Alignement multiple

Nadia El-Mabrouk

Plan

1. Introduction
2. Solution exacte pour l'alignement multiple
3. Heuristique bornée
4. Alignement phylogénétique
5. Heuristiques usuelles: Méthodes progressives
6. Heuristiques usuelles: Méthodes itératives
7. Heuristiques usuelles: Méthodes par point d'ancrage

1. Introduction

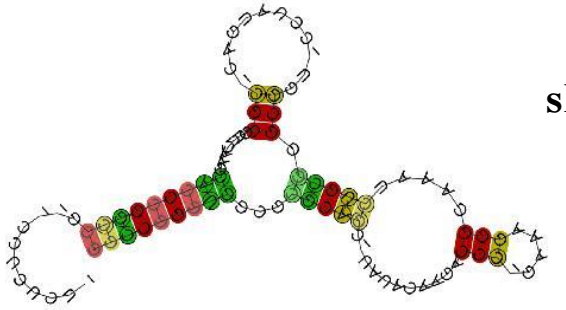
- Généralisation de l'alignement de 2 séquences
- Données: Un ensemble de séquence homologues (nucléotides ou AA): $S_1, S_2, \ldots, S_k$
- Alignement multiple: Matrice $A = (a_{ij}),\ 1 \leq i \leq k;\ 1 \leq j \leq l.$

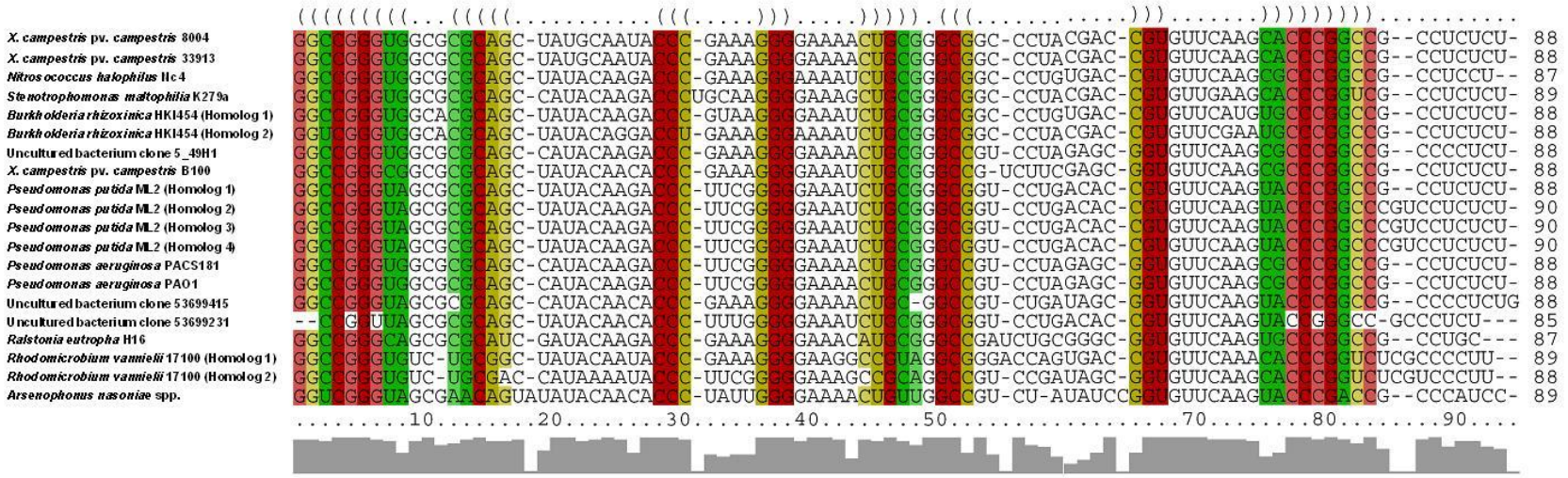
  a_{ij} symboles de l'alphabet ou '-', tq contaténation des caractères à la ligne i produit S_i.

$$\begin{bmatrix} A & A & G & A & A & - & A \\ A & T & - & A & A & T & G \\ C & T & G & - & G & - & G \\ C & C & - & A & G & T & T \\ C & C & G & - & G & - & - \end{bmatrix}$$

Autrement dit, si L est le nombre de colonnes de l'alignement multiple, on a : $\max_{1 \leq i \leq k} |Si| \leq L \leq \sum_{1 \leq i \leq k} |Si|$

