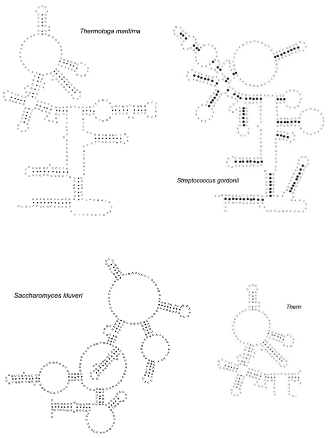
Pourquoi comparer des ARN ?

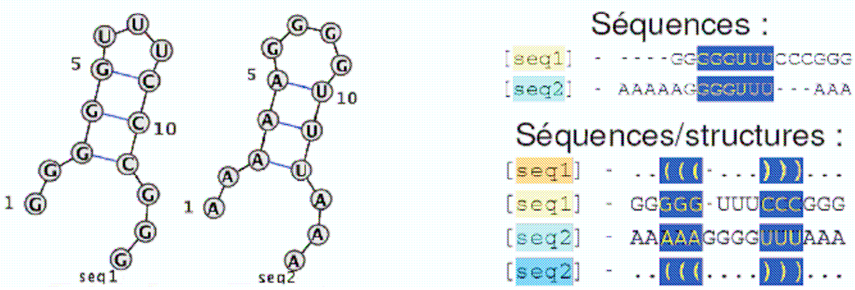


- A quel point ces structures sont-elles similaires (ou différentes ?)
  - → classification
  - → phylogénie
- Quelles parties des deux structures se ressemblent le plus ?
- La petite est-elle similaire à une partie de la grande ?

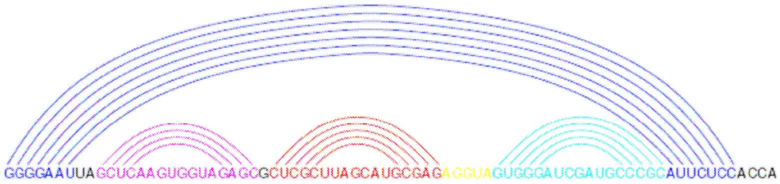
Comparaison → score + correspondance entre les structures

Comparer les séquences ne suffit pas

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

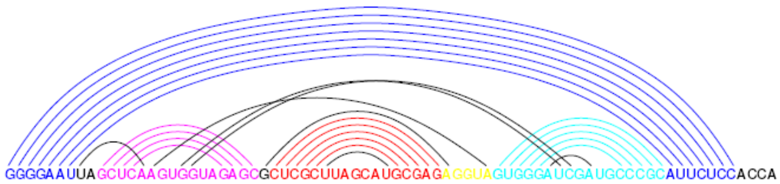


Séquences arc-annotées



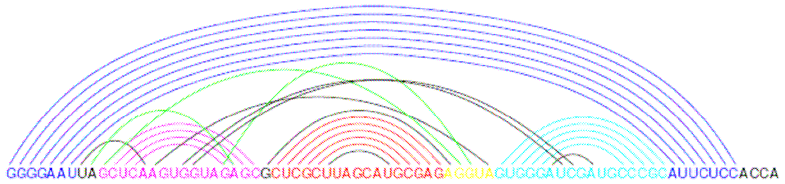
NESTED (imbriquée) : structure secondaire sans pseudo-noeud

Séquences arc-annotées



CROSSING (croisée) : structure secondaire avec pseudo-noeuds

Séquences arc-annotées



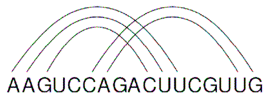
UNLIMITED (générale) : structure tertiaire

Séquences arc-annotées

GGGGAAUUAGCUCAAGUGGUAGAGCGCUCGCUUAGCAUGCGAGAGGUAGUGGGAUCGAUGCCCGCAUUCUCCACCA

PLAIN (sans arcs) : séquence sans appariement

Opérations d'édition



- Opérations on bases:
  - Substitution: A → C
  - Deletion / Insertion: A → ∅
- Operations on arcs:
  - Arc-substitution: C G → U A
  - Arc-deletion / Arc-insertion: C G → ∅
  - Arc-breaking / : C G → C G
  - Arc-altering / : C G → C -

Une édition de deux structures

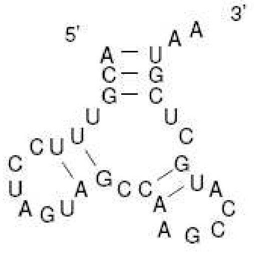
Complexity of the edition problem

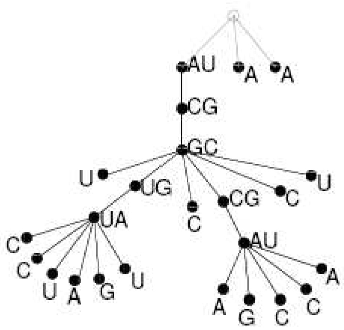
	General	Crossing	Nested	Plain
General	**NP-complete**			
Crossing		**NP-complete**		
Nested			**NP-complete**	**O(nm³)**
Plain				**O(nm / logn)**

