7.91 Amy Keating

# Ab Initio Structure Prediction

# &

# Protein Design

# *Ab initio* prediction

- *Ab initio* = “from the beginning”; in strictest sense uses first principles, not information about other protein structures
- In practice, all methods rely on empirical observations about other structures
  - Force fields
  - Knowledge-based scoring functions
  - Training sets
  - Fragment structures

A good review:

Bonneau, R, and D Baker. "Ab Initio Protein Structure Prediction: Progress and Prospects." *Rev Biophys Biomol Struct.* 30 (2001): 173-89.

# Approaches to *ab initio* folding

- Full MD with explicit solvation (e.g. IBM Blue Gene)
  - VERY expensive
  - May not work

- Reduced complexity models
  - No side chains (sometimes no main chain atoms either!)
  - Reduced degrees of freedom
  - On- or off-lattice
  - Generally have a solvation-based score and a knowledge-based residue-residue interaction term
  - Sometimes used as first step to prune the enormous conformational space, then resolution is increased for later fine-tuning

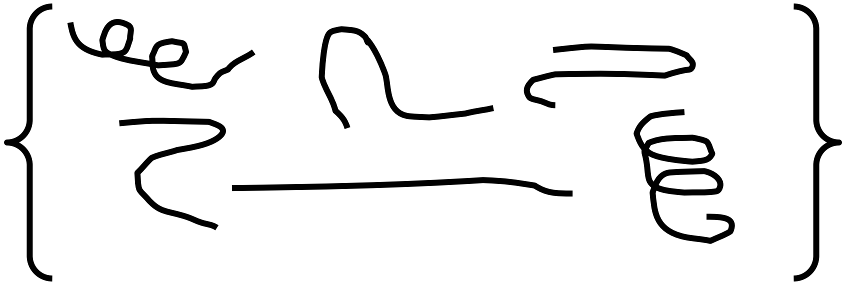
# ROSETTA - the most successful approach to *ab initio* prediction

- David Baker, U. Washington, Seattle
- Based on the idea that the possible conformations of any short peptide fragment (3-9 residues) are well-represented by the structures it is observed to adopt in the pdb
- Generate a library of different possible structures for each sequence segment
- Search the possible combinations of these for ones that are protein-like by various criteria

# ROSETTA fragment libraries

- Remove all homologs of the protein to be modeled (>25% sequence identity)
- For each 9 residue segment in the target, use sequence similarity and secondary structure similarity (compare predicted secondary stucture for target to fragment secondary structure) to select ~25 fragments
- Because secondary structure is influenced by tertiary structure, ensure that the fragments span different secondary structures
- The extent to which the fragments cluster around a consensus structure is correlated with how good a model the fragment is likely to be for the target

**LSERTVARS**



# ROSETTA search algorithm
# Monte Carlo/Simulated Annealing

- Structures are assembled from fragments by:
  - Begin with a fully extended chain
  - Randomly replace the conformation of one 9 residue segment with the conformation of one of its neighbors in the library
  - Evaluate the move:

    Accept or reject based on an energy function
  - Make another random move…
  - After a prescribed number of cycles, switch to 3-residue fragment moves

# ROSETTA scoring function

$$P(structure \mid sequence) = P(structure) \times \frac{P(sequence \mid structure)}{P(sequence)}$$

sequence is constant

need to estimate for decoys built from fragments

Main contributions to *P(structure)*

- secondary structure packing  
  (e.g. ensure β-strands form β-sheets)
- VdW packing

Simons et al. PROTEINS (1999) 34, 82-95

# Native-like structures have characteristic secondary structure packing

Example: b-strand

dipeptide vector

Simons, KT, I Ruczinski, C Kooperberg, BA Fox, C Bystroff, and D Baker. "Improved Recognition of Native-like Protein Structures using A Combination of Sequence-dependent and Sequence-independent Features of Proteins." *Proteins* 34, no. 1 (1 January 1999): 82-95.

# β-strand packing geometry can detect native-like structures

Simons, KT, I Ruczinski, C Kooperberg, BA Fox, C Bystroff, and D Baker. "Improved Recognition of Native-like Protein Structures using A Combination of Sequence-dependent and Sequence-independent Features of Proteins." *Proteins* 34, no. 1 (1 January 1999): 82-95.

# ROSETTA scoring function

$$P(structure \mid sequence) = P(structure) \times \frac{\boxed{P(sequence \mid structure)}}{P(sequence)}$$

sequence is constant

need to estimate for decoys built
from fragments

# ROSETTA scoring function

$$P(sequence \mid structure) = P(aa_1, aa_2, \ldots aa_n \mid X)$$

$$P(aa_1, aa_2, \ldots aa_n \mid X) \approx \prod_i P(aa_i \mid X) \prod_{i<j} \frac{P(aa_i, aa_j \mid X)}{P(aa_i \mid X) P(aa_j \mid X)}$$

$$P(sequence \mid structure) \approx P_{env} P_{pair}$$

$$P_{env} = \prod_i P(aa_i \mid E_i)$$

$E_i$ reflects extent of burial

$$P_{pair} = \prod_{i<j} \frac{P(aa_i, aa_j \mid E_i, E_j, r_{ij})}{P(aa_i \mid E_i, r_{ij}) P(aa_j \mid E_j, r_{ij})}$$

# ROSETTA Obstacles & Enhancements

- Problem 1: generate lots of unrealistic decoys
  - Filter based on contact order, quality of β-sheets, poor packing
- Problem 2: large search space
  - Bias fragment picking by predicted secondary structure, faster computational algorithms
- Problem 3: low confidence in the result
  - Fold many homologs of the target, cluster the answers, report the cluster with highest occupancy

# ROSETTA performance at CASP4 was very impressive

- 17/21 predictions had > 50 residue fragments with rmsd < 6.5Å
- Occasionally found structures *better* than the best representative in the pdb (i.e. better than best-possible fold recognition performance)

6.4 Å rmsd
5 Cys pairs correct

**new folds**

4.9 Å rmsd

Bonneau, R, J Tsai, I Ruczinski, D Chivian, C Rohl, CE Strauss, and D Baker. "Rosetta in CASP4: Progress in Ab Initio Protein Structure Prediction." *Proteins* Suppl 5 (2001): 119-26.

# Flowchart for ROSETTA as used in CASP5

Bradley, P, D Chivian, J Meiler, KM Misura, CA Rohl, WR Schief, WJ Wedemeyer, O Schueler-Furman, P Murphy, J Schonbrun, CE Strauss, and D Baker. "Rosetta Predictions in CASP5: Successes, Failures, and Prospects for Complete Automation." *Proteins* 53, Suppl 6 (2003): 457-68.

