



## Un aperçu de la comparaison de structures d'ARN

Alain Denise  
Cours ABA 2010-2011  
M2 Bioinformatique et Biostatistiques



## « Bio-Algorithmique » de l'ARN

- Prédiction de structure en fonction de la séquence
- Détermination d'une séquence en fonction de la structure
- Détection de motifs structurels dans une séquence ou dans une structure
- Comparaison de deux ou plusieurs structures
- Recherche de sous-structures communes à deux ou plusieurs structures



### Préliminaire :

- distance d' édition de deux séquences
- programmation dynamique



## Distance d'édition de 2 séquences

Deux séquences  $v = v_1v_2\dots v_n$  et  $w = w_1w_2\dots w_m$

Opérations d'édition :

- $\text{ins}(x,i)$
- $\text{suppr}(x,i)$
- $\text{subs}(x,y,i)$

CHAT -  $\text{suppr}(C,1) \rightarrow$  HAT -  $\text{subs}(H,R,1) \rightarrow$  RAT



## Distance d'édition de 2 séquences

- Chaque modification a un poids, dépendant de l'opération et des lettres en cause.
- Distance d'édition entre  $v$  et  $w$  : poids minimal d'une suite d'opérations permettant de transformer  $v$  en  $w$ .

CHAT -  $\text{suppr}(C,1) \rightarrow$  HAT -  $\text{subs}(H,R,1) \rightarrow$  RAT



## Distance d'édition de 2 séquences

$v = v_1v_2\dots v_n$

$w = w_1w_2\dots w_m$

$s(x,y)$  : score de substitution de  $x$  en  $y$

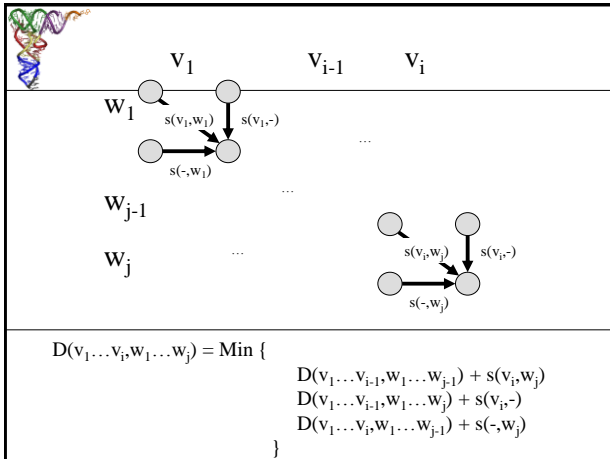
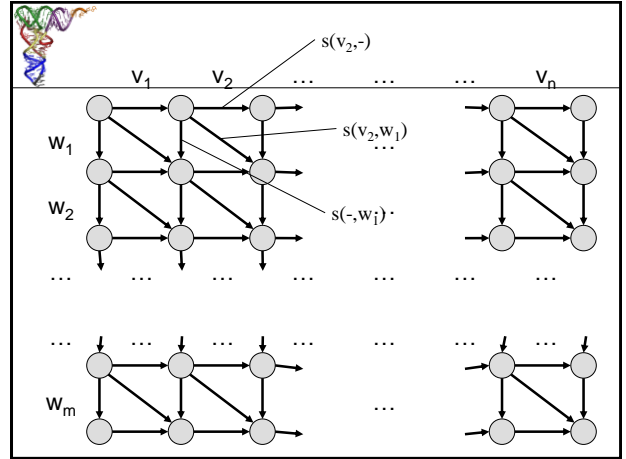
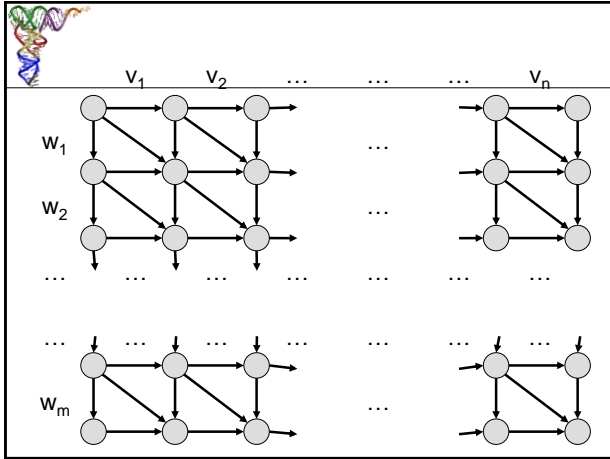
$s(x,-)$  : score de suppression de  $x$