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

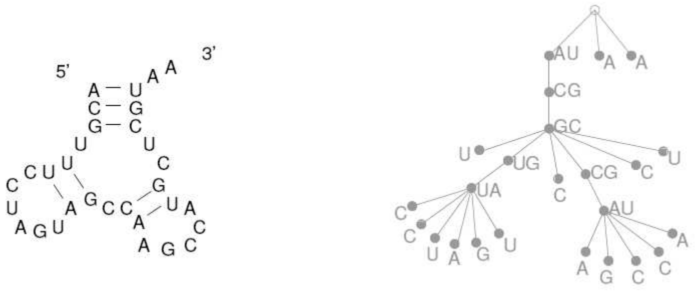
The « nested-nested » case

→ Secondary structures (without pseudokots)
→ Tree comparison

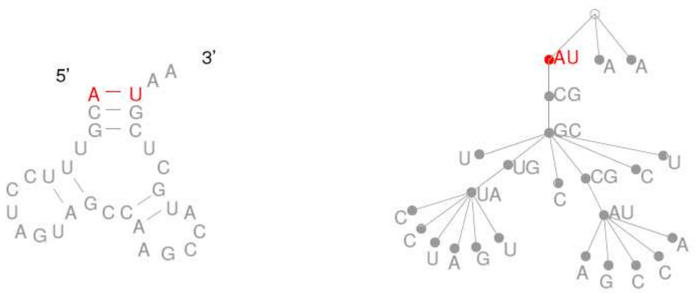




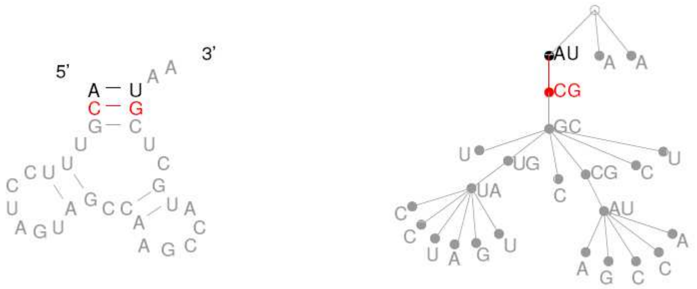
Structure 2^{aire} ↔ arbre



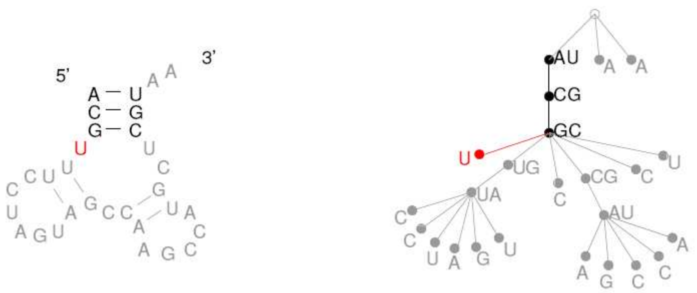
Structure 2^{aire} ↔ arbre



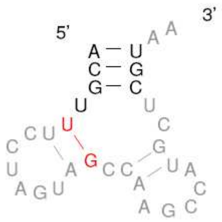
Structure 2^{aire} ↔ arbre

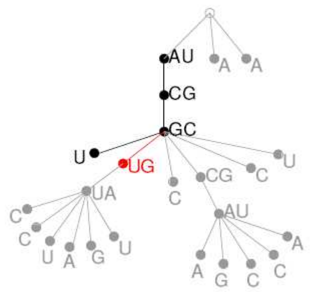


Structure 2^{aire} ↔ arbre

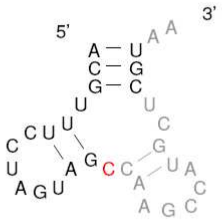


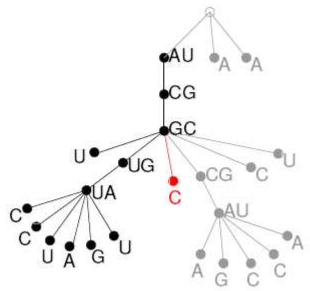
Structure 2^{aire} ↔ arbre



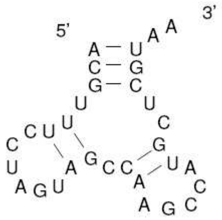


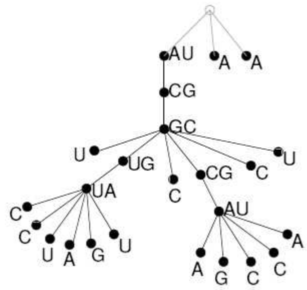
Structure 2^{aire} ↔ arbre



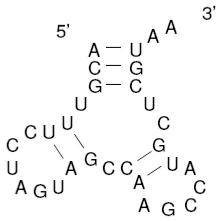


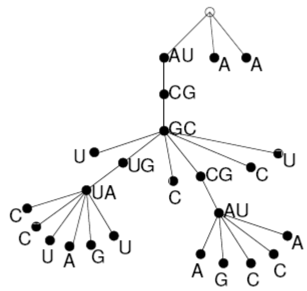
Structure 2^{aire} ↔ arbre





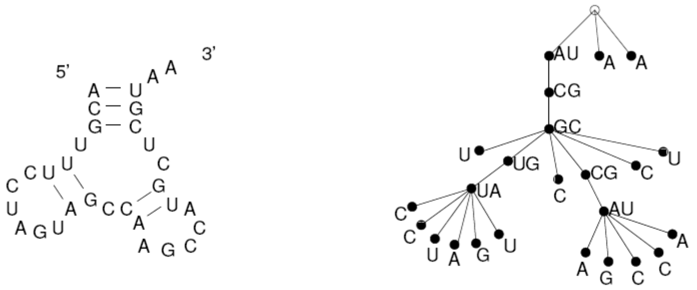
Structure 2^{aire} ↔ arbre





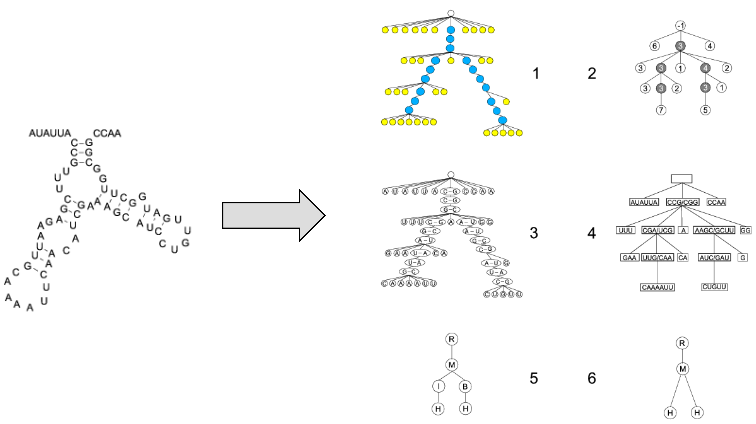
- Il est possible d’ajouter un racine pour obtenir un arbre.

Structure 2^{aire} ↔ arbre



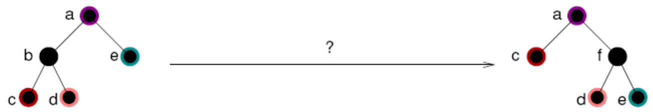
- Il est possible d'ajouter un racine pour obtenir un arbre.
- Les bases non appariées sont des feuilles.
- Les paires de bases sont des sommets internes.

Différentes échelles de représentation

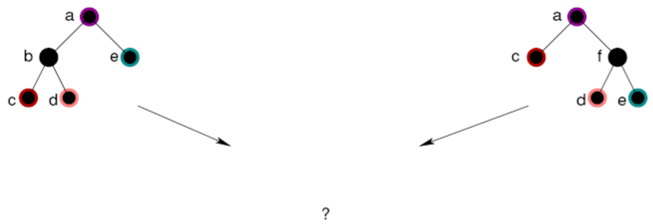


Edition et alignement d’arbres

Edition : Transformer un arbre en un autre.

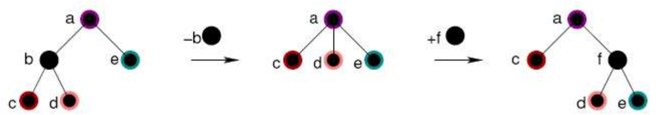


Alignement : Construire un super-arbre commun à deux arbres.

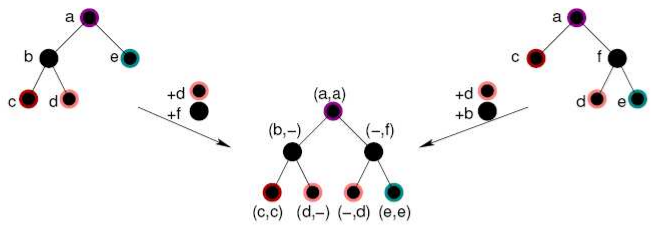


Edition et alignement d’arbres

Edition : Transformer un arbre en un autre.



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

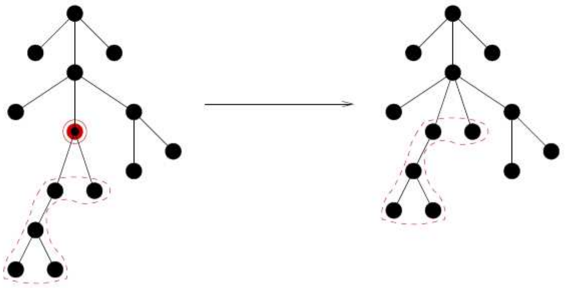


Opérations classiques pour les arbres

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

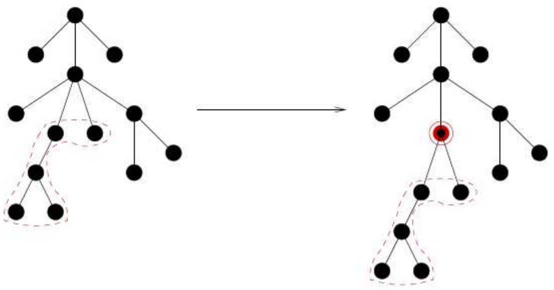
1. fils du père de *v*,
2. frères droits des frères gauches de *v*,
3. frères gauches des frères droits de *v*,

et leur ordre est conservé.



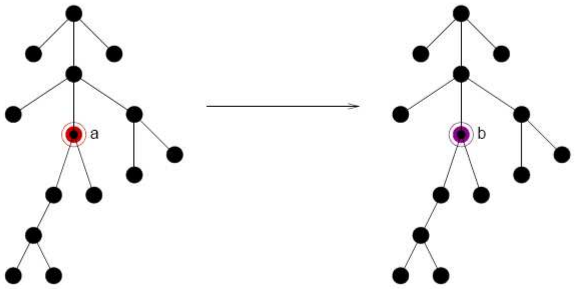
Opérations classiques pour les arbres

Insertion du sommet *v* :
un sommet *v* est ajouté comme père d'un cer
nombre de frères consécutifs (opération symé
la suppression).



Opérations classiques pour les arbres

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

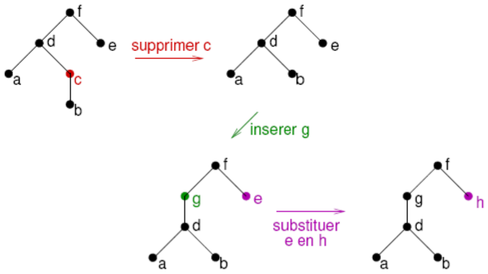


Opérations classiques pour les arbres

Chaque opération d'édition a un coût.

Edition : Transformer un arbre en un autre

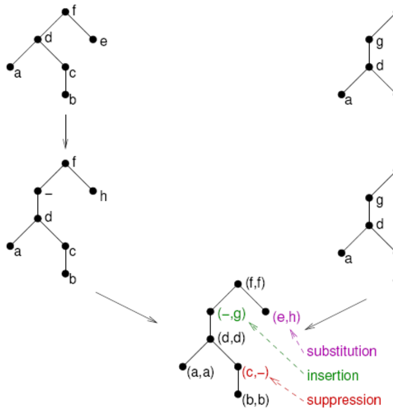
- par une suite d'opérations d'édition,
- en minimisant le coût.



