

Cours M2 BIBS - Séance 1

Repliement *in silico* de l'ARN

Yann Ponty

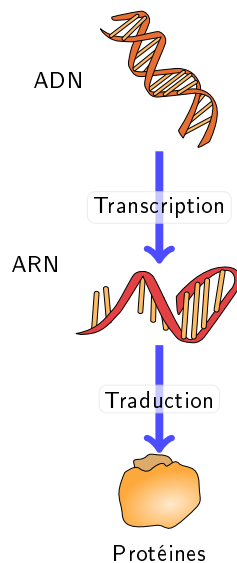
Bioinformatics Team
École Polytechnique/CNRS/INRIA AMIB - France

6 Décembre 2010

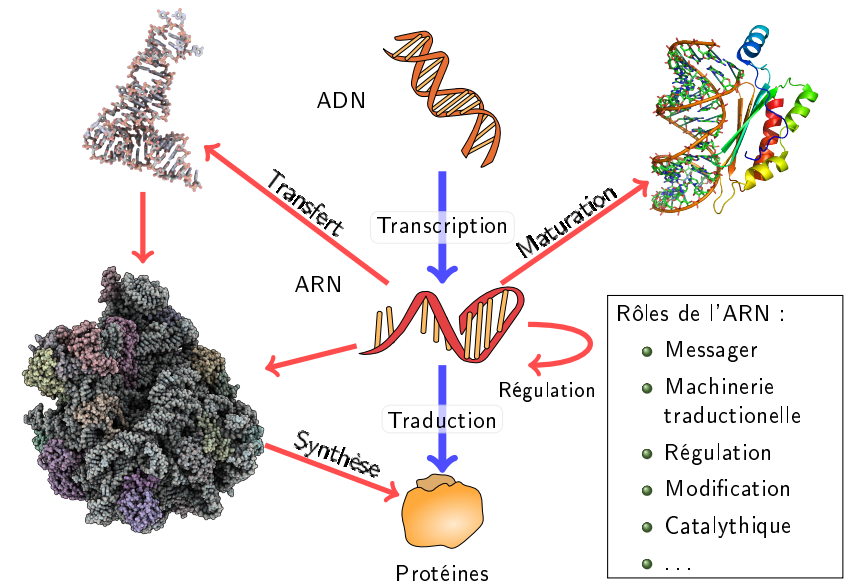
Résumé

- 1 Introduction
 - Fonction(s) de l'ARN
 - Repliement et structure
 - Représentations de la structure secondaire
- 2 Formalisation du repliement et outils disponibles
 - Aparté thermodynamique
 - Programmation dynamique : Rappels
- 3 Minimisation de l'énergie libre
 - Modèle de Nussinov
 - Modèle de Turner
 - MFold/Unafold
 - Performances et approches comparatives
 - Vers une prédiction ab-initio 3D

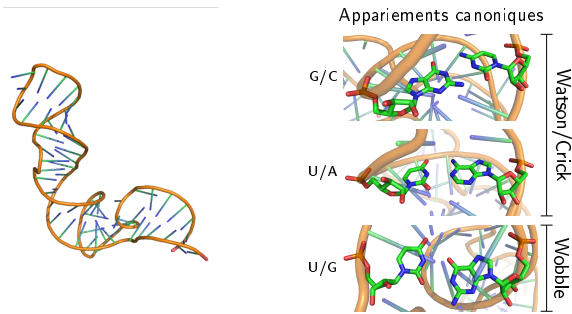
Dogme fondamental de la biologie moléculaire



Dogme fondamental de la biologie moléculaire



ARN = Biopolymère composé de nucléotides A,C,G et U
 A : Adénosine, C : Cytosine, G : Guanine et U : Uracile

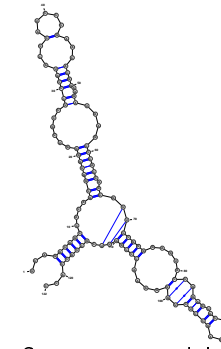


Repliement de l'ARN = Processus stochastique continu dirigé par (résultant en) un appariement des nucléotides.

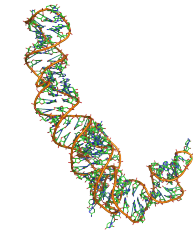
Comprendre le repliement des ARN aide à comprendre et prédire leur fonction.

Trois¹ niveaux de représentation :

```
UUAGGCGGCCACAGC
GGUGGGGUUGCCUCC
CGUACCCAUCCCGAA
CACGGAAGAUAGCC
CACCAGGUUCCGGG
GAGUACUGGAGUGCG
CGAGCCUCUGGAAA
CCGGUUGCCGCCA
CC
```



Structure primaire



Structure secondaire

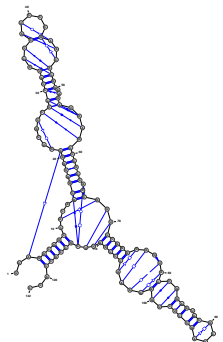
Structure tertiaire

Source : 5s rRNA (PDB 1K73 :B)

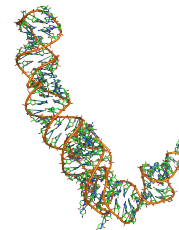
1. Enfin, presque ...

Trois¹ niveaux de représentation :

```
UUAGGCGGCCACAGC
GGUGGGGUUGCCUCC
CGUACCCAUCCCGAA
CACGGAAGAUAGCC
CACCAGGUUCCGGG
GAGUACUGGAGUGCG
CGAGCCUCUGGAAA
CCGGUUGCCGCCA
CC
```



Structure primaire

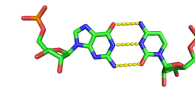


Structure secondaire⁺

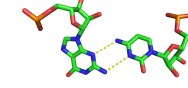
Structure tertiaire

Source : 5s rRNA (PDB 1K73 :B)

- Appariements non-canoniques
 Toute paire de base autre que {(A-U), (C-G), (G-U)}
 Ou interagissant sur un bord non-standard (WC/WC-Cis) [LW01].

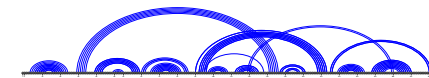


Paire CG canonique (WC/WC-Cis)



Paire CG non canonique (Sucre/WC-Trans)

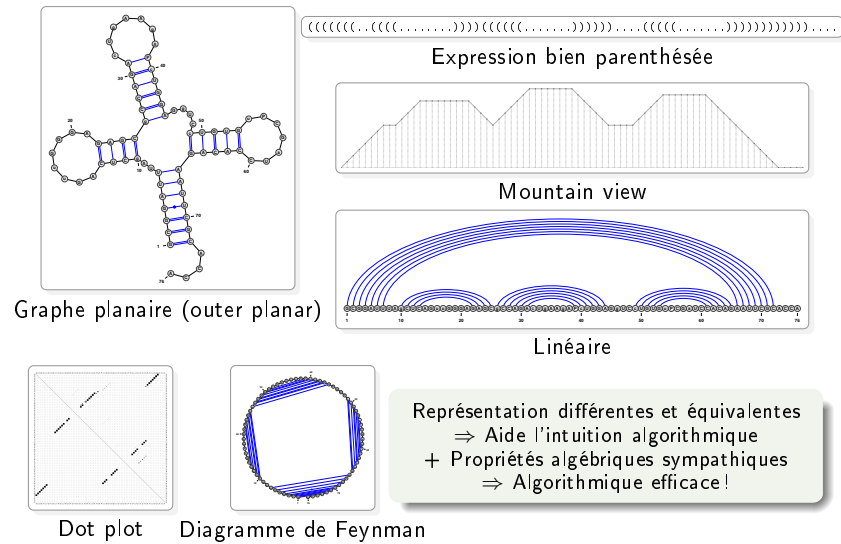
- Pseudonoeuds



Structure pseudonoeud d'un Ribozyme du Groupe I (PDBID : 1Y0Q :A)

Plus expressif, mais repliement général *in silico* avec pseudonoeud :
 ⇒ NP-Complet [LP00] ... polynomial pour certaines classes [CDR⁺04].

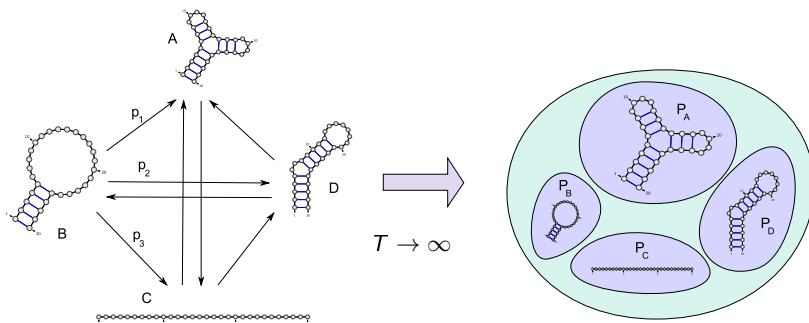
1. Enfin, presque ...



