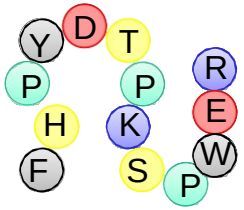
Protein-Protein Interface Design

Samuel Schmitz
Meiler Lab
Rosetta Workshop
May 2018

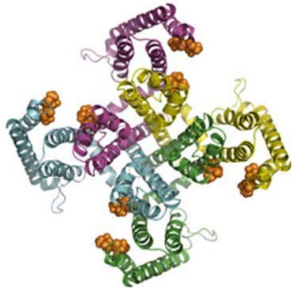
Protein Design is the Inverse Protein Folding Problem



Protein Folding Problem

Sequence ⇌ Structure

Inverse Folding Problem

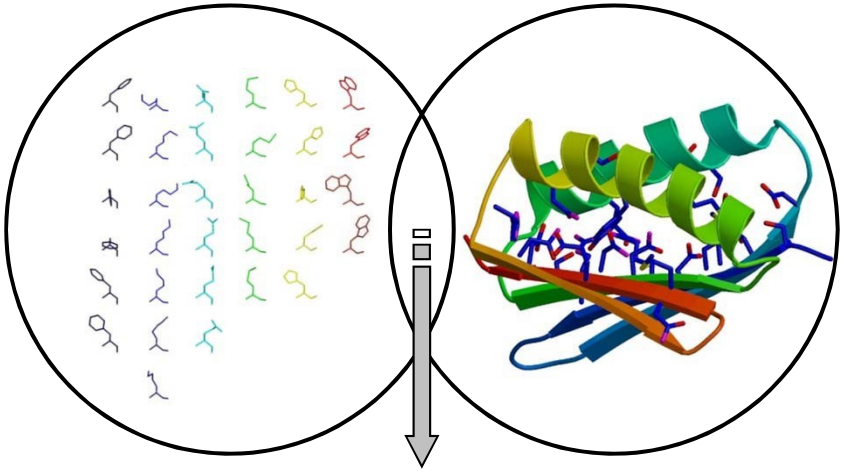


Given a protein fold – which primary sequence(s) can fold into it?

Protein Design Uses the Rosetta Energy Function and Local Rotamer Libraries

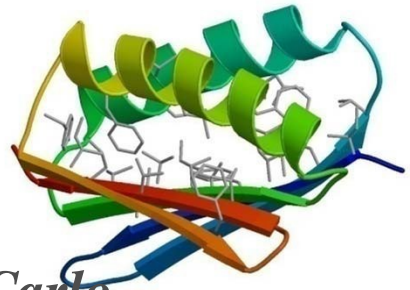
Local Rotamer Bias

Approximate interactions between sidechains using the distribution of sidechain conformations seen in known protein structures



Energy function

- *VDW interactions*
- *solvation*
- *hydrogen bonding potential*
- elec *interactions*
- *rotamer probability*



Simulated Annealing Monte Carlo optimization

The Residue File Guides Design

ALLAA	allow all 20 amino
ALLAAwc	allow all 20 amino(default)
ALLAAxc	allow all amino acids except cysteine
POLAR	allow only canonical polar amino acids
APOLAR	allow only canonical non-polar amino acids
NOTAA	disallow only the specified amino acids
PIKAA	allow only the specified amino acids
NATAA	allow only the native amino acid (repack)
NATRO	preserve input rotamer
EMPTY	disallow all canonical amino acids
NC <ResName>	allow the specific non canonical residue

You can also combine commands (see tutorial).

```
25 A POLAR NOTAA K
```

When on separate lines (first specify a range, then specific residues):

```
5-20 A ALLAA
15 A PIKAA Y
```

Residue 15 on chain A will only sample Y, whereas 5-14 and 16-20 will sample all

https://www.rosettacommons.org/docs/latest/rosetta_basics/file_types/resfiles

Basic Format of Residue Files

```
<Header> #instructions for all positions not specified in body
#The header can also use commands such as EX 1, EX 2,
#and USE_INPUT_SC to apply to all positions not specified below
START #keyword
<Body> #instructions for specific chains and identifiers
<PDBNUM> <CHAIN> <COMMANDS>
#Basic format for lines in body
* <CHAIN> <COMMANDS>
#used to specify a command for an entire chain
```

