Multi-state Design in Rosetta

Samuel Thompson

Nov. 12th, 2014

Timeline

- **11/12 (Today) Overview of multi-state design**
  - Multi-state design (MSD) in Rosetta
  - Fitness functions for optimization in MSD
  - How to design an patch with MSD
- **11/17 (Monday) Present your plan**
  - Expectations at the end of this presentation
  - We will distribute MSD scripts
- **11/19 (Wednesday) Check-in on progress**

Goals for Multi-state Design in PUBS

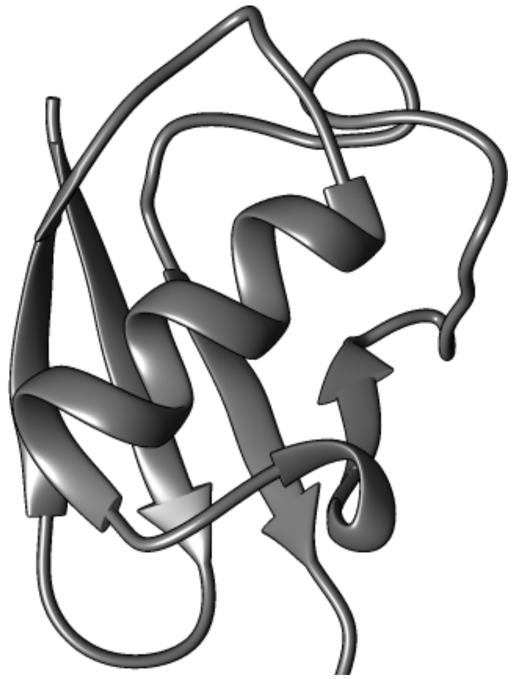
- Examine how interactions in multiple states shape protein sequences

- Model the interactions that might inform analysis of your experimental selection data

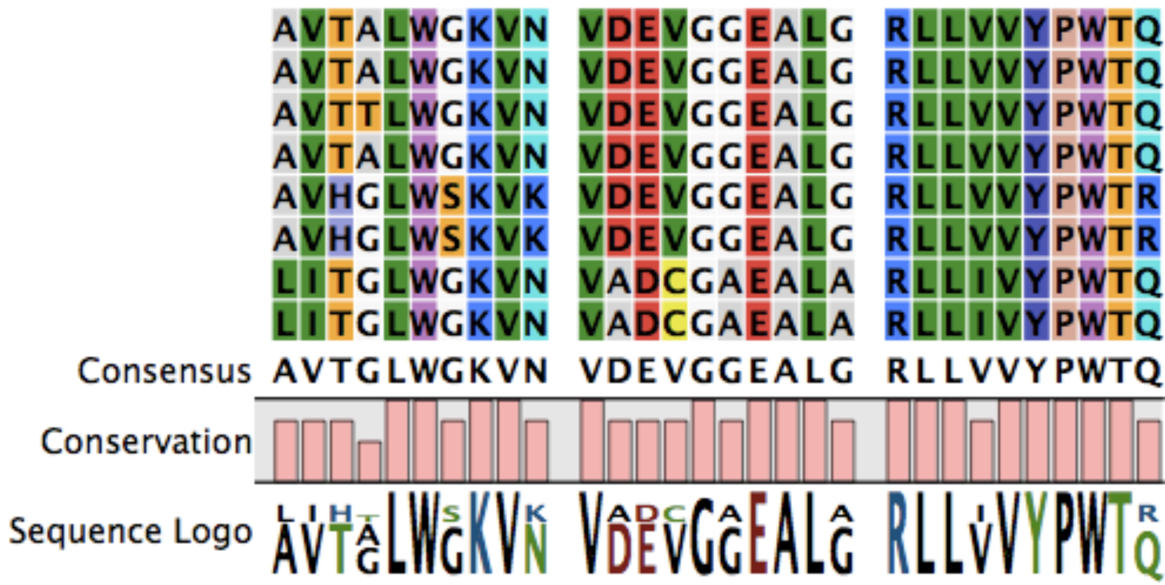
MULTI-STATE DESIGN IN ROSETTA

Rosetta Stabilizes a Protein Fold/ Conformation

- We want to be able to model function in terms of 1) a structure and 2) its biophysical energy



Input structure

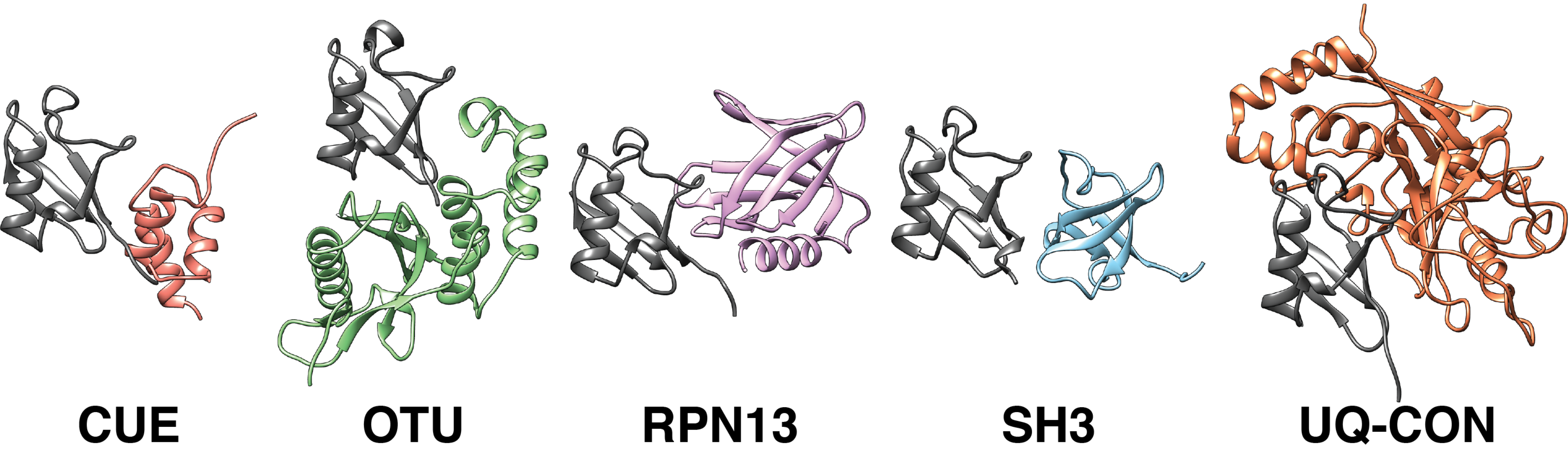


```
             AVTALWGKVN  VDEVGGEALG  RLLVVYPWTQ
             AVTALWGKVN  VDEVGGEALG  RLLVVYPWTQ
             AVTTLWGKVN  VDEVGGEALG  RLLVVYPWTQ
             AVTALWGKVN  VDEVGGEALG  RLLVVYPWTQ
             AVHGLWSKVK  VDEVGGEALG  RLLVVYPWTR
             AVHGLWSKVK  VDEVGGEALG  RLLVVYPWTR
             LITGLWGKVN  VADCGAEALA  RLLIVYPWTQ
             LITGLWGKVN  VADCGAEALA  RLLIVYPWTQ
Consensus    AVTGLWGKVN  VDEVGGEALG  RLLVVYPWTQ
```

