Comparative Modelling

Summary

1. Basic concepts of Homology Modeling
2. Schema of the method
    1. Fold assignment
    2. Template selection
    3. Model building
    4. Evaluation
    5. Improvement

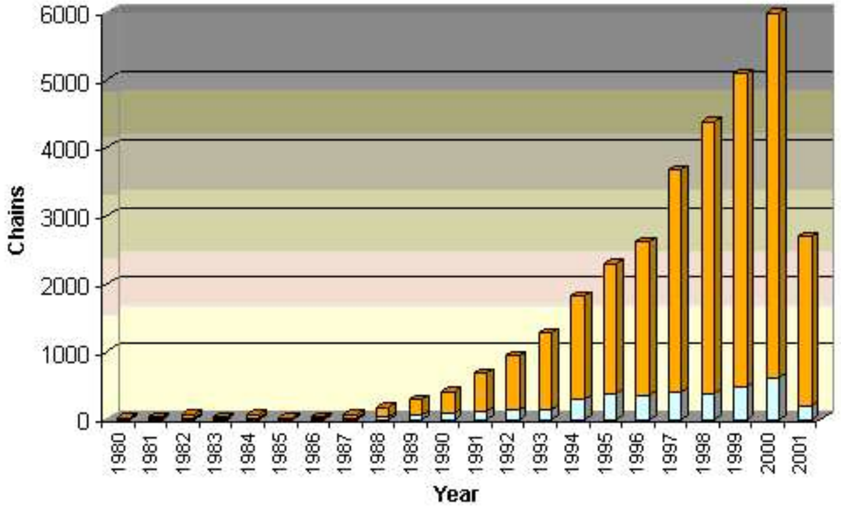
1. Basic concepts of Homology Modeling

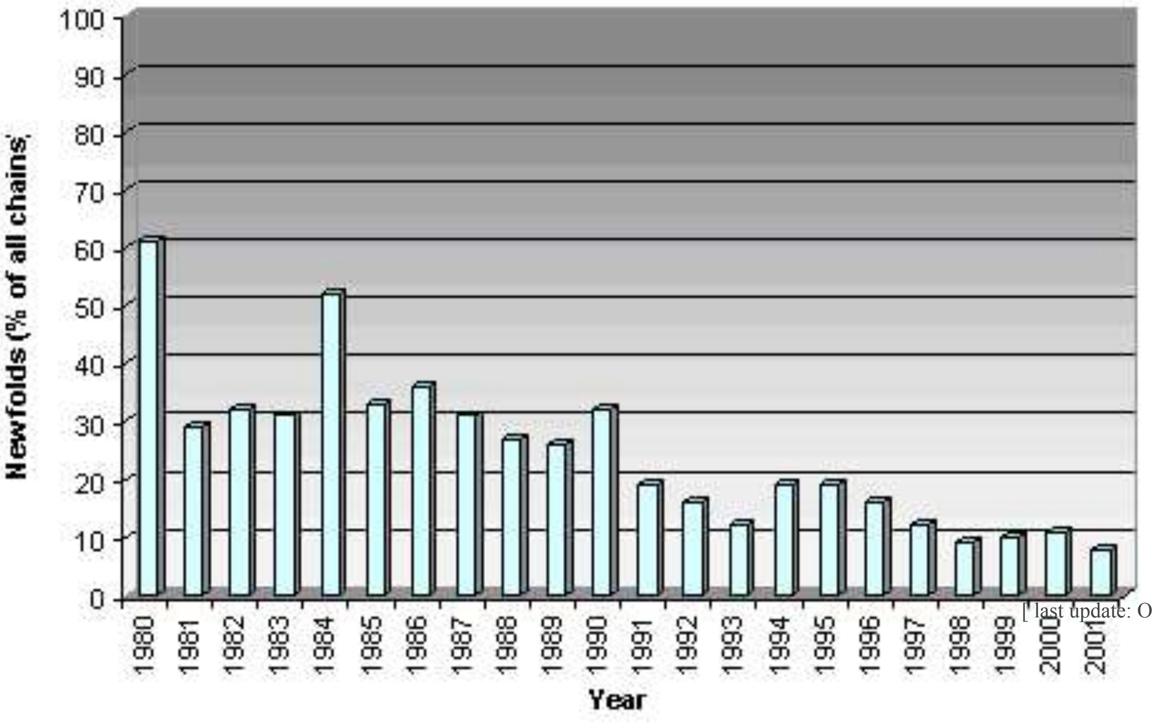
Definition

Extrapolation of the structure for a new (target) sequence from the known 3D-structures of related family members (templates).

1. Basic concepts of Homology Modeling

The number of different protein folds is limited:

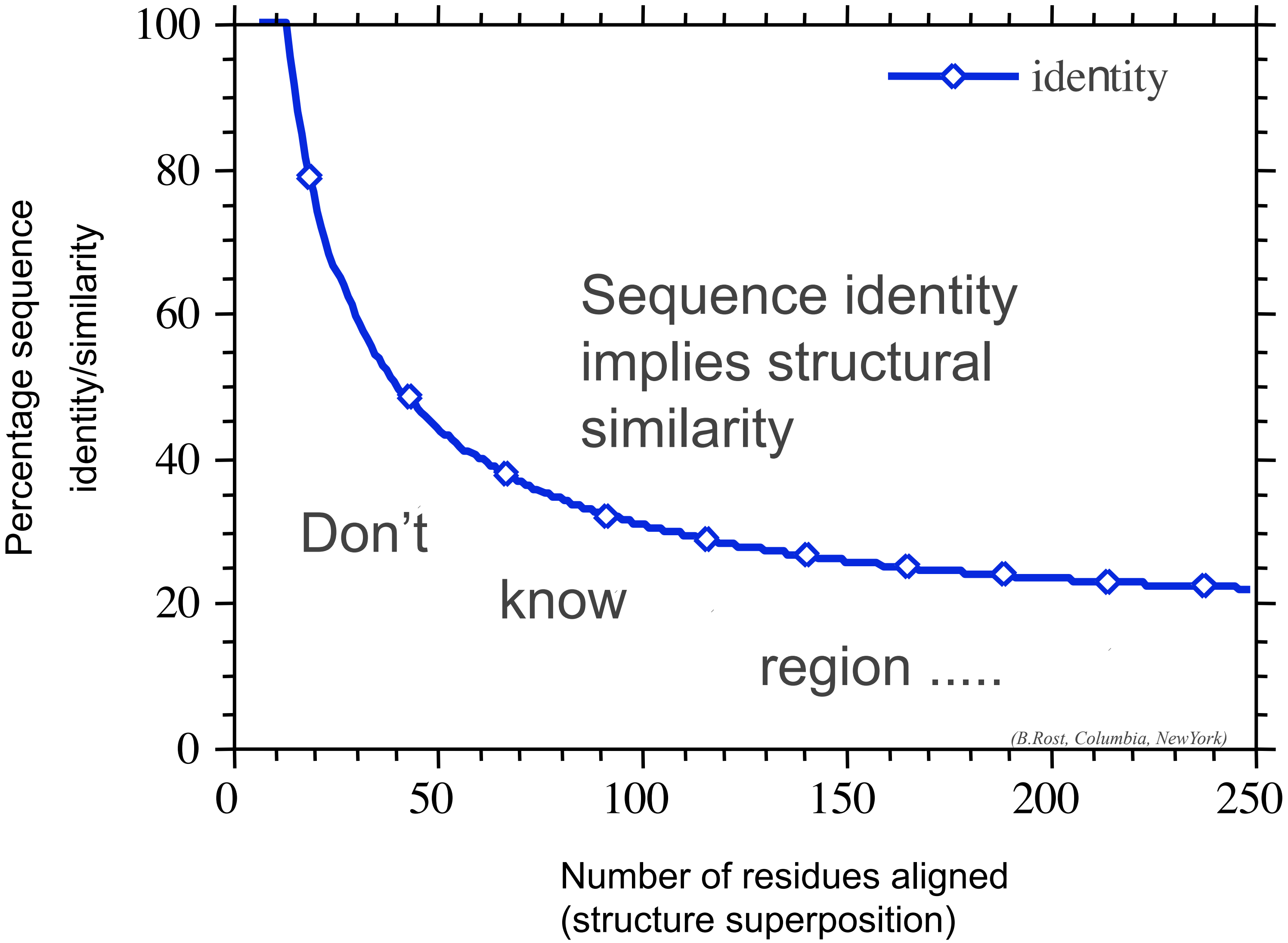




[last update: Oct 2001]

1. Basic concepts of Homology Modeling

Sequence similarity implies structural similarity?

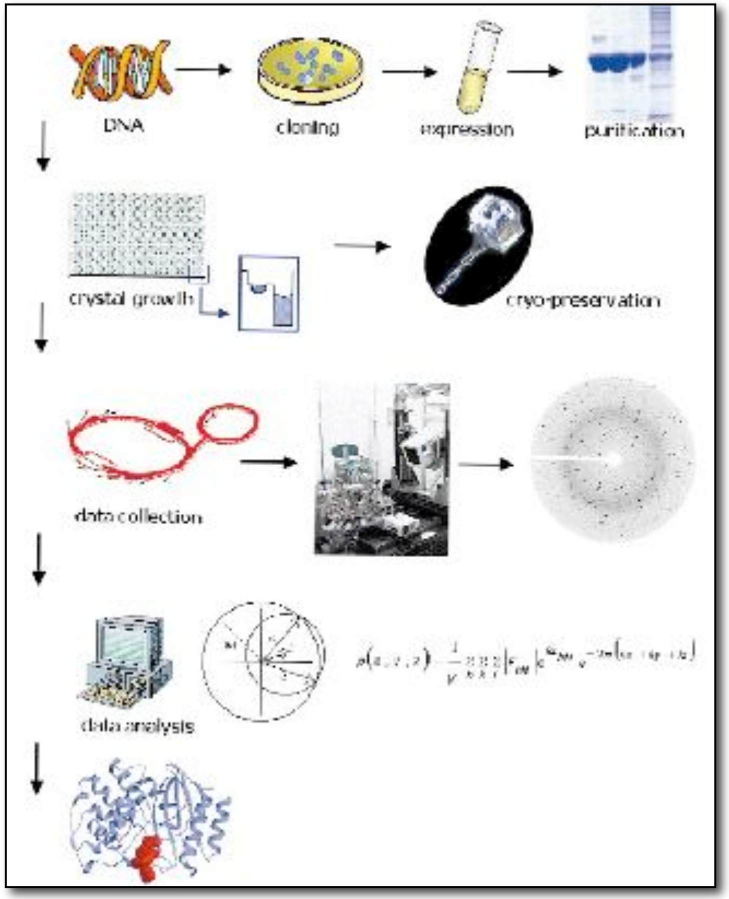


(B.Rost, Columbia, NewYork)

1. Basic concepts of Homology Modeling

- Fold is more conserved than sequence.
- Secondary structure are the most conserved parts
- Loops have the higher variability in structure.

