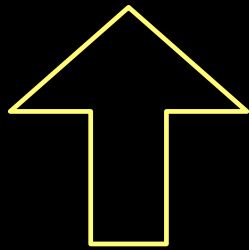


# Computational Challenges in Molecular Design

Tanja Kortemme, UCSF

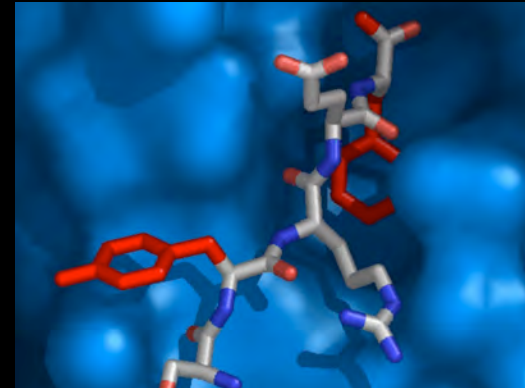
# Design of proteins parts for biological engineering

- recognition
- catalytic

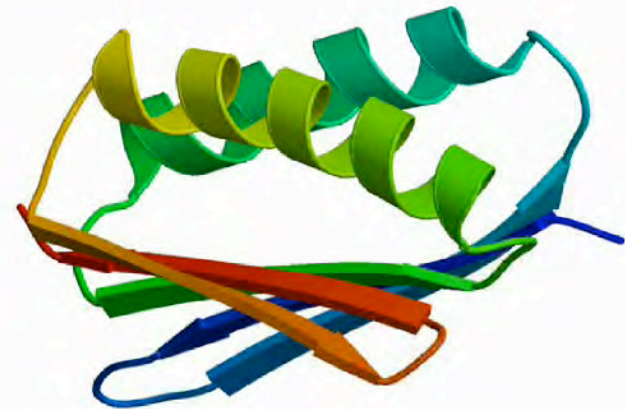


## Computational Models

- physical model (atom-level)
- sampling: structure/sequence
- fitness function
- optimization method



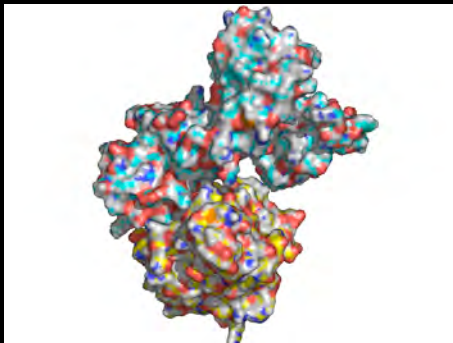
Enzyme  
Substrate ----->Product



# Computational Protein Design

Ponder & Richards, Mayo, Desjarlais & Handel, Hellinga, Harbury, Serrano, et al

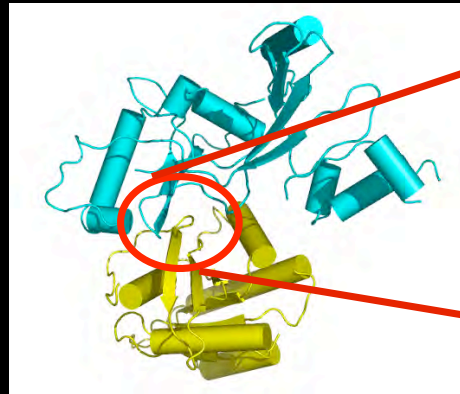
Interaction



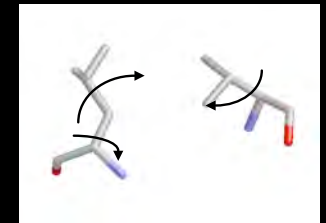
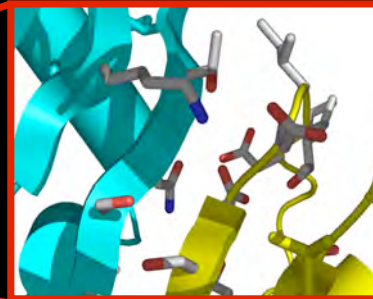
Desired interaction



