



Computational Protein Design: un outil pour l'ingénierie des protéines et la biologie synthétique

David Mignon

sous la direction de Thomas Simonson
Laboratoire de Biochimie, **École Polytechnique**

Computational Protein Design (CPD)

Concevoir ou modifier des protéines par informatique pour leur conférer de nouvelles propriétés

Application:

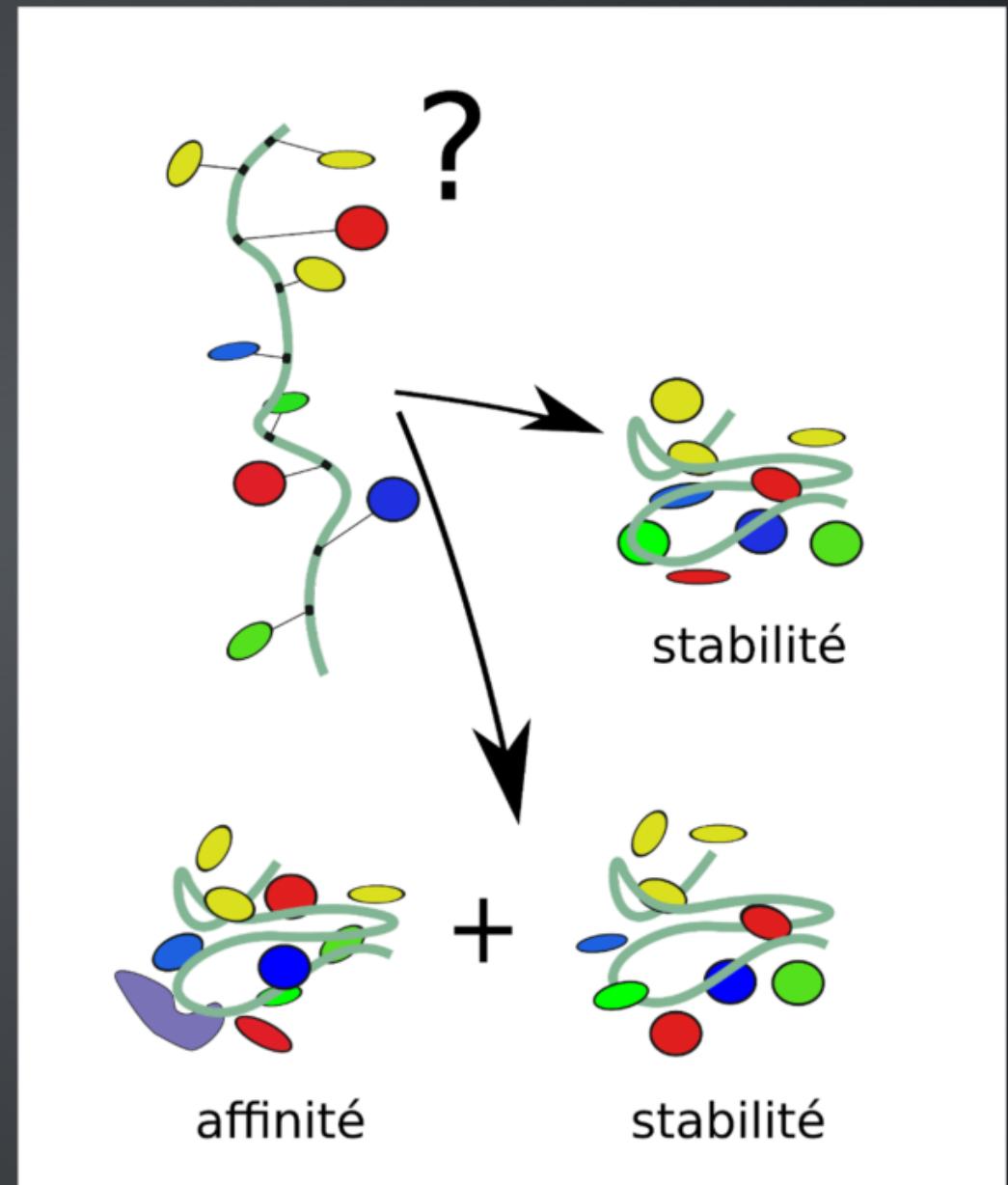
- protéines redessinées
- enzymes, complexes
- insertion d'acides aminés non-naturels

Principaux éléments:

- un espace de conformations de la protéine
- une fonction d'énergie
- un algorithme d'exploration de l'espace de séquences-conformations

Principaux programmes:

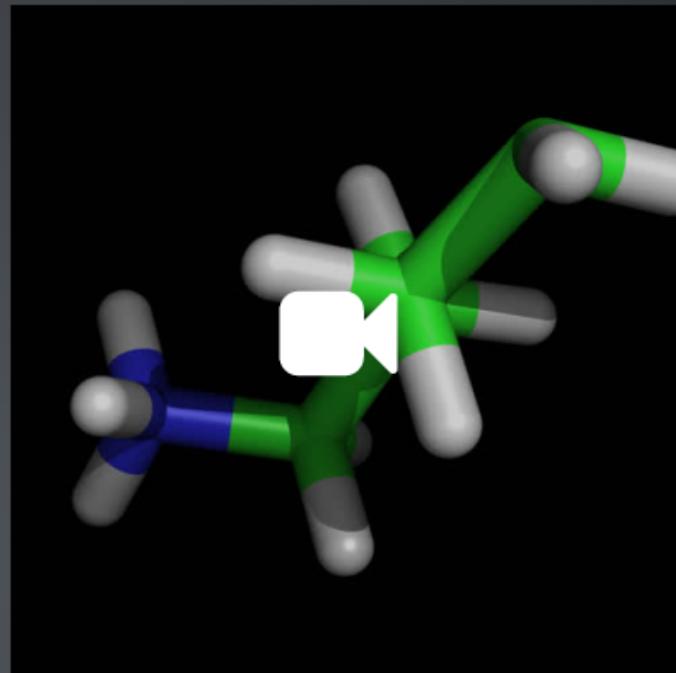
- ORBIT (Mayo,1996)
- Toulbar2 (Allouche,2014)
- Proteus (Simonson,2008)
- Rosetta (Baker,2003)



Le CPD avec Proteus

L'espace de conformation:

- Un backbone fixe
Nous utilisons le squelette d'une protéine native
- positionnent des chaînes latérales discrétisées (rotamères)
Utilisation d'un bibliothèque de rotamères:Tuffery95
- Les prolines et les glycines natives sont conservées



L'état déplié: l'énergie de référence
Pour une séquence S de type t_i

$$E^u(S) = \sum_i^N E_{t_i}^u$$

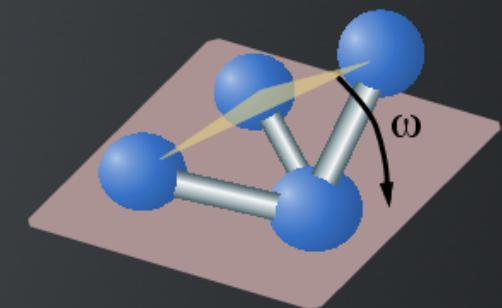
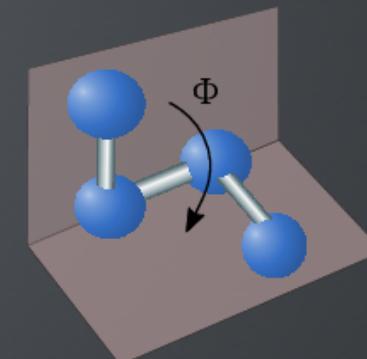
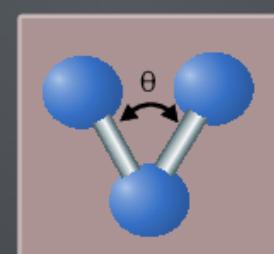
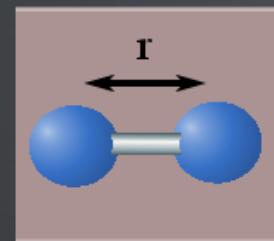
La fonction d'énergie

L'énergie interne à la protéine:

Utilisation de la mécanique moléculaires avec le champ de force Amber

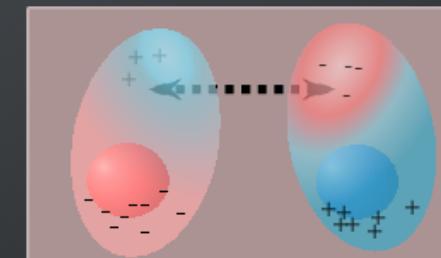
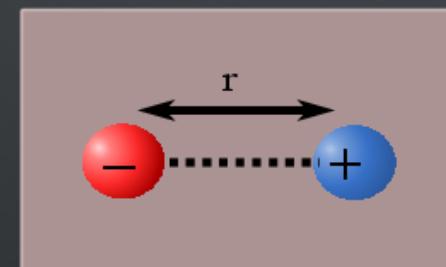
- Les interactions liées:

$$E_{liée} = E_{liaison} + E_{angle} + E_{diedre} + E_{impr}$$



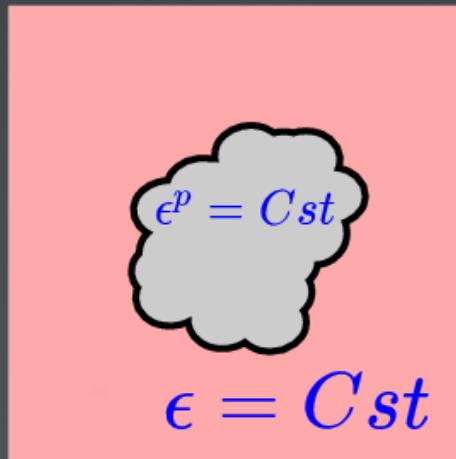
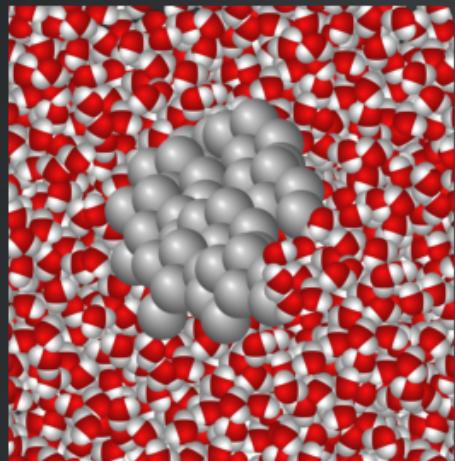
- Les interactions non liées

$$E_{non \ liées} = E_{elec} + E_{vdw}$$



La fonction d'énergie

Les modèles de solvant implicites



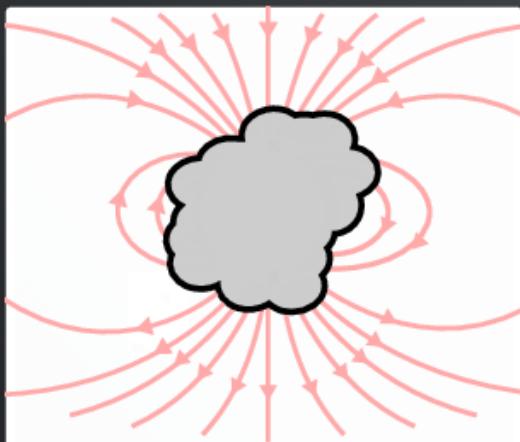
1. l'effet hydrophobe:
modèle Surface Area (SA)

$$E_{hydro} \approx \sum_i \sigma_{t_i} A_i$$

2. Traitement des interactions électrostatiques:

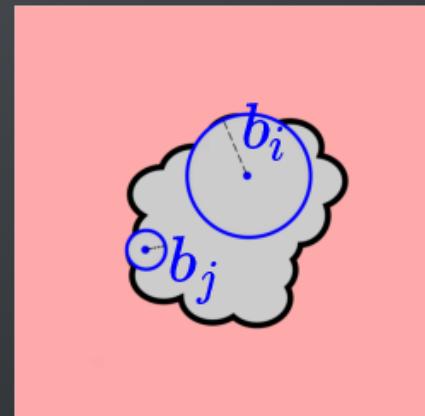
Coulomb

$$E_{solv}^{elec} \approx \left(\frac{1}{\epsilon} - 1\right) E_{coul}^p$$



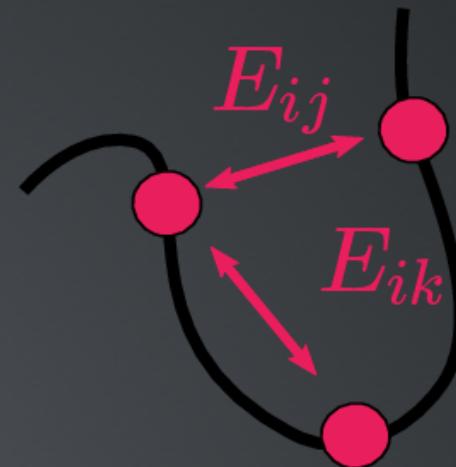
Generalised Born (GB)

$$E_{coul}^p + E_{solv}^{elec} = \frac{1}{2} \sum_{i \neq j} \frac{q_i q_j}{\epsilon_s r_{ij}} - \tau \frac{1}{2} \sum_{i,j}^N \tau q_i q_j (r_{ij}^2 + b_i b_j \exp(-\frac{r_{ij}^2}{4b_i b_j}))^{-1/2}$$