Opérations classiques pour les arbres

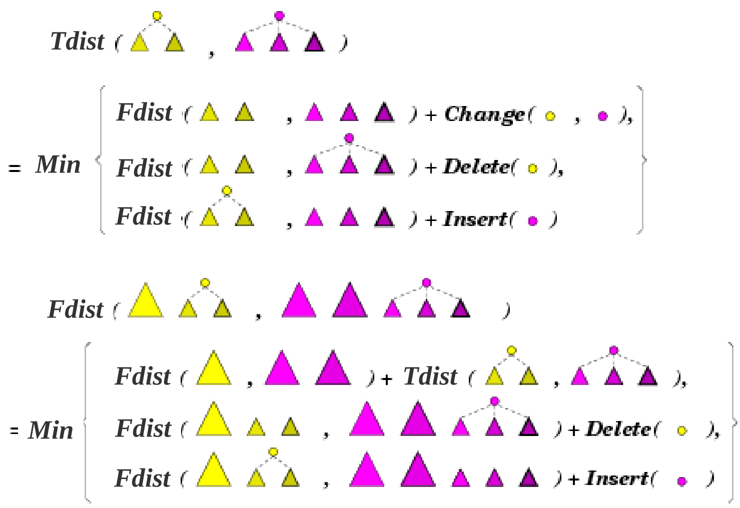
Alignement : Construire un super-arbre commun à deux arbres

