

Accurate prediction for atomic-level protein design and its application in diversifying the near-optimal sequence space

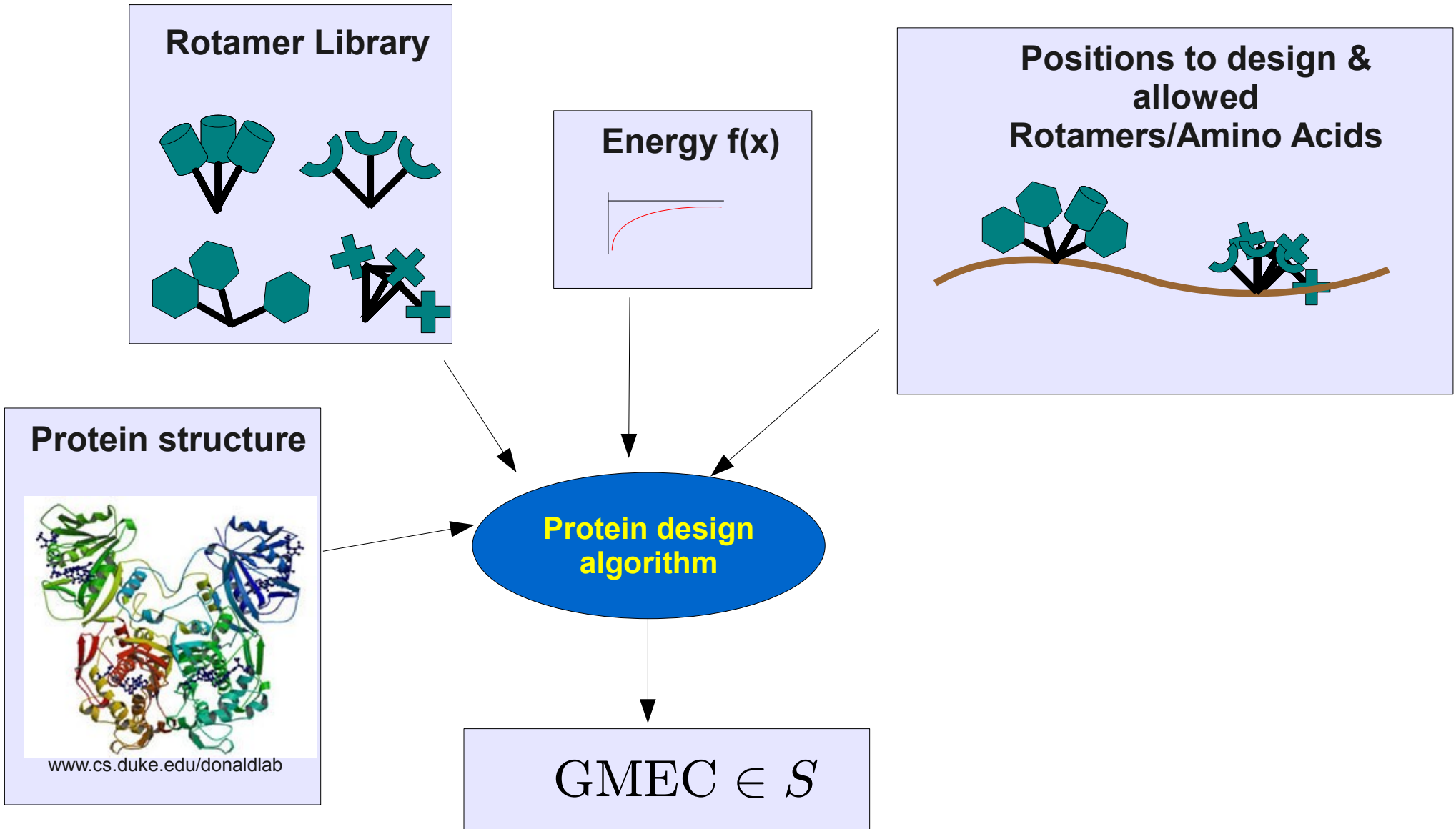
Pablo Gainza

CPS 296: Topics in Computational Structural Biology
Department of Computer Science
Duke University

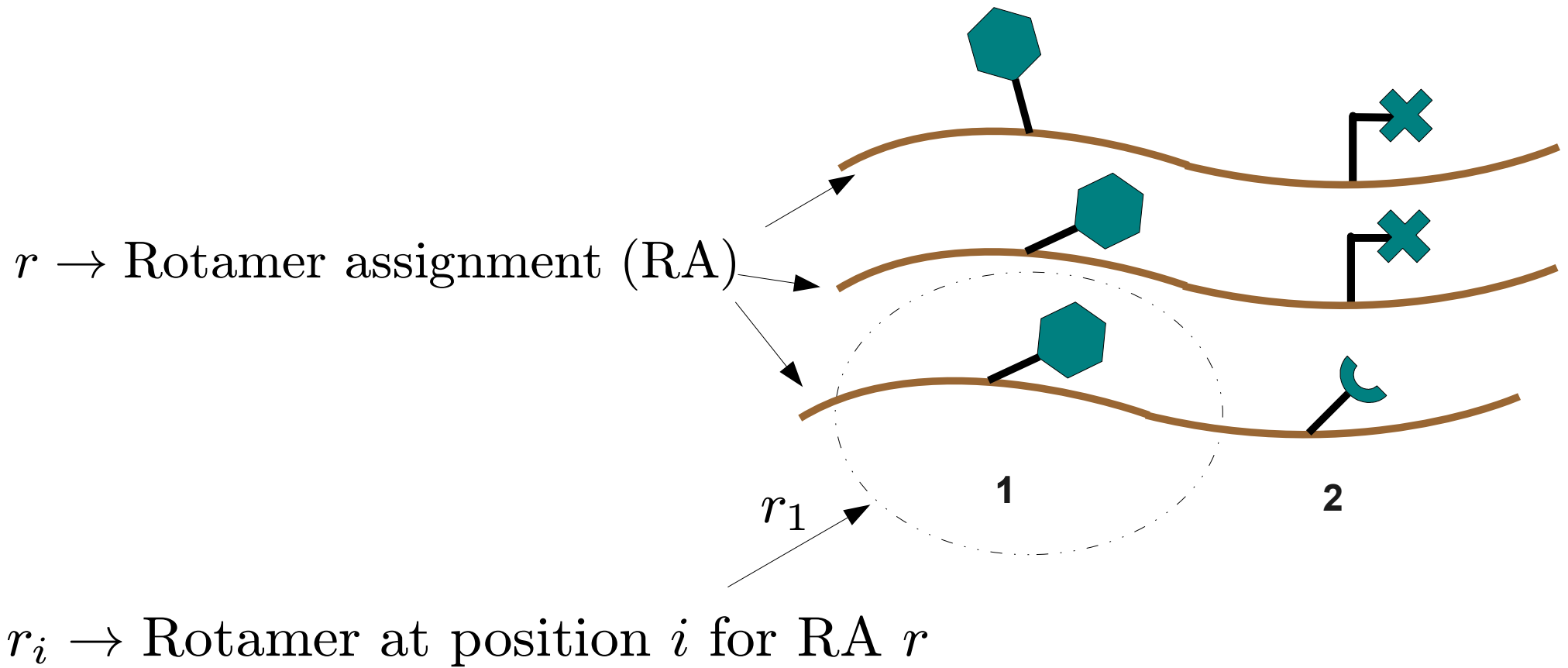
Outline

- 1) Problem definition
- 2) Formulation as an inference problem
- 3) Graphical Models
- 4) tBMMF algorithm
- 5) Results
- 6) Conclusions

1. Problem Definition



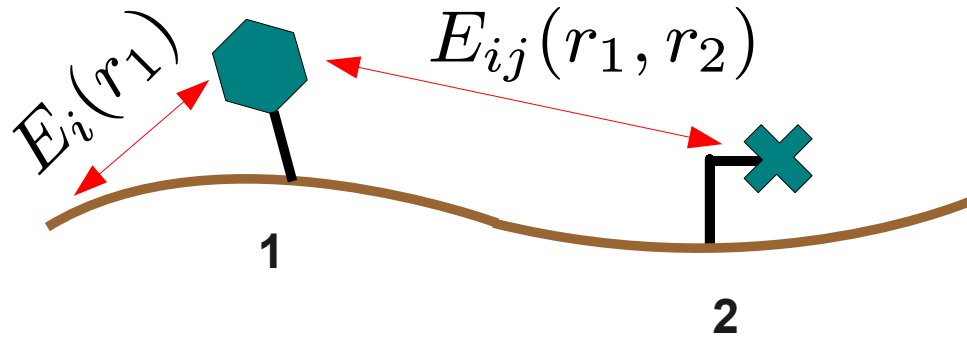
1. Problem Definition (2)



1. Problem Definition (3)

$E_i(r_i) \rightarrow$ Energy between rotamer r_i and fixed backbone

$E_{ij}(r_i, r_j) \rightarrow$ Energy between rotamers r_i and r_j



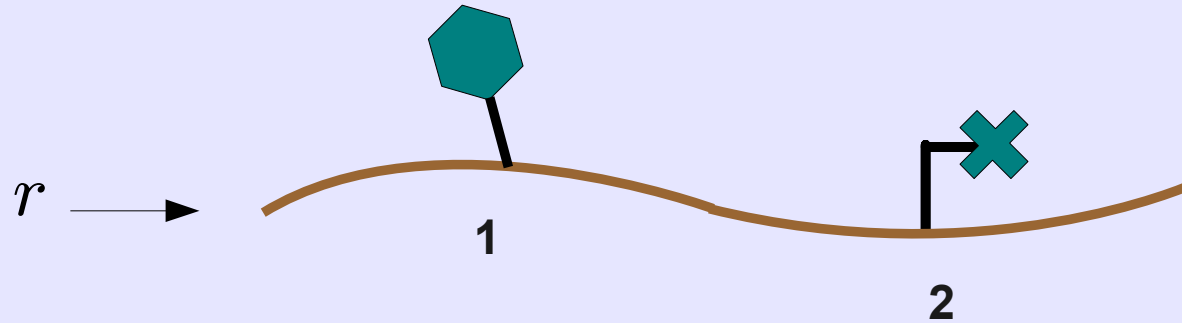
$E(r) \rightarrow$ Energy of rotamer assignment r