Décomposition par paires de la fonction d'énergie

$$E(C) = \sum_i E_i + \sum_{i \neq j} E_{ij}$$



- rendre possible l'utilisation de plusieurs algorithmes
- accélérer de l'étape d'exploration par pré-calculation des interactions -> stockage dans une matrice.
- nécessite des approximations supplémentaires:
 1. Pour la décomposition du terme surfacique
 2. Pour la décomposition des rayons de solvations (GB)

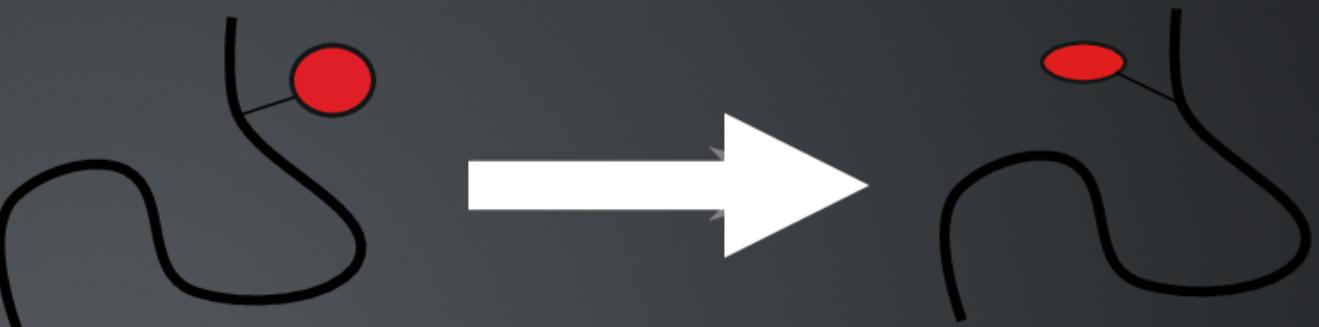
Première méthode: Native Environnement Approximation (NEA)

Les rayons de solvatation d'une chaîne latérale sont calculés en fixant tout le reste du système dans sa séquence et sa conformation native.

Principe de l'exploration

Déplacement dans l'espace des conformations:

modification du rotamère à une position i sur la protéine repliée

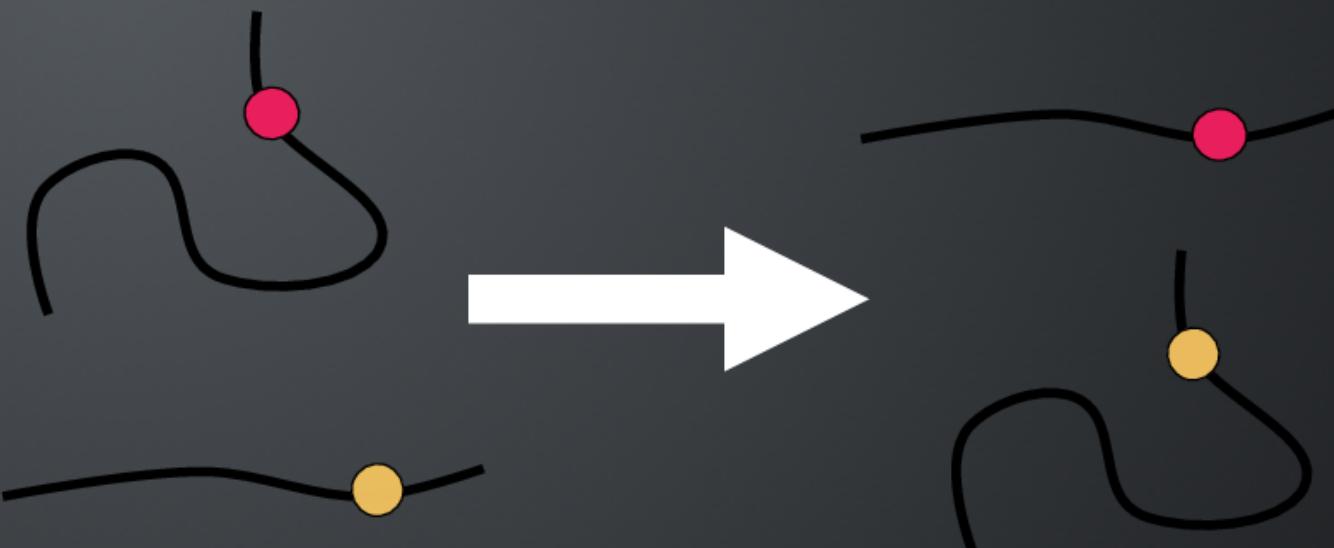


$$\Delta E = E(.., rot_i^{new}, ..) - E(.., rot_i^{old}, ..)$$

Déplacement dans l'espace des séquences:

modification du type de chaîne latérale à une position i sur la protéine repliée.

En même temps, une mutation inverse sur la protéine dépliée, en i



$$\Delta E = \Delta E_f - \Delta E_{uf}$$

Le "Multistart Steepest Descent" (Wernisch,Wodak)

Le but est d'obtenir un ensemble de séquences-rotamères qui approchent le minimum globale.

Spécifique à l'espace d'état

Pour chaque cycle heuristique

une séquence-conformation **S** est choisie aléatoirement

Tant que l'énergie de **S** est améliorée

Pour **i** allant de la première position de **S** jusqu'à la dernière

S est fixée sauf à la position **i**

le meilleur rotamère possible en **i** est déterminé.

Ce rotamère est utilisé pour fixer **S** en **i**

Fin de Pour

Fin de Tant que

S est sauvegardée

Fin du cycle heuristique

Une méthode heuristique spécifique à la structure de l'espace (Wernisch,Wodak)

Le but est d'obtenir un ensemble de séquences-rotamères qui approchent le minimum globale.

Pour chaque cycle heuristique

une séquence-conformation **S** est choisie aléatoirement

Tant que l'énergie de **S** est améliorée

Pour **i** allant de la première position de **S** jusqu'à la dernière

S est fixée sauf à la position **i**

le meilleur rotamère possible en **i** est déterminé.

Ce rotamère est utilisé pour fixer **S** en **i**

Fin de Pour

Fin de Tant que

S est sauvegardée

Fin du cycle heuristique



Une méthode heuristique spécifique à la structure de l'espace (Wernisch,Wodak)

Le but est d'obtenir un ensemble de séquences-rotamères qui approchent le minimum globale.

Pour chaque cycle heuristique
une séquence-conformation **S** est choisie aléatoirement

Tant que l'énergie de **S** est améliorée

Pour **i** allant de la première position de **S** jusqu'à la dernière
S est fixée sauf à la position **i**

le meilleur rotamère possible en **i** est déterminé.

Ce rotamère est utilisé pour fixer **S** en **i**

Fin de Pour

Fin de Tant que

S est sauvegardée

Fin du cycle heuristique



Une méthode heuristique spécifique à la structure de l'espace (Wernisch,Wodak)

Le but est d'obtenir un ensemble de séquences-rotamères qui approchent le minimum globale.

Pour chaque cycle heuristique
une séquence-conformation **S** est choisie aléatoirement

Tant que l'énergie de **S** est améliorée

Pour **i** allant de la première position de **S** jusqu'à la dernière

S est fixée sauf à la position **i**

le meilleur rotamère possible en **i** est déterminé.

Ce rotamère est utilisé pour fixer **S** en **i**

Fin de Pour

Fin de Tant que

S est sauvegardée

Fin du cycle heuristique



Une méthode heuristique spécifique à la structure de l'espace (Wernisch,Wodak)

Le but est d'obtenir un ensemble de séquences-rotamères qui approchent le minimum globale.

Pour chaque cycle heuristique
une séquence-conformation **S** est choisie aléatoirement
Tant que l'énergie de **S** est améliorée
Pour **i** allant de la première position de **S** jusqu'à la dernière
S est fixée sauf à la position **i**
le meilleur rotamère possible en **i** est déterminé.
Ce rotamère est utilisé pour fixer **S** en **i**
Fin de Pour
Fin de Tant que
S est sauvegardée
Fin du cycle heuristique



Une méthode heuristique spécifique à la structure de l'espace (Wernisch,Wodak)

Le but est d'obtenir un ensemble de séquences-rotamères qui approchent le minimum globale.

Pour chaque cycle heuristique
une séquence-conformation **S** est choisie aléatoirement
Tant que l'énergie de **S** est améliorée
Pour **i** allant de la première position de **S** jusqu'à la dernière
S est fixée sauf à la position **i**
le meilleur rotamère possible en **i** est déterminé.
Ce rotamère est utilisé pour fixer **S** en **i**
Fin de Pour
Fin de Tant que
S est sauvegardée
Fin du cycle heuristique



Une méthode heuristique spécifique à la structure de l'espace (Wernisch,Wodak)

Le but est d'obtenir un ensemble de séquences-rotamères qui approchent le minimum globale.

Pour chaque cycle heuristique
une séquence-conformation **S** est choisie aléatoirement
Tant que l'énergie de **S** est améliorée
Pour **i** allant de la première position de **S** jusqu'à la dernière
S est fixée sauf à la position **i**
le meilleur rotamère possible en **i** est déterminé.
Ce rotamère est utilisé pour fixer **S** en **i**

Fin de Pour

Fin de Tant que

S est sauvegardée

Fin du cycle heuristique



Une méthode heuristique spécifique à la structure de l'espace (Wernisch,Wodak)

Le but est d'obtenir un ensemble de séquences-rotamères qui approchent le minimum globale.

Pour chaque cycle heuristique
une séquence-conformation **S** est choisie aléatoirement

Tant que l'énergie de **S** est améliorée

Pour **i** allant de la première position de **S** jusqu'à la dernière

S est fixée sauf à la position **i**

le meilleur rotamère possible en **i** est déterminé.

Ce rotamère est utilisé pour fixer **S** en **i**

Fin de Pour

Fin de Tant que

S est sauvegardée

Fin du cycle heuristique



Une méthode heuristique spécifique à la structure de l'espace (Wernisch,Wodak)

Le but est d'obtenir un ensemble de séquences-rotamères qui approchent le minimum globale.

