11 EMMSB 2024

Engenharia Computacional de Proteínas

Instrutores: Roberto Lins, Danilo Coêlho, Elton Chaves

Tutorial elaborado por Danilo Coêlho, Roberto Lins e Elton Chaves, inspirado no capítulo de livro ***Motif-Driven Design of Protein–Protein Interfaces por*** Daniel-Adriano Silva, Bruno E. Correia and Erik Procko (https://doi.org/10.1007/978-1-4939-3569-7_17), como parte da série de livros Methods in Molecular Biology.

Designing a novel estrogen receptor binding through the Motif Grafting method

Introduction

The estrogen receptors (ERs; ERα and ERβ) are intracellular receptors that are activated by binding to the hormone estrogen, regulating transcription driving growth, proliferation, and differentiation, among many cellular processes. An artificially designed protein that binds to a given ER can have many applications. Breast cancer targeted therapy, (as ERs are often overexpressed) and breast cancer diagnostics are examples.

In this tutorial, we describe a step-by-step workflow for the design of new proteins based on motif grafting and interacting interface design. The majority of the protocols described can easily be run on a single personal computer, though large clusters and supercomputers will increase sampling and help find better solutions.

It will be required the following software:

ROSETTA. The ROSETTA software suite includes algorithms for protein modeling and design. ROSETTA is free for academic users and can be downloaded from: https://www.rosettacommons.org/software .

Molecular Visualization. A molecular graphics-viewing program is required. VMD (Visual Molecular Dynamics) software is recommended, as it is part of previous tutorials.

It can be downloaded from: http://www.ks.uiuc.edu/Research/vmd/

Methods workflow

The workflow for computational interface design using motif grafting is comprised of the following steps (also illustrated in Figure 1):

1. Defining the binding motif for seeded interface design.
2. Preparing a scaffold database/library.
3. Matching for putative scaffolds (i.e., motif structural alignment/grafting).
4. Sequence design.
5. Selection and improvement of designs.