$s(-,y)$  : score d'insertion de  $y$

$D(v,w)$  : distance d'édition de  $v$  et  $w$

$$D(v_1\dots v_i, w_1\dots w_j) = \text{Min} \left\{ \begin{array}{l} D(v_1\dots v_{i-1}, w_1\dots w_{j-1}) + s(v_i, w_j) \\ D(v_1\dots v_{i-1}, w_1\dots w_j) + s(v_i, -) \\ D(v_1\dots v_i, w_1\dots w_{j-1}) + s(-, w_j) \end{array} \right\}$$

Needleman, Wunsch 1970, Gotoh 1982



### Comparaison de structures d'ARN

### Pourquoi comparer des ARN ?

- A quel point ces structures sont-elles similaires (ou différentes ?)
  - classification
  - phylogénie
- Quelles parties des deux structures se ressemblent le plus ?
- La petite est-elle similaire à une partie de la grande ?

Comparaison → score + correspondance entre les structures

07/02/2011      Projet Brasero - ANR Blanc 2006      11

### Comparer les séquences ne suffit pas

- La fonction dépend de la structure.
- Des séquences différentes peuvent avoir la même structure.

Séquences :

```

[seq1] - ...-GGCCU-CCCCG
[seq2] - AAAAAAGGUU-...AAA
    
```

Séquences/structures :

```

[seq1] - ..(((...)))...
[seq1] - GGAGU-UUUUUGG
[seq2] - AAAAAAGGUUAAA
[seq2] - ..(((...)))...
    
```

07/02/2011      Projet Brasero - ANR Blanc 2006      12

### Séquences arc-annotées

GGGGAUUUAGCUCAAUGUGUAAGCGUCUCCUUAACAUGCGAGAAGUUGUGGGAUUGAUGCCCGCAUUCUCCACCA

NESTED (imbriquée) : structure secondaire sans pseudo-noeud

### Séquences arc-annotées

GGGGAUUUAGCUCAAUGUGUAAGCGUCUCCUUAACAUGCGAGAAGUUGUGGGAUUGAUGCCCGCAUUCUCCACCA

CROSSING (croisée) : structure secondaire avec pseudo-noeuds

### Séquences arc-annotées

GGGGAUUUAGCUCAAUGUGUAAGCGUCUCCUUAACAUGCGAGAAGUUGUGGGAUUGAUGCCCGCAUUCUCCACCA

UNLIMITED (générale) : structure tertiaire

### Séquences arc-annotées

GGGGAUUUAGCUCAAUGUGUAAGCGUCUCCUUAACAUGCGAGAAGUUGUGGGAUUGAUGCCCGCAUUCUCCACCA

PLAIN (sans arcs) : séquence sans appariement

### Opérations d'édition

AAGUCCAGACUUCGUUG

- Opérations on bases:
  - Substitution:  $A \rightarrow C$
  - Deletion / Insertion:  $A \rightarrow \emptyset$
- Opérations on arcs:
  - Arc-substitution:  $\overset{\frown}{C} \overset{\smile}{G} \rightarrow \overset{\frown}{U} \overset{\smile}{A}$
  - Arc-deletion / Arc-insertion:  $\overset{\frown}{C} \overset{\smile}{G} \rightarrow \emptyset$
  - Arc-breaking / :  $\overset{\frown}{C} \overset{\smile}{G} \rightarrow C \ G$
  - Arc-altering / :  $\overset{\frown}{C} \overset{\smile}{G} \rightarrow C \ -$

### Une édition de deux structures

AAGAAUAAUUUACGGGACCCUAUAAA

base-mismatch

CAGAAUAAUUUACGGGACCCUAUAAA

arc-mismatch

CGAGAAUAAUUUACGGGACCCUAUAAA

base-deletion

CGAGAAUACAUUUACGGGACCCUAUAAA

arc-altering

CGAGAAUACAUUACGGGACCCUAUAAA

arc-breaking

CGAGAAUACAUUACGGGACCCUAUAAA

arc-removing

### Complexity of the edition problem

	General	Crossing	Nested	Plain
General	NP-complete			
Crossing		NP-complete		
Nested			NP-complete	$O(nm^3)$
Plain				$O(nm / \log n)$

• Jiang, Lin, Ma, Zhang 2002  
 • Blin, Fertin, Rusu, Sinoquet 2003  
 • Crochemore, Landau, Ziv-Ukelson 2002