Pour chaque cycle heuristique
une séquence-conformation **S** est choisie aléatoirement
Tant que l'énergie de **S** est améliorée
Pour **i** allant de la première position de **S** jusqu'à la dernière
S est fixée sauf à la position **i**
le meilleur rotamère possible en **i** est déterminé.
Ce rotamère est utilisé pour fixer **S** en **i**

Fin de Pour

Fin de Tant que
S est sauvegardée

Fin du cycle heuristique



Une méthode heuristique spécifique à la structure de l'espace (Wernisch,Wodak)

Le but est d'obtenir un ensemble de séquences-rotamères qui approchent le minimum globale.

Pour chaque cycle heuristique
une séquence-conformation **S** est choisie aléatoirement
Tant que l'énergie de **S** est améliorée
Pour **i** allant de la première position de **S** jusqu'à la dernière
S est fixée sauf à la position **i**
le meilleur rotamère possible en **i** est déterminé.
Ce rotamère est utilisé pour fixer **S** en **i**
Fin de Pour
Fin de Tant que
S est sauvegardée
Fin du cycle heuristique



Une méthode heuristique spécifique à la structure de l'espace (Wernisch,Wodak)

Le but est d'obtenir un ensemble de séquences-rotamères qui approchent le minimum globale.

Pour chaque cycle heuristique
une séquence-conformation **S** est choisie aléatoirement
Tant que l'énergie de **S** est améliorée
Pour **i** allant de la première position de **S** jusqu'à la dernière
S est fixée sauf à la position **i**
le meilleur rotamère possible en **i** est déterminé.
Ce rotamère est utilisé pour fixer **S** en **i**
Fin de Pour
Fin de Tant que
S est sauvegardée
Fin du cycle heuristique



Le Monte-Carlo

algorithme Metropolis-Hastings

Le but est de générer une collection d'états échantillonnes selon la distribution de Boltzmann.

$$p(x) = \frac{1}{Z} e^{-\frac{E}{RT}}$$

L'algorithme définit une chaîne de Markov pour laquelle:

1- La distribution de probabilité des états est stationnaire

C'est garantie par la balance détaillée.

2- Il n'y a qu'une seule distribution stationnaire.

C'est garantie par le caractère ergodique de la chaine.

Le Monte-Carlo

Algorithme Metropolis-Hastings

La distribution cible étant donnée $p(\cdot)$
Une séquence-conformation S est choisie aléatoirement
Pour chaque pas de la trajectoire

Une modification possible est sélectionnée à partir d'une distribution conditionnelle:

$$S' \sim q(S'|S)$$

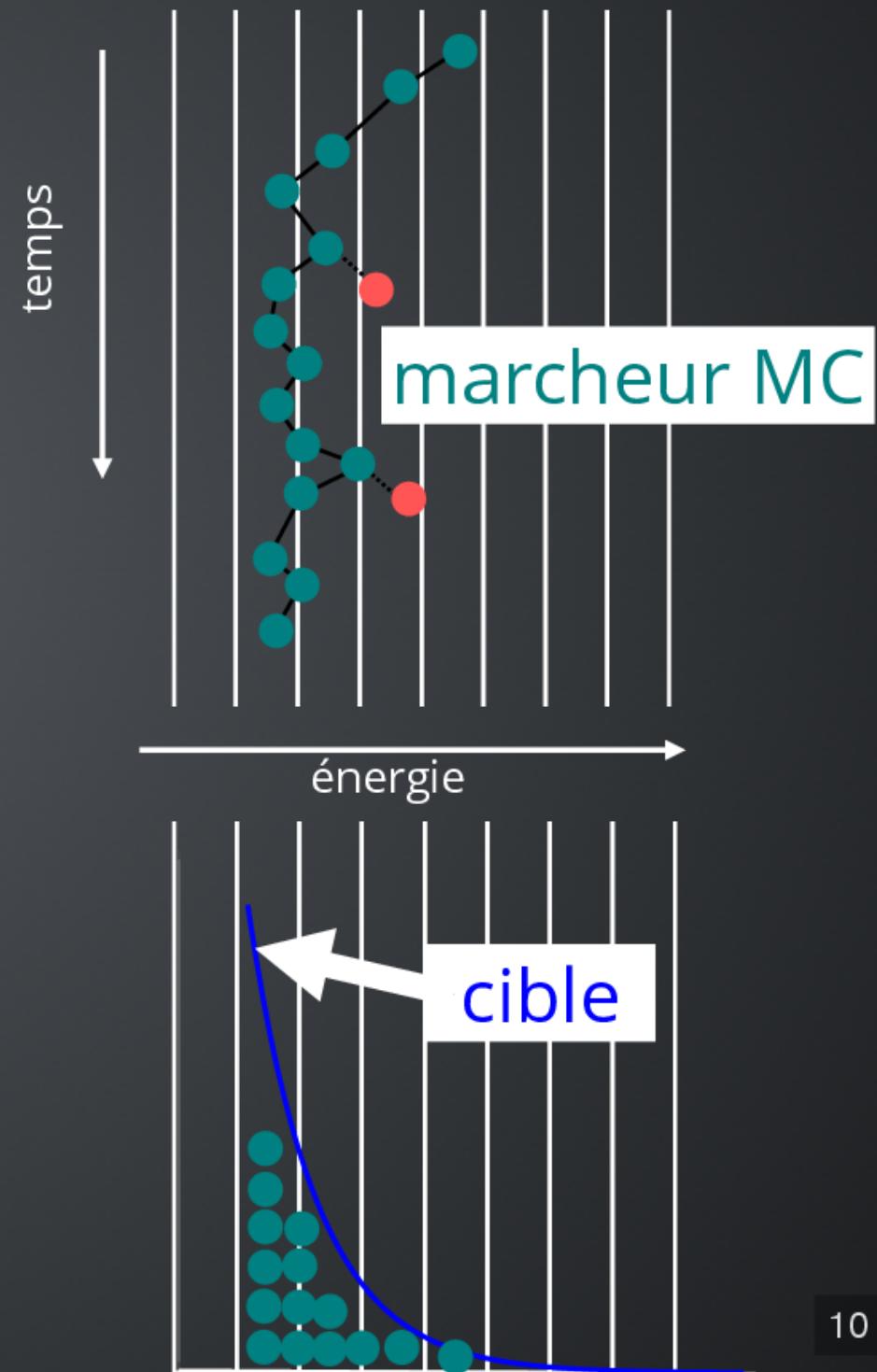
La modification est acceptée selon la probabilité:

$$\alpha(S', S) = \min\left\{1, \frac{p(S')}{p(S)} \frac{q(S|S')}{q(S'|S)}\right\}$$

Fin de Pour

indépendant de la fonction de partition

généralisation d'Hastings



Replica Exchange Monte Carlo

accélérer la convergence en visitant plusieurs zones énergétiques simultanément

Lancement en parallèle de N marcheurs Monte Carlo aux températures ordonnées (t_1, \dots, t_n)

Périodiquement un couple de marcheurs aux températures (t_i, t_{i+1}) est sélectionnés aléatoirement.

Les températures entre les deux marcheurs sont échangées selon la probabilité:

$$\beta = \min\left\{1, \exp\left(-\left(\frac{1}{kt_i} - \frac{1}{kt_{i+1}}\right)(E_{t_i} - E_{t_{i+1}})\right)\right\}$$

- L'échange de températures est une mouvement supplémentaire dans l'espace généralisé des N répliques
- β est l'expression de la balance détaillée pour ce mouvement

Donc les propriétés Monte Carlo de la trajectoire sont conservées.

Une méthode exacte par optimisation combinatoire (Toulbar2)

Le but est la recherche du minimum globale de la fonction d'énergie (GMEC)

La décomposition par paire de notre fonction d'énergie permet une représentation sous forme d'un réseau de fonctions de coûts

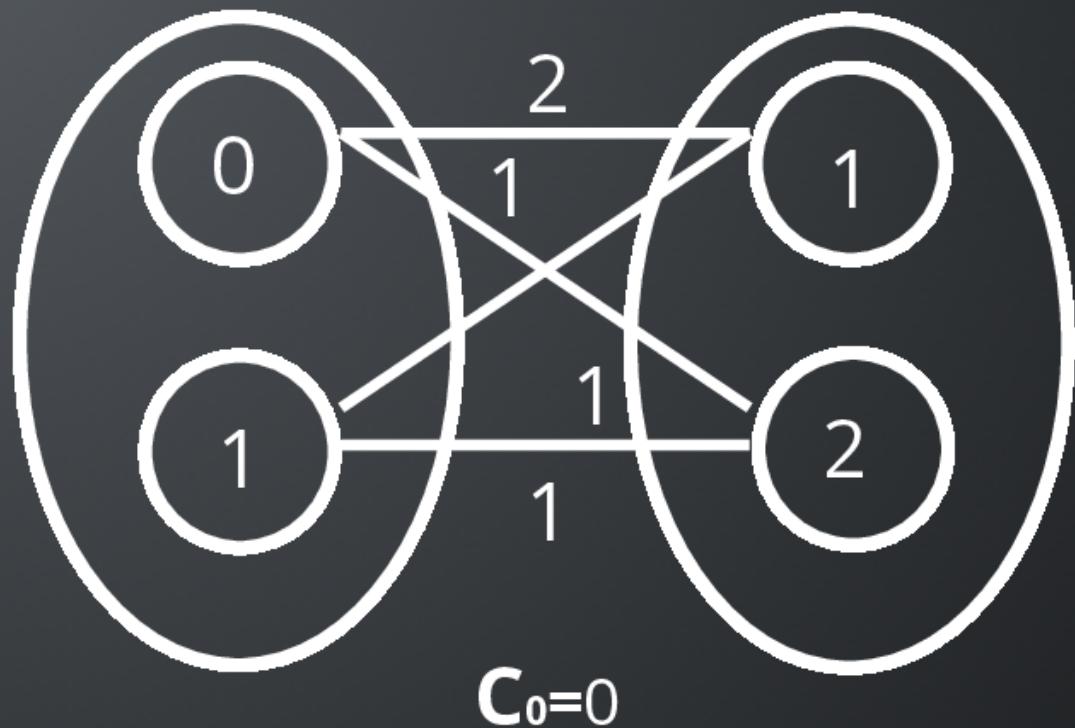
- Une interaction entre acides aminés \Leftrightarrow une arête du réseau
- Une énergie d'un rotamère \Leftrightarrow un nœud du réseau

Equivalence Preserving Transformation (EPT)

Un ensemble de transformations qui préservent l'équivalence des problèmes est appliqués sur le réseau

Deux transformations de base:

- la projection
- la distribution



Une méthode exacte par optimisation combinatoire (Toulbar2)

Le but est la recherche du minimum globale de la fonction d'énergie (GMEC)

La décomposition par paire de notre fonction d'énergie permet une représentation sous forme d'un réseau de fonctions de coûts