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



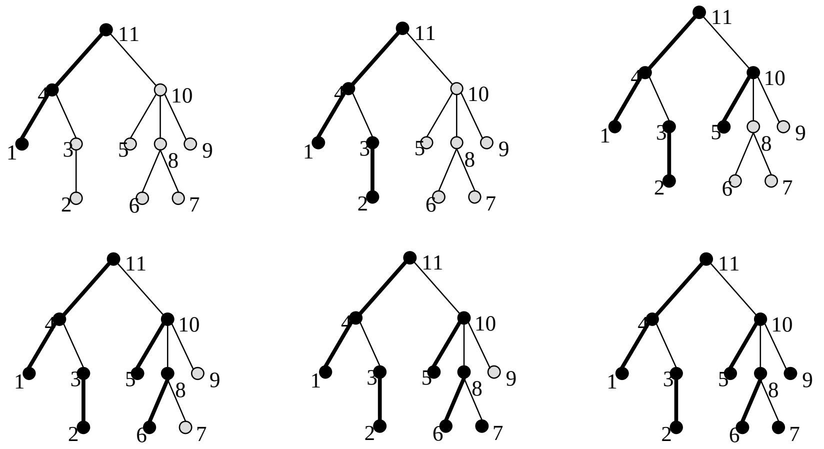
Tree edition algorithm

Zhang, Shasha 1989



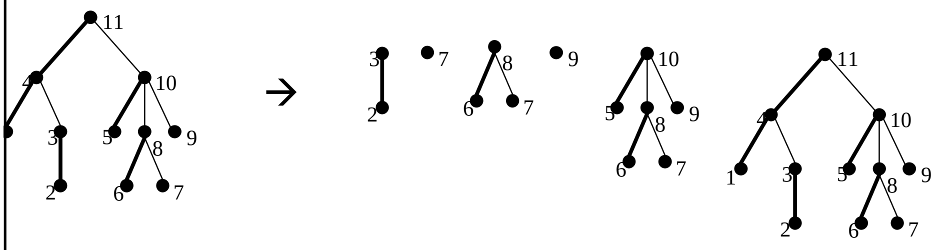
Algorithme d'édition

Décomposition en branches gauches



Algorithme d'édition

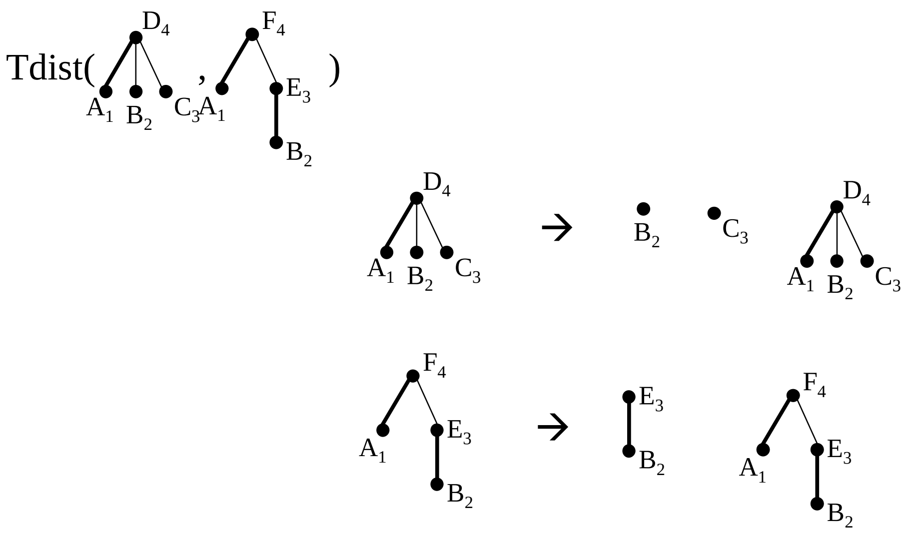
Sous-arbres spéciaux



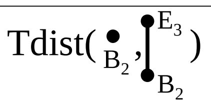
LINUX → L LI LIN LINU LINUX

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

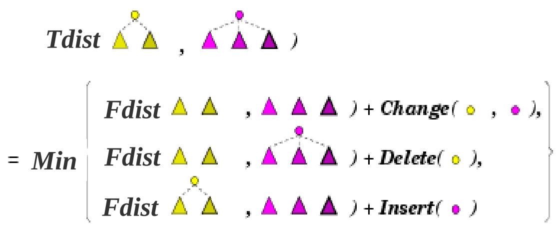
Algorithme d'édition

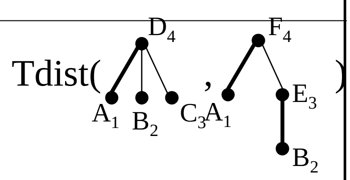


Algorithme d'édition



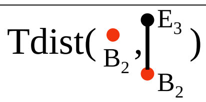
		B2	E3
	0	1	2
B2	1		



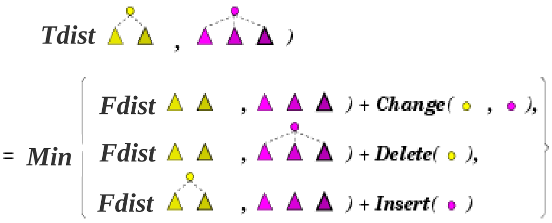


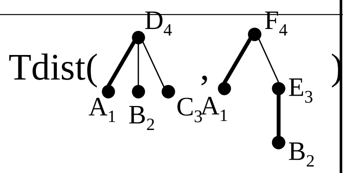
	1	2	3	4
1				
2				
3				
4				

Algorithme d'édition



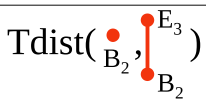
		B2	E3
	0	1	2
B2	1	0	



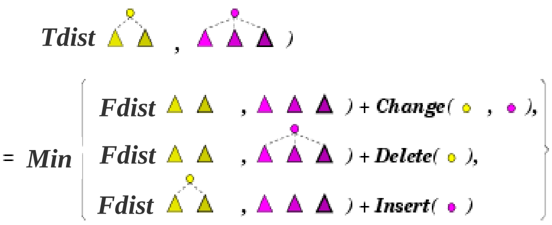


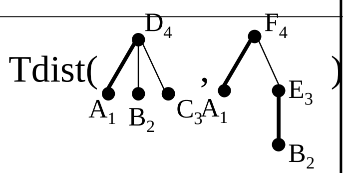
	1	2	3	4
1				
2		0		
3				
4				

Algorithme d'édition



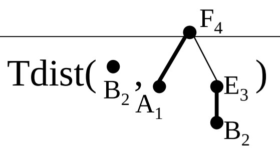
		B2	E3
	0	1	2
B2	1	0	1



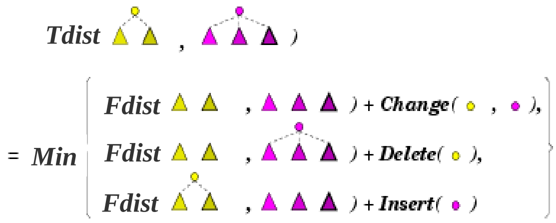


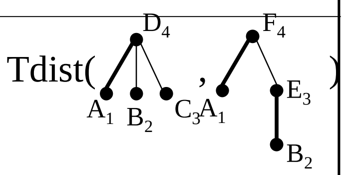
	1	2	3	4
1				
2		0	1	
3				
4				

Algorithme d'édition



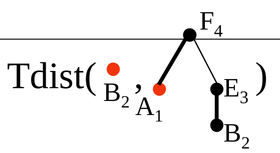
		A1	B2	E3	F4
B2	0	1	2	3	4
	1				



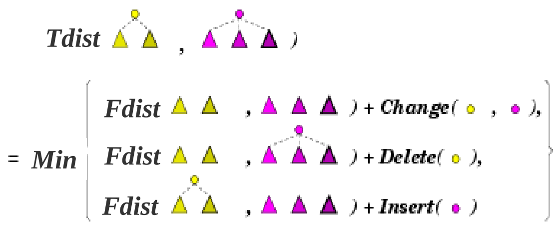


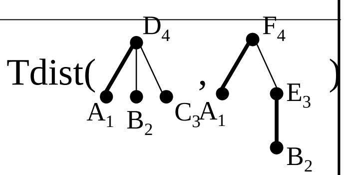
	1	2	3	4
1				
2		0	1	
3				
4				

Algorithme d'édition



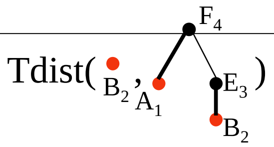
		A1	B2	E3	F4
B2	0	1	2	3	4
	1	1			



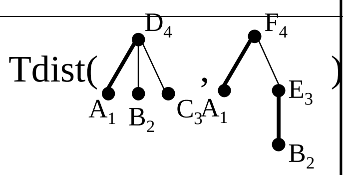


	1	2	3	4
1				
2	1	0	1	
3				
4				

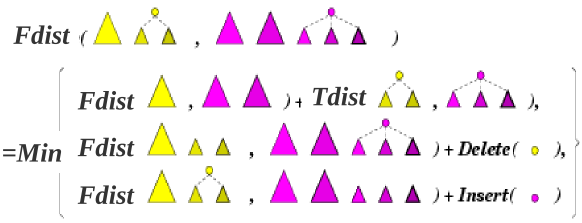
Algorithme d'édition



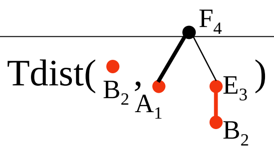
		A1	B2	E3	F4
	0	1	2	3	4
B2	1	1	1		



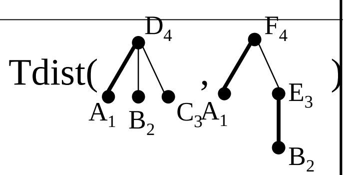
	1	2	3	4
1				
2	1	0	1	
3				
4				



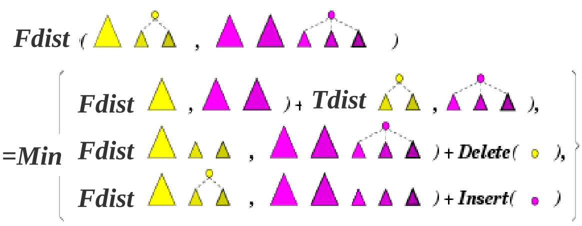
Algorithme d'édition



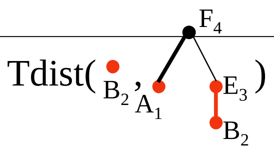
		A1	B2	E3	F4
	0	1	2	3	4
B2	1	1	1	2	



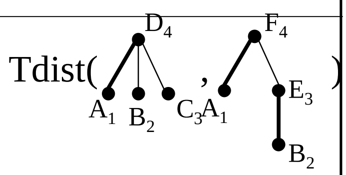
	1	2	3	4
1				
2	1	0	1	
3				
4				



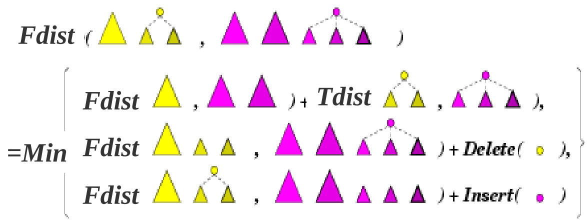
Algorithme d'édition



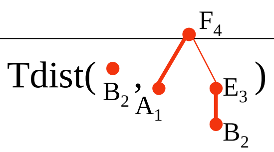
		A1	B2	E3	F4
B2	0	1	2	3	4
	1	1	1 ←	2	



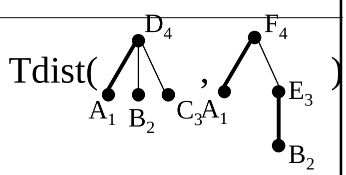
	1	2	3	4
1				
2	1	0	1	
3				
4				



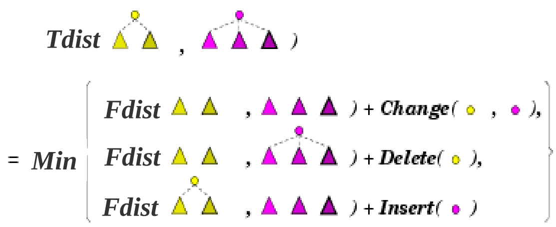
Algorithme d'édition



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



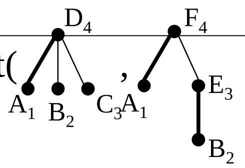
	1	2	3	4
1				
2	1	0	1	3
3				
4				



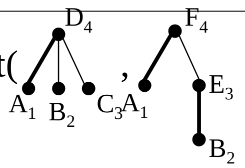
Algorithme d'édition

• • •

Algorithme d'édition

Tdist()

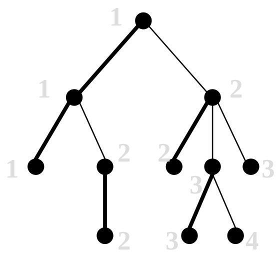
		A1	B2	E3	F4
	0	1	2	3	4
A1	1	0	1	2	3
B2	2	1	0	1	2
C3	3	2	1	2	3
D4	4	3	2	3	3

Tdist()

	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*).



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

Dans le pire des cas :

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

Complexité moyenne

[Dulucq, Tichit 2003]

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

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

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

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

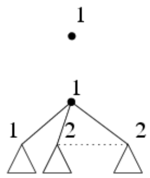
Où C_n désigne le nombre d’arbres à n+1 sommets :

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

Complexité moyenne

[Dulucq, Tichit 2003]

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



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

Complexité moyenne

[Dulucq, Tichit 2003]

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

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

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

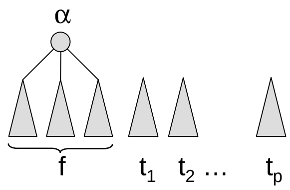
$\boxed{C_{edit} = HR(n) \times HR(m) \sim n^{3/2}\ m^{3/2}}$

Tree edition algorithm

Zhang, Shasha 1989

Score($\alpha(f)$, $\alpha'(f')$) = Max

$$\left\{\begin{array}{l} \mathrm{Subs}(\alpha, \alpha') + \mathrm{Score}(f,f') \\ \mathrm{Ins}(\alpha') + \mathrm{Score}(\alpha(f) , f') \\ \mathrm{Del}(\alpha) + \mathrm{Score}(f , \alpha'(f')) \end{array}\right.$$



Score($[\alpha(f) \circ t_1 \circ \ldots \circ t_p]$, $[\alpha'(f'), t'_1 \circ \ldots \circ t'_q]$) = Max

$$\left\{\begin{array}{l} \mathrm{Score}(\alpha(f), \alpha'(f')) + \mathrm{Score}([t_1 \circ \ldots \circ t_p], [t'_1 \circ \ldots \circ t'_q]) \\ \mathrm{Ins}(\alpha') + \mathrm{Score}([\alpha(f) \circ t_1 \circ \ldots \circ t_p] , [f', t'_1 \circ \ldots \circ t'_q]) \\ \mathrm{Del}(\alpha) + \mathrm{Score}([f \circ t_1 \circ \ldots \circ t_p] , [\alpha'(f') \circ t'_1, \ldots \circ t'_q]) \end{array}\right.$$

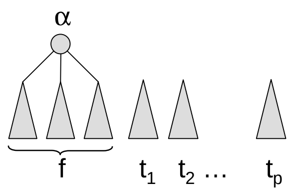
$O(n^3 \log n)$ [Klein 1998]