### The « nested-nested » case

→ Secondary structures (without pseudokots)  
 → Tree comparison

### Structure 2<sup>aire</sup> ↔ arbre

### Structure 2<sup>aire</sup> ↔ arbre

### Structure 2<sup>aire</sup> ↔ arbre

### Structure 2<sup>aire</sup> ↔ arbre



### Edition et alignement d'arbres

**Edition :** Transformer un arbre en un autre.

**Alignement :** Construire un super-arbre commun à deux arbres.

### Edition et alignement d'arbres

**Edition :** Transformer un arbre en un autre.

**Alignement :** Construire un super-arbre commun à deux arbres.

### Opérations classiques pour les arbres

**Suppression** du sommet  $v$  :  
les fils de  $v$  deviennent

1. fils du père de  $v$ ,
2. frères droits des frères gauches de  $v$ ,
3. frères gauches des frères droits de  $v$ ,

et leur ordre est conservé.

### Opérations classiques pour les arbres

**Insertion** du sommet  $v$  :  
un sommet  $v$  est ajouté comme père d'un certain nombre de frères consécutifs (opération symétrique à la suppression).

### Opérations classiques pour les arbres

**Substitution** du sommet  $v$  :  
l'étiquette de  $v$  est modifiée.

### Opérations classiques pour les arbres

Chaque opération d'édition a un coût.

**Edition :** Transformer un arbre en un autre

- ▶ par une suite d'opérations d'édition,
- ▶ en minimisant le coût.

### Opérations classiques pour les arbres

**Alignement :** Construire un super-arbre commun à deux arbres

- par une suite d'opérations d'édition,
- en minimisant le coût.

### Tree edition algorithm

Zhang, Shasha 1989

$$Tdist(\triangle, \blacktriangle) = \min \left\{ \begin{aligned} & Fdist(\triangle, \blacktriangle) + Change(\bullet, \bullet), \\ & Fdist(\triangle, \blacktriangle) + Delete(\bullet), \\ & Fdist(\triangle, \blacktriangle) + Insert(\bullet) \end{aligned} \right\}$$

$$Fdist(\triangle, \blacktriangle) = \min \left\{ \begin{aligned} & Fdist(\triangle, \blacktriangle) + Tdist(\triangle, \blacktriangle), \\ & Fdist(\triangle, \blacktriangle) + Delete(\bullet), \\ & Fdist(\triangle, \blacktriangle) + Insert(\bullet) \end{aligned} \right\}$$

### Algorithme d'édition

Décomposition en branches gauches

### Algorithme d'édition

Sous-arbres spéciaux

LINUX → L LI LIN LINU LINUX

Ce n'est pas la seule décomposition possible [Klein 1998 ; Dulucq, Touzet 2003]

### Algorithme d'édition

$$Tdist(\begin{matrix} D_4 & F_4 \\ A_1 & B_2 & C_3 & A_1 \\ & & & E_3 \\ & & & B_2 \end{matrix})$$

### Algorithme d'édition

$$Tdist(\begin{matrix} & E_3 \\ B_2 & B_2 \end{matrix})$$

	B2	E3
B2	0	1
E3	1	

$$Tdist(\begin{matrix} D_4 & F_4 \\ A_1 & B_2 & C_3 & A_1 \\ & & & E_3 \\ & & & B_2 \end{matrix})$$

	1	2	3	4
1				
2				
3				
4				

$$Tdist(\triangle, \blacktriangle) = \min \left\{ \begin{aligned} & Fdist(\triangle, \blacktriangle) + Change(\bullet, \bullet), \\ & Fdist(\triangle, \blacktriangle) + Delete(\bullet), \\ & Fdist(\triangle, \blacktriangle) + Insert(\bullet) \end{aligned} \right\}$$

### Algorithme d'édition

$Tdist(B_2, E_3)$

	B2	E3	
B2	0	1	2
	1	0	

$Tdist(\triangle, \triangle)$

$= \text{Min} \left\{ \begin{array}{l} Fdist(\triangle, \triangle) + \text{Change}(\bullet, \bullet), \\ Fdist(\triangle, \triangle) + \text{Delete}(\bullet), \\ Fdist(\triangle, \triangle) + \text{Insert}(\bullet) \end{array} \right\}$