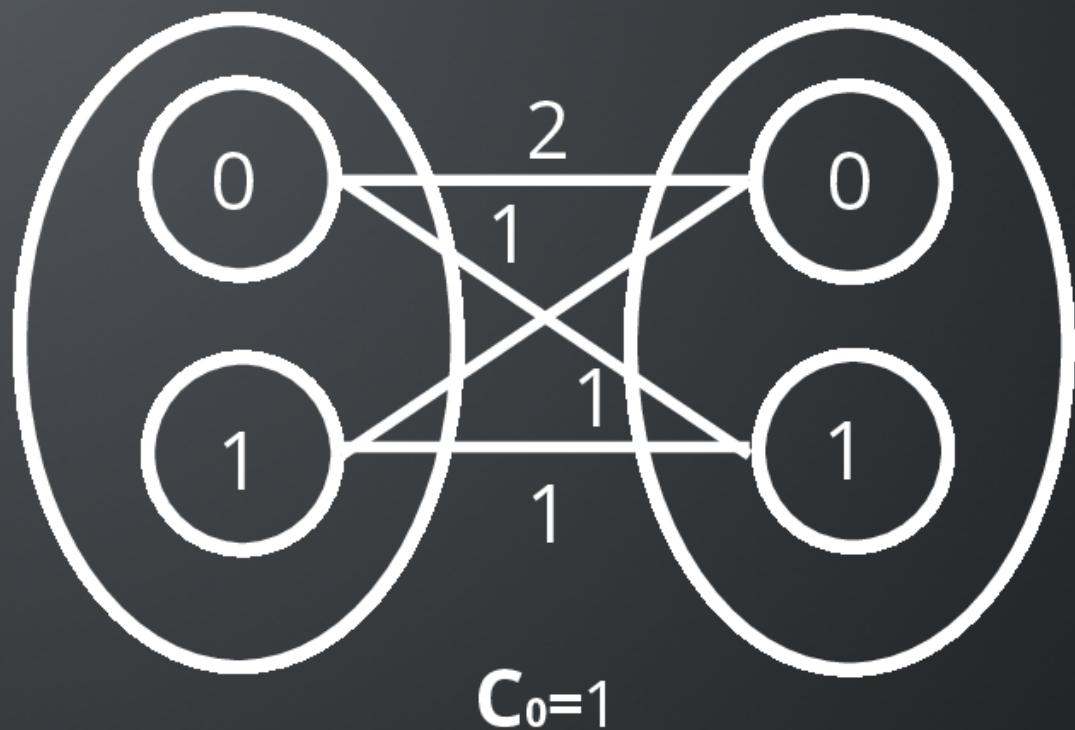
- Une interaction entre acides aminés \Leftrightarrow une arête du réseau
- Une énergie d'un rotamère \Leftrightarrow un nœud du réseau

Première étape:

Un ensemble de transformations qui préservent l'équivalence des problèmes est appliqués sur le réseau

Deux transformations de base:

- la projection
- la distribution



Une méthode exacte par optimisation combinatoire (Toulbar2)

Le but est la recherche du minimum globale de la fonction d'énergie (GMEC)

La décomposition par paire de notre fonction d'énergie permet une représentation sous forme d'un réseau de fonctions de coûts

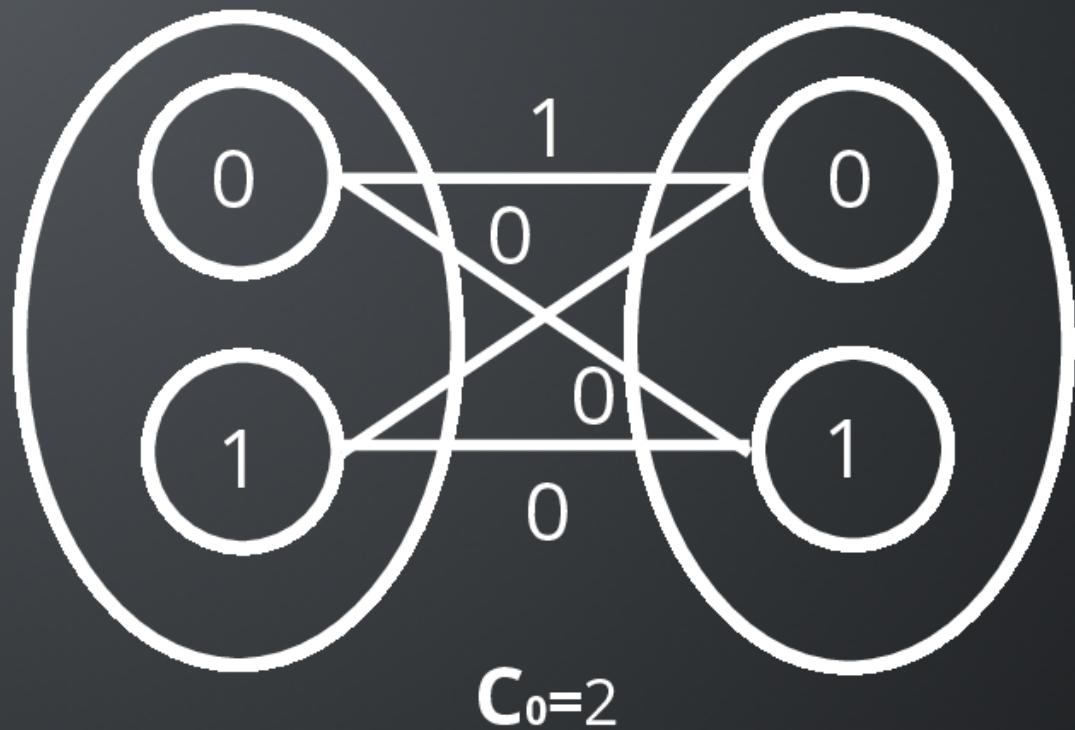
- Une interaction entre acides aminés \Leftrightarrow une arête du réseau
- Une énergie d'un rotamère \Leftrightarrow un nœud du réseau

Première étape:

Un ensemble de transformations qui préservent l'équivalence des problèmes est appliqués sur le réseau

Deux transformations de base:

- la projection
- la distribution



Une méthode exacte par optimisation combinatoire (Toulbar2)

Le but est la recherche du minimum globale de la fonction d'énergie (GMEC)

La décomposition par paire de notre fonction d'énergie permet une représentation sous forme d'un réseau de fonctions de coûts

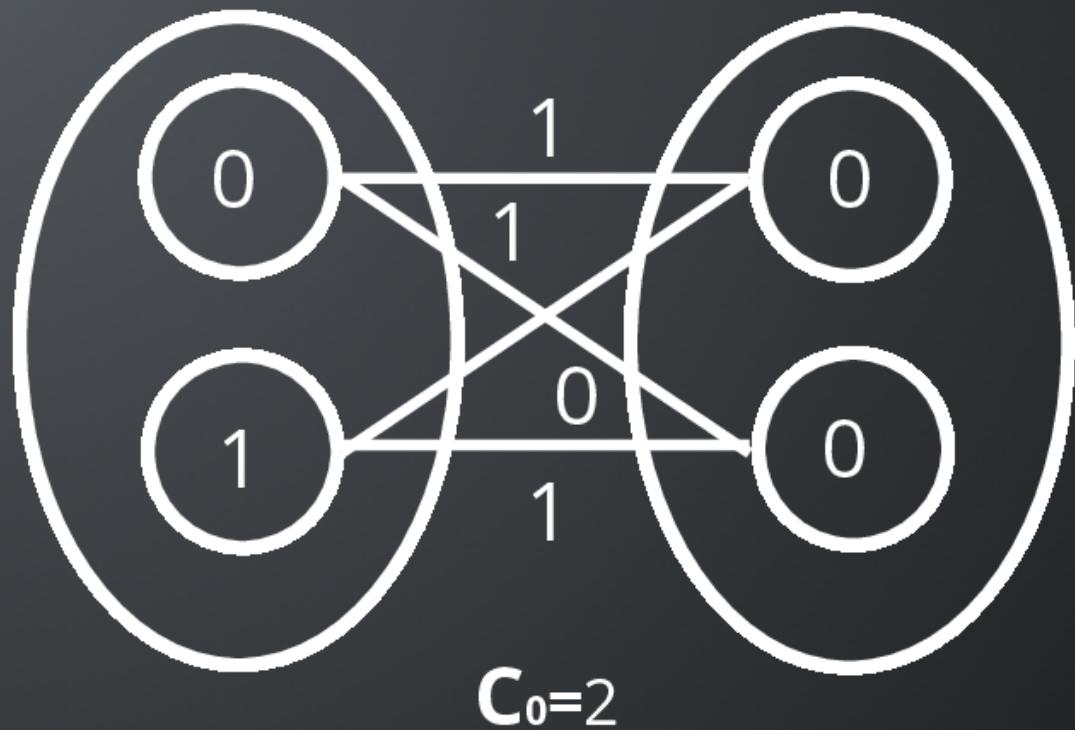
- Une interaction entre acides aminés \Leftrightarrow une arête du réseau
- Une énergie d'un rotamère \Leftrightarrow un nœud du réseau

Première étape:

Un ensemble de transformations qui préservent l'équivalence des problèmes est appliqués sur le réseau

Deux transformations de base:

- la projection
- la distribution



Une méthode exacte par optimisation combinatoire (Toulbar2)

Le but est la recherche du minimum globale de la fonction d'énergie (GMEC)

La décomposition par paire de notre fonction d'énergie permet une représentation sous forme d'un réseau de fonctions de coûts

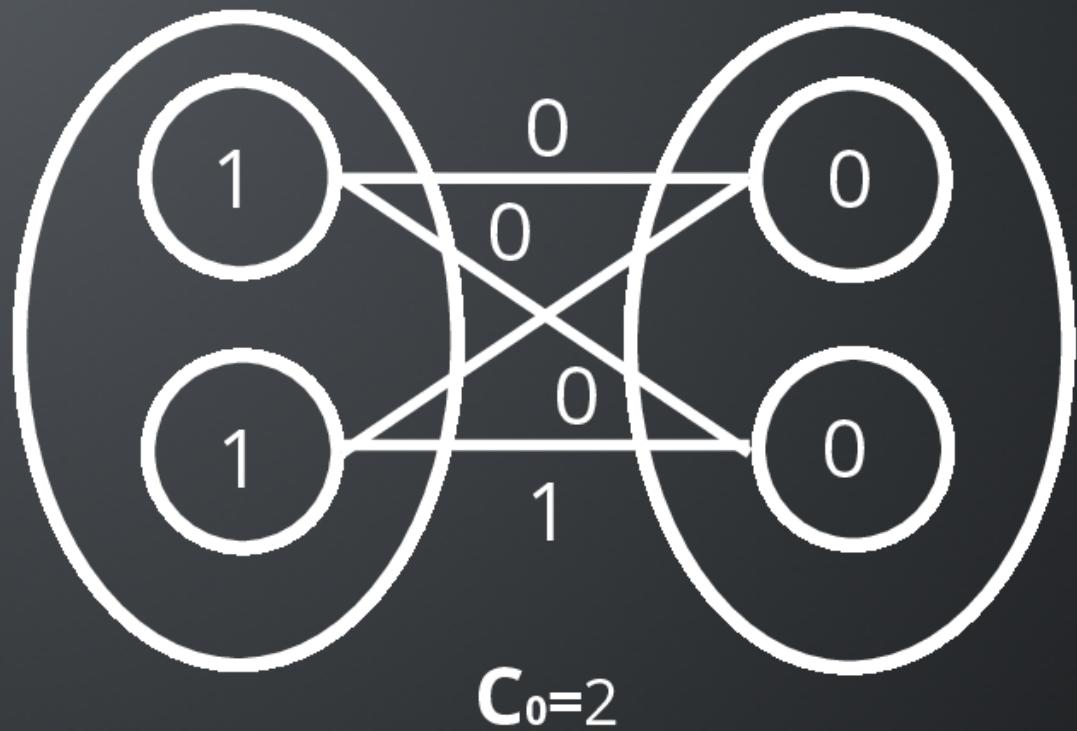
- Une interaction entre acides aminés \Leftrightarrow une arête du réseau
- Une énergie d'un rotamère \Leftrightarrow un nœud du réseau

Première étape:

Un ensemble de transformations qui préservent l'équivalence des problèmes est appliqués sur le réseau

Deux transformations de base:

- la projection
- la distribution



Une méthode exacte par optimisation combinatoire (Toulbar2)

Le but est la recherche du minimum globale de la fonction d'énergie (GMEC)