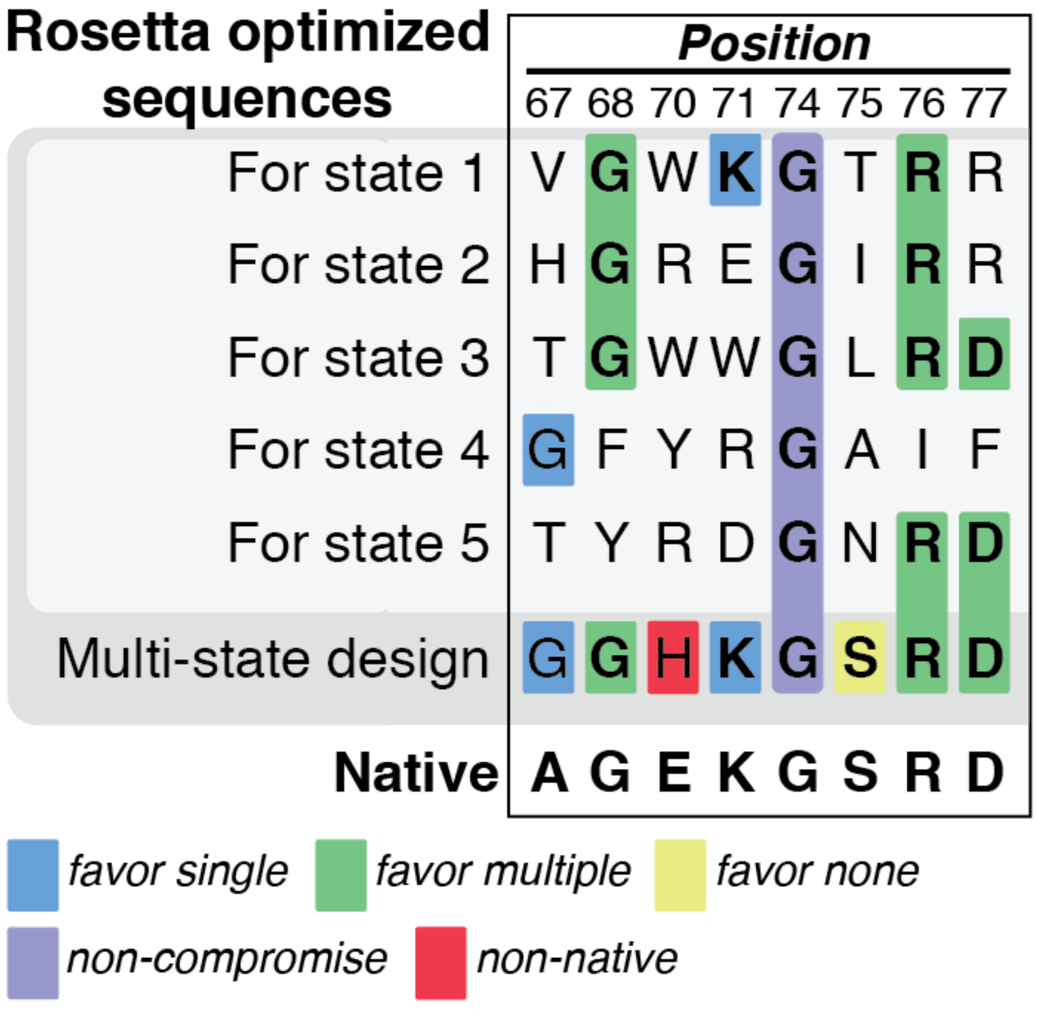
Tolerated sequence space

Rosetta Stabilizes a Protein Fold/ Conformation

- We want to be able to model function in terms of 1) a structure and 2) its biophysical energy
- What about proteins that adopt multiple states/conformations?



Hypothesis: sequences are an energetic compromise between states



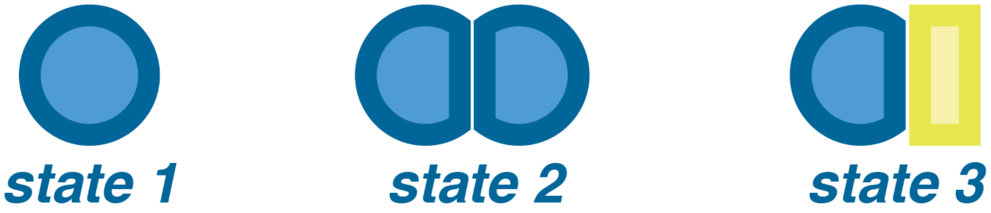
Rosetta optimized sequences	***Position*** 67	68	70	71	74	75	76	77
For state 1	V	🟩 **G**	W	🟦 **K**	🟪 **G**	T	🟩 **R**	R
For state 2	H	🟩 **G**	R	E	🟪 **G**	I	🟩 **R**	R
For state 3	T	🟩 **G**	W	W	🟪 **G**	L	🟩 **R**	🟩 **D**
For state 4	🟦 G	F	Y	R	🟪 **G**	A	I	F
For state 5	T	Y	R	D	🟪 **G**	N	🟩 **R**	🟩 **D**
Multi-state design	🟦 G	🟩 G	🟥 H	🟦 K	🟪 G	🟨 S	🟩 R	🟩 D
Native	**A**	**G**	**E**	**K**	**G**	**S**	**R**	**D**