$Tdist(D_4, F_4)$

	1	2	3	4
1				
2		0		
3				
4				

### Algorithme d'édition

$Tdist(B_2, E_3)$

	B2	E3	
B2	0	1	2
	1	0	1

$Tdist(\triangle, \triangle)$

$= \text{Min} \left\{ \begin{array}{l} Fdist(\triangle, \triangle) + \text{Change}(\bullet, \bullet), \\ Fdist(\triangle, \triangle) + \text{Delete}(\bullet), \\ Fdist(\triangle, \triangle) + \text{Insert}(\bullet) \end{array} \right\}$

$Tdist(D_4, F_4)$

	1	2	3	4
1				
2		0	1	
3				
4				

### Algorithme d'édition

$Tdist(B_2, A_1)$

	A1	B2	E3	F4	
B2	0	1	2	3	4
	1				

$Tdist(\triangle, \triangle)$

$= \text{Min} \left\{ \begin{array}{l} Fdist(\triangle, \triangle) + \text{Change}(\bullet, \bullet), \\ Fdist(\triangle, \triangle) + \text{Delete}(\bullet), \\ Fdist(\triangle, \triangle) + \text{Insert}(\bullet) \end{array} \right\}$

$Tdist(D_4, F_4)$

	1	2	3	4
1				
2		0	1	
3				
4				

### Algorithme d'édition

$Tdist(B_2, A_1)$

	A1	B2	E3	F4	
B2	0	1	2	3	4
	1	1			

$Tdist(\triangle, \triangle)$

$= \text{Min} \left\{ \begin{array}{l} Fdist(\triangle, \triangle) + \text{Change}(\bullet, \bullet), \\ Fdist(\triangle, \triangle) + \text{Delete}(\bullet), \\ Fdist(\triangle, \triangle) + \text{Insert}(\bullet) \end{array} \right\}$

$Tdist(D_4, F_4)$

	1	2	3	4
1				
2	1	0	1	
3				
4				

### Algorithme d'édition

$Tdist(B_2, A_1)$

	A1	B2	E3	F4	
B2	0	1	2	3	4
	1	1	1		

$Fdist(\triangle, \triangle)$

$= \text{Min} \left\{ \begin{array}{l} Fdist(\triangle, \triangle) - Tdist(\triangle, \triangle), \\ Fdist(\triangle, \triangle) + \text{Delete}(\bullet), \\ Fdist(\triangle, \triangle) + \text{Insert}(\bullet) \end{array} \right\}$

$Tdist(D_4, F_4)$

	1	2	3	4
1				
2	1	0	1	
3				
4				

### Algorithme d'édition

$Tdist(B_2, A_1)$

	A1	B2	E3	F4	
B2	0	1	2	3	4
	1	1	1	2	

$Fdist(\triangle, \triangle)$

$= \text{Min} \left\{ \begin{array}{l} Fdist(\triangle, \triangle) - Tdist(\triangle, \triangle), \\ Fdist(\triangle, \triangle) + \text{Delete}(\bullet), \\ Fdist(\triangle, \triangle) + \text{Insert}(\bullet) \end{array} \right\}$

$Tdist(D_4, F_4)$

	1	2	3	4
1				
2	1	0	1	
3				
4				

### Algorithme d'édition

Tdist( $B_2, A_1, E_3$ )

	A1	B2	E3	F4
B2	0	1	2	3
	1	1	1	-2

Tdist( $A_1, B_2, C_3, A_1, E_3$ )

	1	2	3	4
1				
2	1	0	1	
3				
4				

$Fdist(\triangle, \triangle, \triangle, \triangle)$   
 $= \text{Min} \left\{ \begin{array}{l} Fdist(\triangle, \triangle, \triangle) + Tdist(\triangle, \triangle, \triangle) \\ Fdist(\triangle, \triangle, \triangle) + Delete(\triangle, \triangle) \\ Fdist(\triangle, \triangle, \triangle) + Insert(\triangle) \end{array} \right\}$

### Algorithme d'édition

Tdist( $B_2, A_1, E_3$ )

	A1	B2	E3	F4
B2	0	1	2	3
	1	1	1	-3

Tdist( $A_1, B_2, C_3, A_1, E_3$ )

	1	2	3	4
1				
2	1	0	1	3
3				
4				

$Tdist(\triangle, \triangle, \triangle)$   
 $= \text{Min} \left\{ \begin{array}{l} Fdist(\triangle, \triangle, \triangle) + Change(\triangle, \triangle) \\ Fdist(\triangle, \triangle, \triangle) + Delete(\triangle) \\ Fdist(\triangle, \triangle, \triangle) + Insert(\triangle) \end{array} \right\}$

### Algorithme d'édition

...