La décomposition par paire de notre fonction d'énergie permet une représentation sous forme d'un réseau de fonctions de coûts

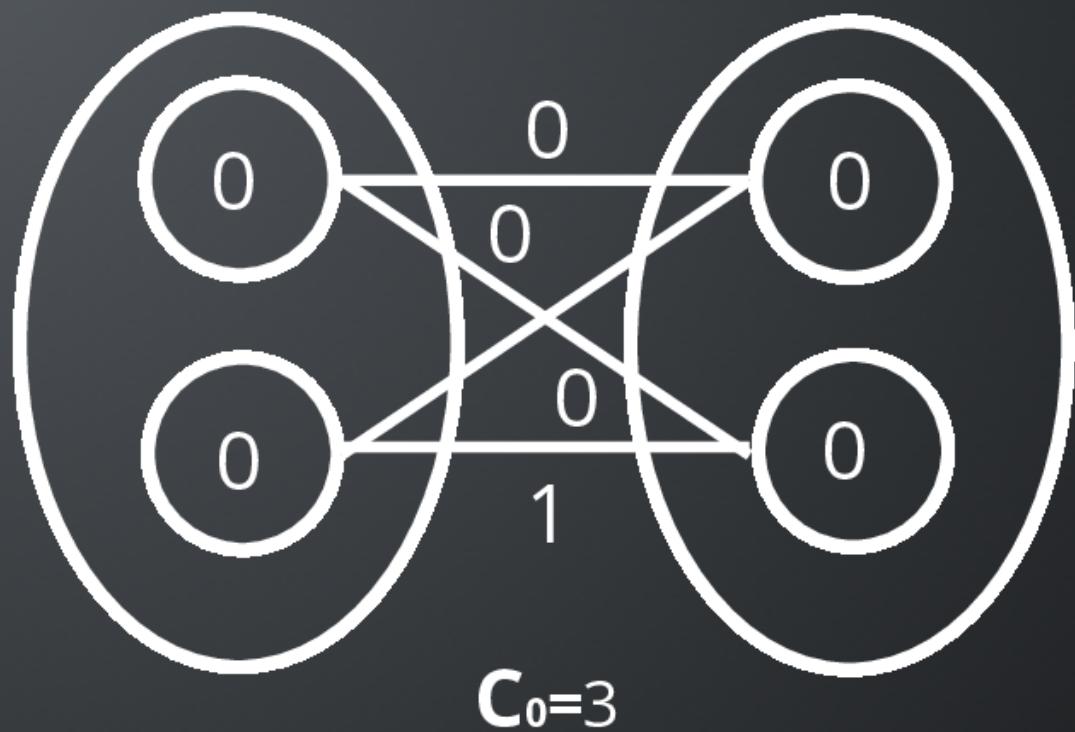
- Une interaction entre acides aminés \Leftrightarrow une arête du réseau
- Une énergie d'un rotamère \Leftrightarrow un nœud du réseau

Première étape:

Un ensemble de transformations qui préservent l'équivalence des problèmes est appliqués sur le réseau

Deux transformations de base:

- la projection
- la distribution



Une méthode exacte par optimisation combinatoire (Toulbar2)

$$M_{S_0} = 0$$

L'algorithme "Depth-First Branch and Bound"

Le principe de séparation: partitionner l'ensemble des séquences-conformations en sous-ensemble fils

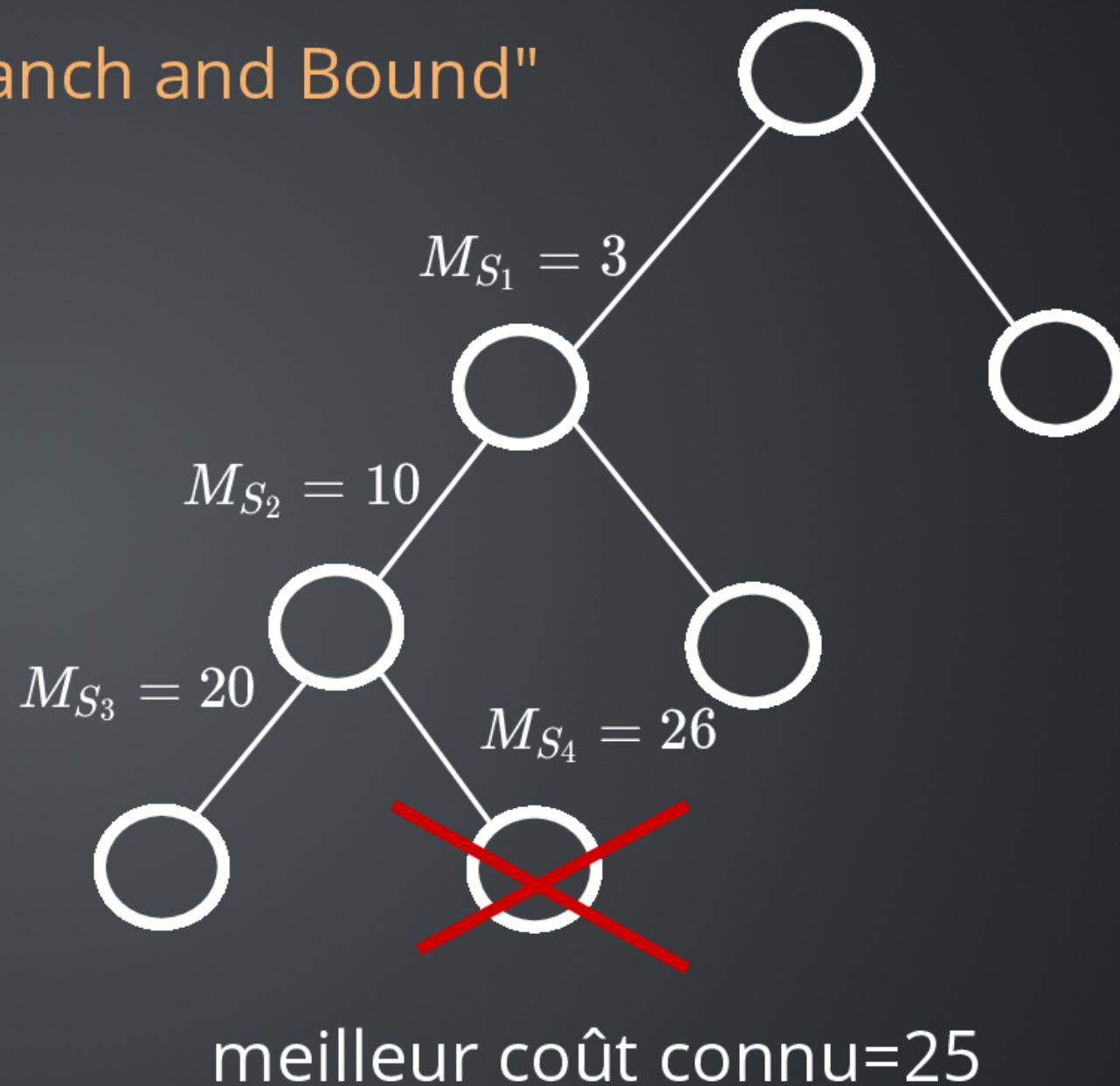
→ construction d'un arbre

Le développement de l'arbre: descendre dans les branches autant que possible, sinon remonter d'un sommet.

L'EPT produit un **minorant** M_S des coûts d'un sommet.

Si le meilleur coût connu est inférieur à un minorant d'un sommet S , on peut élaguer l'arbre en S .

Mise à jour des minorants



"Dead-End Elimination" en complément

Comparaison d'algorithmes

Comparaison des algorithmes

L'ensemble de tests

Systèmes

Modèle

Protéine	nb résidus	famille
1A81	108	SH2
1BM2	98	SH2
1M61	109	SH2
1O4C	104	SH2
1ABO	58	SH3
1CKA	57	SH3
1G9O	91	PDZ
1R6J	82	PDZ
2BYG	97	PDZ

- Amber (ff99SB)
 - CASA, $\epsilon = 23$
 - toutes les positions mutables, sauf GLY et PRO
 - énergies de références optimisées
- sur les 3 familles

Comparaison des algorithmes

méthodes

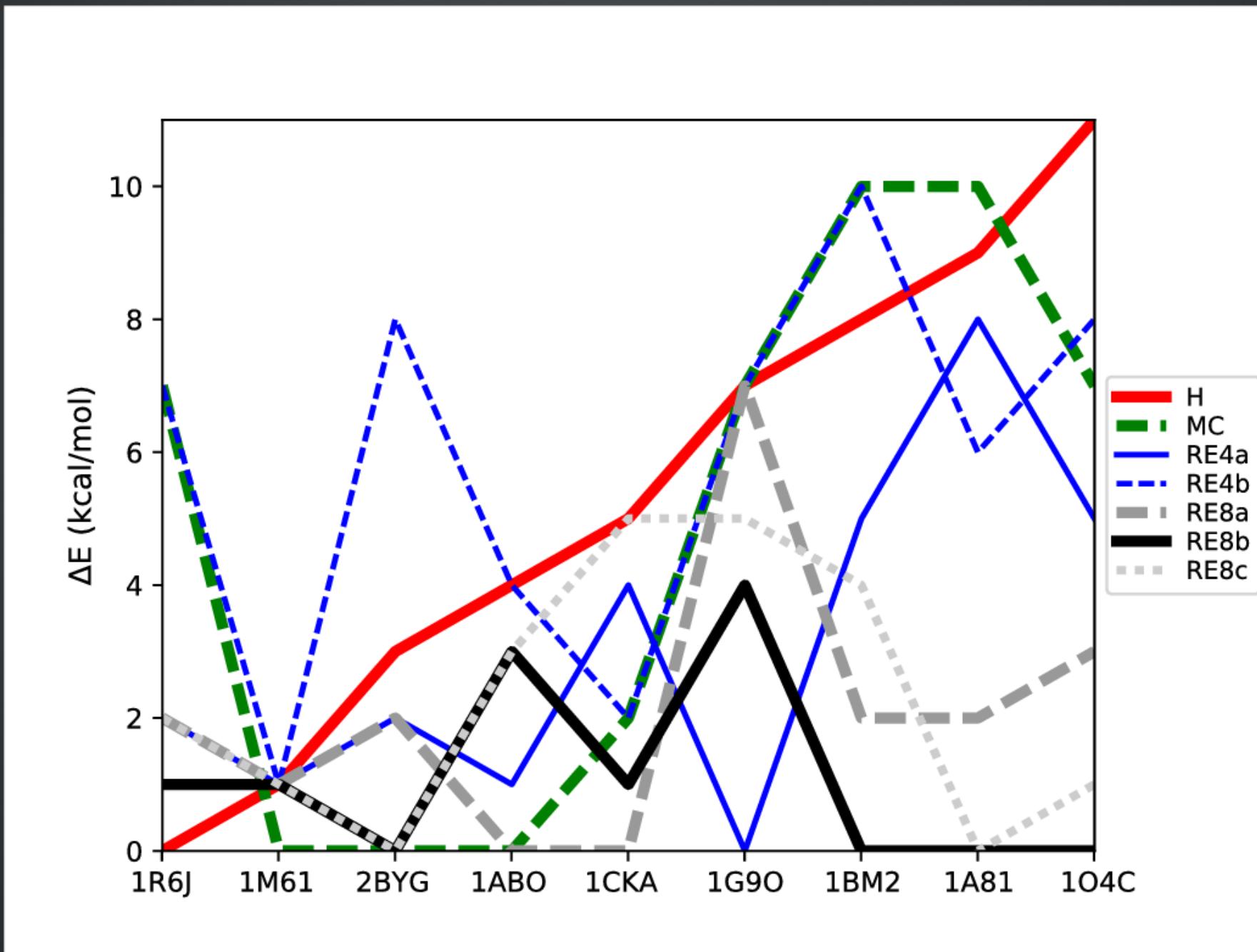
temps d'exécution limité à 24h

- Heuristique: 110 000 cycles
- Monte carlo: 6 milliards de pas
- REMC: 6 milliards de pas cumulés sur les marcheurs