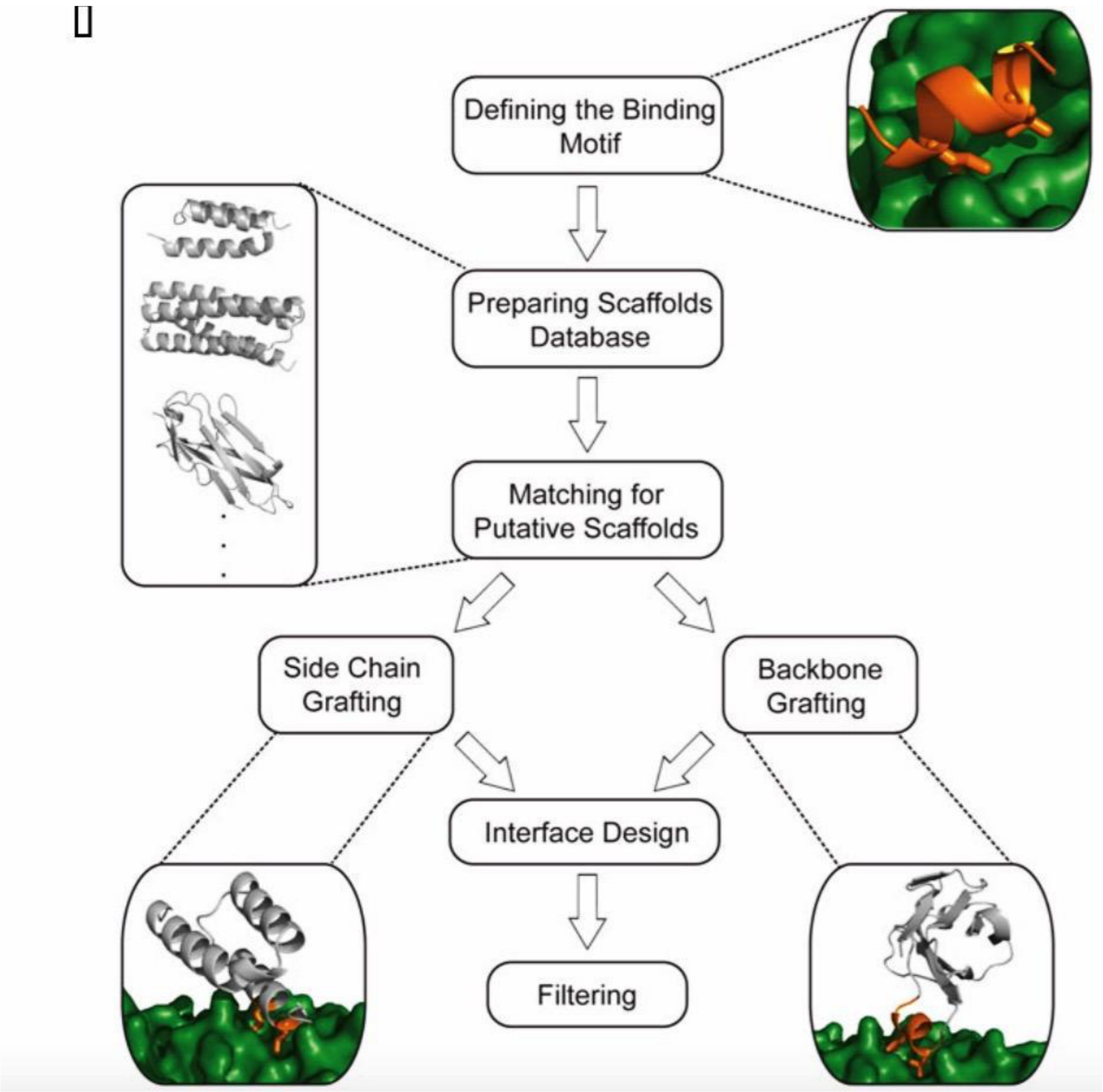
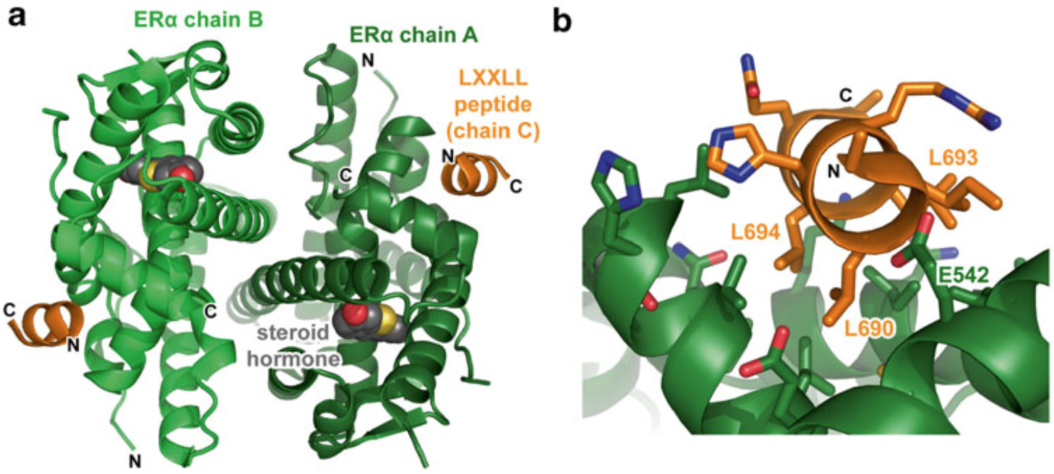

Defining the Binding Motif
Preparing Scaffolds Database
Matching for Putative Scaffolds
Side Chain Grafting
Backbone Grafting
Interface Design
Filtering

Figure 1. Scheme of the motif grafting and interface design workflow.

Defining the Binding Motif

For this tutorial we are going to design a protein binder for the estrogen receptor alpha (ERα), based on a known peptide interaction. The crystal structure of ERα has been solved with a bound helical peptide (PDB ID 1GWQ). This natural protein–peptide complex will provide an initial structural motif for seeded interface design.

The bound peptide provides the core of the interface, and the design process involves transplanting/grafting the motif into alternative protein scaffolds, followed by design of neighboring residues close to the target protein surface, creating an extended interface for **improved affinity and specificity**.



ERα is a steroid hormone-activated transcription factor that recruits coactivators to a target gene. The ERα-coactivator interaction is established through a helical motif that bears the signature sequence LXXLL (where L is leucine and X is any amino acid), with the leucine residues (hot spots) binding a hydrophobic cleft on the ERα surface.

In the following sections, we show how to graft the helical motif into a new protein scaffold. The assumptions guiding this design strategy are: (1) stabilization of the bound conformation of the LXXLL motif by embedding it within a stable scaffold reduces the entropic penalty of binding a flexible peptide, and (2) expanding the interfacial contact area can create new favorable interactions with the target.

Formatting input PDB files

First go to https://www.rcsb.org/ and download the PDB **1gwq**.

PDB files must be correctly formatted for compatibility with ROSETTA. All heteroatoms, including water molecules, should be removed (unless you have parameterized and included them into ROSETTA's force-field). In ROSETTA "TER" statements designate different proteins or chains in a complex, and therefore any "TER" statements within a single protein chain must be removed.

