Automates (max,+) et Alignements de sequences biologiques

Jean-Paul COMET

INRIA Domaine du Voluceau
Rocquencourt BP 105
78153 Le Chesnay cedex

Jean-Paul.Comet@inria.fr

PROBLEMATIQUE

Séquence biologique : ACGTCGTTC

- ADN/ARN : alphabet de taille 4
- protéines : alphabet de taille 20

Comparaison de séquences : Pourquoi ?

- Liens biologiques
- Classification

Comparaison de séquences - 3 méthodes :

- Programmation Dynamique : méthode rigoureuse mais coûteuse *[Needleman & Wunsch, 1970] [Smith & Waterman, 1981]*
- FASTA *[Lipman & Pearson, 1985]*: heuristique avec insertion/délétion
- BLAST *[Altschul et al., 1990]*: sans insertion/délétion

Programmation Dynamique :

▷ opération max et $+$

▷ Utilisation de l'algèbre $(max, +)$

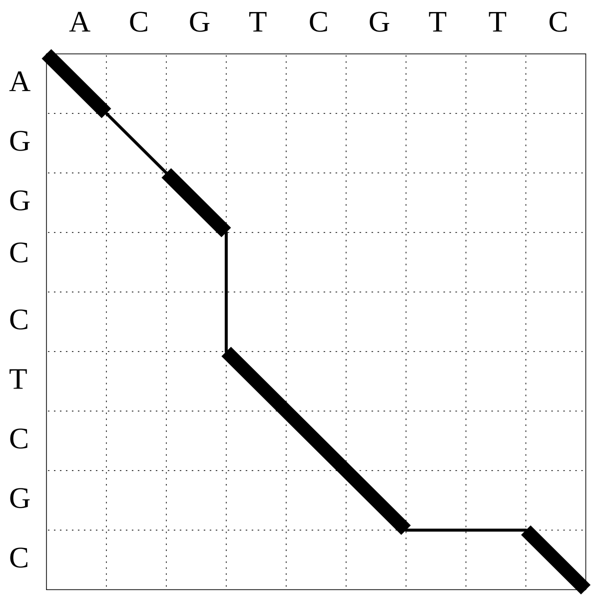
```
hemoglobin alpha chain - Arabian camel        (PIR1 : HACMA)
hemoglobin beta  chain - moustached tamarin   (PIR1 : HBMKM)


 1 V-LSSKDKTNVKTAFGKIGGHAAEYGAEALERMFLGFPTTKTYFPHF-DL
   |-|    |  | |  || --   | | ||| |     | |   |  |-||
 1 VHLTGEEKSAVTTLWGKV--NVEEVGGEALGRLLVVYPWTQRFFDSFGDL

50     .    :    .    :    .    :    .    :    .    :
49 SH-----GSAQVKAHGKKVGDALTKAADHLDDLPSALSALSDLHAHKLRV
   | -----    ||||||||  |      ||| |      || ||  || |
49 SSPDAVMNNPKVKAHGKKVLGAFSDGLAHLDNLKGTFAQLSELHCDKLHV

100    .    :    .    :    .    :    .    :    .
94 DPVNFKLLSHCLLVTVAAHHPGDFTPSVHASLDKFLANVSTVLTSKYR
   || || ||   |    | |    ||| | |   |  | |   |  ||
99 DPENFRLLGNVLVCVLAHHFGKEFTPQVQAAYQKVVAGVANALAHKYH
```

Un exemple d'alignement

$$\begin{array}{rcl} A & = & ACGTCGTTC \\ B & = & AGGCCTCGC. \end{array}$$

Un alignement entre ces deux séquences est :

$$\begin{array}{l} A\,C\,G - - T\,C\,G\,T\,T\,C \\ A\,G\,G\,C\,C\,T\,C\,G - - C \end{array}$$



On associe à chaque alignement un **score**:

Exemple : $\sigma(a,a) = 1,$

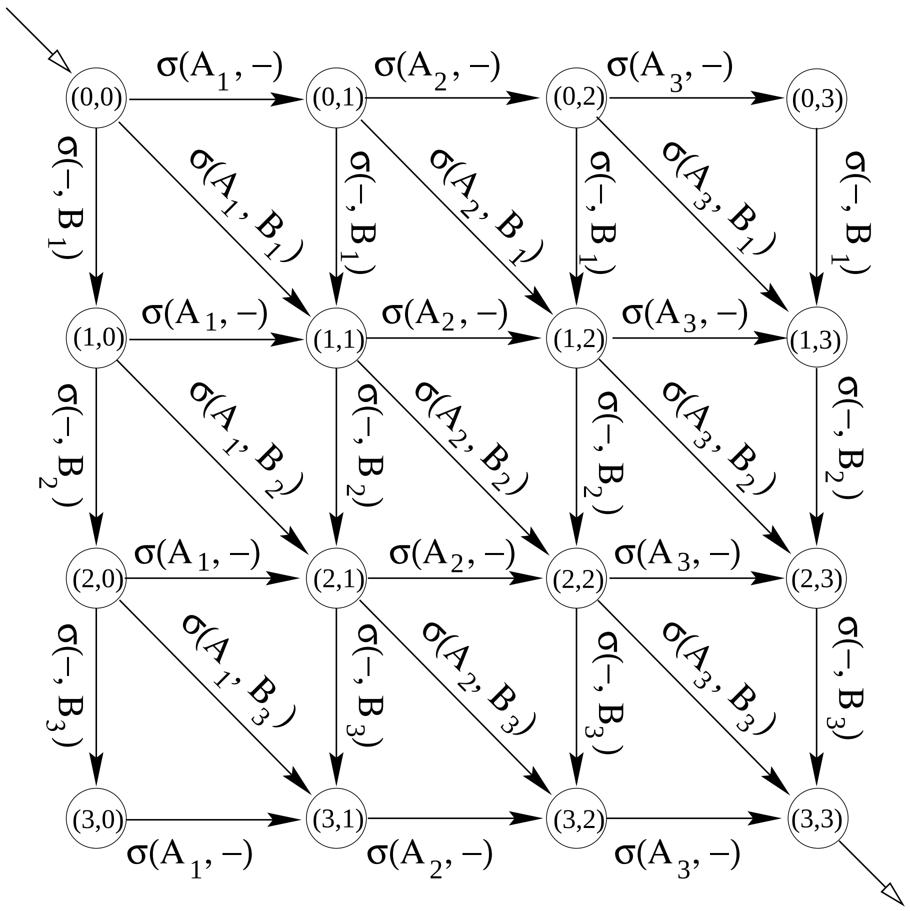
$\sigma(a,b) = 0,$

$\sigma(-,a) = \sigma(a,-) = -\delta = -1$

Alors $S(A,B) = 6 - 4 = 2$

Formulation à l'aide d'un graphe orienté valué :

1 alignement $\simeq$ 1 chemin dans un graphe



- *[Needleman & Wunsch, 1970] :*
  $NW_{i,j}$: score d'alignement optimal des séquences $A[1,i]$ et $B[1,j]$.

$$NW(i,j) \quad = \quad \max \begin{cases} NW(i-1,j) & + \quad \sigma(A_i,-) \quad \text{délétion} \\ NW(i-1,j-1) & + \quad \sigma(A_i,B_j) \quad \text{substitution} \\ NW(i,j-1) & + \quad \sigma(-,B_j) \quad \text{insertion} \end{cases}$$

Complexité : $O(nm)$

La programmation dynamique : SW

[Smith & Waterman, 1981]

- SW : On cherche 2 sous-séquences $A[i_1, i_2]$ et $B[j_1, j_2]$ telles que

$$NW(A[i_1,i_2],B[j_1,j_2]) = \max_{\substack{i,j,k,l \\ i<j \; k<l}} NW(A[i,j],B[k,l])$$

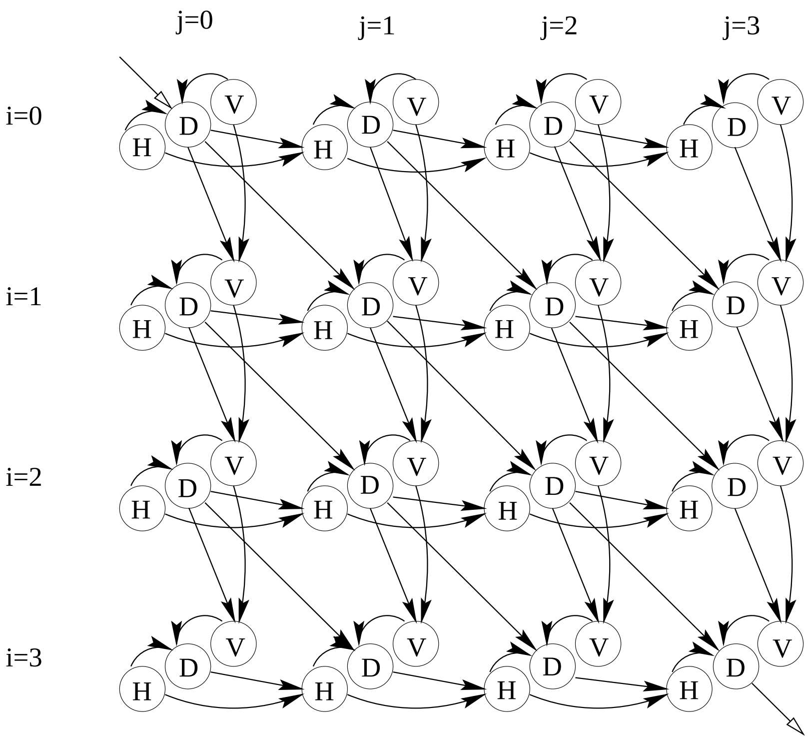
$$SW(i,j) \quad = \quad \max \begin{pmatrix} SW(i-1,j-1) & + & \sigma(A_i,B_j), \\ SW(i-1,j) & + & \sigma(A_i,-), \\ SW(i,j-1) & + & \sigma(-,B_j), \\ & 0 & \end{pmatrix}$$

$$\begin{aligned} SW(0,0) &= 0 \\ SW(i,0) &= 0 \quad 1 \le i \le n \\ SW(0,j) &= 0 \quad 1 \le j \le m \end{aligned}$$

- Alignement : phase de remontée dans le graphe.

