RosettaLigandCholesterol Protocol Capture

Prediction (TransformMover and HighResMover)

This protocol capture describes the RosettaLigandCholesterol protocol. Rosetta can be obtained through www.rosettacommons.org. The following protocol has been tested with Rosetta version ad2739a. Here the example structure is the Serotonin 2B receptor crystallized in a complex with cholesterol (PDB: 4IB4). The complex was downloaded from https://opm.phar.umich.edu. All files are included in the Rosetta/main/demos/protocol_capture/rosetta_ligand_cholesterol

Input/

The following files are included in the **inputs/** directory to run the Prediction protocol:

4ib4.pdb	Serotonin 2B PDB downloaded from OPM
4ib4_A.pdb	Cleaned protein
4ib4_A_relax.pdb	Relaxed protein
4ib4_A_dock.pdb	Combined protein and cholesterol for docking input
4ib4_A_positions	Lipid accessible regions for global docking (https://rosie.graylab.jhu.edu/mp_lipid_acc)
4ib4_A.span[1]	Rosetta spanfile
CLR.params[2]	Rosetta params file for cholesterol
CLR.pdb	Cholesterol pdb file
CLR_conformers.pdb[3]	Cholesterol conformer library
CLR_conformers.sdf	Cholesterol conformer output from MOE
dock.sh	Rosetta run command for docking
dock.xml	RosettaScripts XML file for docking
relax.sh	Rosttea run command for relaxing the protein
relax_wo_clr.xml	RosettaScripts XML file for relaxing the protein

[1] Indicates protein segments that should be embedded in the membrane.
[2] Pre-computed information about the geometry and chemical features of the cholesterol.
[3] Necessary for ligand flexibility.

Model Preparation

Before docking in Rosetta, we must prepare the protein and cholesterol separately.

Protein

#1 The PDB file is cleaned and renumbered for use in Rosetta.

```
Rosetta/tools/protein_tools/scripts/clean_pdb.py 4ib4.pdb A
```

Output: 4ib4_A.pdb

#2 Generate a spanfile from the cleaned PDB to indicate the transmembrane spanning regions.

```
Rosetta/main/source/bin/span_from_pdb.linuxgccrease -in:file:s
4ib4_A.pdb
```

Output: 4ib4_A.span

#3 Refine the protein to correct local errors and optimize the membrane position. The protein is relaxed without cholesterol.

```
## relax_wo_clr.sh command
Rosetta/main/source/bin/rosetta_scripts.linuxgccrelease
-in:file:s 4ib4_A.pdb
-mp:setup:spanfiles 4ib4_A.span
-parser:protocol relax_wo_clr.xml
-relax:jump_move true
-packing:pack_missing_sidechains false
-mp:scoring:hbond true
-relax:constrain_relax_to_start_coords

## relax_wo_clr.xml XML
<ROSETTASCRIPTS>
 <SCOREFXNS>
  <ScoreFunction name="frank" weights="franklin2019"/>
 </SCOREFXNS>
 <TASKOPERATIONS>
  <InitializeFromCommandline name="commandline_init"/>
  <RestrictToRepacking name="restrict_to_repacking"/>
 </TASKOPERATIONS>
 <MOVERS>
  <AddMembraneMover name="add_memb"/>
  <MembranePositionFromTopologyMover name="init_pos"/>
  <FastRelax name="fastrelax" disable_design="True" scorefxn="frank"
task_operations="commandline_init,restrict_to_repacking"
repeats="3"/>
 </MOVERS>
 <PROTOCOLS>
  <Add mover="add_memb"/>
  <Add mover="init_pos"/>
  <Add mover="fastrelax"/>
 </PROTOCOLS>
</ROSETTASCRIPTS>
```

Output: 4ib4_A_relax.pdb

Lipid

#1 The SDF file is downloaded from PubChem and conformers are generated with MOE.
Output: CLR_conformers.sdf

