

## Cours M2 BIBS - Séance 1

### Repliement *in silico* de l'ARN

Yann Ponty

Bioinformatics Team  
École Polytechnique/CNRS/INRIA AMIB - France

<http://www.lix.polytechnique.fr/~ponty/index.php?page=bibscasm2012>

9 Janvier 2012

## Avant propos ...

...ou comment gagner 1 million de dollars en rendant la monnaie !!

**Problème** : Vous disposez de pièces de **1**, **20** et **50** centimes. Le client souhaite minimiser la monnaie reçue (en nombre de pièces).  
Comment rendre **N** en monnaie sans perdre un client ?

**Stratégie 1** : Commencer par les *grosses* pièces puis compléter avec les *petites*.

$$21 = \text{50c} + \text{1c}$$

55??

## Avant propos ...

...ou comment gagner 1 million de dollars en rendant la monnaie !!

**Problème** : Vous disposez de pièces de **1**, **20** et **50** centimes. Le client souhaite minimiser la monnaie reçue (en nombre de pièces).  
Comment rendre **N** en monnaie sans perdre un client ?

**Stratégie 1** : Commencer par les *grosses* pièces puis compléter avec les *petites*.

$$21 = ??$$

## Avant propos ...

...ou comment gagner 1 million de dollars en rendant la monnaie !!

**Problème** : Vous disposez de pièces de **1**, **20** et **50** centimes. Le client souhaite minimiser la monnaie reçue (en nombre de pièces).  
Comment rendre **N** en monnaie sans perdre un client ?

**Stratégie 1** : Commencer par les *grosses* pièces puis compléter avec les *petites*.

$$21 = \text{50c} + \text{1c}$$

$$55 = \text{50c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c}$$

60??

... ou comment gagner 1 million de dollars en rendant la monnaie !!

**Problème :** Vous disposez de pièces de **1**, **20** et **50** centimes. Le client souhaite minimiser la monnaie reçue (en nombre de pièces). Comment rendre **N** en monnaie sans perdre un client ?

**Stratégie 1 :** Commencer par les *grosses* pièces puis compléter avec les *petites*.

$$\begin{aligned} 21 &= \text{50c} + \text{1c} \\ 55 &= \text{50c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} \\ 60 &= \text{50c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} ?? \end{aligned}$$

**Stratégie 2 :** Il existe une récurrence donnant le nombre minimal de pièce :

$$NbPieces(N) = \text{Min} \begin{cases} \text{1c} \rightarrow 1 + NbPieces(N - 1) \\ \text{20c} \rightarrow 1 + NbPieces(N - 20) \\ \text{50c} \rightarrow 1 + NbPieces(N - 50) \end{cases}$$

Avec un peu de mémoire ( $N$  résultats intermédiaires/cas à retenir), on peut alors répondre après  $N \times \#P$  pièces calculs.

**Remarque :** On n'a pas gagné le million, car  $N$  a une valeur exponentielle sur son codage. Cet algorithme est donc en temps exponentiel au regard de la théorie de la complexité.

Mais on a optimisé, en évitant un parcours exhaustif de l'arbre des possibles :  
 ⇒ Programmation dynamique.

... ou comment gagner 1 million de dollars en rendant la monnaie !!

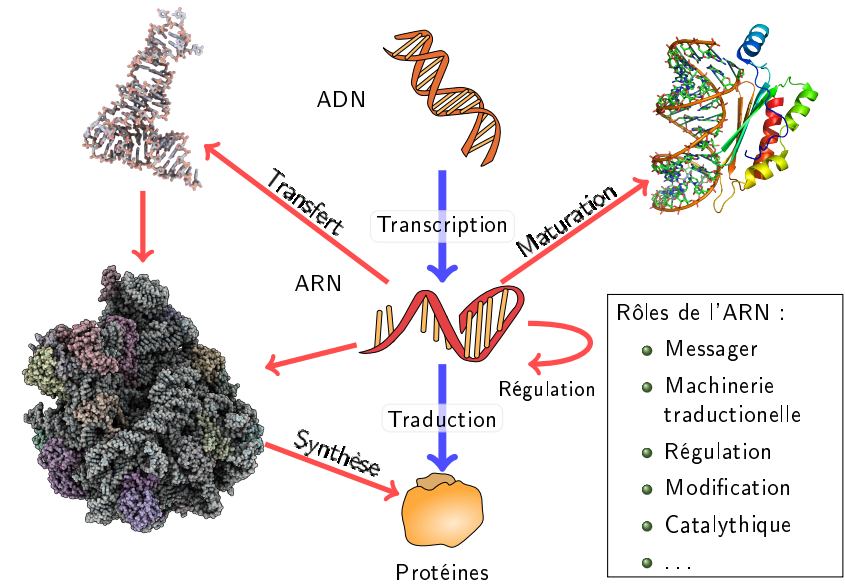
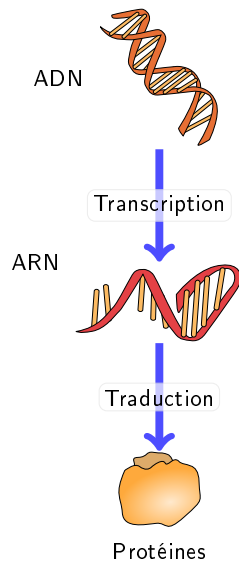
**Problème :** Vous disposez de pièces de **1**, **20** et **50** centimes. Le client souhaite minimiser la monnaie reçue (en nombre de pièces). Comment rendre **N** en monnaie sans perdre un client ?

**Stratégie 1 :** Commencer par les *grosses* pièces puis compléter avec les *petites*.

$$\begin{aligned} 21 &= \text{50c} + \text{1c} \\ 55 &= \text{50c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} \\ 60 &= \text{50c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} + \text{1c} ?? \\ &= \text{50c} + \text{20c} + \text{20c} ! \end{aligned}$$

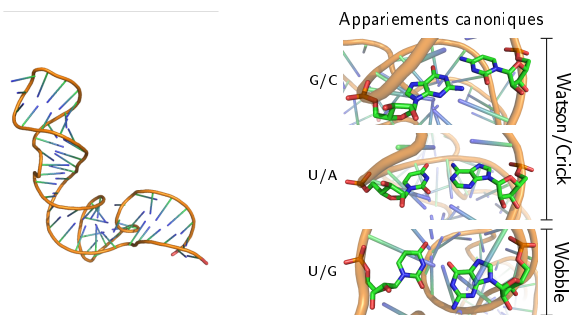
Problème *a priori* (?! ) non-résolvable en général par une approche *gloutonne* car problème plus simple NP-complet (Existe t il même une façon efficace de rendre la monnaie ? ⇒ 1M\$).