$$E(r) = \sum_i E_i(r_i) + \sum_{i,j} E_{ij}(r_i, r_j)$$

1. Problem Definition (4)

$T(k) \rightarrow$ returns amino acid type of rotamer k

$T(r) \rightarrow$ returns sequence of rotamer assignment r

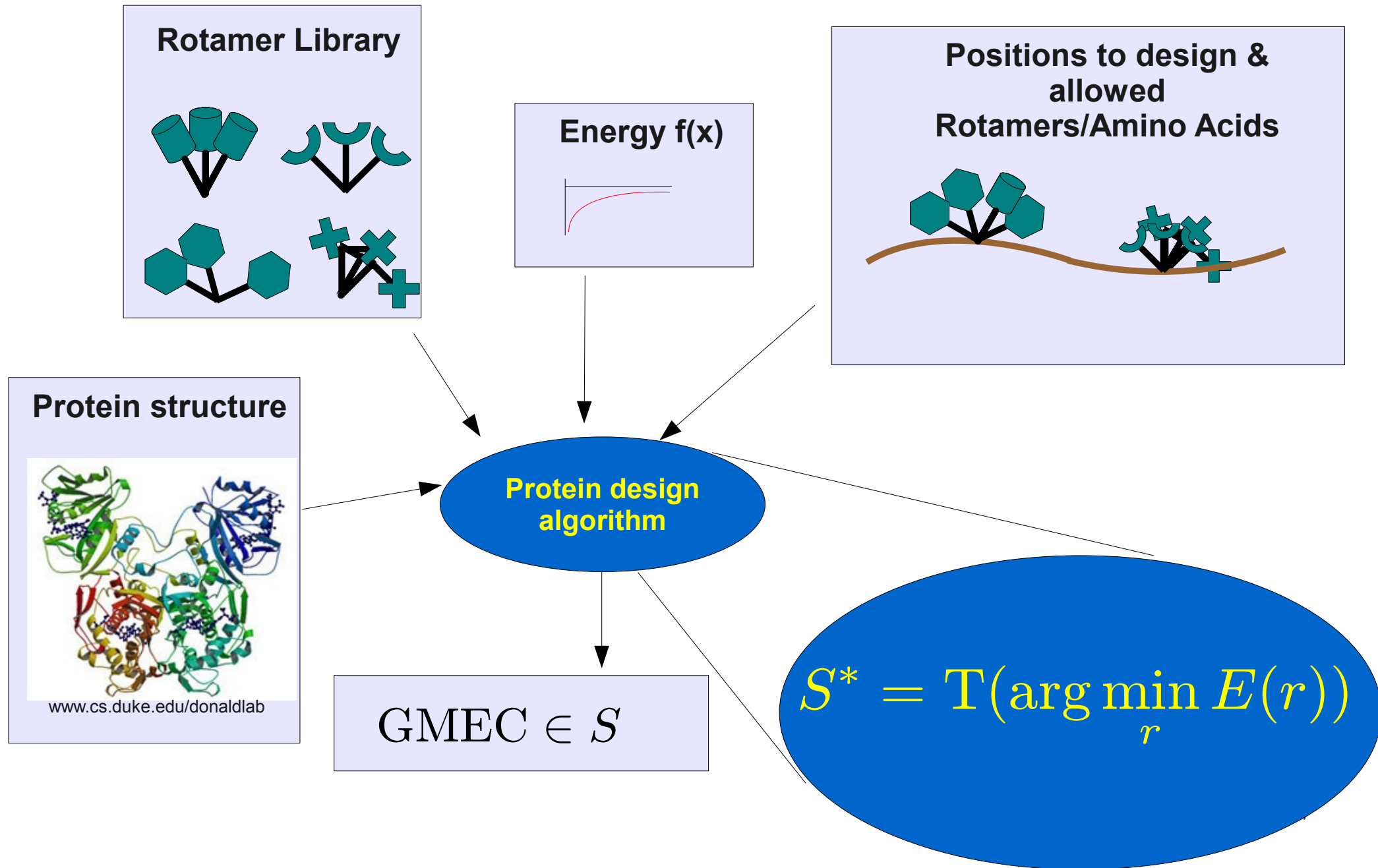


$T(r_1) = \text{hexagon}$

$T(r_2) = \text{cross}$

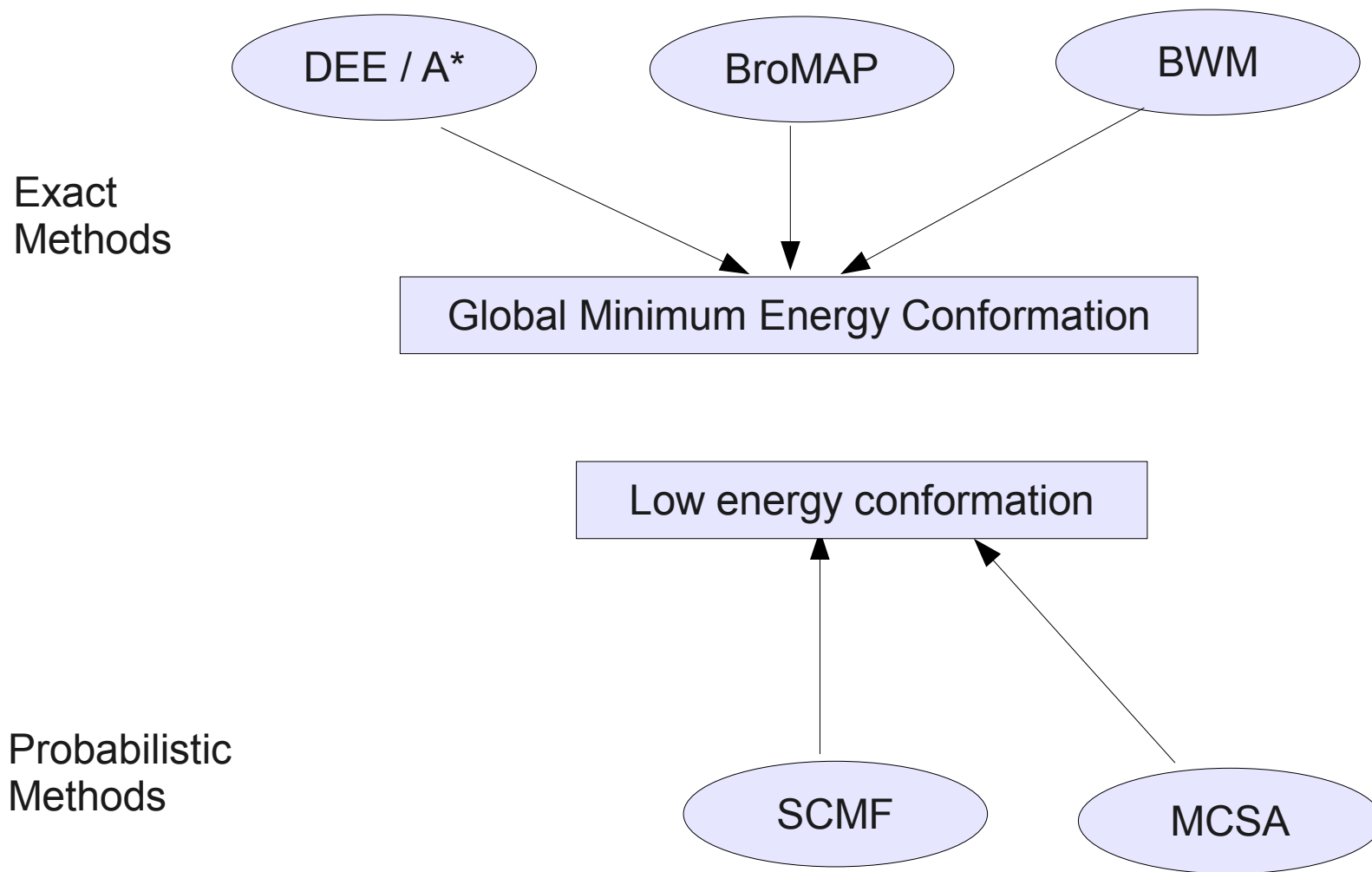
$T(r) = \text{hexagon, cross}$

1. Problem Definition (5)



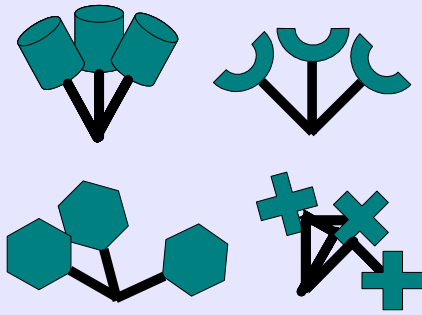
1. Problem Definition (6)

Related Work



1. Problem Definition (7)

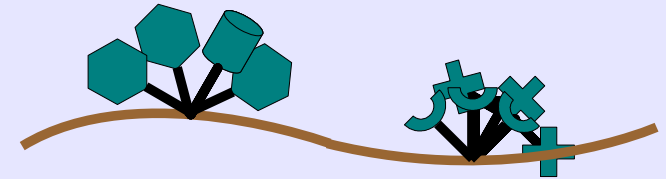
Rotamer Library



Energy $f(x)$



Positions to design & allowed Rotamers/Amino Acids



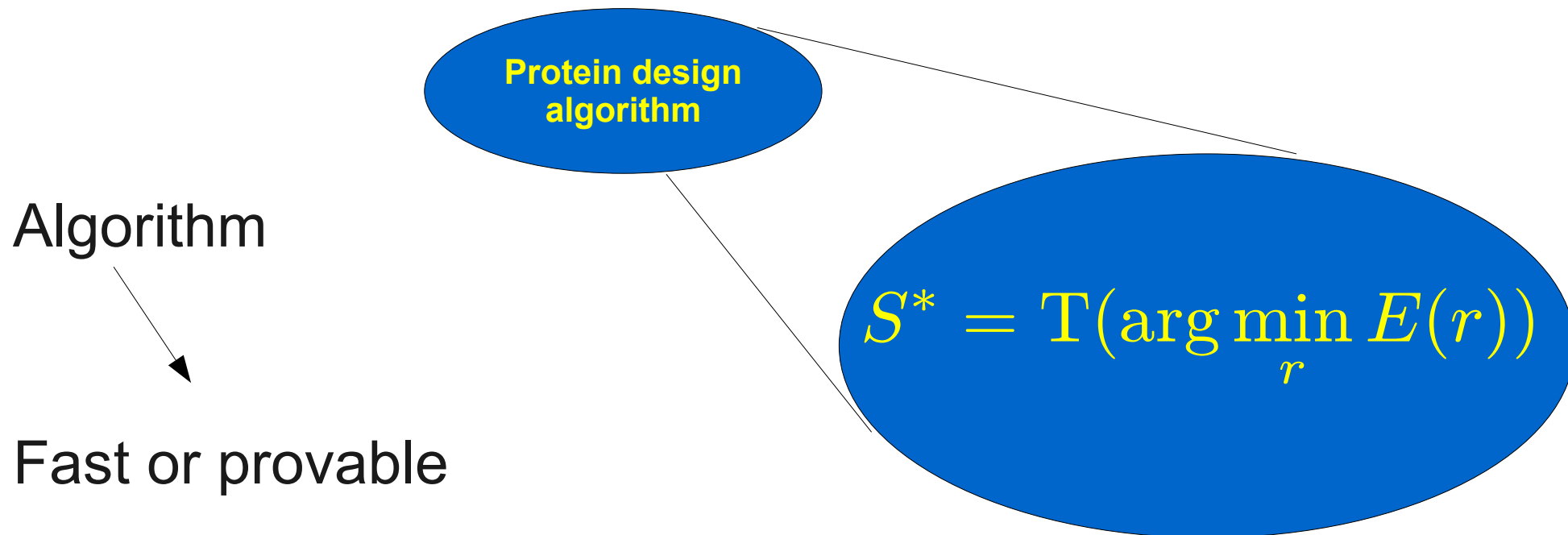
Protein structure



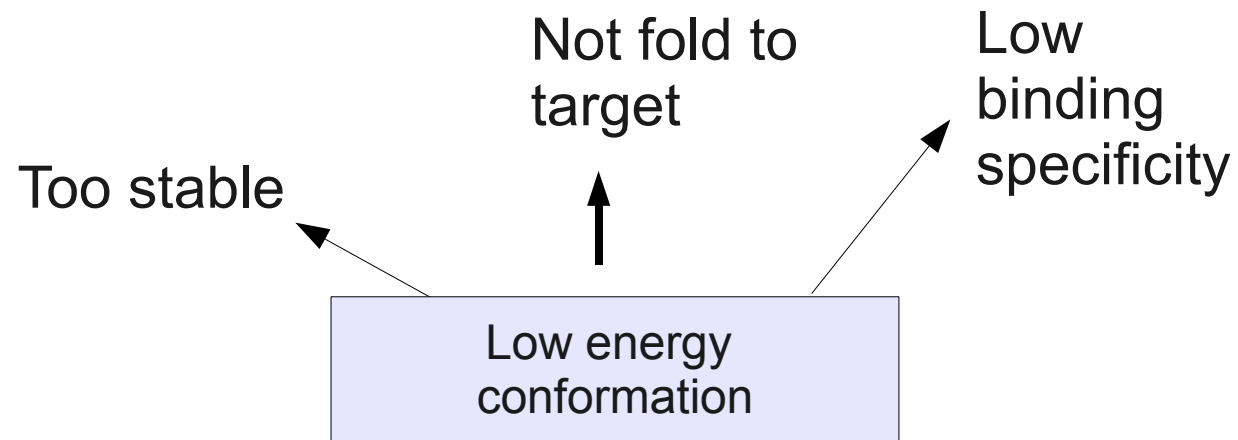
www.cs.duke.edu/donaldlab

Model \longrightarrow Inaccurate!

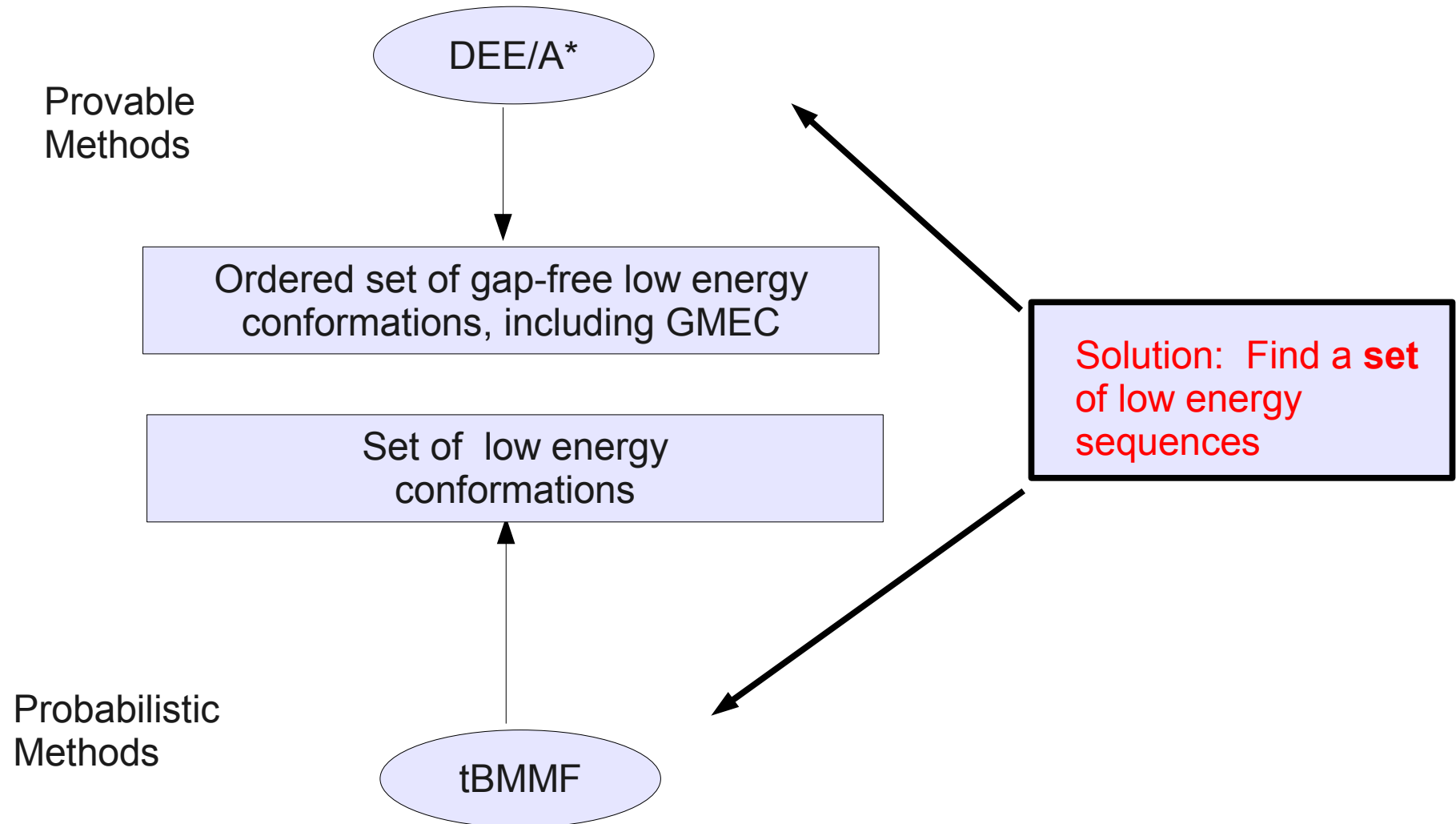
1. Problem Definition (8)



1. Problem Definition (9)



1. Problem Definition (10)



Problem Definition: Summary

- Protein design algorithms search for the **sequence** with the Global Minimum Energy Conformation (**GMEC**).
- Our model is **inaccurate**: more than one low energy sequence is desirable.
- Fromer et al. Propose **tBMMF** to generate a **set** of low energy sequences.

2. Our problem as an inference problem

Probabilistic factor for self-interactions

$$\psi_i(r_i) = e^{\frac{-E_i(r_i)}{T}}$$

Probabilistic factor for pairwise interactions

$$\psi_{ij}(r_i, r_j) = e^{\frac{-E_{ij}(r_i, r_j)}{T}}$$

2. Inference problem (2)

Partition function

$$Z = \sum_r e^{\frac{E(r)}{T}}$$

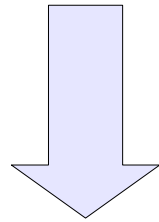
Probability **distribution** for rotamer assignment r

$$P(r_1, \dots, r_N) = \frac{1}{Z} \prod_i \psi_i(r_i) \prod_{i,j} \psi_{ij}(r_i, r_j) = \frac{1}{Z} e^{\frac{-E(r)}{T}}$$

2. Inference problem (3)

Minimization goal (from **definition**)

$$S^* = T(\arg \min_r E(r))$$





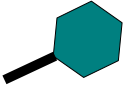
Minimization goal for a graphical model problem

$$S^* = T(\arg \max_r Pr(r))$$

2. Inference problem (4)

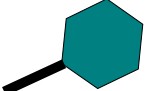
$$E_{ij}(r_1, r_2)$$



Position #1

Position #2			
		-4	-2

$$E_i(r_1)$$

$$E_i(r_2)$$

	-1
--	----

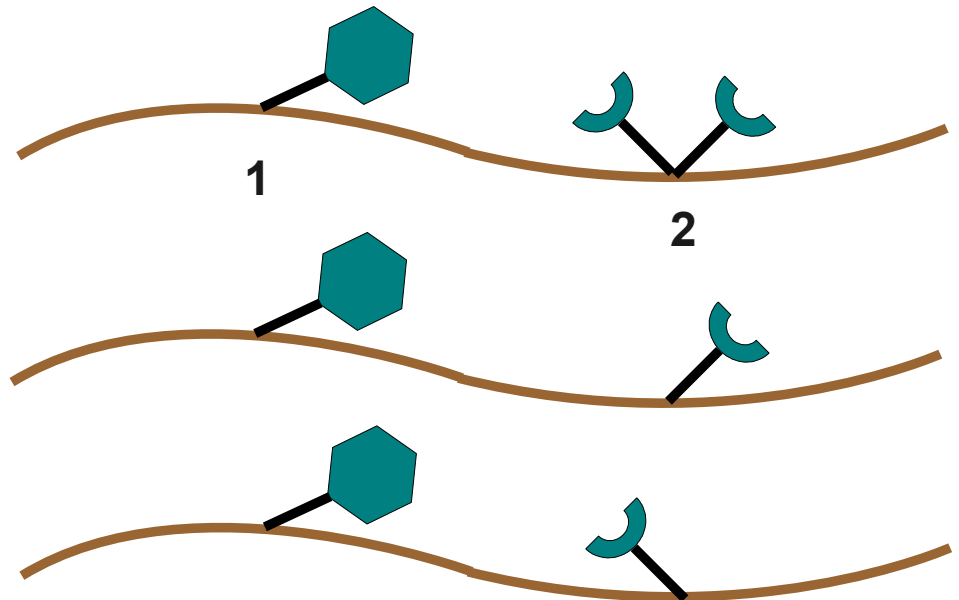
	-5
	-3

Example: Inference problem

Allowed

r'

r''



$$E(r') = ?$$



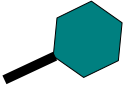
$$E(r'') = ?$$

What is our GMEC??