Backbone



Side Chain Library



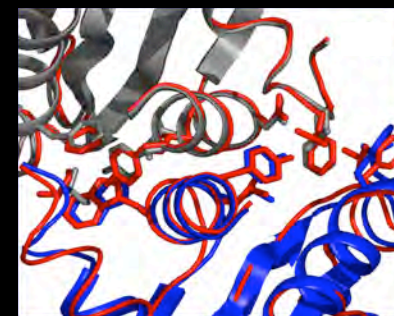
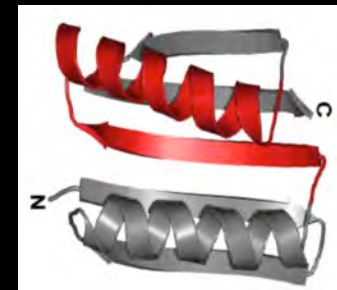
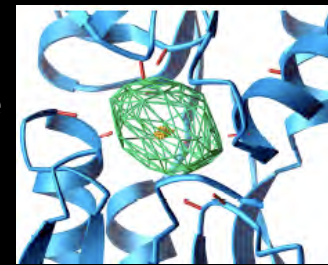
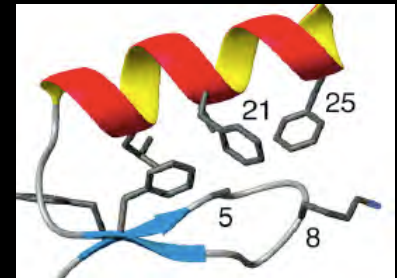
Vary sequences and rotamers

scoring function  
optimization

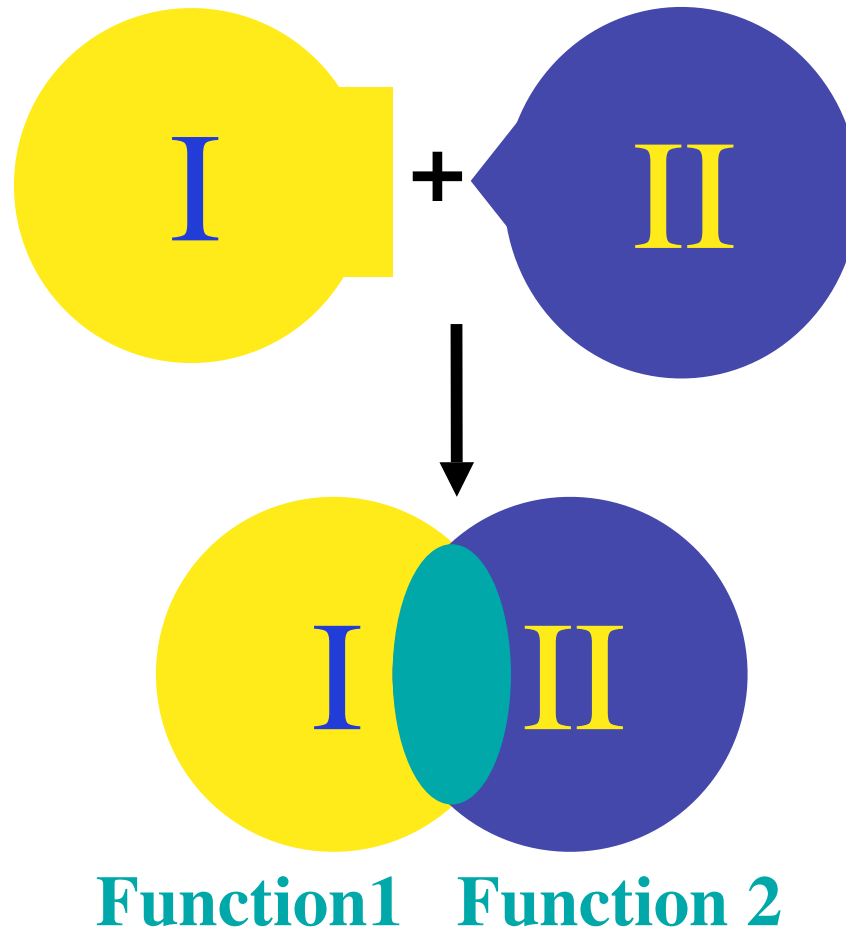
models of sequence/structure  
compatible with design target