Formulation à l'aide d'un graphe orienté valué :

$$\sigma(\left[\begin{smallmatrix} a_{i+1} & a_{i+2} & \dots & a_{i+k} \\ - & - & & - \end{smallmatrix}\right]) = \alpha + (k-1).\beta$$



- *[Needleman & Wunsch, 1970] :*

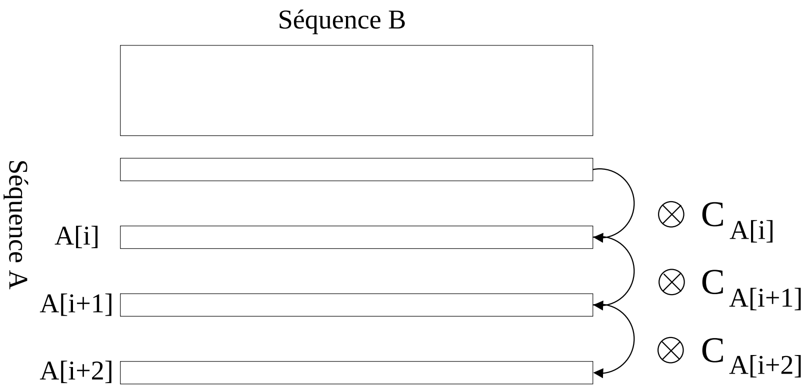
$$\begin{aligned}
NW(i,j) &= \max \begin{pmatrix} NW(i-1,j-1) + s(a_i,b_j), \\ I(i,j), \\ D(i,j) \end{pmatrix} \\
I(i,j) &= \max \begin{pmatrix} NW(i,j-1) - \alpha, \\ I(i,j-1) - \beta \end{pmatrix} \\
D(i,j) &= \max \begin{pmatrix} NW(i-1,j) - \alpha, \\ D(i-1,j) - \beta \end{pmatrix}
\end{aligned}$$

L'algèbre (max, +) au secours de la Programmation Dynamique

- Opérations utilisées : max, +
- notations :

max	$\oplus$
+	$\otimes$

▷ Pour une séquence *requête* donnée :



▷ on passe d'une ligne à une autre par application linéaire de l'algèbre $(max, +)$

▷ L'ensemble des vecteurs lignes est fini (projectivement fini)

▷ Il existe un automate dont le parcours par une séquence est équivalent au calcul du score.

▷ Complexité linéaire par rapport à la longueur de la séquence *cible*

Un cas simple : NW (cas linéaire)

- 2 séquences A et B de longueurs l_A et l_B.

$$\begin{array}{rcl} NW(i,j) & = & \max\begin{pmatrix} NW(i-1,j-1) & + & \sigma_{i,j}, \\ NW(i-1,j) & - & \delta, \\ NW(i,j-1) & - & \delta \end{pmatrix} \\ & = & \left\{ \begin{array}{cccl} & \sigma_{i,j} & \otimes & NW(i-1,j-1) \\ \oplus & \delta^{-1} & \otimes & NW(i-1,j) \\ \oplus & \delta^{-1} & \otimes & NW(i,j-1) \end{array} \right. \end{array}$$

- X_n le vecteur correspondant à la n-ème ligne du tableau

$$X_n = \begin{pmatrix} NW(n,0) & NW(n,1) & NW(n,2) & ... & NW(n,l_B) \end{pmatrix}$$

- $X_n = X_{n-1} \otimes A_n \quad \oplus \quad X_n \otimes B$

$$A_n = \begin{pmatrix} \delta^{-1} & \sigma_{1,n} & . & \cdots & . \\ . & \delta^{-1} & \sigma_{2,n} & \cdots & . \\ . & . & \delta^{-1} & \ddots & . \\ \vdots & \vdots & \vdots & \ddots & \sigma_{l_B,n} \\ . & . & . & . & \delta^{-1} \end{pmatrix} \qquad B = \begin{pmatrix} . & \delta^{-1} & . & . & . \\ . & . & \delta^{-1} & . & . \\ . & . & . & \ddots & . \\ . & . & . & . & \delta^{-1} \\ . & . & . & . & . \end{pmatrix}$$

Puisque la série $\sum_{i=0}^{i=k} B^i$ converge, alors on pose

$$B^* = \sum_{i=0}^{i=\infty} B^i = \begin{pmatrix} 0 & \delta^{-1} & \delta^{-2} & \delta^{-3} & \cdots & \delta^{-m} \\ . & 0 & \delta^{-1} & \delta^{-2} & \cdots & \delta^{-m+1} \\ . & . & 0 & \delta^{-1} & \cdots & \delta^{-m+2} \\ . & . & . & \ddots & \ddots & \vdots \\ . & . & . & . & 0 & \delta^{-1} \\ . & . & . & . & . & 0 \end{pmatrix}$$

- On a :

$$
\begin{aligned}
C_n &= A_n \otimes B^* \\
X_n &= X_{n-1} \otimes C_n
\end{aligned}
$$

(C_n triangulaire supérieure)

avec

$$
C_n = \begin{pmatrix}
\delta^{-1} & & & & \\
. & \delta^{-1} & & C_{i,j} & \\
. & . & \delta^{-1} & & \\
. & . & . & & \\
. & . & . & & \delta^{-1}
\end{pmatrix}
$$

$$
\begin{cases}
si\ i < j & C_{i,j} = \begin{cases} (\delta^{-2} \oplus \sigma(i,n)) \\ \otimes \delta^{-(|i-j|-1)} \end{cases} = \begin{cases} \max(-2\delta, \sigma(i,n)) \\ -(|i-j|-1).\delta \end{cases} \\
si\ i > j & C_{i,j} = \varepsilon \quad = \quad -\infty \\
si\ i = j & C_{i,j} = \delta^{-1} \quad = \quad -\delta
\end{cases}
$$

si $\delta^{-2} \leq \min_{a,b}(\sigma(a,b))$:

$$
\begin{cases}
si\ i < j & C_{i,j} = (\sigma(i,n)) \otimes \delta^{-(|i-j|-1)} = \sigma(i,n) - (|i-j|-1).\delta \\
si\ i > j & C_{i,j} = \varepsilon \quad = \quad -\infty \\
si\ i = j & C_{i,j} = \delta^{-1} \quad = \quad -\delta
\end{cases}
$$