- 1 Introduction
 - Fonction(s) de l'ARN
 - Repliement et structure
 - Représentations de la structure secondaire
- 2 Formalisation du repliement et outils disponibles
 - Aparté thermodynamique
 - Programmation dynamique : Rappels
- 3 Minimisation de l'énergie libre
 - Modèle de Nussinov
 - Modèle de Turner
 - MFold/Unafold
 - Performances et approches comparatives
 - Vers une prédiction ab-initio 3D

Aparté thermodynamique

A l'échelle nanoscopique, la structure de l'ARN *fluctue*.



Convergence vers une **distribution stationnaire** de probabilité, l'équilibre de Boltzmann, où la probabilité est exponentiellement faible sur l'énergie libre.
Corollaire : La conformation initiale est sans d'importance.

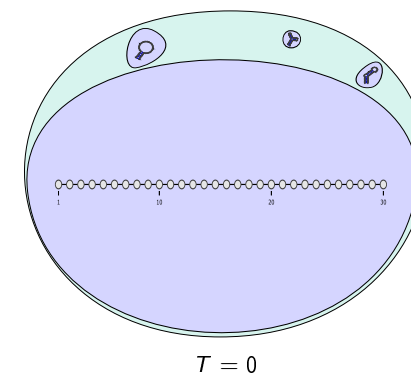
Problèmes soulevés :

Étant donnés des modèles pour l'ensemble des conformations et l'énergie libre.

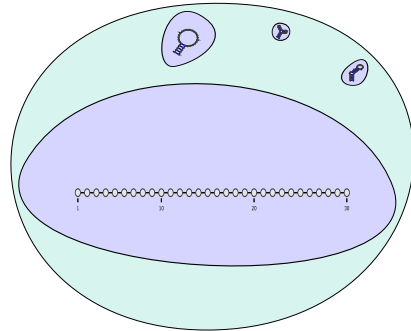
- A. Déterminer la structure la plus probable (= Energie libre minimale) à l'équilibre
- B. Déterminer des propriétés moyennes de l'ensemble de Boltzmann

Hors de l'équilibre

Transcription : ARN synthétisé sans appariement (Sauf exception)

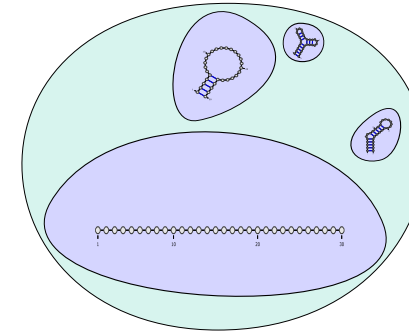


Transcription : ARN synthétisé sans appariement (Sauf exception)



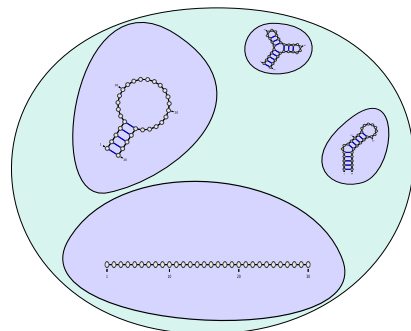
$T = 1h$

Transcription : ARN synthétisé sans appariement (Sauf exception)



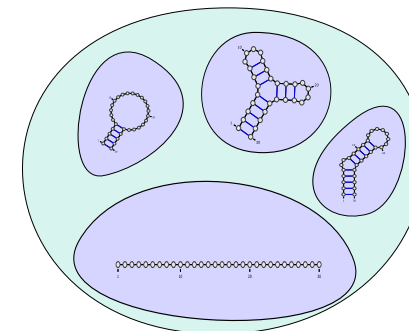
$T = 2h$

Transcription : ARN synthétisé sans appariement (Sauf exception)



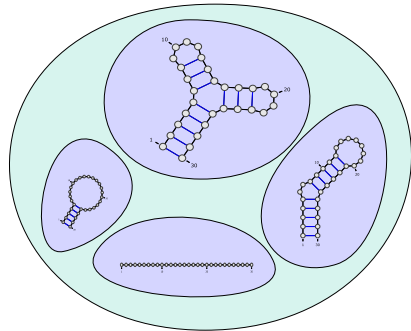
$T = 5h$

Transcription : ARN synthétisé sans appariement (Sauf exception)



$T = 10h$

Transcription : ARN synthétisé sans appariement (Sauf exception)



$T \rightarrow \infty$

Mais majorité des ARNm dégradés avant 7h (Org. : Souris [SSN⁺09]).

Résumé

1 Introduction

- Fonction(s) de l'ARN
- Repliement et structure
- Représentations de la structure secondaire

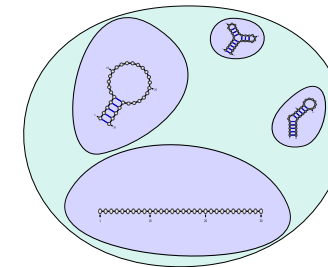
2 Formalisation du repliement et outils disponibles

- Aparté thermodynamique
- Programmation dynamique : Rappels

3 Minimisation de l'énergie libre

- Modèle de Nussinov
- Modèle de Turner
- MFold/Unafold
- Performances et approches comparatives
- Vers une prédiction ab-initio 3D

Transcription : ARN synthétisé sans appariement (Sauf exception)



$T = 10h$

Mais majorité des ARNm dégradés avant 7h (Org. : Souris [SSN⁺09]).

- Déterminer la structure la plus probable (= Energie libre min.) à l'équilibre
- Déterminer des propriétés moyennes de l'ensemble de Boltzmann
- Déterminer la structure la plus probable à temps T .
(c.f. H. Isambert par simulation, NP-complet en déterministe [MTSC09])

Programmation dynamique : Principe général

Programmation dynamique = Technique générale pour l'optimisation.

Condition : Solution optimale pour P peut être reconstruite à partir de solutions pour des sous-problèmes strictes de P .

Bioinformatique :

Espaces de solutions *discrets* (alignements, repliements)

+ Fonctions Objectif *additives* (score, énergie)

⇒ Schémas de programmation dynamique souvent efficaces.

Exemple : Alignement local (Smith/Waterman)

$$W(i, 0) = 0$$

$$W(0, j) = 0$$

$$W(i, j) = \max \begin{cases} W(i-1, j-1) + m_{i,j} \\ W(i-1, j) + p_i \\ W(i, j-1) + p_d \end{cases}$$

Un schéma fait intervenir des *classes* de sous-problèmes dont on sait calculer le score du *champion*.

Étant donné un schéma, deux étapes :

- **Calcul matrices** : Sauvegarde des meilleurs scores sur classes de sous-problèmes (Ordre inverse de celui induit par les dépendances).
- **Remontée** : Reconstitue le parcours ayant mené au meilleur score. (Parcours = Instance)

Complexité du calcul dépend alors :

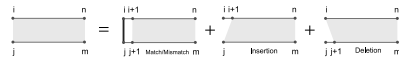
- **Taille** de l'espace des sous-problèmes
- **Nombres** de sous-problèmes considérés (#Termes décomposition)

Exemple S/W :

$$i : 1 \rightarrow n + 1 \Rightarrow \Theta(n)$$

$$j : 1 \rightarrow m + 1 \Rightarrow \Theta(m)$$

Trois opération pour chaque sous-calcul
 $\Rightarrow \Theta(m.n)$ temps/mémoire



Exemple : Aligement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

$$W(i, 0) = 0$$

$$W(0, j) = 0$$

$$W(i, j) = \max \begin{cases} W(i-1, j-1) + m_{i,j} \\ W(i-1, j) + p_i \\ W(i, j-1) + p_d \end{cases}$$

		A	C	A	C	A	C	T	A
	0	0	0	0	0	0	0	0	0
A	0								
G	0								
C	0								
A	0								
C	0								
A	0								
C	0								
A	0								

Exemple : Aligement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

$$W(i, 0) = 0$$

$$W(0, j) = 0$$

$$W(i, j) = \max \begin{cases} W(i-1, j-1) + m_{i,j} \\ W(i-1, j) + p_i \\ W(i, j-1) + p_d \end{cases}$$

		A	C	A	C	A	C	T	A
	0	0	0	0	0	0	0	0	0
A	0	2							
G	0								
C	0								
A	0								
C	0								
A	0								
C	0								
A	0								

Note: A red arrow points from the cell (A, A) at (1,1) to the cell (A, C) at (1,2).

Exemple : Aligement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

$$W(i, 0) = 0$$

$$W(0, j) = 0$$

$$W(i, j) = \max \begin{cases} W(i-1, j-1) + m_{i,j} \\ W(i-1, j) + p_i \\ W(i, j-1) + p_d \end{cases}$$

		A	C	A	C	A	C	T	A
	0	0	0	0	0	0	0	0	0
A	0	2	1						
G	0								
C	0								
A	0								
C	0								
A	0								
C	0								
A	0								

Note: Red arrows indicate the path from (A, A) at (1,1) to (A, C) at (1,2) and then to (A, C) at (2,3).

Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

		A	C	A	C	A	C	T	A
	0	0	0	0	0	0	0	0	0
A	0	2	1	2					
G	0								
C	0								
A	0								
C	0								
A	0								
C	0								
A	0								

Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

		A	C	A	C	A	C	T	A
	0	0	0	0	0	0	0	0	0
A	0	2	1	2	1				
G	0								
C	0								
A	0								
C	0								
A	0								
C	0								
A	0								

Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

		A	C	A	C	A	C	T	A
	0	0	0	0	0	0	0	0	0
A	0	2	1	2	1	2	1	0	2
G	0								
C	0								
A	0								
C	0								
A	0								
C	0								
A	0								

Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

		A	C	A	C	A	C	T	A
	0	0	0	0	0	0	0	0	0
A	0	2	1	2	1	2	1	0	2
G	0	1	1	1	1	1	1	0	1
C	0								
A	0								
C	0								
A	0								
C	0								
A	0								

Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

		A	C	A	C	A	C	T	A
0		0	0	0	0	0	0	0	0
A	0	2	1	2	1	2	1	0	2
G	0	1	1	1	1	1	1	0	1
C	0	0	3	2	3	2	3	2	1
A	0								
C	0								
A	0								
C	0								
A	0								

Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

		A	C	A	C	A	C	T	A
0		0	0	0	0	0	0	0	0
A	0	2	1	2	1	2	1	0	2
G	0	1	1	1	1	1	1	0	1
C	0	0	3	2	3	2	3	2	1
A	0	2	2	5	4	5	4	3	4
C	0								
A	0								
C	0								
A	0								

Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

		A	C	A	C	A	C	T	A
0		0	0	0	0	0	0	0	0
A	0	2	1	2	1	2	1	0	2
G	0	1	1	1	1	1	1	0	1
C	0	0	3	2	3	2	3	2	1
A	0	2	2	5	4	5	4	3	4
C	0	1	4	4	7	6	7	6	5
A	0	2	3	6	6	9	8	7	8
C	0	1	4	5	8	8	11	10	9
A	0	2	3	6	7	10	10	10	12

Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

		A	C	A	C	A	C	T	A
0		0	0	0	0	0	0	0	0
A	0	2	1	2	1	2	1	0	2
G	0	1	1	1	1	1	1	0	1
C	0	0	3	2	3	2	3	2	1
A	0	2	2	5	4	5	4	3	4
C	0	1	4	4	7	6	7	6	5
A	0	2	3	6	6	9	8	7	8
C	0	1	4	5	8	8	11	10	9
A	0	2	3	6	7	10	10	10	12

Exemple complet

Exemple : Alignement local de séquences AGCACACA et ACACACTA

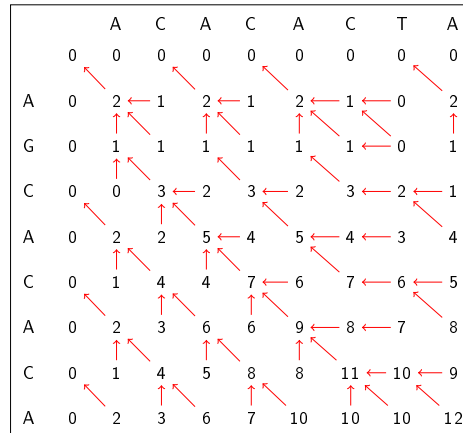
Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

Meilleur alignement



Exemple complet

Exemple : Alignement local de séquences AGCACACA et ACACACTA

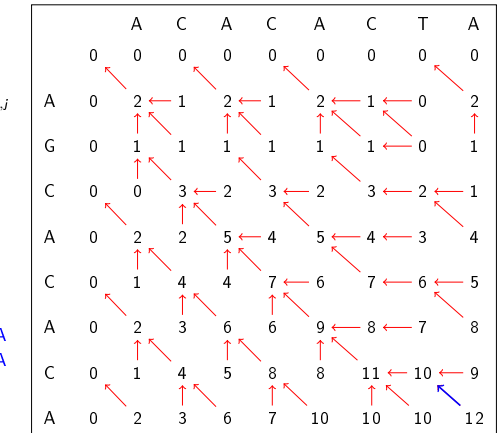
Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

Meilleur alignement



Exemple complet

Exemple : Alignement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

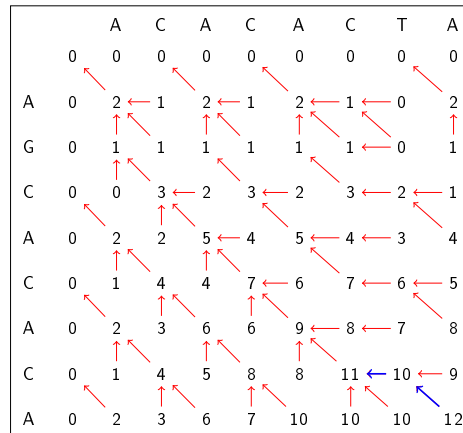
$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

Meilleur alignement

- A
T A



Exemple complet

Exemple : Alignement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

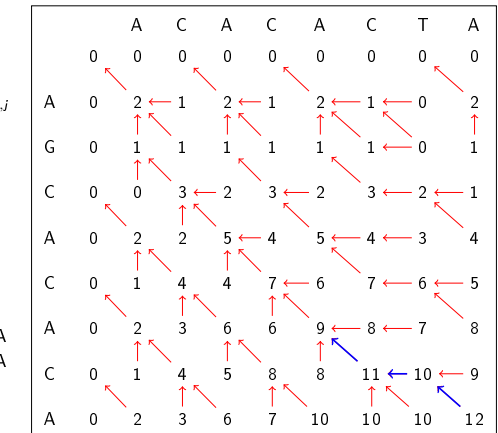
$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

Meilleur alignement

C - A
C T A



Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

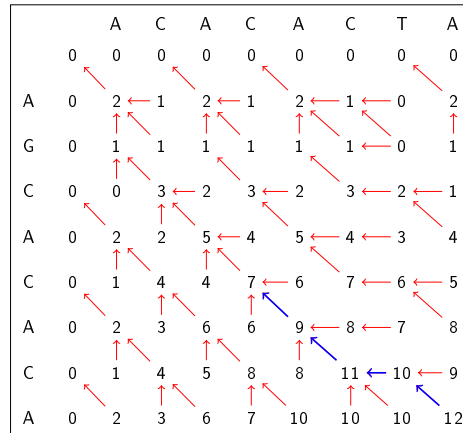
$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

Meilleur alignement

A C - A
A C T A



Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

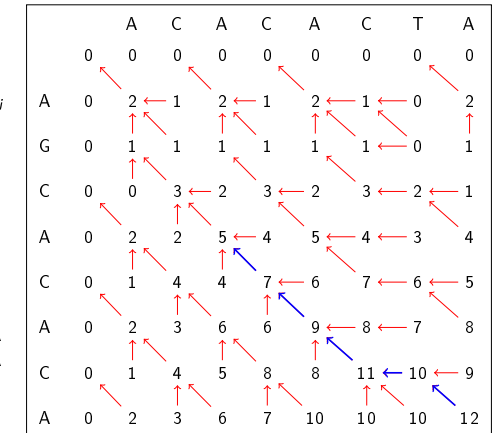
$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

Meilleur alignement

C A C - A
C A C T A



Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

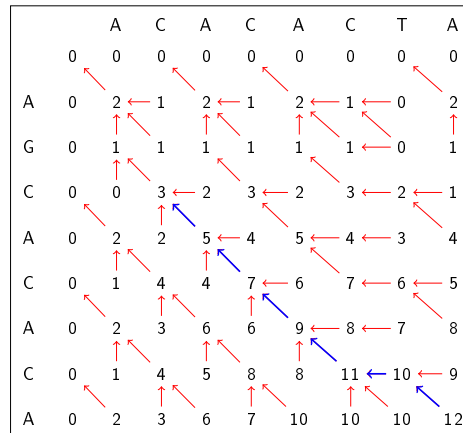
$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

Meilleur alignement

A C A C - A
A C A C T A



Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

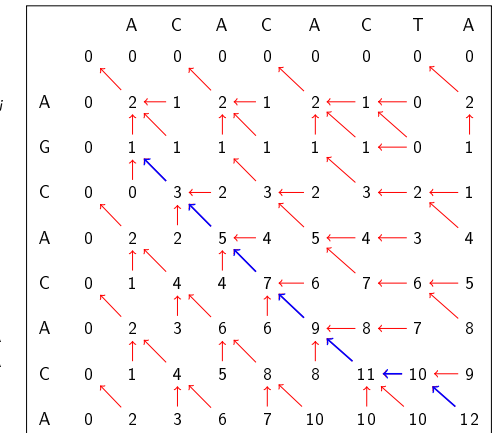
$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