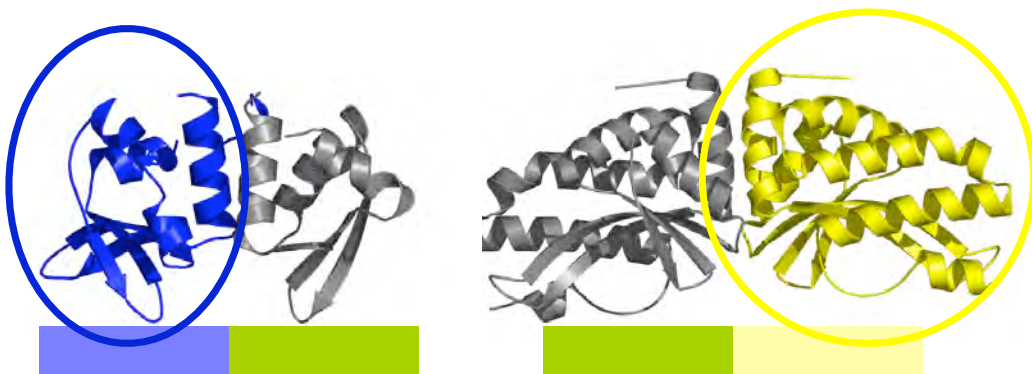
# Molecular Design Successes

- predict a new sequence for an existing protein structure (Steve Mayo & coworkers, 1997)
- engineer a protein to become small molecule (TNT) sensor (Homme Hellinga et al., 2003)
- construct a protein with a structure not observed in nature (Brian Kuhlman & David Baker, 2003)
- engineer new functions by modular recombination of proteins (our work)



# Engineering New Molecules: New functions by fusion

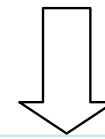
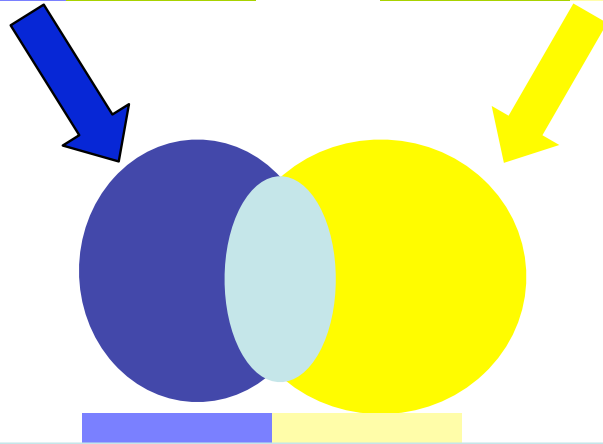




Novel gene-specific reagents  
by combination of modules

**I-DmoI** (N-terminal domain)

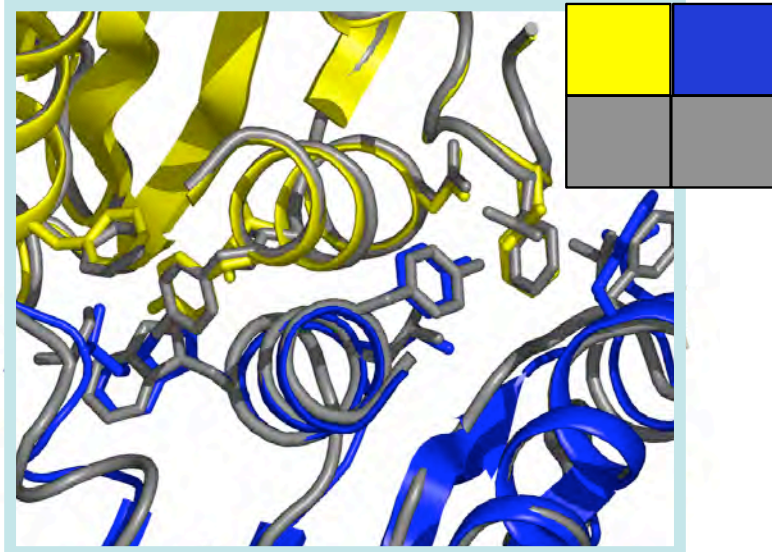
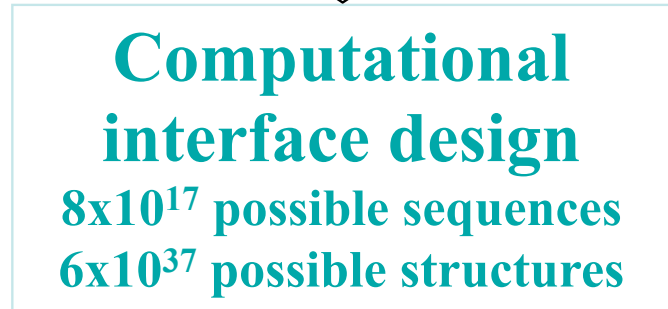
**I-CreI** (one monomer of homodimer)