### Algorithme d'édition

Tdist( $A_1, B_2, C_3, A_1, E_3$ )

	A1	B2	E3	F4
A1	0	1	2	3
B2	1	0	1	2
C3	2	1	0	1
A1	3	2	1	2
D4	4	3	2	3

Tdist( $A_1, B_2, C_3, A_1, E_3$ )

	1	2	3	4
1	0	1	2	3
2	1	0	1	3
3	1	1	2	4
4	3	3	3	3

### Complexité de l'algorithme

Chaque sommet  $s$  intervient autant de fois qu'il est dans un sous-arbre de la décomposition. C'est sa hauteur réduite  $HR(s)$ .

$C_{\text{edit}}(T_1, T_2) = HR(T_1) \times HR(T_2)$

Dans le pire des cas :

$$C_{\text{edit}}(T_1, T_2) = |T_1|^2 \times |T_2|^2$$

### Complexité moyenne

[Dulucq, Tichit 2003]

Considérons la série génératrice des arborescences relativement à leur taille et à leur hauteur réduite,

$$f(q, t) = \sum_{n \geq 1, k \geq 1} a_{n,k} q^n t^k$$

dans laquelle  $a_{n,k}$  est le nombre d'arborescences ayant  $n$  sommets et une hauteur réduite égale à  $k$ .

La hauteur réduite moyenne d'un arbre à  $n+1$  sommets est

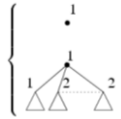
$$HR(n+1) = \frac{1}{C_n} \sum_k k \cdot a_{n+1,k} = \frac{1}{C_n} \left[ \frac{d}{dq} f(q, t) \right]_{q=1, t^{n+1}}$$

Où  $C_n$  désigne le nombre d'arbres à  $n+1$  sommets :

$$C_n = \frac{1}{n+1} \binom{2n}{n}$$

### Complexité moyenne

[Dulucq, Tichit 2003]

$$f(q, t) = \sum_{n \geq 1, k \geq 1} a_{n,k} q^k t^n$$


$$f(q, t) = qt + qt \cdot f(q, t) \frac{1}{1 - f(q, qt)}$$

### Complexité moyenne

[Dulucq, Tichit 2003]

$$\left[ \frac{d}{dq} f(q, t) \right]_{q=1} = \frac{t(1 - 2t + \sqrt{1 - 4t})}{2(1 - 4t)}$$

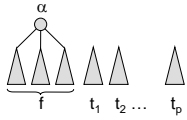
$$\left( \left[ \frac{d}{dq} f(q, t) \right]_{q=1}, t^{n+1} \right) = \begin{cases} 4^{n-1} + \binom{2n-1}{n} & \text{si } n \geq 1 \\ 1 & \text{si } n = 0 \end{cases}$$

$$HR(n+1) = (n+1) \frac{\binom{2n-1}{n} + 4^{n-1}}{\binom{2n}{n}}$$

⇒  $C_{\text{edit}} = HR(n) \times HR(m) \sim n^{3/2} m^{3/2}$

### Tree edition algorithm

Zhang, Shasha 1989

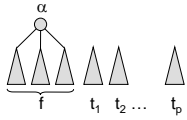
$$\text{Score}(\alpha(f), \alpha'(f')) = \text{Max} \begin{cases} \text{Subs}(\alpha, \alpha') + \text{Score}(f, f') \\ \text{Ins}(\alpha') + \text{Score}(\alpha(f), f') \\ \text{Del}(\alpha) + \text{Score}(f, \alpha'(f')) \end{cases}$$


$$\text{Score}(\alpha(f) \circ t_1 \circ \dots \circ t_p, [\alpha'(f'), t'_1 \circ \dots \circ t'_q]) = \text{Max} \begin{cases} \text{Score}(\alpha(f), \alpha'(f')) + \text{Score}([t_1 \circ \dots \circ t_p], [t'_1 \circ \dots \circ t'_q]) \\ \text{Ins}(\alpha') + \text{Score}(\alpha(f) \circ t_1 \circ \dots \circ t_p, [f', t'_1 \circ \dots \circ t'_q]) \\ \text{Del}(\alpha) + \text{Score}([f \circ t_1 \circ \dots \circ t_p], [\alpha'(f') \circ t'_1 \circ \dots \circ t'_q]) \end{cases}$$