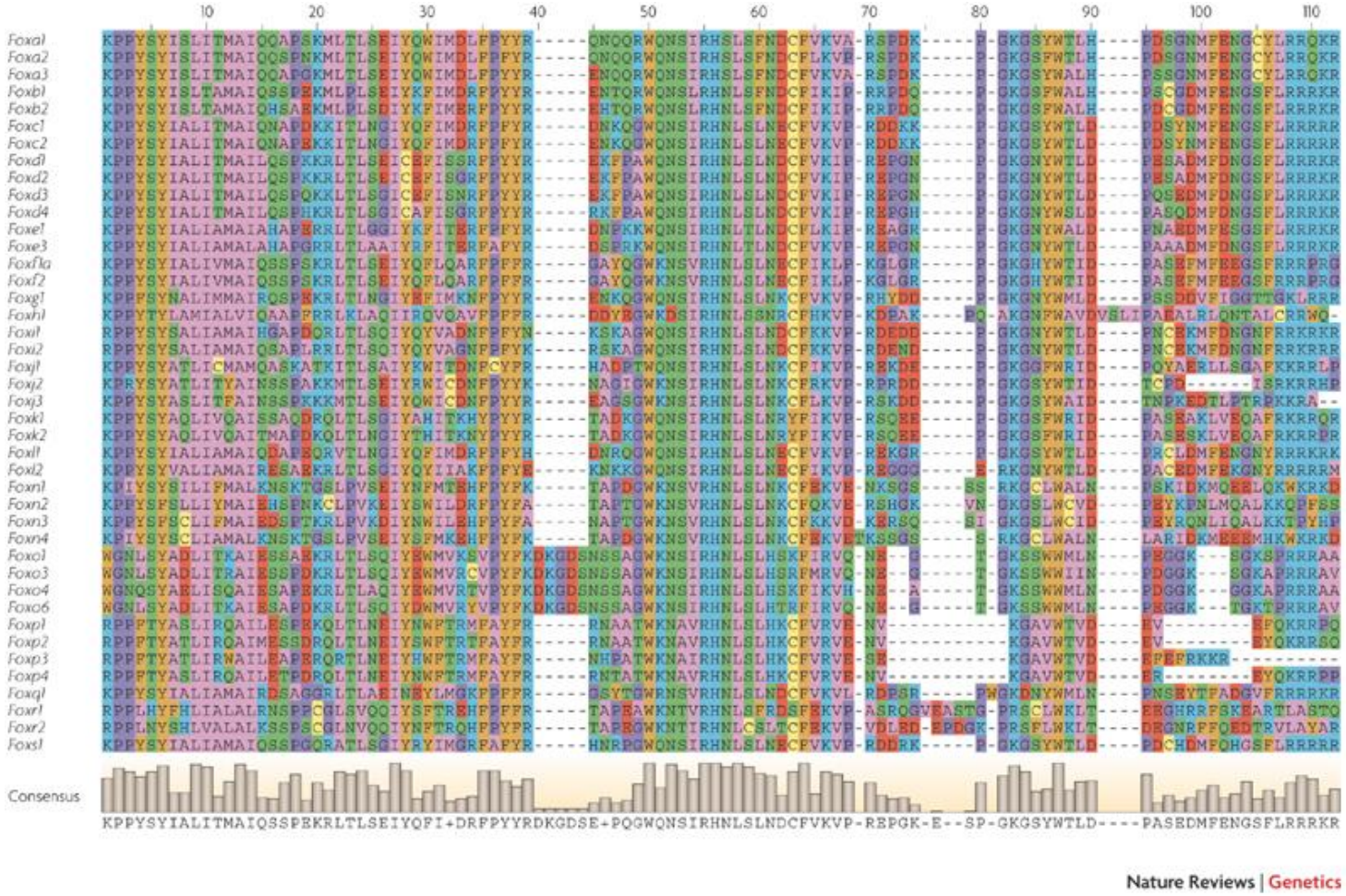
Exemple: Alignement multiple d'ARN





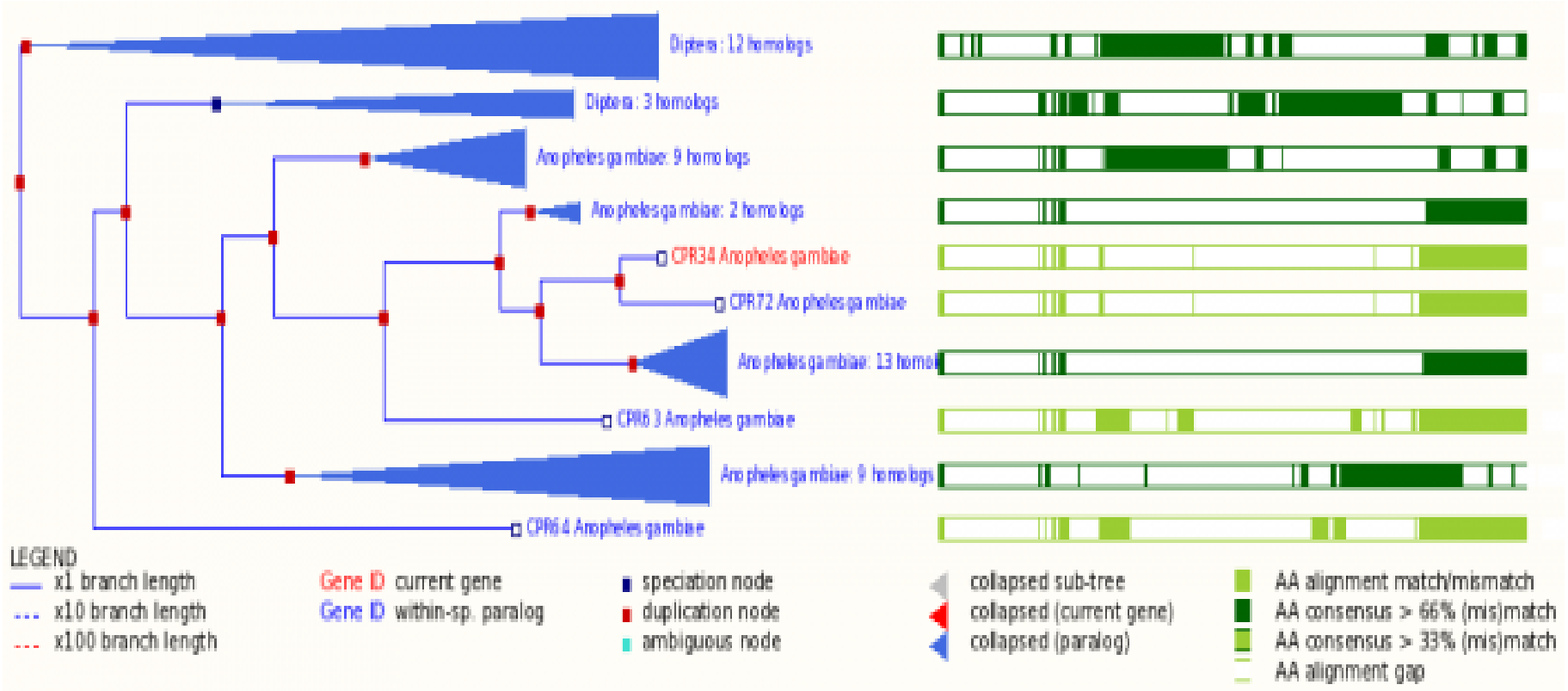
http://en.wikipedia.org/wiki/File:Multiple_alignment_and_the_consensus_secondary_structure_model_of_the_sRNA-Xcc1_homologs.jpg

Alignement multiple d'acides aminés



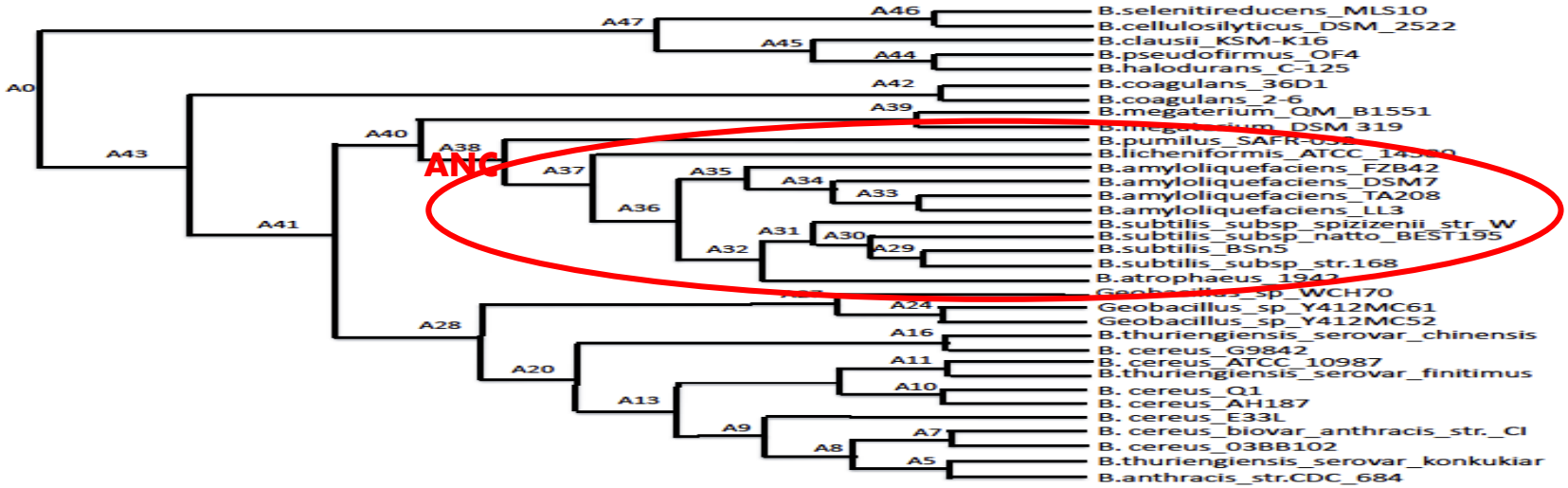
Alignment of forkhead box (Fox) genes in mice.