**Computational  
interface design**

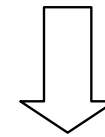
$8 \times 10^{17}$  possible sequences

$6 \times 10^{37}$  possible structures



**Experiment**

PREDICTION



**Functional module  
with novel properties**



# Computational parts design for synthetic biology

---

- **generalizability: computational framework**
  - automated, predictive
  - optimize parts using a mathematical description of part “fitness”
- **interface with existing cellular processes**
  - detail may be useful: recognition selectivity & **orthogonality**
- **what about modularity, standards, abstraction?**
  - generate part families: **tunability**
  - optimize parts as components of modular devices
    - **regulation**

# Computational parts design for synthetic biology

---

- **generalizability: computational framework**
  - automated, predictive
  - optimize parts using a mathematical description of part “fitness”
- **interface with existing cellular processes**
  - detail may be useful: recognition selectivity & orthogonality
- **what about modularity, standards, abstraction?**
  - generate part families: tunability
  - optimize parts as components of modular devices
    - regulation

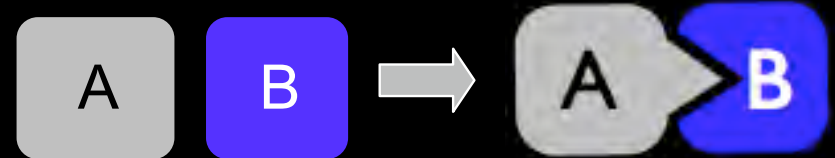


# Design criteria for protein parts

---

→ *to adopt molecular function:*

- protein stability
- interaction strength & rates, ...

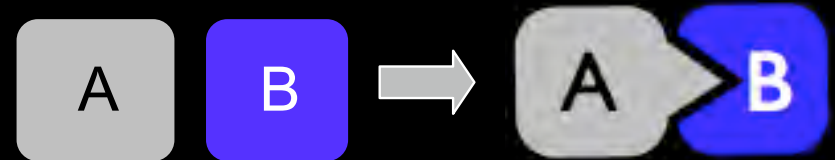


# Design criteria for protein parts

---

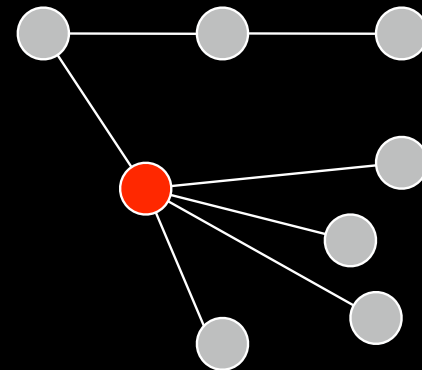
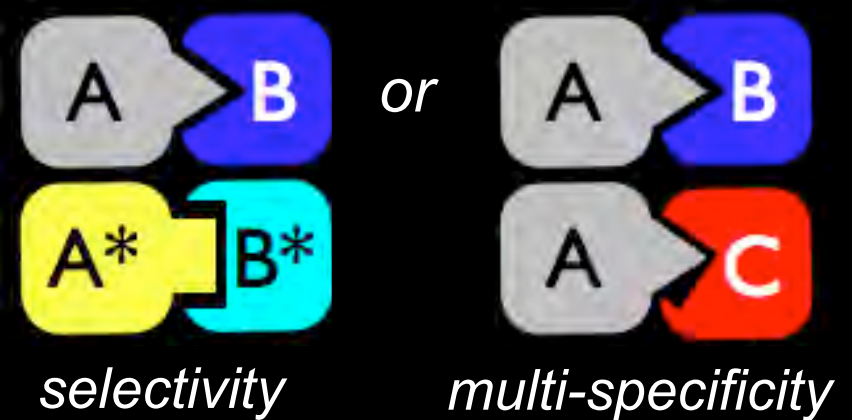
→ *to adopt molecular function:*