$O(n^4 \log n)$  [Klein 1998]

### Tree alignment algorithm

Jiang, Wang, Zhang 1995

$$\text{Score}(r(f), r'(f')) = \text{Max} \begin{cases} \text{Subs}(\alpha, \alpha') + \text{Score}(f, f') \\ \text{Ins}(\alpha') + \text{Score}(\alpha(f), f') \\ \text{Del}(\alpha) + \text{Score}(f, \alpha'(f')) \end{cases}$$


$$\text{Score}(\alpha(f) \circ t_1 \circ \dots \circ t_p; \alpha'(f') \circ t'_1 \circ \dots \circ t'_q) = \text{Max} \begin{cases} \text{Score}(\alpha(f); \alpha'(f')) + \text{Score}(t_1 \circ \dots \circ t_p; t'_1 \circ \dots \circ t'_q) \\ \text{Ins}(\alpha') + \text{Max}_i \{ \text{Score}(\alpha(f) \circ \dots \circ t_i; f') + \text{Score}(t_{i+1} \circ \dots \circ t_p; t'_1 \circ \dots \circ t'_q) \} \\ \text{Del}(\alpha) + \text{Max}_j \{ \text{Score}(f; \alpha'(f') \circ t'_1 \circ \dots \circ t'_j) + \text{Score}(t_1 \circ \dots \circ t_p; t'_{j+1} \circ \dots \circ t'_q) \} \end{cases}$$

$O(n^4 \log n)$

### Edition vs Alignment

$$\text{Score}([r(f), t_1, \dots, t_p], [r'(f'), t'_1, \dots, t'_q]) = \text{Max} \begin{cases} \dots \\ \text{Ins}(r') + \text{Score}([r(f), t_1, \dots, t_p], [f', t'_1, \dots, t'_q]) \\ \dots \end{cases}$$

$$\text{Score}([r(f), t_1, \dots, t_p], [r'(f'), t'_1, \dots, t'_q]) = \text{Max} \begin{cases} \dots \\ \text{Ins}(r') + \text{Max} \{ \text{Score}([r(f), \dots, t_i], f') + \text{Score}([t_{i+1}, \dots, t_p], [t'_1, \dots, t'_q]) \} \\ \dots \end{cases}$$

### Edition vs Alignment

$$\text{Score}(\Delta \Delta \Delta \Delta \Delta, \blacktriangle \blacktriangle \blacktriangle \blacktriangle \blacktriangle) = \text{Max} \begin{cases} \dots \\ \text{Ins}(\bullet) + \text{Score}(\Delta \Delta \Delta \Delta \Delta, \blacktriangle \blacktriangle \blacktriangle \blacktriangle \blacktriangle) \\ \dots \end{cases}$$

$$\text{Score}(\Delta \Delta \Delta \Delta \Delta, \blacktriangle \blacktriangle \blacktriangle \blacktriangle \blacktriangle) = \text{Max} \begin{cases} \dots \\ \text{Ins}(\bullet) + \text{Max} \{ \text{Score}(\Delta \Delta \Delta \Delta, \blacktriangle \blacktriangle \blacktriangle) + \text{Score}(\Delta \Delta, \blacktriangle \blacktriangle \blacktriangle \blacktriangle) \} \\ \dots \end{cases}$$

## Retour à l'ARN

<i>base-match</i>	$A \rightarrow A$	
<i>base-mismatch</i>	$A \rightarrow G$	
<i>base-deletion</i>	$A \rightarrow -$ $- \rightarrow A$	
<i>arc-match</i>	$A \widehat{U} \rightarrow A \widehat{U}$	
<i>arc-mismatch</i>	$A \widehat{U} \rightarrow G \widehat{C}$	
<i>arc-removing</i>	$A \widehat{U} \rightarrow - -$ $- - \rightarrow A \widehat{U}$	
<i>arc-breaking</i>	$A \widehat{U} \rightarrow A \widehat{U}$ $A \widehat{U} \rightarrow A \widehat{U}$	
<i>arc-altering</i>	$A \widehat{U} \rightarrow A -$ $A \widehat{U} \rightarrow - U$ $A - \rightarrow A \widehat{U}$ $- U \rightarrow A \widehat{U}$	

## Les opérations classiques sur les arbres ne suffisent pas pour l'ARN.

Delete( $\circ$ )  
Insert( $\circ$ )  
Insert( $\circ$ ) } 3 operations!