1. Basic concepts of Homology Modeling
Structural Genomics



express & purify

cristallize

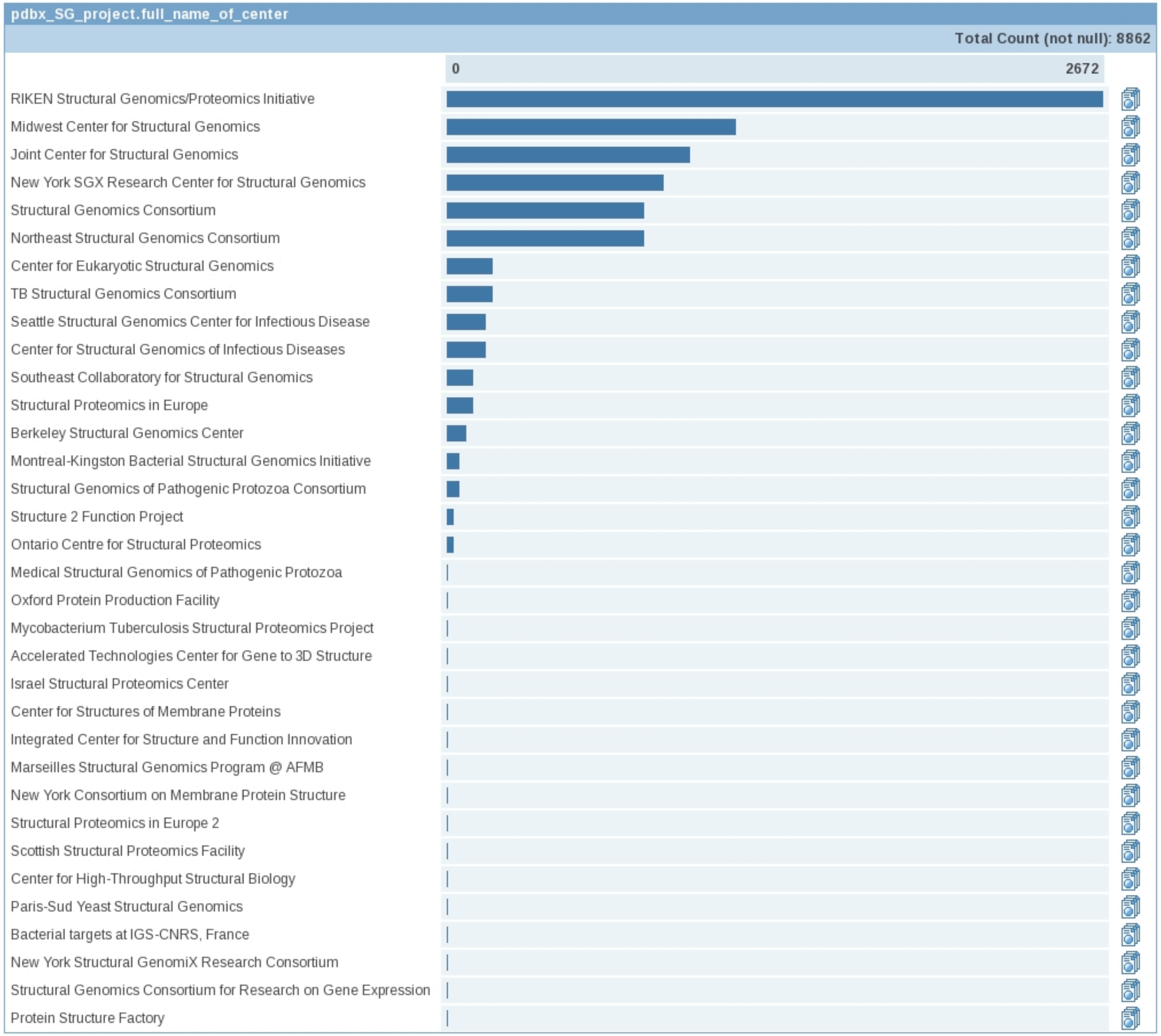
X-ray

analises

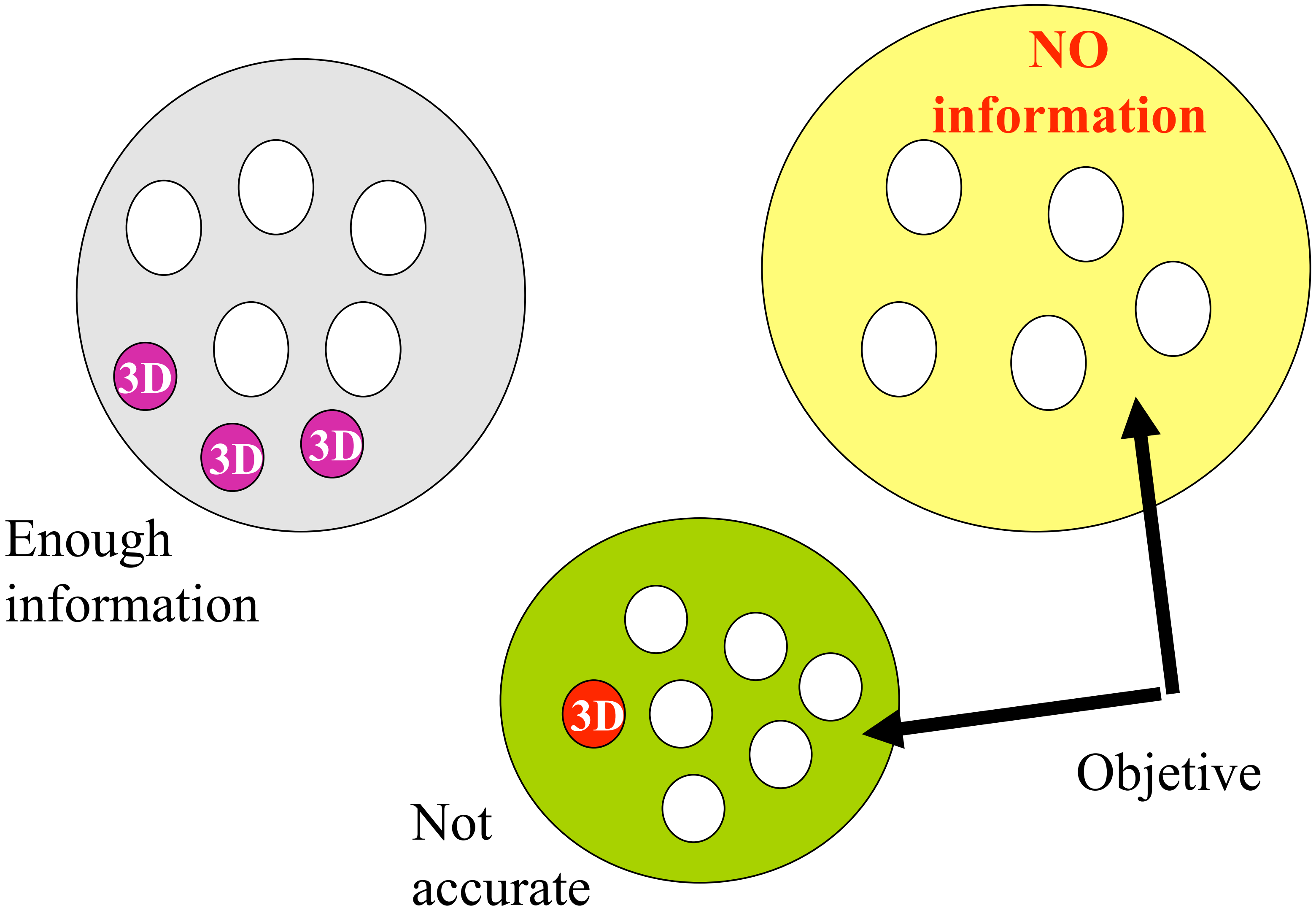
structure

1. Basic concepts of Homology Modeling

Structural Genomics

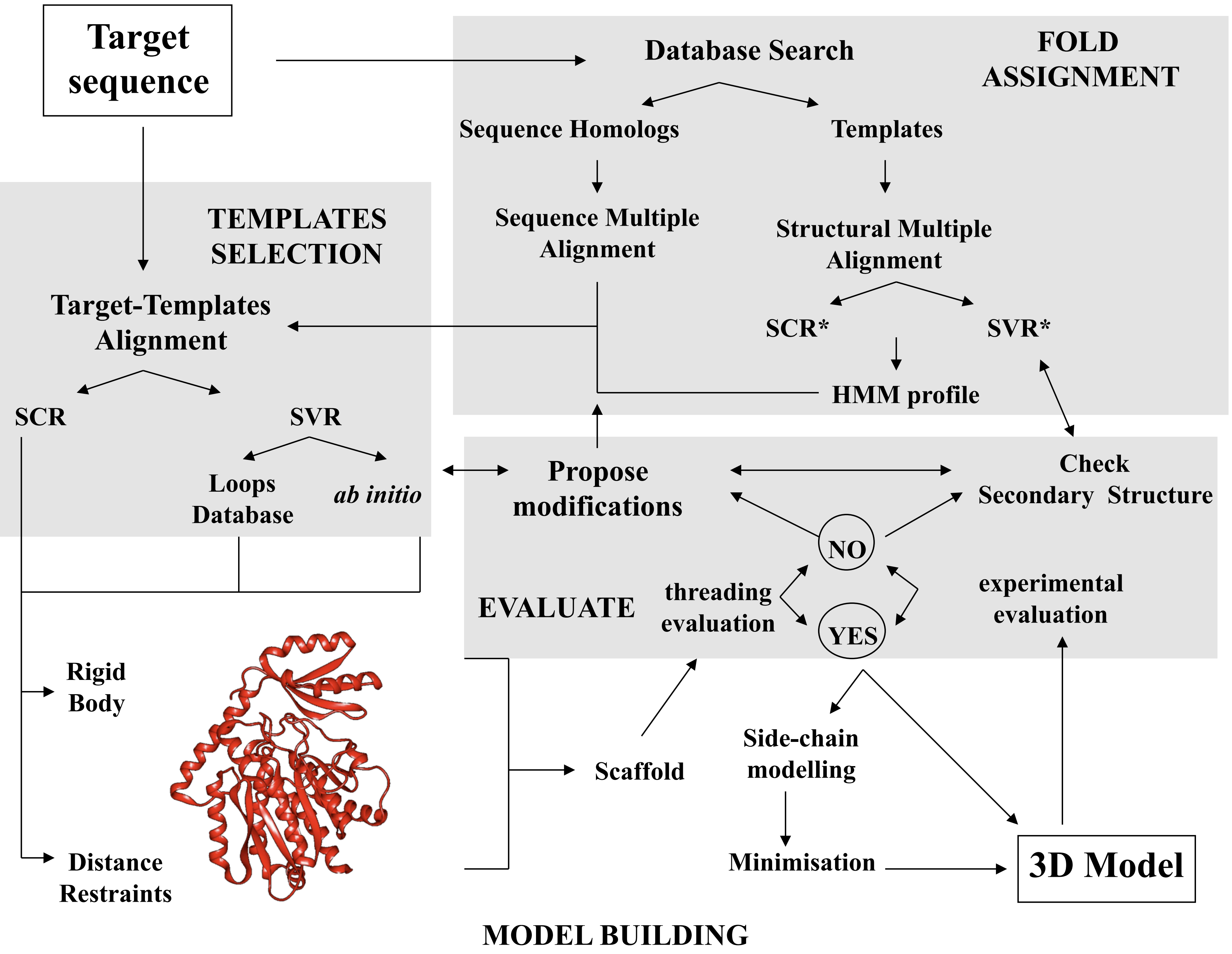


1. Basic concepts of Homology Modeling
Structural Genomics



2. Schema of the method

1. Fold assignment
2. Template selection
3. Model building
4. Evaluation
5. Improvement



Target sequence

FOLD ASSIGNMENT

Database Search

Sequence Homologs

Templates

Sequence Multiple Alignment

Structural Multiple Alignment

SCR*

SVR*

HMM profile

TEMPLATES SELECTION

Target-Templates Alignment

SCR

SVR

Loops Database

ab initio

Propose modifications

Check Secondary Structure

NO

YES

EVALUATE

threading evaluation

experimental evaluation

Rigid Body

Distance Restraints

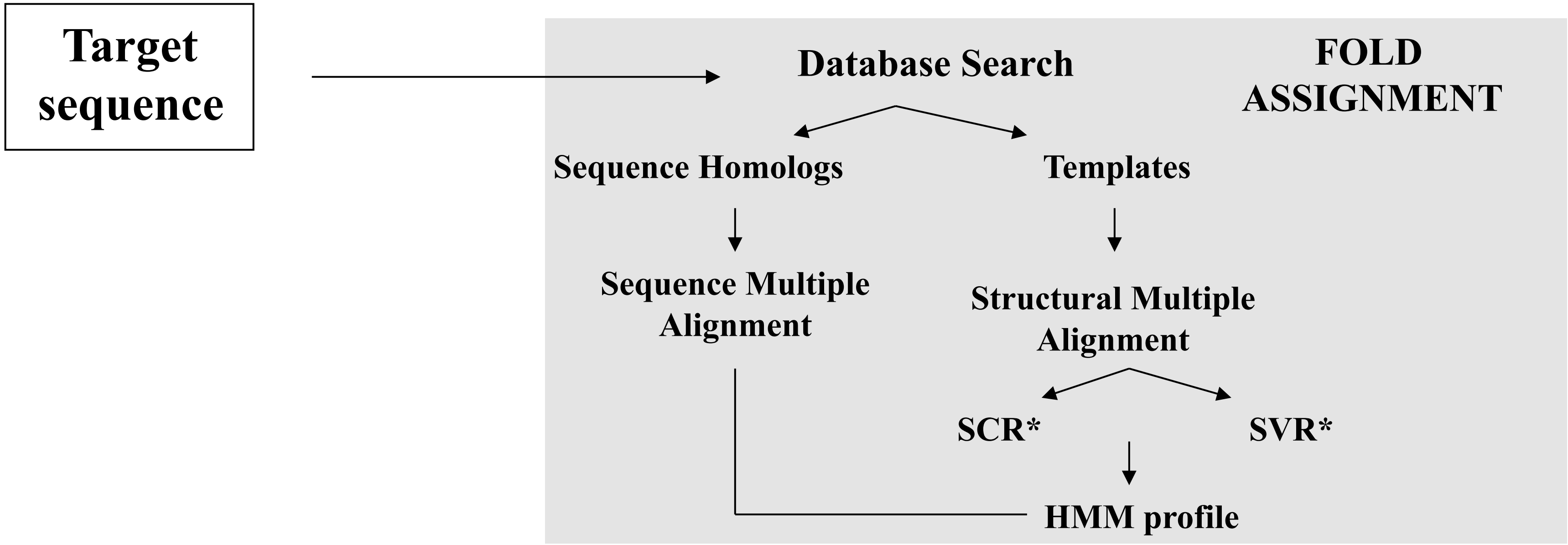
Scaffold

Side-chain modelling

Minimisation

3D Model

MODEL BUILDING



Target sequence

FOLD ASSIGNMENT

Database Search

Sequence Homologs

Templates

Sequence Multiple Alignment

Structural Multiple Alignment

SCR*

SVR*

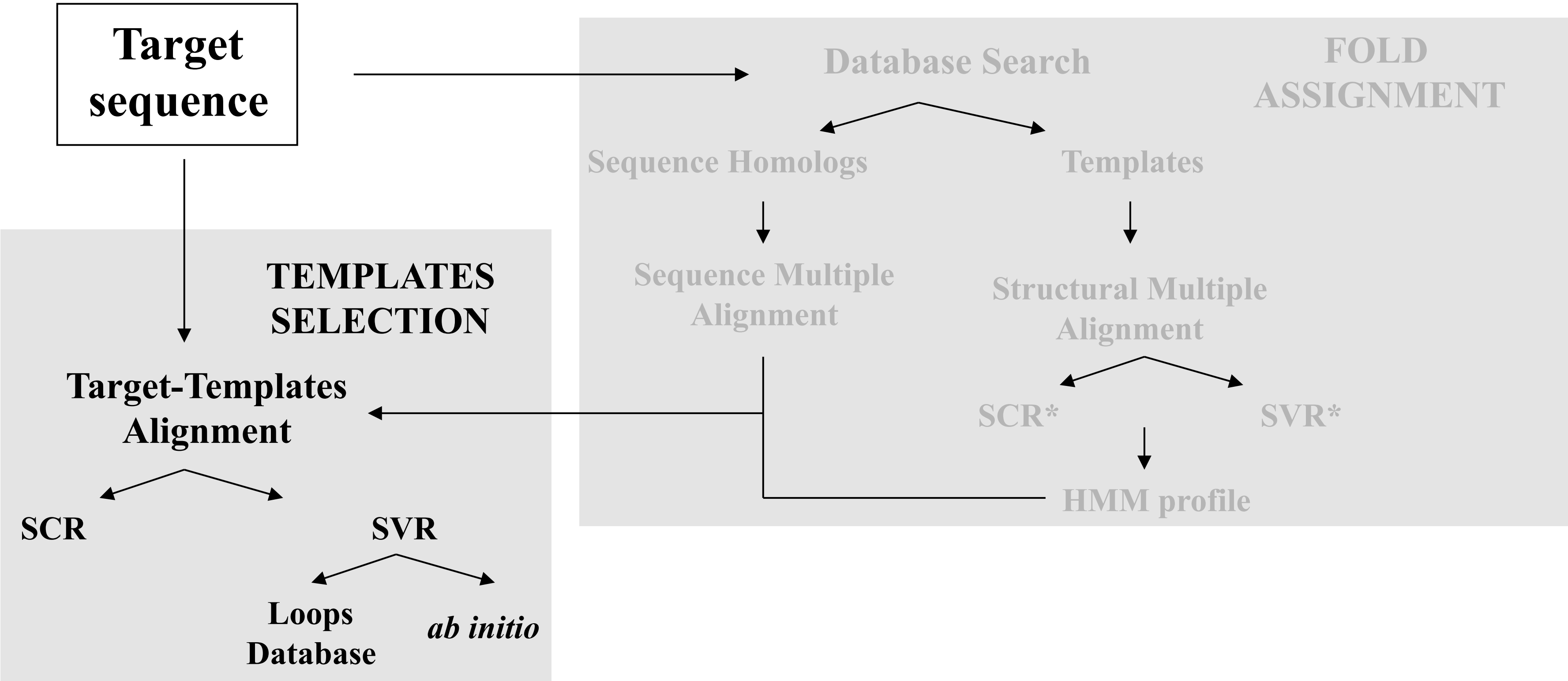
HMM profile

2. Schema of the method

1. Fold assignment

Sequence search with the target

1. Compares the sequence of the target with a set of sequences with known structure
2. Ranking the comparisons by scores.
3. Scores are related to P-values or E-values (high score implies low P-value). P-value is the probability of obtaining the same alignment by chance.
4. Scores are calculated using a residue-substitution matrix:
    1. PAM: based on the alignment of sequences of homologs
    2. BLOSUM: based on the alignment of blocs of similar sequences
5. One sequence can have more than one domain, therefore we can obtain the best scores for partial parts of the target.
6. Methods (see practice)
    1. BLAST algorithm, matches words from a pre-calculated and indexed set and joints them into sentences (forming the sequence)
    2. FastA: Smith & Waterman algorithm
    3. Scanning PFAM: algorithm of Hidden Markov Models


Target sequence
Database Search
FOLD ASSIGNMENT
Sequence Homologs
Templates
Sequence Multiple Alignment
Structural Multiple Alignment
SCR*
SVR*
HMM profile
TEMPLATES SELECTION
Target-Templates Alignment
SCR
SVR
Loops Database
ab initio

2. Schema of the method

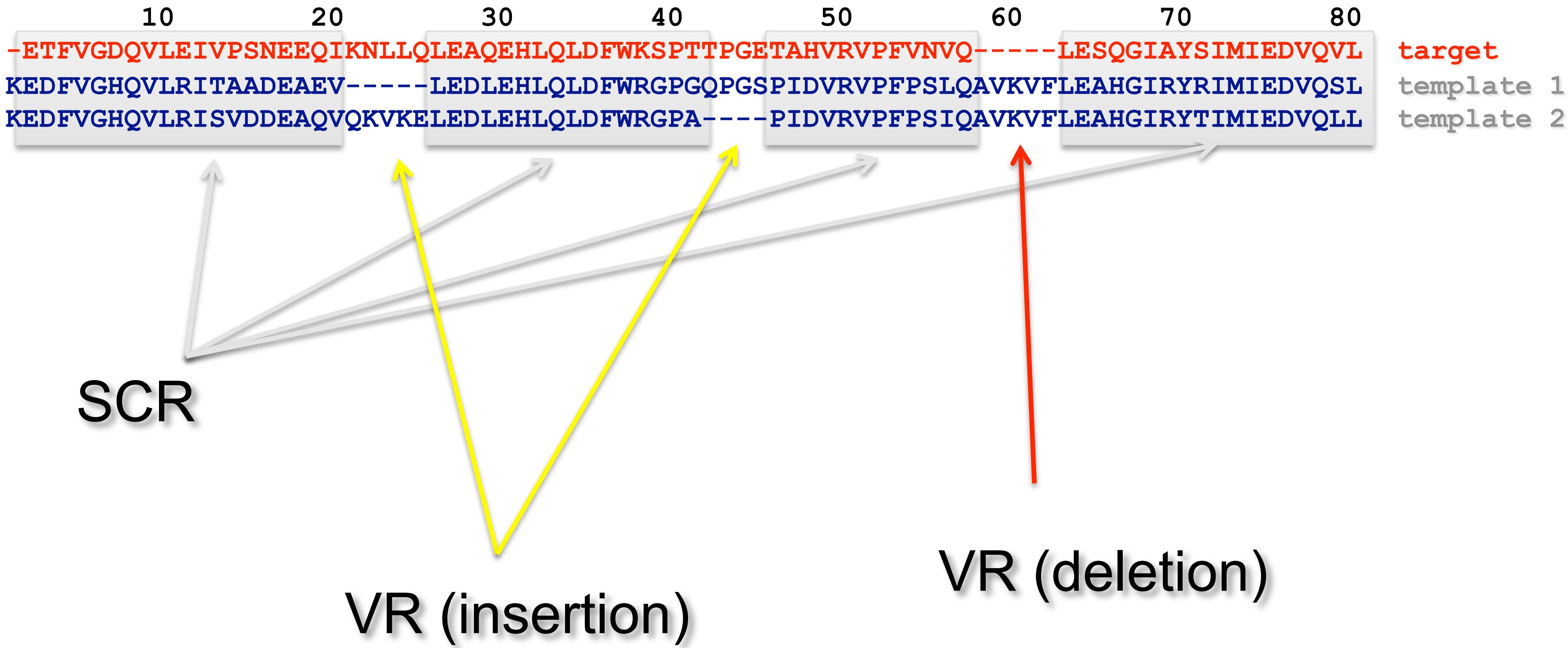
2. Template selection

Selecting the best target-alignment template

1. The template(s) should be the closest homolog(s) to the target
2. Small number of templates to avoid stress on model building
3. Multi-domain proteins require the use of at least one template with the largest coverage of sequence (containing the largest number of domains)
4. Structural alignment of homologs gives the information on position-specific substitutions
5. Detection of structurally conserved regions (SCR) and variable regions (VR)
6. Aligning the target sequence and template sequences using a multiple sequence profile helps to avoid misalignments
7. Methods (see practice)
    1. ClustalW
    2. T-coffee
    3. HMMER
        1. alignment with a known family profile (PFAM)
        2. Alignment with a profile built with the structure of homologs

2. Schema of the method

2. Template selection



```
        10        20        30        40        50        60        70        80
-ETFVGDQVLEIVPSNEEQIKNLLQLEAQEHLQLDFWKSPTTPGETAHVRVPFVNVQ-----LESQGIAYSIMIEDVQVL  target
KEDFVGHQVLRITAADEAEV-----LEDLEHLQLDFWRGPGQPGSPIDVRVPFPSLQAVKVFLEAHGIRYRIMIEDVQSL  template 1
KEDFVGHQVLRISVDDEAQVQKVKELEDLEHLQLDFWRGPA----PIDVRVPFPSIQAVKVFLEAHGIRYTIMIEDVQLL  template 2
```

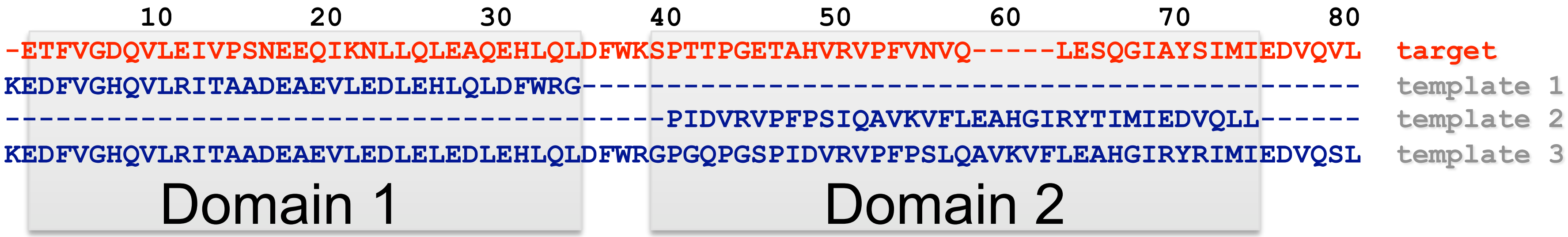
SCR

VR (insertion)

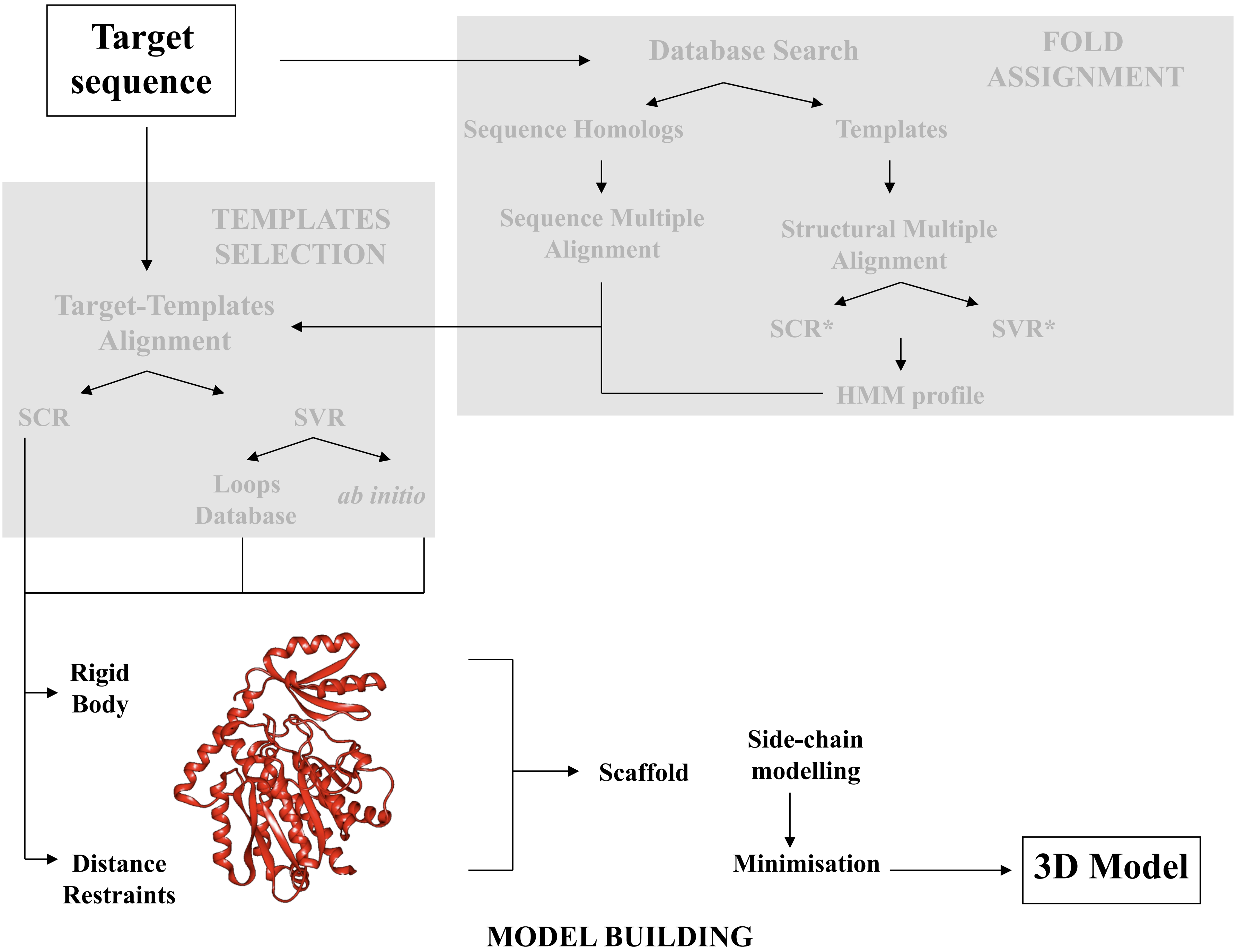
VR (deletion)

2. Schema of the method

2. Template selection



```
        10        20        30        40        50        60        70        80
-ETFVGDQVLEIVPSNEEQIKNLLQLEAQEHLQLDFWKSPTTPGETAHVRVPFVNVQ-----LESQGIAYSIMIEDVQVL  target
KEDFVGHQVLRITAADEAEVLEDLEHLQLDFWRG----------------------------------------------  template 1
---------------------------------------PIDVRVPFPSIQAVKVFLEAHGIRYTIMIEDVQLL------  template 2
KEDFVGHQVLRITAADEAEVLEDLELEDLEHLQLDFWRGPGQPGSPIDVRVPFPSLQAVKVFLEAHGIRYRIMIEDVQSL  template 3
```


Target sequence
Database Search
FOLD ASSIGNMENT
Sequence Homologs
Templates
Sequence Multiple Alignment
Structural Multiple Alignment
SCR*
SVR*
HMM profile
TEMPLATES SELECTION
Target-Templates Alignment
SCR
SVR
Loops Database
ab initio
Rigid Body
Distance Restraints
Scaffold
Side-chain modelling
Minimisation
3D Model
MODEL BUILDING

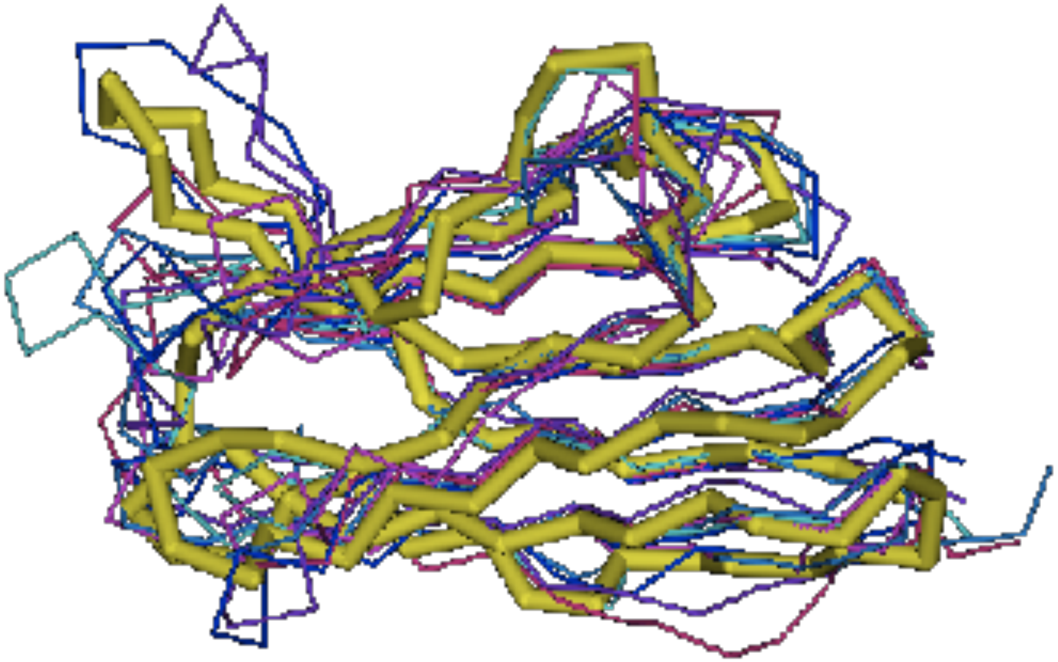
2. Schema of the method

3. Model building

1. Rigid Body Assembly
   1. Core framework (SCR)
   2. Loop modeling (VR)
   3. Energy minimization
2. Spatial restraints
   1. Probability Density Functions (PDF)
   2. Distance restraints
   3. Simulated Annealing
   4. Loop modeling
3. Side-chain modeling
   1. Back-bone dependent rotamer libraries
   2. Energetic and packing criteria

2. Schema of the method

3. Model building: Rigid Body Assembling (core framework)



- Averaging core template backbone atoms

  (weighted by local sequence similarity with the target sequence)

- Leave non-conserved regions (loops) for later ….

2. Schema of the method

3. Model building: Rigid Body Assembling (loop modeling)

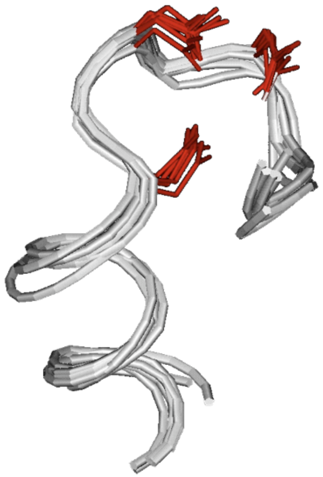
1. Use the “spare part” algorithm to find compatible fragments in a Loop-Database

2. “*ab-initio*” rebuilding of loops (Monte Carlo, molecular dynamics, genetic algorithms, etc.)

2. Schema of the method

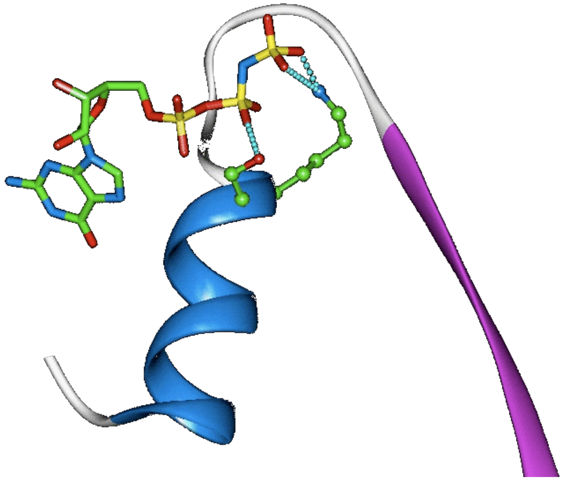
3. Model building: Rigid Body Assembling (loop modeling)

1. Use the “spare part” algorithm to find compatible fragments in a Loop-Database



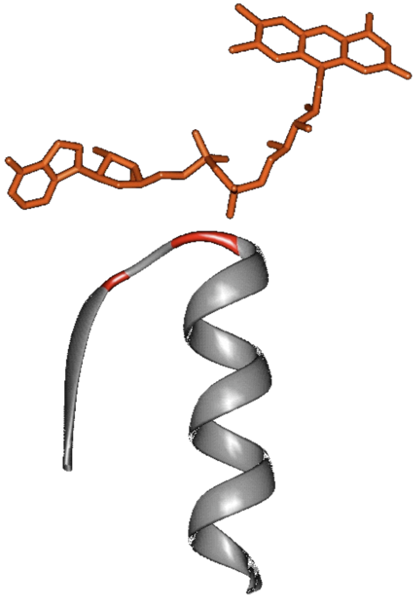
EF-Hand Calcium binding

aa{baalal}bb
Xh{DXDpDG}Xh



P-loop GTP binding

bb{eppgag}aa
hh{GhXXpG}Kp



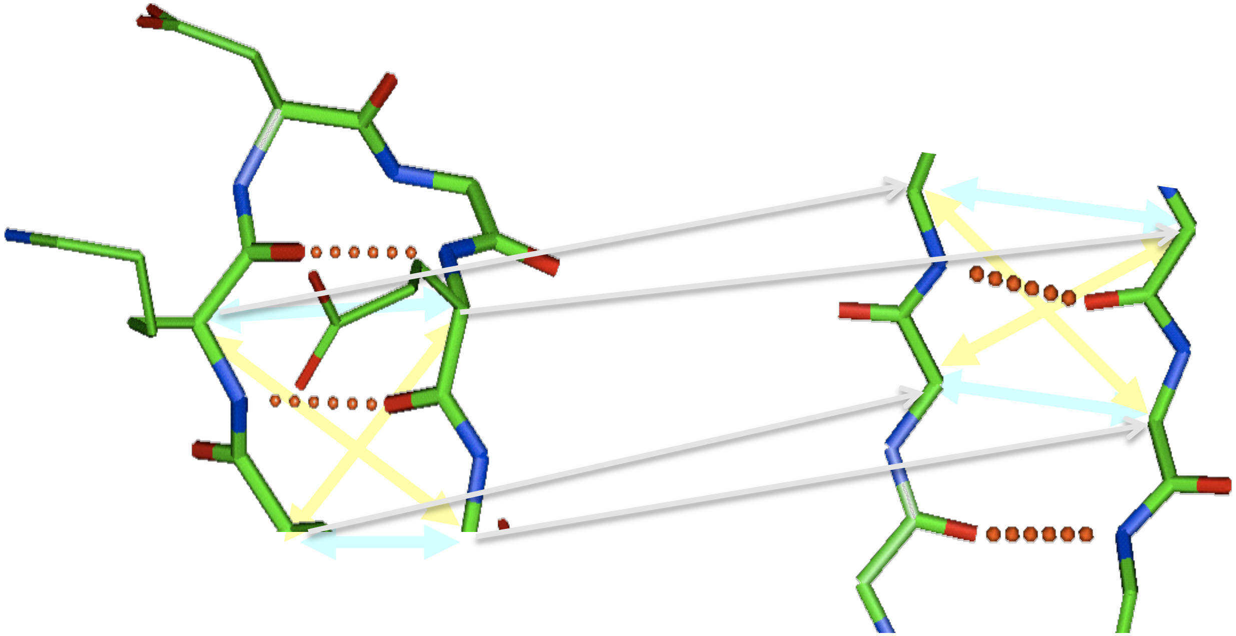
NAD(P)/FAD binding

bb{eab}aa
hh{GhG}hX

2. Schema of the method

3. Model building: Rigid Body Assembling (loop modeling)

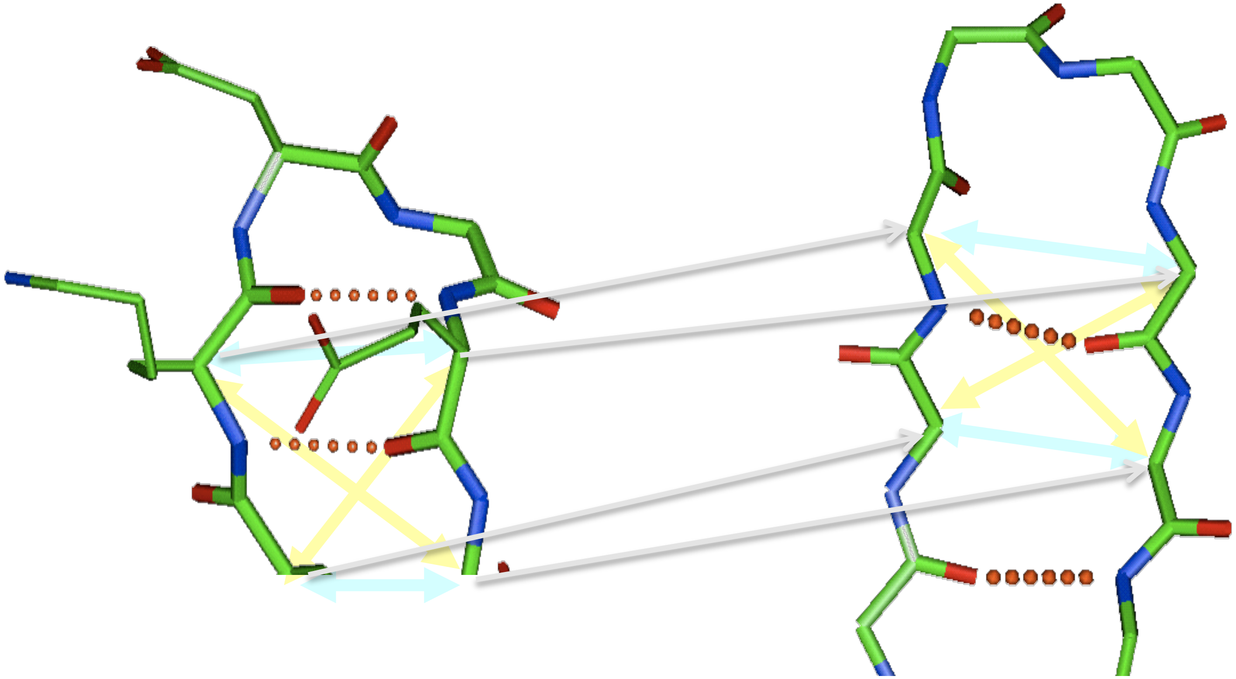
1. Use the “spare part” algorithm to find compatible fragments in a Loop-Database