🟦 *favor single* 🟩 *favor multiple* 🟨 *favor none*

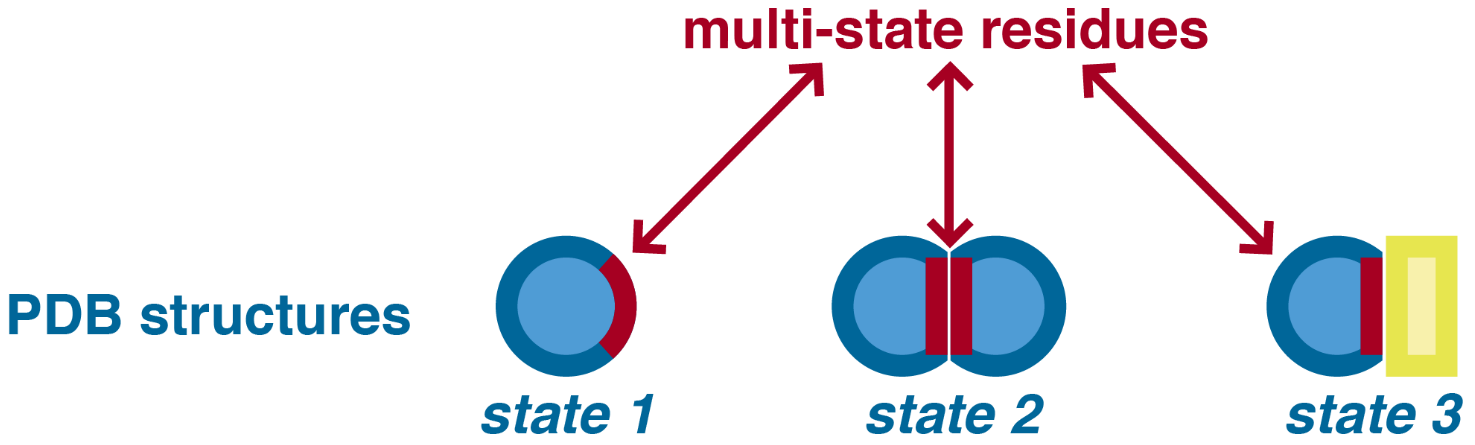
🟪 *non-compromise* 🟥 *non-native*

Multi-state Design in Rosetta

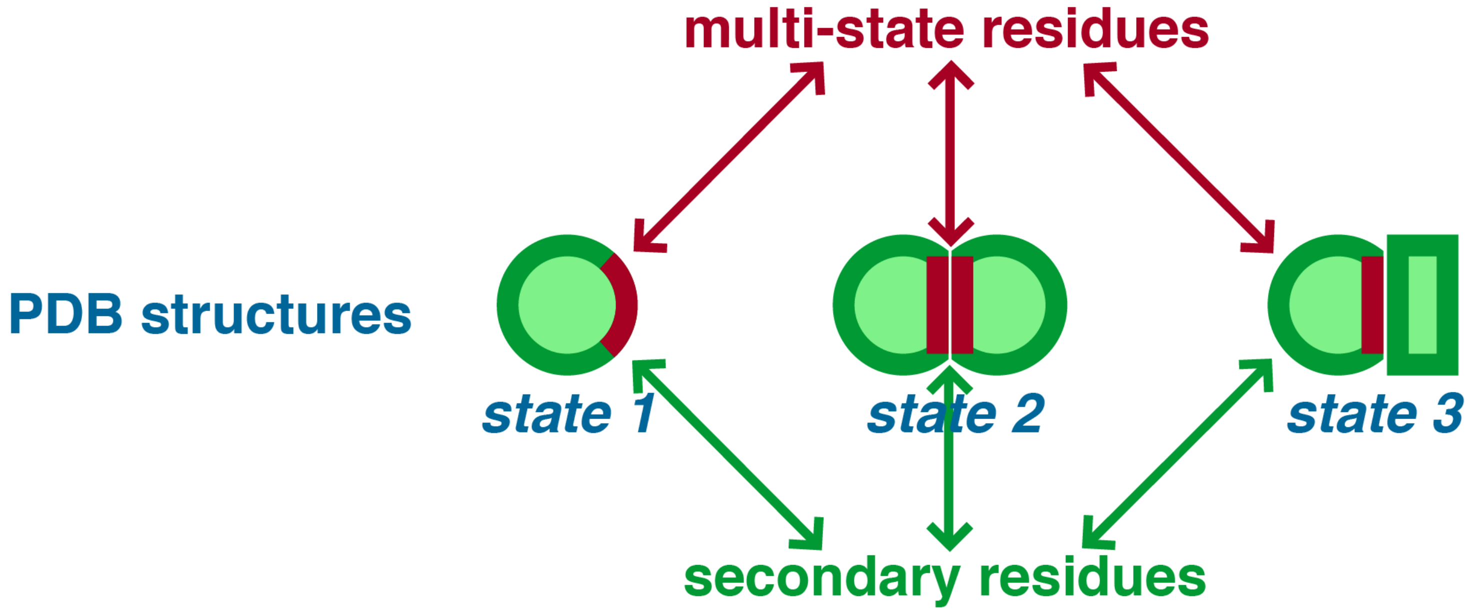
PDB structures



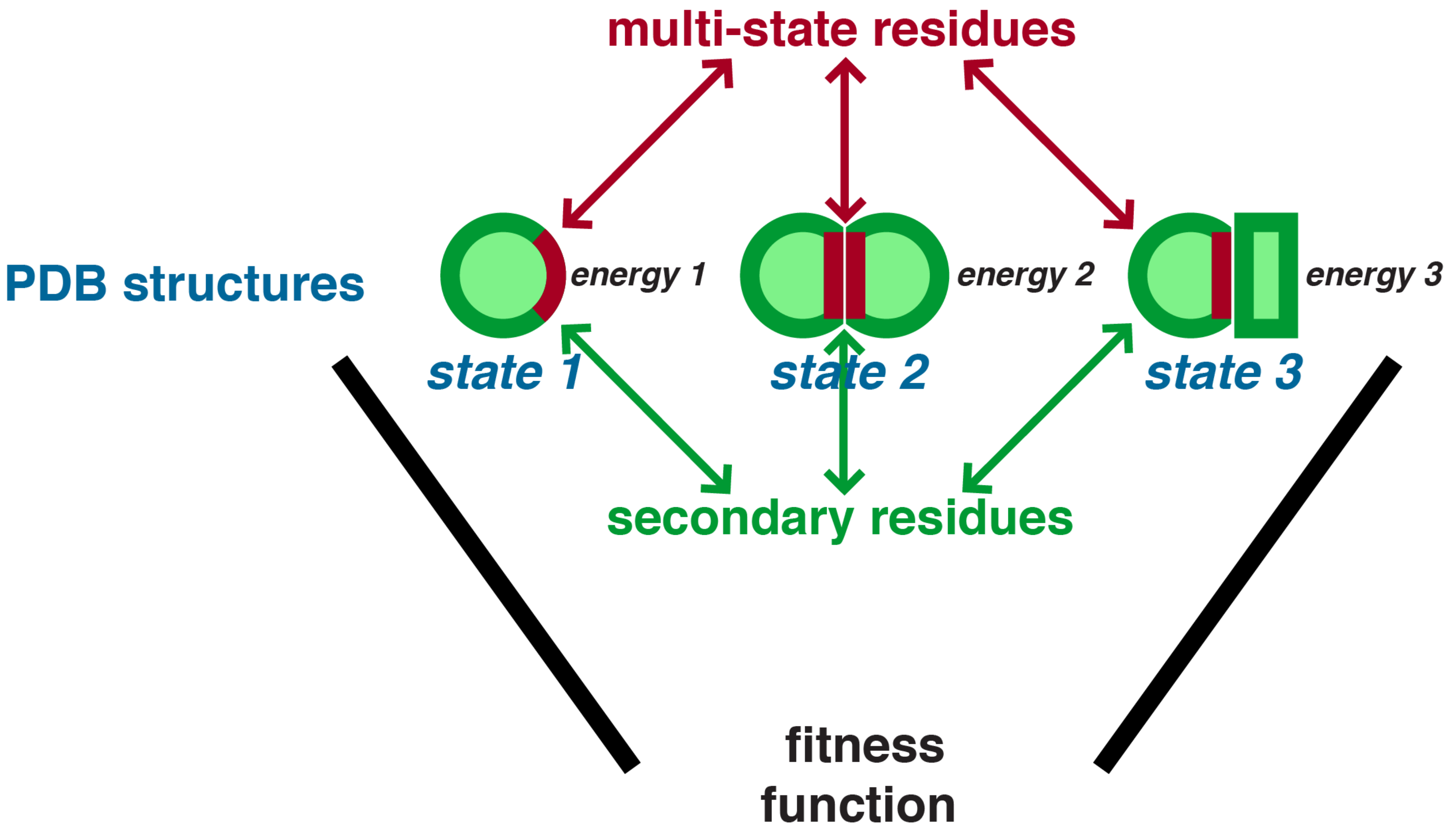
Multi-state Design in Rosetta



Multi-state Design in Rosetta



Multi-state Design in Rosetta



To make your life easier…

We are giving you a python script that generates all the necessary files.