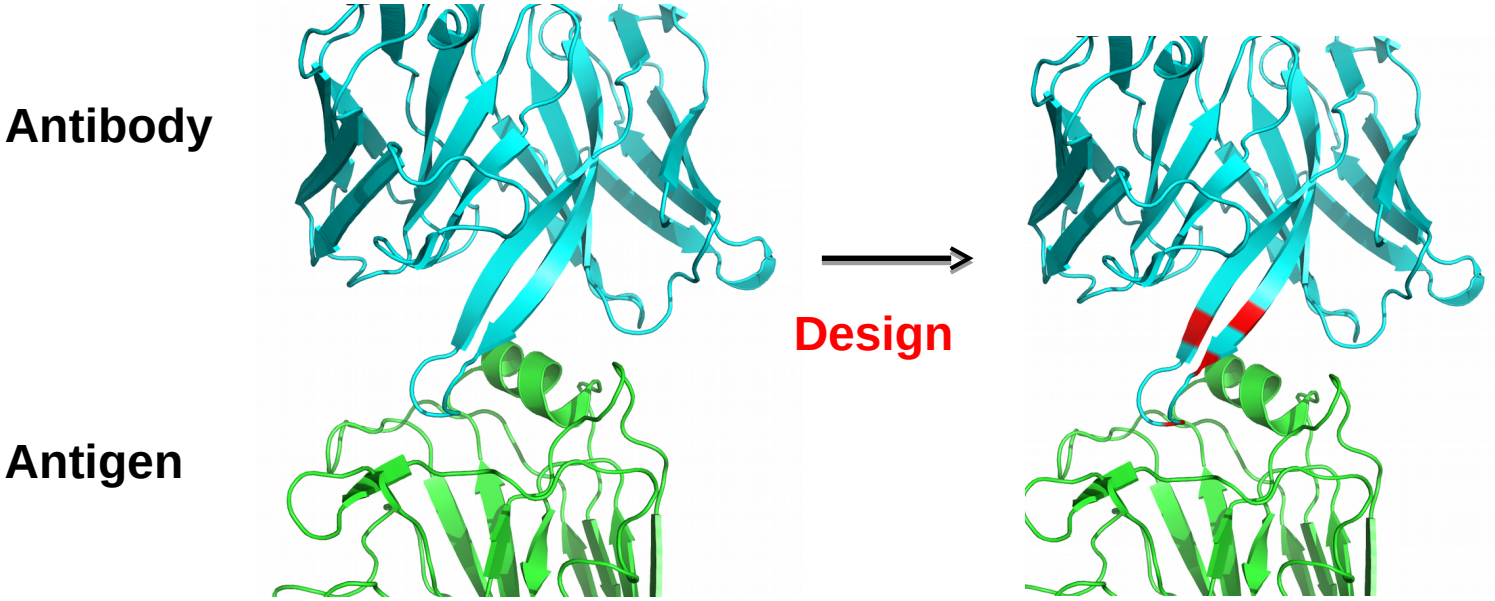
Tutorial Overview – Protein design in Rosetta

Antibody single-state design

Antibody single-state design

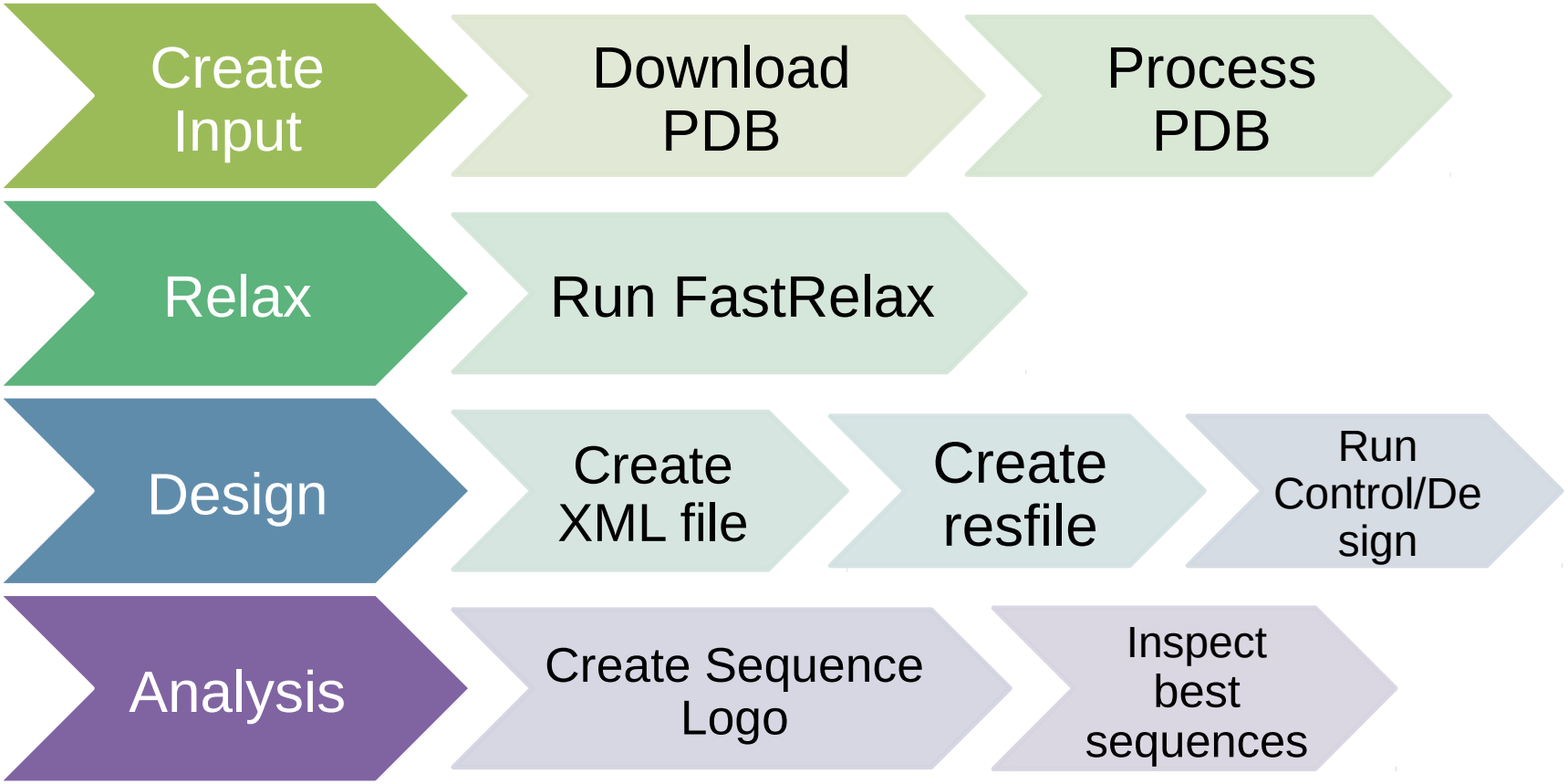
Also known as redesign, computational affinity maturation