Quelques définitions avant de continuer

Définition 1 *L'espace projectif de $\mathbb{R}^n_{max}$ noté $\mathbb{P}\mathbb{R}^n_{max}$, de dimension $n-1$, est défini comme étant le quotient de $\mathbb{R}^n_{max}$ par la relation d'équivalence $\sim$:*

$$\text{pour } u \in \mathbb{R}^n_{\max} \text{ et } v \in \mathbb{R}^n_{\max}$$
$$u \sim v \Longleftrightarrow \exists \lambda \in \mathbb{R}_{max} \backslash \{\varepsilon\}, \quad u = \lambda v$$

Définition 2 *On définit de même l'espace projectif $\mathbb{P}\mathbb{R}^{n\times n}_{max}$ comme étant le quotient de l'espace des matrices carrées $(max, +)$ de dimension n par la relation d'équivalence :*

$$\text{pour } U \in \mathbb{R}^{n\times n}_{\max} \text{ et } V \in \mathbb{R}^{n\times n}_{\max}$$
$$U \sim V \Longleftrightarrow \exists \lambda \in \mathbb{R}_{max} \backslash \{\varepsilon\}, \quad U = \lambda V$$

Définition 3 *On dira qu'un ensemble $S \in \mathbb{R}^{n\times n}_{max}$ est* **projectivement fini** *si l'ensemble quotient $S/\sim$ est fini, c'est-à-dire s'il n'existe qu'un nombre fini d'éléments deux à deux non proportionnels.*

Résultats pour NW

Théorème 1

1. *Le semi-groupe* $\langle C_a, a \in \Sigma \rangle$ *est* projectivement fini.
2. *Le semi-groupe* $\langle \tilde{C}_a = C_a \otimes \delta, a \in \Sigma \rangle$ *est* fini.

Théorème 2

Pour une séquence B fixée, la variable

$$NW(A, B) + \delta \times l_A$$

ne prend qu'un nombre fini de valeurs quand A parcourt Σ^*.

Remarque 1 *Résultats indépendants du caractère entier ou réel des paramètres* $\sigma_{i,j}$ *et* δ.

Le semi-groupe des matrices C_n

Théorème 1 *Le semi-groupe* $\langle \tilde{C}_a = C_a \otimes \delta, a \in \Sigma \rangle$ *est* fini.

Démonstration:

- Définissons un automate.
  - Etats de l'automate : $\mathcal{S} = \{0, 1, 2, \ldots, l_B\}$.
  - Transitions : $(i \xrightarrow{a} j)$ où a désigne une lettre telle que $(\tilde{C}_a)_{i,j} \neq \varepsilon$.
  - Poids de $(i \xrightarrow{a} j)$: $(\tilde{C}_a)_{i,j}$.
- les arcs $(i \longrightarrow j)$ sont tels que $i \leq j$.
- Les seuls circuits : boucles simples $(k \longrightarrow k)$ de poids nul.
- Soit $M \in \langle \tilde{C}_a, a \in \Sigma \rangle$: $M = \bigotimes_{i=1}^{l} (\tilde{C}_{a_i})$.
- $M(ii) = e \quad \forall i \qquad (\tilde{C}_a(i,i) = e \quad \forall a \in \Sigma, \forall i)$.
- $M_{i,j}$: le maximum des coûts des chemins de label $a_1 a_2 a_3 \ldots a_l$.

Soit un chemin quelconque : $i = i_1, i_2, i_3 \ldots i_k = j$.
La suite $(i_l)_{l=1..k}$ est croissante (matrices triang. sup.).
Il n'y a qu'un nombre fini de chemins i ... j (hors boucles).
$\longrightarrow$ Le poids pour un chemin quelconque i ... j ne peut prendre qu'un nombre fini de valeurs.
$\longrightarrow$ $M_{i,j}$ ne peut prendre qu'un nombre fini de valeurs.

□

Résultat pour le score NW

Théorème 2 *Pour une séquence B fixée, la variable*

$$NW(A,B) + \delta \times l_A$$

ne prend qu'un nombre fini de valeurs quand A parcourt Σ^*.

Démonstration:

- $\langle \tilde{C}_a, a \in \Sigma \rangle$ est fini (démonstration précédente).
- Le produit consécutif $\left(\bigotimes_{i=1}^{l_A} \tilde{C}_{A[i]} \right)$ s'écrit :

$$\begin{aligned} \left(\bigotimes_{i=1}^{l_A} \tilde{C}_{A[i]} \right) &= \bigotimes_{i=1}^{l_A} \delta C_{A[i]} \\ &= \delta^{l_A} \otimes \left(\bigotimes_{i=1}^{l_A} C_{A[i]} \right) \end{aligned}$$

- Si on ne regarde que l'élément $(l_B + 1)$ du vecteur $X_0 \otimes \left(\bigotimes_{i=1}^{n} \tilde{C}_{A[i]} \right)$, on a

$$\left(X_0 \ \bigotimes_{i=1}^{l_A} \tilde{C}_{A[i]} \right)_{l_B+1} = \delta^{l_A} \otimes \underbrace{\left(X_0 \ \bigotimes_{i=1}^{l_A} C_{A[i]} \right)_{l_B+1}}_{NW(A,B)}$$