Meilleur alignement

C A C A C - A
C A C A C T A



Exemple complet

Exemple : Alignement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

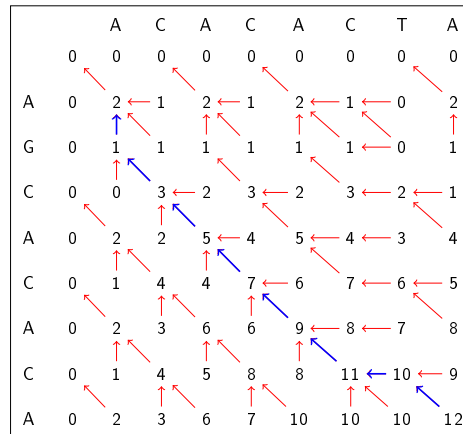
$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

Meilleur alignement

G C A C A C - A
- C A C A C T A



Exemple complet

Exemple : Alignement local de séquences AGCACACA et ACACACTA

Coûts : Match $m_{i,j} = +2$, Insertion/Déletion $p_i = p_j = -1$

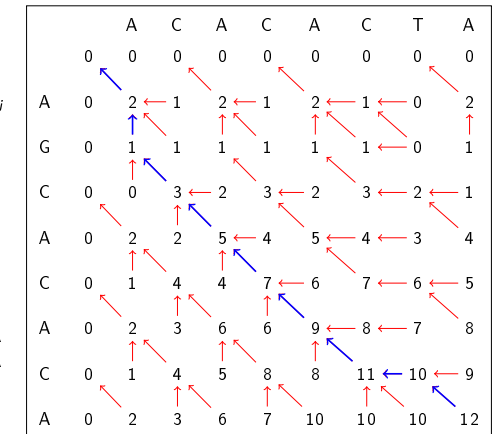
$$W(i,0) = 0$$

$$W(0,j) = 0$$

$$W(i,j) = \max \begin{cases} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{cases}$$

Meilleur alignement

A G C A C A C - A
A - C A C A C T A



Propriétés des schémas

Propriétés requise d'un schéma :

- **Validité** : \forall sous-problème, la valeur obtenue doit être celle de la fonction objectif.

Preuve souvent assez technique.

Propriétés souhaitables d'un schéma :

- **Complétude** : Espace des solutions engendré par la décomposition. Des astuces algorithmiques peuvent *couper des branches*...
- **Non-ambiguïté** : Chaque solution est *engendrée* au plus une fois.

⇒ Possibilité d'énumérer l'espace des solutions.

Résumé

1 Introduction

- Fonction(s) de l'ARN
- Repliement et structure
- Représentations de la structure secondaire

2 Formalisation du repliement et outils disponibles

- Aparté thermodynamique
- Programmation dynamique : Rappels

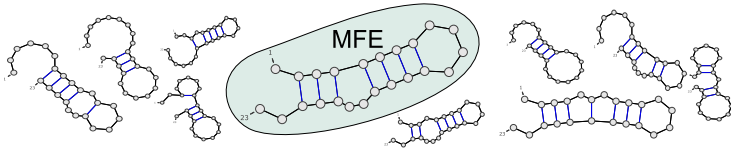
3 Minimisation de l'énergie libre

- Modèle de Nussinov
- Modèle de Turner
- MFold/Unafold
- Performances et approches comparatives
- Vers une prédiction ab-initio 3D

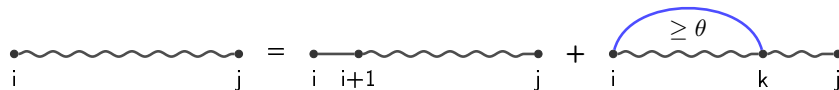
Problème A : Déterminer la structure d'énergie minimale.

Repliement *ab initio* =

Trouver structure d'un ARN ω uniquement à partir de sa séquence.



- **Conformations** : Ensemble S_ω des structures secondaires compatibles avec la structure primaire ω (contrainte d'appariements).
 - **Fonction d'énergie** Énergie libre associant une valeur numérique $E_{\omega,S}$ (KCal.mol^{-1}) à tout couple séquence/conformation (ω, S) .
 - **Structure native** : Conformation *fonctionnelle* de la molécule.
- Remarques :
- Pas nécessairement unique (Cinétique ou structures bi-stables)
 - Présence de pseudo-noeuds : Ambiguïté, quelle est la structure native?



$$N_{i,t} = 0, \quad \forall t \in [i, i + \theta]$$

$$N_{i,j} = \min \begin{cases} N_{i+1,j} & i \text{ non apparié} \\ \min_{k=i+\theta+1}^j E_{i,k} + N_{i+1,k-1} + N_{k+1,j} & i \text{ apparié à } k \end{cases}$$

Modèle de Nussinov/Jacobson (NJ)

Plus *proche voisins* simple :

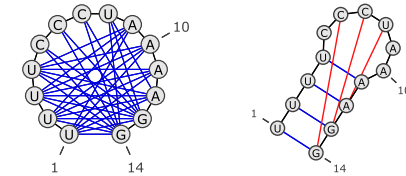
- Seuls les appariements contribuent à l'énergie
- Uniquement liaisons Watson/Crick (A/U, C/G) et Wobble (G/U)

$$\Rightarrow E_{\omega,S} = -\#Paires(S)$$

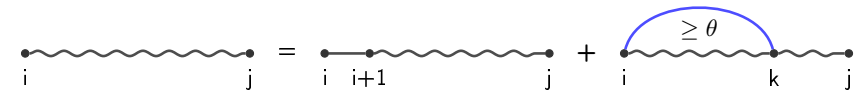
Repliement dans NJ \Leftrightarrow Maximisation du nombre de paires de bases.

Exemple :

UUUUCUUAAAAGG



Variante : Pondérer les paires selon leur nombre de liaisons hydrogène

$$\Delta G(G \equiv C) = -3 \quad \Delta G(A = U) = -2 \quad \Delta G(G - U) = -1$$


$$N_{i,t} = 0, \quad \forall t \in [i, i + \theta]$$

$$N_{i,j} = \min \begin{cases} N_{i+1,j} & i \text{ non apparié} \\ \min_{k=i+\theta+1}^j E_{i,k} + N_{i+1,k-1} + N_{k+1,j} & i \text{ apparié à } k \end{cases}$$

Correction : On cherche à montrer que l'énergie de la structure d'énergie la plus faible ($MFE_{1,n}$) est bien calculée dans $N_{1,n}$. Dans toute structure secondaire restreinte à $[i, j]$ la première position i est :

- Soit *non-appariée* : $MFE_{i,j}$ est constituée des appariements de $MFE_{i+1,j}$.
- Soit *appariée à k* : $MFE_{i,j}$ contient l'appariement (i, k) et l'union des appariements de $MFE_{i+1,k-1}$ et de $MFE_{k+1,j}$. En effet, tout appariement entre les régions $[i + 1, k - 1]$ et $[k + 1, j]$ *croiserait* (i, k) (Pseudonoed).

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	2	5	5	5	8	8	8
U								0	0	0	0	2	3	5	5	6	7	7
U									0	0	0	2	3	5	5	5	7	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	2	2	2	3	3
U												0	0	0	0	1	2	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0

Diagram illustrating the decomposition of a sequence from index i to j into a sub-sequence from i to $i+1$ and a sub-sequence from i to k to j , with a gap of at least θ between $i+1$ and k .

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	2	5	5	5	8	8	8
U								0	0	0	0	2	3	5	5	6	7	7
U									0	0	0	2	3	5	5	5	7	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	2	2	2	3	3
U												0	0	0	0	1	2	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0

Diagram illustrating the decomposition of a sequence from index i to j into a sub-sequence from i to $i+1$ and a sub-sequence from i to k to j , with a gap of at least θ between $i+1$ and k .

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	2	5	5	5	8	8	8
U								0	0	0	0	2	3	5	5	6	7	7
U									0	0	0	2	3	5	5	5	7	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	2	2	2	3	3
U												0	0	0	0	1	2	2
A														0	0	0	0	0
G															0	0	0	0
A																0	0	0
C																	0	0
G																		0
A																		0

Diagram illustrating the decomposition of a sequence from index i to j into a sub-sequence from i to $i+1$ and a sub-sequence from i to k to j , with a gap of at least θ between $i+1$ and k .

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	2	5	5	5	8	8	8
U								0	0	0	0	2	3	5	5	6	7	7
U									0	0	0	2	3	5	5	5	7	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	2	2	2	3	3
U												0	0	0	0	1	2	2
A														0	0	0	0	0
G															0	0	0	0
A																0	0	0
C																	0	0
G																		0
A																		0

Diagram illustrating the decomposition of a sequence from index i to j into a sub-sequence from i to $i+1$ and a sub-sequence from i to k to j , with a gap of at least θ between $i+1$ and k .