Tree alignment algorithm

Jiang, Wang, Zhang 1995

Score($r(f)$, $r'(f')$) = Max

$$\left\{\begin{array}{l} \mathrm{Subs}(\alpha, \alpha') + \mathrm{Score}(f,f') \\ \mathrm{Ins}(\alpha') + \mathrm{Score}(\alpha(f) , f') \\ \mathrm{Del}(\alpha) + \mathrm{Score}(f , \alpha'(f')) \end{array}\right.$$



Score($\alpha(f) \circ t_1 \circ \ldots \circ t_p ; \alpha'(f') \circ t'_1 \circ \ldots \circ t'_q$) = Max

$$\left\{\begin{array}{l} \mathrm{Score}(\alpha(f); \alpha'(f')) + \mathrm{Score}(t_1 \circ \ldots \circ t_p ; t'_1 \circ \ldots \circ t'_q) \\ \mathrm{Ins}(\alpha') + \mathrm{Max}_i \{ \mathrm{Score}(\alpha(f) \circ \ldots \circ t_i ; f') + \mathrm{Score}(t_{i+1} \circ \ldots \circ t_p ; t'_1 \circ \ldots \circ t'_q) \} \\ \mathrm{Del}(\alpha) + \mathrm{Max}_j \{ \mathrm{Score}(f ; \alpha'(f') \circ t'_1 \circ \ldots \circ t'_j) + \mathrm{Score}(t_1 \circ \ldots \circ t_p; t'_{j+1} \circ \ldots \circ t'_q) \end{array}\right.$$

$O(n^4 \log n)$

Edition vs Alignment

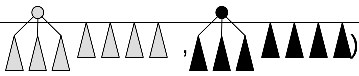
Score([r(f), $t_1,\ldots,t_p$] , [r'(f'), $t'_1,\ldots,t'_q$]) = Max

$$\left\{ \begin{array}{l} \ldots \\ \text{Ins}(r') + \text{Score}(\, [r(f), t_1,\ldots,t_p] , [\, f', t'_1,\ldots,t'_q]) \\ \ldots \end{array} \right.$$

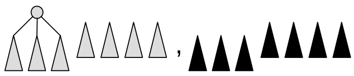
Score([r(f), $t_1,\ldots,t_p$] , [r'(f'), $t'_1,\ldots,t'_q$]) = Max

$$\left\{ \begin{array}{l} \ldots \\ \text{Ins}(r') + \text{Max} \{ \text{Score}(\, [r(f), \ldots t_i] , f') + \text{Score}([t_{i+1},\ldots, t_p], [t'_1,\ldots,t'_q]) \} \\ \ldots \end{array} \right.$$

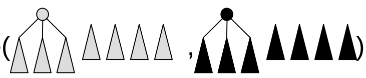
Edition vs Alignment

Score() = Max

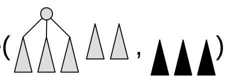
…

Ins(●) + Score()

…

Score() = Max

…

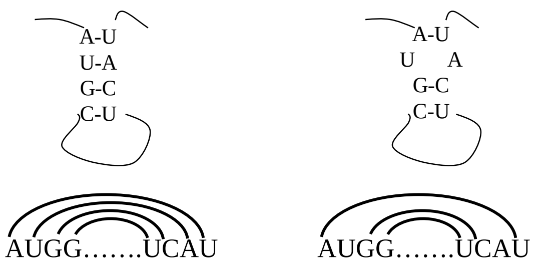
Ins(●) + Max { Score() + Score() }

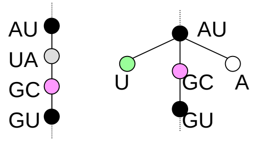
…

Retour à l'ARN

base-match	$\mathrm{A} \longrightarrow \mathrm{A}$
base-mismatch	$\mathrm{A} \longrightarrow \mathrm{G}$
base-deletion	$\mathrm{A} \longrightarrow -$ $- \longrightarrow \mathrm{A}$
arc-match	$\mathrm{A} \frown \mathrm{U} \longrightarrow \mathrm{A} \frown \mathrm{U}$
arc-mismatch	$\mathrm{A} \frown \mathrm{U} \longrightarrow \mathrm{G} \frown \mathrm{C}$
arc-removing	$\mathrm{A} \frown \mathrm{U} \longrightarrow - \quad -$ $- \quad - \longrightarrow \mathrm{A} \frown \mathrm{U}$
arc-breaking	$\mathrm{A} \frown \mathrm{U} \longrightarrow \mathrm{A} \quad \mathrm{U}$ $\mathrm{A} \quad \mathrm{U} \longrightarrow \mathrm{A} \frown \mathrm{U}$
arc-altering	$\mathrm{A} \frown \mathrm{U} \longrightarrow \mathrm{A} \quad -$ $\mathrm{A} \frown \mathrm{U} \longrightarrow - \quad \mathrm{U}$ $\mathrm{A} \quad - \longrightarrow \mathrm{A} \frown \mathrm{U}$ $- \quad \mathrm{U} \longrightarrow \mathrm{A} \frown \mathrm{U}$

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



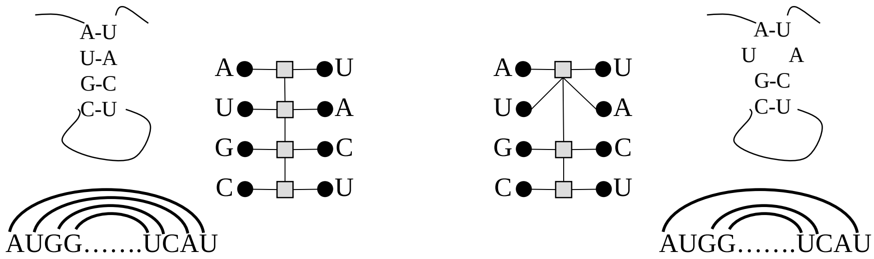


Delete( )
Insert( )
Insert( )

3 operations!

A first solution

Höchsmann, Töller, Gierich, Kurtz 2003
(RNAforester)

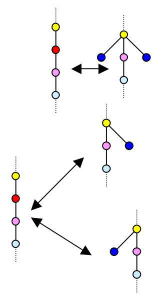


But this implies some constraints on the scores. For example:

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

New edition operations on trees

- Operations on bases:
  - Substitution: A → C
  - Deletion / Insertion: A → ∅
- Operations on arcs:
  - Arc-substitution: C G → U A
  - Arc-deletion / Arc-insertion: C G → ∅
  - Arc-breaking / : C G → C G
  - Arc-altering / : C G → C -



Tree edition, tree alignment

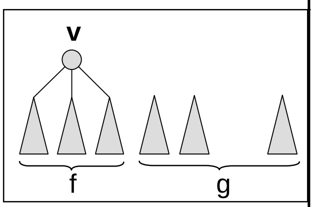
	Tree operations	RNA operations
Edition	$O(n^3 \log n)$ [Zhang-Shasha 1989, Klein 1998]	NP-complete [Blin, Fertin, Sinoquet, Rusu 2003]
Alignment	$O(n^4)$ [Jiang, Wang, Zhang 1995]	?

RNA structure alignment algorithm

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

$$\min \begin{cases} \mathrm{basedel}(v) + \mathrm{A}(g, v'(f')\circ g') & \text{if } v \text{ is a leaf} \\ \mathrm{baseins}(v') + \mathrm{A}(v(f)\circ g, g') & \text{if } v' \text{ is a leaf} \\ \mathrm{basesub}(v, v') + \mathrm{A}(g, g') & \text{if } v, v' \text{ are leaves} \\ \mathrm{pairdel}(v) + \min\{\mathrm{A}(f, p') + \mathrm{A}(g, s') \mid p' \circ s' = v'(f')\circ g'\} & \text{if } v \text{ is an internal node} \\ \mathrm{pairins}(v') + \min\{\mathrm{A}(p, f') + \mathrm{A}(s, g') \mid p \circ s = v(f)\circ g\} & \text{if } v' \text{ is an internal node} \\ \mathrm{pairsub}(v, v') + \mathrm{A}(f, f') + \mathrm{A}(g, g') & \text{if } v, v' \text{ are internal nodes} \\ \mathrm{scission}(v, v', v'_c) + \min\{\mathrm{A}(f, p') + \mathrm{A}(g, s') \mid p' \circ v'_c \circ s' = g'\} & \text{if } v \text{ is an internal node, } v', v'_c \text{ are leaves} \\ \mathrm{fusion}(v, v_c, v') + \min\{\mathrm{A}(p, f') + \mathrm{A}(s, g') \mid p \circ v_c \circ s = g\} & \text{if } v, v_c \text{ are leaves, } v' \text{ is an internal node} \\ \mathrm{leftalt}(v, v'_c) + \min\{\mathrm{A}(f, p') + \mathrm{A}(g, s') \mid p' \circ v'_c \circ s' = v'(f')\circ g'\} & \text{if } v \text{ is an internal node, } v'_c \text{ is a leaf} \\ \mathrm{rightalt}(v, v') + \min\{\mathrm{A}(f, p') + \mathrm{A}(g, s') \mid p' \circ s' = g'\} & \text{if } v \text{ is an internal node, } v' \text{ is a leaf} \\ \mathrm{leftcompl}(v_c, v') + \min\{\mathrm{A}(p, f') + \mathrm{A}(s, g') \mid p \circ v_c \circ s = v(f)\circ g\} & \text{if } v_c \text{ is a leaf, } v' \text{ is an internal node} \\ \mathrm{rightcompl}(v, v') + \min\{\mathrm{A}(p, f') + \mathrm{A}(s, g') \mid p \circ s = g\} & \text{if } v \text{ is a leaf, } v' \text{ is an internal node} \end{cases}$$