- 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



## Repliement de l'ARN

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.

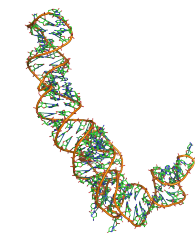
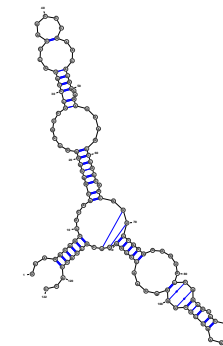
Comprendre le repliement des ARN aide à comprendre et prédire leur fonction.

## Structure de l'ARN

Trois<sup>1</sup> niveaux de représentation :

```
UUAGCGGCCACAGC
GGUGGGUUGCCUCC
CGUACCAUCCGAA
CACGGAAGAUAGCC
CACCAGGUUCCGGG
GAGUACUGGAGUGCG
CGAGCCUCUGGAAA
CCGGUUGCCGCCA
CC
```

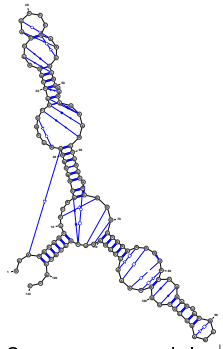
Structure primaire



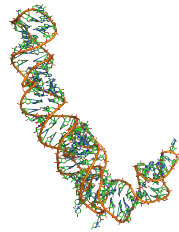
Source : 5s rRNA (PDB 1K73 :B)

Trois<sup>1</sup> niveaux de représentation :

```
UUAGGGGCCACAGC
GGUGGGGUUGCCUC
CGUACCCAUCCGAA
CACGGAAGAUAGCC
CACCAGGUUCCGGG
GAGUACUGGAGUGCG
CGAGCCUCUGGAAA
CCGGUUGCCGCCA
CC
```



Structure primaire



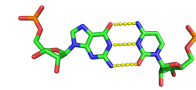
Structure secondaire<sup>+</sup>

Structure tertiaire

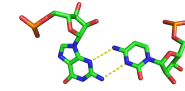
Source : 5s rRNA (PDB 1K73 :B)

1. Enfin, presque ...

- 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].

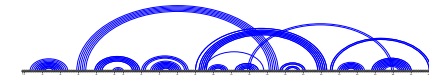


Paire CG canonique (WC/WC-Cis)



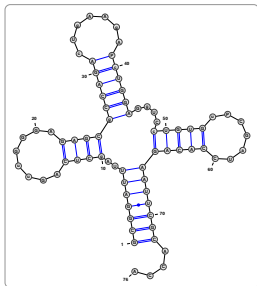
Paire CG non canonique (Sucre/WC-Trans)

- Pseudonoeuds



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

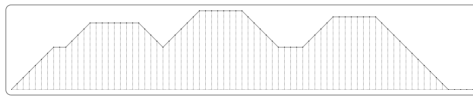
Plus expressif, mais repliement général *in silico* avec pseudonoeud :  
 ⇒ NP-Complet [LP00] ... polynomial pour certaines classes [CDR<sup>+</sup>04].



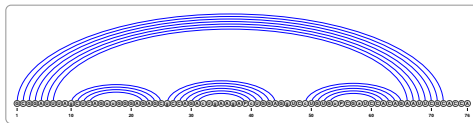
Graphe planaire (outer planar)

(((((((.....))))))((((.....)))).....((((.....)))))).....

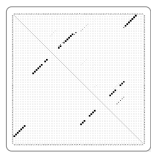
Expression bien parenthésée



Mountain view



Linéaire



Dot plot

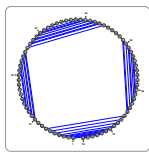
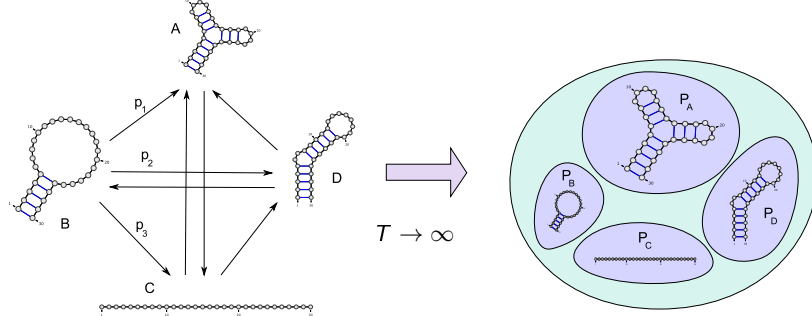


Diagramme de Feynman

Représentation différentes et équivalentes  
 ⇒ Aide l'intuition algorithmique  
 + Propriétés algébriques sympathiques  
 ⇒ Algorithmique efficace!

- 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

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



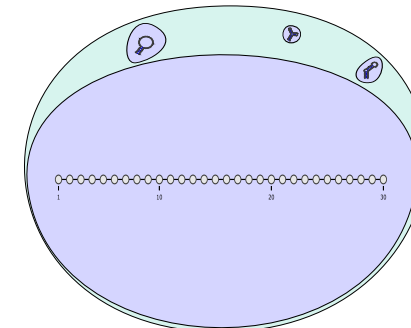
Convergence vers une **distribution stationnaire** de probabilité, l'équilibre de Boltzmann, où la probabilité est exponentiellement faible sur l'énergie libre.  
**Corollaire** : La conformation initiale est sans d'importance.

