ProteinLigEnsemble protocol capture

The following section provides an example for using the double ensemble docking protocol. Rosetta can be obtained through www.rosettacommons.org

All files associated with this protocol capture is provided in the demos/protocol_capture /rosettaligand_ProtLigEnsemble/ directory of the Rosetta distribution. This protocol has been tested to work with Rosetta version 9820fea, released July 12, 2018. Examples commands for this protocol are numbered in the commands file of the protocol capture folder and referenced as (1), (2), (3)... etc.

Starting files

The raw starting files are a single target protein receptor structure in PDB format, and a series of ligands in SDF format. The receptor structure used in this example is neuropeptide Y1 receptor bound to the ligand UR-MK299 (PDB: 5ZBQ). This file can be found in /inputs/ as protein.pdb. The four congeneric ligands in /prep/aligned_ligands/ directory have been aligned by their core scaffold. The reasonable number of ligands depends on the number of protein variants considered as each run generates all possible pairs. We generated up to 30 models per docking run without issue though this number may change depending on your computational setup.

Ligand preparation

Ligand preparation can be performed in the same fashion as previously documented RosettaLigand docking procedures. Commands 1, 2, and 3 cover the process of generating conformations with the BCL conformer generator and creating Rosetta ligand param files. Any conformation generator can be used as long as it can produce ligand files in the required SDF or MOL format. Example ligand conformers have been created for you in /prep/conformers and the necessary param generation files are provided in /prep/make_params/

The final Rosetta input ligand files are provided in /prep/rosetta_inputs/. The ligands have been designated with the letters B,C,D,E though you are free to use any chain designation as long as they are different from each other and the protein receptor. The correspondence between published ligand designations and the Rosetta lettering is provided in ligands.list. The params file process is the same as those for RLE but you can skip adding SAR data to the param files as SAR data will be provided in a separate dedicated file.

Setting up the QSAR file

The QSAR file, an example of which is provided as inputs/qsar.txt, will provide protein-ligand pairs of interest to the ProtLigEnsemble mover. Each line is organized as a protein identifier, a ligand identifier, and an optional binding value. To provide a ligand binding value to a wildtype protein, enter:

WT B 0.17

where WT indicates wildtype, B is the single letter chain of the ligand, and 0.17 is the optional affinity value. Note that the affinity value can be any measure as long as they are self consistent. Rosetta assumes the lower values indicates a more favorable binding. This can be changed by setting a negative correlation weight in the -docking:ligand:ligand_ensemble option. To provide a binding value to a mutant protein, enter:

107 A B 7.5

where 107 A indicates a mutation at residue 107 to alanine, B is the single letter chain of the ligand, and 7.5 is the optional affinity value. This will cause Rosetta to generate a 107A mutant regardless of what the wildtype residue at position 107 is. Note that this numbering system must correspond to Rosetta pose

numbering, where the first residue is numbered 1, the second is numbered 2...and so on. For the time being, PLE is designed to work with single mutants. This QSAR file will be provided to Rosetta in the XML script as the qsar_file tag for the ProtLigEnsemble mover.

Input file organization

For PLE runs, it is preferred to combine all aligned ligand PDBs into a single PDB file. This is provided as ligands.pdb in the /inputs/ folder. Note that the params and conformers files are not combined, just the single ligand PDB inputs.

In addition to structural files, a RosettaScripts XML file and a Rosetta options file. The XML file describes the custom protocol to be used by Rosetta. Details of how to setup an XML file and the meaning of the individual tags can be found by searching the documentation website https://www.rosettacommons.org/docs/latest/. The example dock.xml provided uses the settings from the benchmark. Actual application use may require the user to alter these values according to biological context. The defined scoring function is based on the existing RosettaLigand scoring function, but may be substituted in the XML script. The provided options file defines Rosetta input and output directories along with a number of sampling parameters. A full options list is available on the documentation website. The ligand_ensemble option is necessary to use PLE; a weight of 0 can be used to run PLE without taking SAR data into consideration.

A few XML tags in the ProtLigEnsemble mover are newer features to this mover. The aforementioned qsar_file tag tells Rosetta where to find the QSAR file. The distance tag defines the radius, in angstroms, of the sphere around the ligand considered to be the binding pocket. All residues in this sphere are considered to be flexible. This tag replaces the LigandArea, InterfaceBuilder, and MoveMap tags RosettaLigand users may be familiar with. The ignore_correlation option tells PLE to avoid calculating the rank correlation until there are at least 4 protein-ligand pairs in the dataset. This is because PLE optimizes binding pairs in order of binding affinity. Considering the rank correlation with only a few protein-ligand pairs is not particularly useful. This option may be adjusted based on the number of ligands in your particular dataset.

Run command (4) to perform a single simulation and generate a set of PLE models. Each simulation will produce X models, where X is the number of protein-ligand pairs listed in the SAR file. These example output models are in the /outputs/ directory along with a score.sc scorefile.

Output and analysis

Individual protein-ligand predicted structures are labeled by a protein-ligand pair designation. Wildtype proteins and ligand combinations will be tagged as WT_B_1.pdb through WT_E_1.pdb where the 1 indicates the docking run it came from. Mutant receptor-ligand pairs will be tagged as 107_A_B_1.pdb through 107_A_E_1.pdb. The first two parts indicate the mutant residue number and the mutant residue identity respectively. These are followed by the ligand chain designation and the docking run number.

Structures with the same numeric label are based on the same docking simulation and have a common binding pose. The protein interface contacting each ligand are optimized independently. The score.sc file contains all score terms for each simulation across a single row. Generally, individual ligand interface scores are used to rank models, with a negative score indicating a better model. These ligand interface scores are listed as interface_delta_*, where * corresponds to the protein-ligand prefix tag seen in the PDB files. The values are appended at the end of each output PDB, and also in the scorefile for each protein-ligand pair. One suggestion is for the end user to examine the top ten percent of models for each pair.