#2 Generate Rosetta-readable params file for ligand.

```
ROSETTA/main/source/scripts/python/public/molfile_to_params.py -n
CLR -p CLR --conformers-in-one-file CLR_conformers.sdf
```

Output: CLR.params, CLR.pdb, CLR_conformers.pdb

Docking

#1 Calculate lipid accessible regions on the protein for global docking using the Rosie server:

- Reduce XYZ coordinates from per atom to center of mass of each residue (centroid) in output PDB
- Change the resname (example: ASP) to water (HOH) in output PDB
- Change the atoms (example: CA) to oxygen (O) in output PDB
- Change ATOM to HETATM in output PDB

Output: 4ib4_A_positions.pdb

#2 Prepare docking input file

- Combine **4ib4_A_relax.pdb** with **CLR.pdb**

Output: 4ib4_A_dock.pdb

#3 Dock cholesterol to the protein and generate 10 models

```
## dock.sh command
Rosetta/main/source/bin/rosetta_scripts.linuxgccrelease
-in:file:s 4ib4_A_dock.pdb
-mp:setup:spanfiles 4ib4_A.span
-in:file:extra_res_fa CLR.params
-packing:pack_missing_sidechains false
-parser:protocol dock.xml
-nstruct 10

## dock.xml XML
<ROSETTASCRIPTS>
 <SCOREFXNS>
  <ScoreFunction name="frank" weights="franklin2019"/>
 </SCOREFXNS>
 <SCORINGGRIDS ligand_chain="X" width="25" name="lipid">
  <LipidMemGrid grid_name="head" weight="0.8" mem_weight="2"
ligand_atom="O1" kbpot_file="chol_o_mem_z_smooth_energies"/>
  <LipidMemGrid grid_name="tail" weight="1.0" mem_weight="2"
ligand_atom="C25" kbpot_file="chol_c25_mem_z_smooth_energies"/>
  <ClassicGrid grid_name="classic" weight="0.00000001"/>
 </SCORINGGRIDS>
 <MOVERS>
  <StartFrom name="start" chain="X" >
   <PDB filename="4ib4_A_positions" atom_name="O"/>
  </StartFrom>
  <Transform name="transform" chain="X" box_size="6"
move_distance="0.2" angle="20" cycles="500" repeats="5"
temperature="5" grid_set="lipid"/>
 <MOVERS/>
</ROSETTASCRIPTS>
```

Output/

The following files are included in the **outputs/** directory:

The head and tail atoms of the lipid are optimized independently. The score.sc file contains the individual score terms labeled head_grid_X and tail_grid_X, respectively.

4ib4_A.000*.pdb	output files
score.sc	scorefile

Specificity Filter Calculation

After running the Prediction protocol protein-cholesterol complexes can be further analyzed with the *spec_score* script. *spec_score* is a python script used to predict the likelihood that an integral membrane protein-cholesterol interface is specific.

Input/

The following files are included in the **inputs/** directory to run the SpecificityFilter:

4ib4_A_pdbs/	Prediction protocol output pdbs
buried_area.csv	Names of pockets to investigate with their corresponding buried area (http://schuellerlab.org/dr_sasa/)
4ib4_A_consurf.txt	Consurf rate of evolution predictions (consurf.tau.ac.il)
4ib4_A_netsurfp.txt	Netsurfp lipid accessibility predictions (https://services.healthtech.dtu.dk/service.php?NetSurfP-3.0)
4ib4_A_lips.pdb	Rosetta lipid accessibility prediction (https://rosie.graylab.jhu.edu/mp_lipid_acc)

#1 Use spec_score command to generate the specificity.csv file

```
## python spec_score.py
-q inputs/buried_area.csv
-p inputs/4ib4_A_pdbs/
-c inputs/4ib4_A_consurf.txt
-l inputs/4ib4_A_lips.pdb
-n inputs/4ib4_A_netsurfp.txt
```

Output/

The following files are included in the **outputs/** directory:

The output is a csv file containing the Escore, Pscore and specificity score.

specificity.csv	output file