2. Inference problem (5)

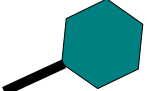
$$E_{ij}(r_1, r_2)$$



Position #1

Position #2			
		-4	-2

$$E_i(r_1)$$

$$E_i(r_2)$$

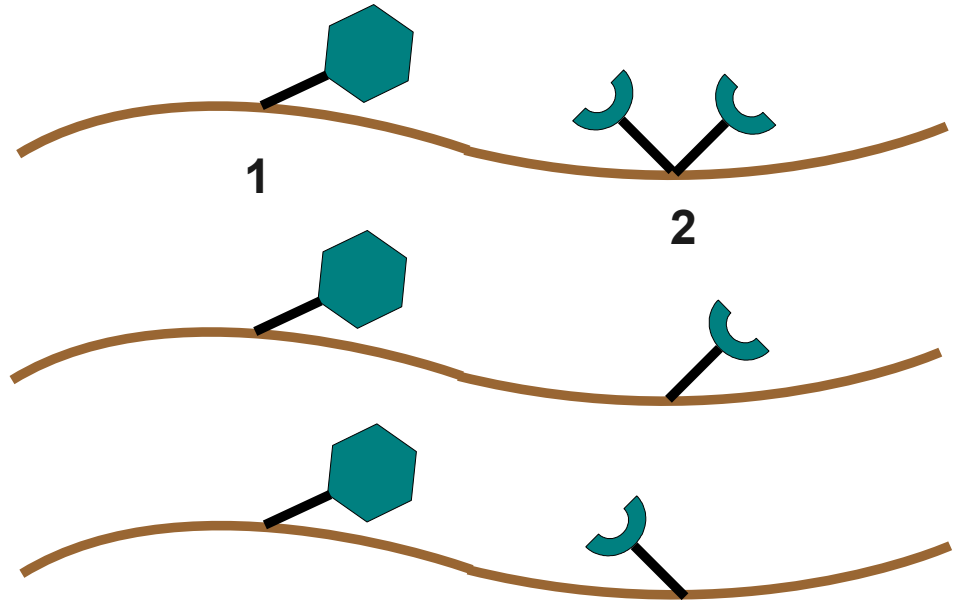
	-1
--	----

	-5
	-3

Allowed

r'

r''



$$\begin{aligned} E(r') &= (-1 + -2) + (-5 + -2) \\ &= -10 \end{aligned}$$



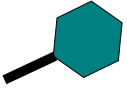
$$\begin{aligned} E(r'') &= (-1 + -4) + (-3 + -4) \\ &= -12 \end{aligned}$$

r'' is our GMEC

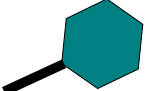
2. Inference problem (6)

$$E_{ij}(r_1, r_2)$$



Position #1

Position #2			
		-4	-2

$$E_i(r_1)$$

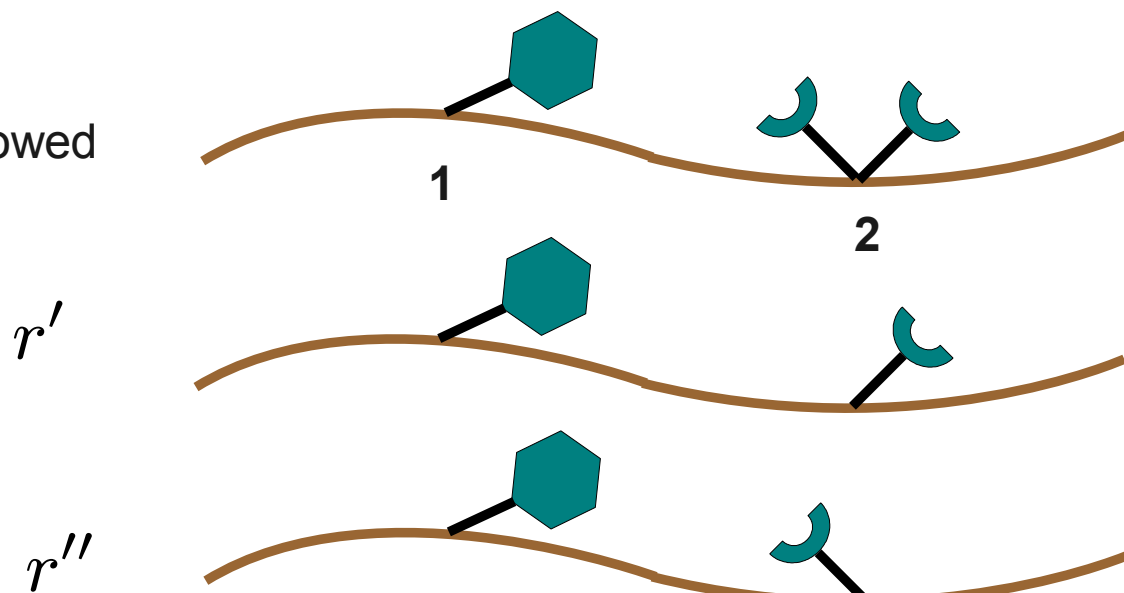
	-1
--	----

$$E_i(r_2)$$

	-5
	-3

$T = 1$ (for our example)

Allowed



$$\psi_i(r'_1) = e^{\frac{-E_i(r'_1)}{T}} = e$$

$$\psi_i(r'_2) = e^{\frac{-E_i(r'_2)}{T}} = e^5$$



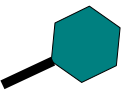
$$\psi_i(r''_1) = e^{\frac{-E_i(r''_1)}{T}} = e$$

$$\psi_i(r''_2) = e^{\frac{-E_i(r''_2)}{T}} = e^3$$

2. Inference problem (7)

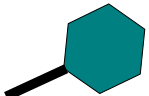


$$E_{ij}(r_1, r_2)$$

Position #1

		
 Position #2	-4	-2

$$E_i(r_1)$$

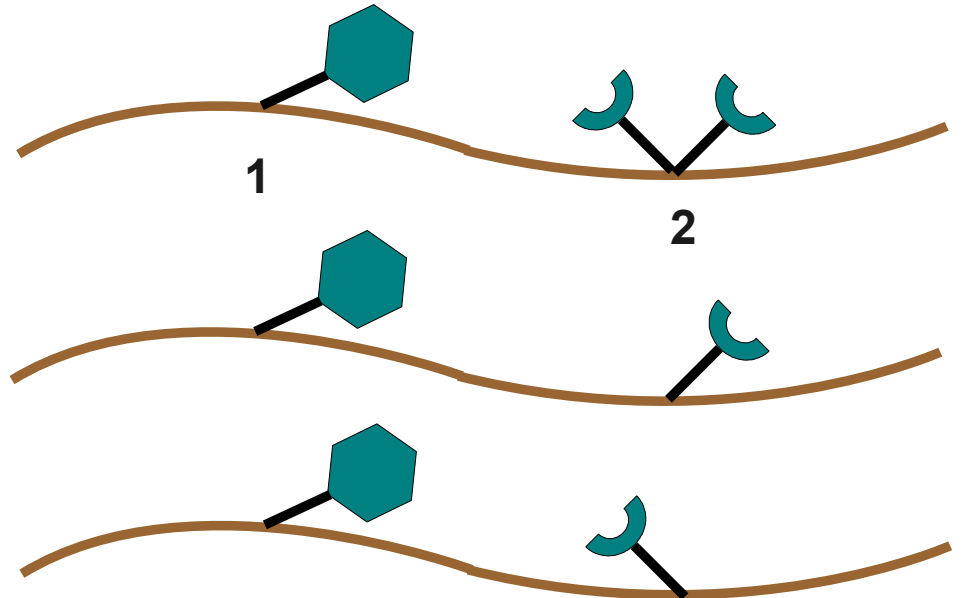
$$E_i(r_2)$$

	-1
	-5
	-3

Allowed

r'

r''



$$\psi_{ij}(r'_1, r'_2) = e^{\frac{-E_{ij}(r'_1, r'_2)}{T}} = e^2$$

$$\psi_{ij}(r''_1, r''_2) = e^{\frac{-E_{ij}(r''_1, r''_2)}{T}} = e^4$$



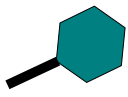
$$Z = \sum_r e^{\frac{E(r)}{T}} = e^{10} + e^{12}$$

$T = 1$ (for our example)

2. Inference problem (8)

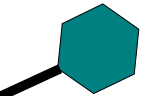
$$E_{ij}(r_1, r_2)$$



Position #1

Position #2			
		-4	-2

$$E_i(r_1)$$

$$E_i(r_2)$$

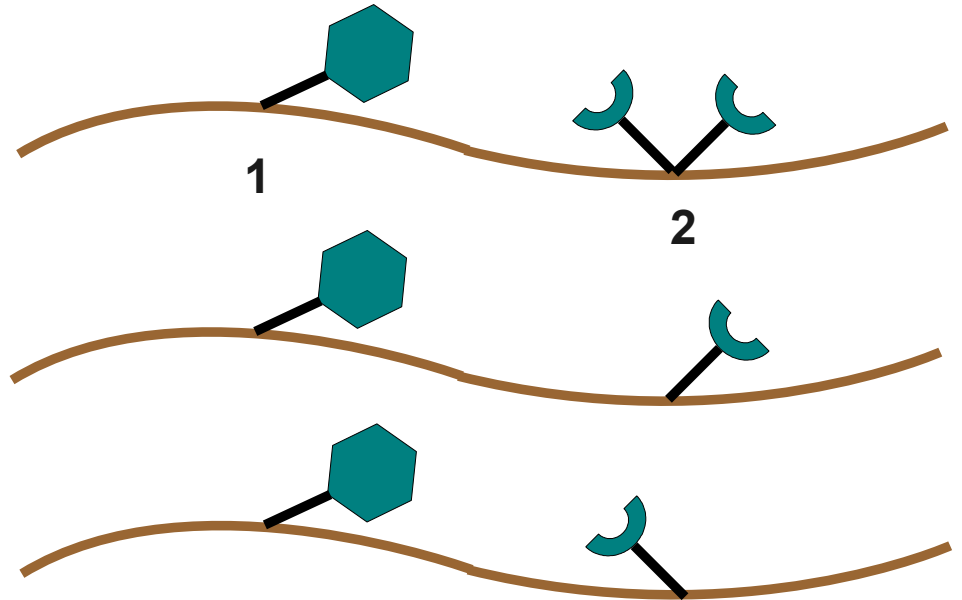
	-1
--	----

	-5
	-3

Allowed

r'

r''



$$P(r'_1, r'_2) = \frac{1}{Z} \prod_i \psi_i(r'_i) \prod_{i,j} \psi_{ij}(r'_i, r'_j)$$



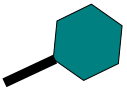
$$= \frac{e^{10}}{e^{10} + e^{12}}$$

$T = 1$ (for our example)

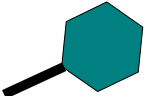
2. Inference problem (9)

$$E_{ij}(r_1, r_2)$$



Position #1

Position #2			
		-4	-2

$$E_i(r_1)$$

	-1
--	----

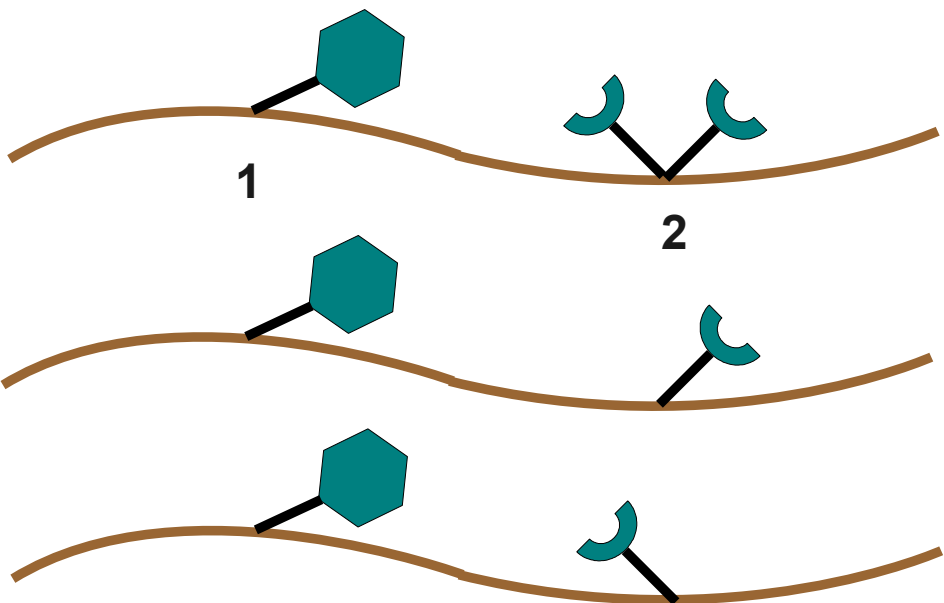
$$E_i(r_2)$$

	-5
	-3

Allowed

r'

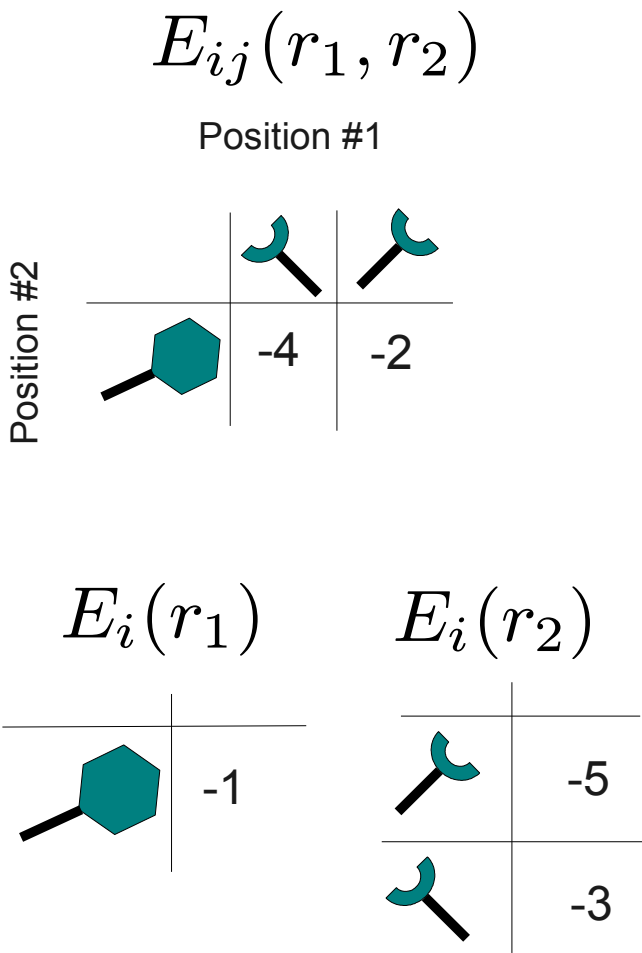
r''



$$\begin{aligned}
 P(r_1'', r_2'') &= \frac{1}{Z} \prod_i \psi_i(r_i') \prod_{i,j} \psi_{ij}(r_i'', r_j'') \\
 &= \frac{e^{12}}{e^{10} + e^{12}}
 \end{aligned}$$