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	(.)	.	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	6	8	10	10	10	10	
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10	
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8	8	
C							0	0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	2	3	5	5	6	7	7	
U									0	0	0	2	3	5	5	5	7	7	
C										0	0	0	0	3	3	3	5	5	
U											0	0	0	2	2	2	3	3	
U												0	0	0	0	0	1	2	
A													0	0	0	0	0	0	
G														0	0	0	0	0	
A															0	0	0	0	
C																0	0	0	
G																	0	0	
A																		0	

$i \dots j = i \dots i+1 \dots j + i \dots k \dots j$ (with $\geq \theta$ loop)

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	(.)	.	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	6	8	10	10	10	10	
A				0	0	0	2	2	2	2	2	2	4	5	7	7	8	10	
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8	8	
C							0	0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	0	2	3	5	5	6	7	
U									0	0	0	0	2	3	5	5	6	7	
C										0	0	0	0	2	3	3	5	5	
U											0	0	0	2	2	2	3	3	
U												0	0	0	0	0	1	2	
A													0	0	0	0	0	0	
G														0	0	0	0	0	
A															0	0	0	0	
C																0	0	0	
G																	0	0	
A																		0	

$i \dots j = i \dots i+1 \dots j + i \dots k \dots j$ (with $\geq \theta$ loop)

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	(.)	.	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	6	8	10	10	10	10	
A				0	0	0	2	2	2	2	2	2	4	5	7	7	8	10	
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8	8	
C							0	0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	2	3	5	5	6	7	7	
U									0	0	0	2	3	5	5	5	7	7	
C										0	0	0	0	3	3	3	5	5	
U											0	0	0	2	2	2	3	3	
U												0	0	0	0	0	1	2	
A													0	0	0	0	0	0	
G														0	0	0	0	0	
A															0	0	0	0	
C																0	0	0	
G																	0	0	
A																		0	

$i \dots j = i \dots i+1 \dots j + i \dots k \dots j$ (with $\geq \theta$ loop)

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	(.)	.	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	6	8	10	10	10	10	
A				0	0	0	2	2	2	2	2	2	4	5	7	7	8	10	
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8	8	
C							0	0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	0	2	3	5	5	6	7	
U									0	0	0	0	2	3	5	5	6	7	
C										0	0	0	0	2	3	3	5	5	
U											0	0	0	2	2	2	3	3	
U												0	0	0	0	0	1	2	
A													0	0	0	0	0	0	
G														0	0	0	0	0	
A															0	0	0	0	
C																0	0	0	
G																	0	0	
A																		0	

$i \dots j = i \dots i+1 \dots j + i \dots k \dots j$ (with $\geq \theta$ loop)

	C	G	G	A	U	A	C	U	U	C	U	A	G	A	C	G	A	
	(.)	.	
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	2	3	5	5	6	7	
U									0	0	0	2	3	5	5	5	7	
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0

$i \dots j = i \dots i+1 \dots k \dots j$ with $\geq \theta$ between $i+1$ and k .

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(.)	.	
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	2	2	2	2	2	2	4	4	5	7	7	8
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	6	7
C										0	0	0	0	2	3	5	5	7
U											0	0	0	0	3	3	3	5
U												0	0	0	0	2	2	3
A													0	0	0	0	0	1
A														0	0	0	0	0
G															0	0	0	0
A																0	0	0
C																	0	0
G																		0
A																		

$i \dots j = i \dots i+1 \dots k \dots j$ with $\geq \theta$ between $i+1$ and k .

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(.)	.	
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	2	2	2	2	2	2	4	4	5	7	7	8
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	2	3	5	5	6	7	
U									0	0	0	2	3	5	5	5	7	
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0

$i \dots j = i \dots i+1 \dots k \dots j$ with $\geq \theta$ between $i+1$ and k .

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(.)	.	
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	2	2	2	2	2	2	4	4	5	7	7	8
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	6	7
C										0	0	0	0	2	3	5	5	7
U											0	0	0	0	3	3	3	5
U												0	0	0	0	2	2	3
A													0	0	0	0	0	1
A														0	0	0	0	0
G															0	0	0	0
A																0	0	0
C																	0	0
G																		0
A																		

$i \dots j = i \dots i+1 \dots k \dots j$ with $\geq \theta$ between $i+1$ and k .

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	(.)	.	.	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10	
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10	
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8	8	
C							0	0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	2	3	5	5	6	7	7	
U									0	0	0	2	3	5	5	5	7	7	
C										0	0	0	0	3	3	3	5	5	
U											0	0	0	2	2	2	3	3	
U												0	0	0	0	0	1	2	
A													0	0	0	0	0	0	
G														0	0	0	0	0	
A															0	0	0	0	
C																0	0	0	
G																	0	0	
A																		0	

$$DP[i][j] = \max \left(DP[i][i+1] + DP[i+1][j], \max_{i < k < j} (DP[i][k] + DP[k][j]) \right)$$

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	(.)	.	.	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10	
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10	
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8	8	
C							0	0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	0	2	3	5	5	6	7	
U									0	0	0	0	2	3	5	5	5	7	
C										0	0	0	0	3	3	3	5	5	
U											0	0	0	2	2	2	3	3	
U												0	0	0	0	0	1	2	
A													0	0	0	0	0	0	
G														0	0	0	0	0	
A															0	0	0	0	
C																0	0	0	
G																	0	0	
A																		0	

$$DP[i][j] = \max \left(DP[i][i+1] + DP[i+1][j], \max_{i < k < j} (DP[i][k] + DP[k][j]) \right)$$

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	(.)	.	.	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10	
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10	
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8	8	
C							0	0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	2	3	5	5	6	7	7	
U									0	0	0	2	3	5	5	5	7	7	
C										0	0	0	0	3	3	3	5	5	
U											0	0	0	2	2	2	3	3	
U												0	0	0	0	0	1	2	
A													0	0	0	0	0	0	
G														0	0	0	0	0	
A															0	0	0	0	
C																0	0	0	
G																	0	0	
A																		0	

$$DP[i][j] = \max \left(DP[i][i+1] + DP[i+1][j], \max_{i < k < j} (DP[i][k] + DP[k][j]) \right)$$

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	(.)	.	.	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10	
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10	
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8	8	
C							0	0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	0	2	3	5	5	6	7	
U									0	0	0	0	2	3	5	5	5	7	
C										0	0	0	0	3	3	3	5	5	
U											0	0	0	2	2	2	3	3	
U												0	0	0	0	0	1	2	
A													0	0	0	0	0	0	
G														0	0	0	0	0	
A															0	0	0	0	
C																0	0	0	
G																	0	0	
A																		0	

$$DP[i][j] = \max \left(DP[i][i+1] + DP[i+1][j], \max_{i < k < j} (DP[i][k] + DP[k][j]) \right)$$

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	((.	.	.))	.	
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	6	8	10	10	10	10
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0

Diagram illustrating the decomposition of a sequence from index i to j into a subsequence from i to $i+1$ and a subsequence from i to k to j , where the gap between $i+1$ and k is at least θ .

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	((.	.	.))	.	
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	6	8	10	10	10	10
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0

Diagram illustrating the decomposition of a sequence from index i to j into a subsequence from i to $i+1$ and a subsequence from i to k to j , where the gap between $i+1$ and k is at least θ .

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	((.	.	.))	.	
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	6	8	10	10	10	10
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0

Diagram illustrating the decomposition of a sequence from index i to j into a subsequence from i to $i+1$ and a subsequence from i to k to j , where the gap between $i+1$ and k is at least θ .

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	((.	.	.))	.	
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	6	8	10	10	10	10
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0

Diagram illustrating the decomposition of a sequence from index i to j into a subsequence from i to $i+1$ and a subsequence from i to k to j , where the gap between $i+1$ and k is at least θ .

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	((.	.	.))	.	
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C							0	0	0	0	0	0	2	5	5	5	8	8
U								0	0	0	0	0	2	3	5	5	6	7
U									0	0	0	0	2	3	5	5	5	7
C										0	0	0	0	3	3	3	5	5
U											0	0	0	0	2	2	2	3
U												0	0	0	0	0	1	2
A													0	0	0	0	0	0
G														0	0	0	0	0
A															0	0	0	0
C																0	0	0
G																	0	0
A																		0

$i \dots j = i \dots i+1 \dots j + i \dots k \dots j$ (with a loop $\geq \theta$ between k and j)

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	((.	.	.))	.		
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10	
A				0	0	0	2	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8	8	
C							0	0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	0	2	3	5	5	6	7	
U									0	0	0	0	2	3	5	5	5	7	
C										0	0	0	0	3	3	3	5	5	
U											0	0	0	0	2	2	2	3	
U												0	0	0	0	0	1	2	
A													0	0	0	0	0	0	
G														0	0	0	0	0	
A															0	0	0	0	
C																0	0	0	
G																	0	0	
A																		0	

$i \dots j = i \dots i+1 \dots j + i \dots k \dots j$ (with a loop $\geq \theta$ between k and j)

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	((.	.	.))	.		
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10	
A				0	0	0	2	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8	8	
C							0	0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	0	2	3	5	5	6	7	
U									0	0	0	0	2	3	5	5	5	7	
C										0	0	0	0	3	3	3	5	5	
U											0	0	0	0	2	2	2	3	
U												0	0	0	0	0	1	2	
A													0	0	0	0	0	0	
G														0	0	0	0	0	
A															0	0	0	0	
C																0	0	0	
G																	0	0	
A																		0	