While these modifications can be made in a text editor, a quick but less accurate way to do that is using the script **extract_chain.bash** provided for you in the **ECP_TUTORIAL_MOTIFGRAFT** directory. First, copy the script to your current working directory:

```
cp  ECP_TUTORIAL_MOTIFGRAFT/extract_chain.bash .
```

Then, type the following commands (**do not copy it, type it yourself in your terminal**):

```
./extract_chain.bash 1gwq.pdb A protein >> 1gwq_clean.pdb
./extract_chain.bash 1gwq.pdb C protein >> 1gwq_clean.pdb
```

As the original crystal structure is a dimer of the complex, we are interested in only one structure of the ligand-binding domain of ERα (chain A) and in the respective structure of the helical LXXLL motif from the transcriptional coactivator TIF2 (chain C).

Now we have a clean and compatible pdb file and can proceed to the minimization step.

Energy minimization of crystallographic structures

It may be advantageous to perform energy minimization of the structures within the ROSETTA energy function prior to matching and design. Structures from experimental data often have residues with high (i.e., energetically unfavorable) energy due to minor clashes or “imperfections,” and these may be inappropriately designed by ROSETTA to alternative amino acids. This is especially problematic for backbone grafting and may lead to unnecessary sequence design of residues that should remain unchanged.

Energy minimization of input PDBs generally resolves this issue. However, it is important that structures do not drift too far during the minimization protocol; after all, the original PDB files are determined from real experimental data, whereas a minimized structure will only be as real as the energy function is accurate.

The starting structure can be minimized using the constrained fast relaxation protocol (**do not copy it, type it yourself in your terminal**):

```
path_to_Rosetta/main/source/bin/relax.linuxgccrelease -database
path_to_Rosetta/main/database/ -s 1gwq_clean.pdb
-relax:constrain_relax_to_start_coords -ex1 -ex2 -use_input_sc
-ignore_unrecognized_res -out:suffix _minimized
```

Important: For the examples presented here, command lines contain the *path_to_Rosetta*, which means the directory in which ROSETTA is installed on the user’s computer. Additionally, the application name ‘`relax.linuxgccrelease`’ must be changed to match user’s compiled version (*e.g.* relax.static.linuxgccrelease).

It will take a few minutes to complete the minimization. Now we have a clean and minimized complex structure **1gwq_clean_minimized_0001.pdb** .

Next, we will divide the structure into two new PDB files, referred to as the "**context.pdb**" and "**motif.pdb**". The "context" file contains the target structure (i.e., ERα; only chain A of PDB ID 1GWQ), while the "motif" file contains the LXXLL peptide (chain C of PDB ID 1GWQ). Run the following command (**do not copy it, type it yourself in your terminal**):

```
./extract_chain.bash 1gwq_clean_minimized_0001.pdb A protein >> context.pdb
```

```
./extract_chain.bash 1gwq_clean_minimized_0001.pdb C protein >> motif.pdb
```

Preparing a Scaffold library

To prepare a scaffold database that can be searched for a variety of structural motifs, we downloaded ~50 structures from the PDB (www.rcsb.org) based on the following four criteria: (1) crystal structures with high-resolution x-ray diffraction data (<2.5 Å), (2) the proteins had been reported to be expressable in *E. coli* (this simplifies later experimental characterization), (3) a single protein chain in the asymmetric unit, and (4) no bound ligands or modified residues.

Due to the short time for finishing the tutorial, the scaffold PDB files were previously formatted for ROSETTA and subjected to an energy minimization step in the same manner as we did before.

The scaffolds pdb files can be in the directory **scaffolds-library** in the **ECP_TUTORIAL_MOTIFGRAFT** directory.

Side chain grafting

The next step is to computationally scan the scaffold library for possible graft sites. If the motif and scaffold backbones superimpose with very low root mean squared deviation (RMSD < 0.2 Å), then only hot spot side chains need be transplanted from the motif to the corresponding positions in the matching site of the scaffold.