2. Schema of the method

3. Model building: Rigid Body Assembling (loop modeling)

1. Use the “spare part” algorithm to find compatible fragments in a Loop-Database



2. Schema of the method

3. Model building: Rigid Body Assembling (Energy minimization)

$$E_{bonding} = \sum_{bonds} \frac{1}{2} k_i \left(d_i - d_i^0 \right)^2 + \sum_{angles} \frac{1}{2} k_j \left(\alpha_j - \alpha_j^0 \right)^2 + \sum_{\substack{improper \\ dihedral}} \frac{1}{2} k_n \left(\omega_n - \omega_n^0 \right)^2 + \sum_{angles} E_m Cos\left(\omega_m \phi_m + \varphi_m \right)^2$$

$$E_{non-bonding} = \frac{1}{4\pi\varepsilon_0} \sum_i \sum_{j>i} \frac{q_i q_j}{r_{ij}} + \sum_i \sum_{j>i} \frac{C_6^{ij}}{r_{ij}^6} - \frac{C_{12}^{ij}}{r_{ij}^{12}}$$

$$E = E_{bonding} + E_{non-bonding}$$

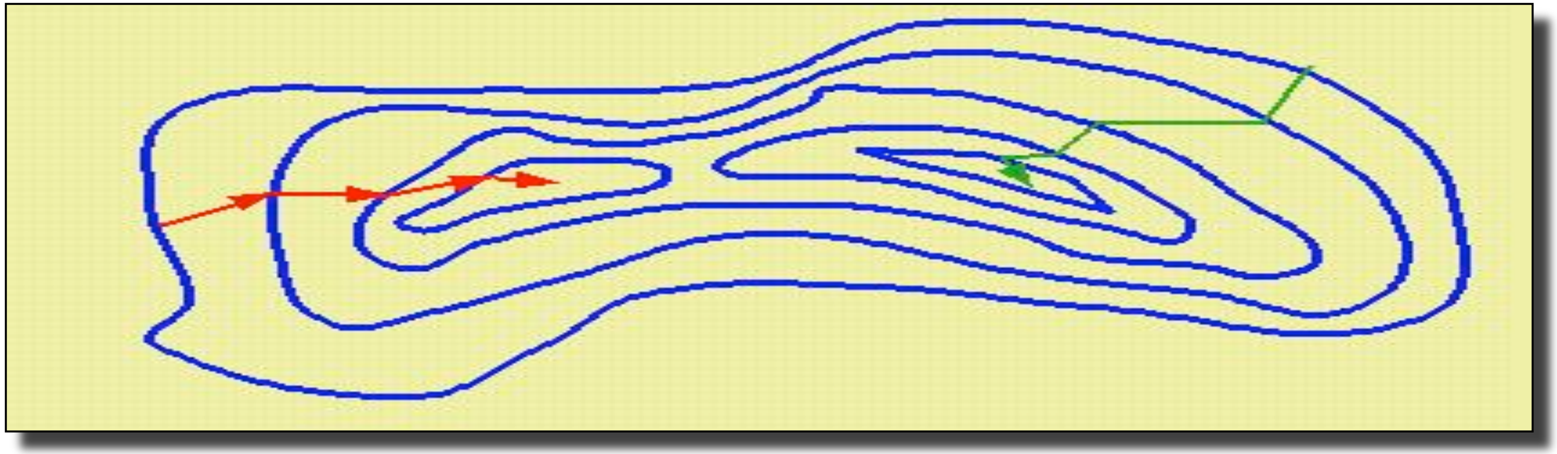
- modeling will produce unfavorable contacts and bonds: idealization of local bond and angle geometry

- extensive energy minimization will move coordinates away: keep it to a minimum

- Methods: Newton Rapson; Steepest Descent; Conjugate Gradient

2. Schema of the method

3. Model building: Rigid Body Assembling (Energy minimization)



$$x_{i+1} = x_i + \lambda \nabla E$$

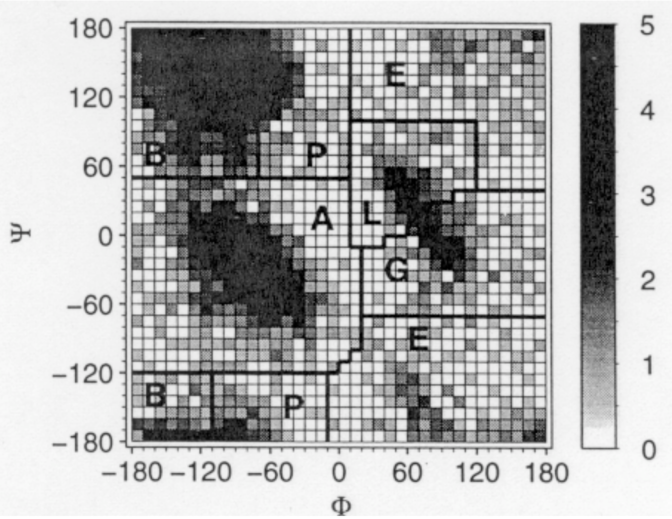
$$\lambda = \begin{cases} E(x_{i+1}) < E(x_i) \Rightarrow \lambda = \lambda + \varepsilon \\ E(x_{i+1}) > E(x_i) \Rightarrow \lambda = \lambda/2 \\ \lambda < \lambda_{\max} \\ E(x_{i+1}) \approx E(x_i) \Rightarrow STOP \end{cases}$$

2. Schema of the method

3. Model building: Spatial restraints (Probability Density Functions)

Feature properties can be associated with

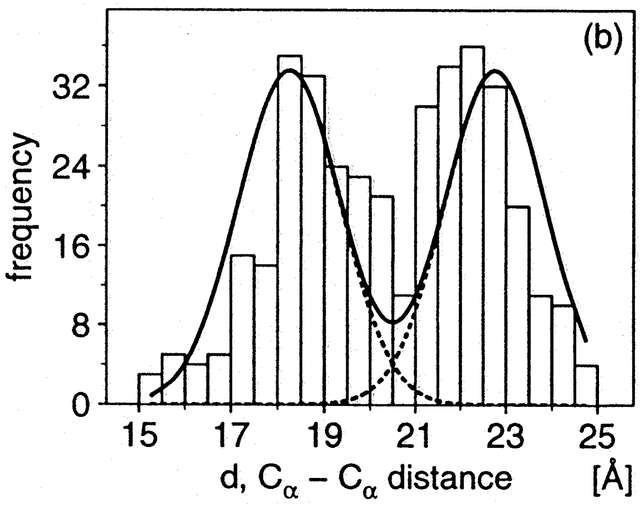
- a protein (e.g. X-ray resolution)
- residues (e.g. solvent accessibility)
- pairs of residues (e.g. C_a - C_a distance)
- other features (e.g. main chain classes)



Example: Ramachandran Plot
Distribution of (ϕ,ψ) angles

2. Schema of the method

3. Model building: Spatial restraints (Probability Density Functions)



Example:
Distribution of $C\alpha - C\alpha$ distances

How can we derive modeling restraints from this data?

A restraint is defined as probability density function (*pdf*), p(x):

$$p(x1 \leq x < x2) = \int_{x2}^{x1} p(x)dx$$

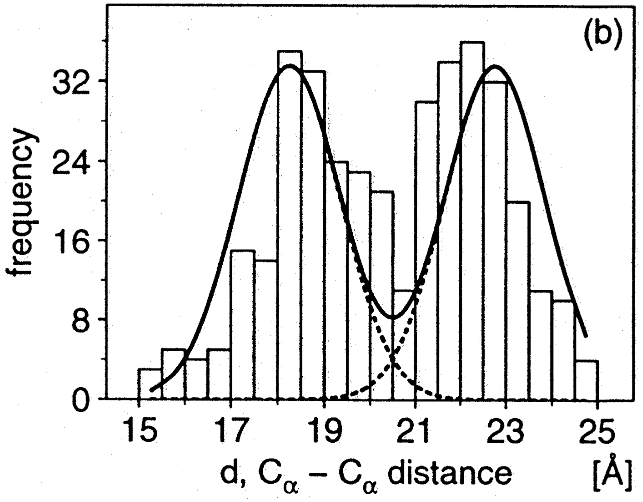
with

$$\int p(x)dx = 1$$

$$p(x) > 0$$

2. Schema of the method

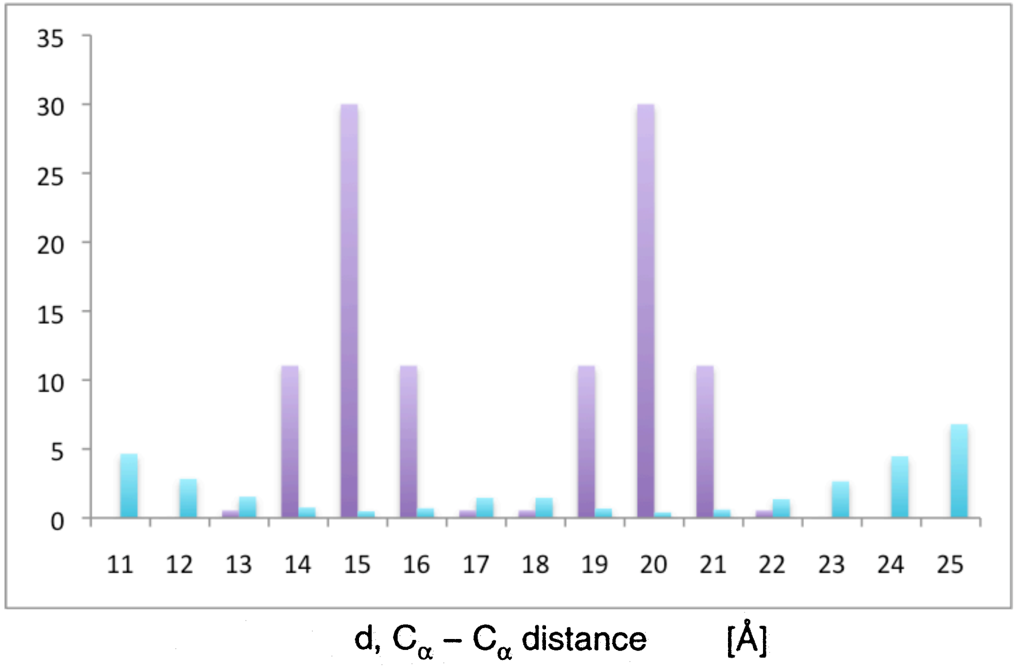
3. Model building: Spatial restraints (Probability Density Functions)



Example:
Distribution of Cα−Cα distances

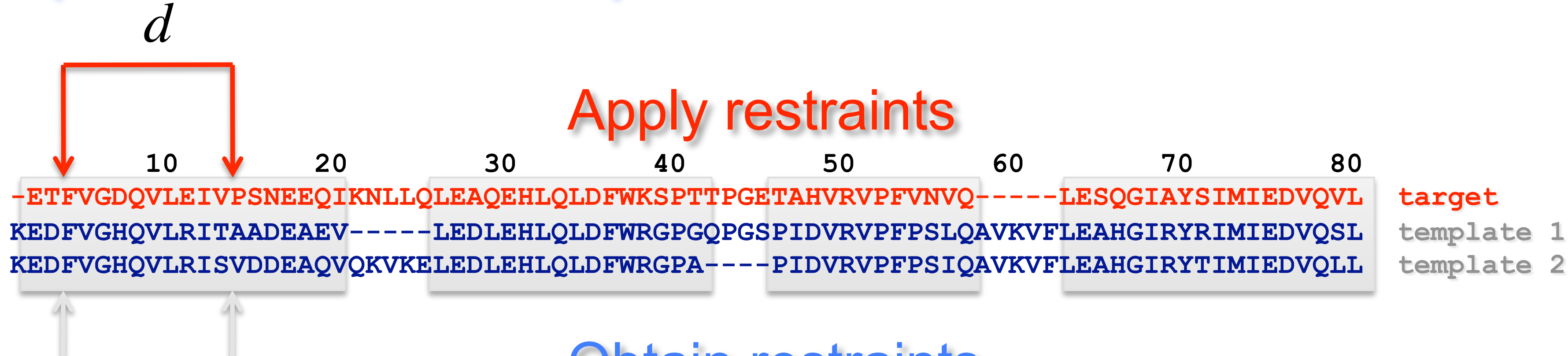
How can we derive modeling restraints from this data?

$$E_{pdf}(x) = -RT\log(p(x))$$



2. Schema of the method

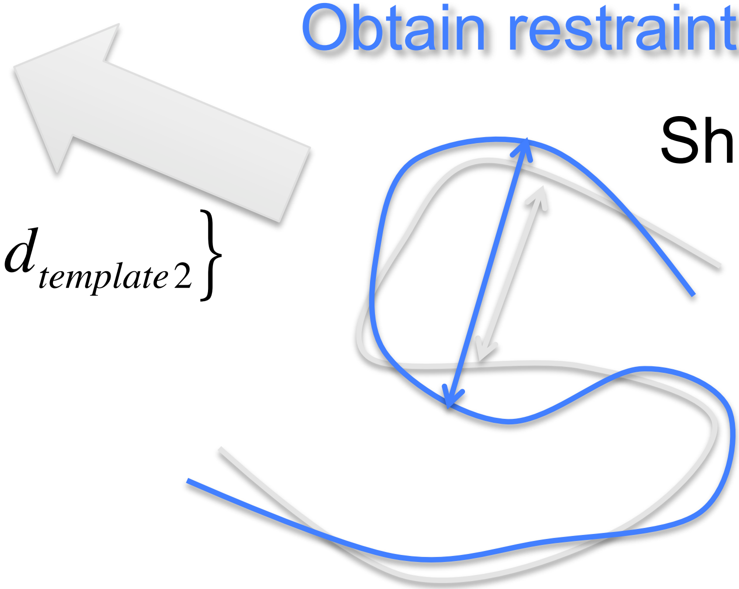
3. Model building: Spatial restraints (Distance restraints)



```
           10        20        30        40        50        60        70        80
-ETFVGDQVLEIVPSNEEQIKNLLQLEAQEHLQLDFWKSPTTPGETAHVRVPFVNVQ-----LESQGIAYSIMIEDVQVL  target
KEDFVGHQVLRITAADEAEV-----LEDLEHLQLDFWRGPGQPGSPIDVRVPFPSLQAVKVFLEAHGIRYRIMIEDVQSL  template 1
KEDFVGHQVLRISVDDEAQVQKVKELEDLEHLQLDFWRGPA----PIDVRVPFPSIQAVKVFLEAHGIRYTIMIEDVQLL  template 2
```

Obtain restraints

d_1 & d_2



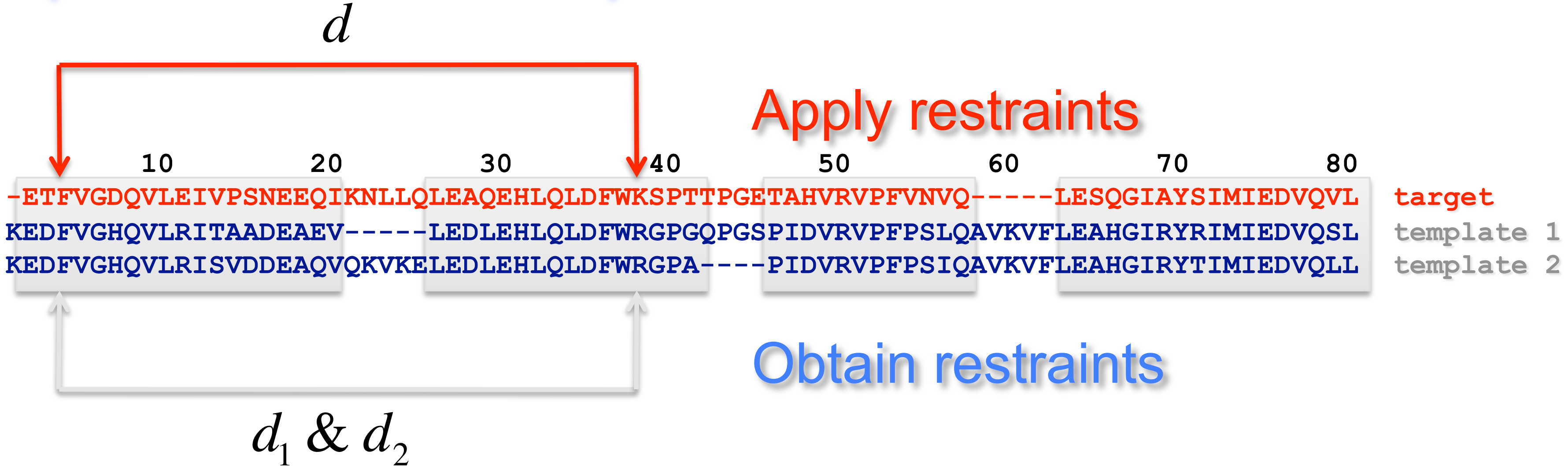
Short distance restraints

$$d \in \left\{ d_{template1}; d_{template2} \right\}$$

$$d_1 < d < d_2$$

2. Schema of the method

3. Model building: Spatial restraints (Distance restraints)



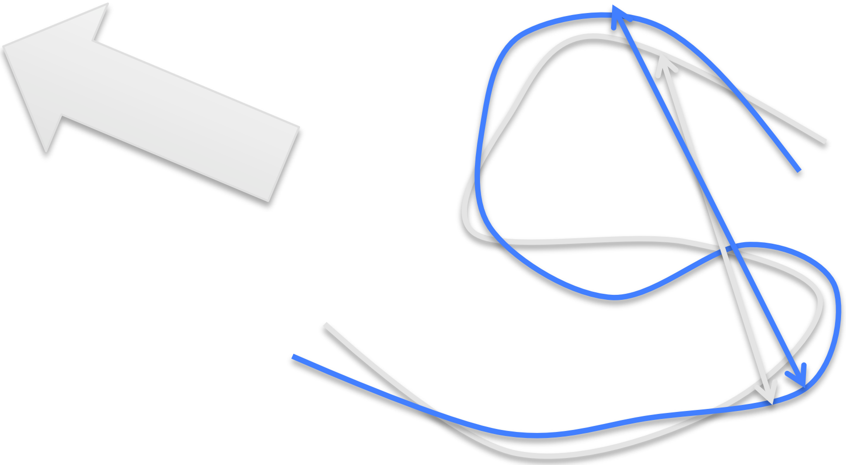
d

Apply restraints

```
         10        20        30        40        50        60        70        80
-ETFVGDQVLEIVPSNEEQIKNLLQLEAQEHLQLDFWKSPTTPGETAHVRVPFVNVQ-----LESQGIAYSIMIEDVQVL  target
KEDFVGHQVLRITAADEAEV-----LEDLEHLQLDFWRGPGQPGSPIDVRVPFPSLQAVKVFLEAHGIRYRIMIEDVQSL  template 1
KEDFVGHQVLRISVDDEAQVQKVKELEDLEHLQLDFWRGPA----PIDVRVPFPSIQAVKVFLEAHGIRYTIMIEDVQLL  template 2
```

Obtain restraints

d_1 & d_2



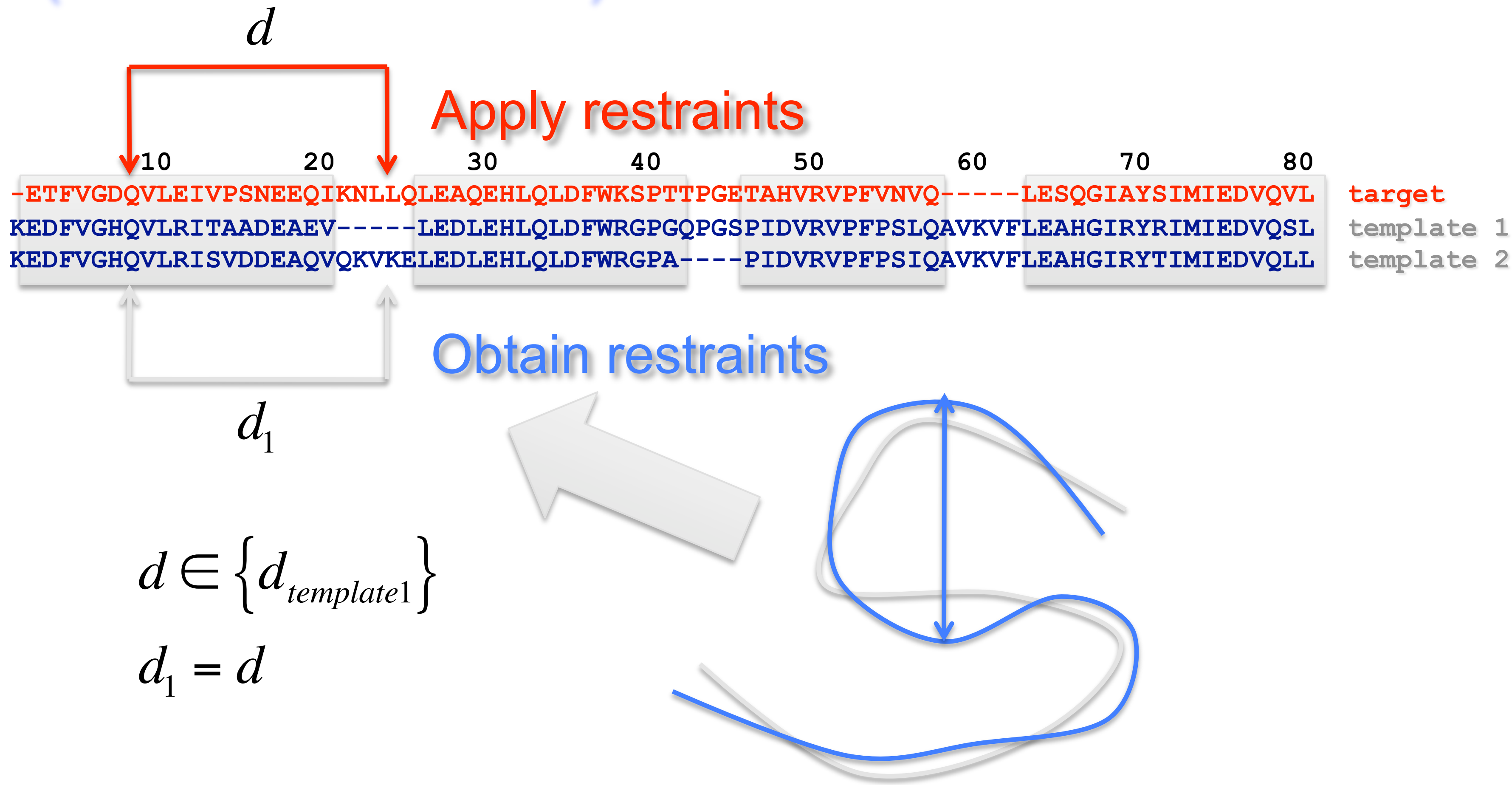
$$d \in \left\{ d_{template1}; d_{template2} \right\}$$

$$d_1 < d < d_2$$

Long distance restraints

2. Schema of the method

3. Model building: Spatial restraints (Distance restraints)



```
        10        20        30        40        50        60        70        80
-ETFVGDQVLEIVPSNEEQIKNLLQLEAQEHLQLDFWKSPTTPGETAHVRVPFVNVQ-----LESQGIAYSIMIEDVQVL  target
KEDFVGHQVLRITAADEAEV-----LEDLEHLQLDFWRGPGQPGSPIDVRVPFPSLQAVKVFLEAHGIRYRIMIEDVQSL  template 1
KEDFVGHQVLRISVDDEAQVQKVKELEDLEHLQLDFWRGPA----PIDVRVPFPSIQAVKVFLEAHGIRYTIMIEDVQLL  template 2
```

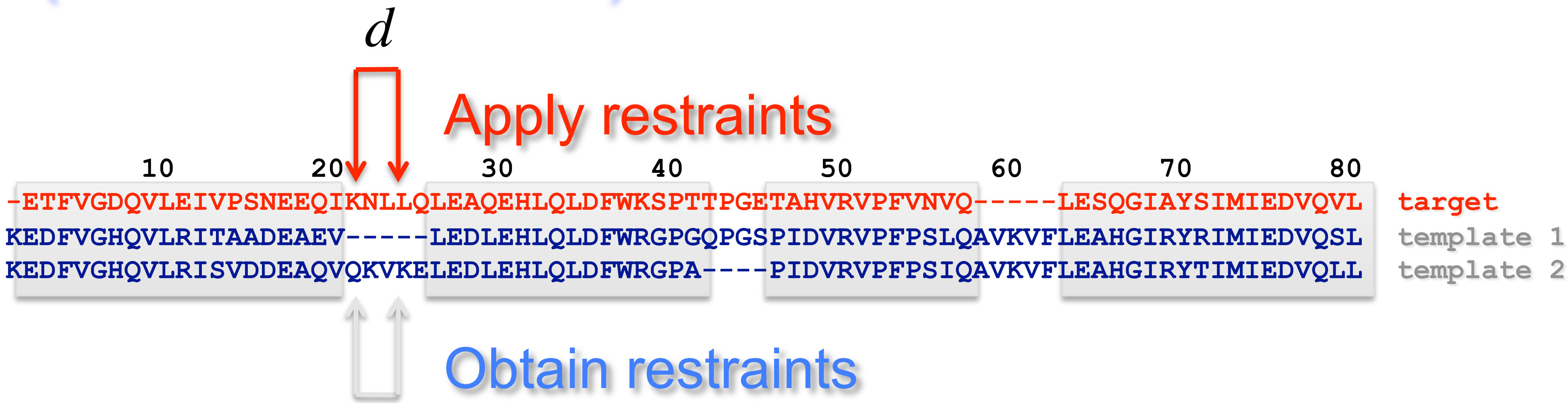
$$d \in \{d_{template1}\}$$

$$d_1 = d$$

Distance restraints between Aa in SCR & VR
(required to locate the conformation of the VR)