## A first solution

Höchstmann, Töller, Gierich, Kurtz 2003 (RNAforester)

But this implies some constraints on the scores. For example:

Arc-deletion = Arc-Breaking + 2 Base-Deletion

## New edition operations on trees

- Operations on bases:
  - Substitution:  $A \rightarrow C$
  - Deletion / Insertion:  $A \rightarrow \emptyset$
- Operations on arcs:
  - Arc-substitution:  $\widehat{C} \widehat{G} \rightarrow \widehat{U} \widehat{A}$
  - Arc-deletion / Arc-insertion:  $\widehat{C} \widehat{G} \rightarrow \emptyset$
  - Arc-breaking / :  $\widehat{C} \widehat{G} \rightarrow C \widehat{G}$
  - Arc-altering / :  $\widehat{C} \widehat{G} \rightarrow C -$

## Tree edition, tree alignment

	Tree operations	RNA operations
Edition	$O(n^3 \log n)$ [Zhang-Shasha 1989, Klein 1998]	NP-complete [Blin, Fertin, Sinoquet, Rusu 2003]
Alignment	$O(n^4)$ [Jiang, Wang, Zhang 1995]	?

## RNA structure alignment algorithm

$$A(v(f) \circ g, v'(f') \circ g') =$$

$\begin{aligned}
 & \text{basedel}(v) + A(g, v'(f') \circ g') \\
 & \quad \text{if } v \text{ is a leaf} \\
 & \text{basesins}(v') + A(v(f) \circ g, g') \\
 & \quad \text{if } v' \text{ is a leaf} \\
 & \text{basesub}(v, v') + A(g, g') \quad \text{[Jiang, Wang, Zhang 1995]} \\
 & \quad \text{if } v, v' \text{ are leaves} \\
 & \text{pairedel}(v) + \min\{A(f, p') + A(g, s') \mid p' \circ s' = v'(f') \circ g'\} \\
 & \quad \text{if } v \text{ is an internal node} \\
 & \text{pairins}(v') + \min\{A(p, f') + A(s, g') \mid p \circ s = v(f) \circ g\} \\
 & \quad \text{if } v' \text{ is an internal node} \\
 & \text{pairsub}(v, v') + A(f, f') + A(g, g') \\
 & \quad \text{if } v, v' \text{ are internal nodes} \\
 & \text{scission}(v, v', v'_c) + \min\{A(f, p') + A(g, s') \mid p' \circ v'_c \circ s' = g'\} \\
 & \quad \text{if } v \text{ is an internal node, } v', v'_c \text{ are leaves} \\
 & \text{fusion}(v, v_c, v') + \min\{A(p, f') + A(s, g') \mid p \circ v_c \circ s = g\} \\
 & \quad \text{if } v, v_c \text{ are leaves, } v' \text{ is an internal node} \\
 & \text{leftalt}(v, v'_c) + \min\{A(f, p') + A(g, s') \mid p' \circ v'_c \circ s' = v'(f') \circ g'\} \\
 & \quad \text{if } v \text{ is an internal node, } v'_c \text{ is a leaf} \\
 & \text{rightalt}(v, v') + \min\{A(f, p') + A(g, s') \mid p' \circ s' = g'\} \\
 & \quad \text{if } v \text{ is an internal node, } v' \text{ is a leaf} \\
 & \text{leftcompl}(v_c, v') + \min\{A(p, f') + A(s, g') \mid p \circ v_c \circ s = v(f) \circ g\} \\
 & \quad \text{if } v_c \text{ is a leaf, } v' \text{ is an internal node} \\
 & \text{rightcompl}(v, v') + \min\{A(p, f') + A(s, g') \mid p \circ s = g\} \\
 & \quad \text{if } v \text{ is a leaf, } v' \text{ is an internal node}
 \end{aligned}$

Spécifique à l'ARN

[Blin, Denise, Dulucq, Herrbach, Touzet 2008]

Complexité au pire :  $O(n^4)$

66



### Application

Allali 2004

*Saccharomyces uvarum*      *Saccharomyces kluyveri*

### Application

Allali 2004

*Saccharomyces uvarum*      *Saccharomyces kluyveri*

```

U G G A A C A G U G G U A
. . . . .
A C C U U G U C G U C G U
    
```

```

C G G A A C A U G G C A
. . . . .
G C U U U G U G C C G U
U
    
```