$\square$

Automate de Cayley

(B fixée)

- But : construire un automate reconnaissant Σ^* :

$$A \in \Sigma^* \longrightarrow \left(\bigotimes_{i=1}^{l_A} \tilde{C}_{A[i]}\right)$$

- Definissons un morphisme noté ϕ :

$$\begin{array}{ccc} \Sigma^* & \longrightarrow & \langle \tilde{C}_a, a \in \Sigma \rangle \\ A & \longrightarrow & \phi(A) = \left(\bigotimes_{i=1}^{l_A} \tilde{C}_{A[i]}\right) \end{array}$$

- Automate de Cayley :
  - Etat initial (noté e) : correspond à la matrice identité,
  - tous les états sont terminaux,
  - il existe une transition $(\tilde{C}_1 \xrightarrow{a} \tilde{C}_2)$ si $\tilde{C}_1 \otimes \tilde{C}_a = \tilde{C}_2$.

Calcul du score Needleman & Wunsch $NW(A,B)$:

- Le score $NW(A,B)$ entre deux séquences A et B (fixée) est

$$NW(A,B) = \delta^{+l_A} \otimes X_0 \otimes \phi(A) \otimes \begin{pmatrix} \varepsilon \\ \vdots \\ \varepsilon \\ e \end{pmatrix}.$$

- $\phi(A)$ se lit directement sur l'automate de Cayley en **temps linéaire**.

Automate réduit

- Relation d'équivalence $\sim$:
  $u \sim v \Longleftrightarrow \exists \lambda \in \mathbb{R}_{max} \backslash \{\varepsilon\}, \quad u = \lambda \otimes v$

- Automate réduit :
  - Etat i : une classe d'équivalence de représentant u_i.
  - Etat initial : classe d'éq. de $X_0 = (0, \delta^{-1}, \delta^{-2}, \ldots, \delta^{-l_B})$.
  - Tous les états sont terminaux.
  - Il existe une transition $(i \xrightarrow{a} j)$ ssi il existe $\lambda \in \mathbb{R}_{max}$ tel que $u_j = \lambda \otimes (u_i \otimes C_a)$,
  - A la transition $(i \xrightarrow{a} j)$, on associe le poids λ, coefficient de proportionnalité : $u_j = \lambda \otimes (u_i \otimes C_a)$.

- Calcul de $X_0 \phi(A)$ pour toute séquence $A \in \Sigma^*$.
  - L'automate donne (en temps linéaire) directement la classe d'équivalence de $X_0 \phi(A) = X_{l_A}$.
  - Calcul du coefficient de proportionnalité λ entre le vecteur X_{l_A} et le représentant de sa classe d'équivalence :

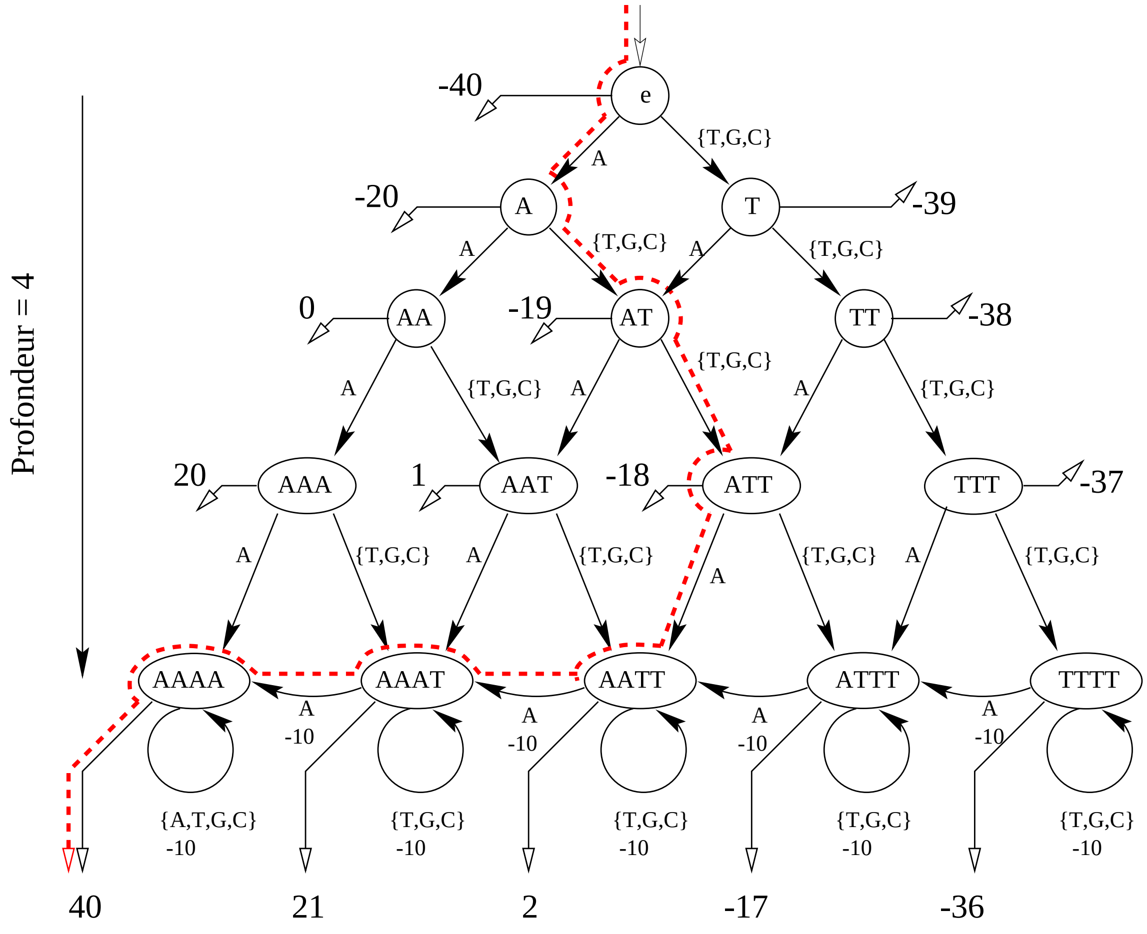
$$\lambda = \bigotimes_{k=1}^{l_A} \lambda_k$$