- optimize an interface sequence for a particular interaction



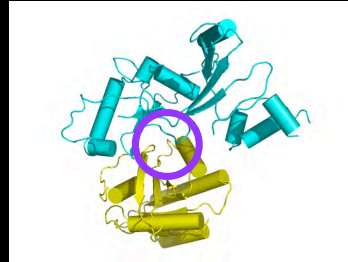
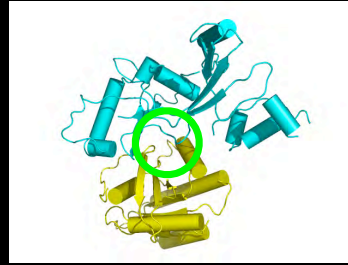
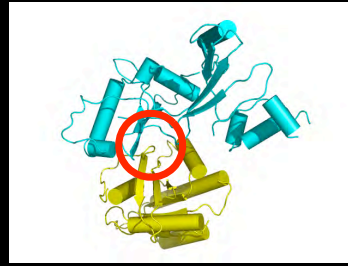
→ *to function within sets of interactions:*

- form desired and avoid unwanted interactions



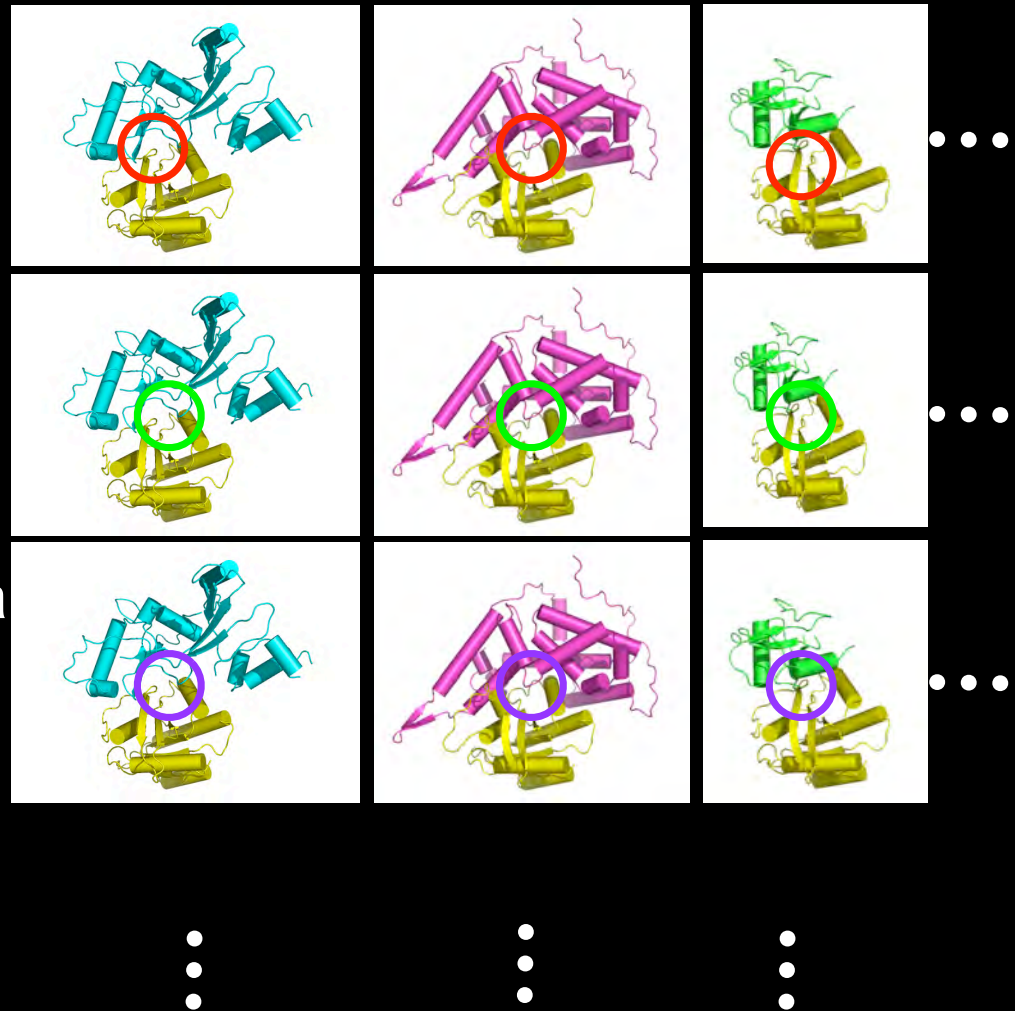
# Sequence optimization for one constraint