### Une approche multi-échelles

Allali, Sagot 2004

*Thermotoga maritima*

### Une approche multi-échelles

Allali, Sagot 2004

Opérations supplémentaires :  
Fusion de nœuds  
Fusion d'arêtes

### Une approche multi-échelles

Allali, Sagot 2004

Secondary Structure: Group I Intron      Secondary Structure: Group I Intron

*Acanthamoeba griffini*  
Nuclear SSU rRNA [1,516]  
(S01337)(IC1)  
1. cellular organelles 2. Bacteria  
3. eukaryotic group  
4. Acetabularia  
5. Acetabularia

*Chlorella sorokiniana*  
Nuclear SSU rRNA  
(2,1046)  
(X73993)(IC1)  
1. cellular organelles 2. Eukaryota  
3. Viridiplantae 4. Chlorophyta  
5. Chlorophyta 6. Chlorophyta  
7. Chlorophyta 8. Chlorophyta  
September 2004

### Une approche multi-échelles

Allali, Sagot 2004

9/7      2/0  
3      3  
11/5  
25/21      2/0  
4

### Une approche multi-échelles

Allali, Sagot 2004

### Une approche multi-échelles

Allali, Sagot 2004

### Une approche multi-échelles

Allali, Sagot 2004


### Logiciels

### NestedAlign

07/02/2011      Projet Brasero - ANR Blanc 2006      83

### Gardenia

07/02/2011      Projet Brasero - ANR Blanc 2006      84



## Varna

VARNA: Visualization Applet for RNA  
A Java Applet implemented as Applet for viewing the RNA secondary structure

<http://varna.lri.fr>


Linear algorithm    Circular algorithm

**IRESite: The database of experimentally verified IRES structures**

Human Integrated adenovirus 2 VA-RNA (RFAM: RF00162)

BVDV1 mRNA within the region 1-38..1034 (Luthe)


85






## MiGaL

<http://www-igm.univ-mlv.fr/~allali/migal/>


07/02/2011    Blanc 2006    86






## Bibliographie I

-  K. Zhang and D. Shasha  
Simple fast algorithms for the editing distance between trees and related problems  
*SIAM J. COMPUT.*, vol. 18, no 6, pp 1245-1262, 1989.  
Klein-Verlag, 1990.
-  S. Dulucq and L. Tichit  
RNA secondary structure comparison : exact analysis of the Zhang-Shasha tree edit algorithm  
*Theoretical Computer Science*, 306, pp 471-484, 2003.
-  P. Klein  
Computing the edit-distance between unrooted ordered trees  
*Proceedings of 6th European Symposium on Algorithms*, pp 91-102, 1998.

39 / 42



## Bibliographie II

-  S. Dulucq and H. Touzet  
Analysis of tree edit distance algorithms  
*Combinatorial Pattern Matching*, 2676, pp 83-95, 2003.
-  T. Jiang, L. Wang and K. Zhang  
Alignment of trees - an alternative to tree edit  
*Theoretical Computer Science*, 143, pp 137-148, 1995.
-  T. Jiang, G. Lin, B. Ma and K. Zhang  
A general edit distance between RNA structures  
*RECOMB'01*.
-  G. Blin, G. Fertin, I. Rusu and C. Sinoquet  
RNA Sequences and the EDIT(NESTED,NESTED) Problem  
*Research Report RR-IRIN-03.07, IRIN*, 2003.

40 / 42



## Bibliographie III

-  M. Höchsmann, T. Töller, R. Giegerich and S. Kurtz  
Local similarity in RNA secondary structures  
*CSB 2003*.
-  M. Höchsmann  
The tree alignment model : algorithms, implementations and applications for the analysis of RNA secondary structures.  
*PhD Thesis*, 2005.
-  C. Herrbach, A. Denise, S. Dulucq and H. Touzet  
Alignment of RNA secondary structures using a full set of operations  
*Research Report LRI-1451*, 2006.



## Bibliographie IV

-  C. Herrbach  
Etude algorithmique et statistique de la comparaison des structures secondaires d'ARN  
*PhD Thesis*, 2007.
-  V. Guignon, C. Chauve and S. Hamel  
An edit distance between RNA stem-loops  
*SPIRE 2005*.
-  J. Allali and M.-F. Sagot  
Novel tree edit operations for RNA secondary structure comparison  
*Proceedings of WABI*, 2006.
-  J. Allali  
Comparaison de structures secondaires d'ARN  
*PhD thesis*, 2004.