To run this script you need to decide two things:

1) Which residues to design?
2) How to weight each state in the fitness function?

(not the same fitness from your experiments)

SELECTING PATCHES FOR DESIGN

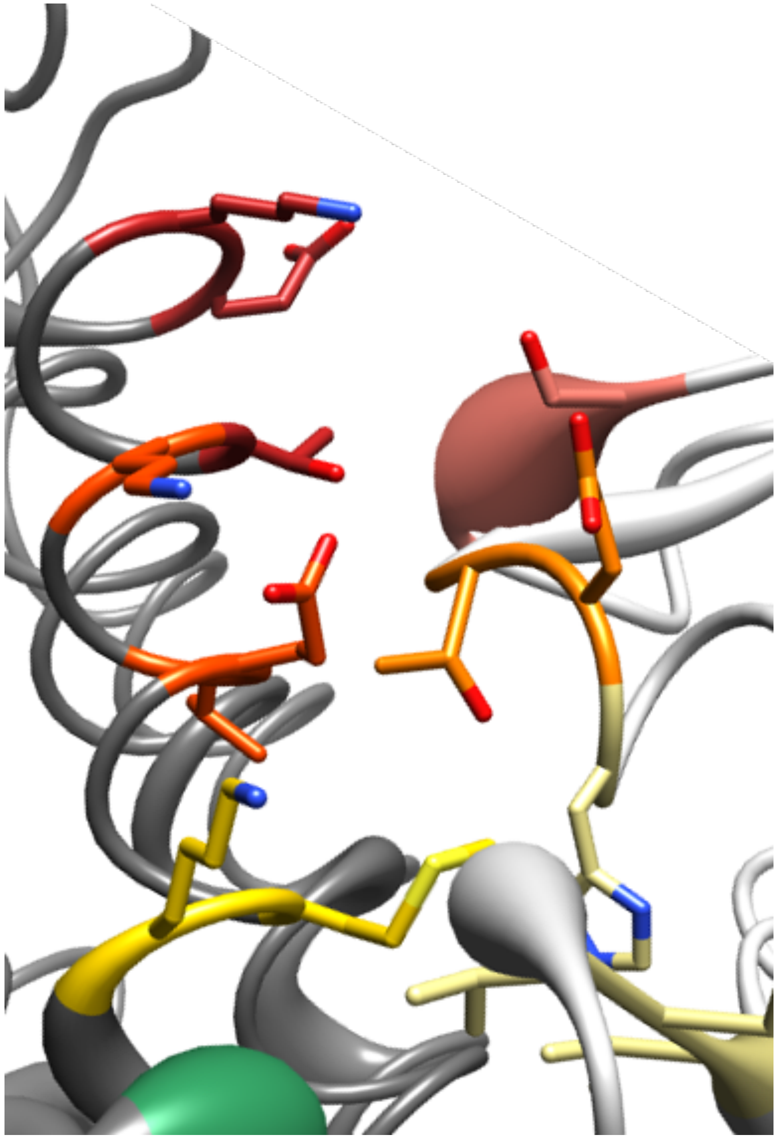
2-3 Minute Group Discussion

With the people near you:

- What residues in a protein with multiple states are likely to compromise?
- How will you identify them?