# CASP5 Rosetta Performance□

Bradley, P, D Chivian, J Meiler, KM Misura, CA Rohl, WR Schief, WJ Wedemeyer, O Schueler-Furman, P Murphy, J Schonbrun, CE Strauss, and D Baker. "Rosetta Predictions in CASP5: Successes, Failures, and Prospects for Complete Automation." *Proteins* 53, Suppl 6 (2003): 457-68.

# Computational Protein Design

# Protein Design as an Inverse Folding Problem

Goal: come up with a sequence that folds to give a protein-like structure with some desired properties

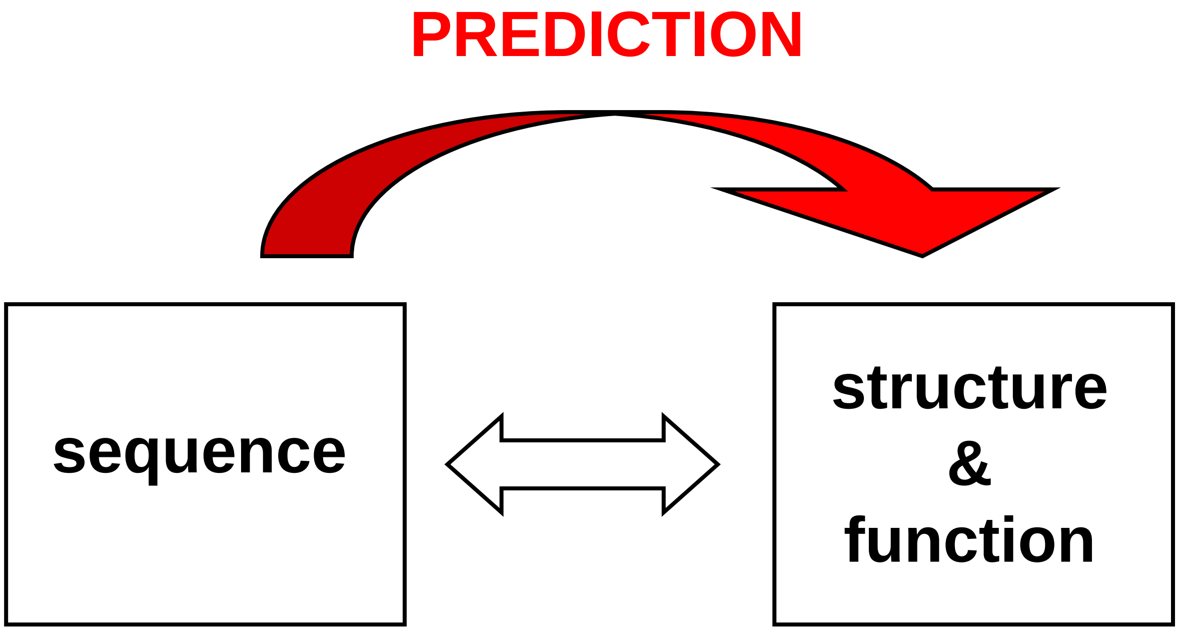
Examples of design goals:

# **Protein Design** as an Inverse Folding Problem

Goal: come up with a sequence that folds to give a protein-like structure with some desired properties

Examples of design goals:

- a pre-defined structure/fold
- a desired oligomerization state
- enhanced thermal stability
- ability to bind a given ligand
- ability to catalyze a certain reaction



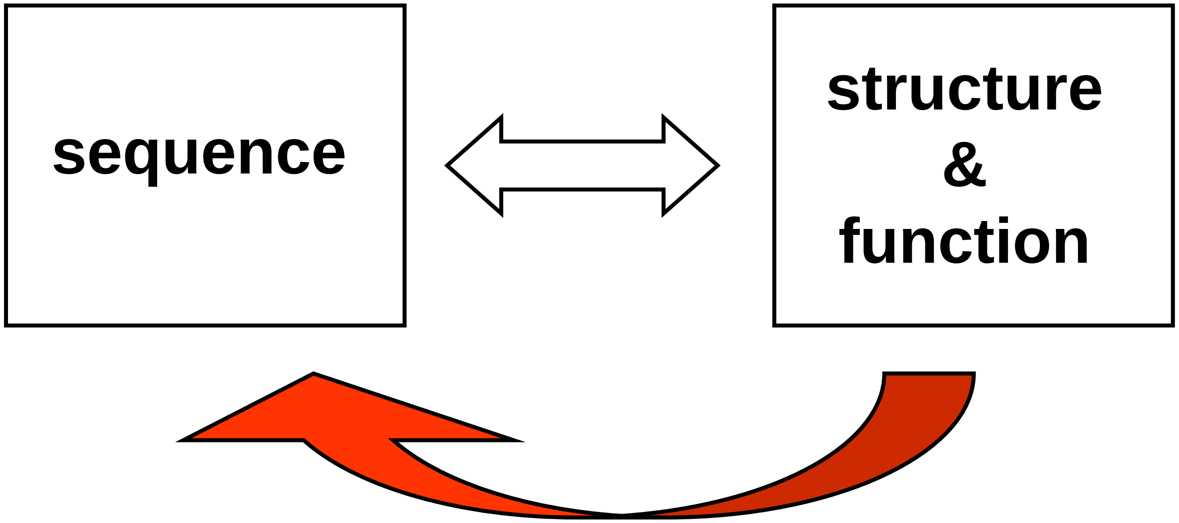
PREDICTION

sequence

structure
&
function

Why is it hard?

- many possible conformations for the protein
- many may have similar energies
- calculated energies are estimates
- hard to tell the correct structure



sequence

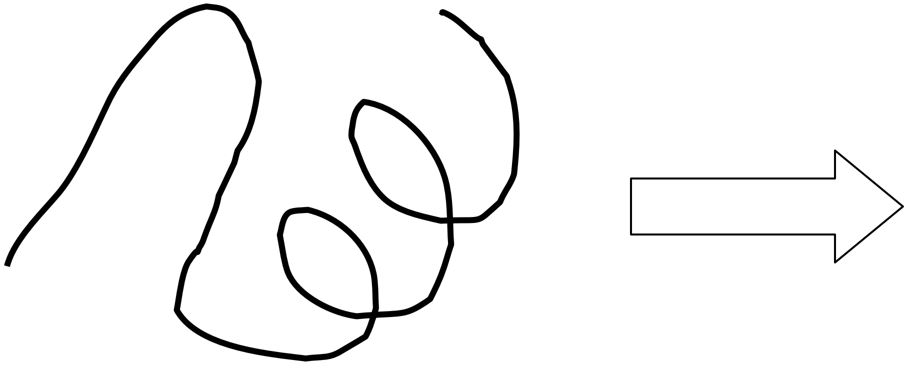
structure
&
function

DESIGN

Why is it hard?

- many possible sequences
- don’t know what structure each sequence adopts
- calculated energies are estimates
- hard to tell the correct structure

## Goal 1: Design a protein sequence that adopts a given structure.



MELKKARSTPAR…

How to judge success?

- what resolution is required?