First, a list is generated containing all PDB files within the scaffold database. Type the following command (**do not copy it, type it yourself in your terminal**):

```
ls scaffolds-library/*.pdb > scaffolds.list
```

To run the protocol we are going to use the RosettaScripts framework. ROSETTA protocols are written in an XML-script format. The script is interpreted using the RosettaScripts parser. Using a simple analogy, RosettaScripts protocols are like cooking recipes; they first define the ingredients (energy functions, task operations, filters, and movers) and then outline the protocol by which these are combined. ***RosettaScripts is easy to use, even for novices with minimal programming experience.***

Open the XML script **MotifGraft_sc.xml** with a Text Editor and investigate its content. It looks like this:

```
<ROSETTASCRIPTS>
        <SCOREFXNS>
                <ScoreFunction name="ref15" weights="ref2015.wts" />
        </SCOREFXNS>
        <RESIDUE_SELECTORS>
        </RESIDUE_SELECTORS>
        <TASKOPERATIONS>
                <ProteinInterfaceDesign name="pido" repack_chain1="1"
repack_chain2="1" design_chain1="0" design_chain2="1"
interface_distance_cutoff="8.0"/>
                <OperateOnCertainResidues name="hotspot_repack" >
                        <ResiduePDBInfoHasLabel property="HOTSPOT"/>
                <RestrictToRepackingRLT/>
                </OperateOnCertainResidues>
        </TASKOPERATIONS>
        <FILTERS>
                <Ddg name="ddg" confidence="0"/>
                <BuriedUnsatHbonds name="unsat" confidence="0"/>
                <ShapeComplementarity name="Sc" confidence="0"/>
        </FILTERS>
        <MOVERS>
                <MotifGraft name="motif_grafting" context_structure="context.pdb"
motif_structure="motif.pdb" RMSD_tolerance="0.2" NC_points_RMSD_tolerance="0.15"
clash_score_cutoff="5" clash_test_residue="GLY" hotspots="3:7"
combinatory_fragment_size_delta="2:2" full_motif_bb_alignment="1"
graft_only_hotspots_by_replacement="1" revert_graft_to_native_sequence="1"/>
                <build_Ala_pose name="ala_pose" partner1="0" partner2="1"
interface_cutoff_distance="8.0" task_operations="hotspot_repack"/>
                <Prepack name="ppk" jump_number="0"/>
                <PackRotamersMover name="design"
task_operations="hotspot_repack,pido"/>
                <MinMover name="rb_min" bb="0" chi="1" jump="1"/>
        </MOVERS>
        <APPLY_TO_POSE>
        </APPLY_TO_POSE>
        <PROTOCOLS>
                <Add mover_name="motif_grafting"/>
                <Add mover_name ="ala_pose"/>
                <Add mover_name ="ppk"/>
                <Add mover_name ="design"/>
                <Add mover_name ="rb_min"/>
                <Add mover_name ="design"/>
                <Add filter_name="unsat"/>
                <Add filter_name ="ddg"/>
                <Add filter_name ="Sc"/>
        </PROTOCOLS>
        <OUTPUT />
</ROSETTASCRIPTS>
```

Within the XML file, the user may first specify which energy function to use from the ROSETTA database or reweight specific score terms.

Next, *Task Operations* define which residues can be altered. The **ProteinInterfaceDesign** task operation restricts design to residues of chain 2 (the scaffold) within 8 Å of the interface, while target residues within 8 Å of the interface may repack to alternative low-energy rotamers.

The second task operation, **RestrictToRepackingRLT**, prevents the two grafted hot spot leucines from being mutated in later design steps, though they can repack to alternative rotamers.

Movers dictate how the protein complex is manipulated, such as sequence design, side chain and backbone minimization, or rigid-body docking.

The protocol begins with the **MotifGraft** mover, which searches for alignments between the scaffold and motif that do not produce steric clashes with the target structure. The **MotifGraft** mover has many options. First, the names of the PDB files for the target (`context_structure`) and motif (`motif_ structure`) must be specified. The option `RMSD_tolerance` sets the maximum RMSD allowed between the motif and scaffold alignment. In this XML script, the RMSD tolerance was set to 0.2 Å (maximum recommended is ~0.5 Å). The option `NC_points_RMSD_tolerance` sets the maximum RMSD allowed between the N-/C-termini of the motif and scaffold graft site to 0.15 Å (maximum recommended 0.5 Å). Once the scaffold has been aligned, the configuration of the system must be checked for clashes. After it is grafted, the motif cannot clash with other parts of the scaffold. In addition, the orientation of the scaffold when aligned with the motif cannot clash with the target surface. Since residues can be designed to smaller amino acids in later steps, clashes are checked after first mutating the motif to small amino acids, such as alanine or glycine (using option `clash_test_residue="GLY"` in this XML script). All the atomic clashes are computed, and if the score is above the `clash_score_cutoff`, the graft fails and an alternative alignment in the scaffold is attempted (it is recommended to set the `clash_score_cutoff` at ≤ 5). The options `full_motif_bb_alignment="1"` and `graft_only_hotspots_by_replacement="1"` indicate that side chain grafting is being performed. Option `hotspots="3:7"` defines which positions in the motif PDB correspond to the two leucine hot spots of the LXXLL peptide. Additional hot spots are each separated by colons. Option `combinatory_fragment_size_delta="2:2"` indicates by how many amino acids the motif may be shortened at each terminus (N-terminus:C-terminus), i.e., whether the full motif must align (“0:0”) or only a partial fragment. Here, the algorithm will attempt to match the full-length motif, as well as each motif fragment shorter by up to two residues at one or both termini. The final option, `revert_graft_to_native_sequence="1"`, means that after the motif has been placed into the scaffold, all residues except for the hot spots are reverted back to their native identities. Therefore, only the two hot spot amino acids are effectively transferred as changes to the scaffold sequence.