Problèmes soulevés :

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

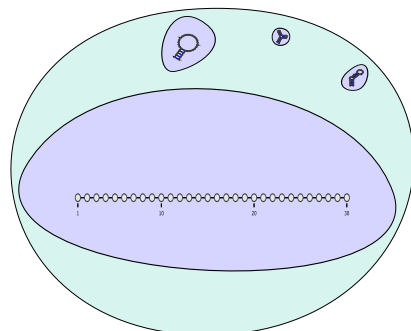
- A. Déterminer la structure la plus probable (= Energie libre minimale) à l'équilibre
- B. Déterminer des propriétés moyennes de l'ensemble de Boltzmann

Transcription : ARN synthétisé sans appariement (Sauf exception)



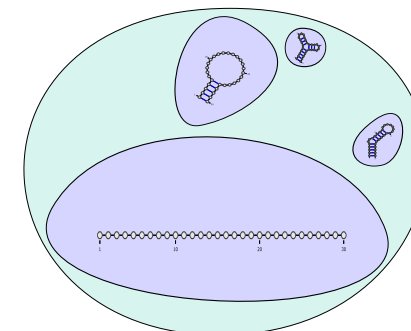
T = 0

Transcription : ARN synthétisé sans appariement (Sauf exception)



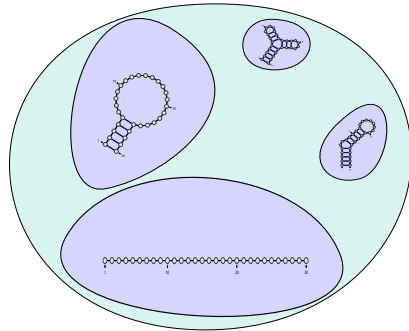
T = 1h

Transcription : ARN synthétisé sans appariement (Sauf exception)



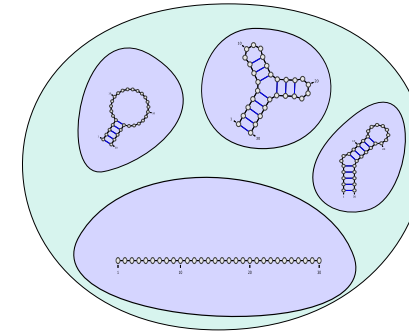
T = 2h

Transcription : ARN synthétisé sans appariement (Sauf exception)



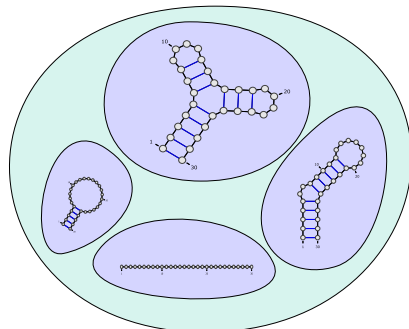
$T = 5h$

Transcription : ARN synthétisé sans appariement (Sauf exception)



$T = 10h$

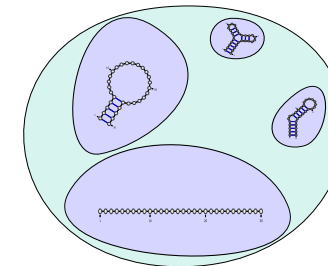
Transcription : ARN synthétisé sans appariement (Sauf exception)



$T \rightarrow \infty$

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

Transcription : ARN synthétisé sans appariement (Sauf exception)



$T = 10h$

Mais majorité des ARNm dégradés avant 7h (Org. : Souris [SSN<sup>+</sup>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])

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

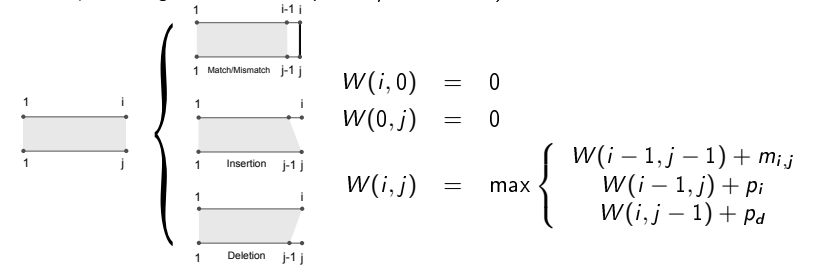
- Modèle de Nussinov
- Modèle de Turner
- MFold/Unafold
- Performances et approches comparatives
- Vers une prédiction ab-initio 3D

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 :

- Espace de solutions *discret* (alignements, repliements)
- + Fonction objectif *additive* (score, énergie)
- ⇒ Schéma de programmation dynamique efficace.

Exemple : Alignement local (Smith/Waterman)



Détails algorithmiques

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)

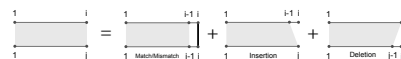
Exemple S/W :

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

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

Trois opérations pour chaque sous-calcul

⇒  $\Theta(m.n)$  temps/mémoire



Exemple complet

Exemple : Alignement local de séquences AGCACACA et ACACACTA

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

$W(i,0) = 0$

$W(0,j) = 0$

$W(i,j) = \max \left\{ \begin{array}{l} W(i-1,j-1) + m_{i,j} \\ W(i-1,j) + p_i \\ W(i,j-1) + p_d \end{array} \right.$

		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 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							
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						
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					
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

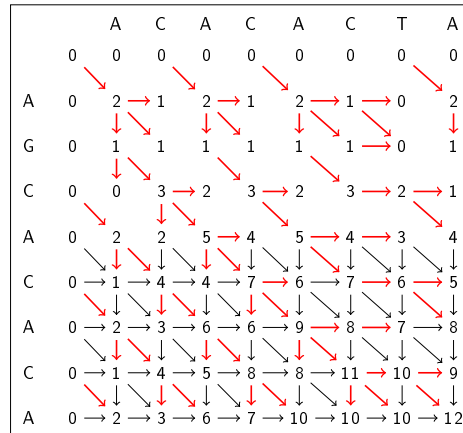
Exemple : Aligement local de séquences AGCACACA et ACACACTA