First task: picking design patches

- Patch: set of residues proximal to each other
- One simulation designs one patch
- Try to limit to 6-8 residues. Don't go above 9.
- Overlap your patches to reduce edge effects



Chimera Demo

- Selecting an interface in Chimera
  - Open the command line tool
  - $sel :.a & :.b z < 5
  - Selects residues in chain A that are 5Å from chain B
- Use the MatchMaker tool to align structures
  - Look for conformational changes
- Use attributes to paint information onto the structure
  - http://www.rbvi.ucsf.edu/chimera/docs/ContributedSoftware/defineattrib/defineattrib.html#attribdef
- Feel free to get more clever in picking your patch…

THE MULTI-STATE DESIGN FITNESS FUNCTIONS

Fitness functions

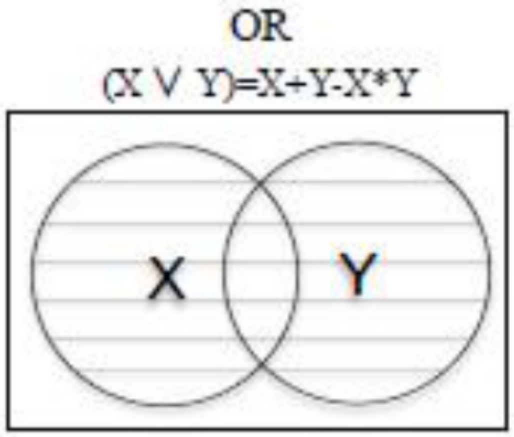
- The fitness function determines what we optimize during the simulation
- We want a fitness function that compromises for all modeled binding interactions
- Simple fitness function: Fitness = $E_1 + E_2 + E_3$…
  - E_1 is the Rosetta energy of state 1, and so on

Fitness functions

- The fitness function determines what we optimize during the simulation
- We want a fitness function that compromises for all modeled binding interactions
- Simple fitness function: Fitness = $E_1 + E_2 + E_3$…
  - E_1 is the Rosetta energy of state 1, and so on
- What are some of the potential problems with a simple fitness function?

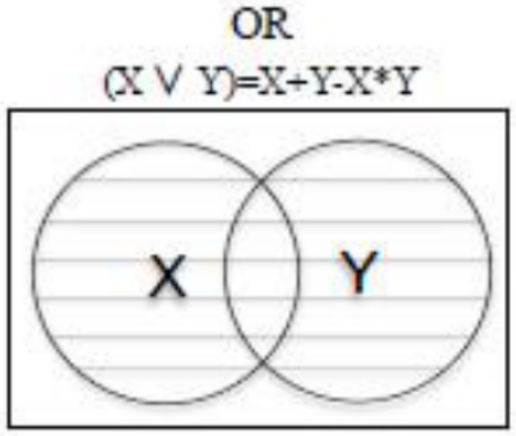
We can think of multi-state design in terms of logic

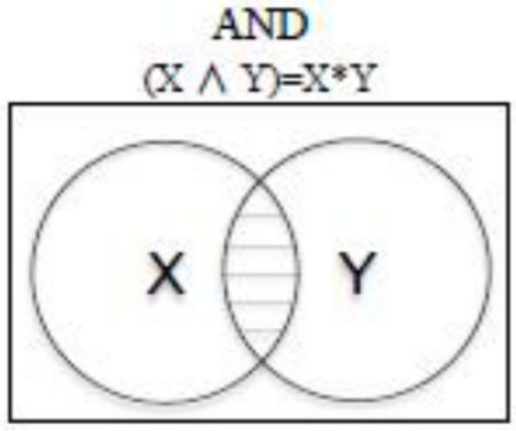
- A simple fitness function gives us "or" logic



We can think of multi-state design in terms of logic

- A simple fitness function gives us "or" logic





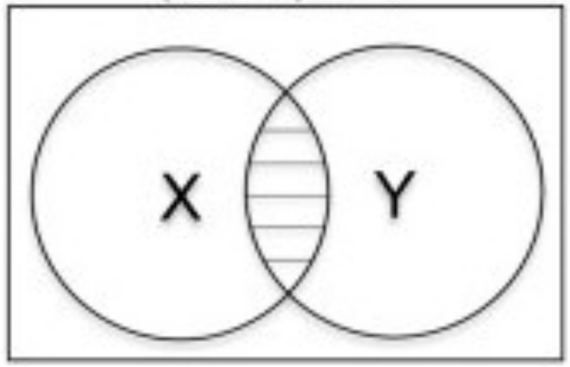
- What we want is "and" logic

To address some of these issues, we will use a fuzzy logic fitness function

- Boolean logic: 1 = True and 0 = False
- Fuzzy logic: 1 > more True > more False > 0
- "And" Fitness = F = $f_1*f_2*f_3*$…
  - Where f1 is the fitness of state 1, and so on…