2. Schema of the method

3. Model building: Spatial restraints (Distance restraints)



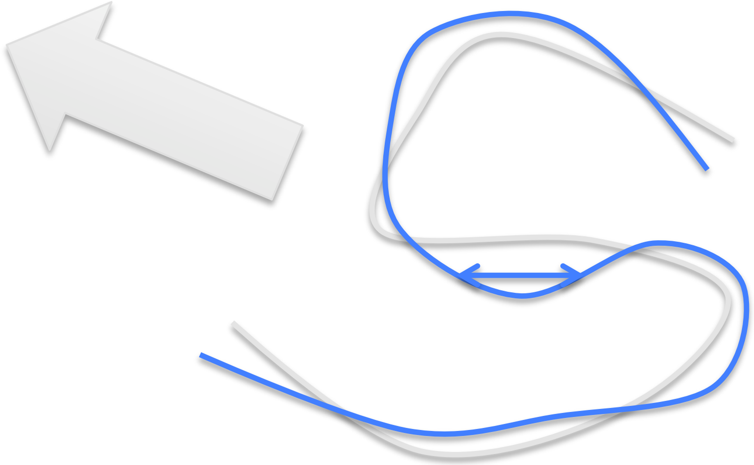
d

Apply restraints

```
         10        20        30        40        50        60        70        80
-ETFVGDQVLEIVPSNEEQIKNLLQLEAQEHLQLDFWKSPTTPGETAHVRVPFVNVQ-----LESQGIAYSIMIEDVQVL  target
KEDFVGHQVLRITAADEAEV-----LEDLEHLQLDFWRGPGQPGSPIDVRVPFPSLQAVKVFLEAHGIRYRIMIEDVQSL  template 1
KEDFVGHQVLRISVDDEAQVQKVKELEDLEHLQLDFWRGPA----PIDVRVPFPSIQAVKVFLEAHGIRYTIMIEDVQLL  template 2
```

Obtain restraints

d_1



$$d \in \left\{ d_{template1} \right\}$$

$$d_1 = d$$

Distance restraints between Aa in VR & VR
(required to obtain the conformation of the VR)

2. Schema of the method

3. Model building: Spatial restraints (Simulated annealing)

Optimizing a target function:

1. Start with e.g. a random conformation model and use only local restraints
2. Minimize some steps using a conjugate gradient optimization and molecular dynamics steps
3. Repeat, introducing more and more long range restraints until all restraints are used

$$E_{bonding} = \sum_{bonds} \frac{1}{2} k_i \left(d_i - d_i^0\right)^2 + \sum_{angles} \frac{1}{2} k_j \left(\alpha_j - \alpha_j^0\right)^2 + \sum_{\substack{improper \\ dihedral}} \frac{1}{2} k_n \left(\omega_n - \omega_n^0\right)^2 + \sum_{angles} E_m Cos\left(\omega_m \phi_m + \varphi_m\right)$$

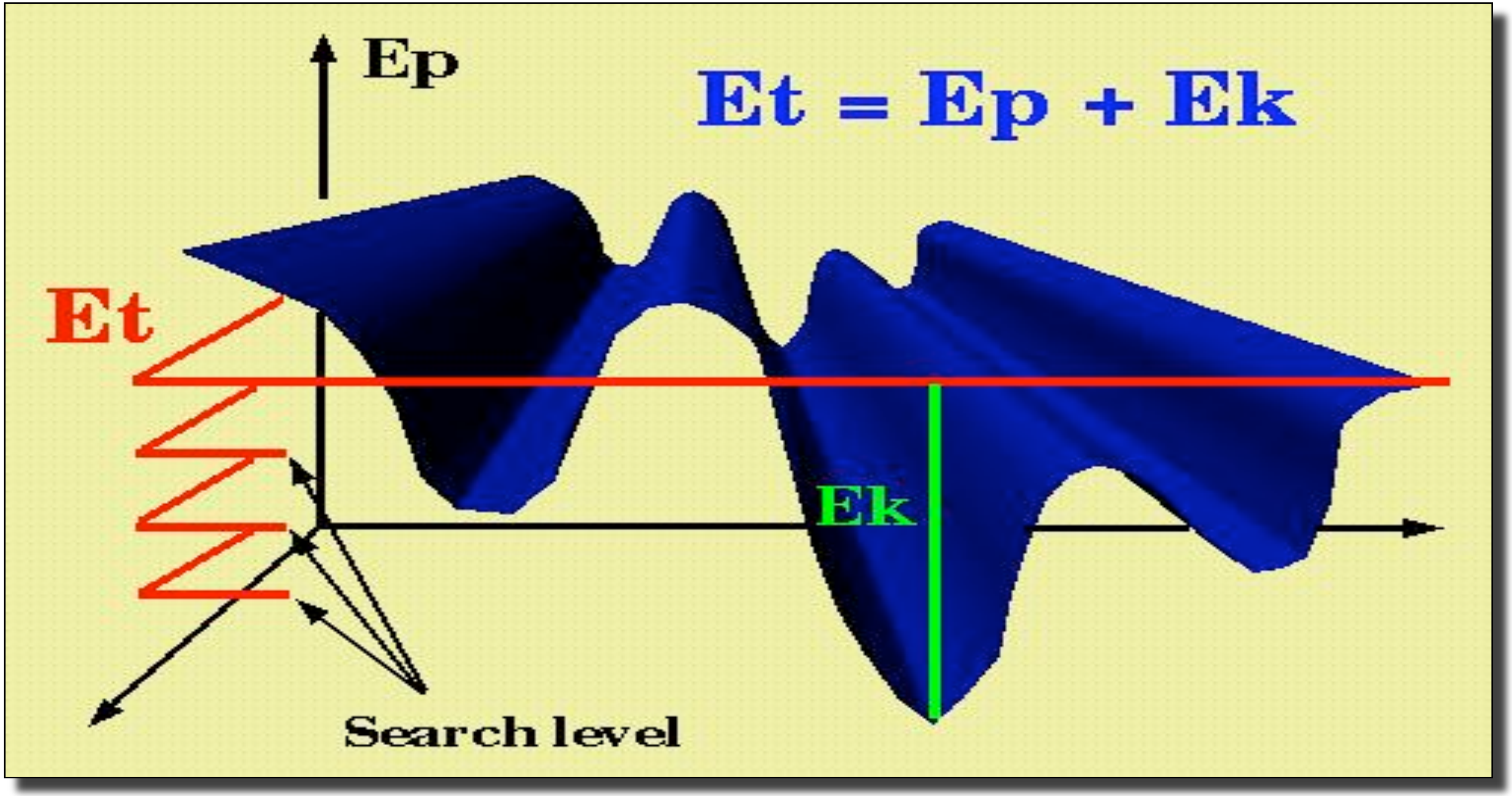
$$E_{non-bonding} = \frac{1}{4\pi\varepsilon_0} \sum_i \sum_{j>i} \frac{q_i q_j}{r_{ij}} + \sum_i \sum_{j>i} \frac{C_6^{ij}}{r_{ij}^6} - \frac{C_{12}^{ij}}{r_{ij}^{12}}$$

$$E_{dist} = \sum_{rest} \frac{1}{2} k_r \left(d_r - \left\langle d_r^0 \right\rangle\right)^2$$

$$E = E_{bonding} + E_{non-bonding} + E_{pdf} + E_{dist}$$

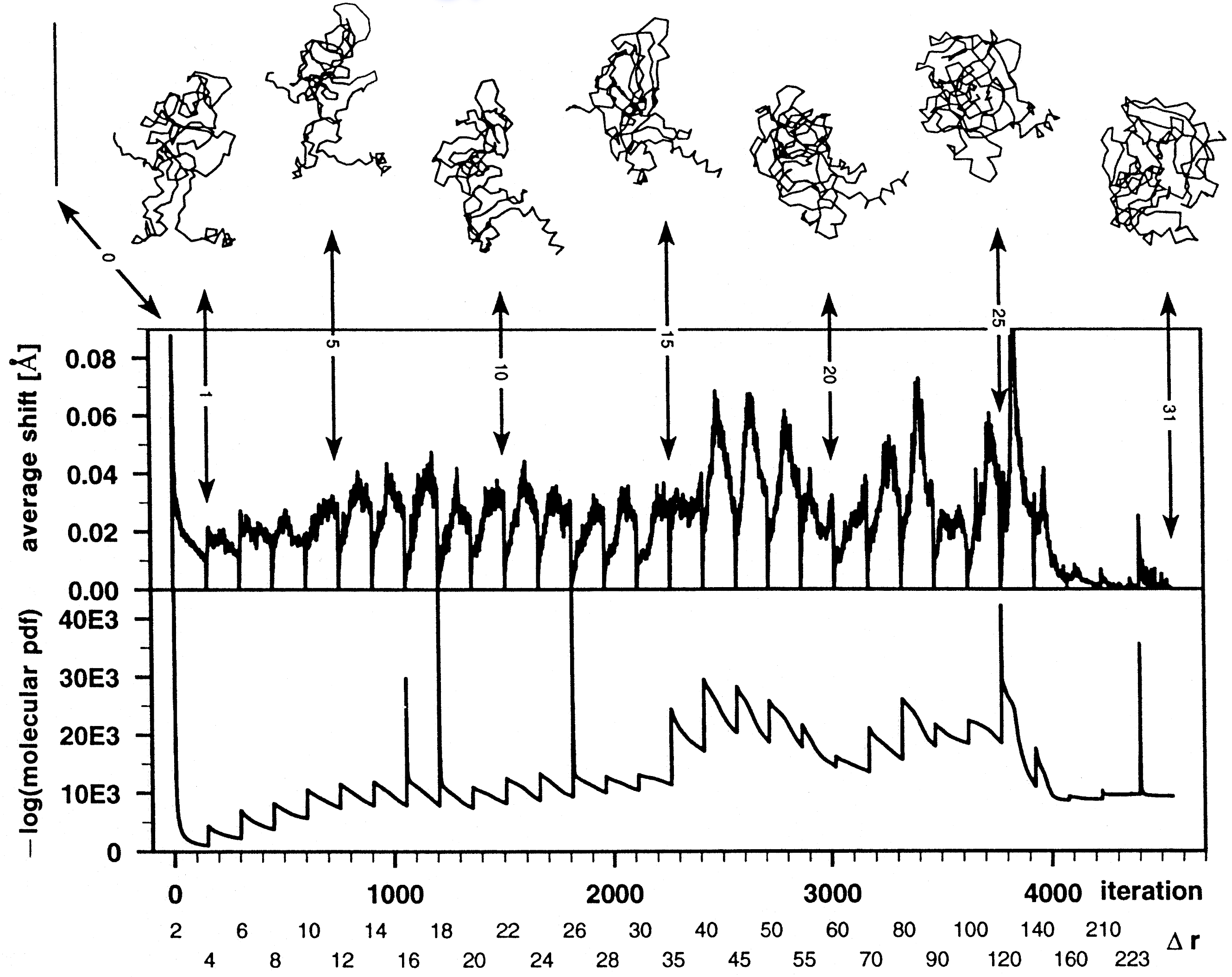
2. Schema of the method

3. Model building: Spatial restraints (Simulated annealing)



2. Schema of the method

3. Model building: Spatial restraints (Simulated annealing)



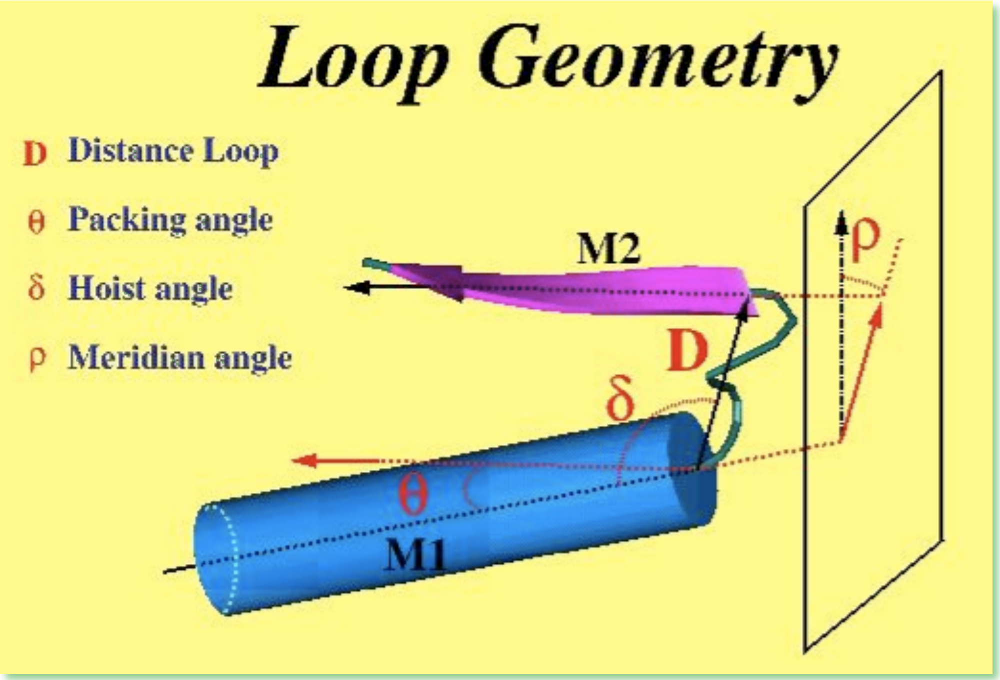
2. Schema of the method

3. Model building: Spatial restraints (Loop modeling using a database of loops)

Apply restraints

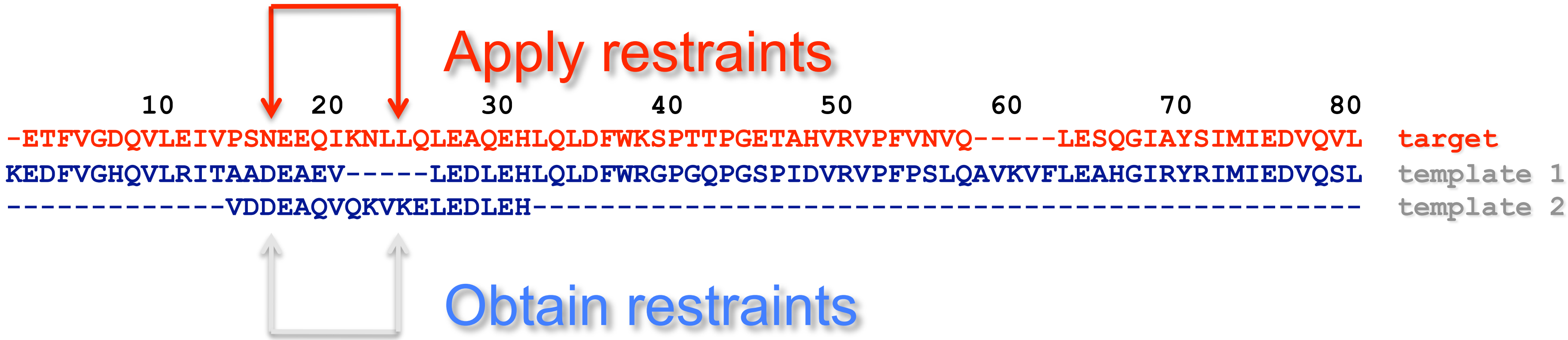
```
        10        20        30        40        50        60        70        80
-ETFVGDQVLEIVPSNEEQIKNLLQLEAQEHLQLDFWKSPTTPGETAHVRVPFVNVQ-----LESQGIAYSIMIEDVQVL  target
KEDFVGHQVLRITAADEAEV-----LEDLEHLQLDFWRGPGQPGSPIDVRVPFPSLQAVKVFLEAHGIRYRIMIEDVQSL  template 1
-------------VDDEAQVQKVKELEDLEH-------------------------------------------------  template 2
```

Obtain restraints



2. Schema of the method

3. Model building: Spatial restraints (Loop modeling using a database of loops)



Apply restraints

```
         10        20        30        40        50        60        70        80
-ETFVGDQVLEIVPSNEEQIKNLLQLEAQEHLQLDFWKSPTTPGETAHVRVPFVNVQ-----LESQGIAYSIMIEDVQVL  target
KEDFVGHQVLRITAADEAEV-----LEDLEHLQLDFWRGPGQPGSPIDVRVPFPSLQAVKVFLEAHGIRYRIMIEDVQSL  template 1
-------------VDDEAQVQKVKELEDLEH-------------------------------------------------  template 2
```

Obtain restraints

Using the structure of a known loop:

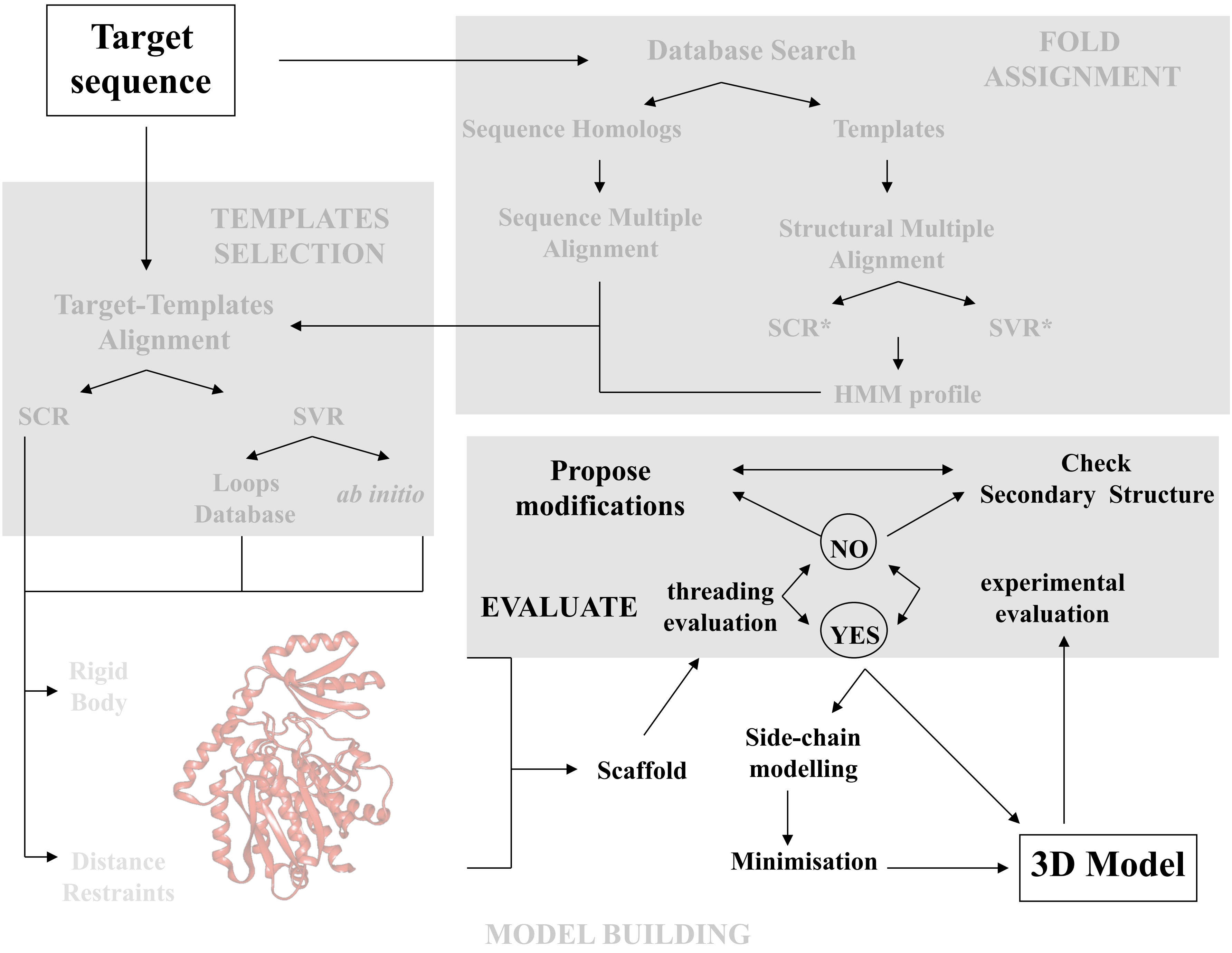
1. The C-tail and N-tail of the loop (template 2) when superposed with the core of the main template (template 1) produce a low RMSD
2. The selection of the loop follow two criteria: similar sequence profile with the target and similar anchoring geometry of the loop with the main template

2. Schema of the method

3. Model building: Spatial restraints (Loop modeling *ab initio*)

Using PDF of loops and minimization methods:

1. Calculate specific PDF residue properties of loops
2. Minimize by simulated annealing the loops
3. Extract main motion from normal modes on templates and apply them as restrictions on the conformational changes of the model
4. Methods:
    1. Loop-model from MODELLER
    2. ArchPred
    3. Rosetta


Target sequence
Database Search
FOLD ASSIGNMENT
Sequence Homologs
Templates
Sequence Multiple Alignment
Structural Multiple Alignment
SCR*
SVR*
HMM profile
TEMPLATES SELECTION
Target-Templates Alignment
SCR
SVR
Loops Database
ab initio
Rigid Body
Distance Restraints
Propose modifications
Check Secondary Structure
NO
YES
EVALUATE
threading evaluation
experimental evaluation
Scaffold
Side-chain modelling
Minimisation
3D Model
MODEL BUILDING

2. Schema of the method

4. Evaluation

Types of Errors

1. *Errors in side-chain packing .*

2. *Shifts of correctly aligned residues .*

3. *Regions without template .*

4. *Errors due to misalignments .*

5. *Errors produced by incorrect templates .*

2. Schema of the method

4. Evaluation

Shifts of correctly aligned residues

```
HHHHHHHH HHH .HHC
GARFIELD THE .CAT
GARFIELD THE CCAT
```

Solution

```
HHHHHHHH HHH HHC.
GARFIELD THE CAT.
GARFIELD THE CCAT
```

2. Schema of the method

4. Evaluation

Errors due to misalignments .

```
GARFIELD THE CAT ...
GARFIELD THE FAT CAT
```

Solution

```
GARFIELD THE ... CAT
GARFIELD THE FAT CAT
```

2. Schema of the method

4. Evaluation

How to test the model?

1. Compare the RMSD between the model and the real structure
2. Check that secondary structures are correctly aligned
3. Calculate the percentage of residues that are closer than a threshold after superposing the model and the real structure
4. Calculate the percentage of identical residues aligned when superposing the real structure and the model.
5. Check the energy of threading to compare the real structure and the model (see next chapter)

2. Schema of the method

4. Evaluation

Model Accuracy Evaluation

CASP

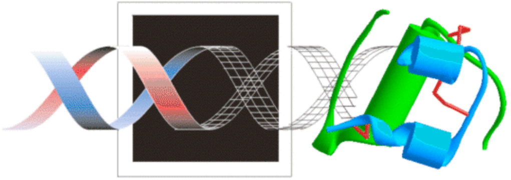
Community Wide Experiment on the Critical Assessment of Techniques for Protein Structure Prediction

http://PredictionCenter.llnl.gov/casp5/

EVA

Evaluation of Automatic protein structure prediction

[Burkhard Rost, Andrej Sali, http://maple.bioc.columbia.edu/eva/]



3D - Crunch

Very Large Scale Protein Modeling Project

http://www.expasy.org/swissmod/SM_LikelyPrecision.html

2. Schema of the method

5. Improvement

How to detect possible errors in the model if we don’t know the solution?

1. Compare the model and all the templates
2. Check that secondary structures are not broken
3. Check if the prediction of secondary structure agrees with the secondary structure of the model
4. Check if the loops of the target are similar to some loops in the database of loops and they agree in sequence and anchoring geometry
5. Check the capping of helices
6. Check pseudo-energies of threading and compare the model with the templates.

2. Schema of the method

5. Improvement

How to improve the model?

1. Decide the changes in the alignment according to the secondary structure prediction or the structure of the templates and recalculate the model
2. Change the main template and recalculate the model
3. Include new templates
4. Calculate the main motion of normal modes from the templates of the homologous family and optimize by molecular dynamics under motion restrictions the conformation
5. Recalculate the pseudo energy profile of the new model and compare with the original model to test the improvement

Fold Prediction

Fold prediction

1. Fold recognition (threading)
2. *ab initio* fold prediction
3. Protein folding (MD with explicit solvent)

Threading

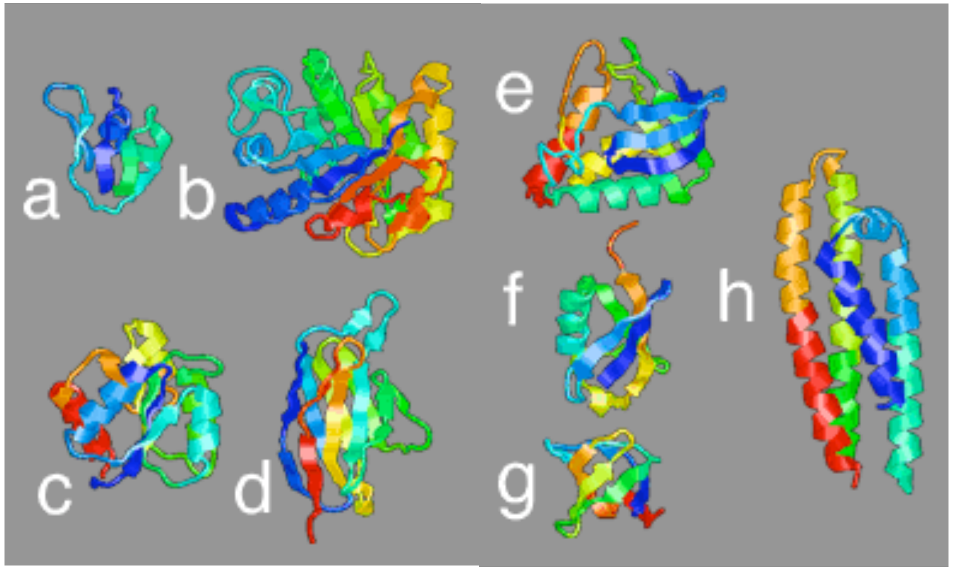
Idea: Find the optimal structure for a new (target) sequence in the set of known 3D-structures (templates) by threading the target sequence.