$i \dots j = i \dots i+1 \dots j + i \dots k \dots j$ (with a loop $\geq \theta$ between k and j)

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	((.	.	.))	.		
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10	
A				0	0	0	2	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8	8	
C							0	0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	0	2	3	5	5	6	7	
U									0	0	0	0	2	3	5	5	5	7	
C										0	0	0	0	3	3	3	5	5	
U											0	0	0	0	2	2	2	3	
U												0	0	0	0	0	1	2	
A													0	0	0	0	0	0	
G														0	0	0	0	0	
A															0	0	0	0	
C																0	0	0	
G																	0	0	
A																		0	

$i \dots j = i \dots i+1 \dots j + i \dots k \dots j$ (with a loop $\geq \theta$ between k and j)

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	((.	.	.)	.	(.)))	.		
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10	
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10	
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8	8	
C							0	0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	0	2	3	5	5	6	7	
U									0	0	0	0	2	3	5	5	5	7	
C										0	0	0	0	3	3	3	5	5	
U											0	0	0	0	2	2	2	3	
U												0	0	0	0	0	1	2	
A													0	0	0	0	0	0	
G														0	0	0	0	0	
A															0	0	0	0	
C																0	0	0	
G																	0	0	
A																		0	

$i \dots j = i \dots i+1 + i \dots k + k \dots j$ (with $\geq \theta$ condition)

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	((.	.	.)	.	(.)))	.		
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10	
A				0	0	0	2	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8	8	
C							0	0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	0	2	3	5	5	6	7	
U									0	0	0	0	2	3	5	5	5	7	
C										0	0	0	0	3	3	3	5	5	
U											0	0	0	0	2	2	2	3	
U												0	0	0	0	0	1	2	
A													0	0	0	0	0	0	
G														0	0	0	0	0	
A															0	0	0	0	
C																0	0	0	
G																	0	0	
A																		0	

$i \dots j = i \dots i+1 + i \dots k + k \dots j$ (with $\geq \theta$ condition)

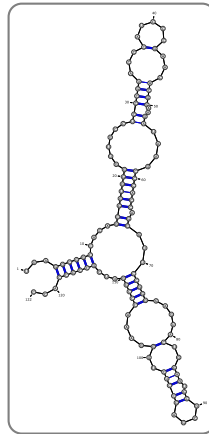
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	((.	.	.)	.	(.)))	.		
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10	
A				0	0	0	2	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8	8	
C							0	0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	0	2	3	5	5	6	7	
U									0	0	0	0	2	3	5	5	5	7	
C										0	0	0	0	3	3	3	5	5	
U											0	0	0	0	2	2	2	3	
U												0	0	0	0	0	1	2	
A													0	0	0	0	0	0	
G														0	0	0	0	0	
A															0	0	0	0	
C																0	0	0	
G																	0	0	
A																		0	

$i \dots j = i \dots i+1 + i \dots k + k \dots j$ (with $\geq \theta$ condition)

	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	((.	.	.)	.	(.)))	.		
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10	
A				0	0	0	2	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8	8	
C							0	0	0	0	0	0	2	5	5	5	8	8	
U								0	0	0	0	0	2	3	5	5	6	7	
U									0	0	0	0	2	3	5	5	5	7	
C										0	0	0	0	3	3	3	5	5	
U											0	0	0	0	2	2	2	3	
U												0	0	0	0	0	1	2	
A													0	0	0	0	0	0	
G														0	0	0	0	0	
A															0	0	0	0	
C																0	0	0	
G																	0	0	
A																		0	

$i \dots j = i \dots i+1 + i \dots k + k \dots j$ (with $\geq \theta$ condition)

Basée sur décomposition non-ambiguë en boucles de la structure 2^{aire} :



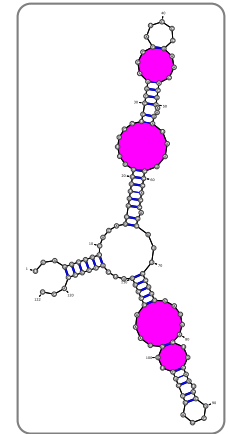
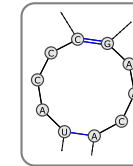
Énergies libres ΔG des boucles dépendent des bases, asymétrie, bases *libres* (dangle) ...

Déterminées expérimentalement
+ Interpolation pour grandes boucles.

Meilleure résultats grâce à la prise en compte de l'empilement.

Basée sur décomposition non-ambiguë en boucles de la structure 2^{aire} :

- Boucles internes



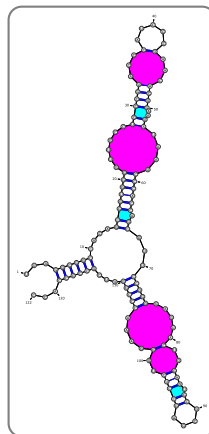
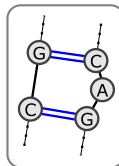
Énergies libres ΔG des boucles dépendent des bases, asymétrie, bases *libres* (dangle) ...

Déterminées expérimentalement
+ Interpolation pour grandes boucles.

Meilleure résultats grâce à la prise en compte de l'empilement.

Basée sur décomposition non-ambiguë en boucles de la structure 2^{aire} :

- Boucles internes
- Renflements



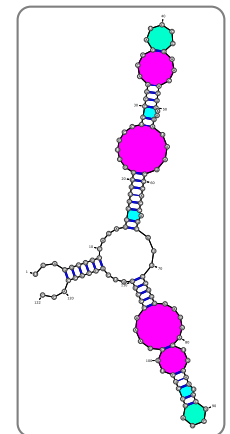
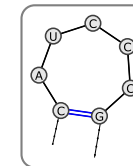
Énergies libres ΔG des boucles dépendent des bases, asymétrie, bases *libres* (dangle) ...

Déterminées expérimentalement
+ Interpolation pour grandes boucles.

Meilleure résultats grâce à la prise en compte de l'empilement.

Basée sur décomposition non-ambiguë en boucles de la structure 2^{aire} :

- Boucles internes
- Renflements
- Boucles terminales



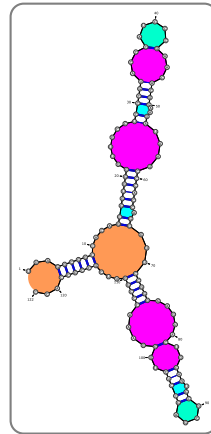
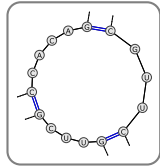
Énergies libres ΔG des boucles dépendent des bases, asymétrie, bases *libres* (dangle) ...

Déterminées expérimentalement
+ Interpolation pour grandes boucles.

Meilleure résultats grâce à la prise en compte de l'empilement.

Basée sur décomposition **non-ambiguë** en **boucles** de la structure 2^{aire} :

- Boucles internes
- Renflements
- Boucles terminales
- Boucles multiples



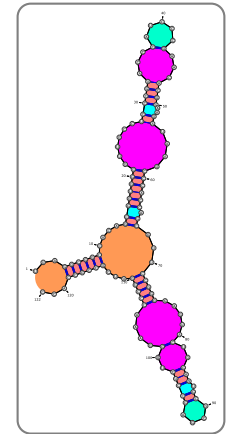
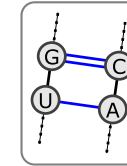
Énergies libres ΔG des boucles dépendent des bases, assymétrie, bases *libres* (dangle) ...

Déterminées expérimentalement
+ Interpolation pour grandes boucles.

Meilleure résultats grâce à la prise en compte de l'empilement.

Basée sur décomposition **non-ambiguë** en **boucles** de la structure 2^{aire} :

- Boucles internes
- Renflements
- Boucles terminales
- Boucles multiples
- Empilements

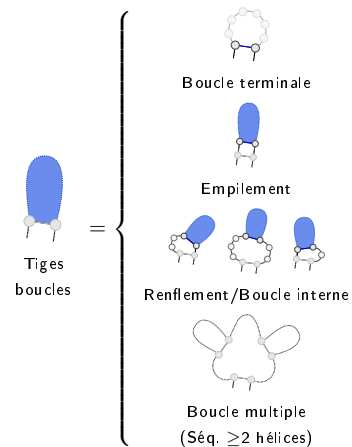


Énergies libres ΔG des boucles dépendent des bases, assymétrie, bases *libres* (dangle) ...