  *must have the correct fold*

  *do the side chains all have to have specific, predicted positions?*
- compare stability to native proteins
- compare structural uniqueness to native proteins
- must solve the structure to know how well you did!

## A formal statement of the problem:

Given a target fold, specified by the atomic coordinates of a backbone structure, find a sequence that will fold to that structure.

There may be many structures that adopt the fold. To increase your chances of success, try to find one of the most stable.

Try to *minimize* the quantity $\Delta G^{fold} = G^{folded} - G^{unfold}$

Need a way to:

a. search through the different possible sequences

b. evaluate $\Delta G^{fold}$

Then pick the best sequence as the design.

*What is ignored in this approach?*

# Two big challenges in computational protein design:

1. SEARCH PROBLEM: There are many possible sequences: $20^N$
   *in general, these can't be enumerated exhaustively*

2. ENERGY PROBLEM: To evaluate $\Delta G^{fold}$ for a sequence we need to know the energy in the folded and unfolded states
   a. what is the structure of the folded state?
      *we know the backbone, but what about the side chain atoms?*
   b. how should we model the unfolded state?
   c. it is hard to model the free energy, G
   d. we need a fast way to evaluate the energies because there are so many sequences to consider

# Assumptions made to address some of these problems:

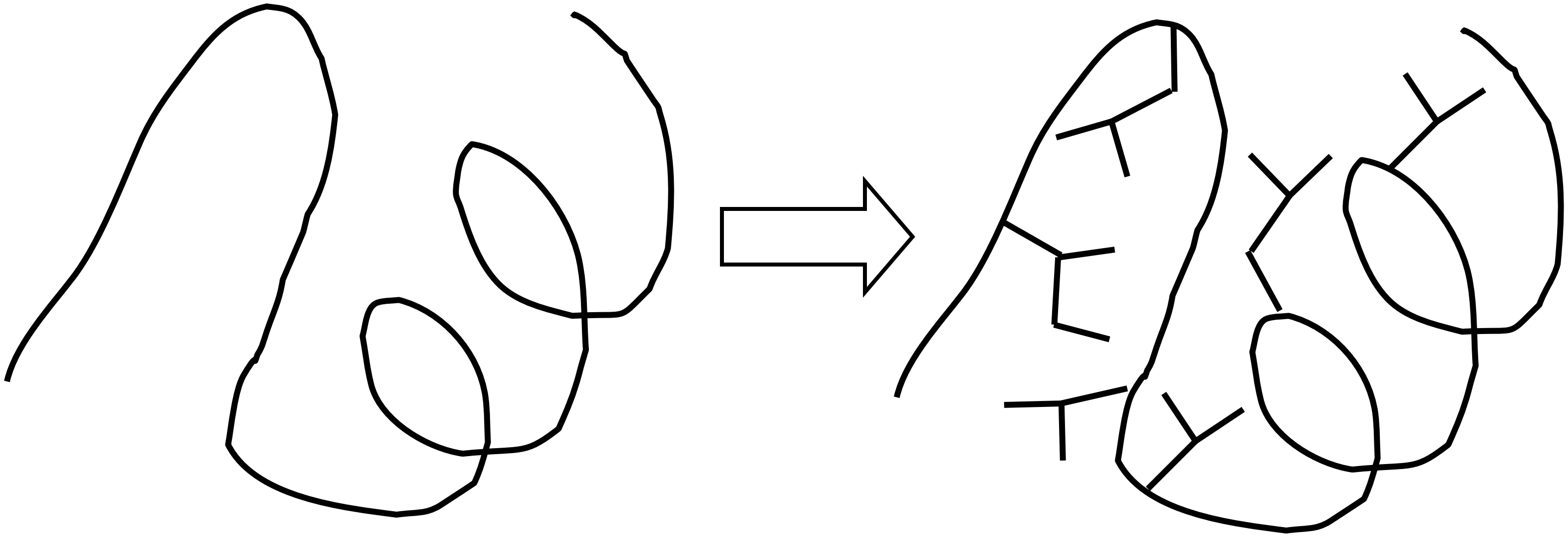
1. Replace $\Delta G$ with $\Delta E$ - assume that the large entropic contributions to protein folding mostly cancel when considering the folding of different sequences to a similar structure.

2. Model $\Delta E$ using molecular mechanics energy functions - do not include explicit solvent but instead use approximate empirical functions.

3. Assume that there are no specific interactions between residues in the unfolded state.
   Corollary: the energy of the unfolded states depends ONLY on the amino acid composition of the protein.

# What is the structure of the folded state?
# The side chain packing problem.

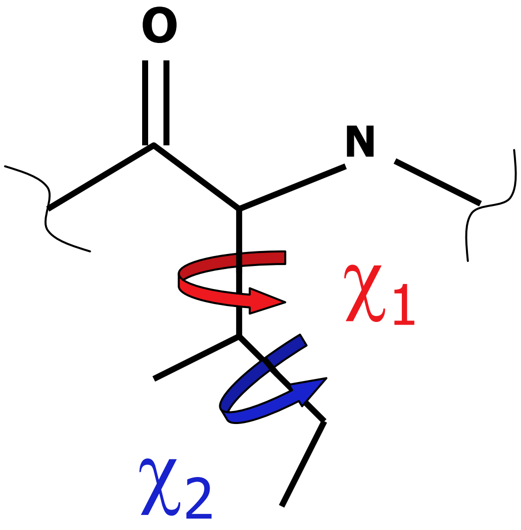
Given the coordinates of the backbone, and the sequence of the protein, put the side chains on in their correct conformation.
This is a sub-problem of protein structure prediction.
Note:  Also useful for final model refinement in homology modeling.

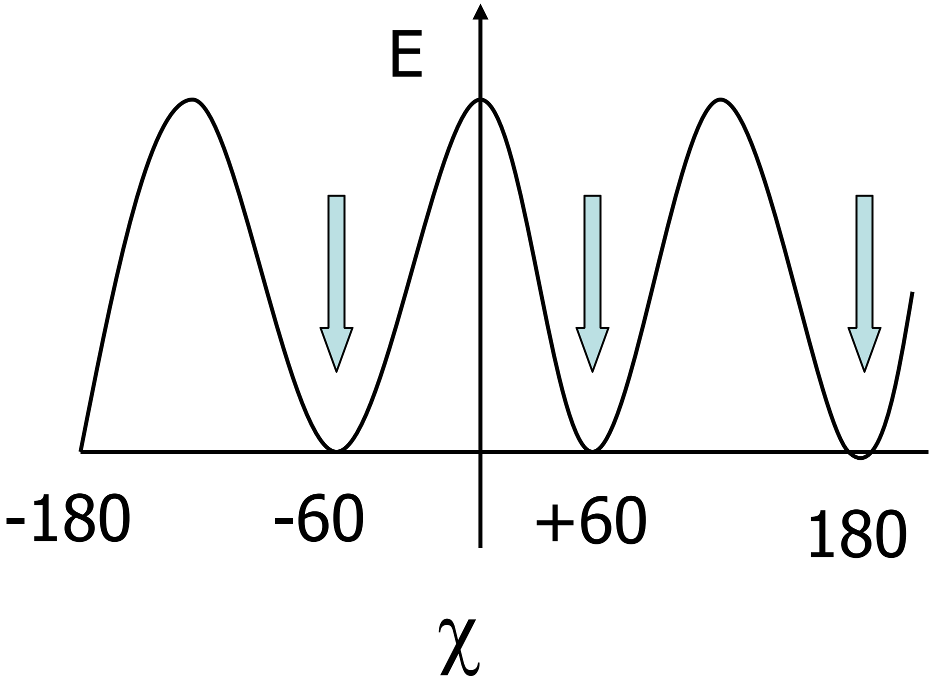


How should we search side chain conformational space?