Goal: take an existing antibody-antigen complex and optimize the antibody sequence for tighter binding



PDB ID 4yk4

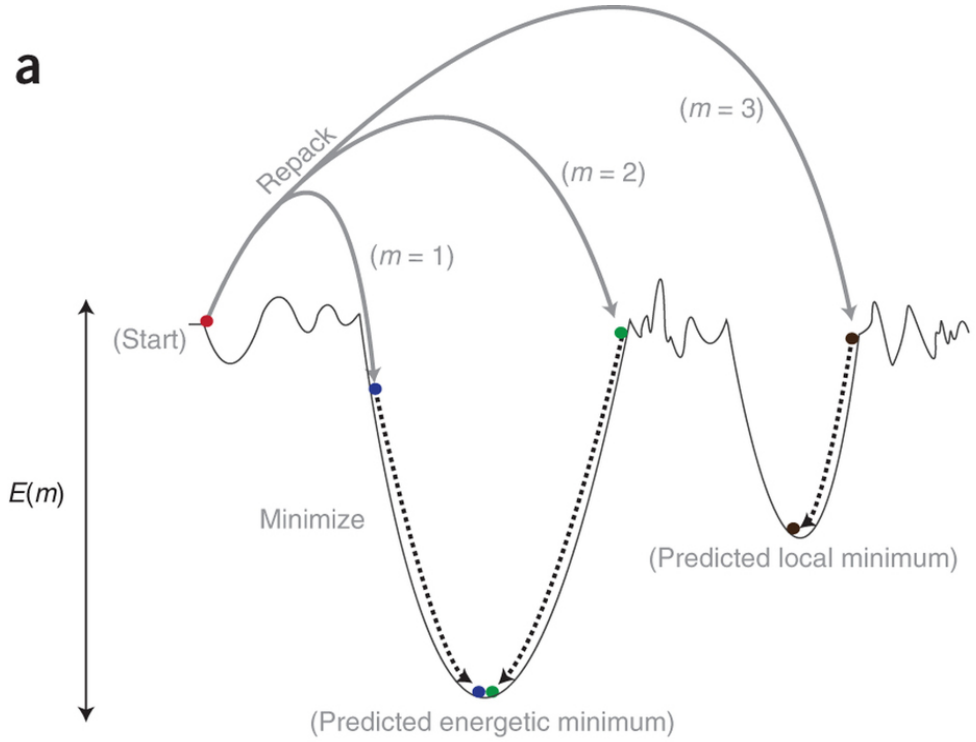
Single-state design protocol overview



FastRelax

FastRelax is designed to optimize the protein backbone/side chains to model at an energy minimum