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

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

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

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



## Exemple complet

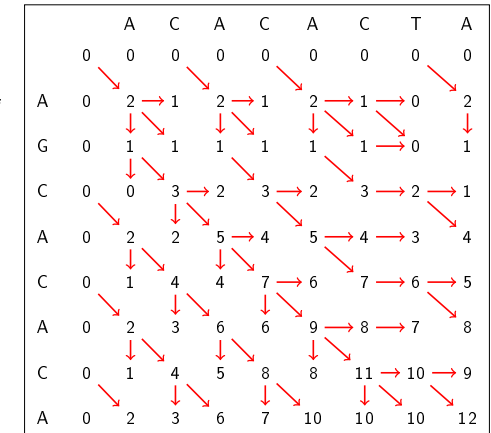
Exemple : Aligement local de séquences AGCACACA et ACACACTA

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

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

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

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



## Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

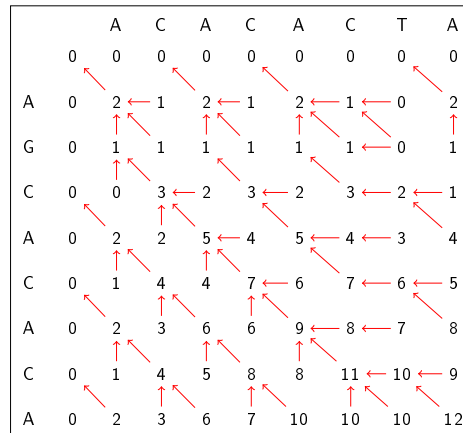
Coûts : Match  $m_{i,j} = +2$ , Insertion/Déletion  $p_i = p_j = -1$

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

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

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

Meilleur aligement



## Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

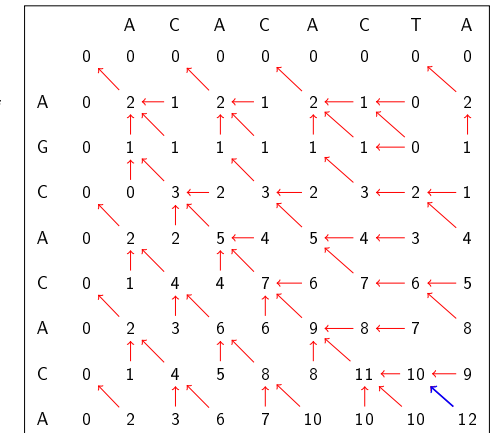
Coûts : Match  $m_{i,j} = +2$ , Insertion/Déletion  $p_i = p_j = -1$