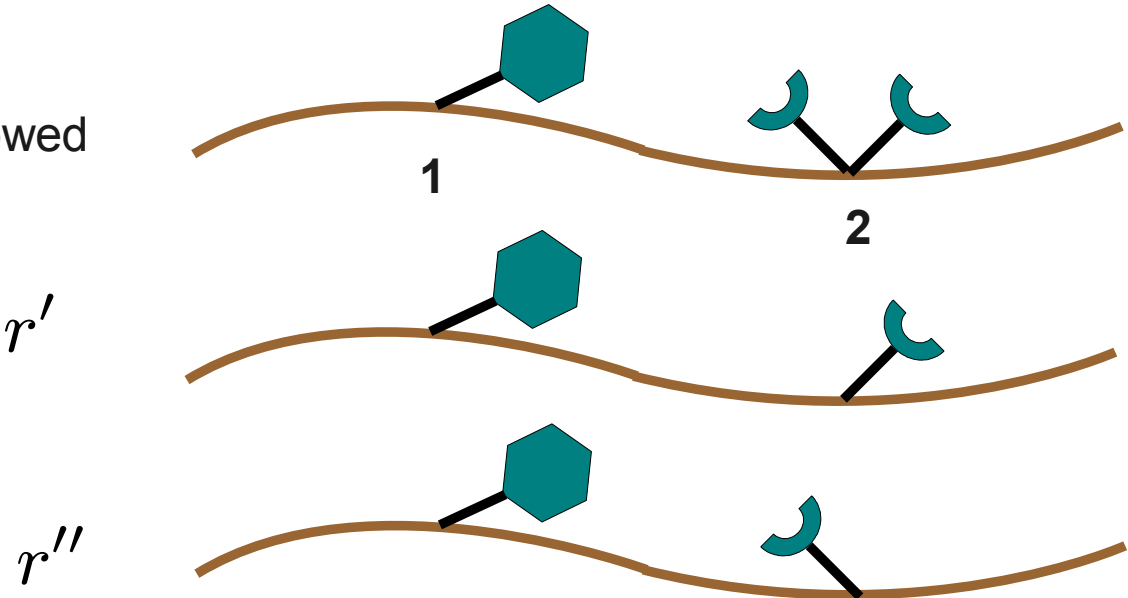
$T = 1$ (for our example)

2. Inference problem (10)



$T = 1$ (for our example)

Allowed



$$S^* = \text{T}(\arg \max_r Pr(r))$$

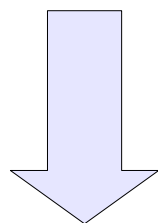
↓

$$S^* = \text{T}(r'')$$

2. Inference problem (11)

Minimization goal (from **definition**)

$$S^* = T(\arg \min_r E(r))$$

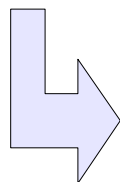


Minimization goal for a graphical model problem

$$S^* = T(\arg \max_r Pr(r))$$



We still have a non-polynomial problem!



But formulated as an
inference problem



**Probabilistic
methods**

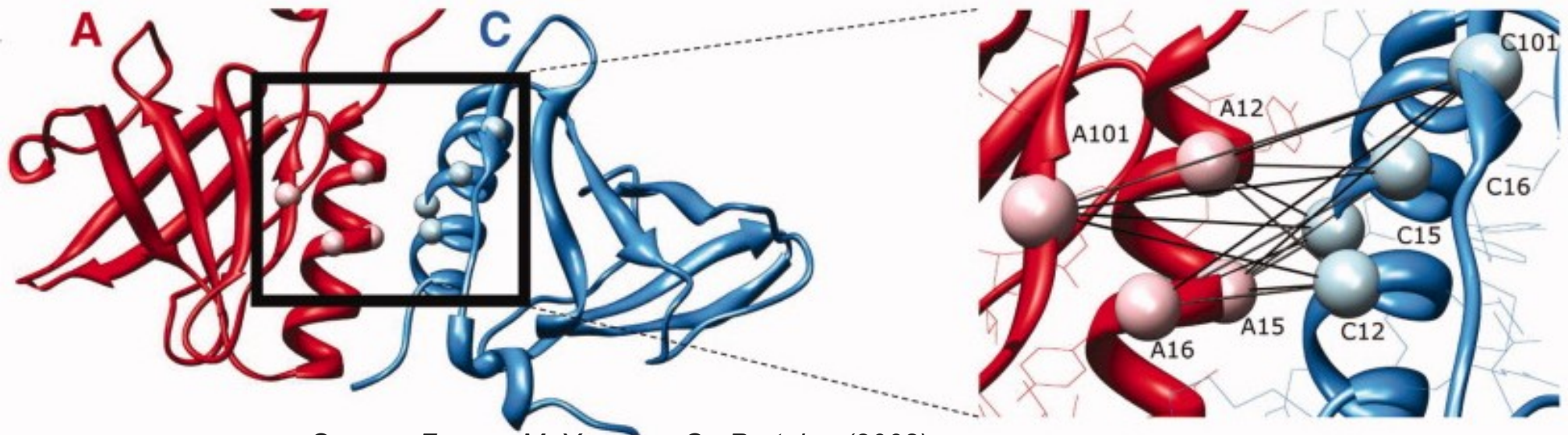
Summary: Inference problem

- We model our problem as an **inference** problem.
- We can use **probabilistic** methods to solve it.

3. Graphical models for protein design and belief propagation (BP)

1. Model each design position as a random variable

2. Build interaction graph that shows conditional independence between variables

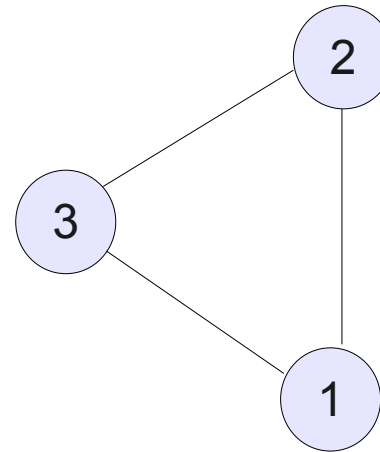
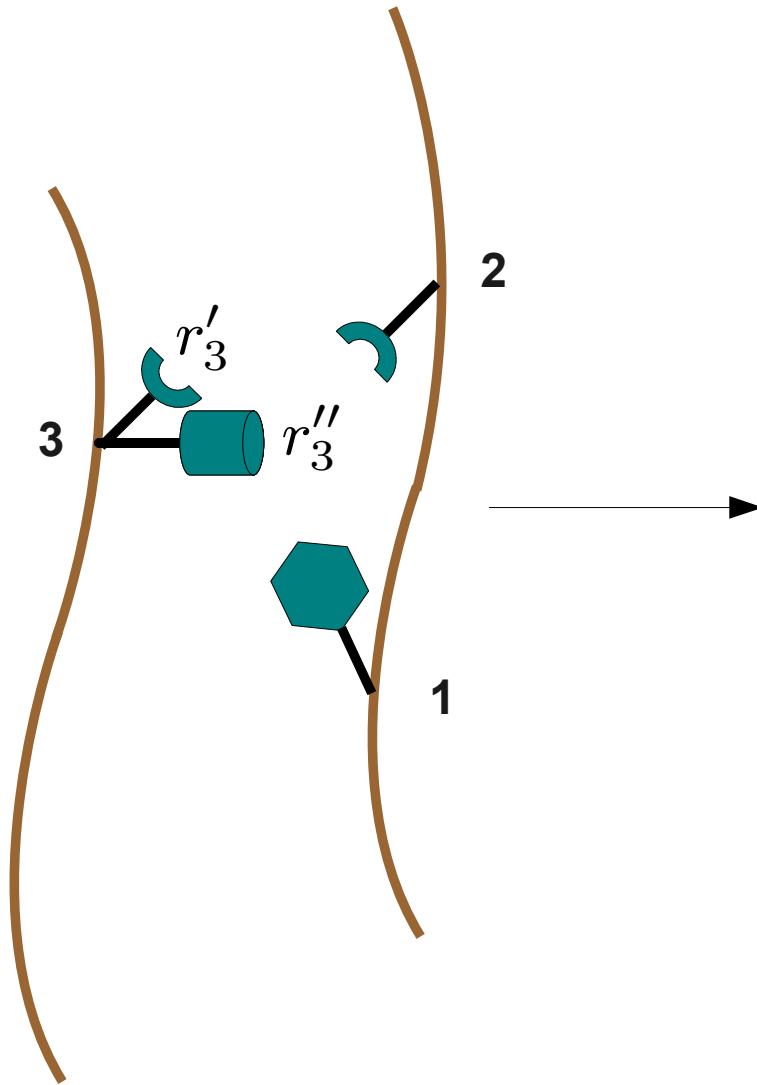


Source: Fromer M, Yanover, C. *Proteins* (2008)

SspB dimer interface: Inter-monomeric interactions (Cα)

3. Graphical Models/BP (2)

Example: Belief propagation

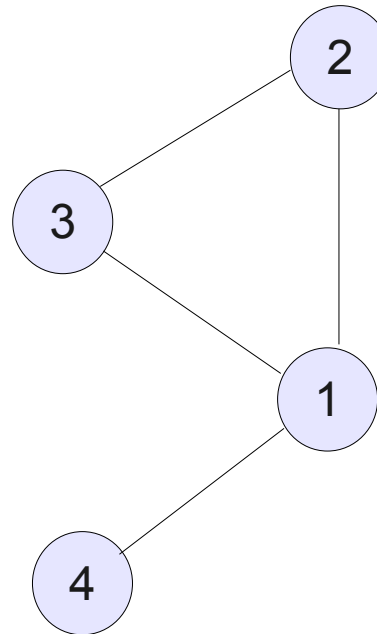
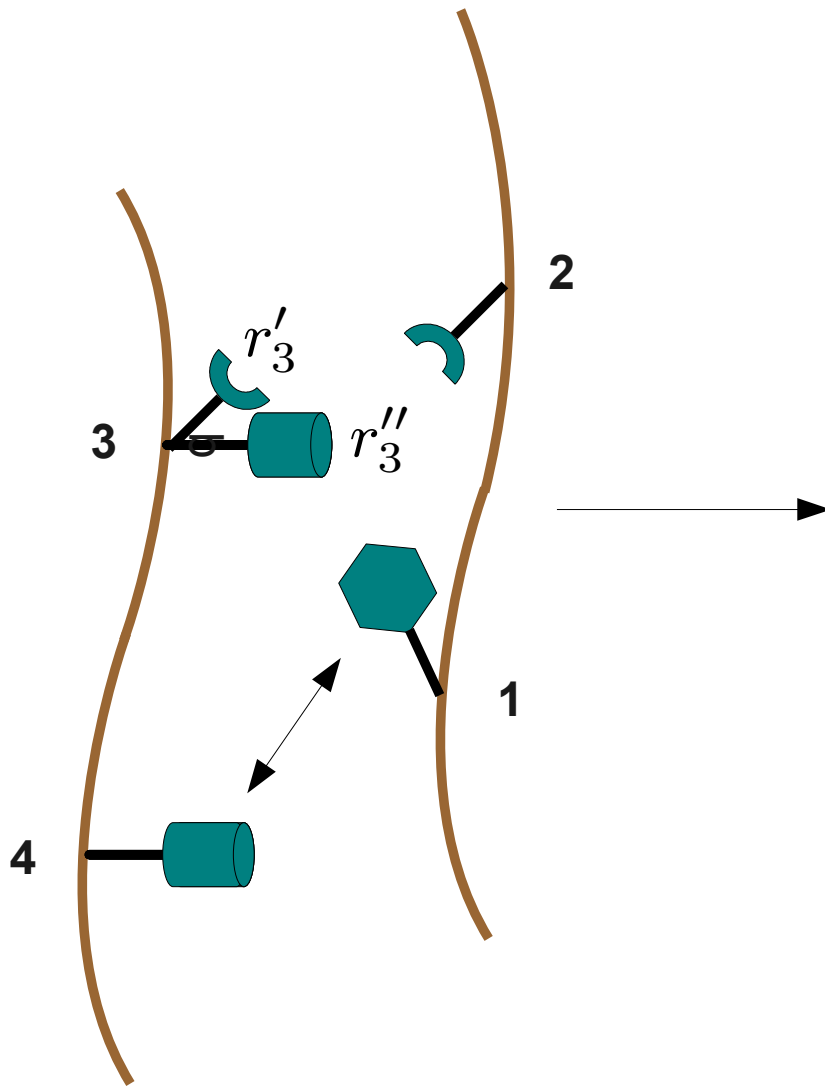


node in the graphical model: **interacting residue** in the structure.