- Stability of a desired fold or interaction
  - **SINGLE CONSTRAINT**



# for “fitness” within a network: Optimization for several constraints

- Stability of a desired fold or interaction
  - **SINGLE CONSTRAINT**
- Multi-specificity: ability to recognize several proteins through a common binding site
  - **MULTI CONSTRAINT**



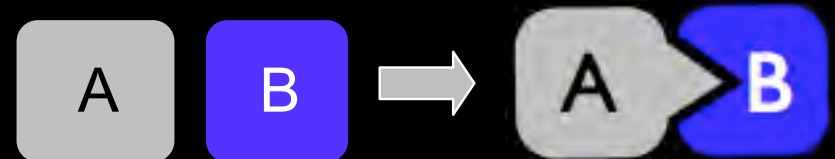
*Humphris & Kortemme, PLoS Computational Biology 2007*

# Design criteria for protein parts

---

→ *to adopt molecular function:*

- optimize an interface sequence for a particular interaction



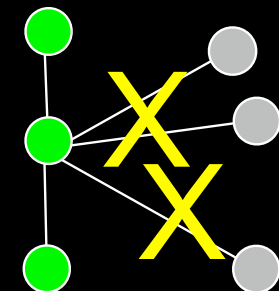
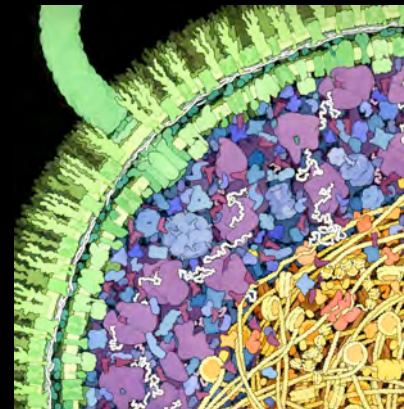
→ *to function within sets of interactions:*

- form desired and avoid unwanted interactions



→ *to function in complex environments (cells):*

- production cost
- interface with cellular processes & cross-talk



# Computational Parts Design Challenges

---

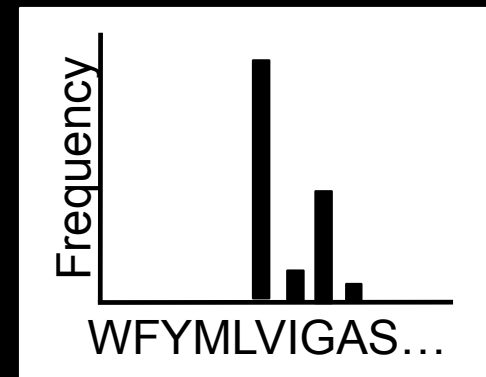
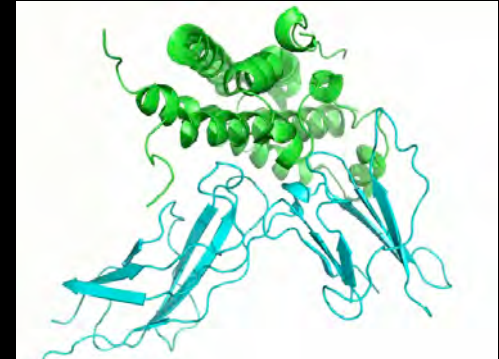
- mathematical description of fitness function
- optimization methods
  - simultaneous sequence/structure optimization (speed)
  - complex fitness functions
- predictiveness:
  - accuracy of the physical model
  - sampling of vast 3D structure space
    - local sampling models are quite successful
    - generate LIBRARIES for testing
    - useful for construction of part families