Fold recognition / Threading

Principle: Find a compatible fold for a given sequence ....

```
>Protein XY
MSTLYEKLGGTTAVDL
AVDKFYERVLQDDRIK
HFFADVDMAKQRAHQ
KAFLTYAFGGTDKYDG
RYMREAHKELVENHGL
NGEHFDAVAEDLLATLK
EMGVPEDLIAEVAAVAG
APAHKRDVLNQ
```

? ≈



Using ...

- 1D – 3D profile matching,
- mean force potentials,
- secondary structure predictions,
- position specific scoring matrices (PSSM),
- keyword statistics,
- ....

1. Fold recognition (threading)
    1. Knowledge-base potentials
        1. Distance dependent potentials
            - Atom-centered
            - Sequence distance
            - Reference state
        2. Solvation
        3. Z-scores and energy profiles
        4. Methods: Prosa, Anolea, DOPE,S^2PServer
    2. Distance homology matrices (PSSM)
        1. Function association
        2. Methods: FUGUE, PHYRE, ModLink
    3. Secondary structure alignment
        1. Secondary structure prediction
            - Machine learning theory
            - Neural Networks
        2. Methods: TOPITS

1. Knowledge-base potentials

    1. Distance dependent potentials

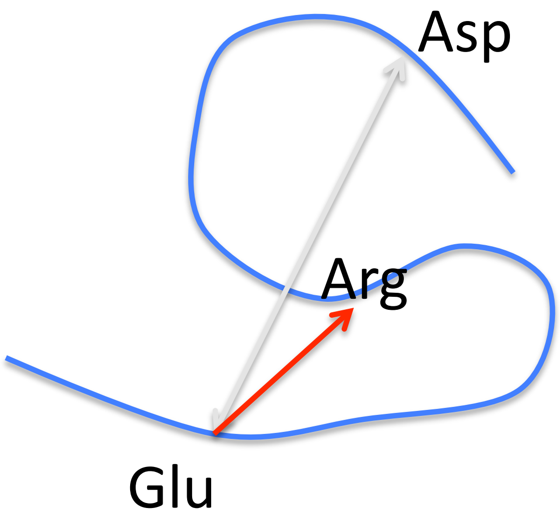
According to Boltzmann law

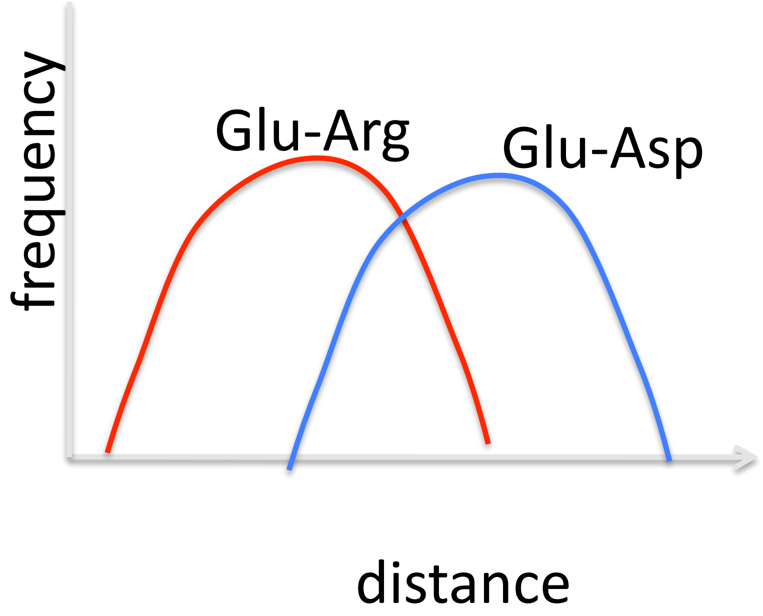
$$P(x) = \frac{1}{Z} e^{-E(x)/k_B T}$$

Therefore, energy is related with probability

$$P(Asp, Asp, d = 10A) \Rightarrow E(Asp, Asp, d = 10A)$$

1. Knowledge-base potentials
    1. Distance dependent potentials

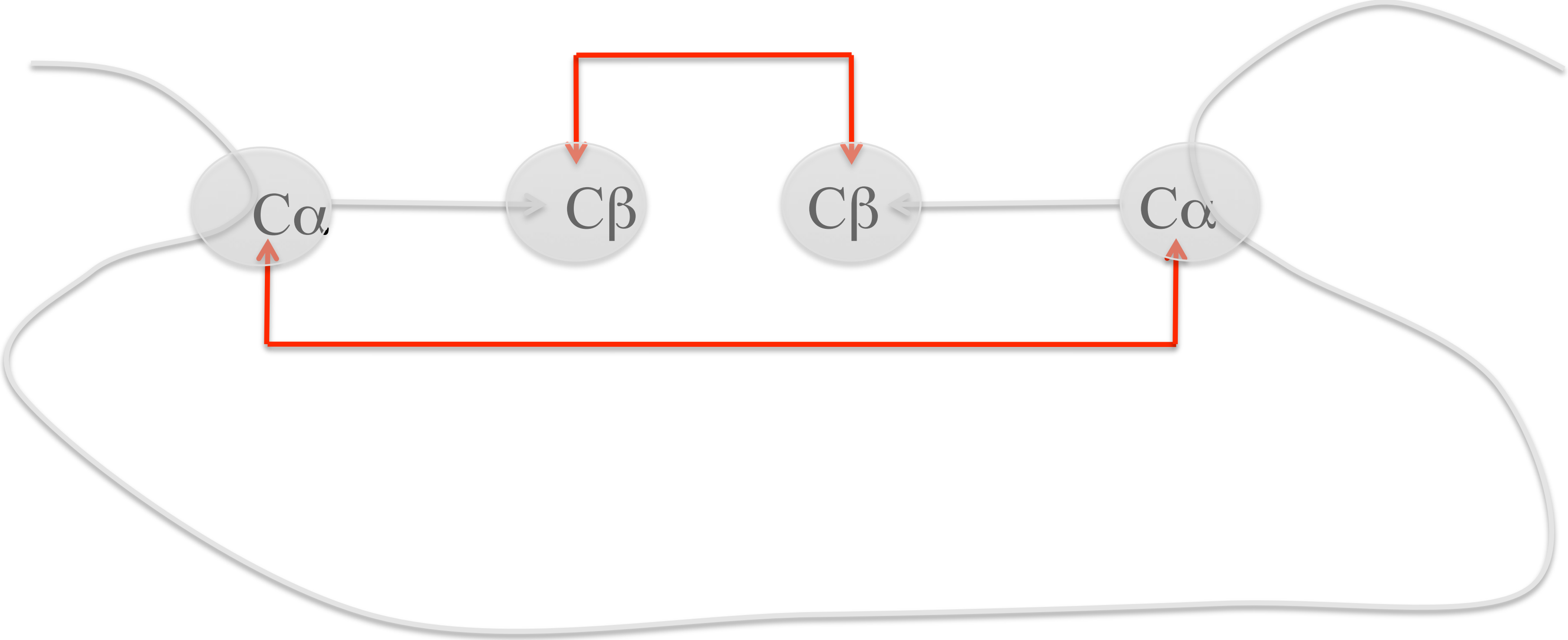




1. Knowledge-base potentials

1. Distance dependent potentials

1. Distances are calculated between atoms: We have to select what atom are we going to use
   •The best choice is Cβ because it indicates the direction of the side-chain



1. Knowledge-base potentials

1. Distance dependent potentials

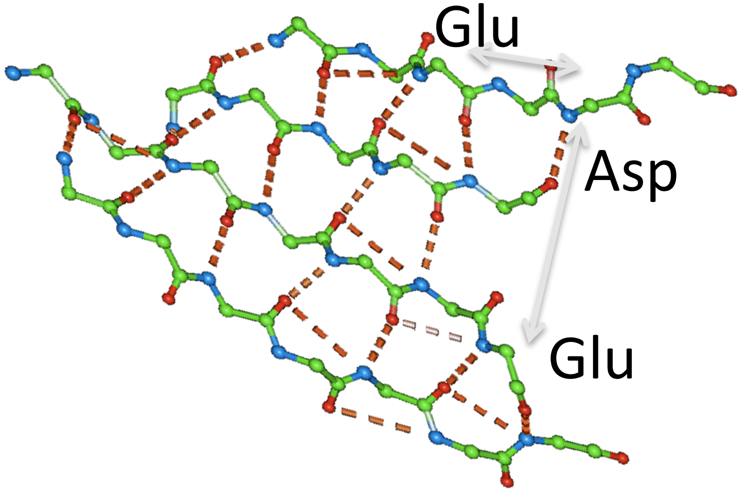
2. The database of structures to extract distances has to avoid redundant structures (between homologs and members of the same family/superfamily)
    - If we use all the structures of the same or similar protein there will be a bias. Thus, we use a set with less than 40% of sequence similarities

1. Knowledge-base potentials

1. Distance dependent potentials

3. The frequency of a pair of residues at distance “r” is different if the residues are close or distant along the sequence

- We split the calculation of frequencies depending on the sequence distance between residues

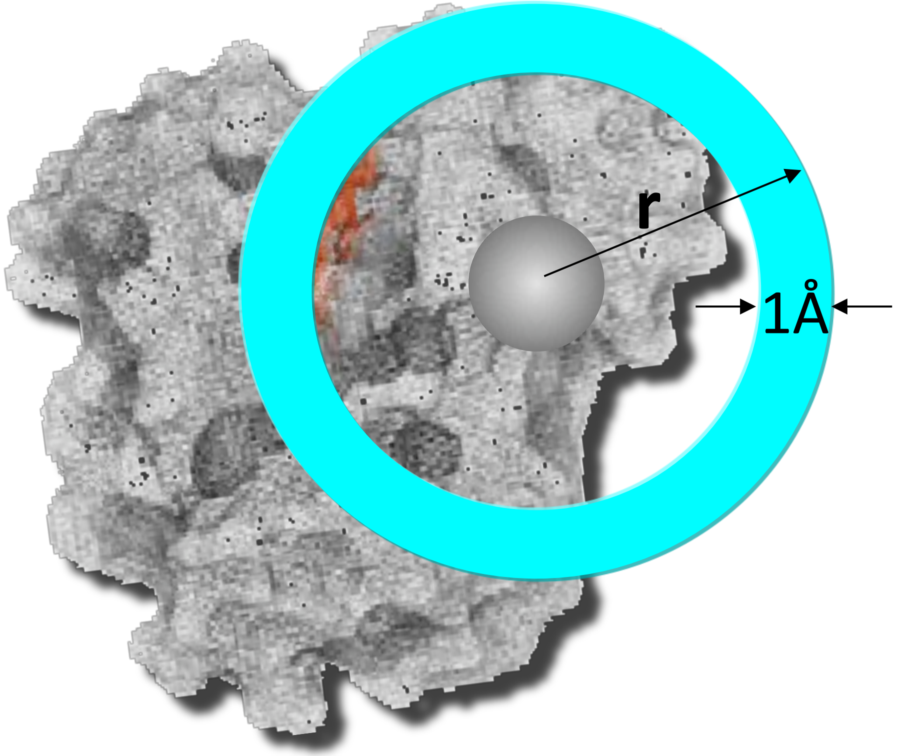


1. Knowledge-base potentials

1. Distance dependent potentials

4. Reference state: The density of residues around one residue is not a continuous model, it depends on the size and shape of the protein.

- We need to normalize by the density ($4\pi r^2 \varepsilon(r)$) and thus defining a reference state



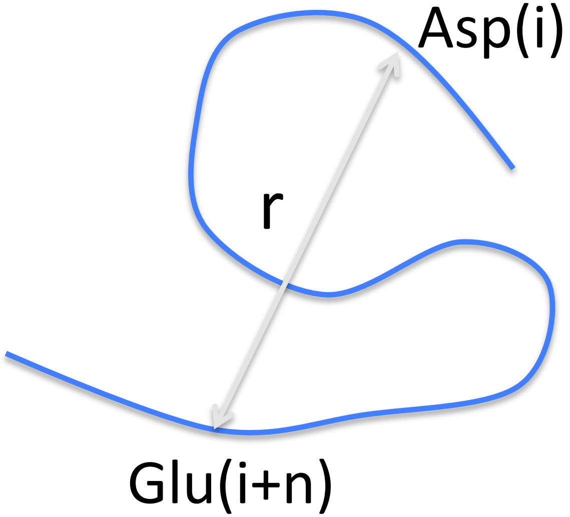
1. Knowledge-base potentials

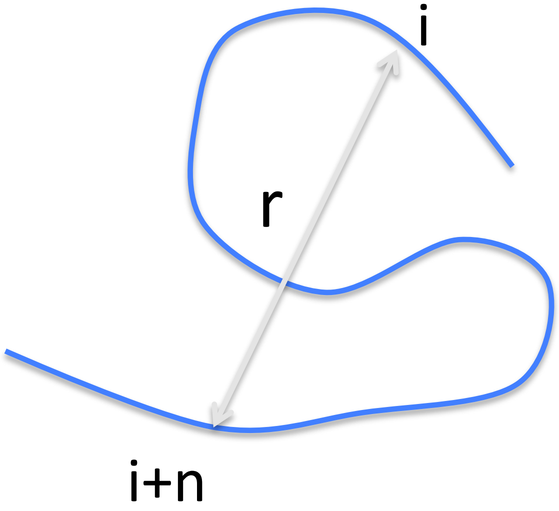
1. Distance dependent potentials

4. Reference state: the simplest definition of the reference state is to use the whole data set of residue pairs, thus instead of using energies we use incremental energies.
   - Let be a pair of residues Asp and Glu at distance n in sequence. Let be N(r/ED,n) the number of pairs ED like this at distance r between their Cβ atoms, and N(r/n) the total of pairs of residues at distance n in sequence and r between their Cβ atoms

1. Knowledge-base potentials

1. Distance dependent potentials

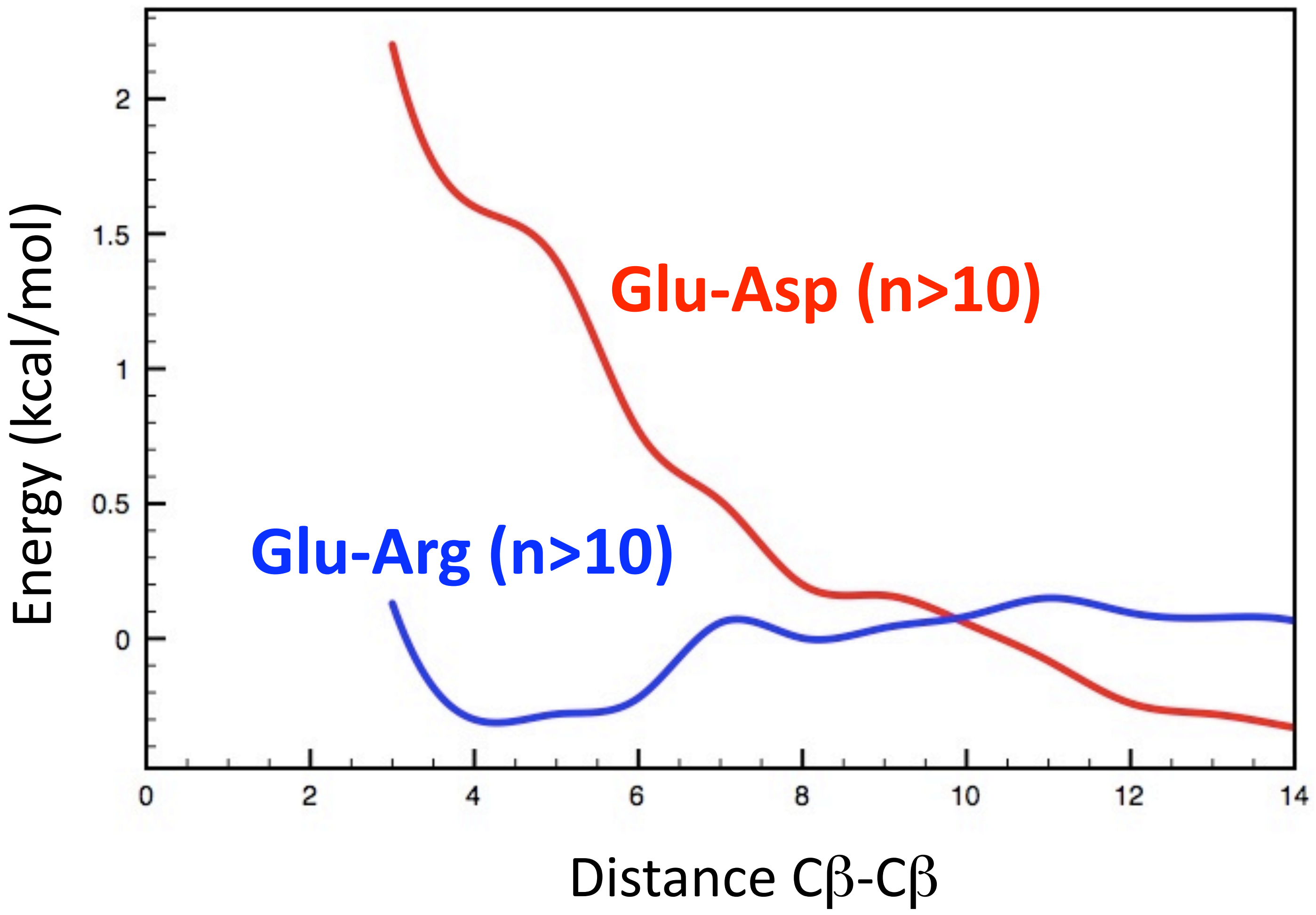




$$\Delta E(r/(Glu,Asp,C\beta,C\beta,n)) = -kT\ln\left(\frac{N(r/ED,n)}{N(r/n)}\right)$$

1. Knowledge-base potentials
    1. Distance dependent potentials

Example of distance dependent knowledge-based potentials



1. Knowledge-base potentials

2. Solvation

1. Solvation of a residue is calculated as proportional to accessible surface area (ASA)
   - The factor of proportion depends on the tendency of the residue (i.e. Asp in position “i” of the sequence) to be solvated (hydrophobicity calculated with water-octanol partition coefficient)

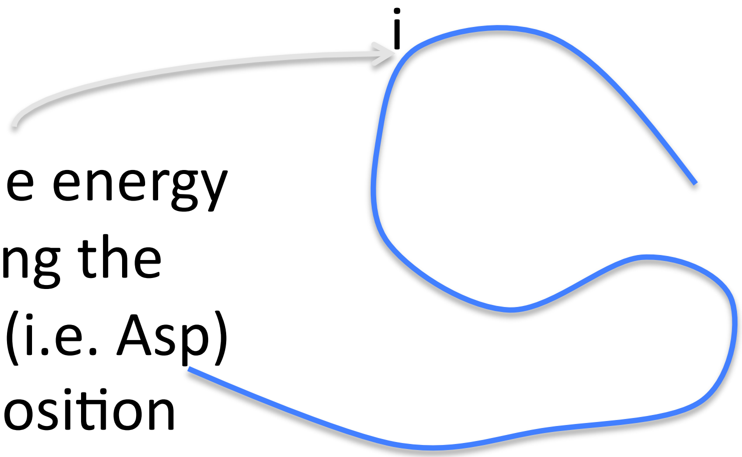
$$E_{sol}(i) = \sigma_{Asp} ASA(i)$$

2. Solvation can also be calculated using the frequency of the residue to be exposed on the surface

1. Knowledge-base potentials

3. Z-scores and energy profiles

Once we have a set of energies for pairs of residues (force field) we can calculate the energy of each residue along the sequence in a specific conformation



E(i) is the energy on placing the residue (i.e. Asp) in this position (i)

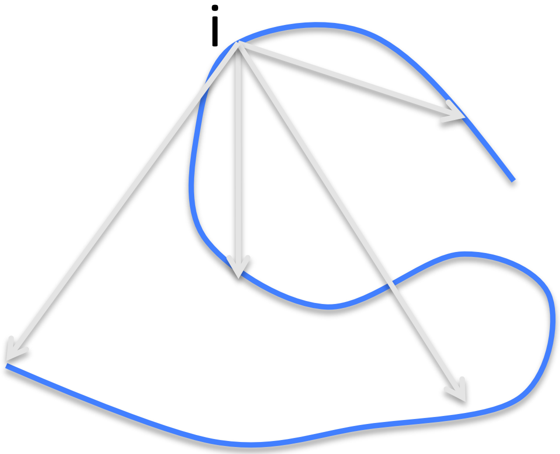
1. Knowledge-base potentials

3. Z-scores and energy profiles

$$E_i = \sum_{j \neq i} E_{ij}\left(r, n = \left|j - i\right|\right)$$

$$E_{ij}\left(r, n = \left|j - i\right|\right) = \Delta E(r/(Glu(j), Asp(i), C\beta, C\beta, n))$$

$$E_{sol}(i) = \sigma_{Asp} ASA(i)$$



Note: we have assumed that in position I we have placed Asp and Glu in position j=i+n

1. Knowledge-base potentials

3. Z-scores and energy profiles

The total energy of a protein is obtained by the sum of the pair-energies and the energy from its surface (solvation)

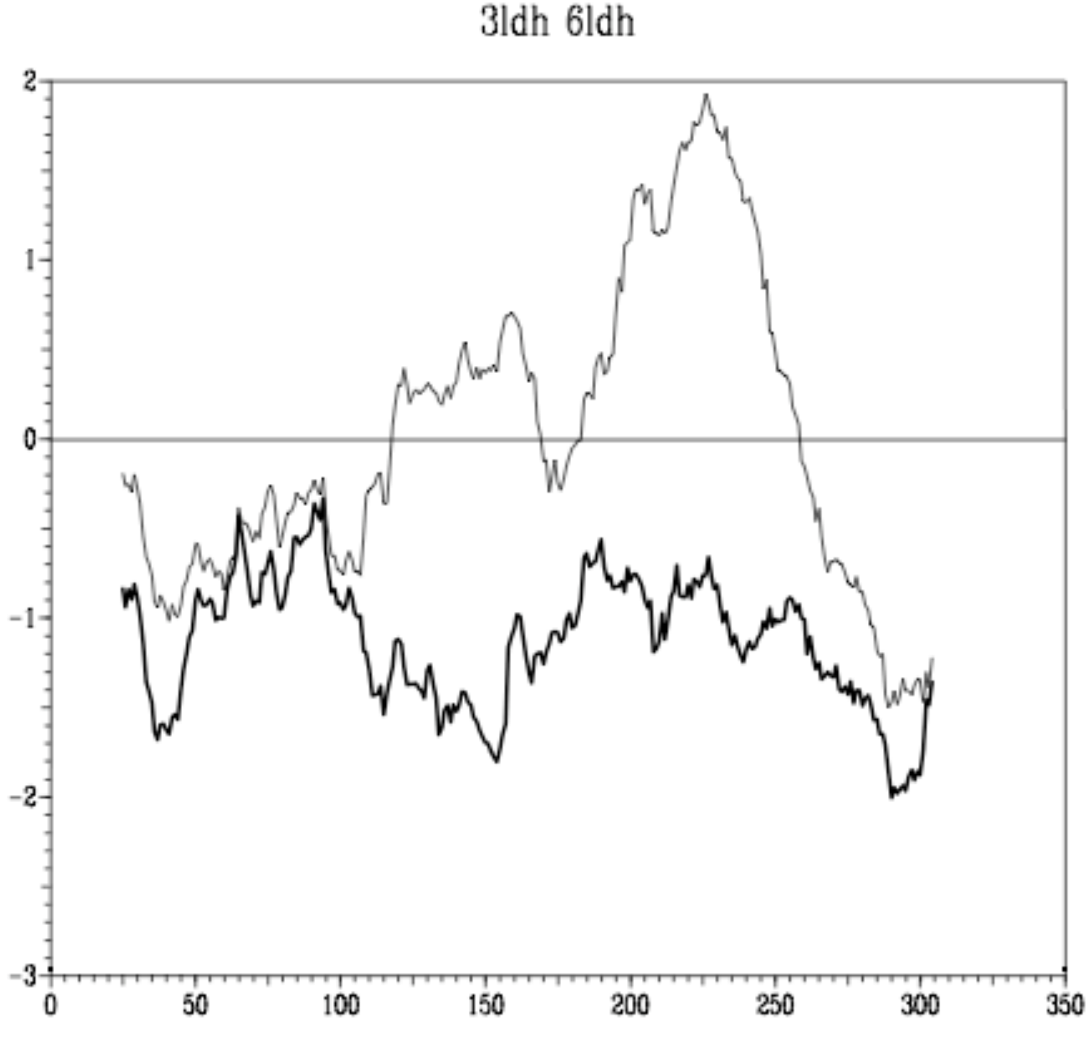
$$E = \sum_i E_i + \beta \sum_i E_{sol}(i)$$

The profile energy is obtained by the curves of the pair-energies, surface energy and combined energy of both with respect to the residue position