Helps relieve clashes that may introduce artifacts into design



Combs, *et al*, Nat. Prot. 2013

FastRelax

input_files/relax.command:

```
~/rosetta_workshop/rosetta/main/source/bin/relax.default.l
inuxgccrelease @relax.options –s 4HKX_renum.pdb
```

input_files/relax.options:

```
-linmem_ig 100 # specify memory to store rotamer pair interactions
-use_input_sc      # Include rotamers from the input structure
-nstruct 1     # Generate 1 model
-relax:fast    # Do a small cycle number fast relax
-relax:constrain_relax_to_start_coords   # Add coordinate constraints
to backbone heavy atoms, based on the input structure.
-scorefile relax.fasc
```

Single state design

Please open single_state_design/input_files/design.xml

Where should you start looking?

```
<PROTOCOLS>
    Run the design protocol
       <Add mover="design" />

    Calculate interface metrics for the final sequence
       <Add mover="analyze" />
</PROTOCOLS>
```

Design movers

Design mover:

```
<PackRotamersMover name=“design” scorefxn=“REF2015”
task_operations=“ifcl,rrf” />
```

Task Operations:

```
Include rotamer options from the command line
<InitializeFromCommandline name=“ifcl” />

Design and repack residues based on resfile
<ReadResfile name=“rrf” filename=“4HKX.resfile”/>
```

Design control

Important to see how much improvement designs have over a nondesigned model

Please open input_files/design_control.xml

Design mover:

```
<PackRotamersMover name="design" scorefxn="REF2015"
task_operations="ifcl,rrf" />
```

Task Operations:

```
Include rotamer options from the command line
<InitializeFromCommandline name="ifcl" />

Design and repack residues based on resfile
<ReadResfile name="rrf" filename="4HKX_control.resfile" />
```

Making resfiles

Use the python script located in protein_design/scripts/define_interface.py

Calculates residues on each side of the interface using a side chain cutoff (default 5 A)

If any atom of a residue is within 5 A of any atom of a residue on the opposing chain – it's considered to be an interface residue

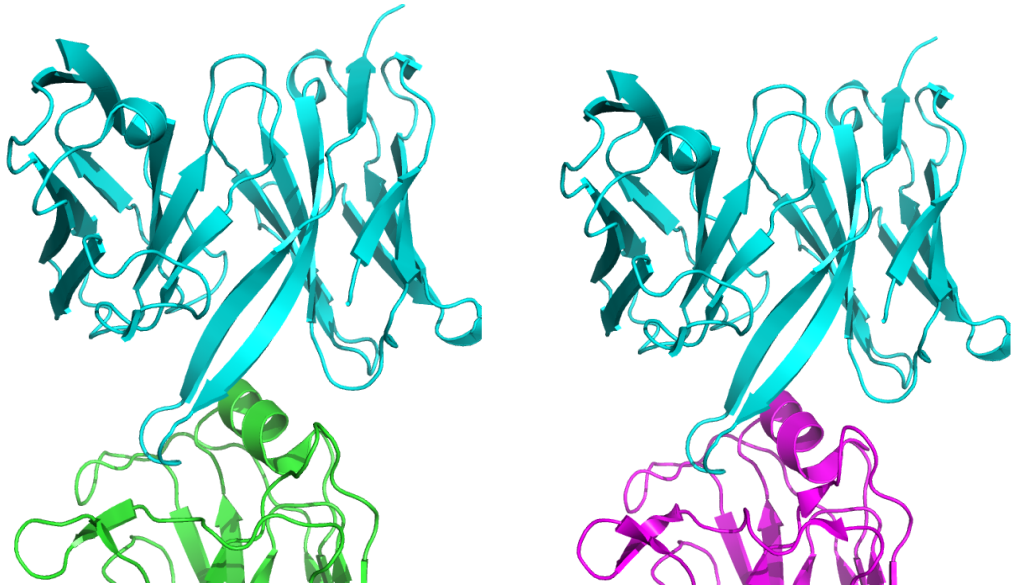
Making resfiles

`--side1=SIDE1`	the chains that make up one side of the interface (as a string, e.g. 'AB')
`--side2=SIDE2`	the chains that make up the other side of the interface (as a string, e.g. 'CD')
`--nearby_atom_cutoff=NBY_ATM_CUT`	SC distance cutoff to define a residue as part of the interface. If any SC atom from a residue on one side is within this cutoff of a residue on the other side it's considered to be in the interface. Default=5.0
`--output=OUTPUT`	Output name for resfile
`--design-side=DESIGN_SIDE`	Side of interface to design - either 1 or 2. Defaults to 1.
`--native`	Just repack the residues on the side flagged "design side”
`--repack`	Repack side of the interface not being designed