Paramétrages

Algo	nb marcheurs	températur es	mutation / pas	change de rotamères/ pas	freq swap
MC	1	0,2	1,1	0,1	-
REMC	4	0,125...1	0,1	1,1	0,005
REMC	4	0,25...2	0,1	1,1	0,005
REMC	8	0,175...3	1,1	0,1	0,01
REMC	8	0,175...3	0,1	1,1	0,01
REMC	8	0,175...3	0,1	1,1	0,001

Comparaison des meilleures énergies de chaque protocole

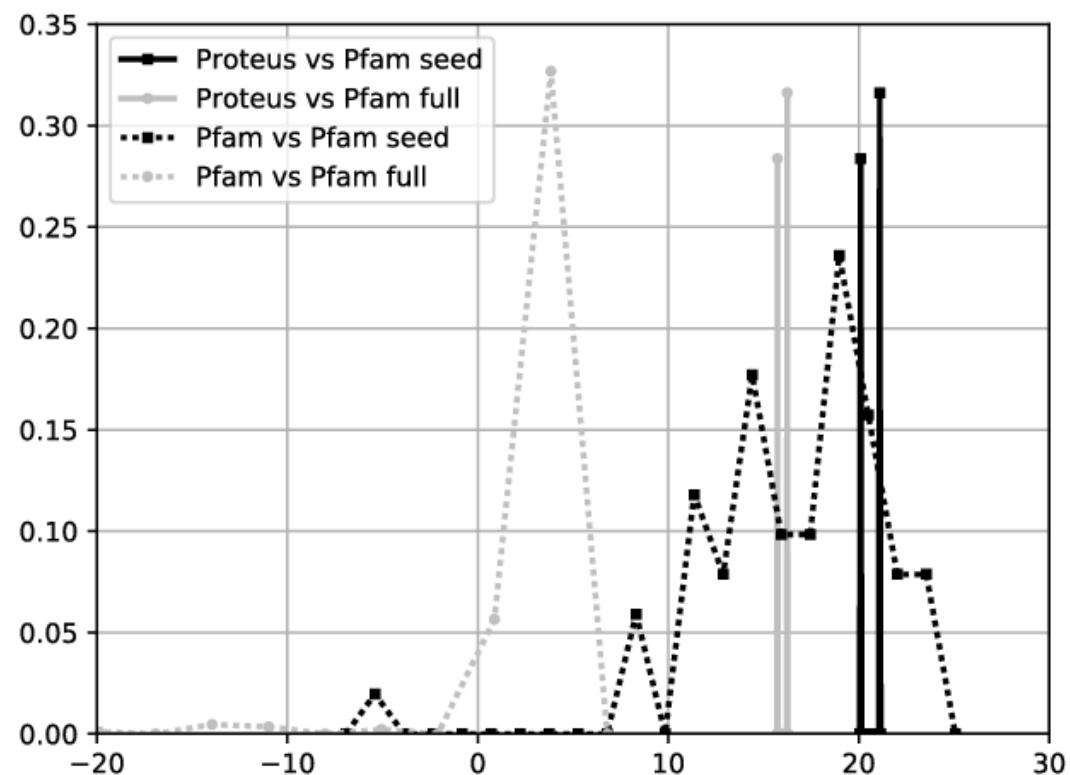


Caractérisation des séquences

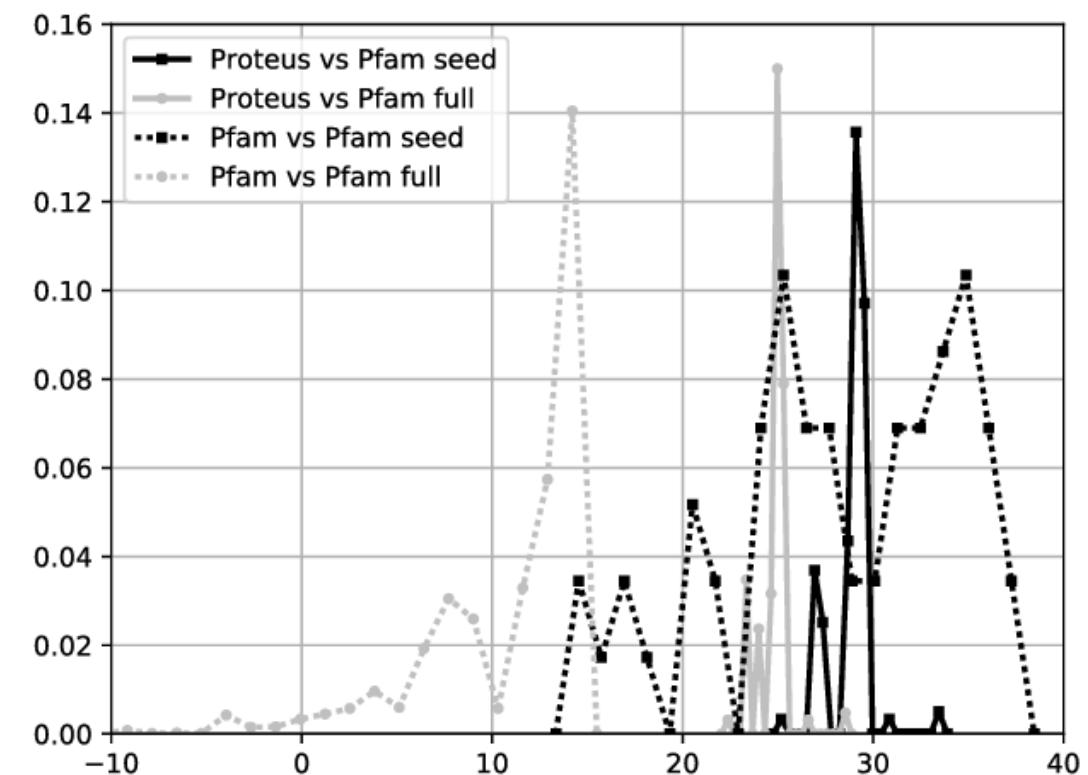
- calcul de similarité aux alignements "seed" et "full" de Pfam (Protein families database) de la famille correspondante, grâce à la matrice BLOSUM64, aux positions du cœur.
- soumission à Superfamily reconnaissance de structure 3D à partir d'une bibliothèque de HMM obtenue à partir de la classification SCOP
- Taux d'identité à la séquence native (backbone)
- Entropie résiduel par position comme mesure de la diversité

Scores de similarité sur les positions du cœur

1ABO



1BM2



Résultats Superfamily et identité

sur les 10000 séquences-rotamères de meilleures énergies

Protéine	nb de séquences	% identité à la native	taille du "match"	E-value Super-famille	% succès Super-famille	E-value famille	taux succès famille
1A81	236	27	none				
1ABO	203	32	51/58	4.4e-4	100%	2.8e-3	100%
1BM2	209	27	78/98	4.2e-5	100%	2.6e-3	100%
1CKA	416	33	40/57	1.1e-5	100%	3.4e-3	100%
1G9O	338	36	79/91	7.0e-7	100%	2.5e-3	100%
1M61	405	42	97/109	7.2e-7	100%	2.6e-4	100%
1O4C	274	21	95/104	2.1e-4	100%	4.6e-3	100%
1R6J	270	34	74/82	9.8e-6	100%	4.6e-3	100%
2BYG	426	28	59/97	1.4e-5	100%	7.1e-3	100%

Recherche du GMEC

l'Espace est réduit progressivement en fixant une partie des résidus avec leur type natif:

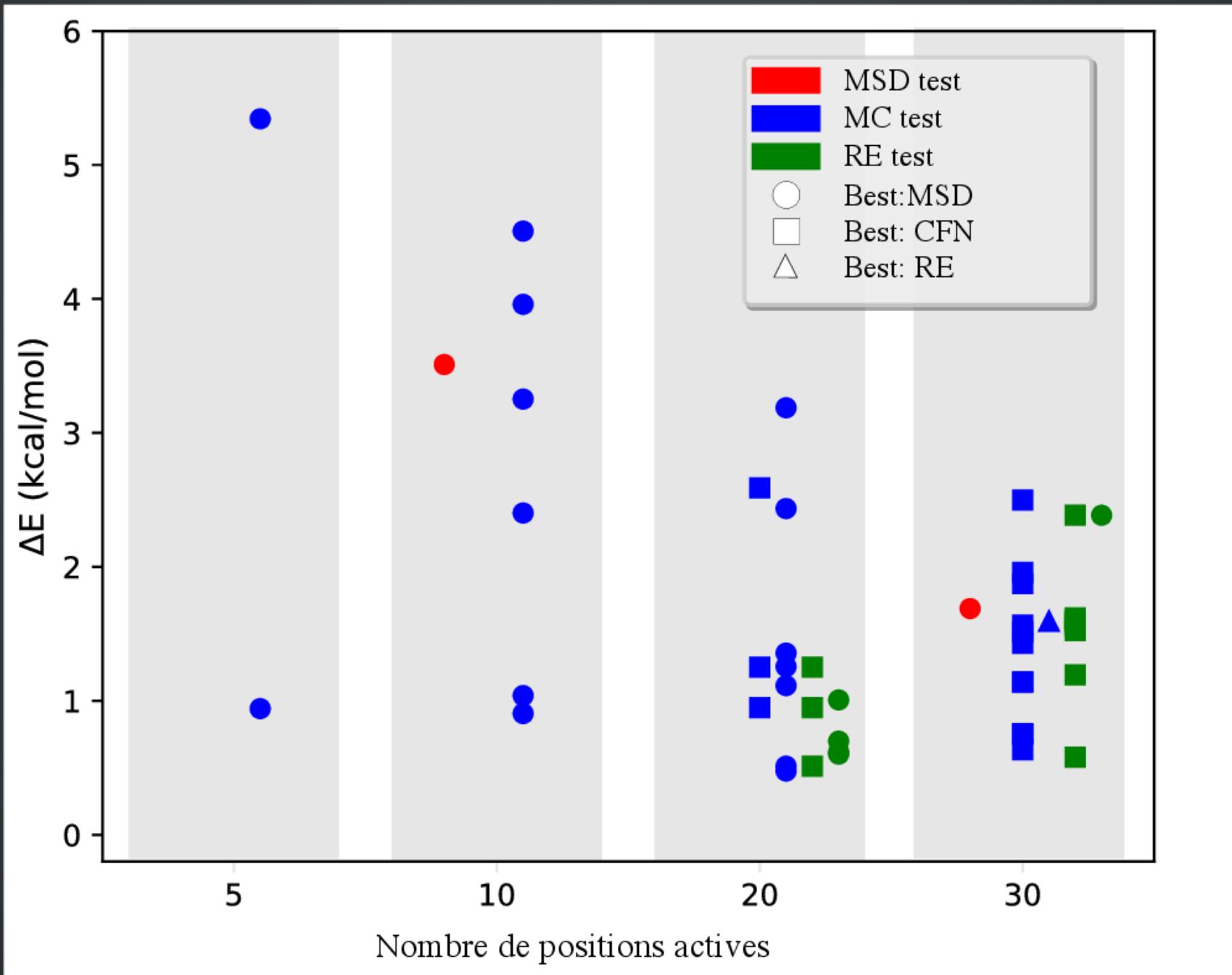
- Tests à 30, 20, 10 et 5 positions actives
- Dans chaque cas 5 sélections en privilégiant les ensembles en interaction pour chacune des 9 protéines.