Alignement multiple de gènes



Base de donnée Ensembl contient plus de 100,000 arbres de gènes

Alignement de génomes



B.subtilis

```
Anc Ile Ala Ser Ile Ala Met Glu Val Thr Lys Leu Gly Leu Arg Pro Ala Asn Thr ---------- Glu Val Thr Tyr Gln Asn Ser Glu Gln Lys Leu Leu Arg Gly ------- --- Met Asp ------- Asn Ser Glu Val Met ---------- Asp Phe Thr Tyr Trp
(1)  Ile Ala Ser Ile Ala Met Glu Val Thr Lys Leu Gly Leu Arg Pro Ala Asn Thr Gly Arg Pro Ala ------- Glu Val Thr Tyr Gln Asn Ser Glu Gln Lys Leu Leu Arg Gly ------- --- Met Asp ------- Asn Ser Glu Val Met Asp Asn Ser Glu Val Met Asp Phe Thr Tyr Trp
(2)  Ile Ala Ser Ile Ala Met Glu ---------- Asn Thr Gly Arg Pro Ala ------- Glu Val Thr Tyr Gln Asn Ser Glu Gln Lys Leu Leu Arg Gly ------- --- Met Asp ------- Asn Ser Glu Val Met Asp Asn Ser Glu Val Met Asp Phe Thr Tyr Trp
(3)  Ile Ala Ser Ile Ala Met Glu Val Thr Lys Leu Gly Leu Arg Pro Ala Asn Thr Gly Arg Pro Ala ------- Glu Val Thr Tyr Gln Asn Ser Glu Gln Lys Leu Leu Arg Gly ------- --- Met Asp ------- Asn Ser Glu Val Met Asp Asn Ser Glu Val Met Asp Phe Thr Tyr Trp
(4)  Ile Ala Ser Ile Ala Met Glu ---------- Asn Thr Gly Arg Pro Ala ------- Glu Val Thr Tyr Gln Asn Ser Glu Gln Lys Leu Leu Arg Gly ------- --- Met Asp ------- Asn Ser Glu Val Met ---------- Asp Phe Thr Tyr Trp
(5)  Ile Ala Ser Ile Ala Met Glu ---------- Asn Thr ---------- Glu Val Thr Tyr Gln Asn Ser Glu Gln Lys Leu Leu Arg Gly ------- --- Met Asp ------- Asn Ser Glu Val Met ---------- Asp Phe Thr Tyr Trp
(6)  Ile Ala Ser Ile Ala Met Glu ---------- Glu Val Thr Tyr Gln Asn Ser Glu Gln Lys Leu Leu Arg Gly ------- --- Met Asp ------- Asn Ser Glu Val Met ---------- Asp Phe Thr Tyr Trp
(7)  ------- Ser Ile Ala Met Glu ---------- Glu Val Thr Tyr Gln Asn Ser Glu Gln Lys Leu Leu Arg Gly ------- --- Met Asp ------- Asn Ser Glu Val Met ---------- Asp Phe Thr Tyr Trp
(8)  Ile Ala Ser Ile Ala Met Glu Val Thr Lys Leu Gly Leu Arg Pro Ala Asn Thr Gly Arg Pro --- Ile Ala Glu Val Thr Tyr Gln Asn Ser Glu Gln Lys Leu Leu Arg Gly Ile Ala --- Met Asp Ile Ala Asn Ser Glu Val Val ---------- Asp Phe Thr Tyr Trp
(9)  ------- Ser ------- Met Glu Val Thr Lys Leu Gly Leu Arg Pro Ala Asn Thr Gly Arg Pro Ala Ile Ala Glu Val Thr Tyr Gln Asn Ser Glu Gln Lys Leu Leu Arg Gly ------- --- Met Asp ------- Asn Ser Glu Val Met ---------- Asp Phe Thr Tyr Trp
(10) Ile Ala Ser Ile Ala Met Glu ---------- Asn Thr ---------- Glu Val Thr Tyr Gln Asn Ser Glu Gln Lys Leu Leu Arg Gly ------- --- Met Asp ------- Asn Ser Glu Val Met ---------- Asp Phe Thr Tyr Trp
(11) ---------- Met Glu ---------- Asn ---------- Glu Val Thr Tyr Gln Asn Ser Glu Gln Lys Leu Leu Arg Gly Ile Ala Ser Met Asp ------- Asn Ser Glu Val Met ---------- Asp Phe Thr Tyr Trp
```

```
Anc   Trp His Gln Gly Cys Leu Leu Gly Val [t] -Arg ---------- Gln -Arg ---  -Glu -Ser -Asn -Ile -Gly -His -Phe -Asp -Met -Ser -Met ---------- -Met -Ala -Pro -Arg -Leu -Gly -Leu -Lys -Thr -Val --------  -Ala -Arg -Phe -Asp -Glu -Lys
(1)   Trp His Gln Gly Cys Leu Leu Gly Val [t] -Arg ---------- Gln -Arg ---  -Glu -Ser -Asn -Ile -Gly -His -Phe -Asp -Met -Ser -Met ---------- -Met -Ala -Pro -Arg -Leu -Gly -Leu -Lys -Thr -Val --------  -Ala -Arg -Phe -Asp -Glu -Lys
(2)   Trp His Gln Gly Cys Leu Leu Gly Val [t] -Arg ---------- Gln -Arg ---  -Glu -Ser -Asn -Ile -Gly -His -Phe -Asp -Met -Ser -Met -Phe -Asp -Met -Ser -Met -Met -Ala -Pro -Arg -Leu -Gly -Leu -Lys -Thr -Val -Ala -Ile -Ala -Arg -Phe -Asp -Glu -Lys
(3)   Trp His Gln Gly Cys Leu Leu Gly Val [t] -Arg ---------- Gln -Arg ---  -Glu -Ser -Asn -Ile -Gly -His -Phe -Asp -Met -Ser -Met ---------- -Met -Ala -Pro -Arg -Leu -Gly -Leu -Lys -Thr -Val --------- -Ala -Arg -Phe -Asp -Glu -Lys
(4)   Trp His Gln Gly Cys Leu Leu Gly Val [t] -Arg ---------- Gln -Arg ---  -Glu -Ser -Asn -Ile -Gly -His -Phe -Asp -Met -Ser -Met ---------- -Met -Ala -Pro -Arg -Leu -Gly -Leu -Lys -Thr -Val --------- -Ala -Arg -Phe -Asp -Glu -Lys
(5)   Trp His Gln Gly Cys Leu Leu Gly Val [t] -Arg -Trp -Arg  Gln -Arg ---  -Glu -Ser -Asn -Ile -Gly -His -Phe -Asp -Met -Ser -Met ---------- -Met -Ala -Pro -Arg -Leu -Gly -Leu -Lys -Thr -Val --------- -Ala -Arg -Phe -Asp -Glu -Lys
(6)   Trp His Gln Gly Cys Leu Leu Gly Val [t] -Arg ---------- Gln -Arg -Arg -Glu -Ser -Asn -Ile -Gly -His -Phe -Asp -Met -Ser -Met ---------- -Met -Ala -Pro -Arg -Leu -Gly -Leu -Lys -Thr -Val --------- -Ala -Arg -Phe -Asp -Glu -Lys
(7)   Trp His Gln Gly Cys Leu Leu Gly Val [t] -Arg ---------- Gln -Arg -Arg -Glu -Ser -Asn -Ile -Gly -His -Phe -Asp -Met -Ser -Met ---------- -Met -Ala -Pro -Arg -Leu -Gly -Leu -Lys -Thr -Val --------- -Ala -Arg -Phe -Asp -Glu -Lys
(8)   Trp His Gln Gly Cys Leu Leu Gly Val [t] -Arg ---------- Gln -Arg -Arg -Glu -Ser -Asn -Ile -Gly -His -Phe -Asp -Met -Ser -Met ---------- -Met -Ala -Pro -Arg -Leu -Gly -Leu -Lys -Thr -Val -Ala -Ile -Ala -Arg -Phe -Asp -Glu -Lys
(9)   Trp His Gln Gly Cys Leu Leu Gly Val [t] -Arg ---------- Gln -Arg -Arg -Glu -Ser -Asn -Ile -Gly -His -Phe -Asp -Met -Ser -Met ---------- -Met -Ala -Pro -Arg -Leu -Gly -Leu -Lys -Thr -Val -Ala -Ile -Ala -Arg -Phe -Asp -Glu -Lys
(10)  Trp His Gln Gly Cys Leu Leu Gly -Val[t] -Arg ---------- Gln -Arg ---- -Glu -Ser -Asn -Ile -Gly -His -Phe -Asp -Met -Ser -Met ---------- -Met -Ala -Pro -Arg  --- -Gly -Leu -Lys -Thr -Val --------- -Ala -Arg -Phe -Asp -Glu -Lys
(11)  Trp His Gln Gly Cys Leu Leu Gly Val [t]  ---------- Gln -Arg ---- -Glu -Ser -Asn -Ile -Gly -His -Phe -Asp -Met -Ser -Met ---------- -Met -Ala -Pro -Arg -Leu -Gly -Leu -Lys -Thr -Val --------- -Ala -Arg -Phe -Asp -Glu -Lys
```

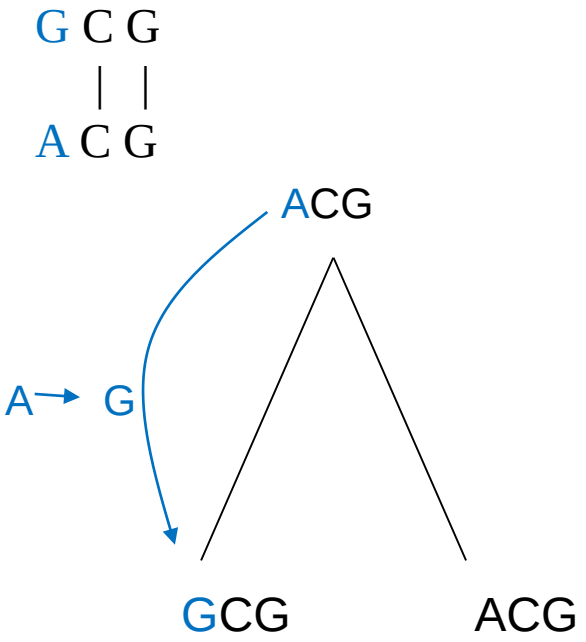
But de l'alignement multiple

- Trouver des contraintes de structures pour les ARN
- Trouver des caractéristiques communes à une famille de protéines;
- Caractériser les régions conservées et les régions variables
- Relier la séquence à la structure et à la fonction
- Reconstuire des phylogénies
  - Sélectionner des séquences homologues
  - Trouver un alignement multiple
  - L'utiliser pour construire l'arbre phylogénétique.
- Inférer des scénarios d'évolution