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

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

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

Meilleur aligement



## Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

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

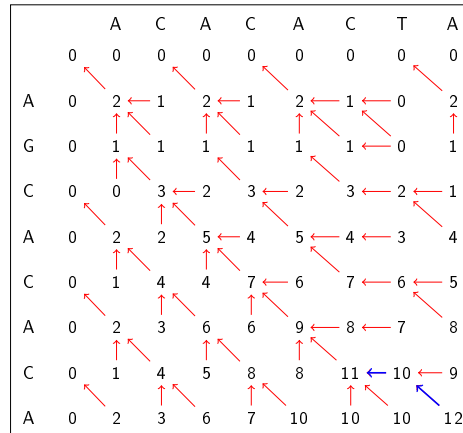
$$W(i,0) = 0$$

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

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

Meilleur alignement

- A  
T A



## Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

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

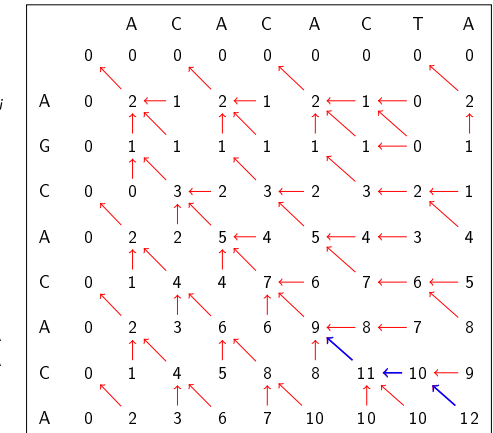
$$W(i,0) = 0$$

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

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

Meilleur alignement

C - A  
C T A



## Exemple complet

Exemple : Aligement local de séquences AGCACACA et ACACACTA

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

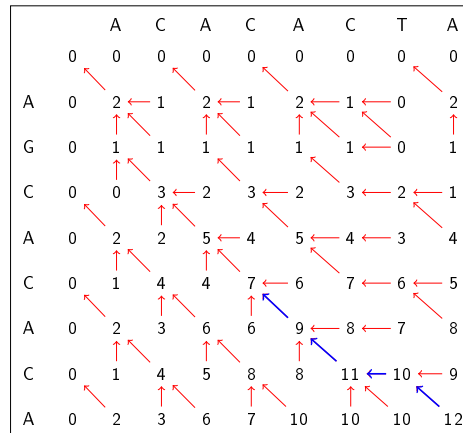
$$W(i,0) = 0$$

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

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

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  $p_i = p_j = -1$

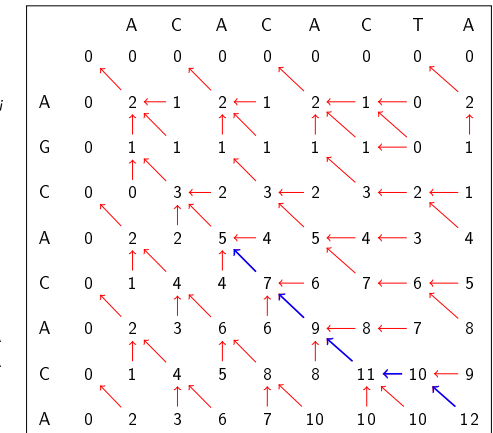
$$W(i,0) = 0$$

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

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

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  $p_i = p_j = -1$

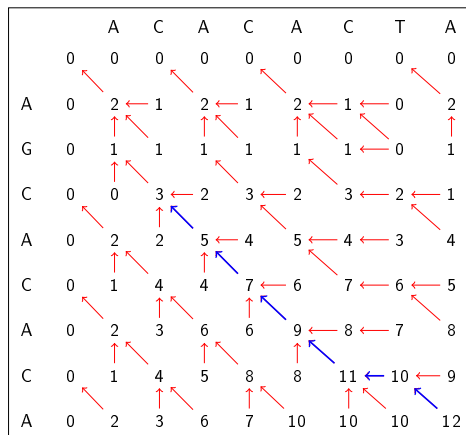
$$W(i,0) = 0$$

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

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

Meilleur aligement

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  $p_i = p_j = -1$

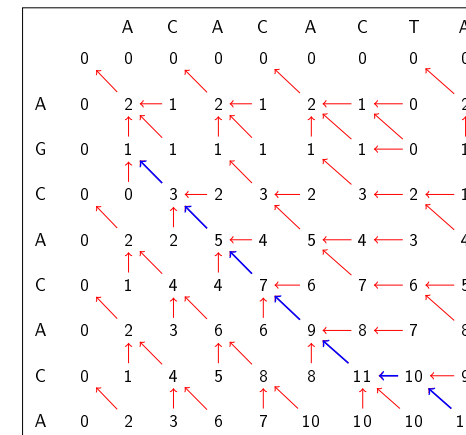
$$W(i,0) = 0$$

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

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

Meilleur aligement

C A C A 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  $p_i = p_j = -1$

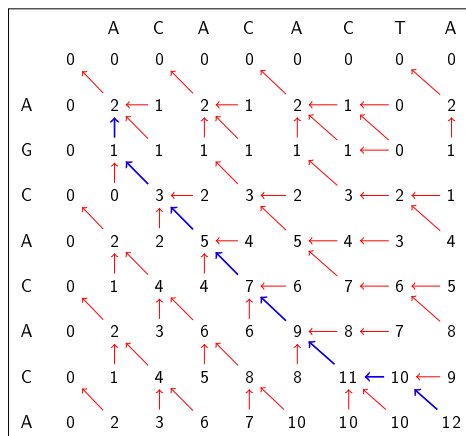
$$W(i,0) = 0$$

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

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

Meilleur aligement

G C A C A 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  $p_i = p_j = -1$

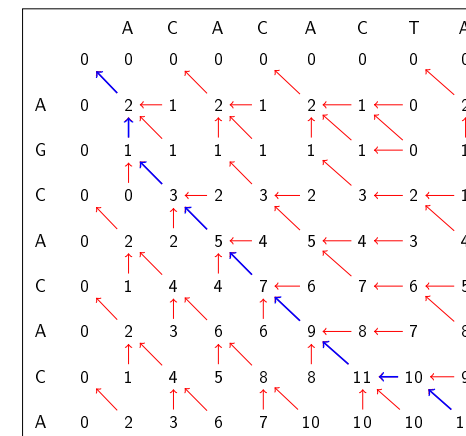
$$W(i,0) = 0$$

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

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

Meilleur aligement

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



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.

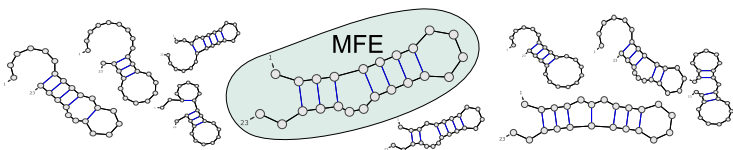
⇒ Possibilité d'énumérer l'espace des solutions.

## Repliement par minimisation d'énergie

Problème A : Déterminer la structure d'énergie minimale.

Repliement *ab initio* =

Trouver structure d'un ARN  $\omega$  uniquement à partir de sa séquence.



- **Conformations** : Ensemble  $S_\omega$  des structures secondaires compatibles avec la structure primaire  $\omega$  (contrainte d'appariements).
- **Fonction d'énergie** Énergie libre associant une valeur numérique  $E_{\omega,S}$  (KCal.mol<sup>-1</sup>) à tout couple séquence/conformation  $(\omega, S)$ .
- **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?

### 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

## Modèle de Nussinov/Jacobson

Modèle de Nussinov/Jacobson (NJ)

*Plus proche voisins* simple :

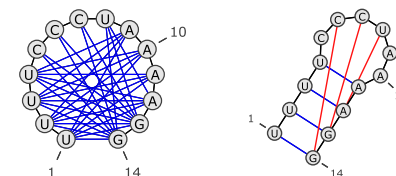
- Seuls les appariements contribuent à l'énergie
- Uniquement liaisons Watson/Crick (A/U, C/G) et Wobble (G/U)

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

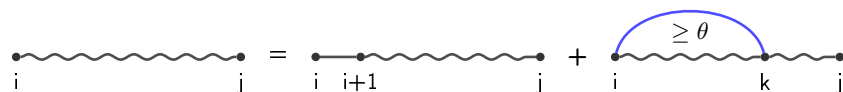
Repliement dans NJ  $\Leftrightarrow$  Maximisation du nombre de paires de bases.

Exemple :

UUUUCUUAAAAGG



**Variante** : Pondérer les paires selon leur nombre de liaisons hydrogène  
 $\Delta G(G \equiv C) = -3$      $\Delta G(A = U) = -2$      $\Delta G(G - U) = -1$

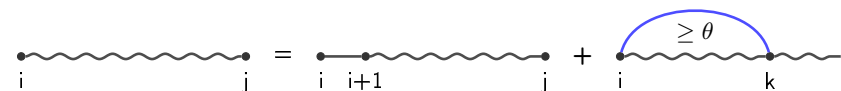


$$N_{i,t} = 0, \quad \forall t \in [i, i + \theta]$$

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

Nussinov/Jacobson

	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	2	2	4	5	7	7	8	10	10
A						0	0	0	0	2	2	2	5	5	5	8	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	7	7	7
C										0	0	0	3	3	3	5	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



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

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

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).

