters Rosetta will use during this run, including the location and format of input and output files, and options that specify details of the algorithm. There is a sample flags file in the /relax/ directory included with this tutorial. It is worth reading the file in a text editor because it is well-commented and contains pointers to more documentation. The parameters of the run can be changed by editing this file. Specifically, the relax mode allows the optional use of a flag -relax:fast which is nearly as effective as the normal protocol and can be up to ten times faster.

3. Conduct the folding run:

```
relax.linuxgccrelease @flags > relax.log
```

Analysis and notes

1. Examine the output files generated during this relax run. This includes the logfile, the output pdb structure file and the scorefile. These are plain text files, so you can read them with a pager or text editor. The details of interpretation of the file formats can be found in the manual. The output structures can be evaluated by the Rosetta energy score, where the lowest energy indicates the best structure. From the users guide, here are the descriptions of the energy terms:

fa_atr:	Lennard-Jones attractive
fa_rep:	Lennard-Jones repulsive
fa_sol:	Lazaridis-Karplus solvation energy
fa_intra_rep:	Lennard-Jones repulsive between atoms in the same residue
pro_close:	proline ring closure energy
fa_pair:	statistics based pair term, favors salt bridges
hbond_sr_bb:	backbone-backbone hbonds close in primary sequence
hbond_lr_bb:	backbone-backbone hbonds distant in primary sequence
hbond_bb_sc:	sidechain-backbone hydrogen bond energy
hbond_sc:	sidechain-sidechain and sidechain-backbone hydrogen bond energy.
dslf_ss_dst:	distance score in current disulfide
dslf_cs_ang:	csangles score in current disulfide
dslf_ss_dih:	dihedral score in current disulfide
dslf_ca_dih:	ca dihedral score in current disulfide
rama:	ramachandran preferences
omega:	omega angles
fa_dun:	internal energy of sidechain rotamers as derived from Dunbrack's statistics.

p_aa_p:	probability of observing an amino acid, given its phi/psi energy method
declaration	
ref:	reference energy for each amino acid
total:	final score

2. Load the input and output structures into your favorite molecular visualization program (Chimera, PyMOL, etc) and compare them visually. You should be able to notice small changes in atomic positions, bond angles, etc., with significant changes especially in surface-exposed side chains.
3. You can edit the provided flags file to use the `-relax:fast` flag and compare the results with the first run. You can also include constraints for problematic structures, and change the number of refinements performed. There is another example of the relax mode in the Rosetta distribution under `/rosetta3_demos/ClassicRelax`. For a complete list of options that can be used in Rosetta's relax mode (there are many) use this command:
relax.linuxgccrelease -help | less