    où λ_k : coefficient associé à la $k^{ième}$ transition du chemin.
  - $NW(A, B) = \lambda \otimes X_0 \otimes \phi(A) \otimes \begin{pmatrix} \varepsilon \\ \vdots \\ \varepsilon \\ e \end{pmatrix}.$

Un Automate associé à NW

séquence requête	:	“**AAAA**”
séquence cible	:	“**ATGAAA**”

$$\begin{array}{lcll} \sigma(a,a) & = & 10 & \forall\ a \ \in \{A,T,G,C\} \\ \sigma(a,b) & = & -9 & \forall\ a, b \neq a \in \{A,T,G,C\} \\ gap & = & -10 & \end{array}$$




```
A - - A A A
|     | | |
A T G A A A
```

Automate réduit associé à l'algorithme de Needleman & Wunsch

Algorithme de construction de l'automate réduit

1. Construction des matrices $C_a, a \in \Sigma$.
2. L'état initial e est associé au vecteur initial $X_0 = (0, \delta^{-1}, \delta^{-2}, \ldots, \delta^{-l_B})$. Le représentant choisi est X_0. Initialisation des deux listes :

$$L = \{e\}$$
$$E = \{e\}$$

3. Pour chaque état i (dont le vecteur représentant est u_i) de E
   - Pour chaque lettre a de l'alphabet, on explore les fils de l'état i :
     - Calcul de $u = u_i \otimes C_a$.
     - S'il existe $\lambda \in \mathbb{R}_{max}$ et $j \in L$ tel que $u_j = \lambda \otimes u$, alors on crée une transition de i vers j, de label a et de coefficient λ.
     - Sinon on crée un nouvel état k, que l'on rajoute dans E. On choisit comme vecteur représentant de ce nouvel état k, le vecteur $u_k = u$.
   - On enlève i de la liste E.
4. Terminaison lorsque la liste E est vide.

Cet algorithme se termine puisque l'automate réduit est fini.

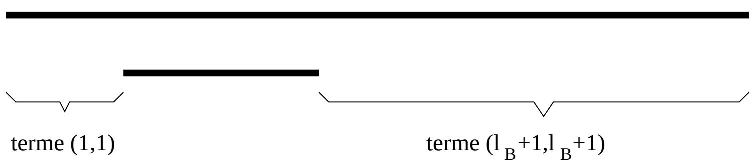
Automates et algorithme NW

mots	taille du semi-groupe des matrices normalisées	taille de l'orbite du vecteur initial	profondeurs
AAAA	14	15	4
ATTA	83	43	5
ATGC	400	84	6
ATCGA	1570	199	8
ATCGAT	5328	439	9
ATCGATG	15271	919	10
ATCGATGC	39047	1873	12

Table 1: **Taille des automates** pour l'algorithme de NW. La taille de l'orbite représente la taille de l'automate réduit.

Recherche de la meilleure occurrence d'un mot au sens NW

- Définition :



- Equation de récurrence : $X_n = X_{n-1} D_n$
- Une grande différence :

$$D_n = \left(\begin{array}{c|ccccc|c} 0 & & & D_{1,j} & & & D_{1,l_B+1} \\ \hline \varepsilon & \delta^{-1} & & & & & \\ \varepsilon & \varepsilon & \delta^{-1} & & D_{i,j} & & D_{i,l_B+1} \\ . & . & . & \delta^{-1} & & & \\ . & . & & \varepsilon & \delta^{-1} & & \\ \hline \varepsilon & \varepsilon & \dots & \varepsilon & \varepsilon & & 0 \end{array} \right)$$

- **Résultats**

Théorème 3 *L'orbite du vecteur initial* $X_0 = (0, \delta^{-1}, \delta^{-2}, \dots, \delta^{-l_B})$ *dans le semi-groupe* $\langle D_a, a \in \Sigma \rangle$ *est finie.*

Corollaire 1 *Soit* $B \in \Sigma^*$. *Le score de la meilleure occurrence du mot* B *au sens NW dans une séquence quelconque, ne prend qu'un nombre fini de valeurs.*

Occurrence d'un mot au sens NW

Théorème 3 ' *Soit* $X_0 \in (\mathbb{R}_{max} \setminus \{-\infty\})^{l_B+1}$ *à coordonnées toutes finies :* $X_0(i) \neq \varepsilon\ \forall i \in [1, l_B + 1]$*. L'ensemble* $\{X_0 D_{a_1} D_{a_2} \cdots D_{a_k}, a_1 a_2 \cdots a_k \in \Sigma^*\}$ *est fini.*