[Jiang, Wang, Zhang 1995]



Spécifique à l'ARN
[Blin, Denise, Dulucq, Herrbach, Touzet 2008]

Complexité au pire : $O(n^4)$

66

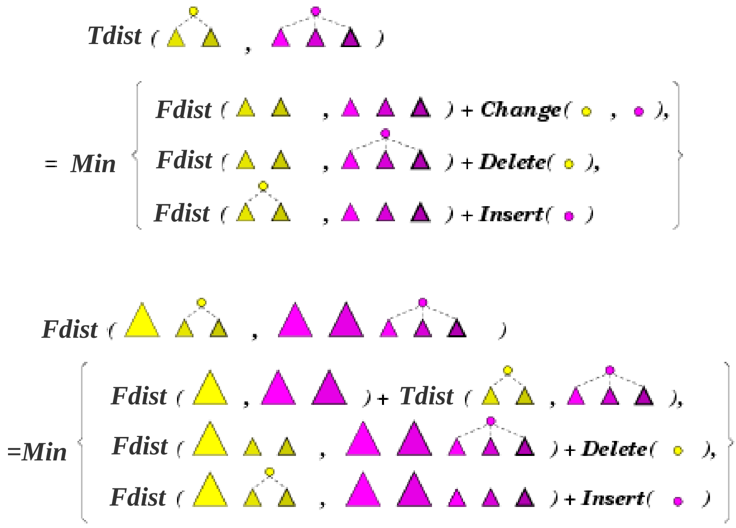
Complexité de la comparaison

	Arbres	ARN
Edition	Pire : $O(n^4)$ / $O(n^3 \log n)$ [Zhang, Shasha 1989 / Klein 1998] Moyenne : $O(n^3)$ [Dulucq, Tichit 2003]	NP-complet [Blin, Fertin, Sinoquet, Rusu 2003]
Alignement	Pire : $O(n^4)$ [Jiang, Wang, Zhang 1995] **Moyenne : $O(n^2)$** [Herrbach, Denise, Dulucq 2007]	**Pire : $O(n^4)$** [Blin, Denise, Dulucq, Herrbach, Touzet 2008] **Moyenne : $O(n^2)$** [Herrbach, Denise, Dulucq 2007]