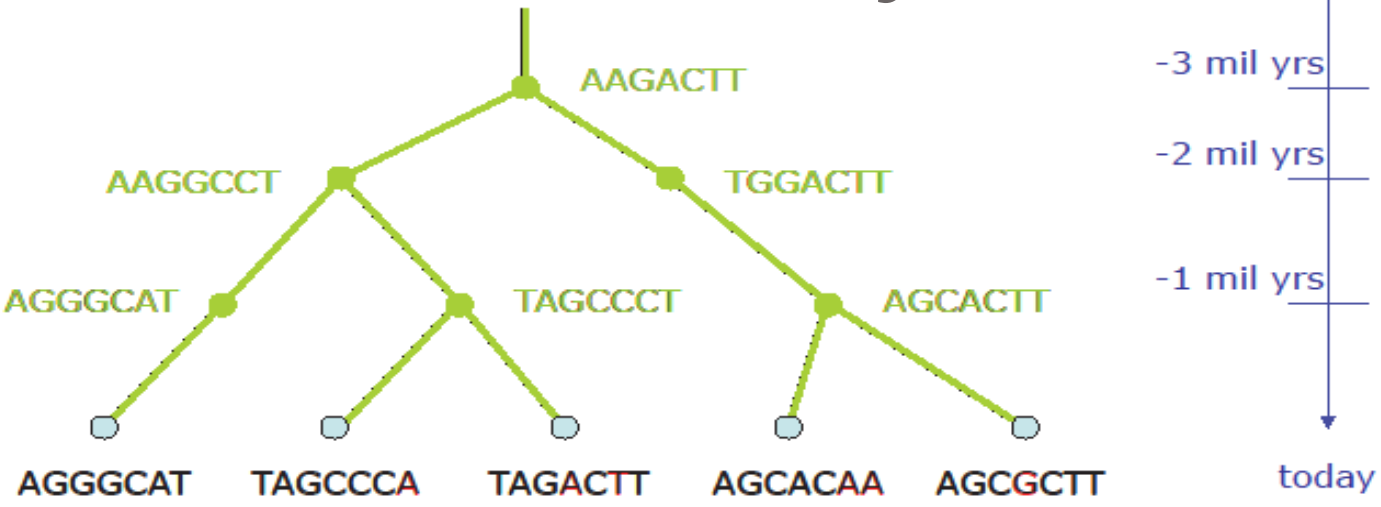
En résumé, l'alignement multiple joue un rôle central pour comprendre les relations entre fonction, évolution, séquence et structure d'une famille de gènes ou de protéines.

Modèle évolutif sous-jacent

- Un *bon* alignement reflète le modèle d'évolution qui a donné lieu aux séquences
- Hypothèses:
  - les séquences à aligner descendent d'un ancêtre commun
  - Les séquences ont évolué par mutations ponctuelles



Modèle évolutif sous-jacent



Alignement multiple induit:

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

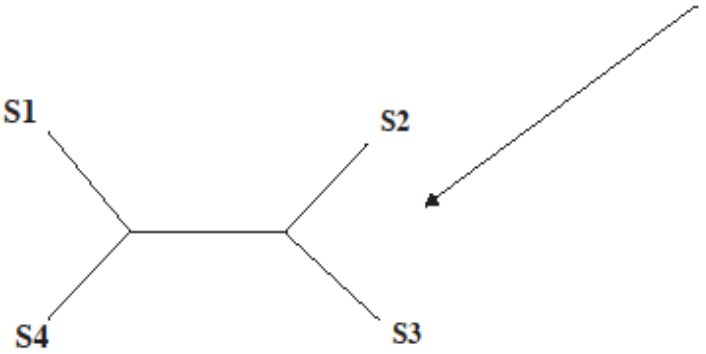
Présentation de Tandy Warnow à MAGE
http://www-etud.iro.umontreal.ca/~lafonman/MAGE2013/slides/Tandy-Warnow-MAGE.pdf

Retrouver la phylogénie

```
S1 = AGGCTATCACCTGACCTCCA
S2 = TAGCTATCACGACCGC
S3 = TAGCTGACCGC
S4 = TCACGACCGACA
```

→

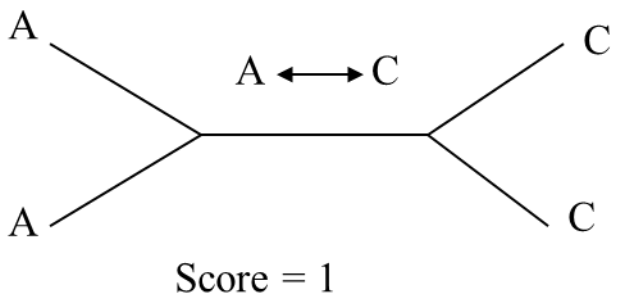
```
S1 = -AGGCTATCACCTGACCTCCA
S2 = TAG-CTATCAC--GACCGC--
S3 = TAG-CT-------GACCGC--
S4 = -------TCAC--GACCGACA
```



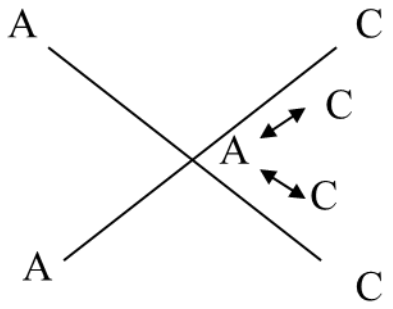
Présentation de Tandy Warnow à MAGE
http://www-etud.iro.umontreal.ca/~lafonman/MAGE2013/slides/Tandy-Warnow-MAGE.pdf

Pondération d'un alignement multiple

- Par rapport à l'arbre phylogénétique produit. Garder l'alignement qui produit l'arbre de poids minimal. Complexité de calcul considérable



Score = 1



Score=2

Pondération d'un alignement multiple

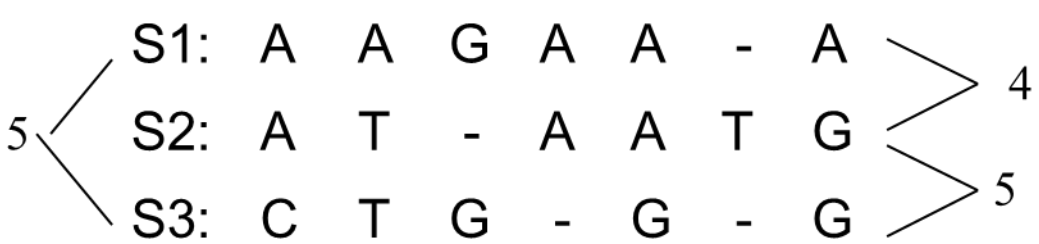
- Presque toutes les méthodes de pondération prennent pour hypothèse l'indépendance statistique des colonnes A_i d'un l'alignement A.
- Fonctions objectives généralement sous la forme:

$$f(A) = G + \sum_i f(Ai)$$

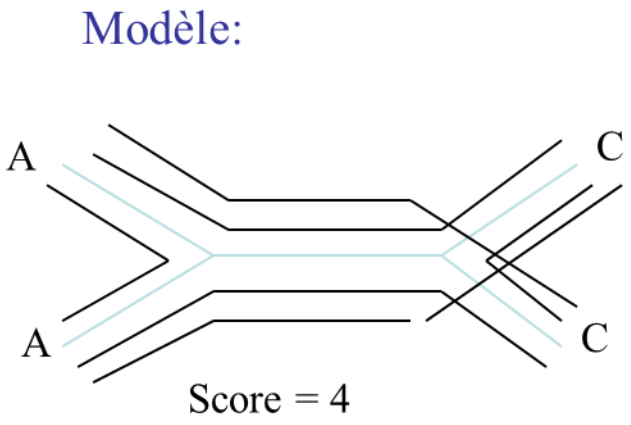
- G est une fonction de pondération des indels. La plus simple façon de procéder est de considérer un espace (-) comme un caractère supplémentaire de l'alphabet des nucléotides, et dans ce cas $f(A) = \sum_i f(Ai)$. Cependant la pondération affine (coût supérieur pour l'ouverture d'un « gap » que pour son élongation) est souvent utilisée.