# Computational parts libraries

---

1. Generate an ensemble of models
2. Sample sequences and record those passing a fitness threshold

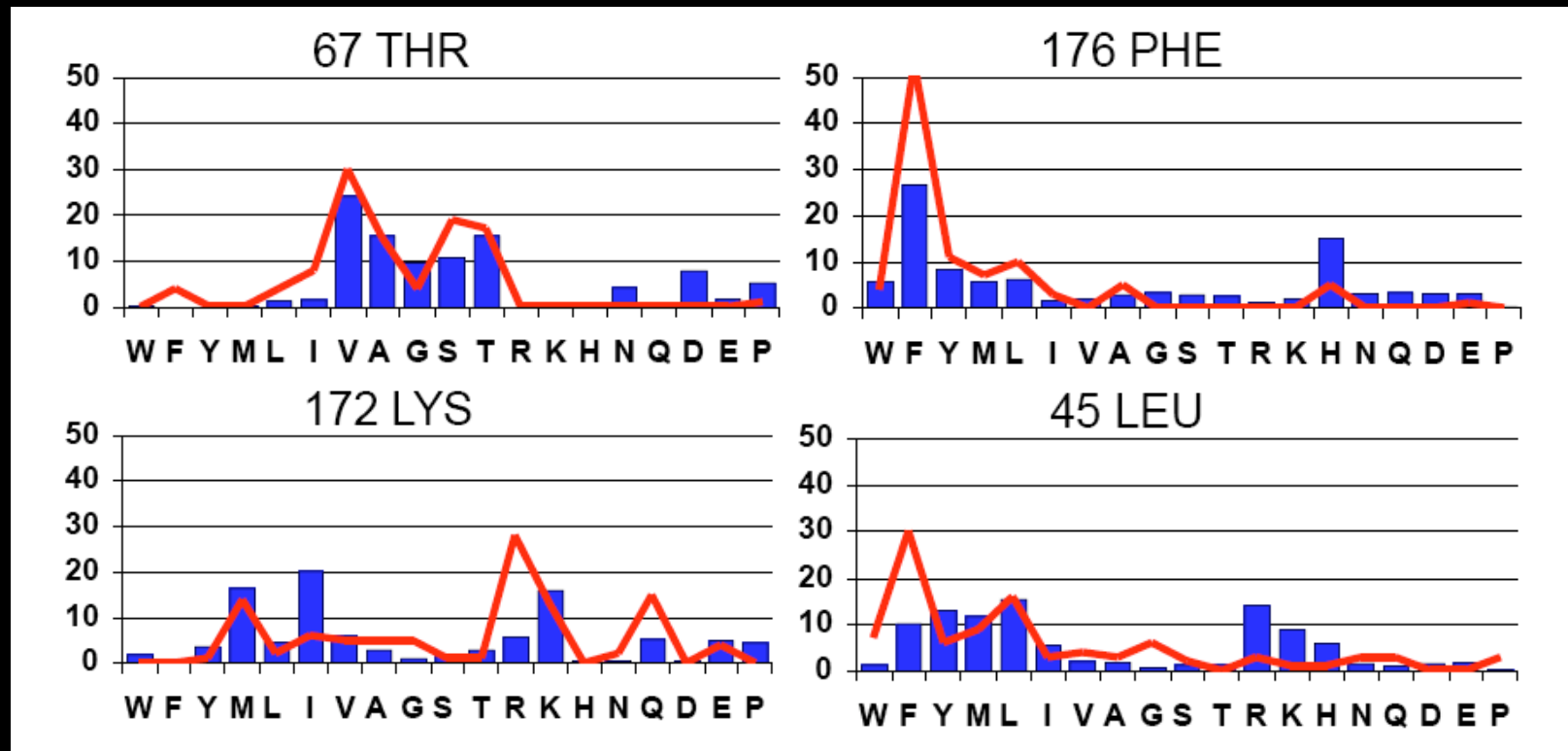
→ results in profile for each sequence position, indicating predicted amino acid choices compatible with the fitness function