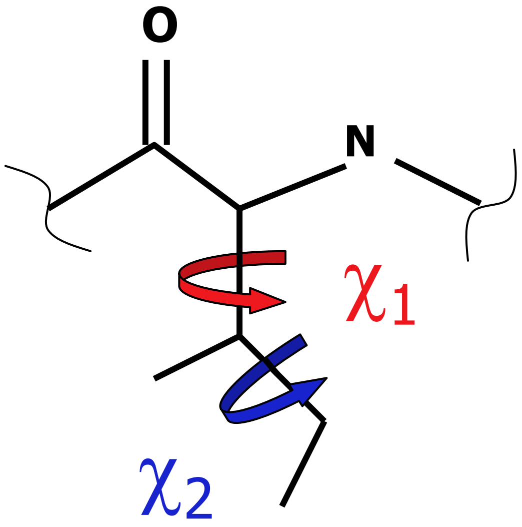
# Side chain **rotamer** approach

- In theory, there are an infinite number of different possible side chain conformations, corresponding to small variations of the side chain bond lengths, bond angles and dihedral angles.
- Only consider the most energetically-favorable possibilities.
- Bond lengths and angles are assumed NOT to change.

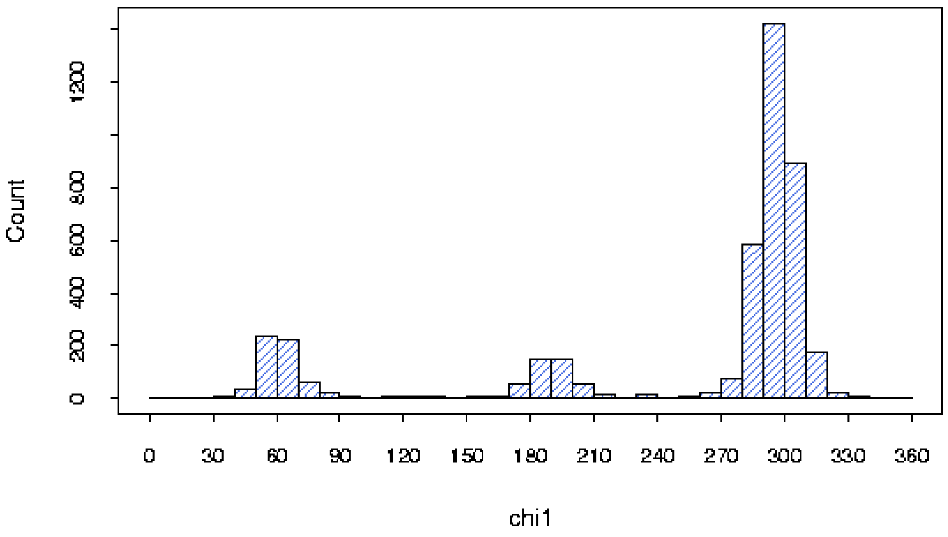




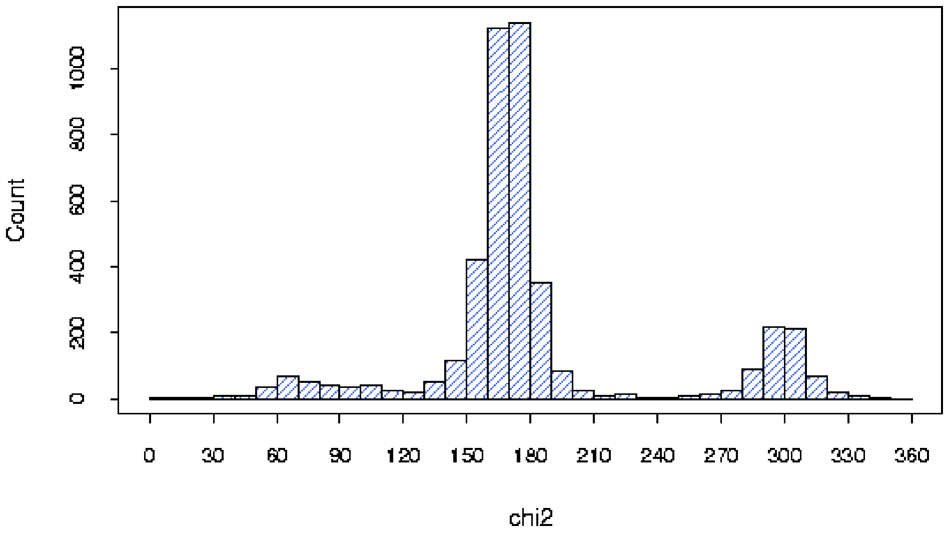
“knowledge-based rotamers”



Ile chi1 distribution



Ile chi2 distribution



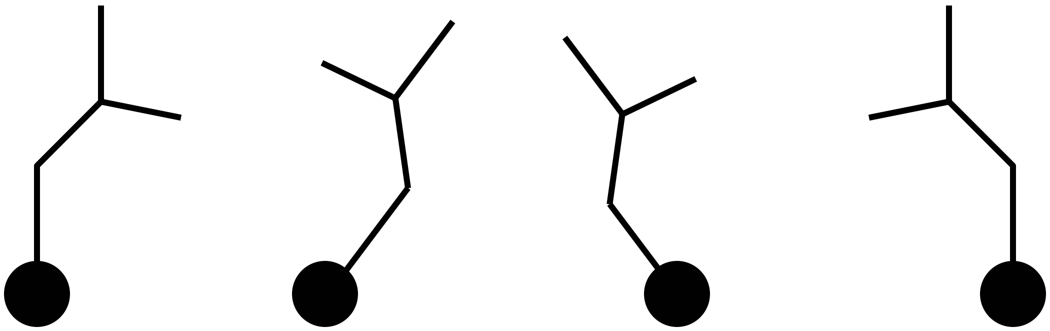
www.fccc.edu/research/labs/dunbrack/confanalysis.html

Courtesy of Roland L. Dunbrack, Jr. Used with permission.