Démonstration:

1. Automate associé aux graphes des matrices de précédence.
   - Les états : $\mathcal{S} = \{0, 1, 2, \ldots, l_B\}$.
   - Les transitions : $(i \xrightarrow{a} j)$ si $(D_a)_{i,j} \neq \varepsilon$.
   - Poids de la transition $(i \xrightarrow{a} j)$: $(D_a)_{i,j}$.
2. L'ensemble des coefficients $(1, j)$ d'un produit de matrices $\{(D_{a_1} D_{a_2} \cdots D_{a_k})_{1j}, a_1 a_2 \cdots a_k \in \Sigma^*\}$ est fini.
   - $(D_{a_1} D_{a_2} \cdots D_{a_k})_{1j}$: poids maximum pour un chemin de label $a_1 a_2 \cdots a_k$ allant de 1 à j.
   - Soit $V_j = \{$poids des chemins élémentaires de 1 à $j\}$.
     V_j est fini (nb fini de chemins).
     M_j : max et m_j : min
   - Soit $V'_j = \{$ poids d'un chemin quelconque de 1 à $j\}$.
     Boucle $(1 \longrightarrow 1)$: poids nul.
     Boucle $(j \longrightarrow j)$: δ^{-1}.
     Le poids d'un chemin avec boucle appartient à :
     $$V'_j \subset V_j \cup \delta^{-1} \otimes V_j \cup \delta^{-2} \otimes V_j \cup \cdots$$
     V'_j est majoré par M_j.
   - Soit un chemin (de poids c) de 1 à j avec k boucles non nulles.
     $k_0^j = \lfloor \frac{M_j - m_j}{\delta} \rfloor + 1$. Si $k > k_0^j$ boucles non nulles, $c < m_j$:
     En effet, $c = c_e - k \times \delta$, avec $c_e \in V_j$.
     $c < M_j - (M_j - m_j) = m_j$.
     Il existe un chemin de meilleur poids : toutes les boucles sont au début.
   - On se restreint aux chemins ayant moins de k_0^j boucles non nulles.

Il y a un nombre fini de chemins ayant au plus k_0^j boucles de poids δ^{-1} (hors chemins avec boucles nulles).
Donc l'ensemble des poids des chemins optimaux allant de 1 à j, pour un mot quelconque $a_1 a_2 \cdots a_k$ est inclue dans

$$W_j \subset V_j'' = V_j \cup \delta^{-1} \otimes V_j \cup \delta^{-2} \otimes V_j \cup \cdots \cup \delta^{-k_0^j} \otimes V_j$$

V_j'' est fini, donc W_j' l'est aussi.

3. Montrons maintenant que $\{(X_0 D_{a_1} D_{a_2} \cdots D_{a_k})_l, a_1 a_2 \cdots a_k \in \Sigma^*\}$ est fini. Posons $w = a_1 a_2 \cdots a_k$.
   - On cherche à calculer $X_0 \otimes \phi(w)$ où l'application ϕ est l'unique morphisme tel que $\forall a \in \Sigma, \phi(a) = D_a$.

$$(X_0 \otimes \phi(w))_l = \bigoplus_{i=1}^{l_B+1} X_0(i) \otimes (\phi(w))_{il}$$

   - $U = \{(X_0 \otimes \phi(w))_l, w \in \Sigma^*\}$ est majoré.
     Un majorant de $(X_0 \otimes \phi(w))_l$ est $\max_{i=1}^{l_B+1}(X_0(i)) + K$.
   - $U = \{(X_0 \otimes \phi(w))_l, w \in \Sigma^*\}$ est minoré.

$$(X_0 \otimes \phi(w))_l \geq X_0(1) \otimes (\phi(w))_{1l} \geq X_0(1) \otimes m_l$$

   - $U = \{(X_0 \otimes \phi(w))_l, w \in \Sigma^*\}$ est fini.
     $(X_0 \otimes \phi(w))_l$ se décompose en un poids d'un chemin élémentaire, un certain nombre de boucles et une pénalisation.

$$\begin{aligned} (X_0 \otimes \phi(w))_l &= X_0(i) \otimes \phi(w)_{il} \\ &= X_0(i) \otimes \delta^{-k} \otimes c_e \end{aligned}$$

     – U est borné,
     – k est majoré.
     – c_e et $X_0(i) \in$ ensembles finis,
     donc U est fini.

4. $\{X_0 D_{a_1} D_{a_2} \cdots D_{a_k}, a_1 a_2 \cdots a_k \in \Sigma^*\}$ est aussi fini.□

Corollaire 0 *L'automate réduit est fini.*

Corollaire 1 *Soit $B \in \Sigma^*$. Le score de la meilleure occurrence du mot B au sens NW dans une séquence quelconque, ne prend qu'un nombre fini de valeurs.*

Démonstration: Dans l'automate réduit, toutes les transitions ont un poids nul. Donc le coefficient de proportionnalité $\left(\otimes_{k=1}^{l_A} \lambda_k\right)$ est nul. Il n'y a donc pas plus de scores que d'états.
$\square$

Occurrence d'un mot au sens NW (1)

- Automate réduit sur les matrices initiales

mots	taille de l'orbite du vecteur initial	profondeur
AAAA	164	14
ATTA	305	12
ATGC	365	10
ATCGA	1680	13
ATCGAT	6162	16
ATCGATC	23116	18
ATCGATCG	79205	23