Score « sum of pairs » (SP)

- Généralisation du score utilisé pour l'alignement de deux séquences
- Le plus utilisé, bonnes propriétes théoriques et pratiques
- Score SP d'un alignement A = somme des scores des alignements induits pour chaque paire de séquences dans A



Score SP = 14



Score = 4

Plan

1. Introduction
2. Solution exacte pour l'alignement multiple
3. Heuristique bornée
4. Alignement phylogénétique
5. Heuristiques usuelles: Méthodes progressives
6. Heuristiques usuelles: Méthodes itératives
7. Heuristiques usuelles: Méthodes par point d'ancrage

2. Solution exacte

Alignement multiple pour le score SP

- Trouver un alignement multiple ayant un score SP minimum.
- Problème NP-complet *(Wang and Jiang 1994)*
- Généralisation de l'alignement de deux séquences: si m séquences de taille n, algorithme en $O(n^m)$. Très inefficace dès que $m>5$ et $n\sim100$

Solution exacte pour n=3

- On considère la distance d'édition avec pondération de l'alphabet.
- S, T, U trois seq. de tailles n_1, n_2, n_3
- $D(i,j,k)$: Score SP de l'al. op. de $S[1..i]$, $T[1..j]$, $U[1..k]$;
- b: score d'un indel;
- $c(i,j)$: sore de l'appariement $(S[i], T[j])$.

Solution exacte pour n=3

- Pour chaque case (i,j,k), examiner 7 cases voisines:
  - $d1 = D(i-1,j-1,k-1)+c(i,j)+c(i,k)+c(j,k)$
  - $d2 = D(i-1,j-1,k)+c(i,j)+2b;$
  - $d3 = D(i-1,j,k-1)+c(i,k) +2b;$
  - $d4 = D(i,j-1,k-1)+c(j,k)+2b$
  - $d5 = D(i-1,j,k)+2b$;
  - $d6=D(i,j-1,k)+2b;$
  - $d7=D(i,j,k-1)+2b.$
- $D(i,j,k) = min(d1,d2,d3,d4,d5,d6,d7)$
- $D_{ST}(i,j)$: Score de l'al. Op. de $S[1..i]$ et $T[1..j]$;
  - $D(i,j,0) = D_{ST}(i,j) +(i+j)b;$
  - $D(i,0,k) = D_{SU}(i,k)+(i+k)b;$
  - $D(0,j,k) = D_{TU}(i,k)+(i+k)b$

Pour m quelconque

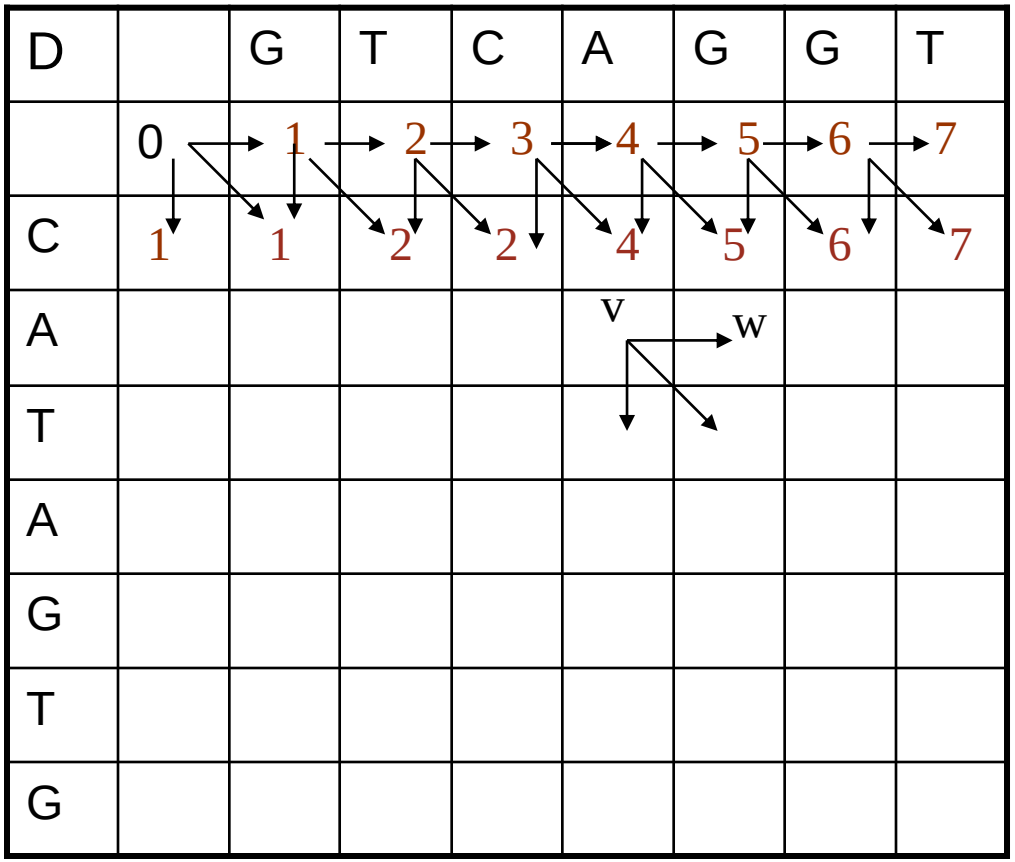
- $2^m - 1$ combinaisons possibles
- Pour des séquences de taille n
  - Espace $O(n^m)$
  - Temps en $O(2^m n^m)$

Optimisation: Algorithme de Carillo & Lipman 1988

Implémenté dans MSA (Lipman, Altschul & Kececioglu 1989)

- Calculer les alignements optimaux pour chaque paire de séquences;
- Trouver un alignement multiple provisoire par une heuristique rapide: *z*
- Effectuer la programmation dynamique en scrutage avant dans un espace d'alignement restreint.

Programmation dynamique avec scrutage avant



Les flèches vont de *(i,j) à (i,j+1), (i+1,j) et (i+1,j+1)*

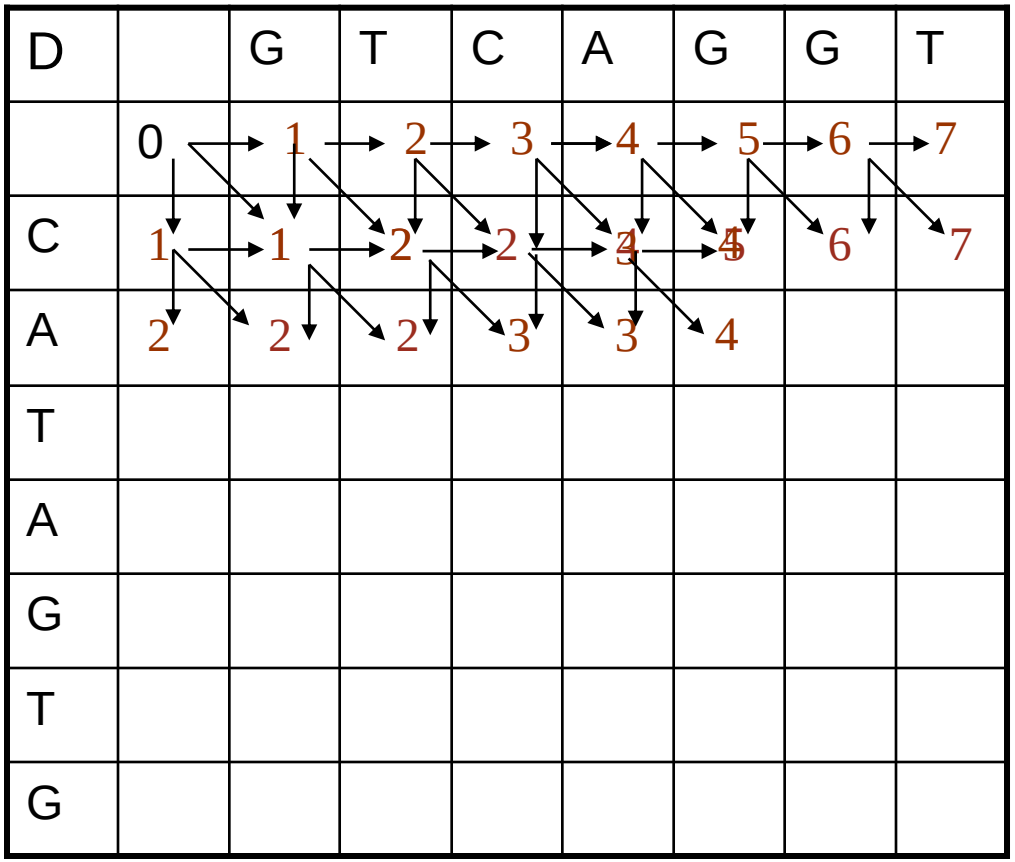
p(v,w): Poids de la flèche de *v* à *w*

p(w): Valeur provisoire de *D(w)*. Après calcul de *D(v)*:

p(w) = min(p(w), D(v)+p(v,w))