AND

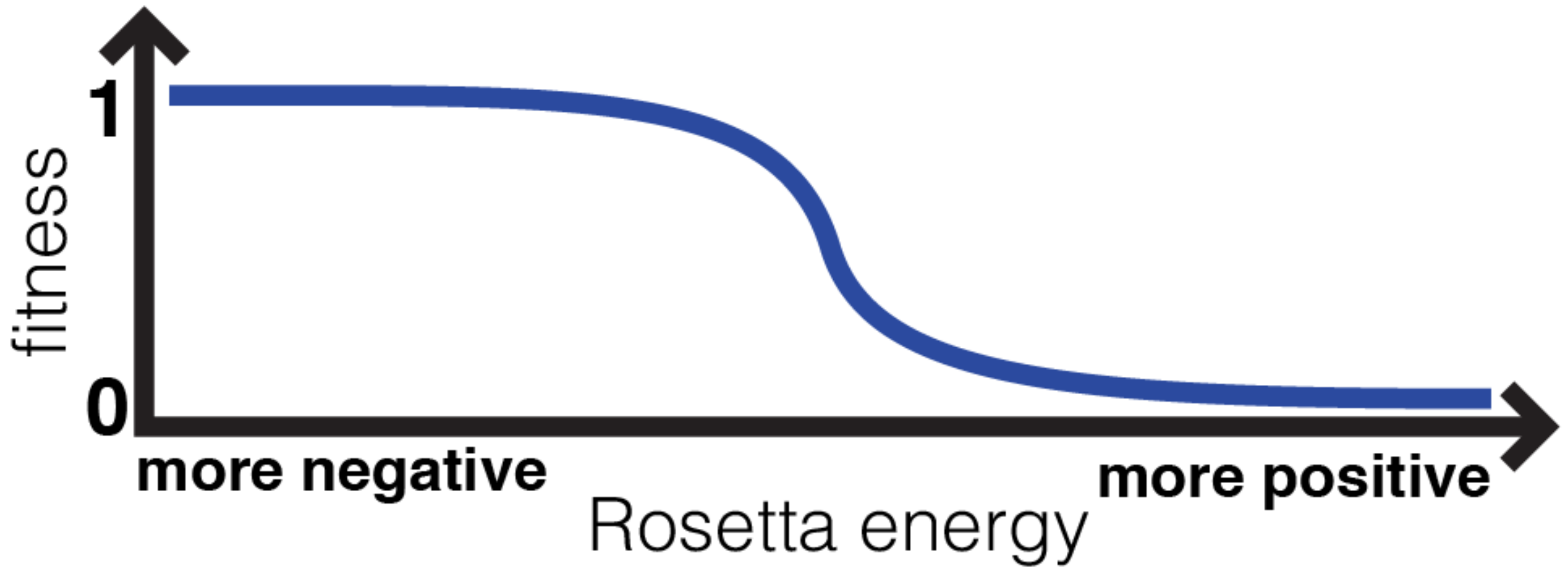
$(X \wedge Y)=X*Y$



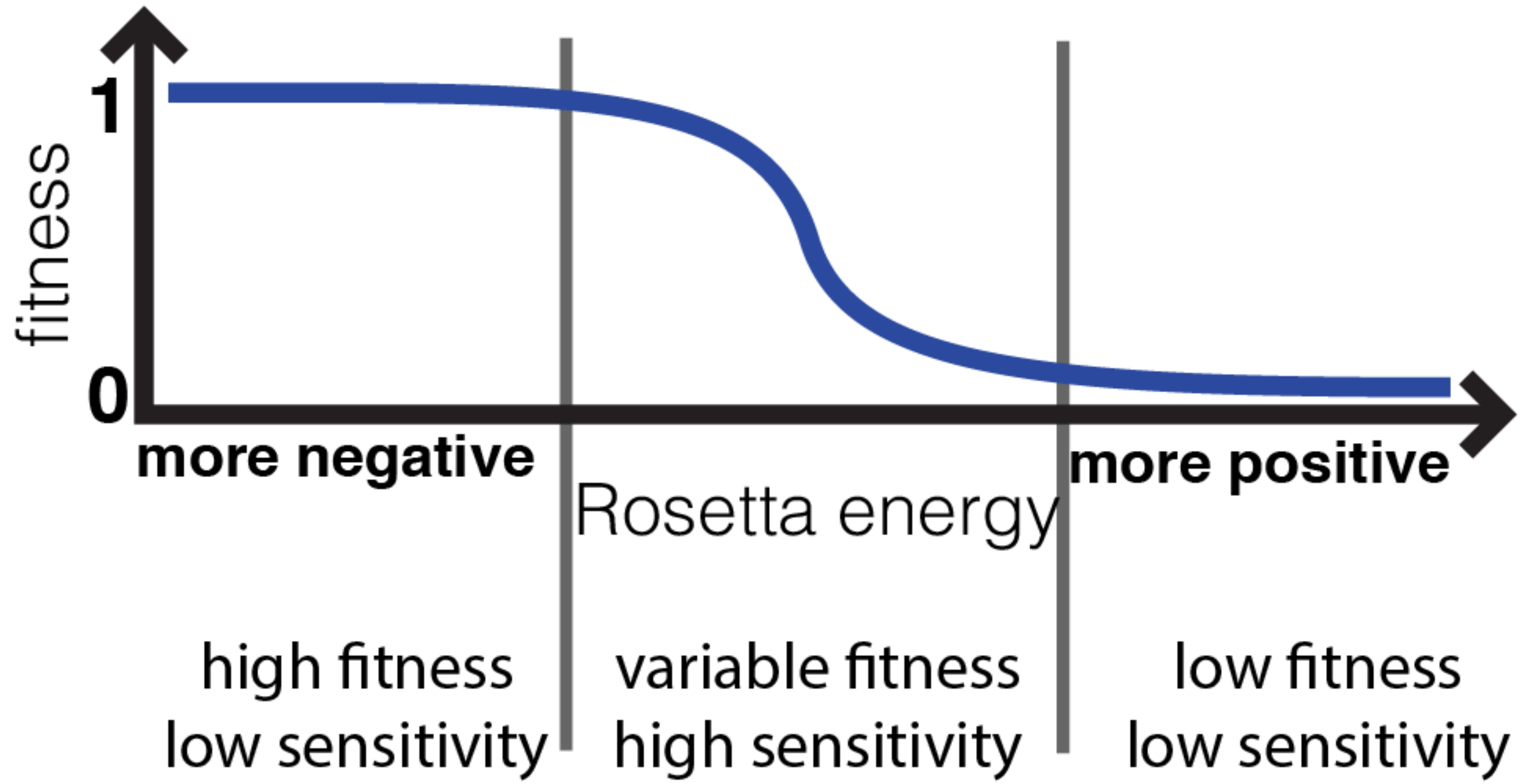
- Want a function for f_i that varies from 1 to 0