Antibody multistate design

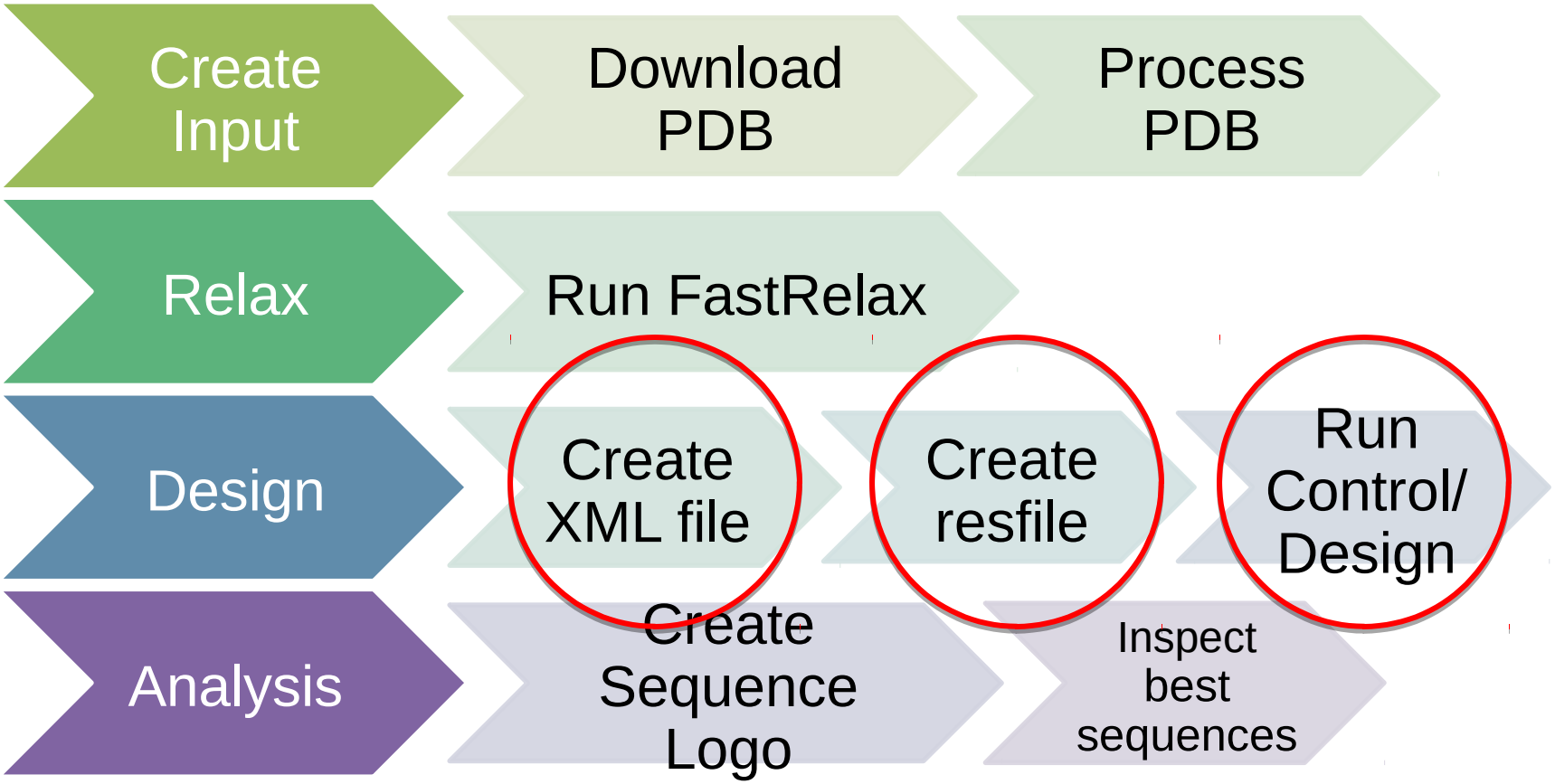
Multistate design: optimize a sequence for low energy in multiple conformations (states)