Nussinov/Jacobson

	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	2	2	4	5	7	7	8	10	10
A						0	0	0	0	2	2	2	5	5	5	8	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	7	7	7
C										0	0	0	3	3	3	5	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

	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	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

	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	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

	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	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

	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	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

	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	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	3	3	3	5	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

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

	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	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	6	7	7
C										0	0	0	3	3	3	5	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

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

	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	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	3	3	3	5	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

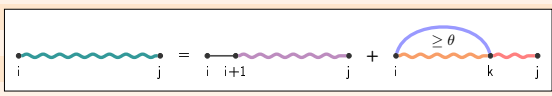
$i \dots j = i \dots i+1 \dots j + i \dots k \dots j$  (with a loop of size  $\geq \theta$ )

	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	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	6	7	7
C										0	0	0	3	3	3	5	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

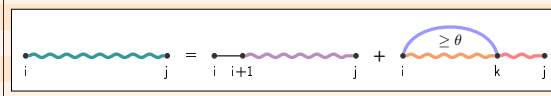
$i \dots j = i \dots i+1 \dots j + i \dots k \dots j$  (with a loop of size  $\geq \theta$ )



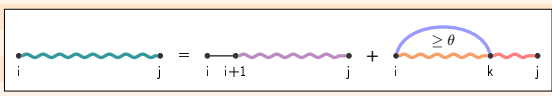
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C		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



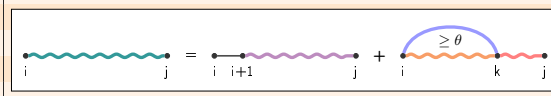
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	(	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	)	.
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10
A				0	0	0	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2	2	2	5	5	5	8	8
C		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



	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	
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



	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	
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



	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	2	5	5	5	8	8	8
U								0	0	0	0	2	3	5	5	6	7	7
U									0	0	0	0	2	3	5	5	7	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 sub-sequence from  $i$  to  $i+1$  and a sub-sequence from  $i$  to  $k$  to  $j$ , where the gap length is  $\geq \theta$ .

	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	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	7	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 sub-sequence from  $i$  to  $i+1$  and a sub-sequence from  $i$  to  $k$  to  $j$ , where the gap length is  $\geq \theta$ .

	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	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	7	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 sub-sequence from  $i$  to  $i+1$  and a sub-sequence from  $i$  to  $k$  to  $j$ , where the gap length is  $\geq \theta$ .

	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	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	7	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 sub-sequence from  $i$  to  $i+1$  and a sub-sequence from  $i$  to  $k$  to  $j$ , where the gap length is  $\geq \theta$ .

	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	0	2	3	5	5	7	7
C										0	0	0	0	3	3	5	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

$i \dots j = i \dots i+1 \dots k \dots j$  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	0	2	3	5	5	7	7
C										0	0	0	0	3	3	5	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

$i \dots j = i \dots i+1 \dots k \dots j$  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	0	2	3	5	5	7	7
C										0	0	0	0	3	3	5	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

$i \dots j = i \dots i+1 \dots k \dots j$  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	0	2	3	5	5	7	7
C										0	0	0	0	3	3	5	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

$i \dots j = i \dots i+1 \dots k \dots j$  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	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 sub-sequence from  $i$  to  $i+1$  and a sub-sequence from  $i$  to  $k$  to  $j$ , where the gap length is  $\geq \theta$ .

	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	

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  $j$ , where the gap length is  $\geq \theta$ .

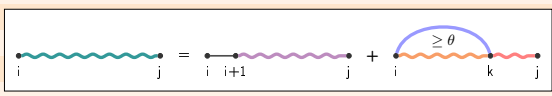
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	(	(	.	.	.	)	.	(	.	.	.	.	.	)	)	.			
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10	
A				0	0	0	2	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8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

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  $j$ , where the gap length is  $\geq \theta$ .

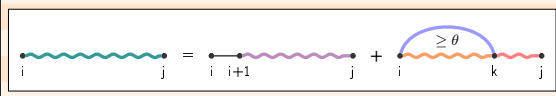
	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A	
	(	(	.	.	.	)	.	(	.	.	.	.	.	)	)	.			
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14	
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11	
G			0	0	0	0	3	3	3	5	5	5	5	6	8	10	10	10	
A				0	0	0	2	2	2	2	2	2	4	4	5	7	7	8	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10	
A						0	0	0	0	0	2	2	2	5	5	5	8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

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  $j$ , where the gap length is  $\geq \theta$ .

	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	4	4	5	7	7	8	10	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	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

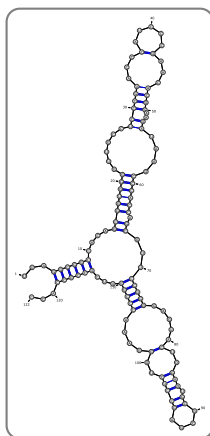


	C	G	G	A	U	A	C	U	U	C	U	U	A	G	A	C	G	A
	( ( ( . . . ) . ( ( . . . ) ) ) ) .																	
C	0	0	0	0	0	0	3	4	4	6	6	6	6	9	9	11	14	14
G		0	0	0	0	0	3	4	4	6	6	6	6	7	9	11	11	11
G			0	0	0	0	3	3	3	5	5	5	6	8	10	10	10	10
A				0	0	0	2	2	2	2	4	4	5	7	7	8	10	10
U					0	0	0	0	0	0	2	2	4	5	7	7	8	10
A						0	0	0	0	0	2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



### Modèle de Turner

Basée sur décomposition non-ambiguë en boucles de la structure 2<sup>aire</sup> :



Énergies libres  $\Delta G$  des boucles dépendent des bases, assymétrie, bases *libres* (dangle) ...

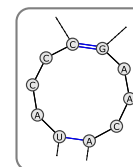
Déterminées expérimentalement  
+ Interpolation pour grandes boucles.

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

### Modèle de Turner

Basée sur décomposition non-ambiguë en boucles de la structure 2<sup>aire</sup> :

- Boucles internes



Énergies libres  $\Delta G$  des boucles dépendent des bases, assymétrie, bases *libres* (dangle) ...

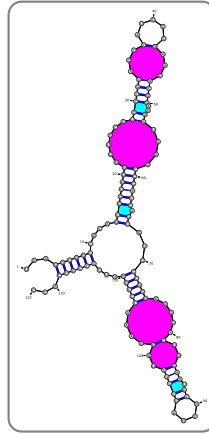
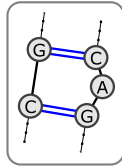
Déterminées expérimentalement  
+ Interpolation pour grandes boucles.