Une approche multi-échelles

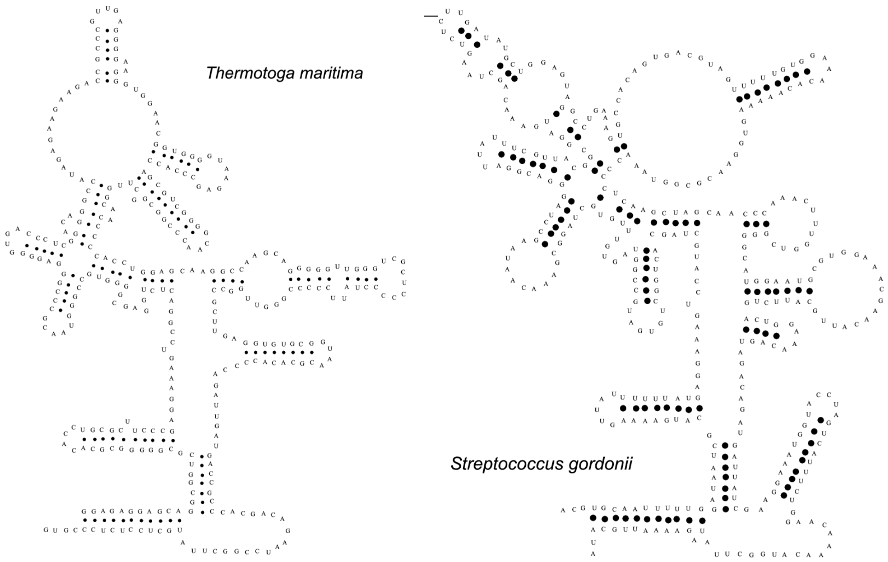
Algorithme d'édition

Zhang, Shasha 1989



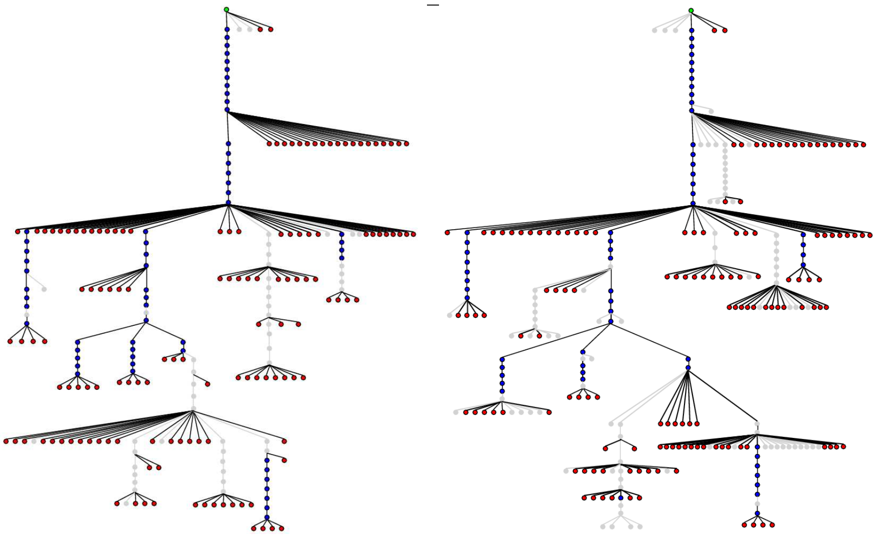
Application

Allali 2004



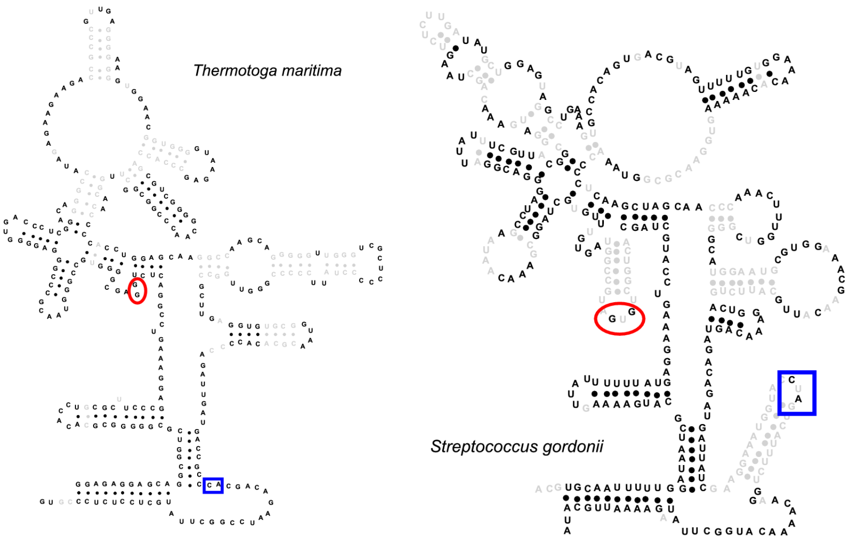
Application

Allali 2004



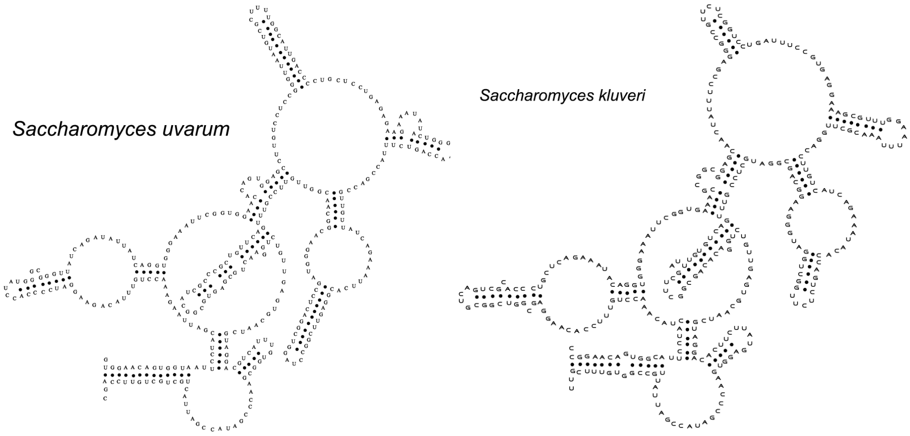
Application

Allali 2004



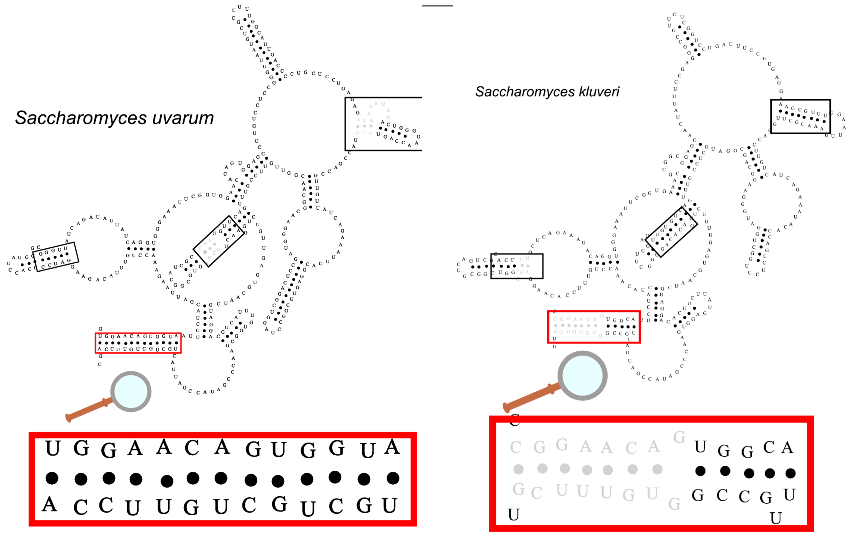
Application

Allali 2004



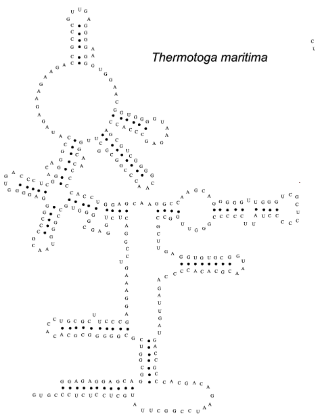
Application

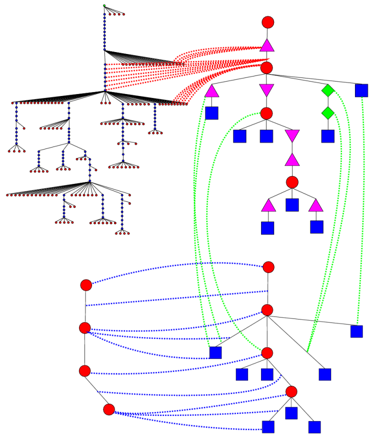
Allali 2004



Une approche multi-échelles

Allali, Sagot 2004

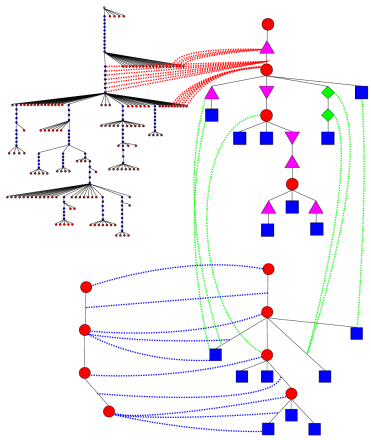




Une approche multi-échelles

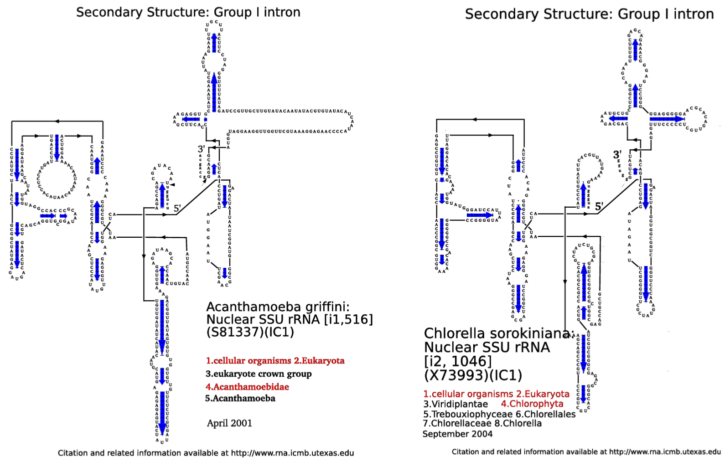
Allali, Sagot 2004

Opérations supplémentaires :
Fusion de nœuds
Fusion d'arêtes



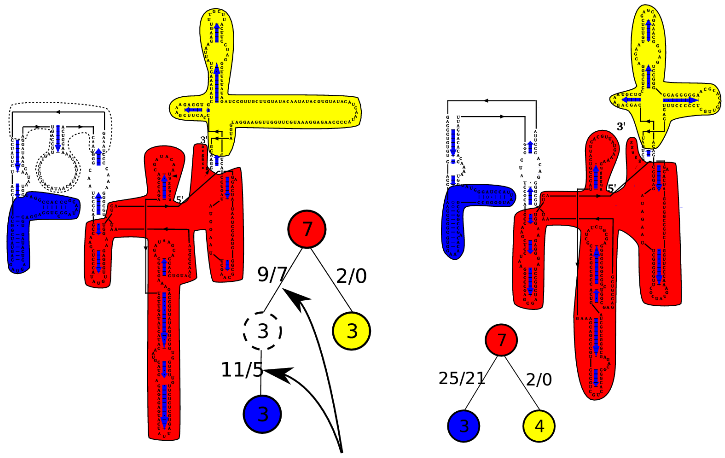
Une approche multi-échelles

Allali, Sagot 2004



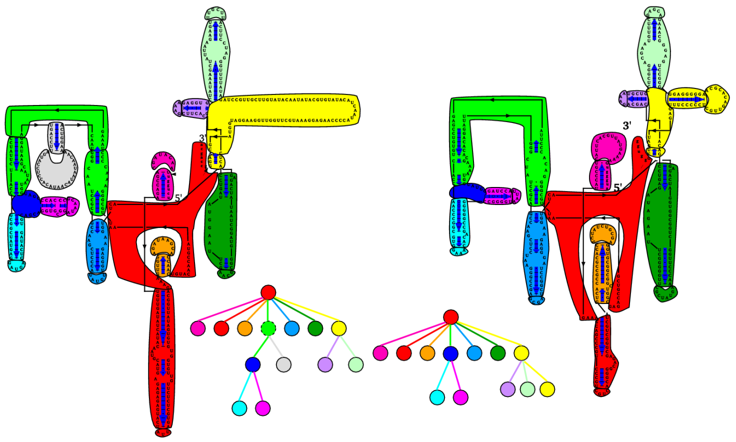
Une approche multi-échelles

Allali, Sagot 2004



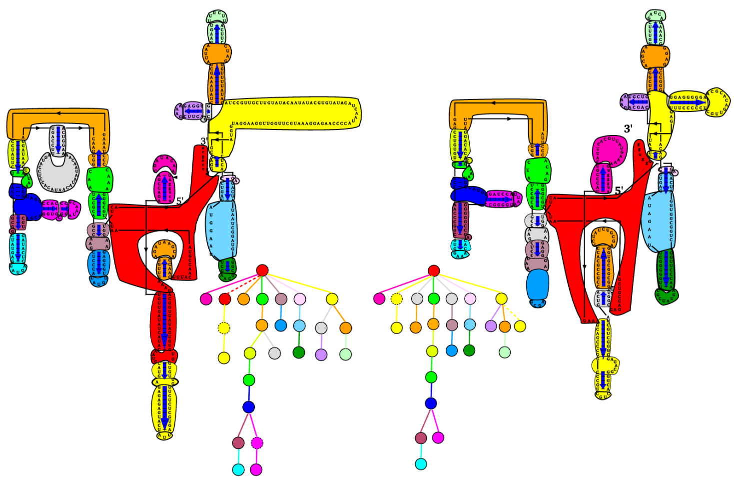