Valeur de *D(w)* = valeur de p(w) après considération de tous les voisins de w

Programmation dynamique avec scrutage avant



Les flèches vont de *(i,j) à (i,j+1), (i+1,j) et (i+1,j+1)*

p(v,w): Poids de la flèche de *v* à *w*

p(w): Valeur provisoire de *D(w)*. Après calcul de *D(v)*:

p(w) = min(p(w), D(v)+p(v,w))

Valeur de *D(w)* = valeur de p(w) après considération de tous les voisins de w

Algorithm:

- $q=(0,0)$ (liste contenant les cases à considérer)
- Tant que q n'est pas vide faire

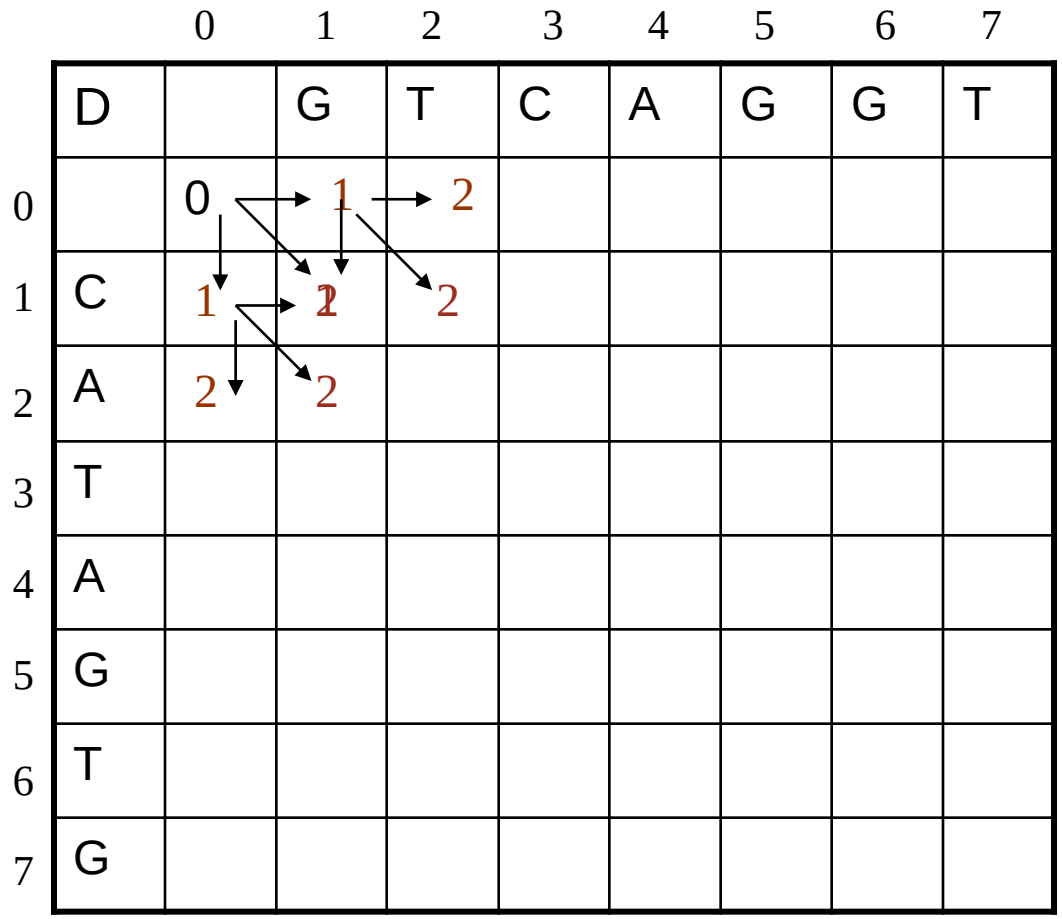
  $v = (i,j)$: première case de q;

  Supprimer v de q; $D(v)=p(v)$;

  Si $w=(i,j+1)$ pas dans q, le rajouter a la fin de q;

  $p(w)=min(p(w),D(v)+p(v,w))$;

  Même chose pour $w=(i+1,j)$ et $w=(i+1,j+1)$



q: ~~(0,0)~~ ~~(0,1)~~ ~~(1,0)~~ (1,1) (0,2) (1,2) (2,0) (2,1)

Accélération de l'alignement SP exact

- $ID_{ST}(i,j)$: Score de l'al. Op. de S*[i..n]* et *T[j..n]*.
  Définition similaire pour $ID_{SU}(i,k)$ et $ID_{TU}(j,k)$.
- z = score d'UN alignement multiple de *S, T, U*

Observation:

- Score SP pour *S[i..n], T[j..n], U[k..n]*
  supérieur ou égal à
  $$ID_{ST}(i,j) + ID_{SU}(i,k) + ID_{TU}(j,k)$$

- Si $D(i,j,k) + ID_{ST}(i,j) + ID_{SU}(i,k) + ID_{TU}(j,k) > z$, alors (i,j,k) ne peut pas faire partie d'un chemin optimal

- Aucun scrutage avant n'est nécessaire pour (i,j,k). Plus important, certaines cases ne sont jamais introduites dans la liste q.

Observation ampirique: Cette méthode peut aligner efficacement jusqu'à 6 séquences de longueur 200. Efficacité dépend beaucoup de la valeur z initiale

Plan

1. Introduction
2. Solution exacte pour l'alignement multiple
3. Heuristique bornée
4. Alignement phylogénétique
5. Heuristiques usuelles: Méthodes progressives
6. Heuristiques usuelles: Méthodes itératives
7. Heuristiques usuelles: Méthodes par point d'ancrage

3. Heuristique bornée pour le score SP

- Heuristique: Algorithme qui n'est pas garanti d'obtenir la solution optimale. Utilisé pour des problèmes difficiles (NP-complet).
- Heuristique bornée: On sait dans quel intervalle se situe la solution.
- Heuristique pour le score SP: Algorithme garanti d'obtenir un alignement dont le score est au plus deux fois plus élevé que le score d'un alignement optimal.

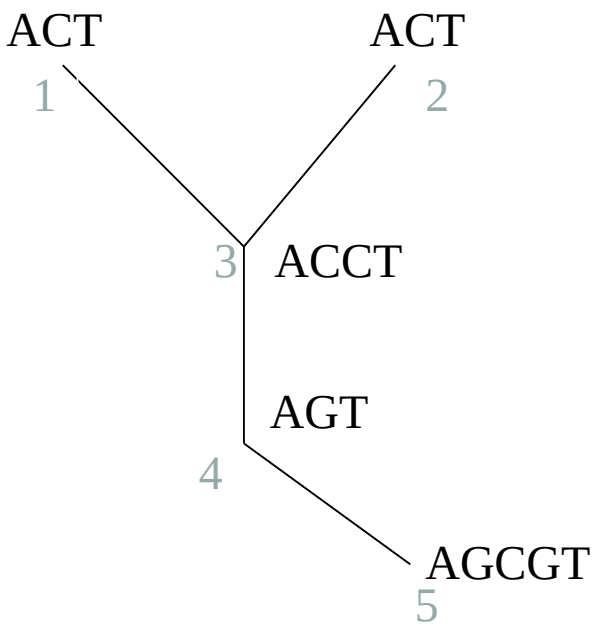
Alignement « consistant » avec un arbre

S: Ensemble de séquences;

T: Arbre « guide » reliant les séq. de ***S***

A: Alignement multiple de ***S***

A « consistant » avec ***T*** ssi: pour tout couple de séquences S_i, S_j reliées par une arête, S_i et S_j sont alignées de façon optimale dans *A*.