Warszawski et al. Journal of Molecular Biology, 2014

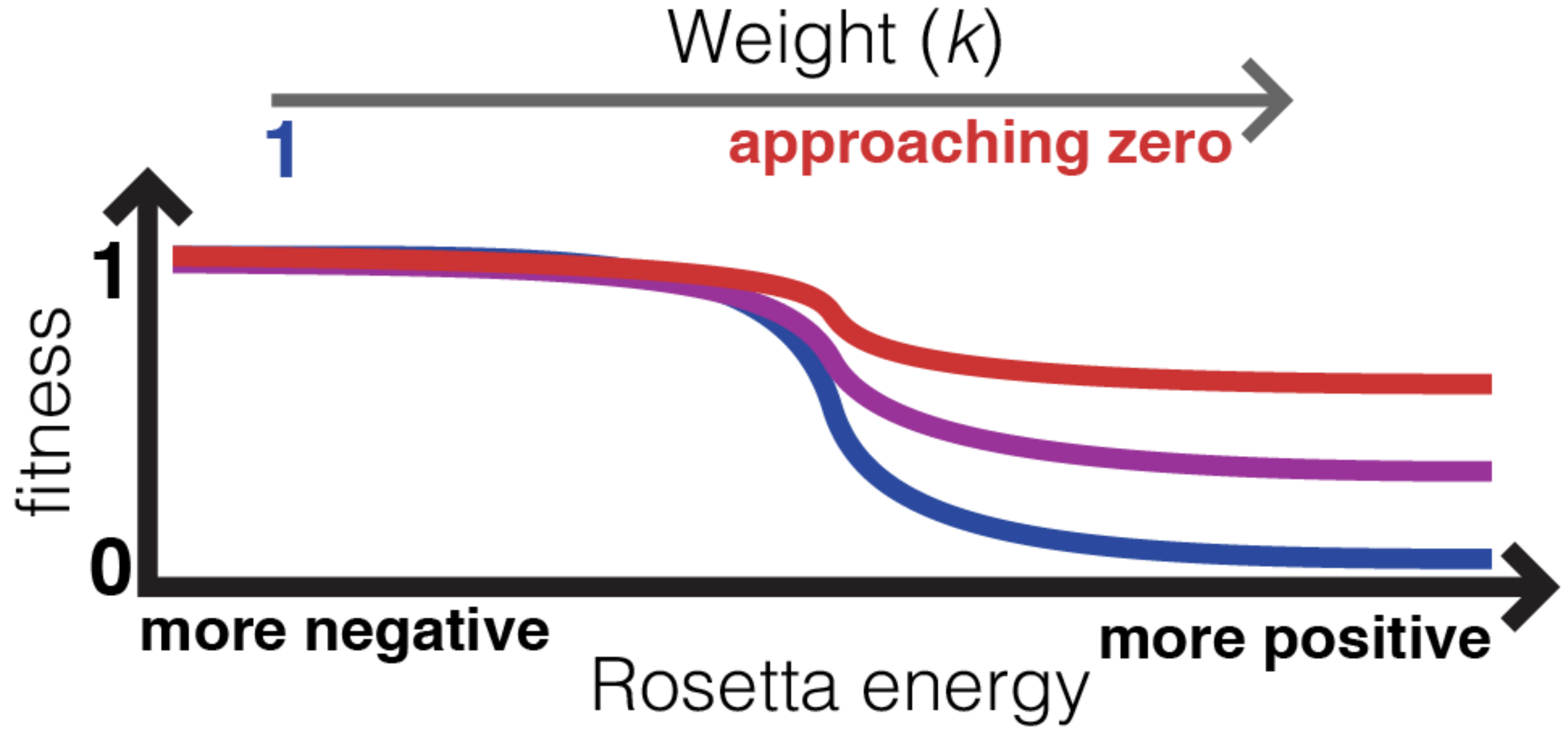
Sigmoids can model the fitness of an individual state



Sigmoids can model the fitness of an individual state

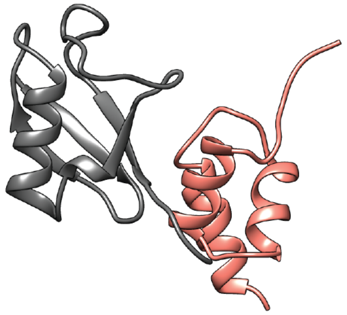


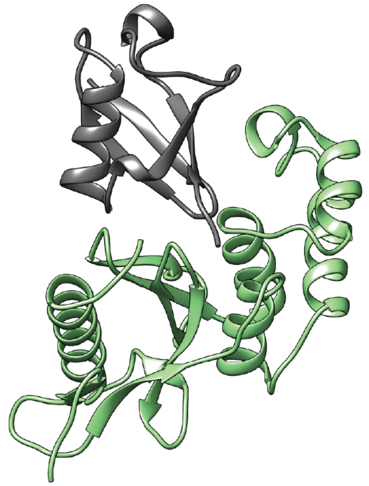
Your task: Adjusting the weights

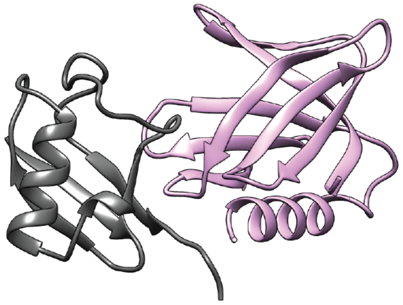


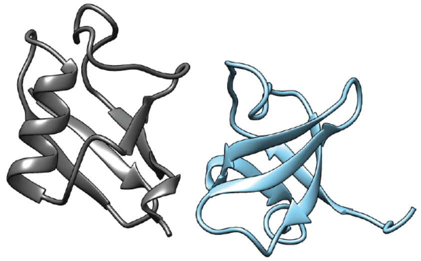
Meaning of changing the weights: the change in fitness for some states may be less important for optimization.

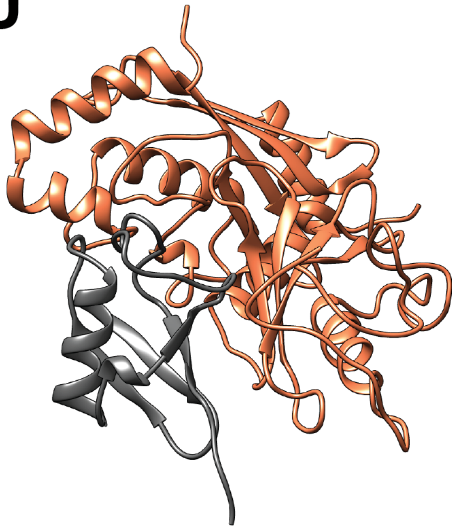
The states that you will be using are the structures of the 5 complexes


CUE


OTU


RPN13


SH3


UQ-CON

2-3 Minute Group Discussion

Within your team:

- What states do you want to examine? Which states would you keep fully weighted? Which states would you down weight?

- What simulation(s) would constitute a control/comparison for these simulations?