node in the graphical model: **random variable**

3. Graphical Models/BP (3)

Example: Belief propagation



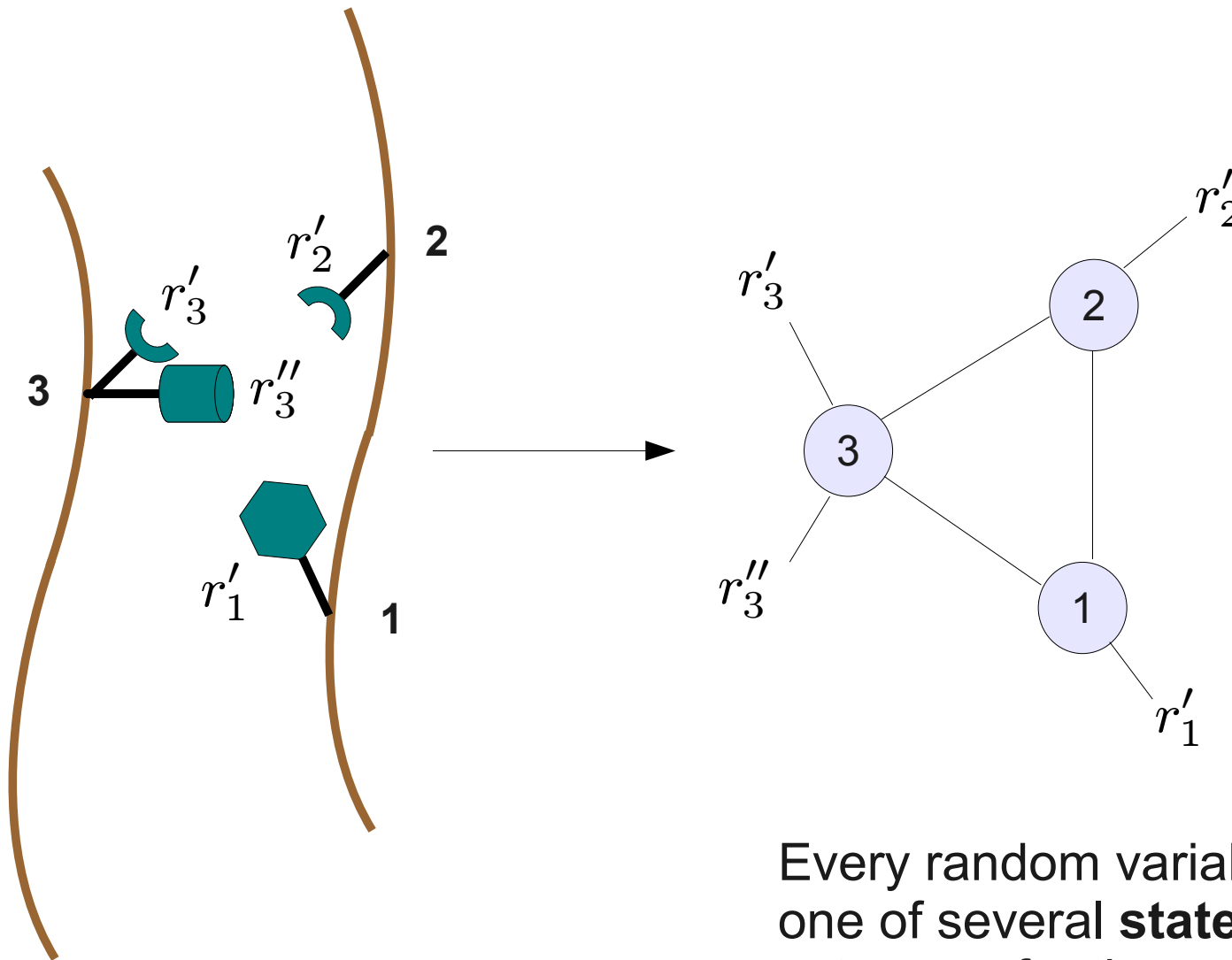
edge: energy
interaction between
two residues.

edge: causal
relationship between
two nodes

If two residues are
distant from each
other, **no edge**
between them.

3. Graphical Models/BP (4)

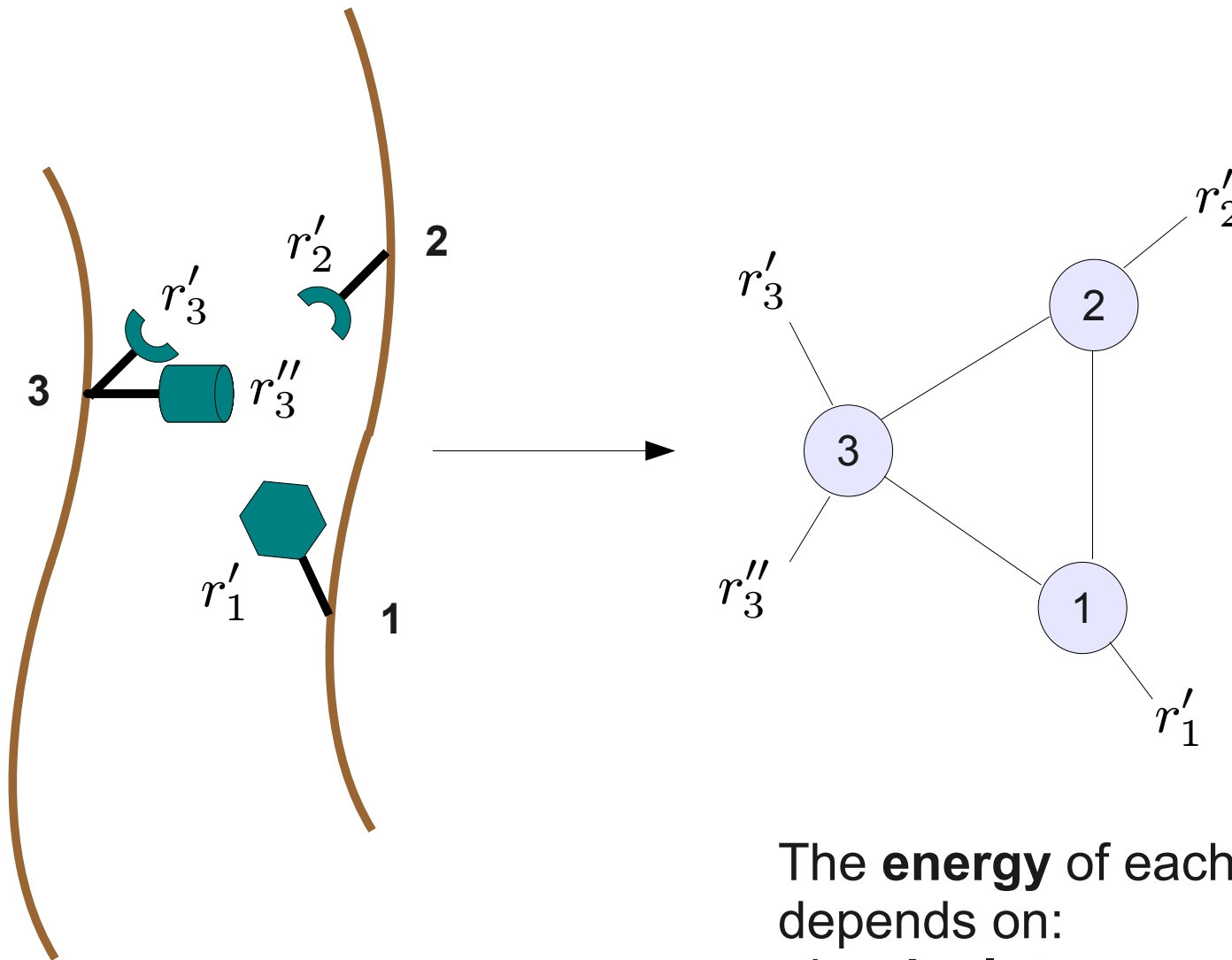
Example: Belief propagation



Every random variable can be in one of several **states**: allowable **rotamers** for that position

3. Graphical Models/BP (5)

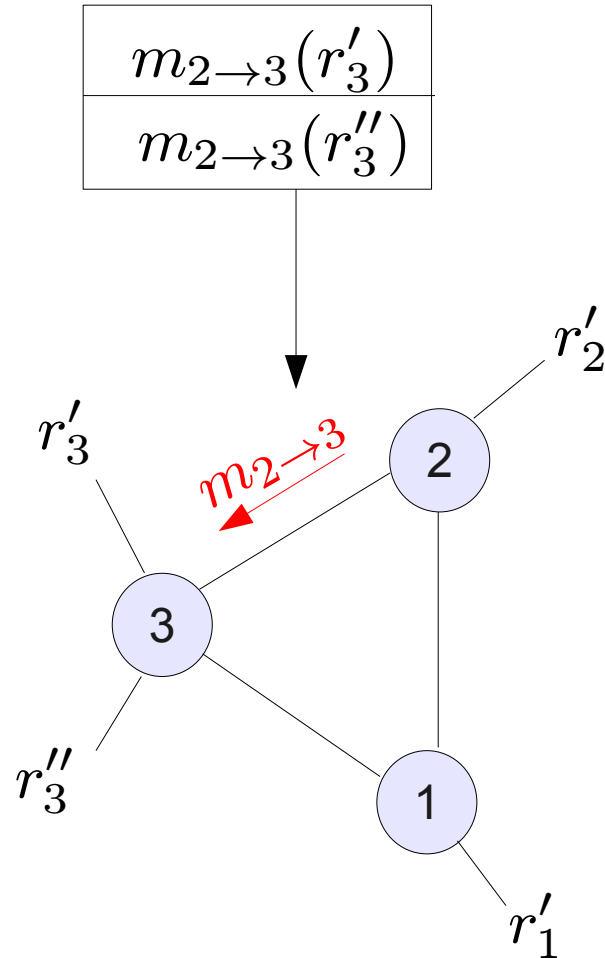
Example: Belief propagation



The **energy** of each state depends on:

- its **singleton** energy
- its **pairwise** energies
- the energies of the states of its **parents**

3. Graphical Models/BP (6)



Example: Belief propagation

Belief propagation: each node tells its **neighbors** nodes what it **believes** their **state** should be

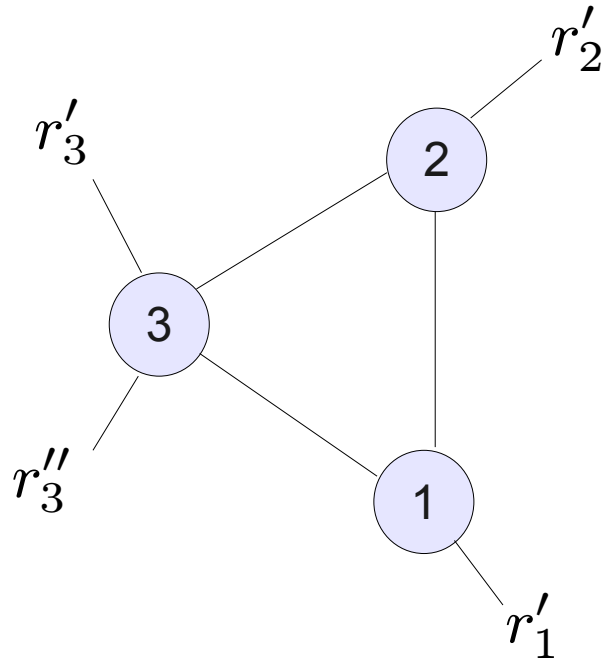
A **message** is sent from node **i** to node **j**

The message is a vector where
of dimensions: allowed
states/rotamers in **recipient**

3. Graphical Models/BP (7)

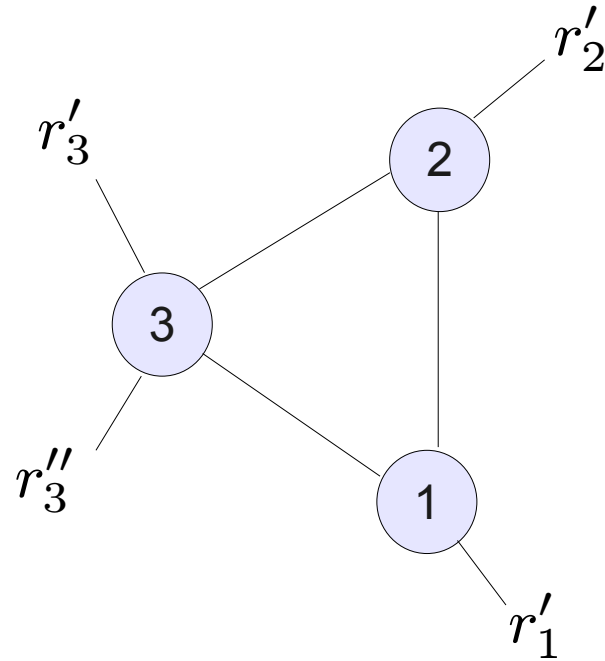
Example: Belief propagation

Who sends the first message?



3. Graphical Models/BP (8)

Example: Belief propagation



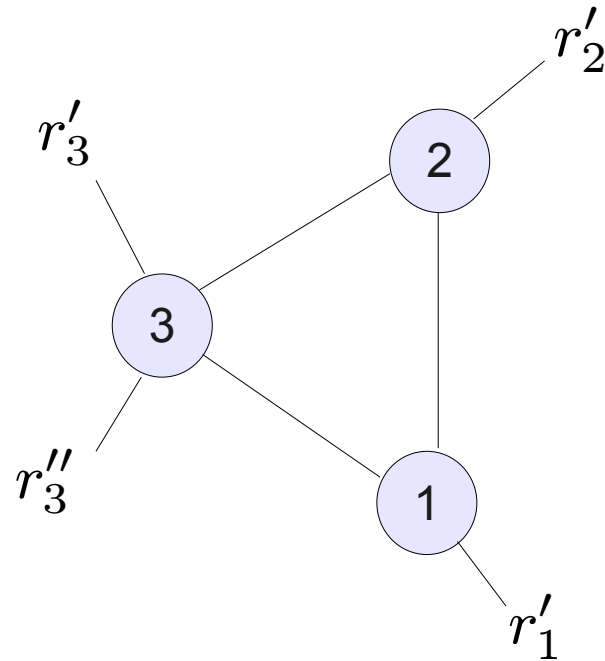
Who sends the first message?

In a **tree**: the **leaves**

- Belief propagation is proven to be correct in a tree!

3. Graphical Models/BP (9)

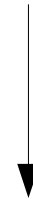
Example: Belief propagation



Who sends the first message?

In a graph with cycles:

- Set initial values
- Send in parallel

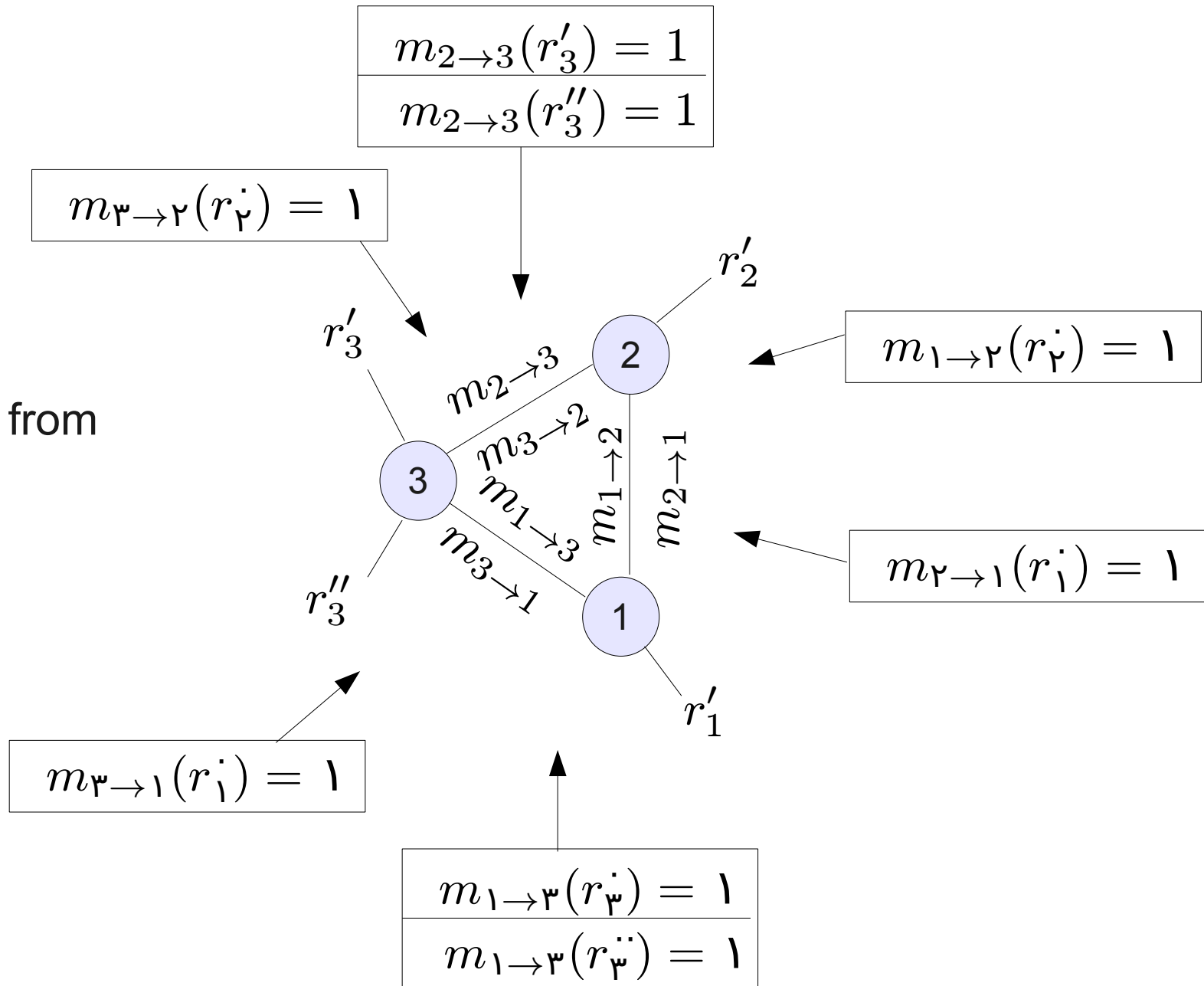


No guarantees can be made! There might not be any convergence

3. Graphical Models/BP (10)

Example: Belief propagation

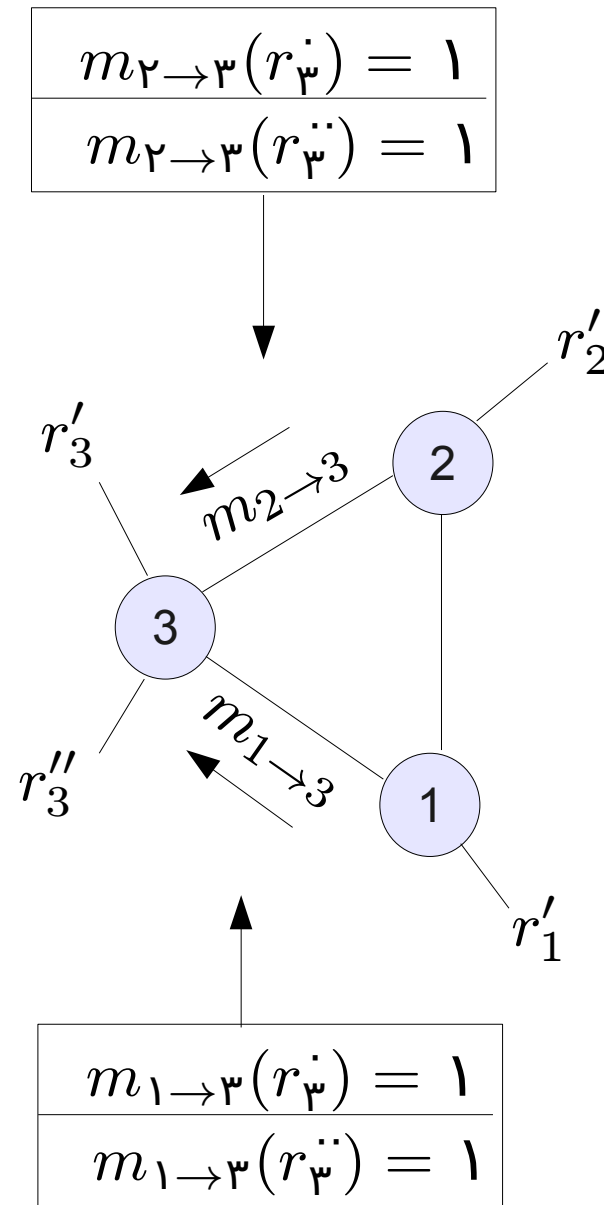
We **iterate** from there.



3. Graphical Models/BP (11)

Example: Belief propagation

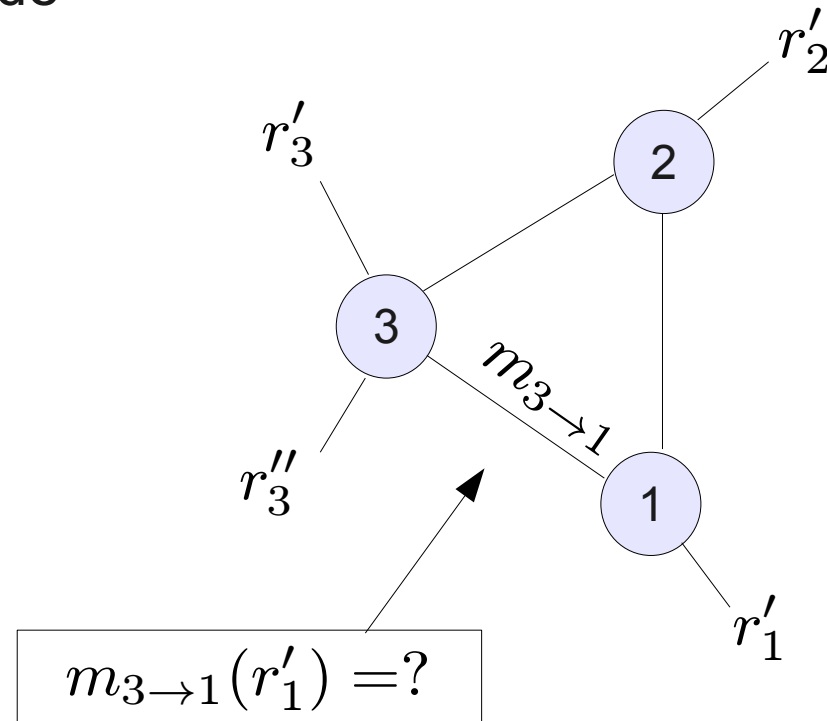
Node 3 receives
messages from nodes 1
and 2



3. Graphical Models/BP (12)

Example: Belief propagation

What message does node 3 send to node 1 on the next iteration?



Belief propagation: message passing

$N(i) \rightarrow$ Neighbors of variable i

Message that gets sent on each iteration

$$m_{i \rightarrow j}(r_j) = \max_{r_i} \left(e^{\frac{-E_i(r_i) - E_{ij}(r_i, r_j)}{t}} \prod_{k \in N(i) \setminus j} m_{k \rightarrow i}(r_i) \right)$$


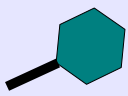
Example: Belief propagation

Pairwise energies

$$E_{ij}(r_1, r_2)$$

Position #1




Position #2

		
	-4	

$$E_{ij}(r_2, r_3)$$

Position #3



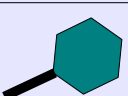
Position #2

		
	-1 r'_3	-3 r''_3

$$E_{ij}(r_1, r_3)$$

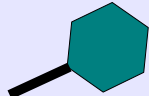
Position #3

Position #1


		
	-1 r'_3	-4 r''_3

Singleton energies



$$E_i(r_1)$$

	-1
---	----

$$E_i(r_2)$$

	-2
---	----

$$E_i(r_3)$$

	-6 r'_3
	-2 r''_3

Iteration 0:

$$m_{3 \rightarrow 1}(r'_1) = \max_{r_3} \left(e^{\frac{-E_i(r_3) - E_{ij}(r'_3, r_1)}{t}} m_{2 \rightarrow 3}(r'_3), \right. \\ \left. e^{\frac{-E_i(r_3) - E_{ij}(r''_3, r_1)}{t}} m_{2 \rightarrow 3}(r''_3) \right)$$

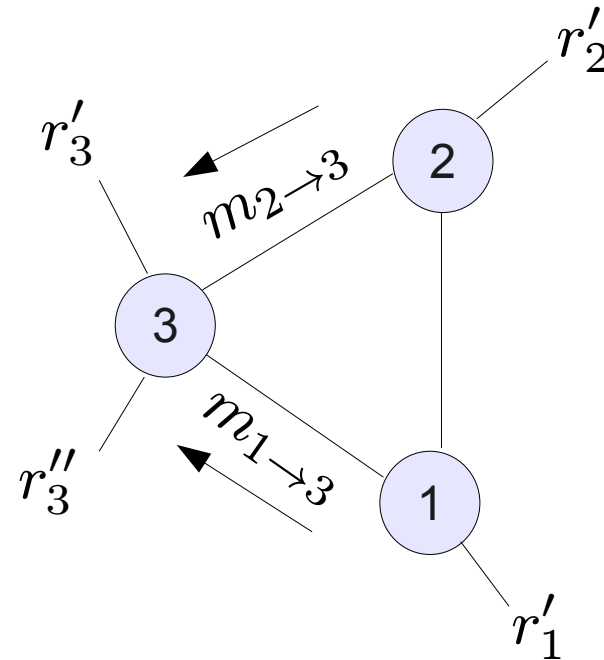
=?

3. Graphical Models/BP (15)

Example: Belief propagation

Once it converges we can compute the belief each node has about **itself**

Belief about one's state:
Multiply all incoming
messages by
singleton energy



Belief propagation: Max-marginals

Belief about each rotamer

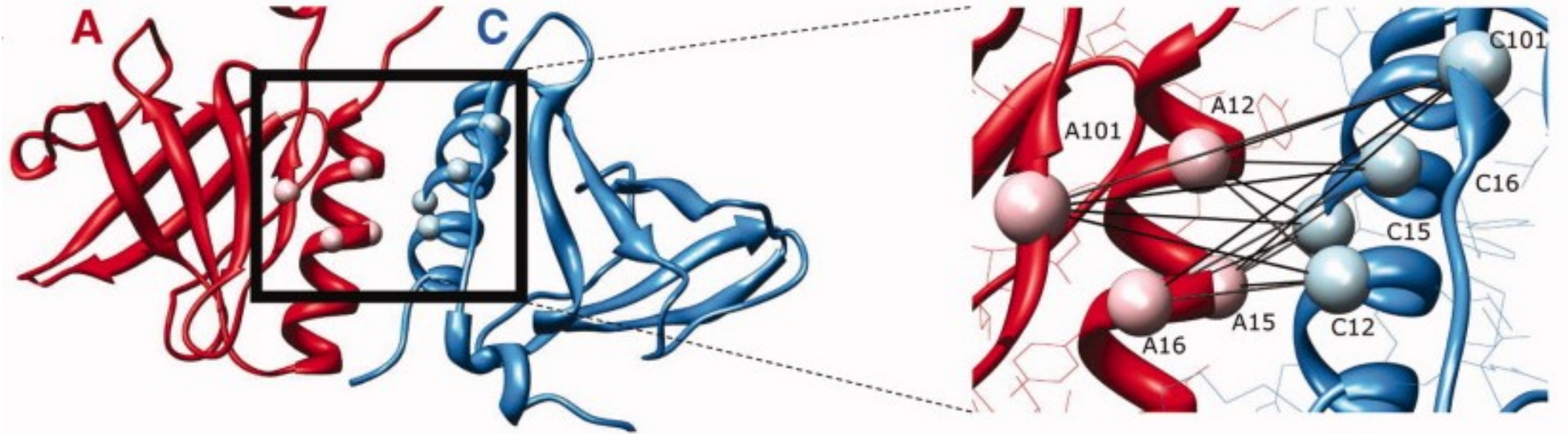
$$MM_i(r_i) = e^{\frac{-E_i(r_i)}{t}} \prod_{k \in N(i)} m_{k \rightarrow i}(r_i)$$

$$\Pr_i^\infty(r_i) = \max_{r': r'_i = r_i} \Pr(r')$$

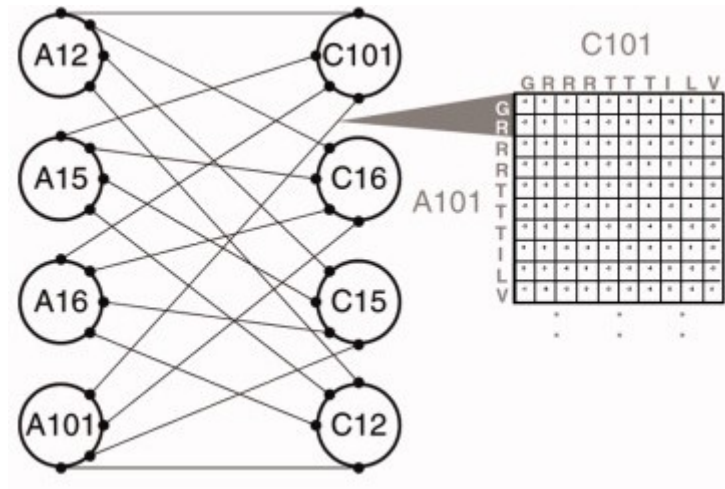
“Most likely” rotamer for position i

$$r_i^* = \arg \max_{r_i \in \text{Rots}_i} \Pr_i^\infty(r_i)$$

3. Graphical Models/BP (17)

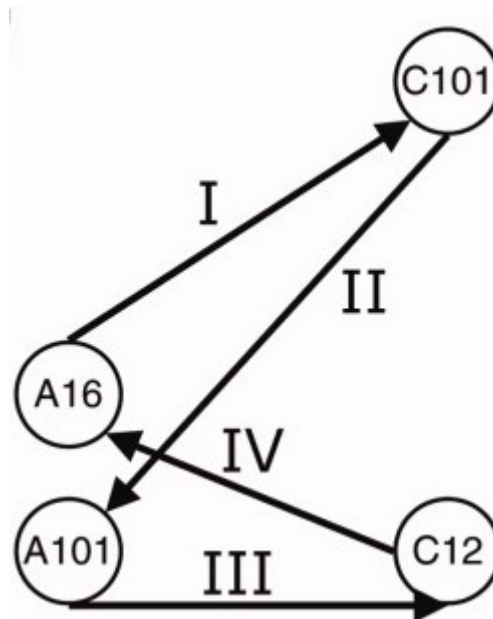


Fromer M, Yanover, C. Proteins (2008)

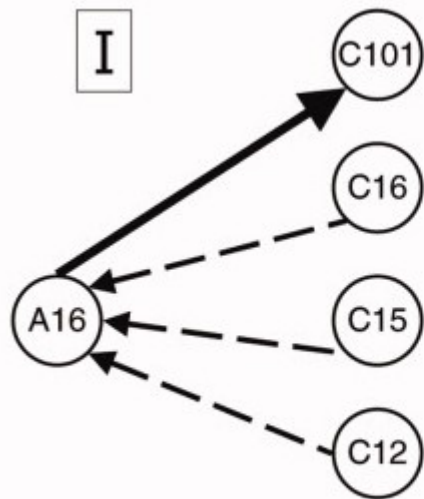


Fromer M, Yanover, C. Proteins (2008)