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

## Modèle de Turner

Basée sur décomposition *non-ambiguë* en *boucles* de la structure 2<sup>aire</sup> :

- Boucles internes
- Renflements



Énergies libres  $\Delta G$  des boucles dépendent des bases, assymétrie, bases *libres* (dangle) ...

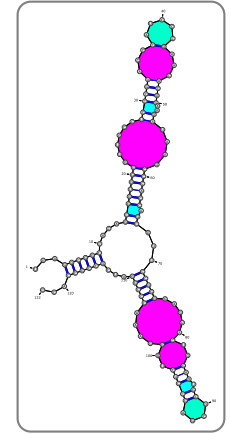
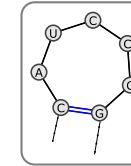
Déterminées expérimentalement  
+ Interpolation pour grandes boucles.

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

## Modèle de Turner

Basée sur décomposition *non-ambiguë* en *boucles* de la structure 2<sup>aire</sup> :

- Boucles internes
- Renflements
- Boucles terminales



Énergies libres  $\Delta G$  des boucles dépendent des bases, assymétrie, bases *libres* (dangle) ...

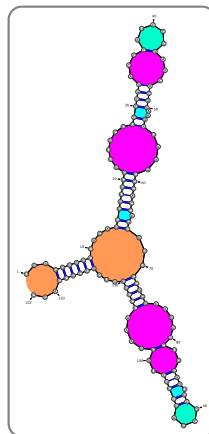
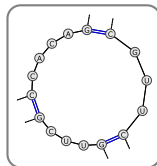
Déterminées expérimentalement  
+ Interpolation pour grandes boucles.

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

## Modèle de Turner

Basée sur décomposition *non-ambiguë* en *boucles* de la structure 2<sup>aire</sup> :

- Boucles internes
- Renflements
- Boucles terminales
- Boucles multiples



Énergies libres  $\Delta G$  des boucles dépendent des bases, assymétrie, bases *libres* (dangle) ...

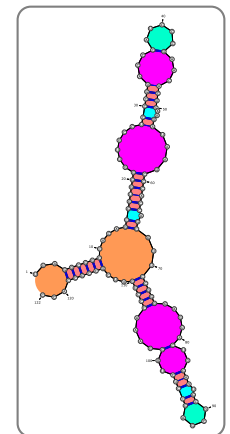
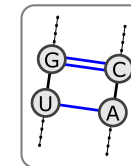
Déterminées expérimentalement  
+ Interpolation pour grandes boucles.

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

## Modèle de Turner

Basée sur décomposition *non-ambiguë* en *boucles* de la structure 2<sup>aire</sup> :

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

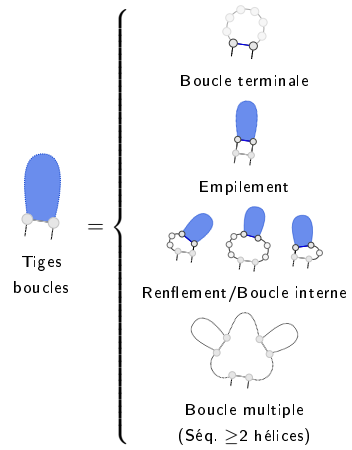


Énergies libres  $\Delta G$  des boucles dépendent des bases, assymétrie, bases *libres* (dangle) ...

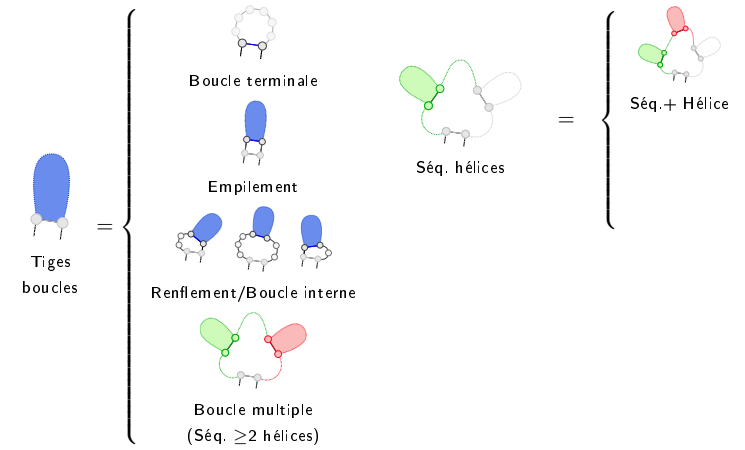
Déterminées expérimentalement  
+ Interpolation pour grandes boucles.