Redesign an antibody to recognize multiple targets

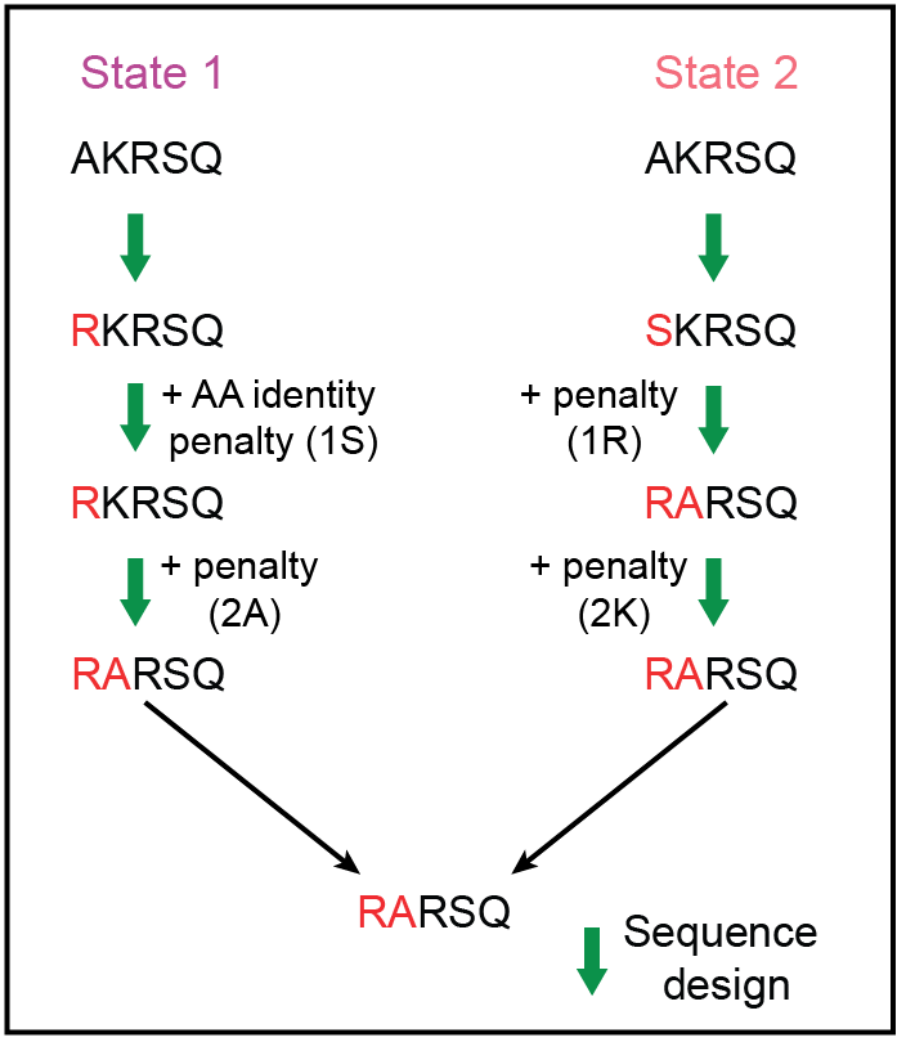


PDB ID 4yk4

Multistate design protocol overview



REstrained CONvergence in MSD (RECON)



Sevy, A. M., Jacobs, T. M., Crowe, J. E. & Meiler, J. *PLoS Comput. Biol.* **11,** e1004300 (2015).

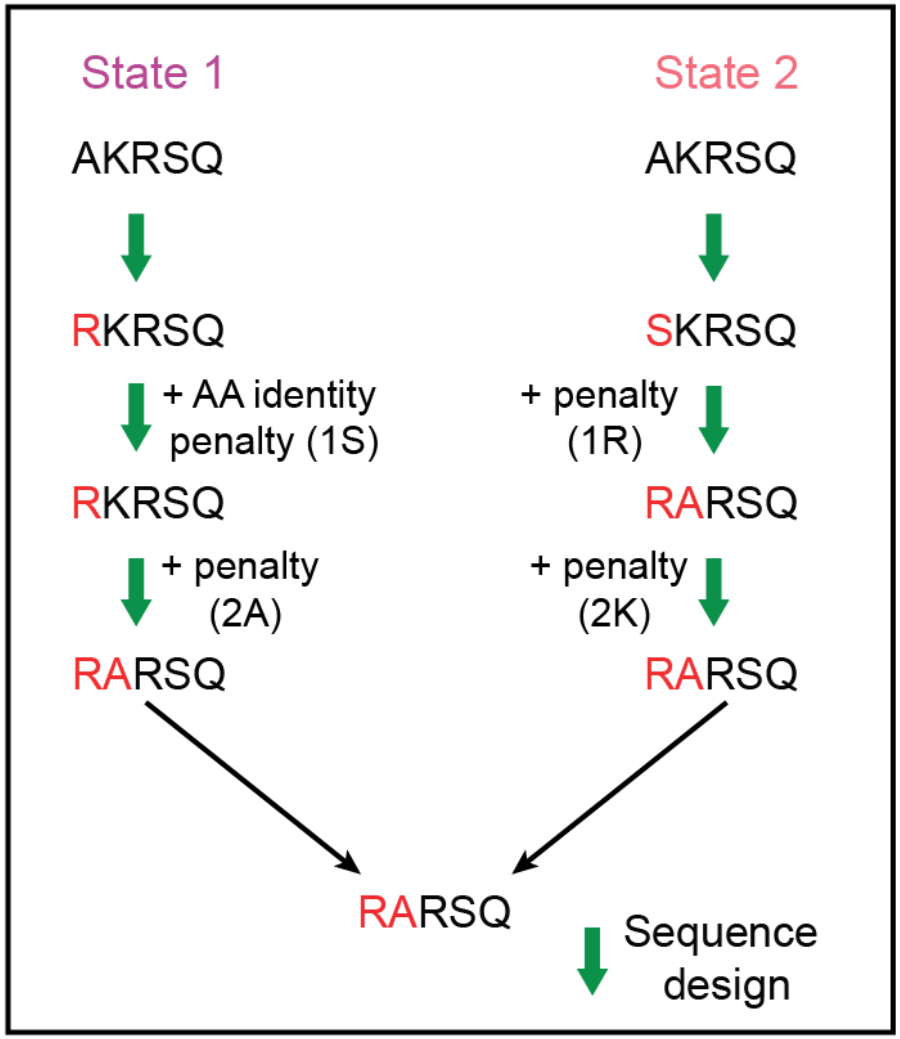
REstrained CONvergence in MSD (RECON)