1. Knowledge-base potentials

3. Z-scores and energy profiles

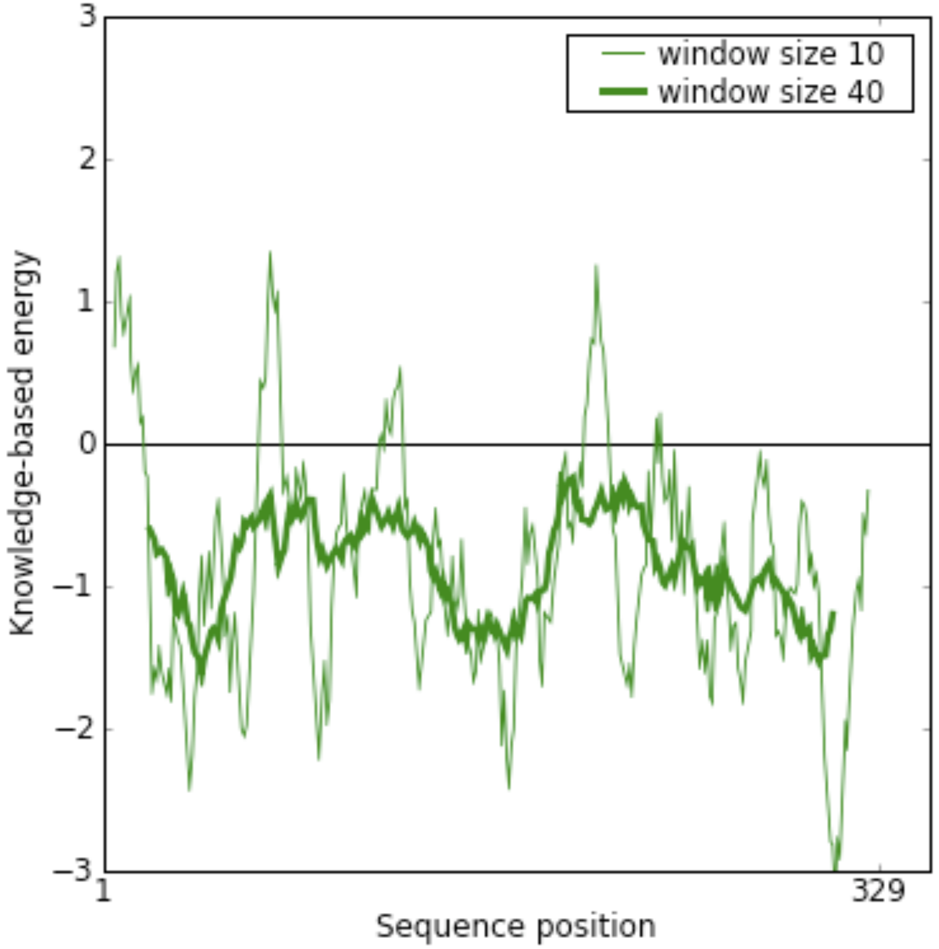
Example of profile energy from PROSA



1. Knowledge-base potentials

3. Z-scores and energy profiles

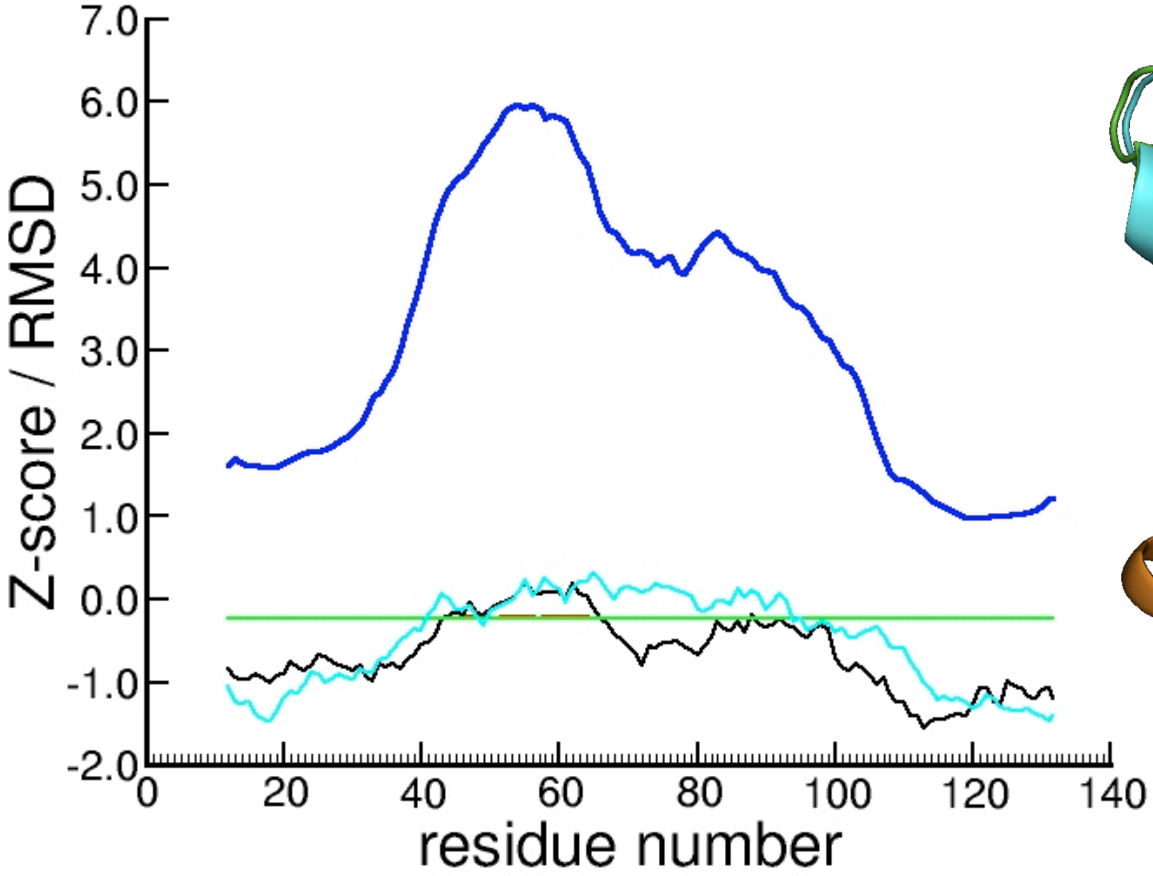
Often the curve is smoothed by windowing the curve: the value on each point is defined by the average of a window of W residues and the window moves along the X axis.

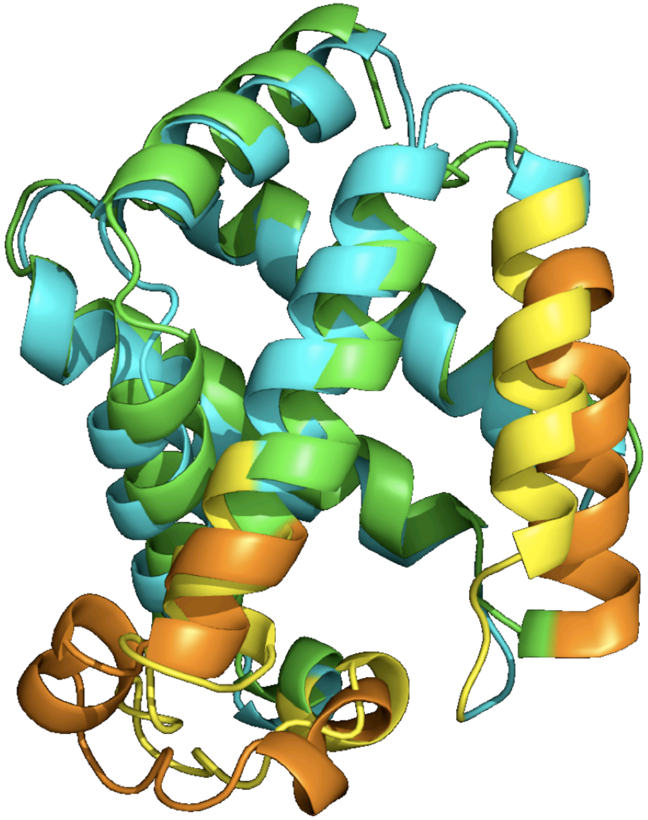


1. Knowledge-base potentials

3. Z-scores and energy profiles

Energy profiles can be used to detect errors in modeling



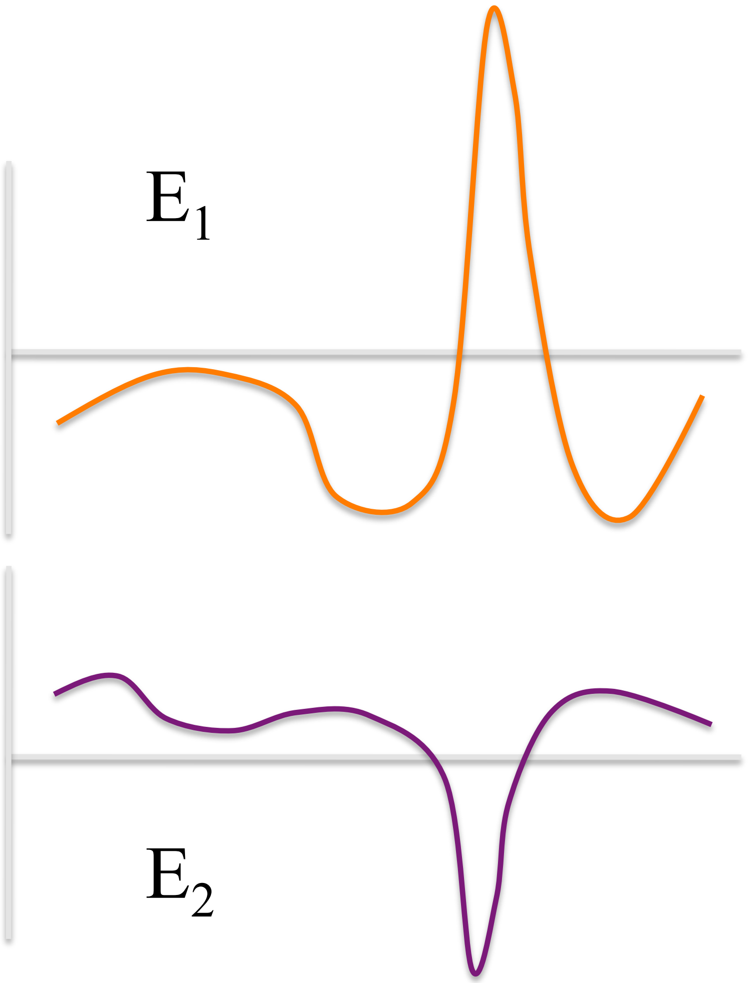


1. Knowledge-base potentials

3. Z-scores and energy profiles

Question:

Can we use the total energy to discriminate correct folds among wrong conformations (decoys)?



$$E_1 > E_2$$

Wrong solution

1. Knowledge-base potentials

3. Z-scores and energy profiles

Question:

Can we use the total energy to discriminate correct folds among wrong conformations (decoys)?

$$0 > E_1 > E_2 > E_3 \cdots > E_n$$

Many solutions is a wrong solution

Solution:

Define a new function statistically meaningful, the Z-score

1. Knowledge-base potentials

3. Z-scores and energy profiles

Threading Z-score is defined by comparing the energy on one fold (j) with the average of the real folds from the database (i.e. transforms the function "energy" into a Gaussian distribution centered at zero)

$$Zscore_j = \frac{E_j - \langle E \rangle}{\sigma}$$

$$\langle E \rangle = \frac{\sum_{i=1}^{N_{folds}} E_i^{real}}{N_{folds}}$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N_{folds}} \left(E_i - \langle E \rangle \right)^2}{N_{folds} - 1}}$$

This is the same problem as the following:

Consider the final marks in the class after the exam. We can calculate the 10 best alumni according to their marks.

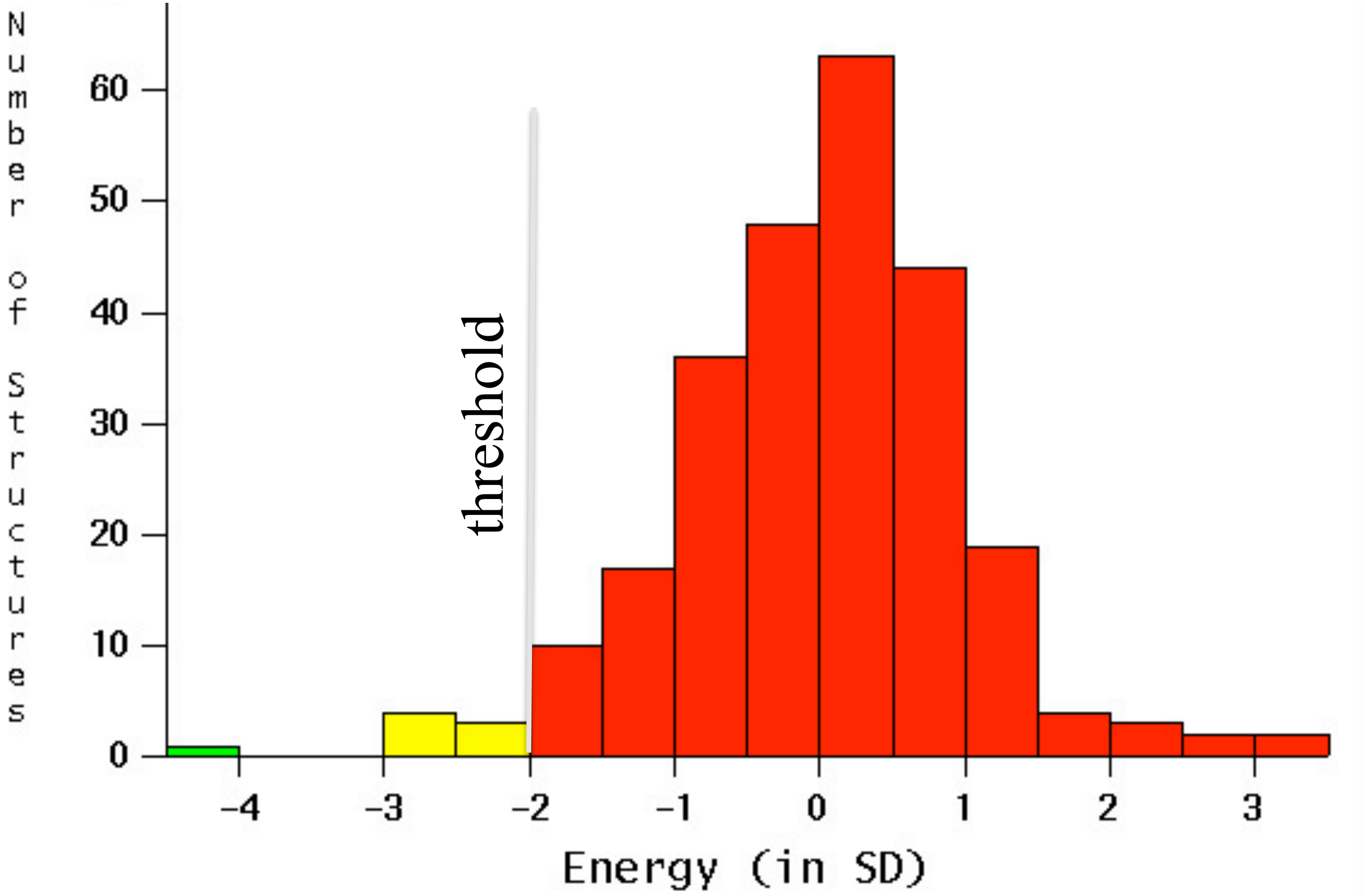
Are these the best alumni of SBI in the world?

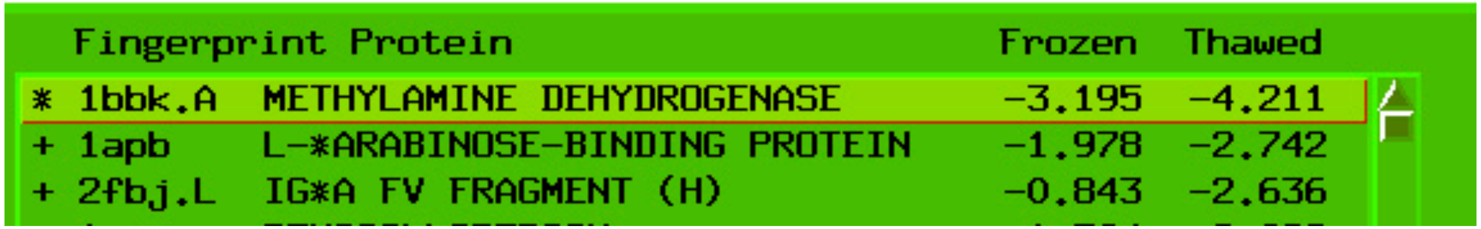
We have to weight their marks with the best students of the world, assuming the exam was the same.

To do that, we use the set of marks of the total of SBI teachers in the world, and we assume they are the best set

1. Knowledge-base potentials

3. Z-scores and energy profiles



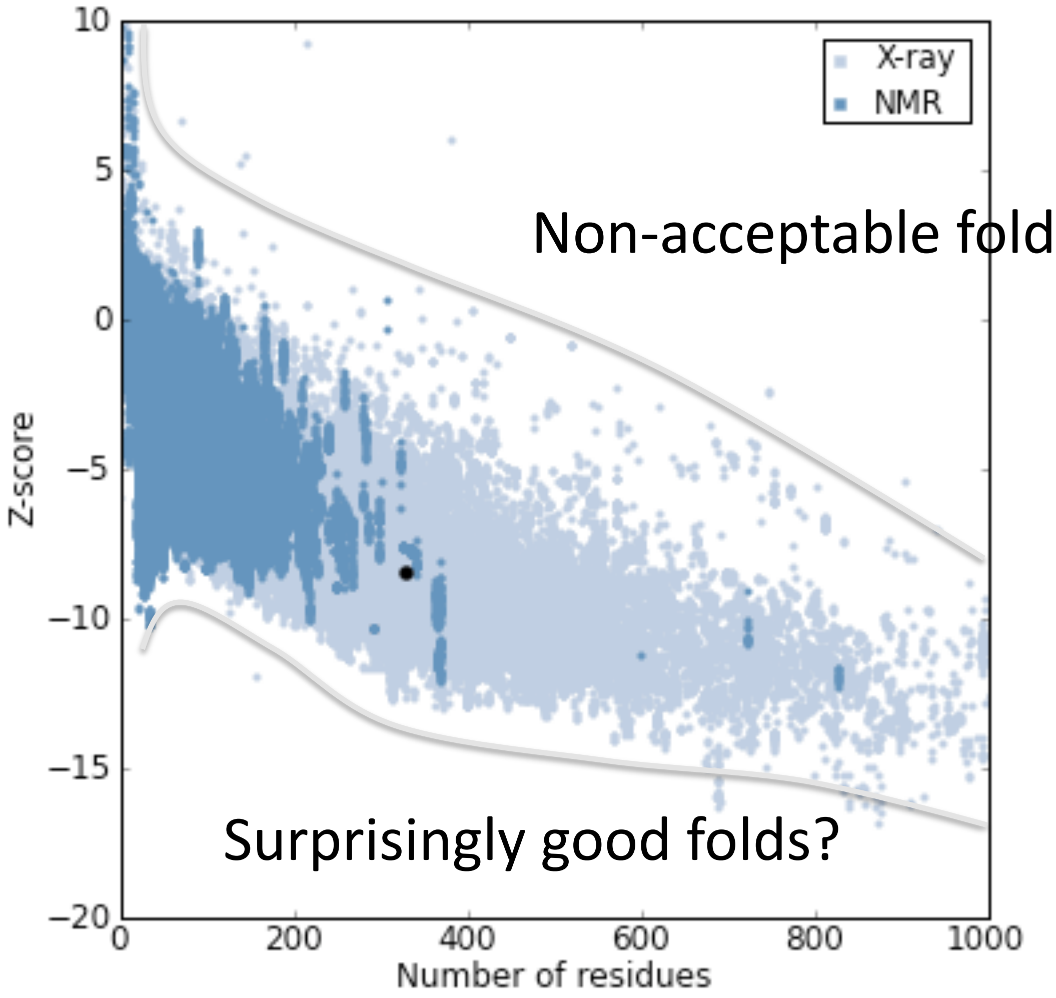


	Fingerprint	Protein	Frozen	Thawed
*	1bbk.A	METHYLAMINE DEHYDROGENASE	-3.195	-4.211
+	1apb	L-*ARABINOSE-BINDING PROTEIN	-1.978	-2.742
+	2fbj.L	IG*A FV FRAGMENT (H)	-0.843	-2.636

1. Knowledge-base potentials

3. Z-scores and energy profiles

Zscores can also be presented as a function of the length of the protein sequence



Non-acceptable folds

Surprisingly good folds?

2. Remote homologs (PSSM)

We can use sequence alignments with position specific substitution matrices (PSSM) (see theory in practices)

1. Alignment between one sequence and a Hidden Markov Model profile (hmmpfam, hmmscan)
2. Alignment between two Hidden Markov Model profiles (HHSearch, HBlitz, PRC)
3. Alignment between sequences using PSSMs (BLAST, fugue)

2. Remote homologs (PSSM)

1. Function association

PHYRE / 3D-PSSM

Remotely homologous structures that can't be found by conventional methods are detected by using profiles (or PSSMs) generated by PSI-Blast for both target sequence and the sequences of the known structures. Phyre performs a profile-profile matching algorithm together with predicted secondary structure matching.

The functional keywords are found by gathering homologues of the target sequence from Swissprot, taking the keywords associated with the Swissprot homologues and weighting them according to their background frequency across the whole Swissprot database using SAWTED

1. Knowledge-base potentials

3. Z-scores and energy profiles

SAWTED

What is SAWTED?

SAWTED stands for **S**tructure **A**ssignment **W**ith **Te**xt **D**escription. It is a method to improve the coverage of the detection of remote homologues of known structure by sequence searches (e.g. PSI-BLAST) and fold recognition programs.

What does it do?

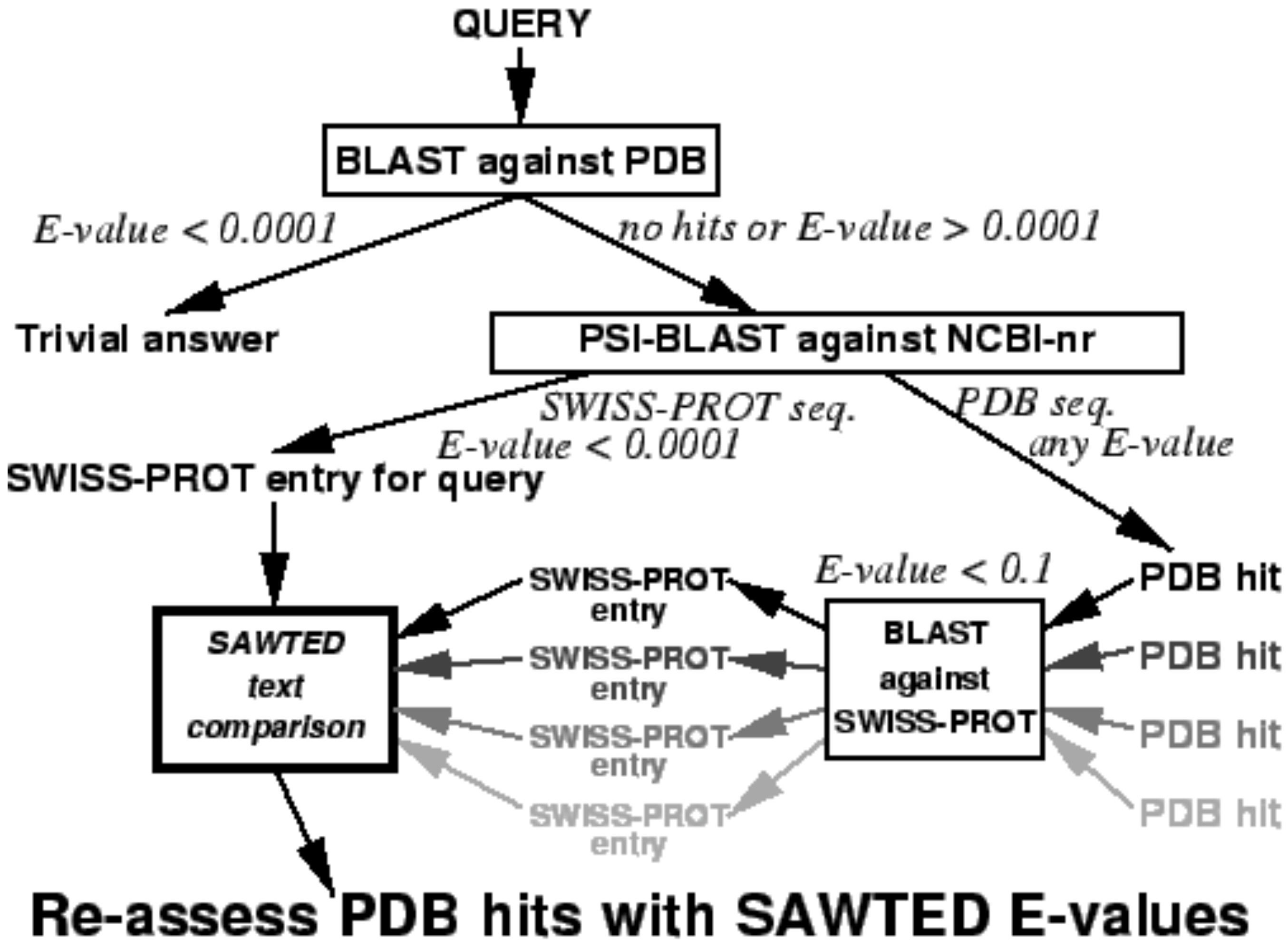
When sequence database searches return only hits with scores worse than an accepted threshold for reliability the user will often compare what is known about the function of the query sequence with that known about the poor scoring hits. Some hits may appear more sensible than others and deserve closer inspection. In SAWTED this comparison is made automatically using an algorithm to compare the text of SWISS-PROT annotations related to the query and to the poor scoring hits. A single E-value is given for the user to assess the similarity of function.