CFN: Temps d'exécution max 24h, en cas d'échec relance avec une seconde configuration

Nos algorithmes: MSD et le meilleur RMEC

Résultats

Différences avec la meilleure énergie

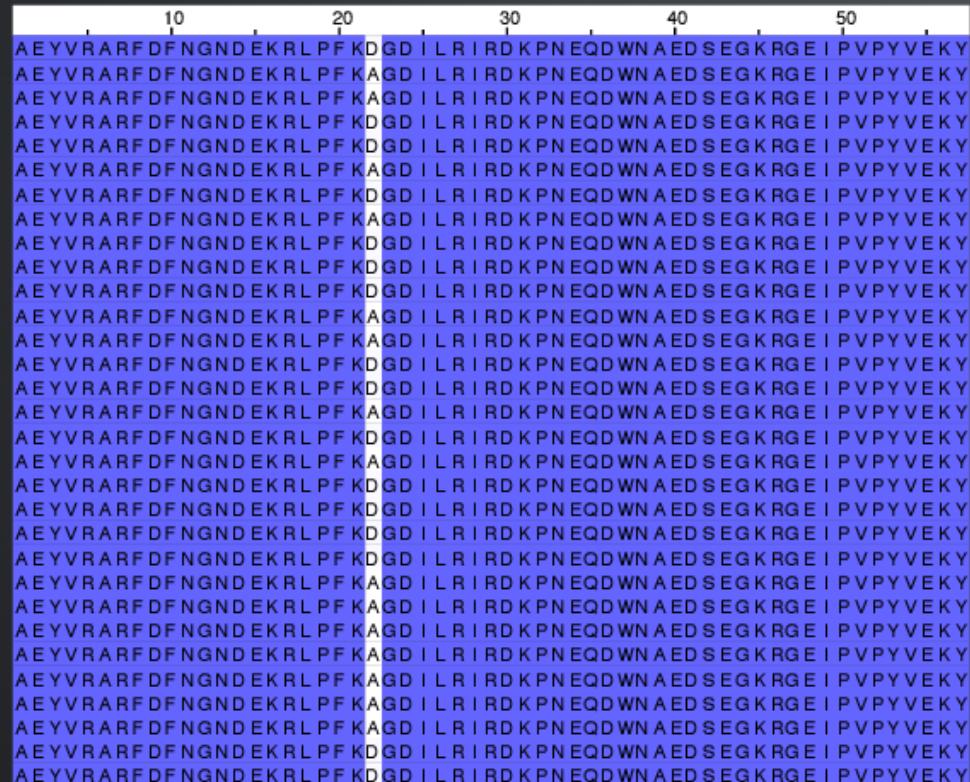


Études de quelques cas

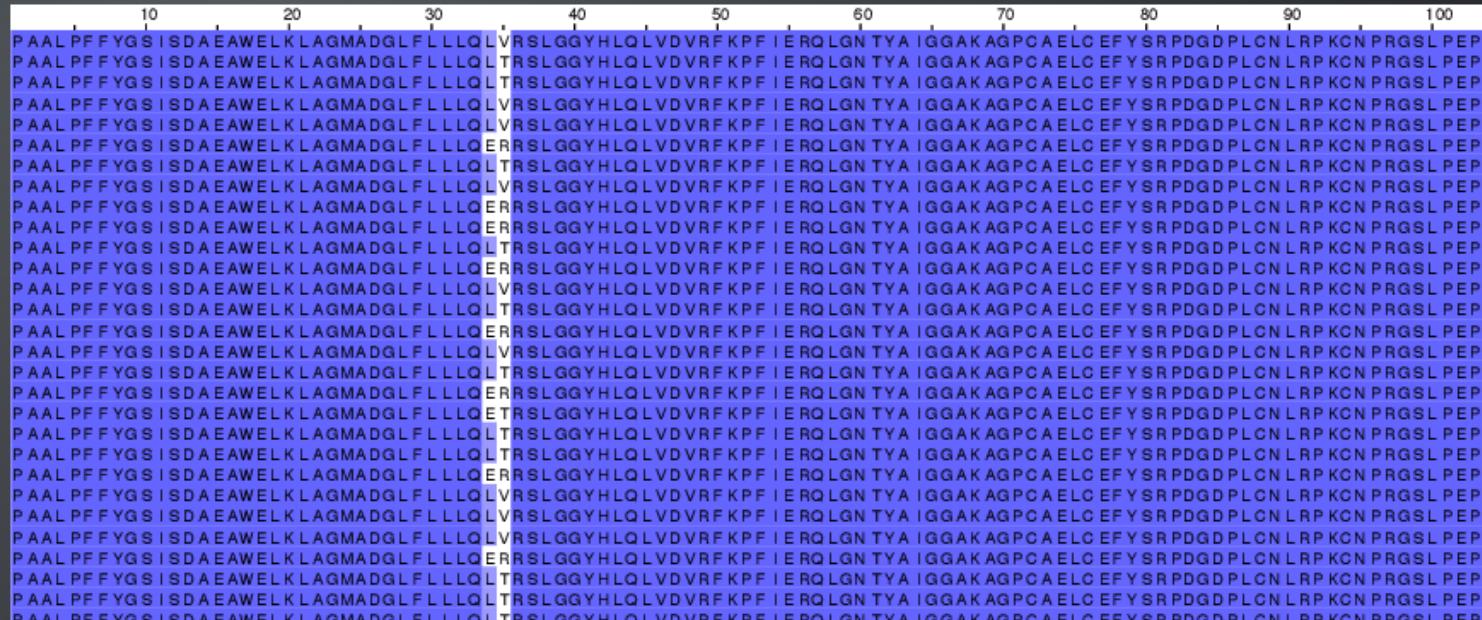
Séquences au voisinage du GMEC

1CKA 10 actifs

1M61 10 actifs

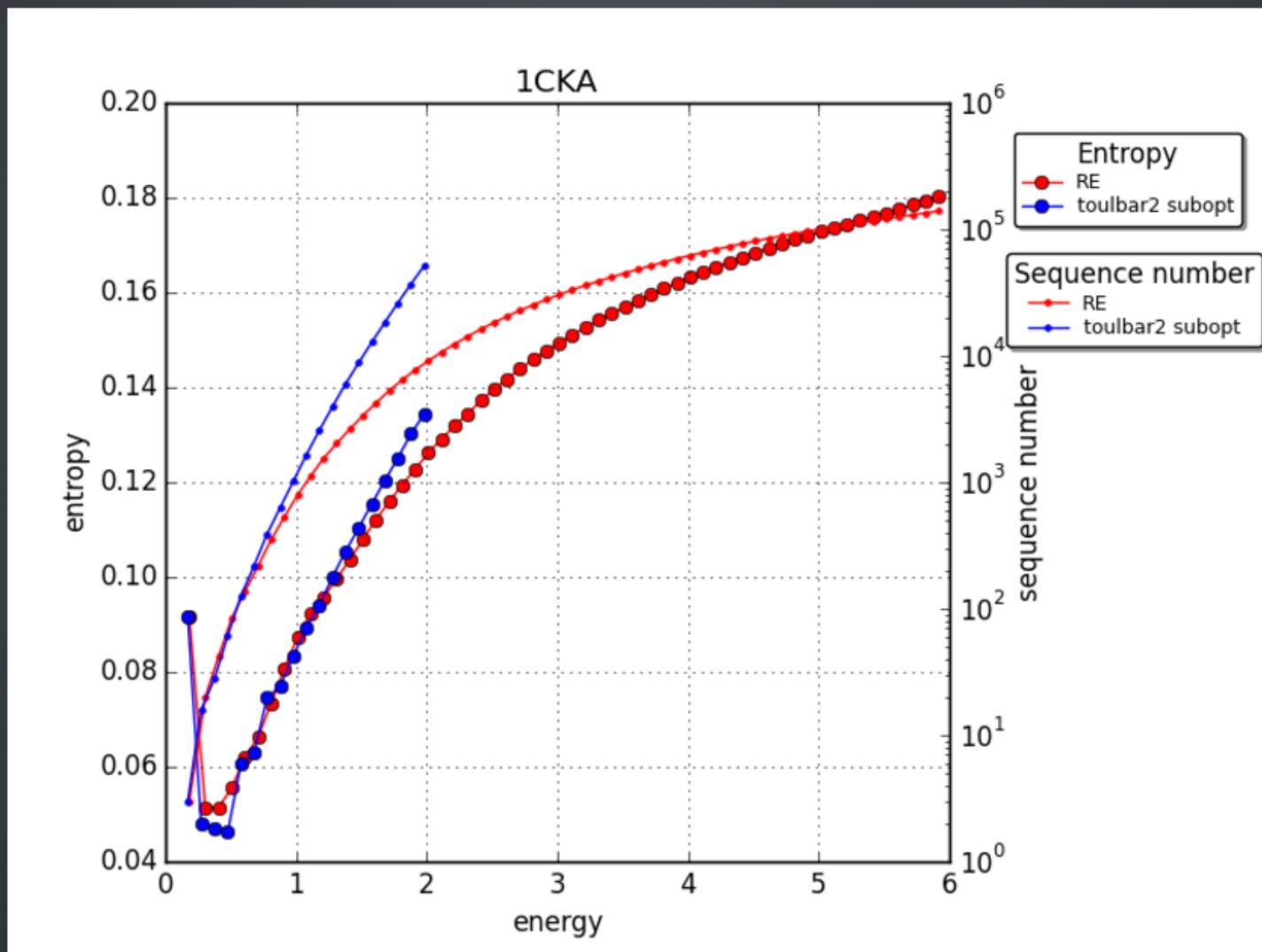


Difficile pour le Monte



Difficile pour le MSD

Études de quelques cas densité au voisinage du GMEC



Mise à jour Proteus

- optimisation du Monte Carlo
- notion de rotamère dans Xplor
- nouvelle méthode de calcul des énergies de références
- nouvelle approximation GB

Optimisation du Monte Carlo

- Ajustement du facteur Hasting
- critère sur la meilleure énergie
- changement de la détermination d'un pas

notion de rotamère dans Xplor

Maximiser la vraisemblance des énergies de références

Définition: Pour une séquence s , l'énergie de l'état déplié est de la forme:

$$E_s^u = \sum_{i \in s} E_{t_i}^r$$

Ce sont des paramètres ajustables.

L'objectif est de déterminer les E_t^r pour obtenir les bonnes fréquences d'acide aminé.

La méthode (maximum de vraisemblance):

Soit \mathcal{S} un ensemble de séquences de Swissprot , $p(\mathcal{S})$ sa probabilité de Boltzmann est une fonction des E_t^r .

Nous cherchons les E_t^r qui maximisent $p(\mathcal{S})$, elles réalisent notre objectif.