# A rotamer library

http://dunbrack.fccc.edu/bbdep/

```
Res r1r2r3r4 n(r1) n(r1,r2) p(r1,r2) sig p(r2|r1)  sig   chi1  sig     chi2  sig
=================================================================================
LEU 1 1 0 0    239    137    0.90   0.06   57.12   2.60   58.0 16.1     80.4 16.7
LEU 1 2 0 0    239     93    0.61   0.05   38.87   2.56   69.8 18.9    162.8 20.3
LEU 1 3 0 0    239      9    0.06   0.02    4.01   1.03   69.9 22.0    -65.4 30.5
LEU 2 1 0 0   4936   4239   27.70   0.30   85.86   0.40 -177.4 13.1     63.1 11.0
LEU 2 2 0 0   4936    553    3.62   0.12   11.21   0.37 -158.5 19.0   -179.5 29.9
LEU 2 3 0 0   4936    144    0.95   0.06    2.93   0.20 -166.5 16.5    -75.8 23.1
LEU 3 1 0 0  10124   1095    7.16   0.17   10.82   0.25  -91.2 15.3     43.9 26.9
LEU 3 2 0 0  10124   8758   57.23   0.33   86.50   0.28  -65.4 10.4    175.4 10.0
LEU 3 3 0 0  10124    271    1.78   0.09    2.68   0.13  -83.6 14.3    -47.9 25.1
```



Courtesy of Roland L. Dunbrack, Jr. Used with permission.

## Side chain packing is a large combinatorial problem

- Rotamer libraries have ~ $3^{(\# \chi \text{ angles})}$ entries per amino acid
- Side chains have 0 (Ala, Gly) to 4 (Lys, Arg) dihedral angles
- Proteins have ~ 100 amino acids per domain
- Total possible side chain conformations ~ $10^{100}$

# What to do?

1. Some rotamers are much more favorable than others
2. Local backbone conformation strongly influences the side chain conformation
3. Some rotamers clash with the (local or non-local) backbone

BUT - the space left to search is still REALLY BIG!

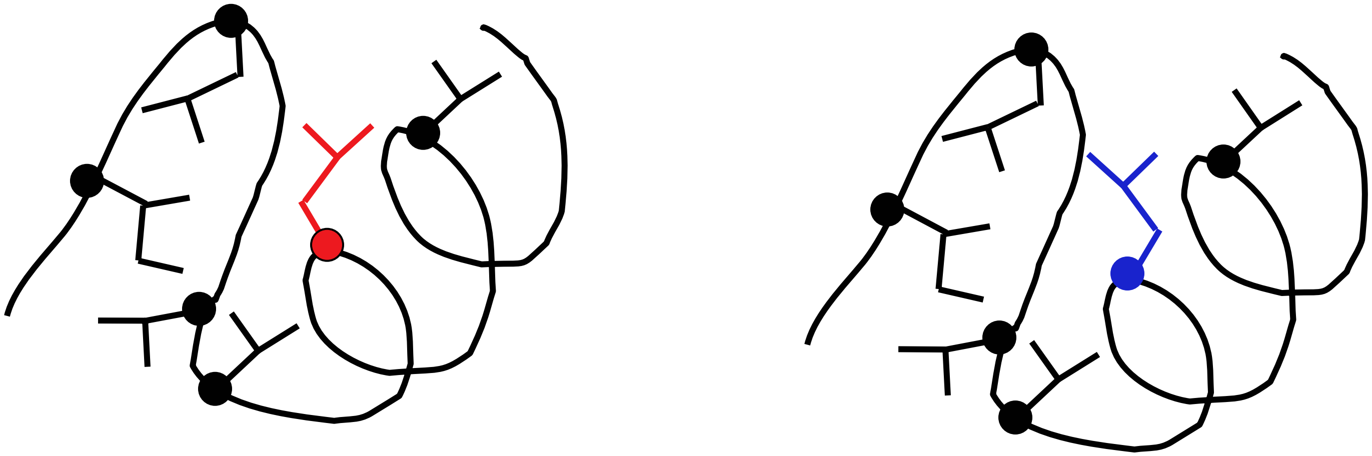
# Search algorithms for large spaces

- Exhaustive search - TOO SLOW (but useful for testing small systems)
- Stochastic searches
  - Monte Carlo
  - Genetic Algorithms
- Pruning Algorithms
  - Branch and Bound
  - Dead End Elimination

# Dead End Elimination

Main idea: eliminate, one at a time, rotamer choices that CAN NOT UNDER ANY CIRCUMSTANCES be part of the minimum energy solution