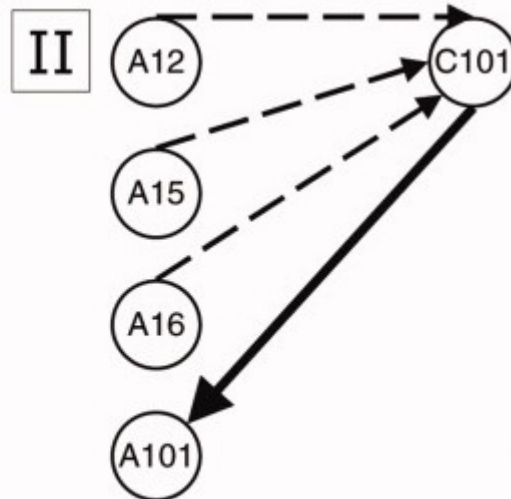
3. Graphical Models/BP (18)



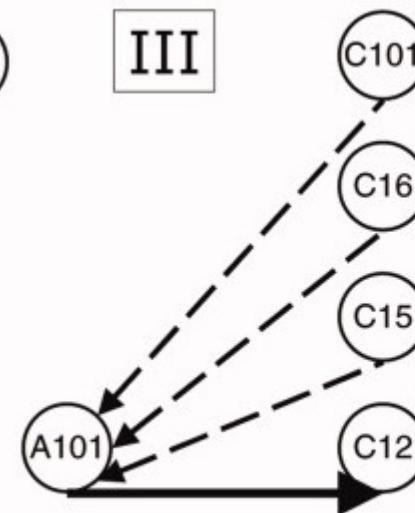
Fromer M, Yanover, C. *Proteins* (2008)



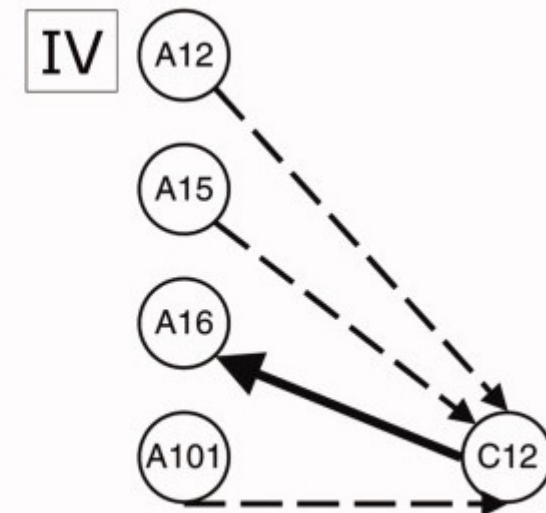
$$m_{A16 \rightarrow C101}(r_{C101})$$



$$m_{C101 \rightarrow A101}(r_{A101})$$



$$m_{A101 \rightarrow C12}(r_{C12})$$



$$m_{C12 \rightarrow A16}(r_{A16})$$

Fromer M, Yanover, C. *Proteins* (2008)

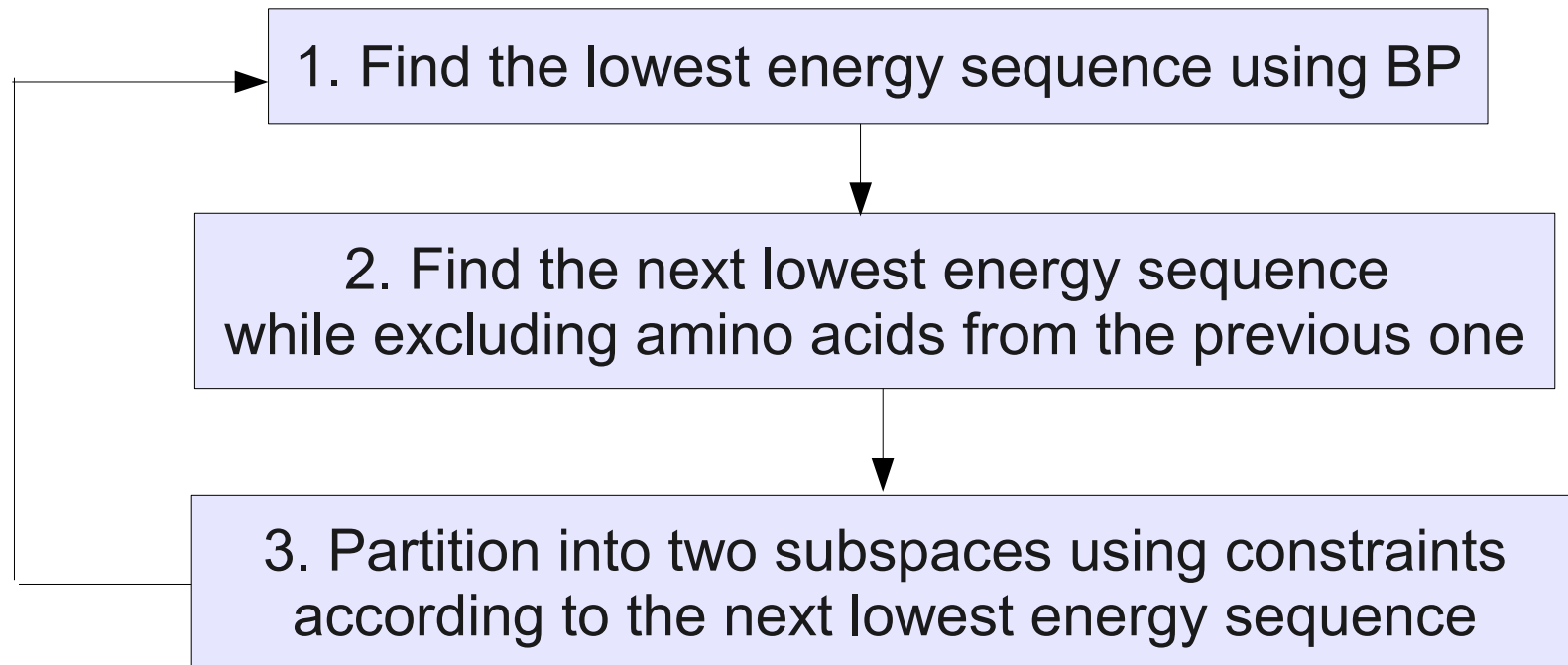
3. Graphical Models: Summary

- Formulate as an inference problem
- Model our design problem as a graphical model
- Establish edges between interacting residues
- Use Belief Propagation to find the beliefs for each position

4. tBMMF: type specific BMMF

- Paper's main contribution
- Builds on previous work by C. Yanover (2004)
- Uses **Belief propagation** to find lowest energy sequence and **constrains** space to find subsequent sequences

TBMMF (simplification)



4. tBMMF (3)

Example: tBMMF (1)

A

Position #2			Position #1			
	aa		G_1		G_2	
	aa	rot.	g_{11}	g_{12}	g_{21}	g_{22}
	H_1					
		h_{11}	-15	-11	-6	-3
		h_{12}	-14	-10	-7	-2
	H_2	h_{21}	-8	-9	0	-5
		h_{22}	-12	-13	-4	-1

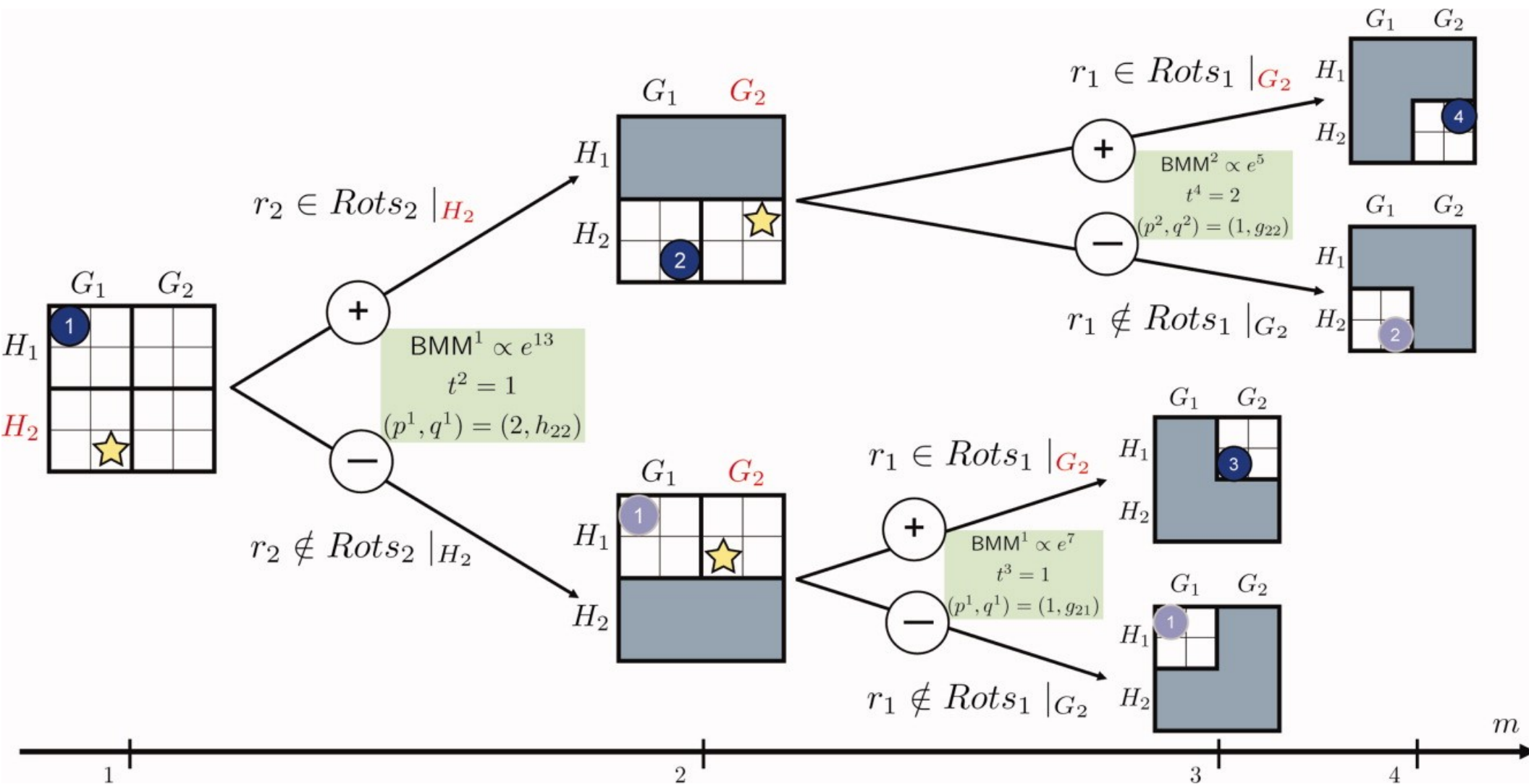
B

r	$E(r)$	$T(r)$
(g_{11}, h_{11})	-15	(G_1, H_1)
(g_{11}, h_{12})	-14	(G_1, H_1)
(g_{12}, h_{22})	-13	(G_1, H_2)
(g_{11}, h_{22})	-12	(G_1, H_2)
(g_{12}, h_{11})	-11	(G_1, H_1)
(g_{12}, h_{12})	-10	(G_1, H_1)
(g_{12}, h_{21})	-9	(G_1, H_2)
(g_{11}, h_{21})	-8	(G_1, H_2)
(g_{21}, h_{12})	-7	(G_2, H_1)
(g_{21}, h_{11})	-6	(G_2, H_1)
(g_{22}, h_{21})	-5	(G_2, H_2)
(g_{21}, h_{22})	-4	(G_2, H_2)
(g_{22}, h_{11})	-3	(G_2, H_1)
(g_{22}, h_{12})	-2	(G_2, H_1)
(g_{22}, h_{22})	-1	(G_2, H_2)
(g_{21}, h_{21})	0	(G_2, H_2)

Fromer M, Yanover, C. *Proteins* (2008)

4. tBMMF (4)

Example: tBMMF (2)



Fromer M, Yanover, C. Proteins (2008)

Results

prion



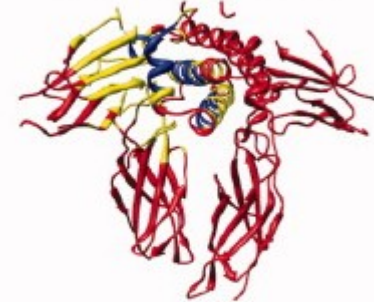
1I4M
(a+b)

SspB



1OU9
all beta

hGH-hGHR



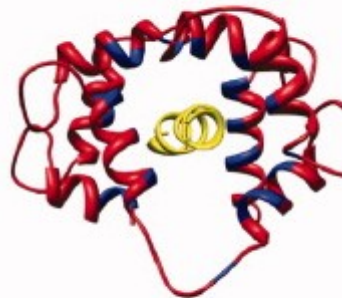
3HHR
all alpha

CaM-smMLCK



1CDL
all alpha

CaM-skMLCK



2BBM
all alpha

Top7



1QYS
(a+b)

Results (2)

		Num. Positions (Chains ^a)				Search Space Cardinality (log ₁₀)			Rotamer Library	
		Design		Shell ^b		Sequence	Rotamer	td-DEE ^c	Read ^d	Added ^e
SMALL	prion	7	(A)	7	(B)	8.95	31.88	26.60	Full	χ_1, χ_2
	SspB	8	(A,C)	0		10.23	24.82	24.07	Full	χ_1, χ_2
MEDIUM	hGH-hGHR 1	6	(A) →	135	(A,B)	7.67	238.95	168.01	Full	χ_1, χ_2
	hGH-hGHR 2	6	(A)	135	(A,B)	7.67	237.63	164.78	Full	χ_1, χ_2
	hGH-hGHR 3	5	(A)	136	(A,B)	6.39	235.01	163.43	Full	χ_1, χ_2
	hGH-hGHR 4	6	(A)	135	(A,B)	7.67	237.68	167.38	Full	χ_1, χ_2
	hGH-hGHR 5	6	(A)	135	(A,B)	7.67	236.49	159.71	Full	χ_1, χ_2
	hGH-hGHR 6	6	(A)	135	(A,B)	7.67	237.25	163.53	Full	χ_1, χ_2
LARGE 1	CaM-smMLCK	24	(A)	19	(B)	30.69	86.07	82.18	Limited	χ_1
	CaM-skMLCK	24	(A)	19	(B)	30.69	80.63	73.75	Limited	χ_1
LARGE 2	hGH-hGHR	35	(A)	106	(A,B)	44.76	213.89	167.32	Limited	χ_1
	Top7	92	(A)	0		117.65	202.20	200.19	Limited	—

^aPeptide chains to which the corresponding positions belong, labeled arbitrarily.

^bNon-designed, conformationally varying positions.

^cRotamer space cardinality after application of type-dependent Goldstein DEE.

^dFull: all rotamers read from library; Limited: highest probability rotamers read.