3: A C C - T
1: A C - - T
2: A - C - T
4: A G - - T
5: A G C G T

Méthode

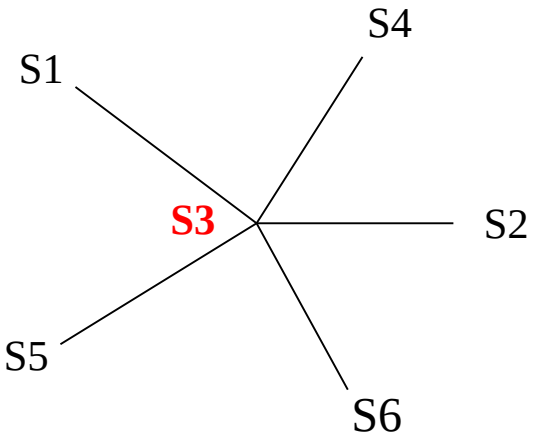
- Choisir deux séquences quelconques adjacentes dans l'arbre et former un alignement optimal A;
- Choisir une séquence non encore alignée S_i, adjacente à une séquence alignée S_j
- Aligner S_i et S_j.
- Incorporer l'alignement à A.
  - Si un nouvel indel a été rajouté dans S_j, rajouter un espace à chaque ligne à la colonne correspondante dans A

Complexité: k séquences de taille n , $O(kn^2)$

Arbre étoile

S: ensemble de séquences

- Séquence centrale S_c: Séquence de ***S*** tq la somme des distances à toutes les autres séquences de ***S*** est minimale.
- Arbre étoile T_c: Arbre en étoile, connectant toutes les séquences de ***S***, et de racine S_c.



S={S1,S2,S3,S4,S5,S6}

Trouver un Alignement « consistant » avec l'arbre étoile

k = nb de séquences, n = taille de chaque séquence

Complexité:

- Trouver la séquence centrale S_c:

  $O(k^2n^2)$

- Alignement A_c consistant avec T_c:

  $O(kn^2)$

Bornes

- $d(A)$: Score SP d'un alignement multiple A
- A_c : Alignement « consistant » avec l'arbre étoile
- $d_c(Si,Sj)$: Score induit par A_c pour S_i, S_j
- $D(Si,Sj)$: Score d'un alignement optimal de S_i et S_j
- A^*: Alignement multiple optimal de $\boldsymbol{S}$
- $d^*(Si,Sj)$: Score induit par A^*

Si le score considéré vérifie l'inégalité triangulaire:

$$e(x,z) \leq e(x,y)+e(y,z)$$

alors

$$d_c(S_i,S_j) \leq d_c(S_i,S_c) + d_c(S_c,S_j) = D(S_i,S_c)+D(S_c,S_j)$$

Et donc:

$$d(A_c)/d(A^*) \leq 2(k-1)/k < 2$$

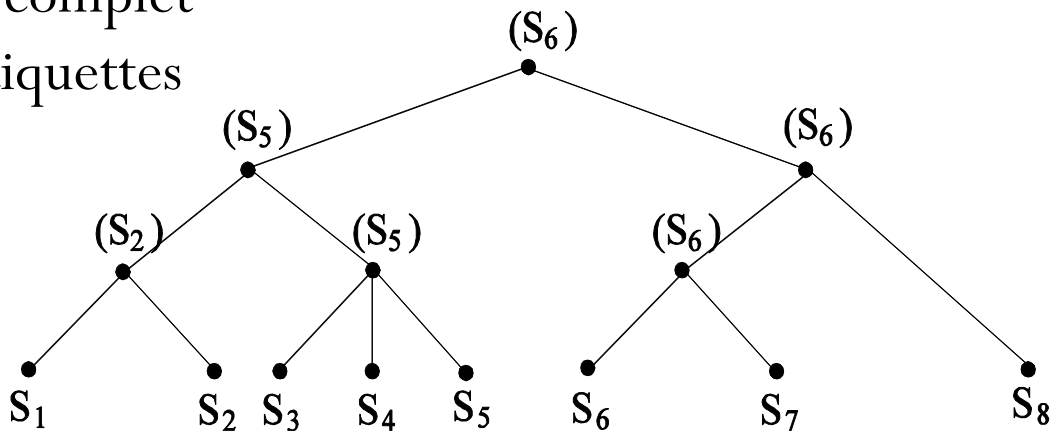
Plan

1. Introduction
2. Solution exacte pour l'alignement multiple
3. Heuristique bornée
4. Alignement phylogénétique
5. Heuristiques usuelles: Méthodes progressives
6. Heuristiques usuelles: Méthodes itératives
7. Heuristiques usuelles: Méthodes par point d'ancrage

4. Alignement phylogénétique

- **Données:** Un ensemble de séquences *S*, et un arbre phylogénétique *T* pour *S*.
- **Problème:** Trouver un étiquetage des nœuds internes de *T* qui minimise la score de *T* (somme des poids des arêtes)
- L'arbre *T* avec étiquetage de ses nœuds internes est appelé alignement phylogénétique.
- Un alignement phylogénétique *T** induit un alignement de S: c'est l'alignement consistant avec T*.
- Problème de l'étiquetage: NP-complet
- **Alignement soulevé:** Les étiquettes

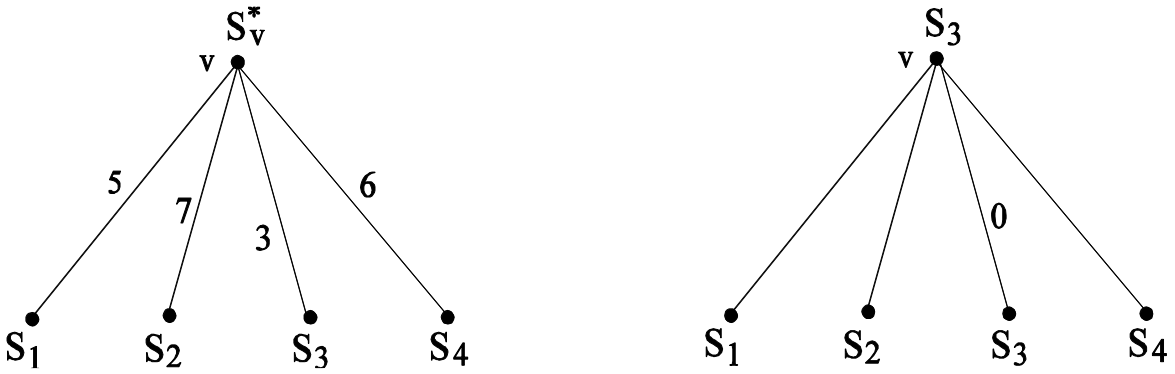
  Sont des séquences de ***S***



Alignement soulevé optimal: borne sup pour l'al. phyl. opt.

- T^*: alignement phylogénétique optimal
- On veut construire un alignement soulevé T^S à partir de T^*

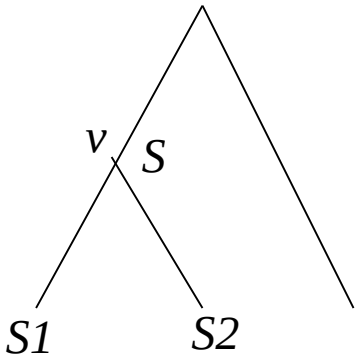
Dans T^S , v est étiqueté par la séquence de S la plus proche de S_v^*

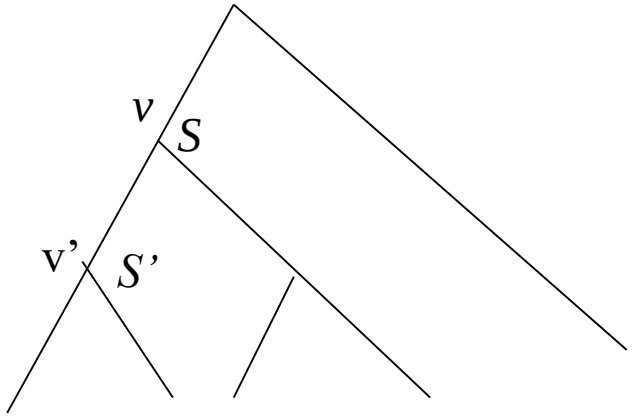


Score de $T^S \leq 2$ fois score de T^*

Alignement soulevé optimal

- T_v: sous-arbre de racine *v de* T
- *d(v,S):* Score de l'al. phyl. opt. de T_v sachant que v étiqueté par *S*





$d(v,S) = D(S,S1)+D(S,S2)$ $d(v,S) = \Sigma_{v'} min_{S'} [D(S,S') + d(v',S')]$

Valeur de l'al. Soulevé op. = minimum de *d(r,S)* où r racine de l'arbre

Complexité: *k* seq. de taille *n*.