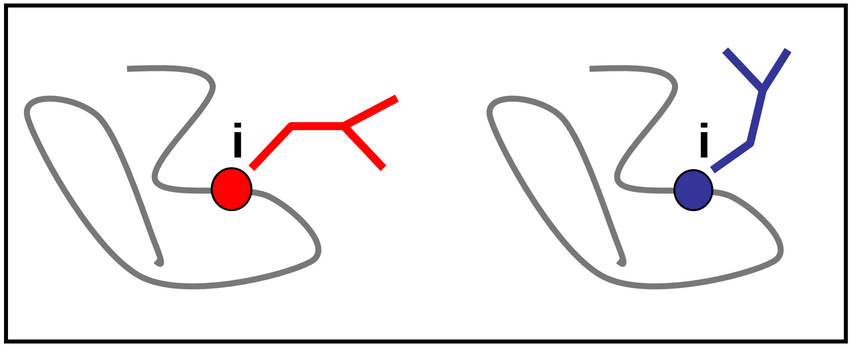
**How can you know this?**

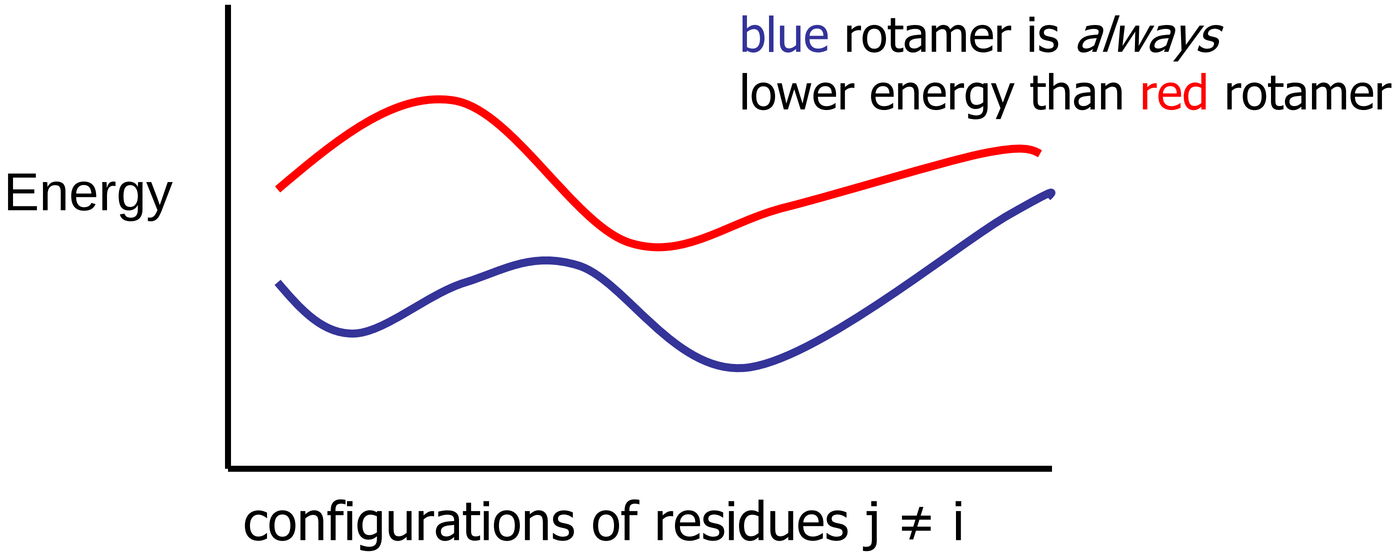


Desmet,De Maeyer, Hazes & Lasters, *Nature* (**1992**) *356*, 539
Goldstein, *Biophys. J.* (**1994**) *66*, 1335

# Dead End Elimination algorithm

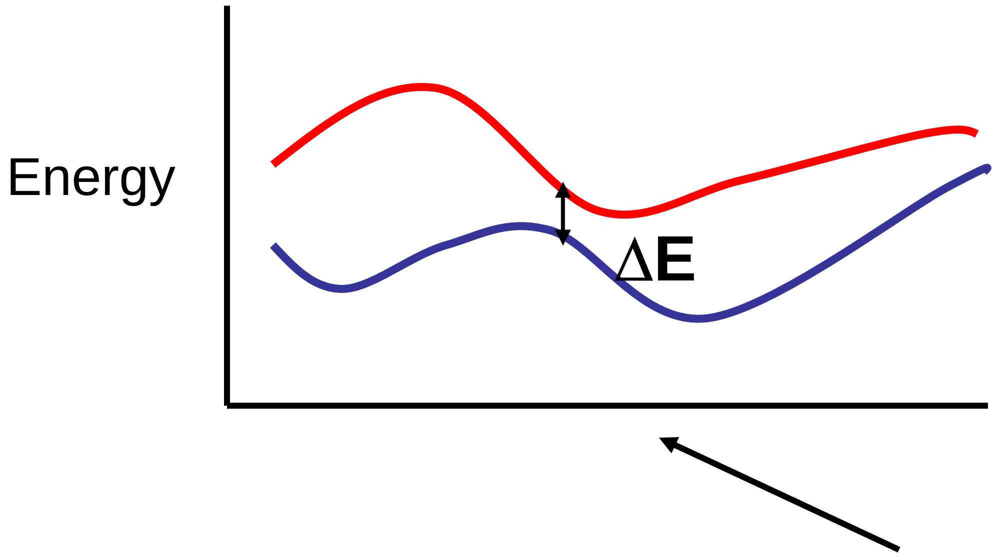
identify and eliminate rotamers which *can not* be part of the best solution





blue rotamer is *always* lower energy than red rotamer

# What energy function to use?



Can’t afford to calculate energies at all these configurations!

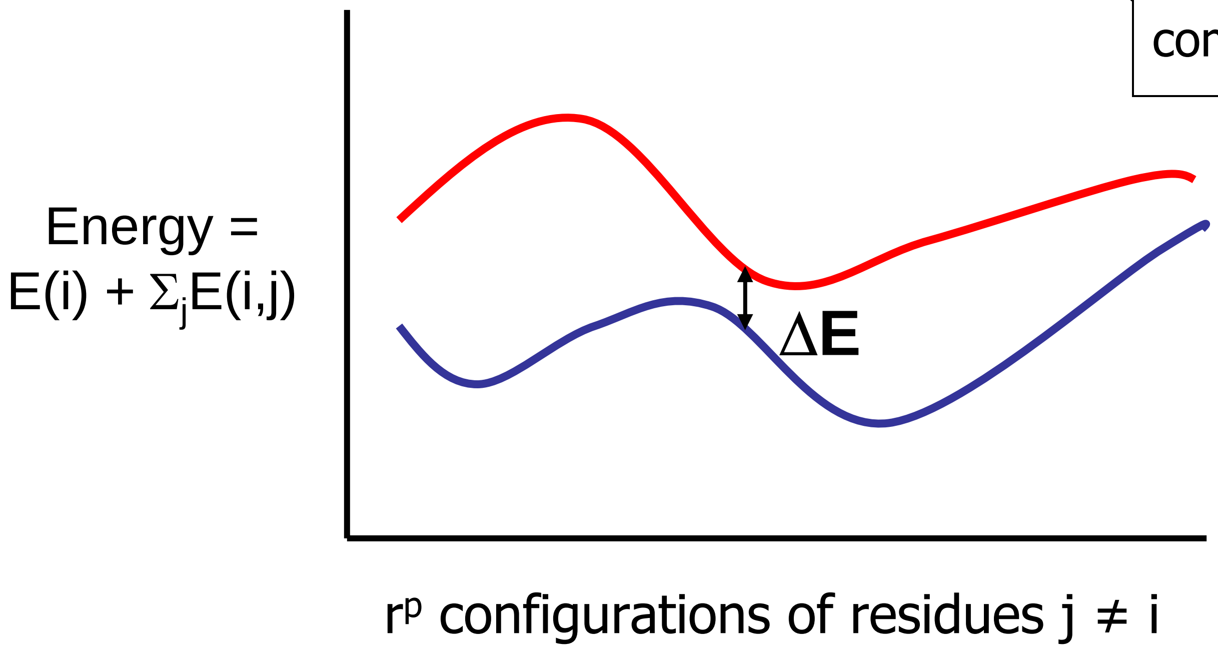
Use a pairwise energy function.
The energies E are based on molecular
mechanics and include the torsional, van der Waals and
electrostatic energies that we have talked about.

$$E_{total} = \Sigma_i E(i_r) + \Sigma_i \Sigma_j E(i_r, j_s)$$

# What is the least energy it could cost to replace $i_s$ with $i_r$?

$$\min \Delta E = E(i_r) - E(i_s) + \Sigma_j[\min t\{E(i_r,j_t) - E(i_s,j_t)\}]$$

only need to do p x r comparisons, not $r^p$



Goldstein, RF. "Efficient Rotamer Elimination Applied to Protein Side-chains and Related Spin Glasses." *Biophys. J.* 66 (1994): 1335-1340.