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

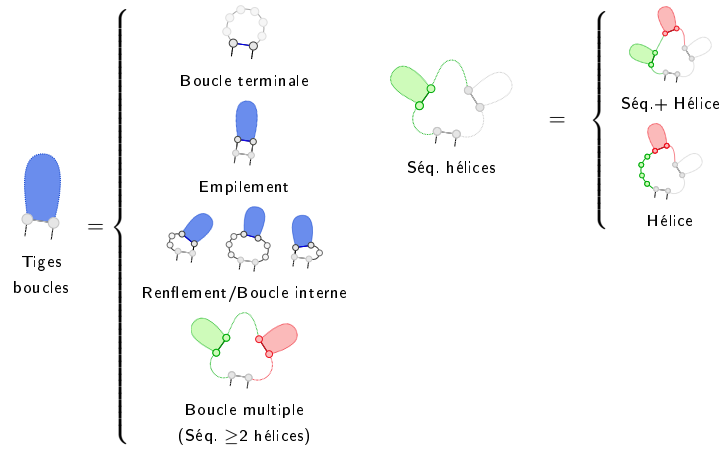
# MFE DP equations



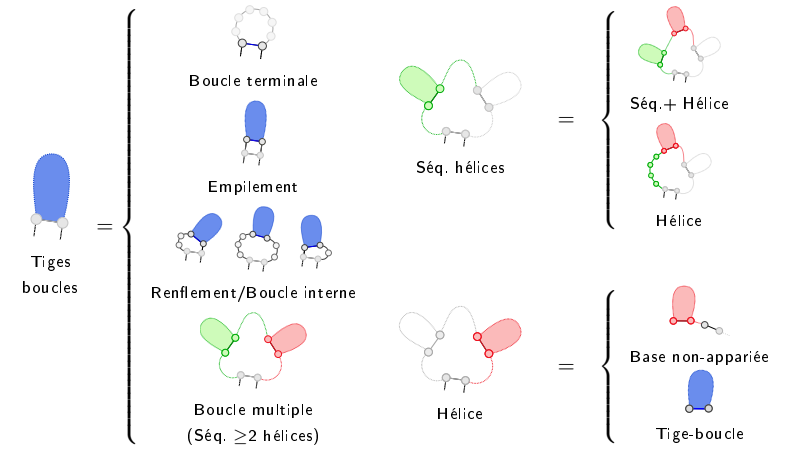
# MFE DP equations



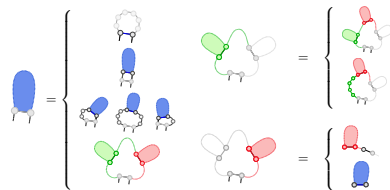
# MFE DP equations



# MFE DP equations



- $E_H(i, j)$  : 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

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

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 :

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

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

Reconstruction de la structure d'énergie minimale :

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

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

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

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

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

$\mathcal{O}(n)$  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 précalcul<sup>2</sup>

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

Reconstruction de la structure d'énergie minimale :

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

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

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

$\mathcal{O}(n)$  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 précalcul<sup>2</sup>

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

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

## Deux approches

### Definition (Replieement ab initio)

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

#### Avantages :

- Explication mécanique
- Complexité raisonnable  $\mathcal{O}(n^3)/\mathcal{O}(n^2)$  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

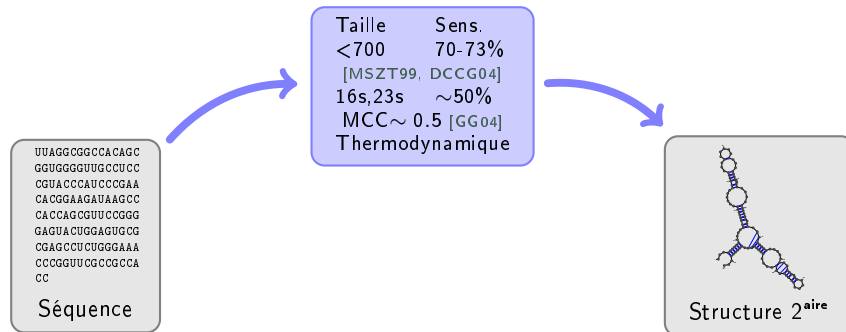
#### Limites :

- Complexité élevée
- Exploration non-exhaustive

## Performances

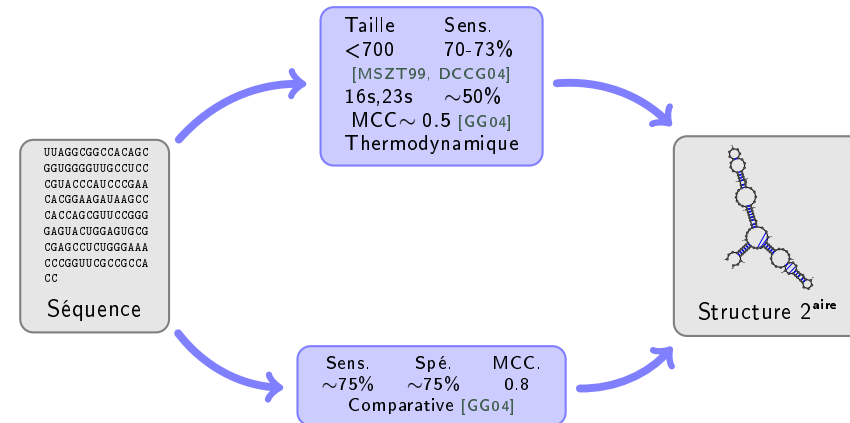


Rappel :  $MCC = \frac{e^+e^- - f^+f^-}{\sqrt{(e^++f^+)(e^++f^-)(e^-+f^+)(e^-+f^-)}}$



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

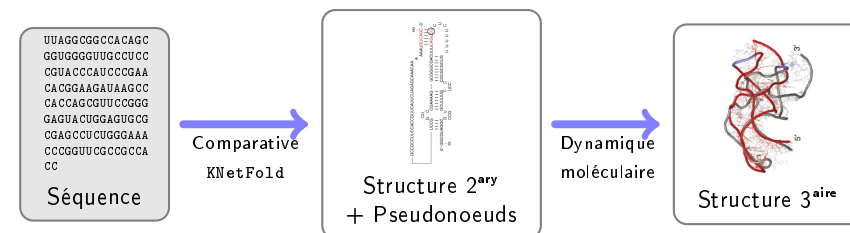
But : De la séquence à des modèles tri-dimensionnels!!!



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

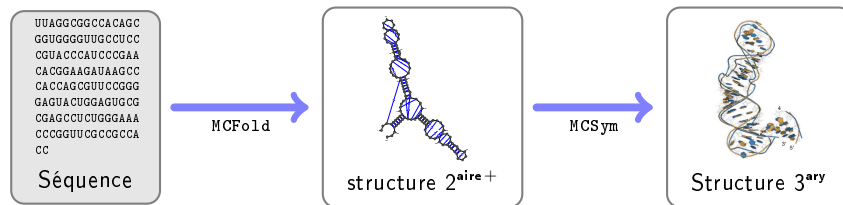
But : De la séquence à des modèles tri-dimensionnels!!!

- 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.

D.H. Mathews, J. Sabina, M. Zuker, and D.H. Turner. **Expanded sequence dependence of thermodynamic parameters improves prediction of RNA secondary structure.** *J Mol Biol*, 288 :911–940, 1999.

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 Nedozov, 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/repliection des structures secondaires (Python)

<http://www.lix.polytechnique.fr/~ponty/enseignement/2012-01-BIBS-TP1-RappelsPython.pdf>