SAWTED is currently implemented to enhance PSI-BLAST searches against the PDB, and as part of our 3D-PSSM fold recognition server

1. Knowledge-base potentials

3. Z-scores and energy profiles

SAWTED in PHYRE & 3D-PSSM



3. Secondary structure alignment

1. secondary structure prediction (machine learning)

M = { set of data obtained with a predictive model}

D = { set of data known}

Bayes Theorem

$$P(D/M) = \frac{P\left(D \cap M\right)}{P(M)}$$

$$P(M/D) = \frac{P\left(D \cap M\right)}{P(D)}$$

$$P(M/D) = P(D/M)\frac{P(M)}{P(D)}$$

3. Secondary structure alignment

1. secondary structure prediction (machine learning)

M = { set of data obtained with a predictive model}

D = { set of data known}

Optimizing Function Φ (minimum Φ)

$$\Phi = -\log\left(P(M/D)\right)$$

$$\Phi = -\log\left(P(D/M)\right) - \log\left(P(M)\right) + \log\left(P(D)\right)$$

$$Min\left(\Phi\right) = Min\left(-\log\left(P(D/M)\right) - \log\left(P(M)\right)\right)$$

Maximum a priori

$$Min\left(\Phi\right) \approx Min\left(-\log\left(P(D/M)\right)\right)$$

Maximum likelihood

3. Secondary structure alignment

1. secondary structure prediction (machine learning)

Training set

Set of data without redundancies (i.e. a set of non-homologous sequences). This is used to optimize the parameters describing the model

Test set

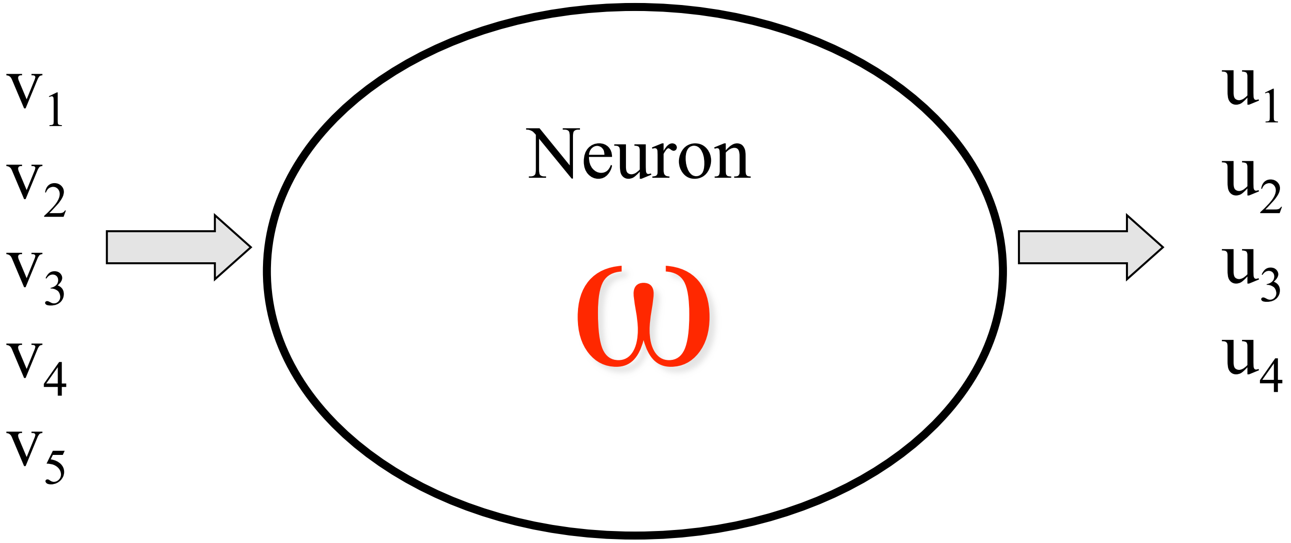
Set of data without any element used on the training set or similar to some element of the training set (i.e. a set of sequences non-homologous between them and non-homologous to any of the elements of the training set). This set is used to test the approach and validate the statistical accuracy of the method.

3. Secondary structure alignment

1. secondary structure prediction (Neural Network)

input $= \{v_i \,/\, v_i \; i=1,n\}$

output $= \{u_i \,/\, u_i \; j=1,m\}$



3. Secondary structure alignment

1. secondary structure prediction (Neural Network)

Parameters for the model: ω

$$x_j = \sum_k \omega_k^j v_k + \omega_0^j$$

$$y_j = f\left(x_j\right) = \frac{1}{1 + e^{-x_j}}$$

We need to optimize the parameters in order to get y_j as close as possible to u_j

3. Secondary structure alignment

1. secondary structure prediction (Neural Network)

Working hypothesis:

The error between the expected output values (u) and the output obtained with this "neuron" approach follows a multiple gaussian distribution. Therefore, the probability to obtain the output data, given the parameters of the neuron (ω and function f), is:

$$P(D \mid M) = P(u \mid \omega, f) = \prod_{j=1}^{m} \frac{1}{\sigma\sqrt{2\pi}} \times e^{\frac{-(u_j - y_j)^2}{2\sigma^2}}$$

$$\sigma = \sqrt{\frac{\sum_{j=1}^{m} (u_j - y_j)^2}{m - 1}}$$

3. Secondary structure alignment

1. secondary structure prediction (Neural Network)

Maximum Likelihood solution:

This implies we can solve the optimization by means of the maximum likelihood approach. It also can be further simplified by assuming a constant standard deviation.

$$\Phi \approx \sum_j \frac{1}{2\sigma^2}\left(u_j - y_j\right)^2 - \frac{1}{2}\log 2\pi - \log\sigma$$

$$0 = \frac{\partial\Phi}{\partial\omega_k^j} = -\frac{\left(u_j - y_j\right)}{\sigma^2} \times \frac{e^{-x_j}}{\left(1 + e^{-x_j}\right)^2} \times v_k$$

3. Secondary structure alignment

1. secondary structure prediction (Neural Network)

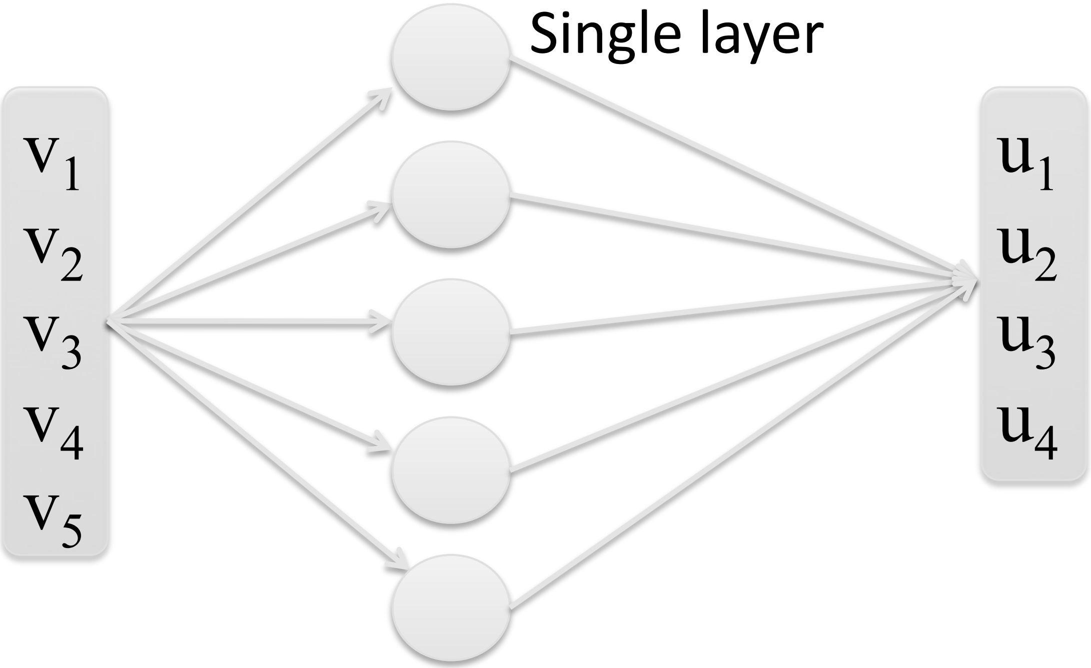
Neural Network

The protein sequence can be transformed into a set of vectors on the space of residues (dimension 20)

Inputs can check by windows of 15 Aa along the sequence

We can use more than one neuron, forming a layer of neurons.

We can add multiple layers formed by neurons.

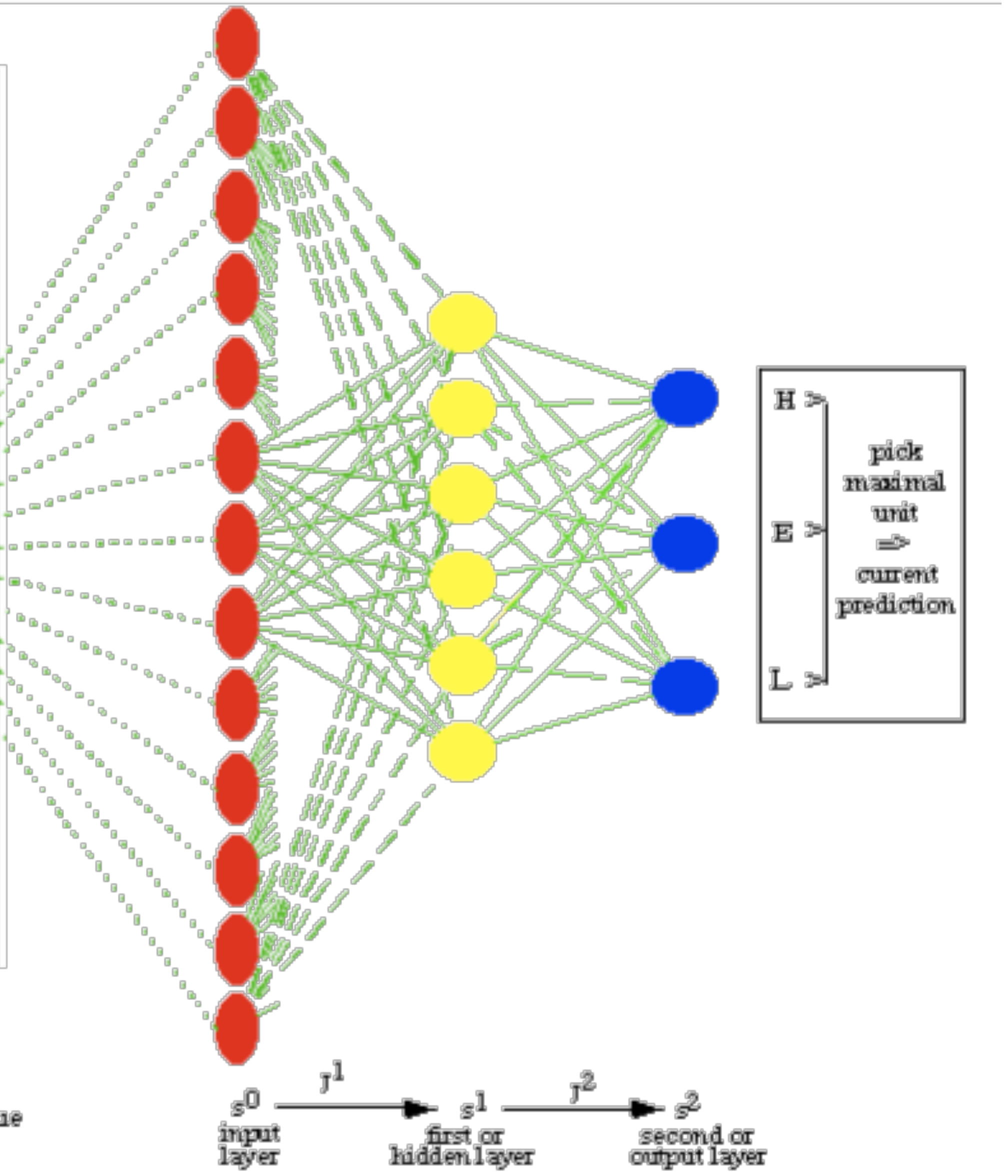


3. Secondary structure alignment

1. secondary structure prediction (Neural Network)

Neural Network (PHD)

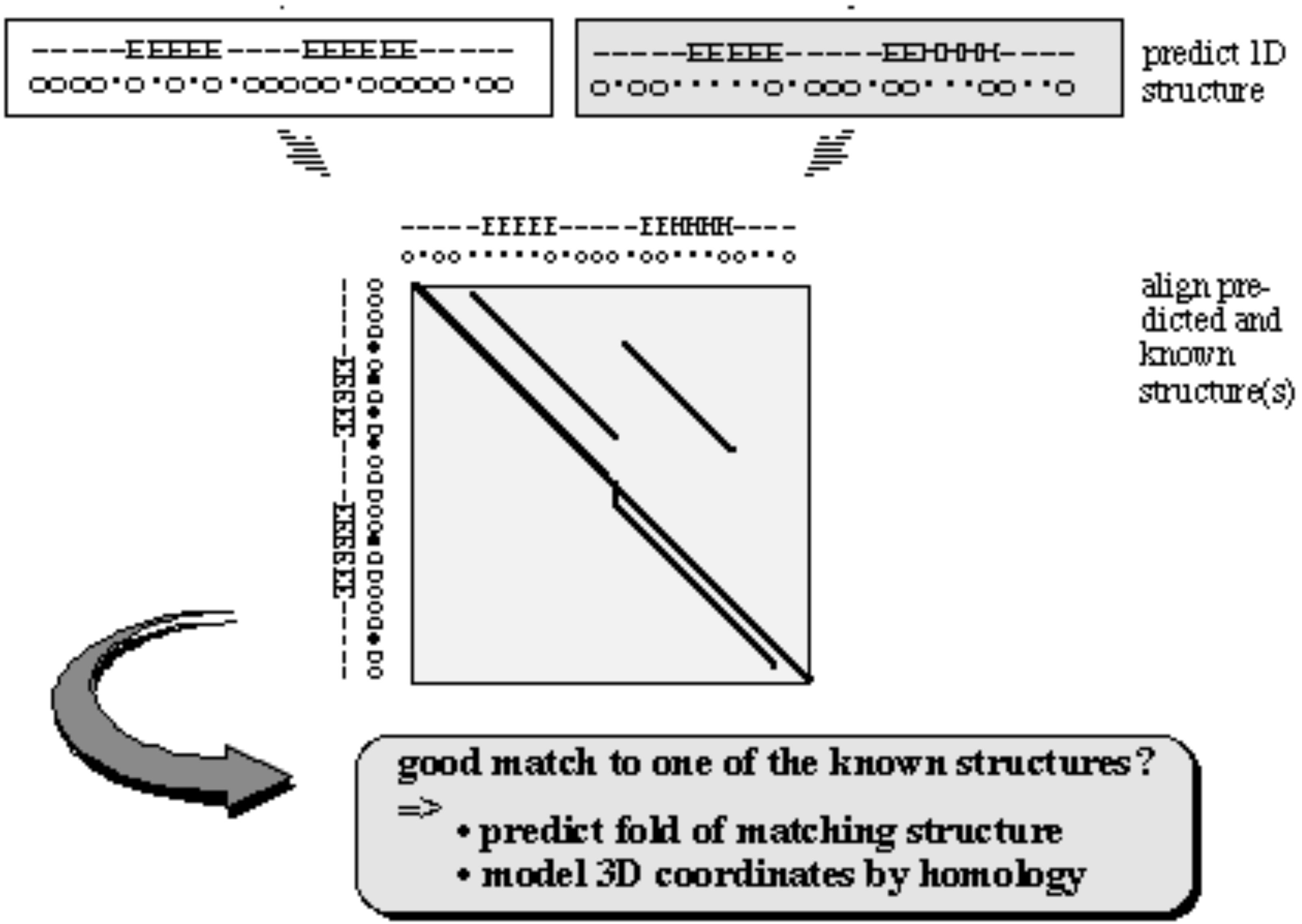
Protein	Alignments	profile table			
		GSAPD	NTEKQ	CVHIR	LMYFW
:	: : : :				
G	GGGG	5....	.....	.....	.....
Y	YYYY	.....	.....	.....	..5..
I	IIEE	.....	..2..	...3.	.....
Y	YYYY	.....	.....	.....	..5..
D	DDDD	5	.....	.....	.....
P	PPPP	...5.	.....	.....	.....
E	AEAA	..3..	..2..	.....	.....
D	VVEE	1	..2..	.2...	.....
G	GGGG	5....	.....	.....	.....
D	DDDD	5	.....	.....	.....
P	PPPP	...5.	.....	.....	.....
D	DTDD	4	.1...	.....	.....
D	NQNN	1	3...1	.....	.....
G	GNGG	4....	1....	.....	.....
V	VIVV	.....	.....	.4.1.	.....
N	EPKK	...1.	1.12.	.....	.....
P	PPPP	...5.	.....	.....	.....
G	GGGG	5....	.....	.....	.....
T	TTTT	.....	.5...	.....	.....
D	EKSA	.11.1	..11.	.....	.....
F	FFFF	.....	.....	.....	...5.
:	: : : :				



● corresponds to the the 21*3 bits coding for the profile of one residue

3. Secondary structure alignment

2. Method of fold recognition TOPITS and THREADER



-----EEEEE----EEEEEE-----
oooo•o•o•o•ooooo•ooooo•oo
-----EEEEE-----EEHHHH----
o•oo•••••o•ooo•oo•••oo••o

predict 1D structure

align predicted and known structure(s)

good match to one of the known structures?

=>

- predict fold of matching structure
- model 3D coordinates by homology

Fold prediction

2. *ab initio* fold prediction (Rosetta)

    1. Revisiting the knowledge-based potential
    2. New potential based on conditional probabilities
    3. 9-Fragment database of structures
    4. Simulated Annealing construction
    5. Mutual Information
    6. Examples

1. Revisiting the knowledge-based potential

Given the radius of gyration of a protein structure (RG), we approximate the probability that this is the structure for a given sequence, where the sequence is defined as the vector (aa_1, aa_2,aa_3,…..aa_N)

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

$$P(sequence \mid structure) = \prod_{i<j} P(aa_i, aa_j) \times \frac{P(r_{ij} \middle| aa_i, aa_j)}{P(r_{ij})}$$

$$P(structure \mid sequence) \cong e^{-RG^2} \times \prod_{i<j} \frac{P(r_{ij} \middle| aa_i, aa_j)}{P(r_{ij})} \qquad \text{(Equation 1)}$$

Where the term on the right contains the distance dependent knowledge-based potential: $P(r_{ij}|aa_i,aa_j)$ /$P(r_{ij})$

2. New potential based on conditional probabilities

By applying Bayes theorem on a sequence (set of elements amino-acids), we can approach the conditional probability with respect to the structure in which the sequence is folded with the first two terms of the expansion:

$$P(x_1,x_2,x_3,\ldots,x_n) \cong \prod_i P(x_i) \times \prod_{i<j} \frac{P(x_i,x_j)}{P(x_i)P(x_j)} \cdots$$

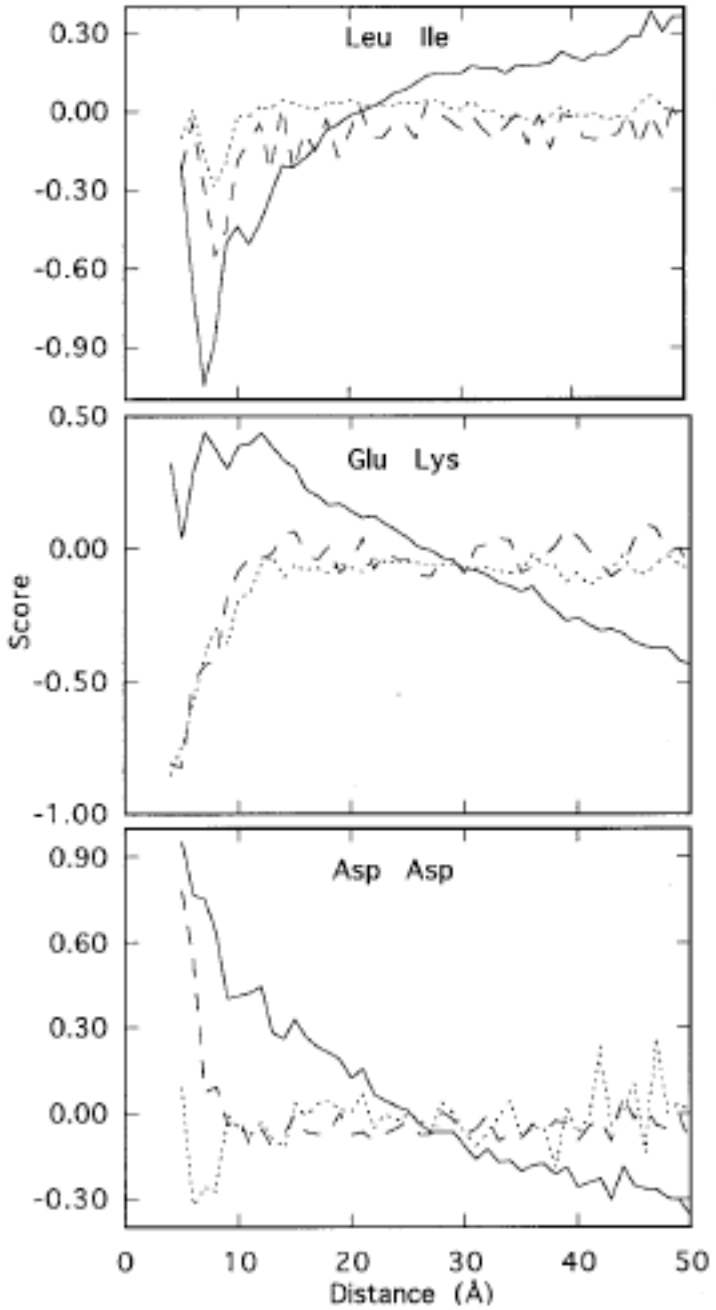
$$P(sequence \mid structure) = P(aa_1,aa_2,\ldots aa_n \mid structure)$$

$$P(aa_1,aa_2,\ldots aa_n \mid structure) \cong \prod_i P(aa_i|E_i) \times \prod_{i<j} \frac{P(aa_i,aa_j|r_{ij},E_i,E_j)}{P(aa_i|r_{ij},E_i,E_j)P(aa_j|r_{ij},E_i,E_j)}$$

$$P(structure \mid sequence) \cong e^{-RG^2} \times P(aa_1,aa_2,\ldots aa_n \mid structure)$$ (Equation 2)

Where E_i is the environment (secondary structure, accessibility, etc.) of resi

2. New potential based on conditional probabilities



Example of differences between potentials calculated with equation 1 and equation 2.