# Dead End Elimination Criterion

$$\text{if } E(i_r) - E(i_s) + \Sigma_j[\textit{min over t}\{E(i_r,j_t) - E(i_s,j_t)\}] > 0$$

$$\text{then eliminate } i_r$$

Apply iteratively to all rotamer pairs

As rotamers are eliminated the energy profile changes, leading to elimination of further rotamers

When no more rotamers are eliminated, the algorithm can be generalized to identify *pairs* of rotamers that are not consistent with the global minimum solution

# Side chain repacking performance

Limitations come from:

- the finite library from which side-chain positions are chosen
- The fixed bond length and bond angle assumption
- the ability of the energy function to discriminate which choice is best

| | $\chi_1$ correct | $(\chi_1+\chi_2)$ correct |
|---|---|---|
| core sidechains | ~90% | ~80% |
| all sidechains | ~80% | ~70% |

# Pairwise energy functions for protein design

$$E_{total} = \Sigma_i E(i_r) + \Sigma_i \Sigma_j E(i_r, j_s)$$

*assume* ***p*** *design positions*
***r*** *rotamers (on average)*

## single residue term $E(i_r)$

energy of interaction between a single residue i with rotamer r and the *template;*
there are p*r of these

the *template* consists of all atoms that don't change position in the design

## pair energy terms $E(i_r, j_s)$

energy of interaction between residue i in rotamer r and residue j in rotamer s;
there are $p(p-1)r^2/2$ of these

some terms are only single residue: bonds, angles (not usually used), torsions
some terms are easy to make pair-wise:
VdW interactions, Coulomb electrostatic interactions, H-bonds (if using)
some terms are harder to make pair-wise:
solvation energies, screened electrostatic interactions

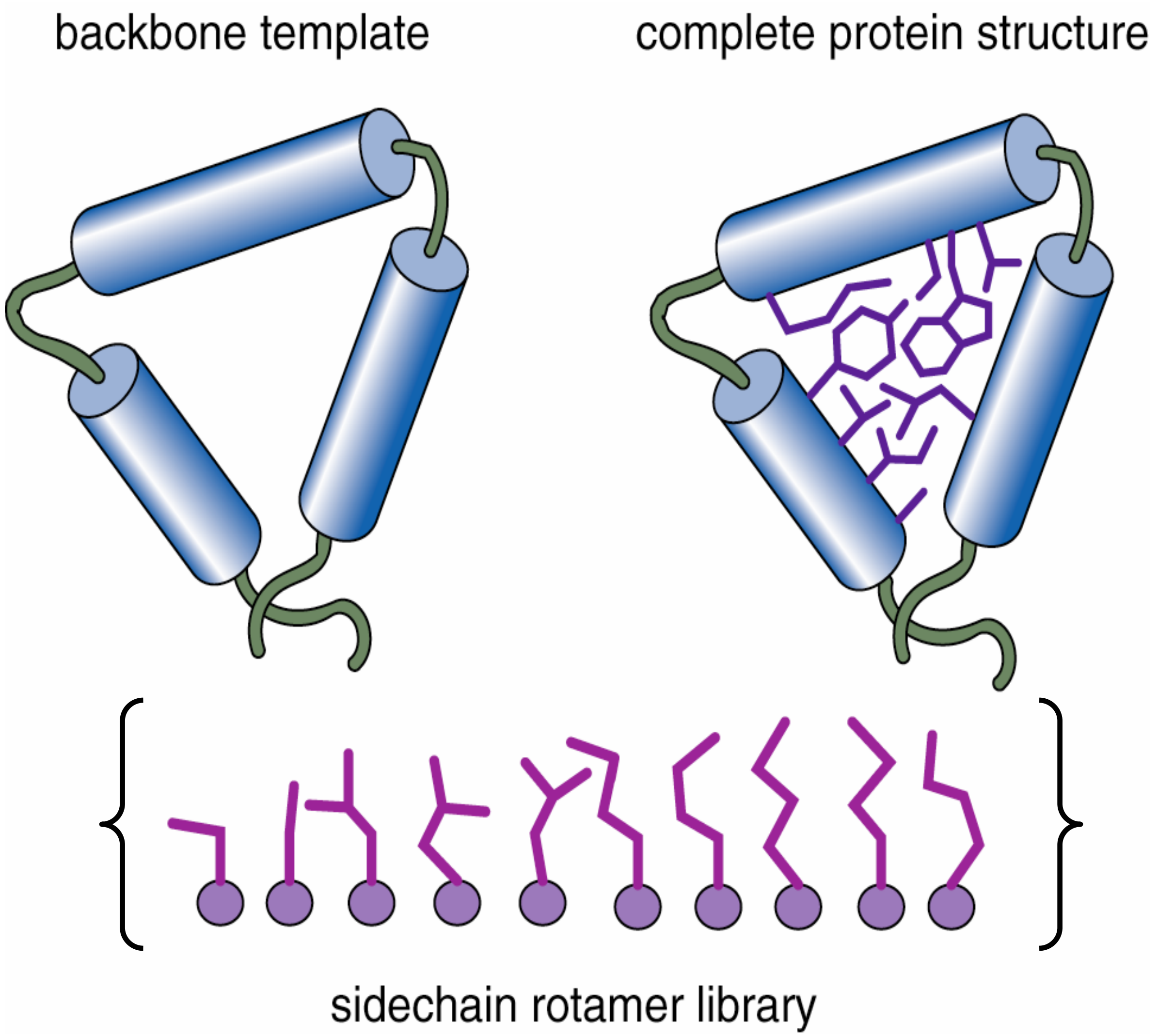
PRE-COMPUTE all of the single residue and pair energy terms

# Two big challenges in computational protein design:

1. SEARCH PROBLEM: There are many possible sequences: $20^N$
   *in general, these can’t be enumerated exhaustively*

2. ENERGY PROBLEM: To evaluate □$G^{fold}$ for a sequence we need to know the energy in the folded and unfolded states
   a. what is the structure of the folded state?
      *we know the backbone, but what about the side chain atoms?*
   b. how should we model the unfolded state?
   c. we know it is hard to measure the free energy, G
   d. we need a fast way to evaluate the energies because there are so many sequences to consider

# Generalizing the side chain packing problem to protein design



library now contains all choices of amino acids AND all choices of rotamers

# Complications when generalizing side-chain repacking to design

1. The conformational space gets MUCH larger
   $20^N$ -> ~$200^N$ (for a moderate rotamer library)

2. Note that for side-chain repacking the sequence didn't change, so the energy of the unfolded state was constant in our model. This is NOT the case when doing design.

   Now E must represent $E^{folded}$ - $E^{unfold}$ and we need to estimate how the energies of both the folded AND unfolded states change when you change the sequence.

THIS IS A HARD PROBLEM THAT HAS NOT BEEN GENERALLY SOLVED!