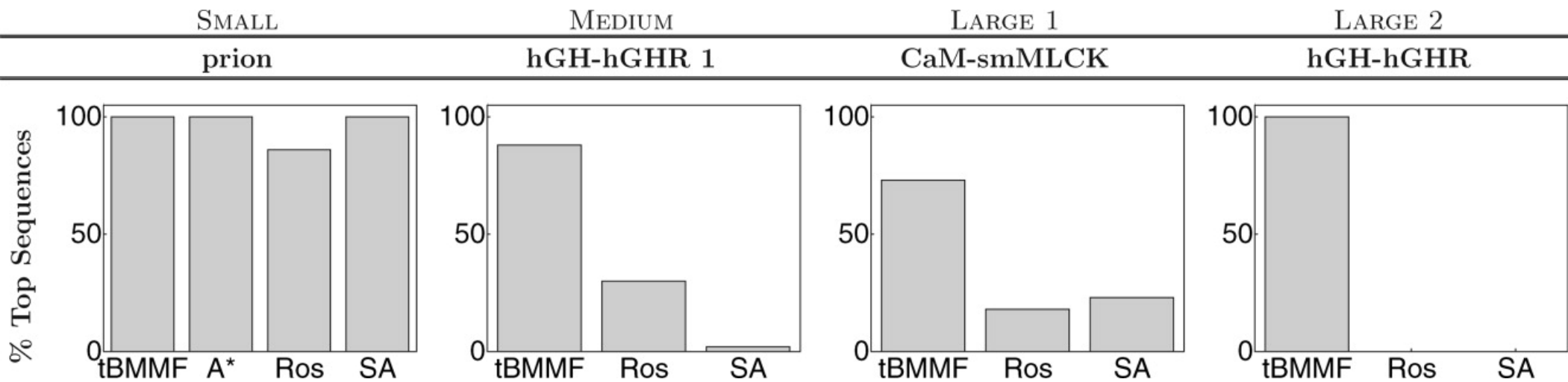
^eSide-chain angles around which additional rotamers were super-sampled from library rotamers.

Fromer M, Yanover, C. Proteins (2008)

Results(3)

- Algorithms tried:
 - DEE / A* (Goldstein, 1-split, 2-split, Magic Bullet)
 - tBMMF
 - Ros: Rosetta
 - SA: Simulated annealing over sequence space

Results (4): Assessment results



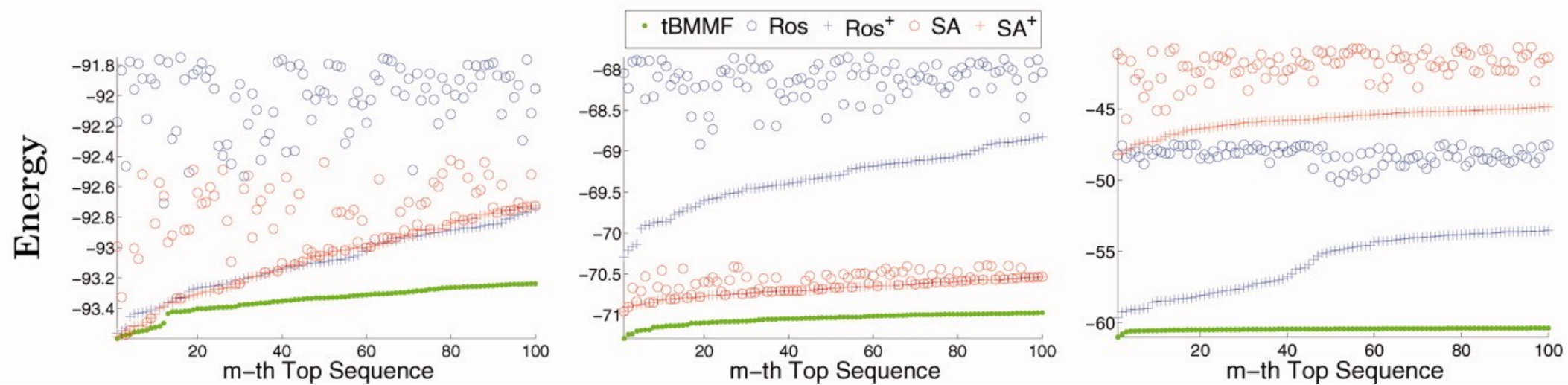
Fromer M, Yanover, C. Proteins (2008)

Results(5)

CaM-smMLCK

CaM-skMLCK

hGH-hGHR



Fromer M, Yanover, C. *Proteins* (2008)

Results(6)

Table I. Assessment and Analysis of the Algorithms Tested

		tBMMF		Ros		SA		A*		(A* Rotamer Space)		
		Top	Time	Top	Time	Top	Time	Top ^a	Time	td-DEE ^b	DEE ^c	
Small	prion	100%	58.9 m	86%	9.3 h	100%	12 h	100%	3.4 m	26.60	→	11.68
	SspB	100%	11 h	100%	11.4 h	97%	9.6 h	1% ^d	3 d	24.07	→	15.91
Medium	hGH-hGHR 1	88%	13.4 h	30%	2.1 d	2%	7.3 d	Failed	12 d	168.01	→	131.81
	hGH-hGHR 2	60%	7.6 h	50%	2 d	0%	5.9 d	Failed	12 d	164.78	→	130.61
	hGH-hGHR 3	100%	4.1 h	73%	1.7 d	0%	5.9 d	Failed	12 d	163.43	→	128.58
	hGH-hGHR 4	100%	8.5 h	22%	2.1 d	0%	7.4 d	Failed	12 d	167.38	→	134.18
	hGH-hGHR 5	100%	2.9 h	27%	2 d	0%	5.8 d	Failed	12 d	159.71	→	120.98
	hGH-hGHR 6	100%	8.5 h	42%	2.2 d	0%	6.1 d	Failed	12 d	163.53	→	132.21
Large 1	CaM-smMLCK	73%	10.6 h	18%	18 h	23%	1 d	Failed	12 d	82.18	→	45.97
	CaM-skMLCK	100%	2 h	0%	10.7 h	0%	20.7 h	Failed	7.2 d	73.75	→	47.41
Large 2	hGH-hGHR	100%	17.6 h	0%	2 d	0%	2.3 d	Failed	12 d	167.32	→	115.63
	Top7	100%	7.1 h	33%	1.7 d	0%	1.7 d	Failed	12 d	200.19	→	168.38

Fromer, M., Yanover, C. Proteins (2008)

Results(6)

Table I. Assessment and Analysis of the Algorithms Tested

		tBMMF		Ros		SA		A*		(A* Rotamer Space)	
		Top	Time	Top	Time	Top	Time	Top ^a	Time	td-DEE ^b	DEE ^c
Small	prion	100%	58.9 m	86%	9.3 h	100%	12 h	100%	3.4 m	26.60	→ 11.68
	SspB	100%	11 h	100%	11.4 h	97%	9.6 h	1%	3 d	24.07	→ 15.91
Medium	hGH-hGHR 1	88%	13.4 h	30%	2.1 d	2%	7.3 d	Failed	12 d	168.01	→ 131.81
	hGH-hGHR 2	60%	7.6 h	50%	2 d	0%	5.9 d	Failed	12 d	164.78	→ 130.61
	hGH-hGHR 3	100%	4.1 h	73%	1.7 d	0%	5.9 d	Failed	12 d	163.43	→ 128.58
	hGH-hGHR 4	100%	8.5 h	22%	2.1 d	0%	7.4 d	Failed	12 d	167.38	→ 134.18
	hGH-hGHR 5	100%	2.9 h	27%	2 d	0%	5.8 d	Failed	12 d	159.71	→ 120.98
	hGH-hGHR 6	100%	8.5 h	42%	2.2 d	0%	6.1 d	Failed	12 d	163.53	→ 132.21
Large 1	CaM-smMLCK	73%	10.6 h	18%	18 h	23%	1 d	Failed	12 d	82.18	→ 45.97
	CaM-skMLCK	100%	2 h	0%	10.7 h	0%	20.7 h	Failed	7.2 d	73.75	→ 47.41
Large 2	hGH-hGHR	100%	17.6 h	0%	2 d	0%	2.3 d	Failed	12 d	167.32	→ 115.63
	Top7	69%	7.1 h	31%	1.5 d	0%	1.7 d	Failed	12 d	200.19	→ 168.38

Fromer M, Yanover, C. *Proteins* (2008)

Results (7)

- DEE/A* was **not feasible** for any case except the prion
- SspB: A* could only output one sequence
- DEE also did not finish after 12 days
- BD/K* did not finish after 12 days

Results (8)

- Predicted sequences where **highly similar** between themselves. (high sequence identity)
- Very **different** from **wild type** sequence
- Solution: *grouped tBMMF*: apply constraints to whole groups of amino acids – proof of concept only

Conclusions

- Fast and accurate algorithm
- Outperforms all other algorithms:
 - A^* is not feasible
 - Better accuracy than other probabilistic algorithms

Conclusions (2)

- tBMMF produces a large set of very similar low energy results.
- This might be due to the many inaccuracies in the model
- Grouped tBMMF can produce a diverse set of low energy sequences

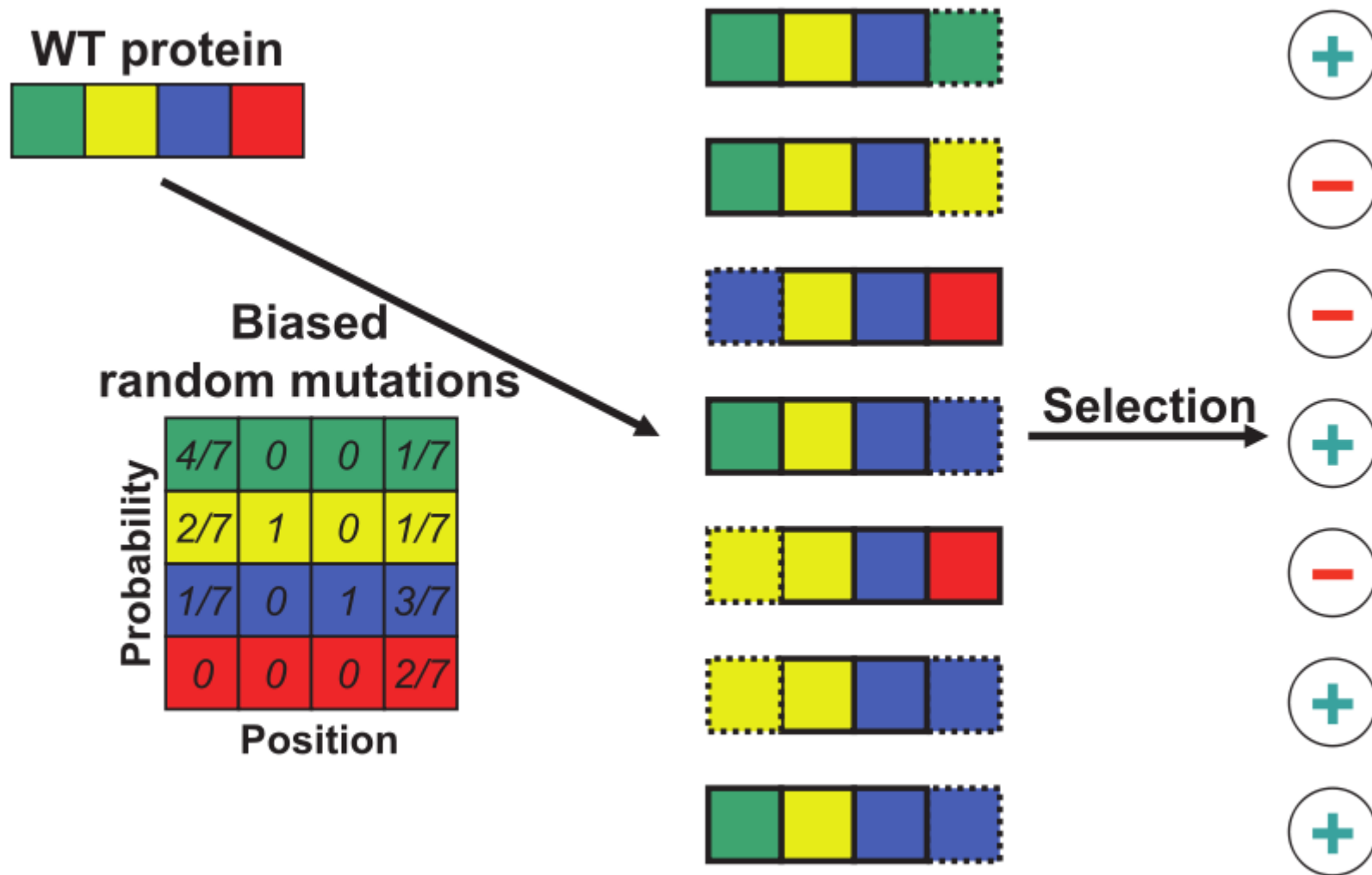
Conclusions (3)

- The results lack experimental data for validation.

Related Work: (Fromer et al. 2008)

- Fromer F, Yanover C. *A computational framework to empower probabilistic protein design*. ISMB 2008
- **Phage display:**
 - $10^9 - 10^{10}$ randomized protein sequences
 - Simultaneously tested for relevant biological function

Related Work: (Fromer et al. 2008)



Related Work: (Fromer et al. 2008)

- Uses sum-product instead of max-product
- Obtain **per-position** amino acid **probabilities**
- Tried until convergence or 100000 iterations; all structures converged

Related Work: (Fromer et al. 2008)

- Conclusions:
 - Model results in probability distributions far from those observed experimentally.
 - Limitations of the model:
 - Imprecise **energy function**
 - Decomposition into **pairwise energy** terms
 - Assumption of a **fixed backbone**
 - **Discretization** of side chain conformations

tBMMF algorithm

```

1 for  $m \leftarrow 1$  to  $M$  do
2   if  $m = 1$  then
3      $Cons^m \leftarrow \emptyset$ 
4   else
5     /*  $t^m, p^{t^m}, q^{t^m}$  are the sub-space,
6        position, rotamer to yield the
7        next lowest energy sequence */
8      $t^m \leftarrow \arg \max_{m' < m} BMM^{m'}$ 
9      $a \leftarrow T(q^{t^m})$  // aa type of  $q^{t^m}$ 
10    // Add pos. constraint to  $Cons^m$ :
11     $Cons^m \leftarrow Cons^{t^m} \cup \{r_{p^{t^m}} \in Rots_{p^{t^m}} | a\}$ 
12    // Add neg. constraint to  $Cons^{t^m}$ :
13     $Cons^{t^m} \leftarrow Cons^{t^m} \cup \{r_{p^{t^m}} \notin Rots_{p^{t^m}} | a\}$ 
14    Run BP to obtain:  $MM_p(q) |_{Cons^{t^m}}$ 
15    CalcBMM( $t^m$ ) // calculate  $BMM^{t^m}$ 
16  end
17  Run BP to obtain:  $MM_p(q) |_{Cons^m}$ 
18  for  $i \leftarrow 1$  to  $N$  do
19     $r_i^m \leftarrow \arg \max_{r_i \in Rots_i} MM_i(r_i) |_{Cons^m}$ 
20     $S_i^m \leftarrow T(r_i^m)$  //  $i^{th}$  aa of  $m^{th}$  seq.
21  end
22  CalcBMM( $m$ ) // calculate  $BMM^m$ 
23 end
24 return  $\{S^m\}_{m=1}^M$ 

```

```

/* Use  $MM_p(q) |_{Cons^n}$  to calculate the BMM
   for constrained sub-space  $n$  */
20 Function CalcBMM( $n$ )
21    $(p^n, q^n) \leftarrow \arg \max_{p, q: T(q) \neq S_p^n} MM_p(q) |_{Cons^n}$ 
22    $BMM^n \leftarrow MM_{p^n}(q^n) |_{Cons^n}$ 
23 end

```