Equation 1 is in continuous line

Equation 2 for two buried residues is in dotted line

Equation 2 for two exposed residues is in dashed line

3. 9-Fragment database of structures

Rosetta splits the sequence in fragments of 9 residues, using a window-like method

Rosetta contains a database of 9-residue fragments extracted from the total set of protein structures

Rosetta assigns the first 25 most probable 9-fragment segments to a 9-residue fragment of the target sequence by selecting those with smallest score:

$$score = \sum_{i=1}^{9} \sum_{aa=1}^{20} \left| S(aa,i) - X(aa,i) \right|$$

Where S(aa,i) is the frequency of residue aa in position i of the target sequence and its homologs in the same 9-residues fragment. Similarly, X(aa,i) is the frequency of amino-acid aa in position i for all similar 9-residue fragments (with the same structure)

4. Simulated annealing construction

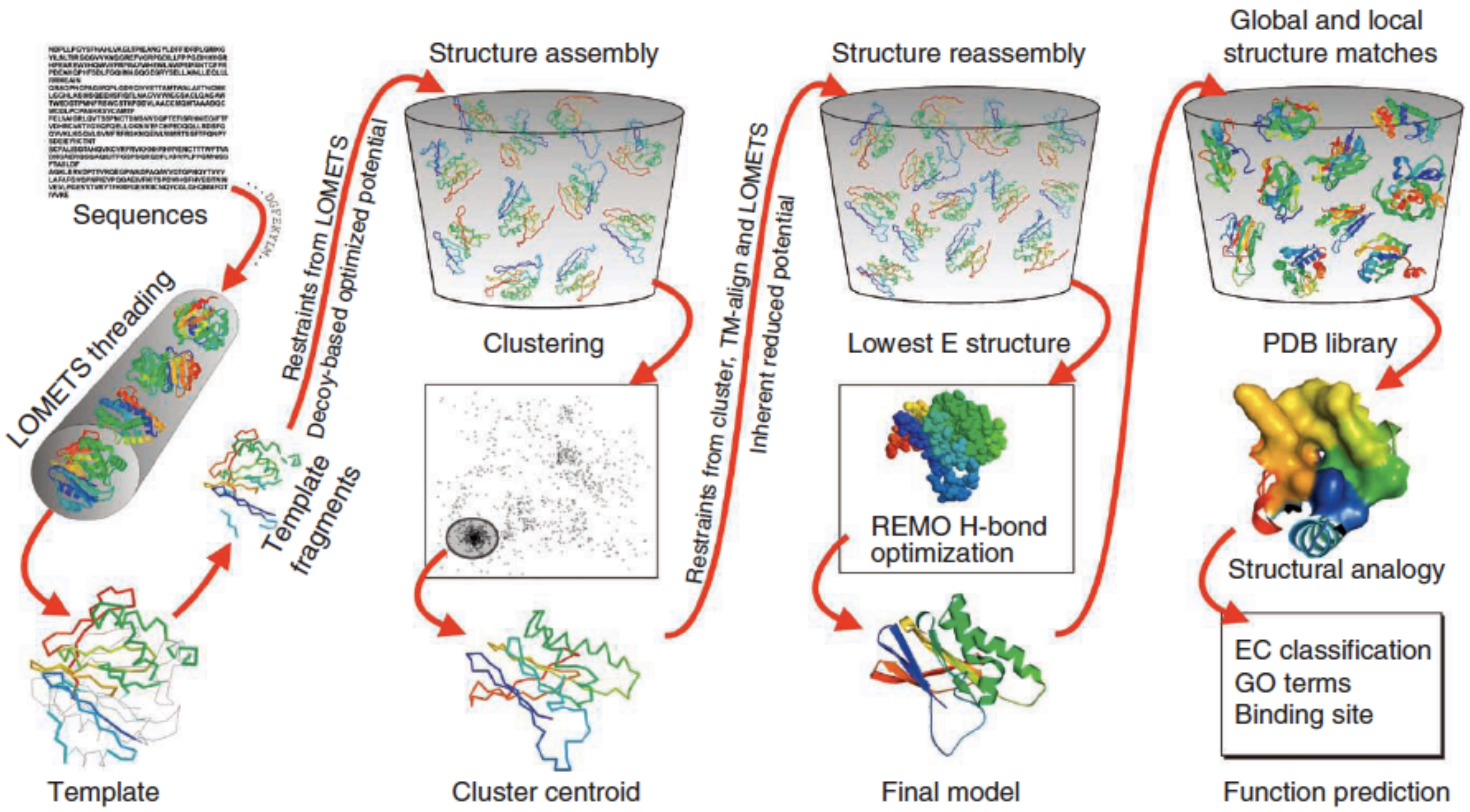
Rosetta applies small changes in torsional angles for each fragment considered in order to join the 9-residue fragmented structures assigned to the 9-residue segment of the target

A conformation is selected according to the most probable structure-score: P(structure|sequence). A Metropolis-Montecarlo simulation is applied using a simulated annealing

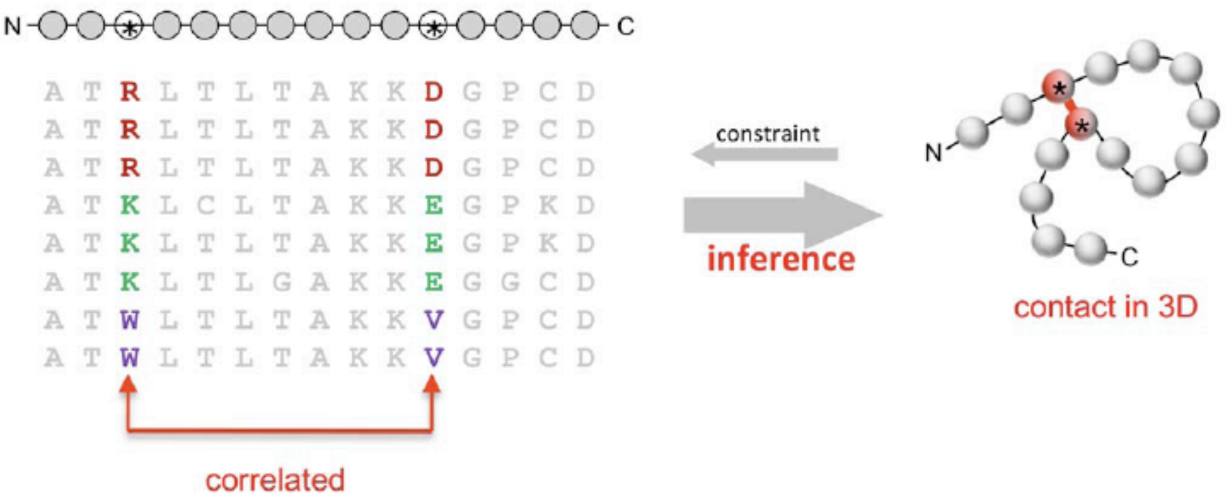
The structure-score is first calculated with equation 1, and when the simulation obtains a closer and more definite structure equation 2 (with more detailed potential) is applied.

5. iTASSER

iTASSER uses LOMETS threading. LOMETS uses the results of several threading approaches based on remote homology (i.e. FUGUE, HHSEARCH, etc.) and selects the common fragment-templates to assemble the target structure. Then it follows a similar approach to Rosetta

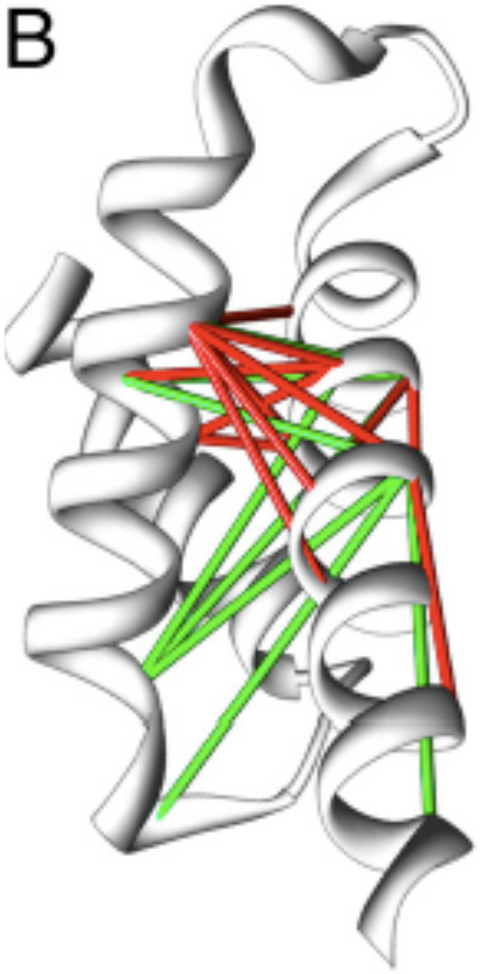


5. Mutual Information



```
A T R L T L T A K K D G P C D
A T R L T L T A K K D G P C D
A T R L T L T A K K D G P C D
A T K L C L T A K K E G P K D
A T K L T L T A K K E G P K D
A T K L T L G A K K E G G C D
A T W L T L T A K K V G P C D
A T W L T L T A K K V G P C D
```

$$\mathrm{MI}_{ij} = \sum_{A,B} f_{ij}(A,B)\ \ln \frac{f_{ij}(A,B)}{f_i(A) f_j(B)},$$



top MI pairs

5. Mutual Information

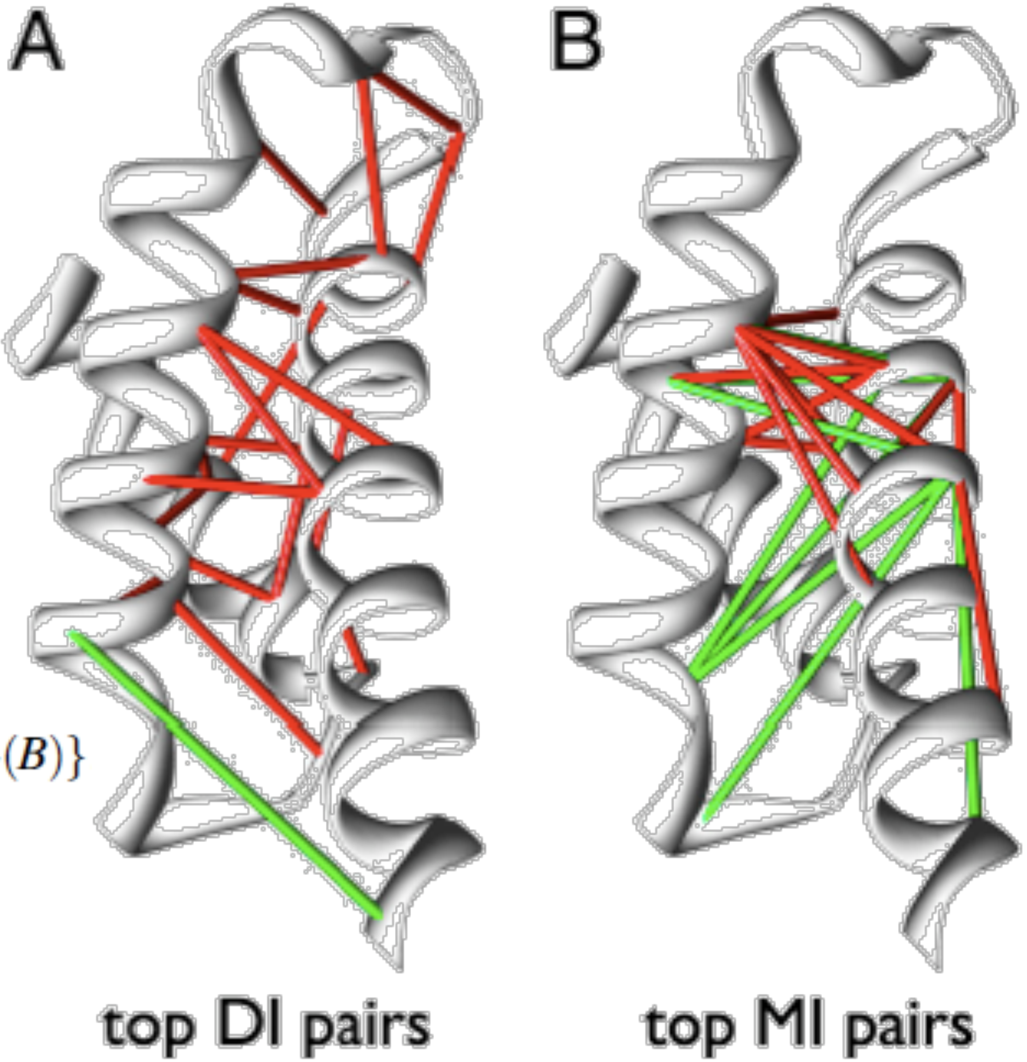
$$\mathrm{MI}_{ij} = \sum_{A,B} f_{ij}(A,B)\ \ln \frac{f_{ij}(A,B)}{f_i(A) f_j(B)},$$

$$\mathrm{DI}_{ij} = \sum_{AB} P^{(\mathrm{dir})}_{ij}(A,B)\ \ln \frac{P^{(\mathrm{dir})}_{ij}(A,B)}{f_i(A)\, f_j(B)}.$$

$$f_i(A) = \sum_{B} P^{(\mathrm{dir})}_{ij}(A,B),$$

$$f_j(B) = \sum_{A} P^{(\mathrm{dir})}_{ij}(A,B).$$

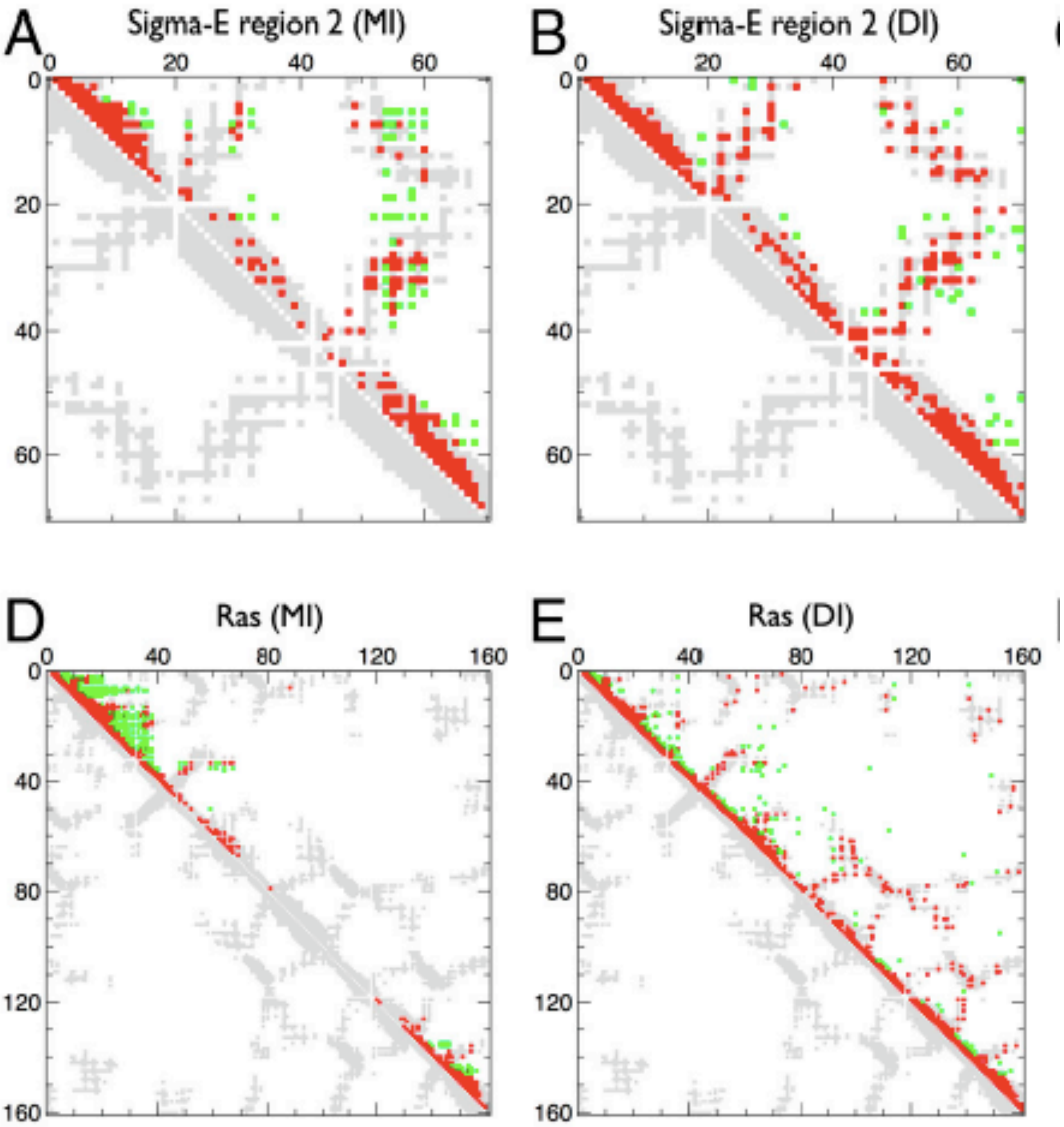
$$P^{(\mathrm{dir})}_{ij}(A,B) = \frac{1}{Z_{ij}}\ \exp\{e_{ij}(A,B) + \tilde{h}_i(A) + \tilde{h}_j(B)\}$$



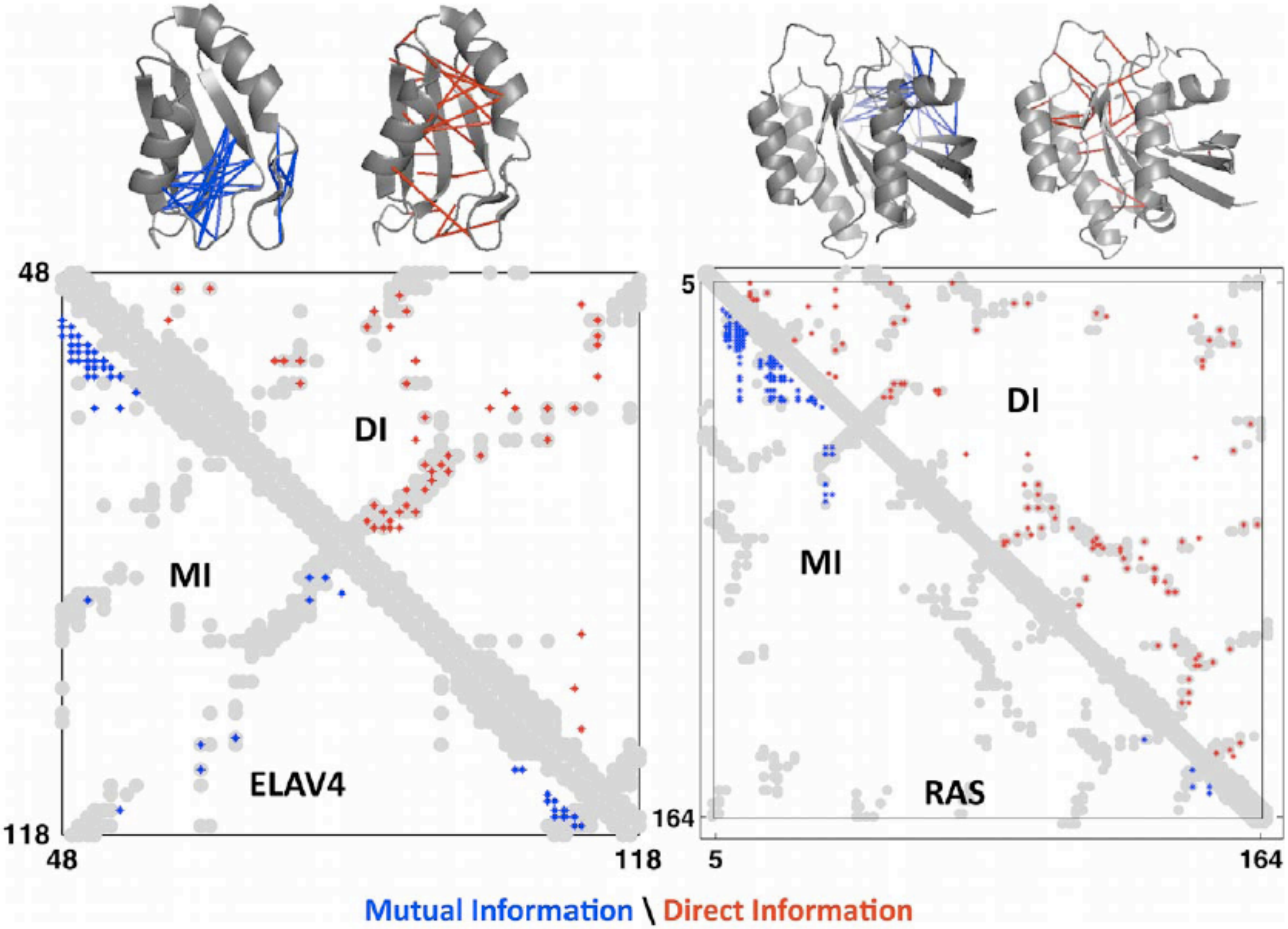
Marks DS et al.. PLoS One. 2011;6(12):e28766. Epub 2011

Morcos F, et al. . Proc Natl Acad Sci U S A. 2011 Dec 6;108(49):E1293-301.

5. Mutual Information

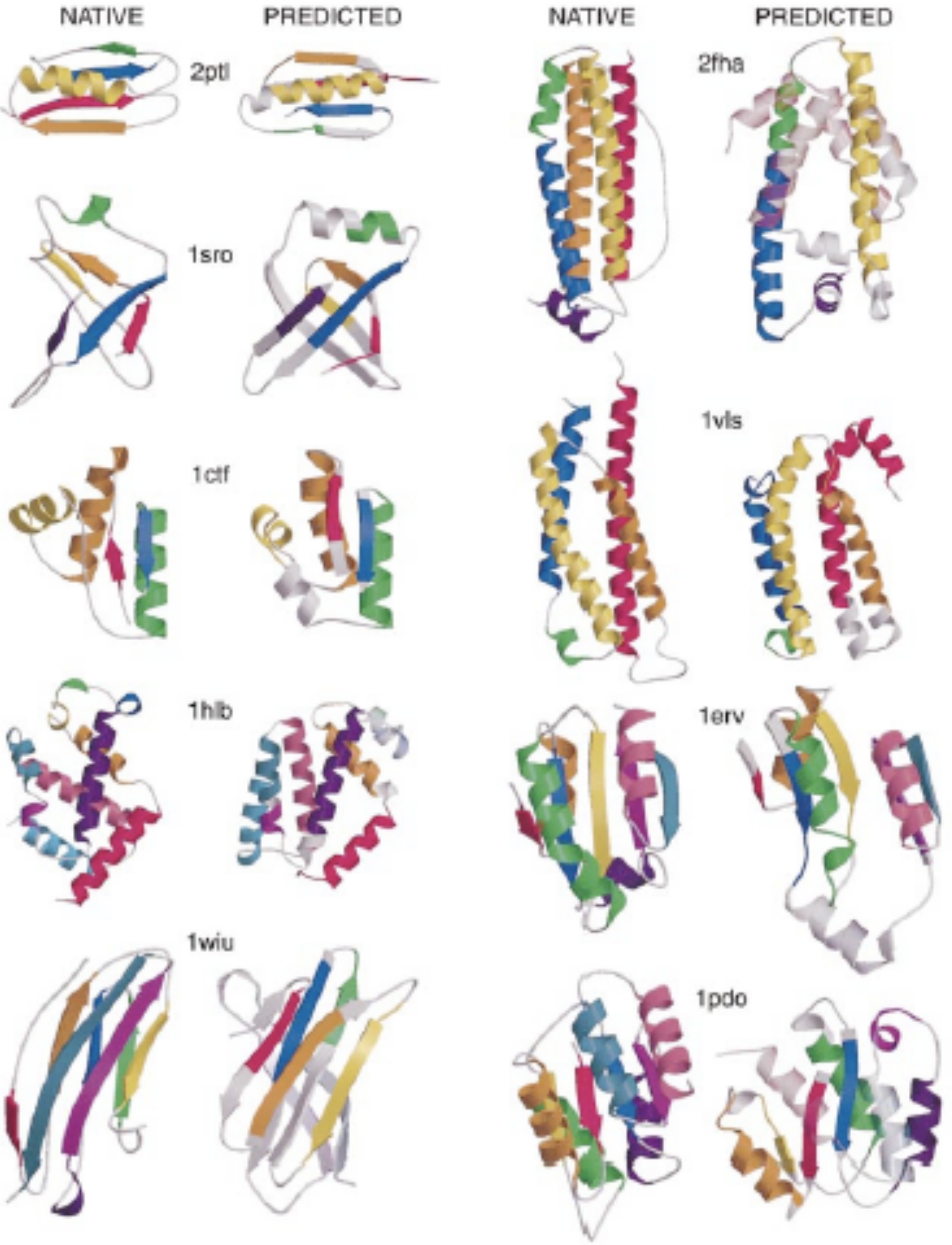


5. Mutual Information



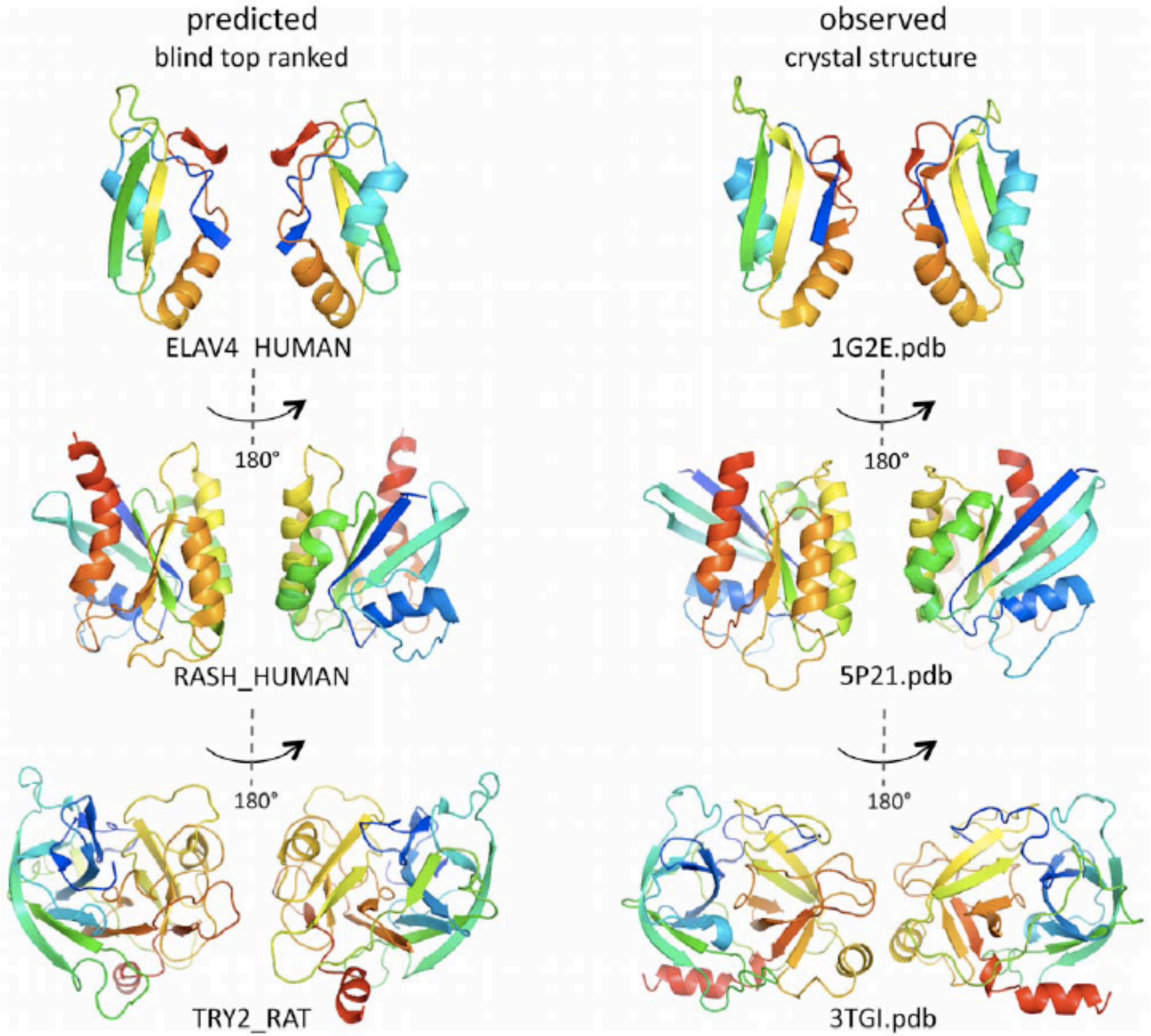
6. Examples

Rosetta



6. Examples

Direct information


predicted
blind top ranked
observed
crystal structure
ELAV4 HUMAN
1G2E.pdb
180°
180°
RASH_HUMAN
5P21.pdb
180°
180°
TRY2_RAT
3TGI.pdb