Au cours d'un prétraitement, calculer tous les *D(Si,Sj)*: $O(k^2n^2)$

Pour chaque nœud v, calculer chaque d(v,S) en *O(k)* : $O(k^2n^2+k^3)$

Plan

1. Introduction
2. Solution exacte pour l'alignement multiple
3. Heuristique bornée
4. Alignement phylogénétique
5. Heuristiques usuelles: Méthodes progressives
6. Heuristiques usuelles: Méthodes itératives
7. Heuristiques usuelles: Méthodes par point d'ancrage

5. Heuristiques usuelles – Méthodes progressives

- Créer un alignement multiple de ***S*** en fusionnant deux alignements de deux sous-ensembles ***S1*** et ***S2*** de ***S***

Méthode générale:

- Calculer les alignements deux à deux;
- Construire un arbre guide des séquences (UPGMA, Neighbour-Joining);
- Incorporer les séquences une à une dans l'alignement multiple, en suivant l'ordre déterminé par l'arbre guide

5.1 Exemple

- Pour commencer, aligner les deux séquences de distance minimale
- À chaque étape, choisir la séquence dont la distance avec une des séquences déjà alignée est minimale

1: A C T G G
2: A C T T G G
3: A C T G C
4: C T T G

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

Etape 1:

1: A C T - G G
2: A C T T G G

Etape 2:

1: A C T - G G
3: A C T - G C

1: A C T - G G
2: A C T T G G
3: A C T - G C

Etape 3:

1: A C T - G G
4: - C T - T G

1: A C T - G G
2: A C T T G G
3: A C T - G C
4: - C T - T G

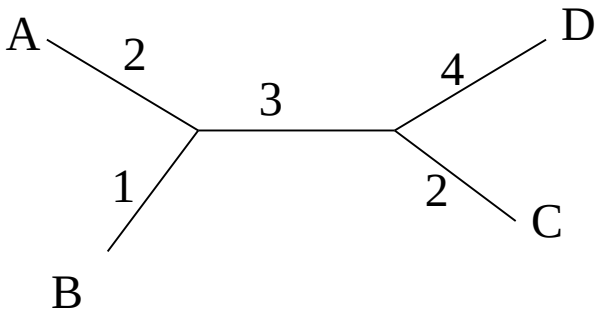
Score SP = 11

5.2 Plusieurs implémentations

- MultAlign, ClustalW, Pileup, T-Coffee, DIALIGN…
- Diffèrent surtout par la méthode de construction de l'arbre guide
- **Avantages:** Rapide, simple à programmer, nécessite peu de mémoire
- **Inconvénients:**
  - Alignement obtenu très dépendant de l'arbre guide considéré. D'où l'importance d'avoir un bon arbre de départ.
  - L'alignement ne peut pas être modifié au cours du processus
  - Produit un seul alignement

5.3 ClustalW (Thompson, Higgins, Gibson 1994)

- Algorithme progressif le plus utilisé
- Calcule les scores d'alignement de chaque paire de séquences.
- Construit un arbre guide par Neighbour-Joining
- Utilise cet arbre pour choisir les séquences à incorporer à l'alignement. Choisit les plus petites distances à chaque fois



Effectue trois sortes d'alignements: Entre deux séquences, une séquence et une matrice consensus, ou deux matrices consensus

Alignement d'une séquence avec une matrice consensus

C1	C2	C3	C4	C5
a	c	g	-	t
a	c	a	c	t
a	g	g	c	-
g	c	-	c	g

	C1	C2	C3	C4	C5
a	0.75	0	0.25	0	0
c	0	0.75	0	0.75	0
g	0.25	0.25	0.50	0	0.25
t	0	0	0	0	0.50
-	0	0	0.25	0.25	0.25

a a c - c g

C1 - C2 C3 C4 C5

Valeur d'un tel alignement?

- Matrice de pondération

	a	c	g	t	-
a	2	-3	-1	-3	-1
c	-3	2	-3	-1	-1
g	-1	-3	2	-3	-1
t	-3	-1	-3	2	-1
-	-1	-1	-1	-1	0

- Matrice consensus

	C1	C2	C3	C4	C5
a	0.75	0	0.25	0	0
c	0	0.75	0	0.75	0
g	0.25	0.25	0.50	0	0.25
t	0	0	0	0	0.50
-	0	0	0.25	0.25	0.25

❑ Alignement :

S: a a c – c g

C1 - C2 C3 C4 C5

p(a,C1) = 2 * 0.75 – 1 * 0.25 = 1.25

p(a,-) = -1*1 = -1 ; p(c,C2)= 2*0.75 -3*0.25 =0.75

p(-,C3) = -1 * 0.25 -1 * 0.50 + 0 * 0.25 = -0.75 …

=> Score alignement = Σ_i p(Ci,ti) =1.25 1+0.75+…= -1

Calcul d'un alignement optimal

$D(i,j)$: Score alignement optimal entre $S[1..i]$ et $C[1..j]$

- $D(i,0) = \Sigma_{k \leq i}\, p(t_k, -)$; $D(0,j) = \Sigma_{k \leq j}\, p(-, C_k)$

- $D(i,j) = max\,[D(i-1,j-1)+p(S_i,C_j),\ D(i-1,j)+p(S_i,-),$
  $D(i,j-1)+p(-,C_j)]$

Complexité: $O(|\Sigma|\ mn)$

(n: nbre de colonnes de C; m: taille de S)

Plan

1. Introduction
2. Solution exacte pour l'alignement multiple
3. Heuristique bornée
4. Alignement phylogénétique
5. Heuristiques usuelles: Méthodes progressives
6. Heuristiques usuelles: Méthodes itératives
7. Heuristiques usuelles: Méthodes par point d'ancrage

6. Heuristiques usuelles: Méthodes itératives

- Un des problèmes des méthodes progressives: alignements intermédiaires « figés »

X: GAAGTT

Y: GAC - TT 1er alignement intermédiaire

Z: GAACTG

W: GTACTG Y aurait dû être: G - ACTT

Méthode itérative

- Obtenir un premier alignement multiple de basse qualité
- Améliorer l'alignement par une suite d'itérations bien définies, jusqu'à ce que l'alignement ne puisse plus être amélioré.
- Méthodes déterministes ou stochastiques (alignement modifié au hasard)
- MultAlign, IterAlign, Praline, SAGA, HMMER…

Algorithme de Barton-Stenberg (MultAlign)

- Calculer tous les alignements deux à deux
- Choisir l'alignement de score max, une première matrice consensus
- À chaque étape,
  - choisir une paire de séquences de score max, tq exactement une des séquences est dans l'alignement partiel obtenu.
  - Aligner la nouvelle séquence avec la matrice consensus courante.
  - Mettre à jour la matrice consensus
  - Recommencer jusqu'à épuisement des séquences
- Retirer S1 et la réaligner avec la matrice consensus de l'al. restant (S2…. Sn). Recommencer avec S2,…,Sn
- Répéter le processus un nbre fixé de fois, ou jusqu'à ce que le score de l'alignement converge.

Plan

1. Introduction
2. Solution exacte pour l'alignement multiple
3. Heuristique bornée
4. Alignement phylogénétique
5. Heuristiques usuelles: Méthodes progressives
6. Heuristiques usuelles: Méthodes itératives
7. Heuristiques usuelles: Méthodes par point d'ancrage

7. Méthode d'alignement par points d'ancrage

Basée sur la recherche de motifs (points d'ancrage, séquences consensus…).

Par exemple, MACAW:

- Rechercher un motif suffisamment long commun à une majorité de séquences
- Problème subdivisé en deux: partie gauche et partie droite par rapport au motif
- Recommencer récursivement avec chaque partie
- Les séquences ne contenant pas le motif sont alignées séparément, par score SP. Les deux sous-alignements sont ensuite fusionnés
- Lorsque les sous-séquences ne contiennent plus de bons motifs, elles sont alignées par score SP

Références

- *Algorithms on Strings, Trees and Sequences – Computer science and Computational biology,* Dan Gusfield, Cambridge University Press, 1997. Chapitre 14.
- *Biological sequence analysis, Probabilistic models of proteins and nucleic acids*, R. Durbin, S. Eddy, A. Krogh and G. Mitchison, Cambridge 1998. Chapitre 6.
- *Handbook of Computational Molecular Biology,* Srinivas Aluru ed., Chapman & Hall/CRC Computer and Information Science Series, 2005. Chapitre 3.