# Test: Compare a predicted library to a selection experiment



% selected  
↑  
% predicted  
↑

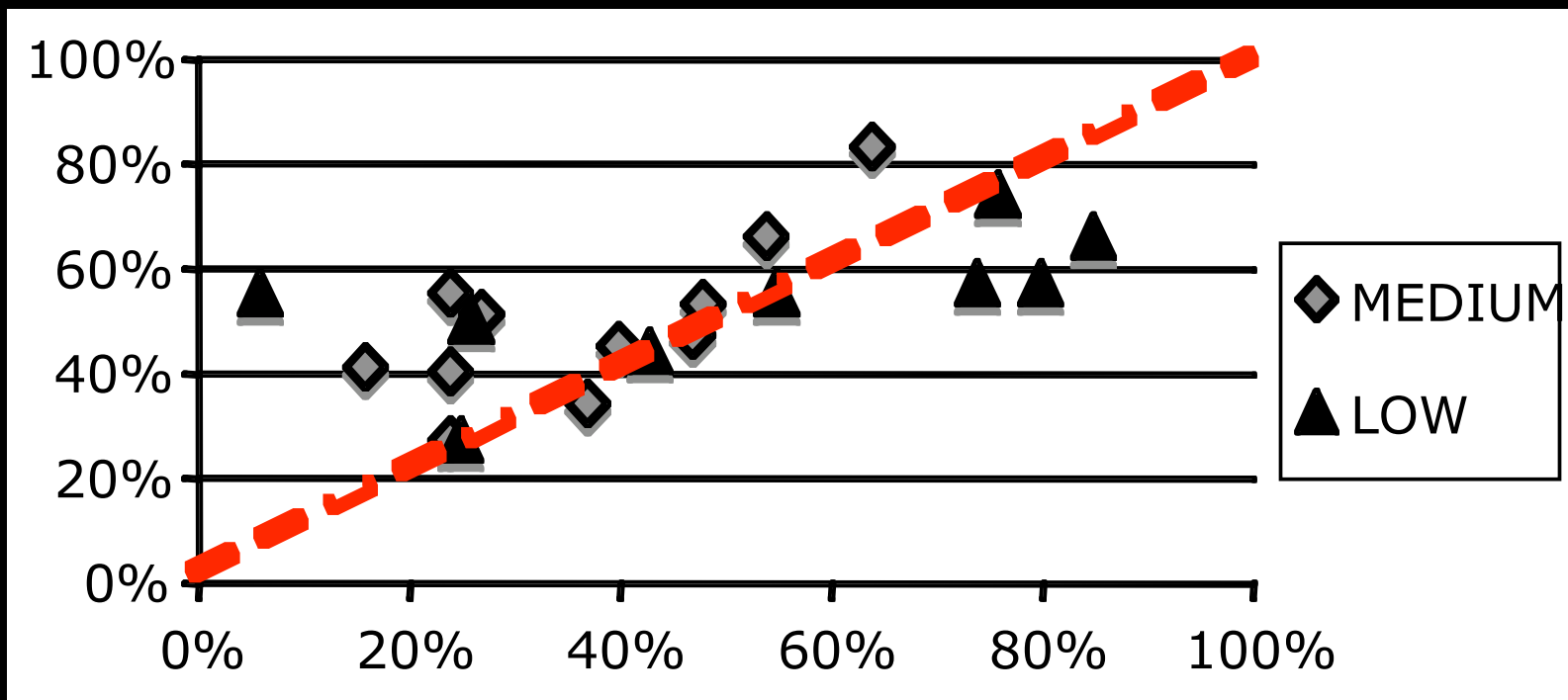


amino acid type



# Coverage of the computational library relative to experimental selection (truth)

coverage using top 3  
experimental choices



coverage of predicted library

library needed to screen all 20 amino acids:  $10^{44}$   
computational library:  $< 10^{11}$

# Conclusions

---

- computational molecular design has been successful for clearly defined molecular functions (stability, binding) if approximations hold: physical model, structures fixed)
- substantial challenges are in:
  - accuracy of physical model
  - sampling & optimization
  - quantitative description of a fitness function
- computational design of libraries can aid in parts optimization and generation of part families

# Thanks to:



## *Computational Models*

Colin Smith

Greg Friedland

Dan Mandell

Anthony Linares (Pomona College)

Mariana Babor

## *Multispecificity*

## *Computational Libraries*

Elisabeth Humphris

## *Engineering Interactions*

Greg Kapp

with Attila Remenyi & Wendell Lim

Cristina Melero

Ryan Ritterson

## *Protein Evolution*

Matt Eames

## *Network Modeling/Analysis*

Rich Oberdorf

SynBERC

Synthetic Biology Engineering Research Center



Alfred P. Sloan Foundation

NIH, Sandler Foundation