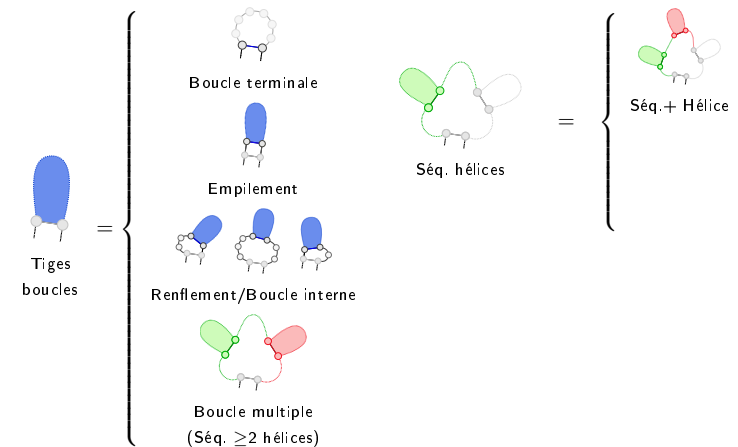
Déterminées expérimentalement
+ Interpolation pour grandes boucles.

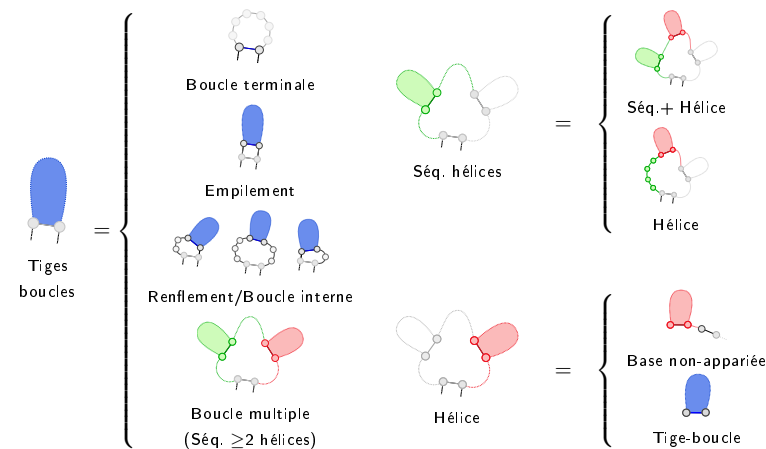
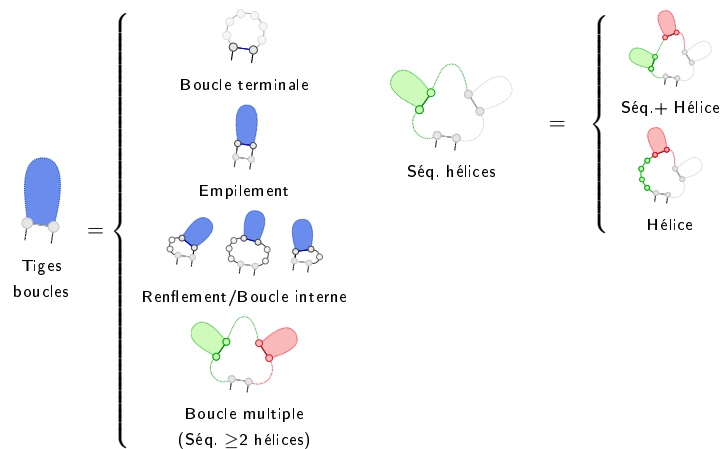
Meilleure résultats grâce à la prise en compte de l'empilement.

MFE DP equations



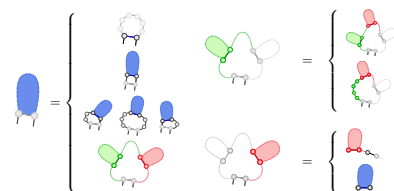
MFE DP equations





MFold Unafold

- $E_H(i, j)$: Energie de boucle terminale *fermée* par une paire (i, j)
- $E_{BI}(i, j)$: Energie de renflement ou boucle interne *fermée* par une paire (i, j)
- $E_S(i, j)$: Energie d'empilement $(i, j)/(i + 1, j - 1)$
- a, c, b : Pénalité de boucle multiple, hélice et non-appariées dans multiboucle.



Calcul des matrices

$$\begin{aligned}
 \mathcal{M}'_{i,j} &= \min \left\{ \begin{array}{l} E_H(i, j) \\ E_S(i, j) + \mathcal{M}'_{i+1, j-1} \\ \text{Min}_{i', j'} (E_{BI}(i, i', j', j) + \mathcal{M}'_{i', j'}) \\ a + c + \text{Min}_k (\mathcal{M}_{i+1, k-1} + \mathcal{M}^1_{k, j-1}) \end{array} \right\} \\
 \mathcal{M}_{i,j} &= \text{Min}_k \{ \min (\mathcal{M}_{i, k-1}, b(k-1)) + \mathcal{M}^1_{k, j} \} \\
 \mathcal{M}^1_{i,j} &= \text{Min}_k \{ b + \mathcal{M}^1_{i, j-1}, c + \mathcal{M}'_{i, j} \}
 \end{aligned}$$

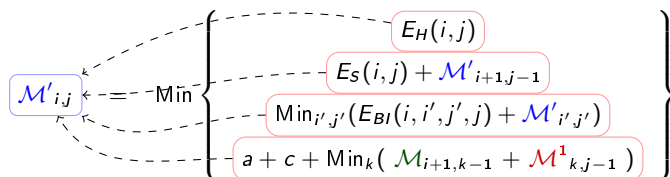
Remontée (Backtracking)

Reconstruction de la structure d'énergie minimale :

$$\begin{aligned}
 \mathcal{M}'_{i,j} &= \text{Min} \left\{ \begin{array}{l} E_H(i, j) \\ E_S(i, j) + \mathcal{M}'_{i+1, j-1} \\ \text{Min}_{i', j'} (E_{BI}(i, i', j', j) + \mathcal{M}'_{i', j'}) \\ a + c + \text{Min}_k (\mathcal{M}_{i+1, k-1} + \mathcal{M}^1_{k, j-1}) \end{array} \right\} \\
 \mathcal{M}_{i,j} &= \text{Min}_k \{ \min (\mathcal{M}_{i, k-1}, b(k-1)) + \mathcal{M}^1_{k, j} \} \\
 \mathcal{M}^1_{i,j} &= \text{Min}_k \{ b + \mathcal{M}^1_{i, j-1}, c + \mathcal{M}'_{i, j} \}
 \end{aligned}$$

2. Avec une astuce pour les bulges/boucles internes ...

Reconstruction de la structure d'énergie minimale :

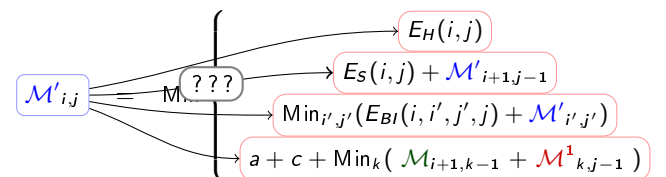


$$\mathcal{M}_{i,j} = \text{Min}_k \{ \min(\mathcal{M}_{i,k-1}, b(k-1)) + \mathcal{M}^1_{k,j} \}$$

$$\mathcal{M}^1_{i,j} = \text{Min}_k \{ b + \mathcal{M}^1_{i,j-1}, c + \mathcal{M}'_{i,j} \}$$

2. Avec une astuce pour les bulges/boucles internes ...

Reconstruction de la structure d'énergie minimale :



$$\mathcal{M}_{i,j} = \text{Min}_k \{ \min(\mathcal{M}_{i,k-1}, b(k-1)) + \mathcal{M}^1_{k,j} \}$$

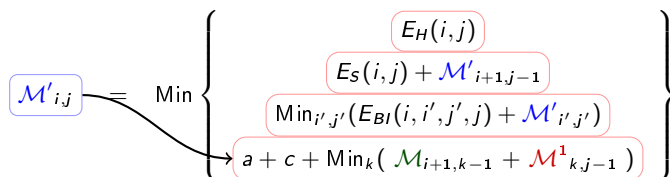
$$\mathcal{M}^1_{i,j} = \text{Min}_k \{ b + \mathcal{M}^1_{i,j-1}, c + \mathcal{M}'_{i,j} \}$$

$\mathcal{O}(n)$ contributeurs potentiels au Min :

⇒ Complexité au pire en $\mathcal{O}(n^2)$ pour un backtrack naïf.

2. Avec une astuce pour les bulges/boucles internes ...

Reconstruction de la structure d'énergie minimale :



$$\mathcal{M}_{i,j} = \text{Min}_k \{ \min(\mathcal{M}_{i,k-1}, b(k-1)) + \mathcal{M}^1_{k,j} \}$$

$$\mathcal{M}^1_{i,j} = \text{Min}_k \{ b + \mathcal{M}^1_{i,j-1}, c + \mathcal{M}'_{i,j} \}$$

$\mathcal{O}(n)$ contributeurs potentiels au Min :

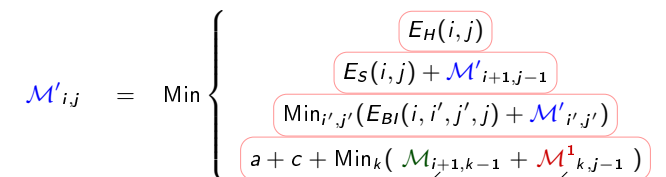
⇒ Complexité au pire en $\mathcal{O}(n^2)$ pour un backtrack naïf.