- Diminution de la taille de l'automate
  - Le dernier élément peut être gardé à δ^{-1}.

mots	taille de l'orbite du vecteur initial	profondeur
AAAA	87	11
ATTA	139	9
ATGC	191	8
ATCGA	599	12
ATCGAT	1840	13
ATCGATC	5489	16
ATCGATCG	15842	19

Résultats : algorithme k-erreurs

$$\sigma(a,a) = 0$$
$$\sigma(a,b) = -1,$$
$$\sigma(-,a) = \sigma(a,-) = -\delta = -1$$

Mots recherchés	long.	Programmation Dynamique		Automate			
		temps (en s.)	vitesse[a]	temps (en s.) total	(1)[b]	(2)[c]	taille de l'automate
BAAABF	6	15.20	8.37	9.795	0.005	9.79	83
KIIKLHEN	8	19.93	8.51	9.86	0.034	9.826	472
VKIIKLHEN	9	22.25	8.58	10.284	0.087	10.197	1114
AASDTGSTYL	10	24.48	8.66	10.442	0.18	10.262	2397
LVIVSVFDLAS	11	26.68	8.74	10.885	0.405	10.48	4928
KNVIGARRASWR	12	28.69	8.87	12.176	1.237	10.939	12033
RAANQDYVITRTN	13	31.11	8.86	13.935	2.716	11.219	24331
QGQQFPNECQLDQL	14	33.16	8.95	17.389	6.089	11.30	50820
QGQQFPNECQLDQLN	15	35.49	8.96	25.916	13.098	12.818	107408

Table 2: **Performances comparées de la Programmation Dynamique simple et de l'algorithme dérivant de l'écriture (max,+)**

Algorithme de Smith & Waterman

- $S_{i,j} = \max(S_{i-1,j-1} + \sigma_{i,j}, S_{i-1,j} - \delta, S_{i,j-1} - \delta, 0)$
$$= \left(S_{i-1,j-1} \otimes \sigma_{i,j}\right) \oplus \left(\delta^{-1} \otimes S_{i-1,j}\right) \oplus \left(\delta^{-1} \otimes S_{i,j-1}\right) \oplus 0$$

- Y_n : vecteur correspondant à la n-ème ligne du tableau

- $Y_n = Y_{n-1} \otimes A_n \quad \oplus \quad Y_n \otimes B \quad \oplus C$

  avec

$$C \quad = \quad (0\ ,\ 0\ ,\ \ldots\ ,\ 0)$$

$$B = \begin{pmatrix} . & \delta^{-1} & . & . & . \\ . & . & \delta^{-1} & . & . \\ . & . & . & \ddots & . \\ . & . & . & . & \delta^{-1} \\ . & . & . & . & . \end{pmatrix} \qquad A_n = \left(\begin{array}{c|cccc} 0 & \sigma_{1,n} & . & \ldots & . \\ \hline . & \delta^{-1} & \sigma_{2,n} & \ldots & . \\ . & . & \delta^{-1} & \ddots & . \\ . & . & . & \ddots & \sigma_{l_B,n} \\ . & . & . & . & \delta^{-1} \end{array}\right)$$

- En posant $D_n = A_n \otimes B^*$:

$$Y_n \quad = \quad Y_{n-1} D_n \quad \oplus \quad C \qquad (1)$$
$$Y_0 \quad = \quad (\ 0\ ,\ 0\ ,\ \ldots\ ,\ 0) \qquad (2)$$

Algorithme de SW (suite)

- Les itérations successives :

$$\begin{aligned} Y_1 &= Y_0 D_1 \oplus C \\ Y_2 &= Y_1 D_2 \oplus C = Y_0 D_1 D_2 \oplus C D_2 \oplus C \\ Y_3 &= Y_2 D_3 \oplus C = Y_0 D_1 D_2 D_3 \oplus C D_2 D_3 \oplus C D_3 \oplus C \\ Y_n &= \bigoplus_{k=1}^{n+1} C \bigotimes_{i=k}^{n} D_i \qquad (3) \end{aligned}$$

  avec la convention :

$$\bigotimes_{i=n+1}^{n} C_i = I$$

- Reformulation : "*chercher parmi tous les suffixes du mot considéré celui pour lequel le score est maximal*"
- Le score SW est donné par :

$$SW = \bigoplus_{k=0}^{l_A} |Y_k| \qquad (4)$$

  où

$$|Y_k| = \bigoplus_{i=1}^{l_B+1} Y_k(i) \qquad (5)$$

- Reformulation propre

$$(e, Y_n) = (e, Y_{n-1}) \otimes \underbrace{\left(\begin{array}{c|c} e & (e, e, e, \ldots) \\ \hline \varepsilon & \\ \varepsilon & \\ \varepsilon & D_n \\ \varepsilon & \\ \vdots & \end{array} \right)}_{\tilde{D}_n} \qquad (6)$$

- **Théorème 4** *L'orbite du vecteur initial* $Y_0 = (0, 0, \ldots, 0)$ *dans le semi-groupe* $\langle \tilde{D}_\alpha, \alpha \in \Sigma \rangle$ *est fini.*

  Démonstration: Toutes les matrices génératrices sont triangulaires. Les éléments diagonaux sont tous négatifs, sauf le premier qui est nul. Les caractéristiques nécessaires à la démonstration du théorème (cas occurrence NW), sont réunies, la même démarche donne le résultat. $\square$