MSDMover 1

MSDMover 2

MSDMover 3

FindConsensusSequence



Sevy, A. M., Jacobs, T. M., Crowe, J. E. & Meiler, J. *PLoS Comput. Biol.* **11,** e1004300 (2015).

Multistate design protocol

```
<PROTOCOLS>
    Run four rounds of design
    <Add mover=msd1 />
    <Add mover=msd2 />
    <Add mover=msd3 />
    <Add mover=msd4 />

    Find a consensus sequence for all states
        <Add mover=finish />

    Calculate interface metrics for the final sequence
    <Add mover=analyze />
</PROTOCOLS>
```

Multistate design movers

```
<PackRotamersMover name=design scorefxn=talaris_cst
task_operations=ifcl />

<MSDMover name=msd1 design_mover=design
constraint_weight=0.5 resfiles=4HKX.resfile,3UBQ.resfile />
<MSDMover name=msd2 design_mover=design
constraint_weight=1.0 resfiles=4HKX.resfile,3UBQ.resfile/>
<MSDMover name=msd3 design_mover=design
constraint_weight=1.5 resfiles=4HKX.resfile,3UBQ.resfile />
<MSDMover name=msd4 design_mover=design
constraint_weight=2.0 resfiles=4HKX.resfile,3UBQ.resfile />


<FindConsensusSequence name=finish scorefxn=talaris_cst
resfiles=4HKX.resfile,3UBQ.resfile />
```

Multistate design movers

```
<SCOREFXNS>
    <ref15_cst weights=ref2015.wts >
           <Reweight scoretype=res_type_constraint weight=1.0 />
    </ref15_cst>
</SCOREFXNS>
```

Have to reweight res_type_constraint term to allow for residue constraints!

If it’s not turned on protocol will run but will give a warning

Multistate design tips

You can use multiple resfiles – lets you be more flexible in which residues are being designed/repacked

Resfiles are matched to structure by order of input – **make sure these are in the same order!**

multistate_design.xml:

```
<MSDMover name=msd1 design_mover=design
constraint_weight=0.5 resfiles=4HKX.resfile,3UBQ.resfile />
```

multistate_design.options:

```
-s 4HKX_relax.pdb 3UBQ_relax.pdb
```

Cysteine design is not recommended

Make sure all resfiles have same number of residues being designed!

Analysis metrics

Total score: score of the entire complex

Interface score: score of residues that are at the interface

Binding energy (ddG, dG_separated): difference in energy between the bound and unbound partners

Binding density (dG_separated/dSASAx100): ddG divided by the buried surface area. Prevents a low binding energy by increasing buried surface area.

Analysis movers

```
<InterfaceAnalyzerMover name="analyze" scorefxn="REF2015"
packstat="0" pack_input="0" pack_separated="1"
fixedchains="H,L" />
```

packstat: activates packstat calculation; can be slow so it defaults to off

fixedchains: comma-delimited list of chain ids to define a group in the interface.

pack_separated: repack the exposed interfaces when calculating binding energy? Usually a good idea.

pack_input: prepack before separating chains when calculating binding energy? Useful if these are non-Rosetta inputs

Sequence logo

Useful to quickly see which residues are being designed, and what amino acids are being put there

Made by WebLogo application through scripts/design_analysis.py

http://weblogo.berkeley.edu/

Per Residue Scores

Use the python script located in protein_design/scripts/PerResidueEnergies.py

Plots relative (input pose – design pose) per residue energies.
Values smaller zero indicate improvements relative to the input pose