Garder les meilleures contributions aux Min ⇒ Backtrack en $\mathcal{O}(n)$

Complexités temps/mémoire en $\mathcal{O}(n^3)/\mathcal{O}(n^2)$ pour le précalcul²

2. Avec une astuce pour les bulges/boucles internes ...

Reconstruction de la structure d'énergie minimale :



$$\mathcal{M}_{i,j} = \text{Min}_k \{ \min(\mathcal{M}_{i,k-1}, b(k-1)) + \mathcal{M}^1_{k,j} \}$$

$$\mathcal{M}^1_{i,j} = \text{Min}_k \{ b + \mathcal{M}^1_{i,j-1}, c + \mathcal{M}'_{i,j} \}$$

$\mathcal{O}(n)$ contributeurs potentiels au Min :

⇒ Complexité au pire en $\mathcal{O}(n^2)$ pour un backtrack naïf.

Garder les meilleures contributions aux Min ⇒ Backtrack en $\mathcal{O}(n)$

Complexités temps/mémoire en $\mathcal{O}(n^3)/\mathcal{O}(n^2)$ pour le précalcul²

⇒ Unafold [MZ08] calcule la structure secondaire d'énergie minimale.

2. Avec une astuce pour les bulges/boucles internes ...

Definition (Replieement ab initio)

Partant de la séquence, trouver la conformation minimisant une fonction d'énergie.

Avantages :

- Explication mécanique
- Complexité raisonnable $O(n^3)/O(n^2)$ temps/mémoire
- Exploration exhaustive

Limites :

- Pas de cinétique
- Pas d'info évolutive
- Performances limitées

Definition (Approche comparative)

Partant de plusieurs séquences homologues ou d'un alignement, trouver une conformation de score (énergie+alignement) élevé.

Avantages :

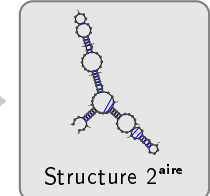
- Meilleures performances
- Affinement permanent

Limites :

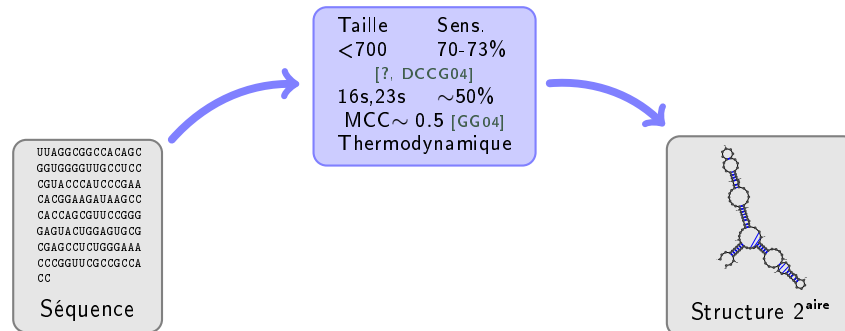
- Complexité élevée
- Exploration non-exhaustive

UUAGGGGGCCA GAGC
GGUGGGGUUGCCUCC
CGUA CCCAU CCGAAA
CACGGAAGAU AAGCC
CACGAGOGUUCGGGG
GAGUA CUGGAGTGGC
CGAGCCU CUGGGAAA
CCGGUUCGCGGCA
CC

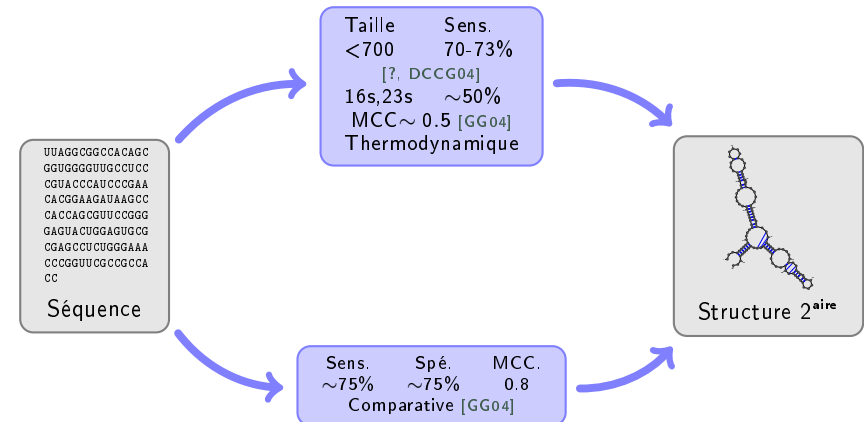
Séquence



Rappel : $MCC = \frac{t^+t^- - f^+f^-}{\sqrt{(t^++f^+)(t^++f^-)(t^-+f^+)(t^-+f^-)}}$



Rappel : $MCC = \frac{t^+t^- - f^+f^-}{\sqrt{(t^++f^+)(t^++f^-)(t^-+f^+)(t^-+f^-)}}$



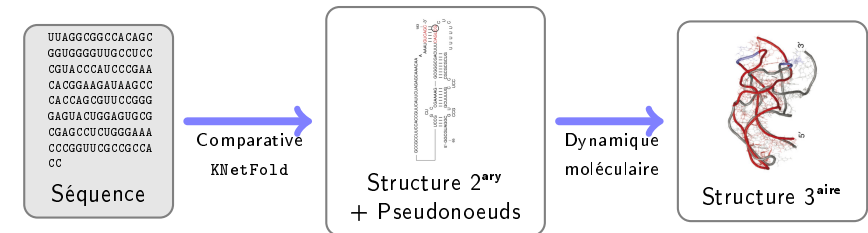
Rappel : $MCC = \frac{t^+t^- - f^+f^-}{\sqrt{(t^++f^+)(t^++f^-)(t^-+f^+)(t^-+f^-)}}$

But : De la séquence à des modèles tri-dimensionnels !!!



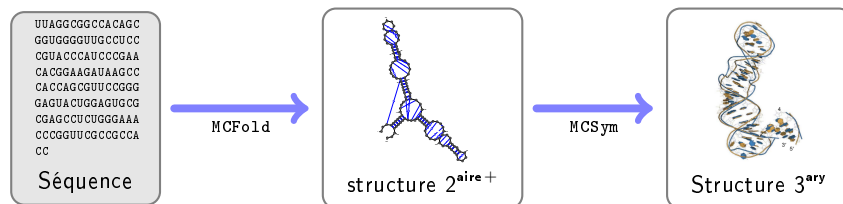
But : De la séquence à des modèles tri-dimensionnels !!!

- Modèles comparatifs + Dynamique moléculaires : RNA2D3D [SYKB07]



But : De la séquence à des modèles tri-dimensionnels !!!

- Pipeline MC-Fold/MC-sym [PM08]



A. Condon, B. Davy, B. Rastegari, S. Zhao, and F. Tarrant. **Classifying RNA pseudoknotted structures.** *Theoretical Computer Science*, 320(1) :35–50, 2004.

K. Doshi, J. J. Cannone, C. Cobaugh, and R. R. Gutell. **Evaluation of the suitability of free-energy minimization using nearest-neighbor energy parameters for rna secondary structure prediction.** *BMC Bioinformatics*, 5(1) :105, 2004.

P. Gardner and R. Giegerich. **A comprehensive comparison of comparative rna structure prediction approaches.** *BMC Bioinformatics*, 5(1) :140, 2004.

R. B. Lyngsø and C. N. S. Pedersen. **RNA pseudoknot prediction in energy-based models.** *Journal of Computational Biology*, 7(3-4) :409–427, 2000.

N. Leontis and E. Westhof. **Geometric nomenclature and classification of RNA base pairs.** *RNA*, 7 :499–512, 2001.

Ján Maňuch, Chris Thachuk, Ladislav Stacho, and Anne Condon. **Np-completeness of the direct energy barrier problem without pseudoknots.** pages 106–115, 2009.

N. R. Markham and M. Zuker. **Bioinformatics**, chapter UNA Fold, pages 3–31. Springer, 2008.

M. Parisien and F. Major. **The MC-Fold and MC-Sym pipeline infers RNA structure from sequence data.** *Nature*, 452(7183) :51–55, 2008.



Lioudmila V Sharova, Alexei A Sharov, Timur Nedorezov, Yulan Piao, Nabeebi Shaik, and Minoru S H Ko.
Database for mrna half-life of 19 977 genes obtained by dna microarray analysis of pluripotent and differentiating mouse embryonic stem cells.
DNA Res, 16(1) :45-58, Feb 2009.



B. A. Shapiro, Y. G. Yingling, W. Kasprzak, and E. Bindewald.
Bridging the gap in rna structure prediction.
Curr Opin Struct Biol, 17(2) :157-165, Apr 2007.

Exercice : Parsing/repliement des structures secondaires en
Python