Une approche multi-échelles

Allali, Sagot 2004



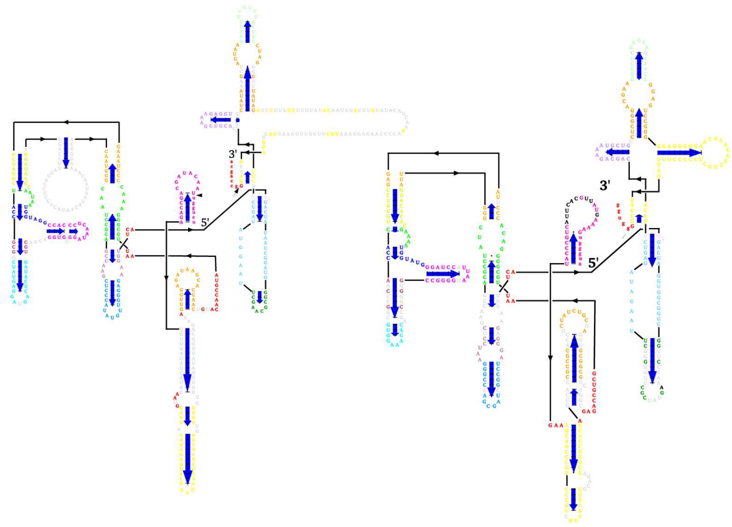
Une approche multi-échelles

Allali, Sagot 2004



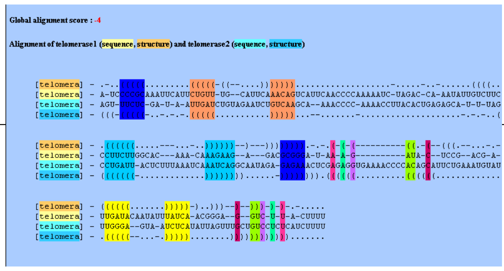
Une approche multi-échelles

Allali, Sagot 2004

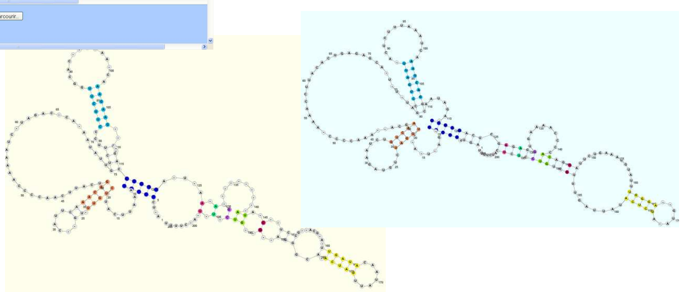


Logiciels

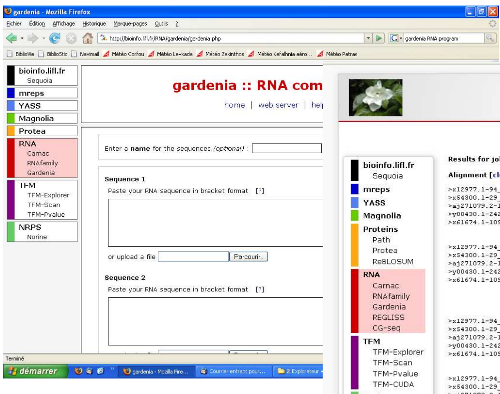
NestedAlign



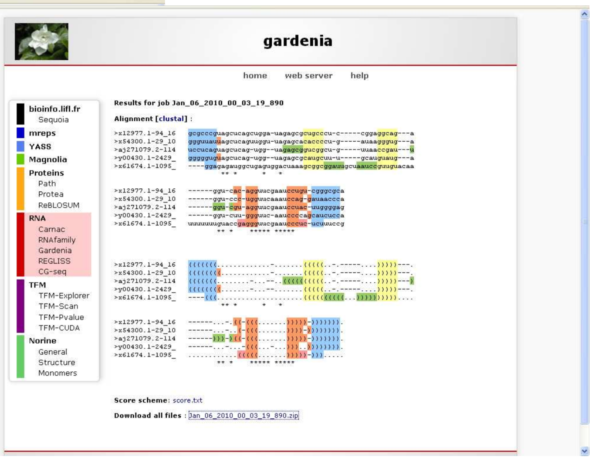
http://nestedalign.lri.fr



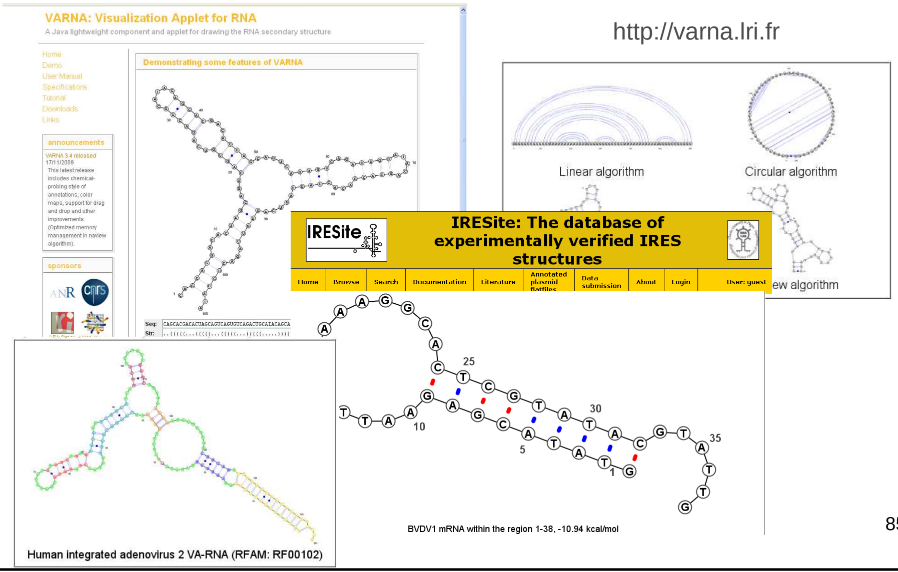
Gardenia



http://bioinfo.lifl.fr/RNA/gardenia/

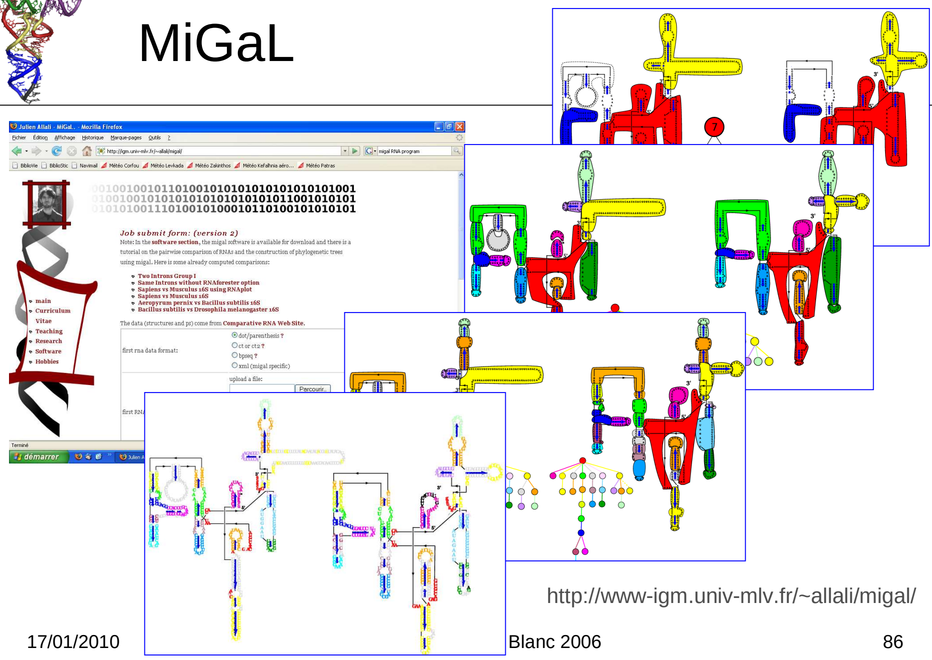


Varna



85

MiGaL



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

17/01/2010 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
Theoritical 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.

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
Theoritical 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.

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.