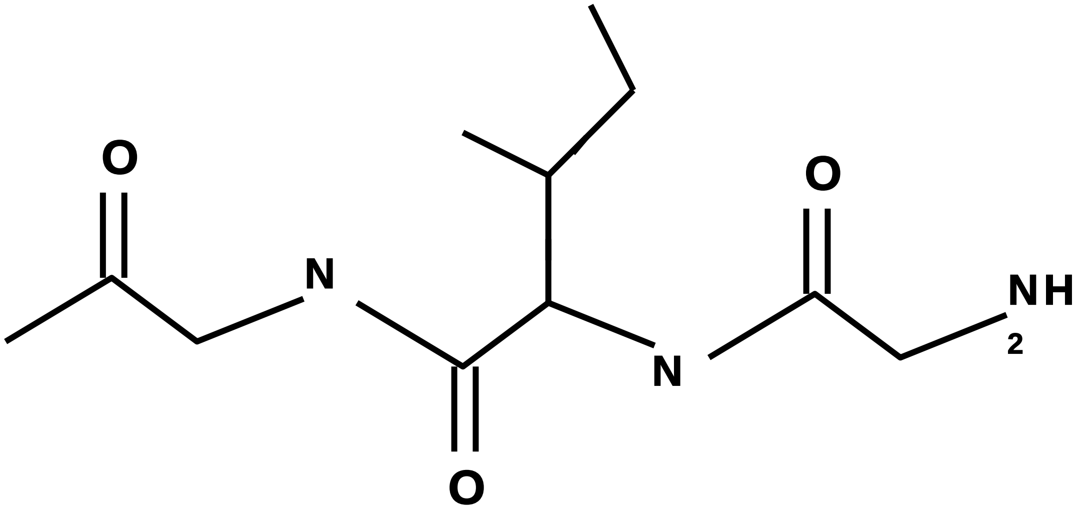
$$E_{total} = E_{fold} - E_{unfold}$$

$$= \Sigma_i E_{fold}(i_r) + \Sigma_i\Sigma_j E_{fold}(i_r,j_s) - \Sigma_i E_{unfold}(i_r) + \overbrace{\Sigma_i\Sigma_j E_{unfold}(i_r,j_s)}^{\text{assume } = 0}$$

$$= \Sigma_i\{E_{fold}(i_r) - E_{unfold}(i_r)\} + \Sigma_i\Sigma_j E_{fold}(i_r,j_s)$$

# Explicit model for the unfolded state

- tripeptide or pentapeptide
- linear backbone or some other structure
- lowest energy conformation for side chain or an average



- one such peptide for every residue
- no side chain-side chain interactions
- calculate the energy using the same terms as for the folded state
- *which energy terms change the most when you change amino acids?*

## An implicit model for the unfolded state - fit to some observable

$$E_{tot} = \Sigma_i\{E_{fold}(i_r) - E_{unfold}(i_r)\} + \Sigma_i\Sigma_j E_{fold}(i_r,j_s)$$

treat as an adjustable parameter, one per amino acid

*How to optimize it?*

One possibility: parameterize $E_{unfold}(i_r)$ so that the native residue is the recognized as the best when all residues are tested at a given site.

*What might not be optimal about this approach?*

See, eg.,

*Proc Natl Acad Sci U S A.* 97, no 19 (12 September 2000): 10383-8.

Erratum in: Kuhlman, B, and D Baker. "Native Protein Sequences are Close to Optimal for Their Structures." *Proc Natl Acad Sci U S A.* 97, no. 24 (21 November 2000): 13460.

# Dahiyat et al. design a zinc-less zinc finger

Please see Figure 2 of

Dahiyat, BI, and SL Mayo. "De novo Protein Design: Fully Automated Sequence Selection." *Science* 278, no. 5335 (3 October 1997): 82-7.

# Dahiyat et al. design a zinc-less zinc finger

NMR structure of FSD-1

Compare experimental and designed structures

Please see Figure 5 and 6 of

Dahiyat, BI, and SL Mayo. "De novo Protein Design: Fully Automated Sequence Selection." *Science* 278, no. 5335 (3 October 1997): 82-7.

# Kuhlman et al. design a new protein fold “Top7”

Please see figure 1 of

Kuhlman, B, G Dantas, GC Ireton, G Varani, BL Stoddard, and D Baker. "Design of A Novel Globular Protein Fold with Atomic-level Accuracy." *Science* 302, no. 5649 (21 November 2003): 1364-8.

## Kuhlman et al. design a new protein fold “Top7”

Science 302, 1364 (2003)

Critical to their success:
iterative use of design and prediction using ROSETTA

1. Choose starting backbones (172 of them)
2. Design sequence to fit backbone
3. Relax the backbone to fit the sequence
4. Iterate

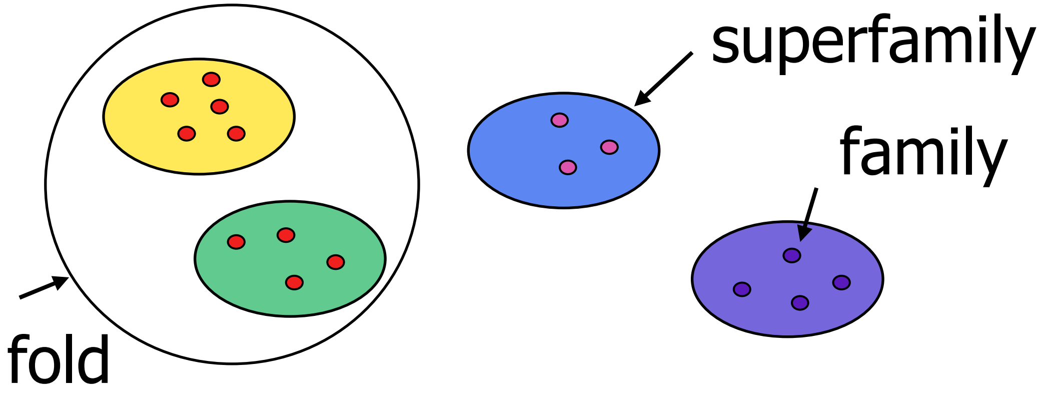
*Why do you need backbone relaxation?*

# X-ray structure of “Top7” compared with the design

Please see figure 4 of

Kuhlman, B, G Dantas, GC Ireton, G Varani, BL Stoddard, and D Baker. "Design of A Novel Globular Protein Fold with Atomic-level Accuracy." *Science* 302, no. 5649 (21 November 2003): 1364-8.

# Why might protein <u>design</u> be easier than ab initio protein <u>folding</u>?



1. There are more correct answers.

2. Come up with a design that <u>exploits those principles that you understand best</u> to design the properties you want into a protein (e.g. hydrophobic packing).

3. In design, you try to make all interactions as good as possible, and hope that this avoids computing subtle tradeoffs between different energy terms.

4. More control over the problem - choose an easy goal!