Un algorithme itératif:

$$E_t^r(n+1) = E_t^r(n) + \delta E \times (freq_t^{exp} - freq_t^{proteus_n})$$

Optimisations des énergies de références

Algorithme

LE GB/FDB

Dessin computationnel de domaines PDZ

les protéines

le protocole

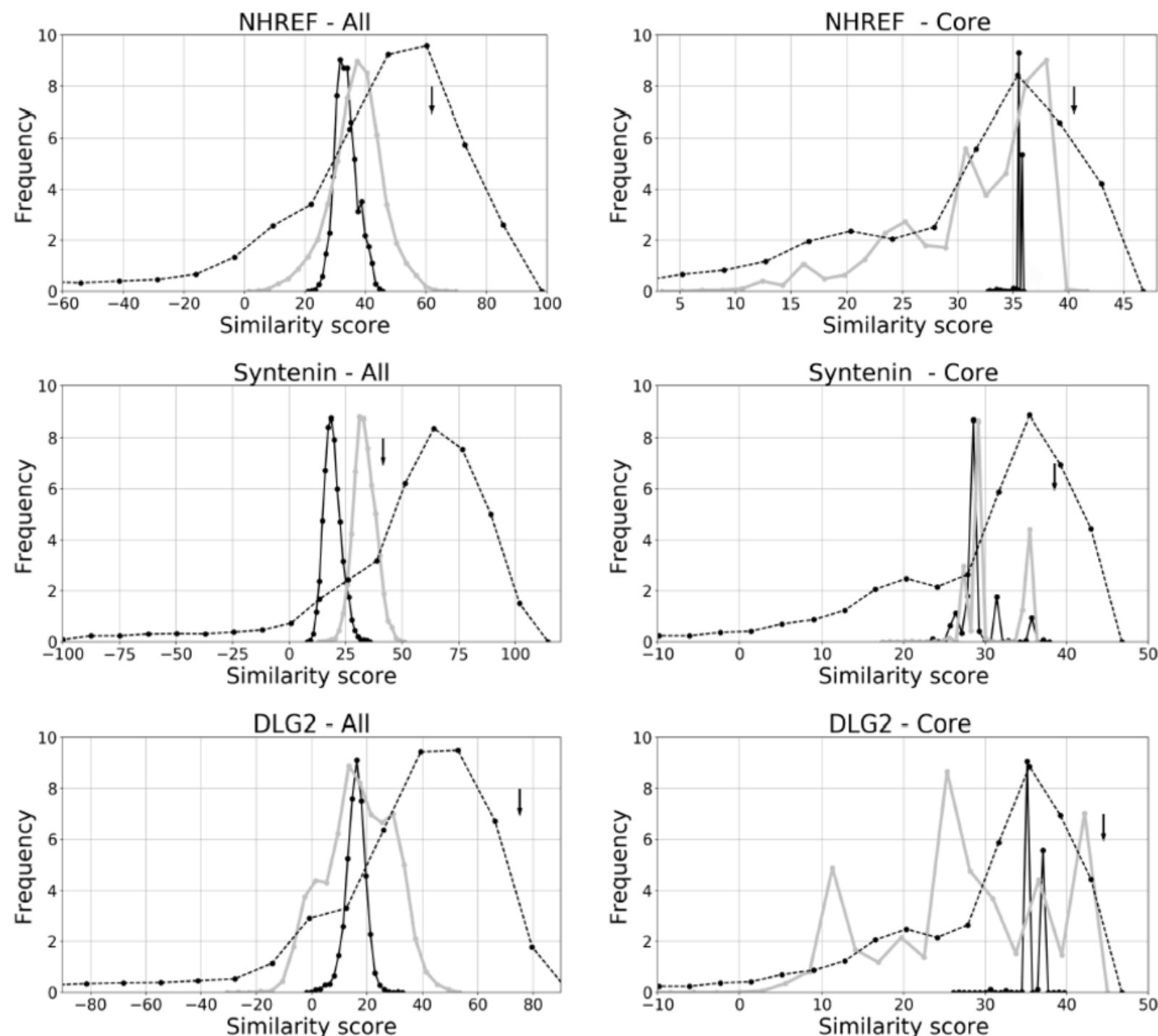
caractérisation/comparaison Rosetta

Superfamilly

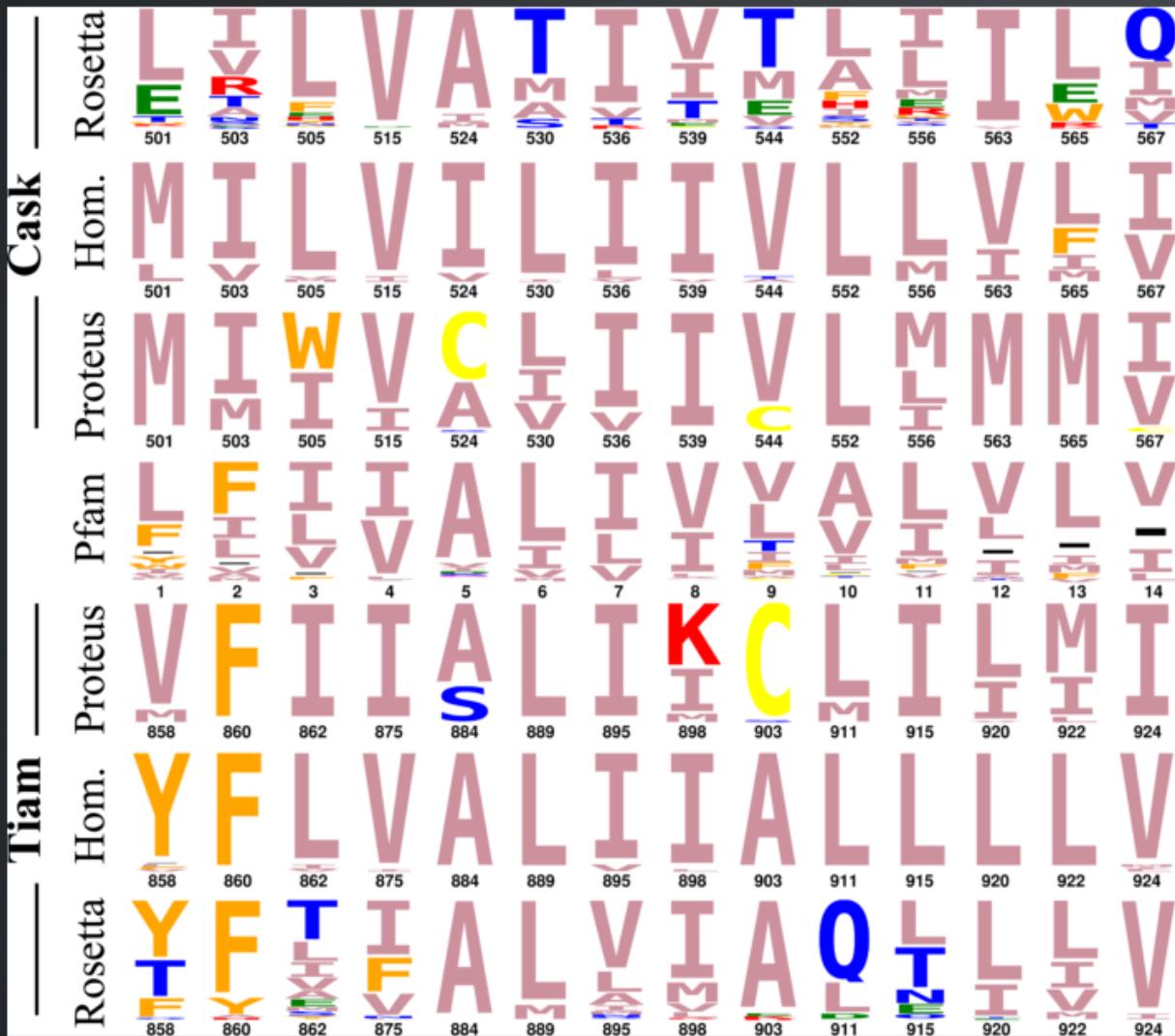
Protein	Proteus		Rosetta	
	Family Eval	Family success	Family Eval	Family success
NHREF	$8.94 \cdot 10^{-2}$	10000	$2.2 \cdot 10^{-3}$	10000
Syntenin	$2.69 \cdot 10^{-3}$	10000	$1.8 \cdot 10^{-3}$	10000
DLG2	$1.96 \cdot 10^{-3}$	10000	$9.6 \cdot 10^{-4}$	10000
Tiam1	$1.96 \cdot 10^{-3}$	10000	$2.8 \cdot 10^{-2}$	9030
Cask	$1.96 \cdot 10^{-3}$	10000	$7.5 \cdot 10^{-3}$	9832

Similarité

- Proteus
- Rosetta
- Pfam
- Native



Séquence Proteus et Rosetta sous forme de logos



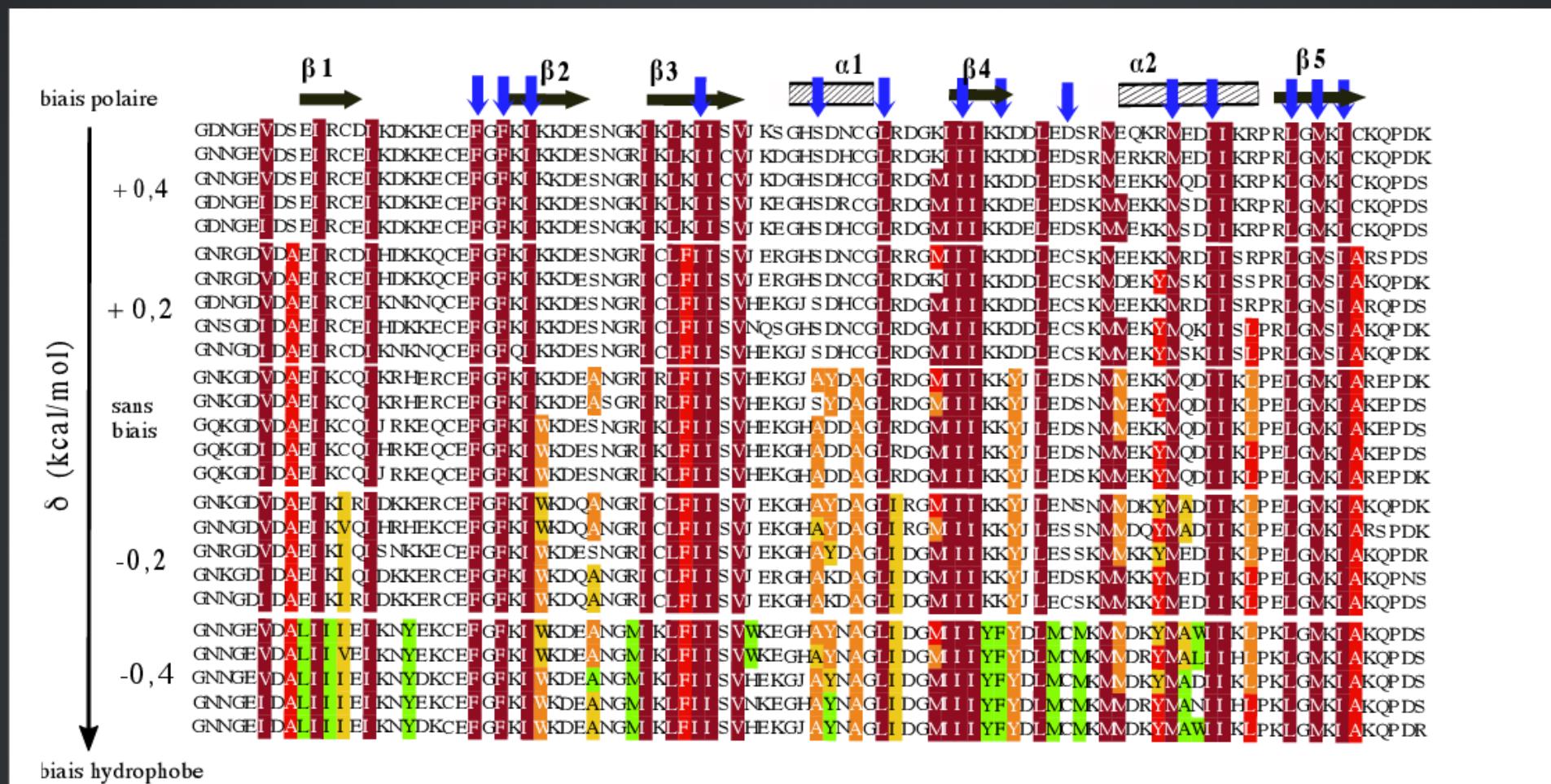
Entropie

Tests de validation croisée

Croissance du noyau hydrophobe

Principe

Croissance du noyau hydrophobe



Conclusion

Perceptives