AFTER YOU HAVE GENERATED DATA WITH MULTI-STATE DESIGN

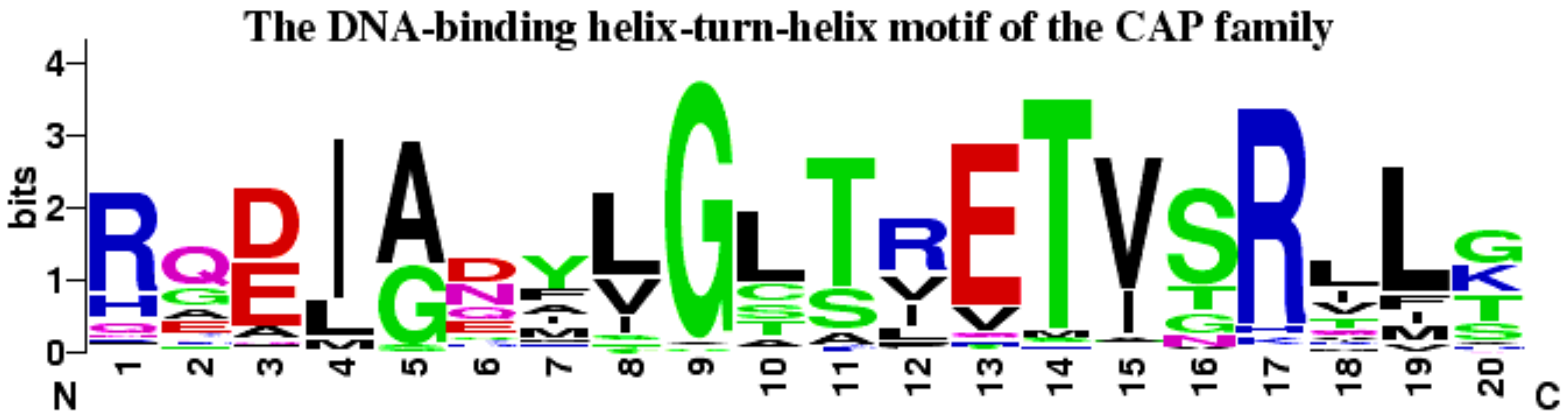
Analyzing the Results

- Output of multi-state design is a fasta file of high-fitness sequences

- Need to compare the results of multiple simulations
  - Multi-state to single state (k = 0 for all but one)
  - Multi-state to multi-state

- Also need to compare Rosetta results to your experimental data

Sequence Logos are Visual and Intuitive



- Column height = Information = Max entropy – Observed entropy
- Character height = Amino acid frequency * Information at position

- Generated from a fasta file of sequences
  - http://weblogo.berkeley.edu/logo.cgi
  - http://weblogo.threeplusone.com/create.cgi

Robison et al., Journal of Molecular Biology, 1998
Schneider and Stephens, Nucleic Acids Research, 1990

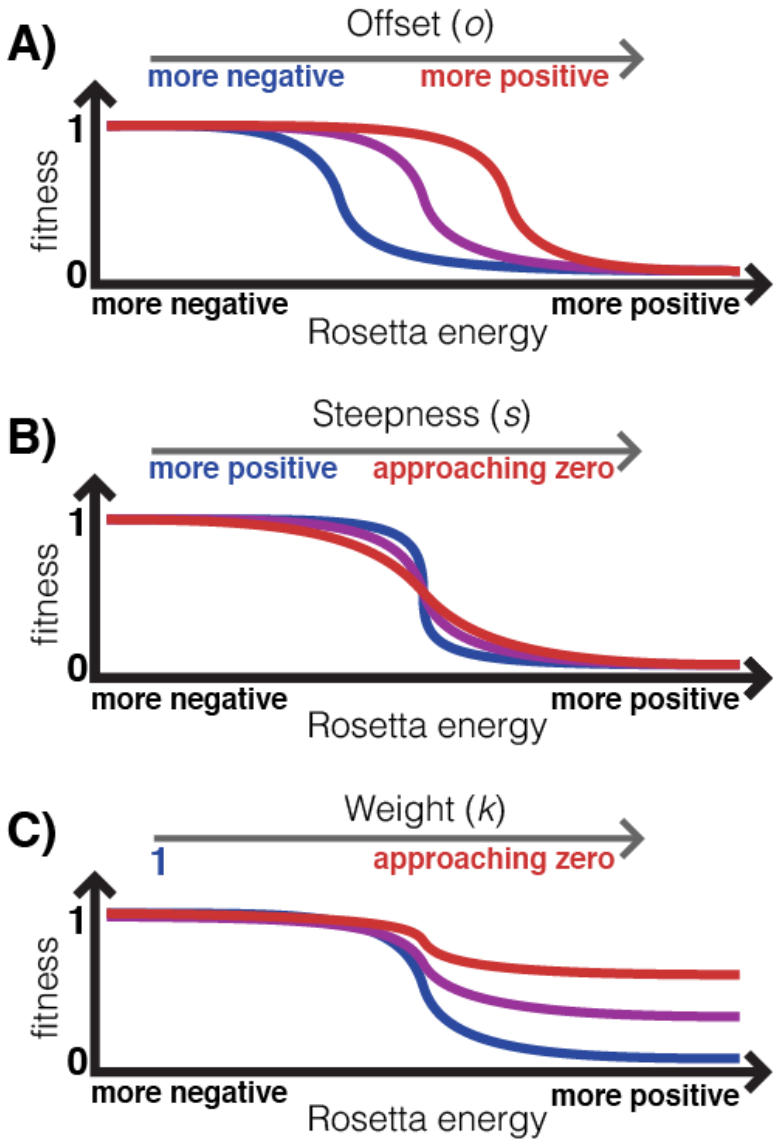
What we are looking for when you present your plan on Monday:

- Justification of using multi-state design to compare with the *in vivo* selection data
- What what residues make up your patch?
    - How did you determine this?
    - Based on your experimental data, which of these residues are you most interested in?
    - What residues are in your patches?
    - Include images of the patch
- What k-values (weights) will you be testing?
- How will you compare sequencing data to the output sequences from Rosetta?
    - Include any relevant outline/flow-chart and equations

Three parameters modulate the fitness curve

sigmoidal equation for fitness of a state

$$f_i = (1 - k) + \frac{k}{1 + e^{s(E_{state} - o)}}$$


A)
Offset (o)
more negative
more positive
fitness
more negative
more positive
Rosetta energy
B)
Steepness (s)
more positive
approaching zero
fitness
more negative
more positive
Rosetta energy
C)
Weight (k)
approaching zero
fitness
more negative
more positive
Rosetta energy