After side chain grafting, the protocol continues by replacing scaffold side chains within 8 Å of the target with alanine using the **build_Ala_pose** mover. Task operations prevent the hot spots from changing. Side chains are now repacked with the **Prepack** mover. During this step, target protein residues that sterically clash with the scaffold have the opportunity to find alternative, non-clashing rotamers.

Next, the interface surrounding the grafted hot spots is designed using the **PackRotamersMover**. Task operations ensure that hot spot and target residues can only change rotamer conformations, whereas scaffold residues within 8 Å of the target surface are available for design.

Side chains and rigid-body orientations of the designed complex are then minimized with **MinMover**, followed by a second round of design.

To generate the decoys, execute the following command (**do not copy it, type it yourself in your terminal**):

```
path_to_Rosetta/main/source/bin/rosetta_scripts.linuxgccrelease
-database path_to_Rosetta/main/database/ -l scaffolds.list
-use_input_sc -nstruct 1 -parser:protocol MotifGraft_sc.xml
-out:suffix _sc
```

The option `-use_input_sc` means that rotamers in the input structure are included in the rotamer library, and option `-nstruct 1` means that the design script will be launched once per input scaffold.

The simulation will take a few minutes to finish. It is normal to see ERROR messages from Rosetta in the screen. It means that not all PDB used as scaffold match the RMSD/Atom Clash criteria.

Once the calculations have finished, you will see a few pdb outputs in your working directory named for instance ***1amx_A_sc_0001.pdb 1qkk_A_sc_0001.pdb 1yg2_A_sc_0001.pdb*** *etc.*

Selection of Designs

Unfortunately, the Rosetta energy scoring function correlates poorly with experimental protein-protein free energy calculations. Therefore, alternative methods should be used to increase the success rate between designs and experimental validation. We will select the most promising candidates using a machine learning ensemble method, as implemented into the PBEE (Protein Binding Energy Estimator) software (https://github.com/chavesejf/pbee).

Now run the PBEE software to calculate the absolute binding free energies for each designed protein to the Erα (**do not copy it, type it yourself in your terminal**):

```
pbee --ipdb *_A_sc_0001.pdb --partner1 A --partner2 B --frcmod_struct
```

The above command takes all pdb files as input via the **--ipdb** flag. These pdb files contain two chains labeled A and B, with chain A referring to the estrogen receptor and chain B referring to the binding protein. Therefore, the flags **--partner1** and **--partner2** receive the strings A and B, respectively. The **--frcmod_struct** flag tells the script to ignore warnings about low-quality structures. Once the calculation is complete, the outputs will be in the "outputs_pbee" directory.

To visualize the binding energies of the complexes, use **visualize_ddG.bash** script provided in the **ECP_TUTORIAL_MOTIFGRAFT** directory. Run the following command in the terminal (**do not copy it, type it yourself in your terminal**):

```
./visualize_ddG.bash
```

Visualize your designed protein versus the peptide and bound to the ERα using the software Visual Molecular Dynamics (VMD) (or other molecular visualization of your choice):

To use VMD, type the following in terminal (**do not copy it, type it yourself in your terminal**). Do not forget to change <your_best_model> to your PDB file name:

```
vmd -m 1gwq_clean_minimized_0001.pdb <your_best_model>.pdb
```

- Go to **Graphics ⇒ Representations**.
- Select the ***1gwq_clean_minimized_0001.pdb*** molecule
- In ***Selected Atom*** field write ***chain C*** and press Enter
- In ***Coloring Method*** choose ***ColorID*** and value **4** (Yellow)
- In ***Drawing Method*** choose ***NewCartoon***
- Now select the ***<your_best_model>.pdb*** molecule
- In ***Selected Atom*** field write ***all*** and press Enter
- In ***Coloring Method*** choose ***Chain***
- In ***Drawing Method*** choose ***NewCartoon***