Red: Mutations

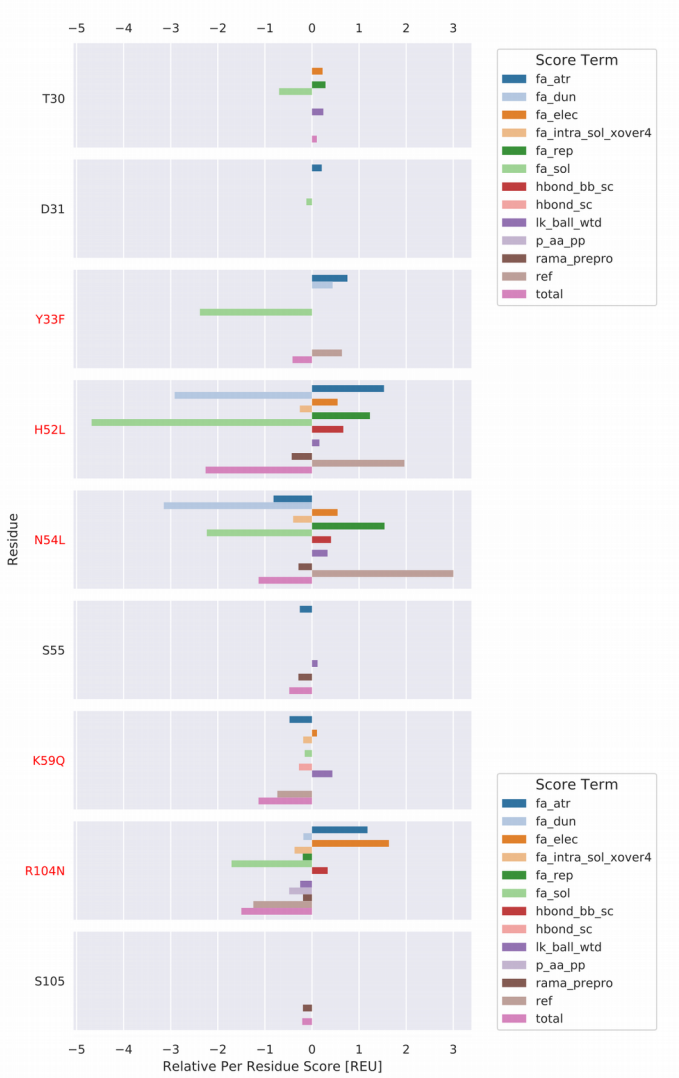
Per Residue Score Term Breakdown

Separate bars for each score term contribution

Understand why ROSETTA chose this mutation

Evaluate mutations or alter them based on chemistry knowledge and inspection of the structure

Red: Mutations



Meaning of scoring terms

The Rosetta All-Atom Energy Function for Macromolecular Modeling and Design
Alford et al (2017)

Table 1. Summary of Terms in REF15 for Proteins

term	description	weight	units	ref(s)
fa_atr	attractive energy between two atoms on different residues separated by a distance d	1.0	kcal/mol	5, 6
fa_rep	repulsive energy between two atoms on different residues separated by a distance d	0.55	kcal/mol	5, 6
fa_intra_rep	repulsive energy between two atoms on the same residue separated by a distance d	0.005	kcal/mol	5, 6
fa_sol	Gaussian exclusion implicit solvation energy between protein atoms in different residues	1.0	kcal/mol	36
lk_ball_wtd	orientation-dependent solvation of polar atoms assuming ideal water geometry	1.0	kcal/mol	50, 71
fa_intra_sol	Gaussian exclusion implicit solvation energy between protein atoms in the same residue	1.0	kcal/mol	36
fa_elec	energy of interaction between two nonbonded charged atoms separated by a distance d	1.0	kcal/mol	50
hbond_lr_bb	energy of short-range hydrogen bonds	1.0	kcal/mol	38, 49
hbond_sr_bb	energy of long-range hydrogen bonds	1.0	kcal/mol	38, 49
hbond_bb_sc	energy of backbone−side-chain hydrogen bonds	1.0	kcal/mol	38, 49
hbond_sc	energy of side-chain−side-chain hydrogen bonds	1.0	kcal/mol	38, 49
dslf_fa13	energy of disulfide bridges	1.25	kcal/mol	49
rama_prepro	probability of backbone ϕ, ψ angles given the amino acid type	(0.45 kcal/mol)/kT	kT	50, 51
p_aa_pp	probability of amino acid identity given backbone ϕ, ψ angles	(0.4 kcal/mol)/kT	kT	51
fa_dun	probability that a chosen rotamer is native-like given backbone ϕ, ψ angles	(0.7 kcal/mol)/kT	kT	52
omega	backbone-dependent penalty for cis ω dihedrals that deviate from 0° and trans ω dihedrals that deviate from 180°	(0.6 kcal/mol)/AU	AU[a]	72
pro_close	penalty for an open proline ring and proline ω bonding energy	(1.25 kcal/mol)/AU	AU	51
yhh_planarity	sinusoidal penalty for nonplanar tyrosine χ_3 dihedral angle	(0.625 kcal/mol)/AU	AU	49
ref	reference energies for amino acid types	(1.0 kcal/mol)/AU	AU	1, 51

[a]AU = arbitrary units.