

Cours 5

L'alignement multiple

Définition

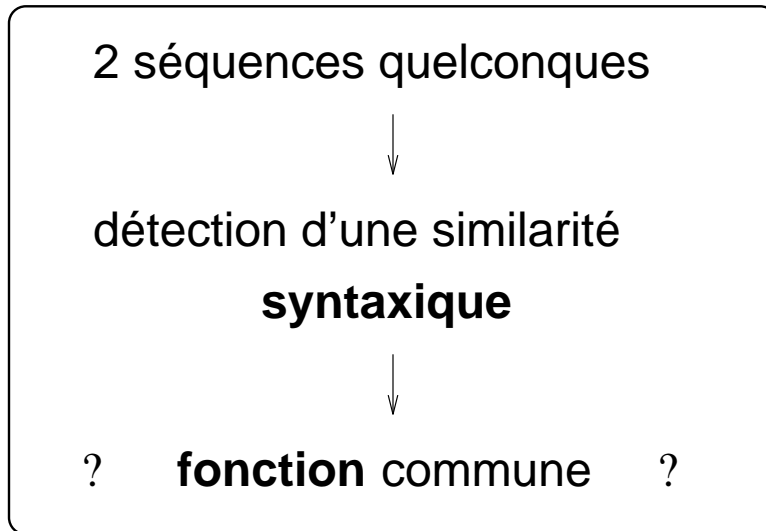
▷ **Entrée** : k séquences

```
* * * * * * * * * * * * * *
* * * * * * * * * *
* * * * * * * * * * * * * *
* * * * * * * * * *
```

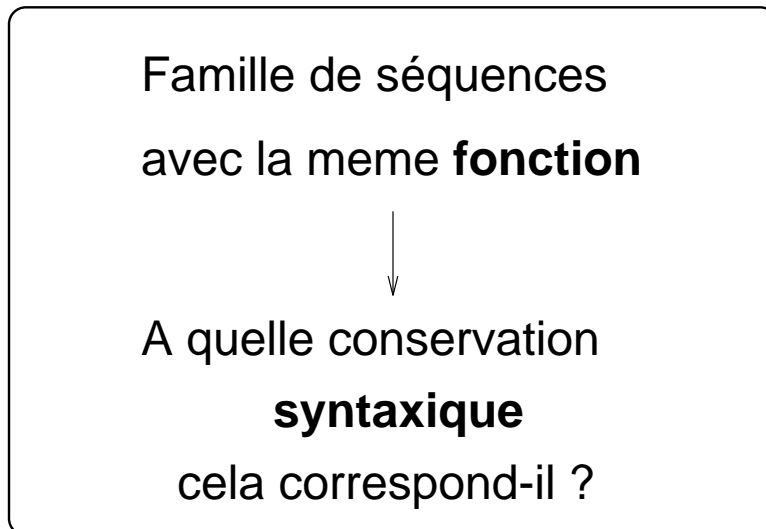
▷ **Sortie** : un tableau contenant les k séquences, avec des indels

```
* * * * * * * * * - * * * *
* * * - - - * * * - * * * *
* * * - * * * * * * * * * *
* * * - - * * - - * * * * *
```

Alignement 2 à 2



Alignement multiple

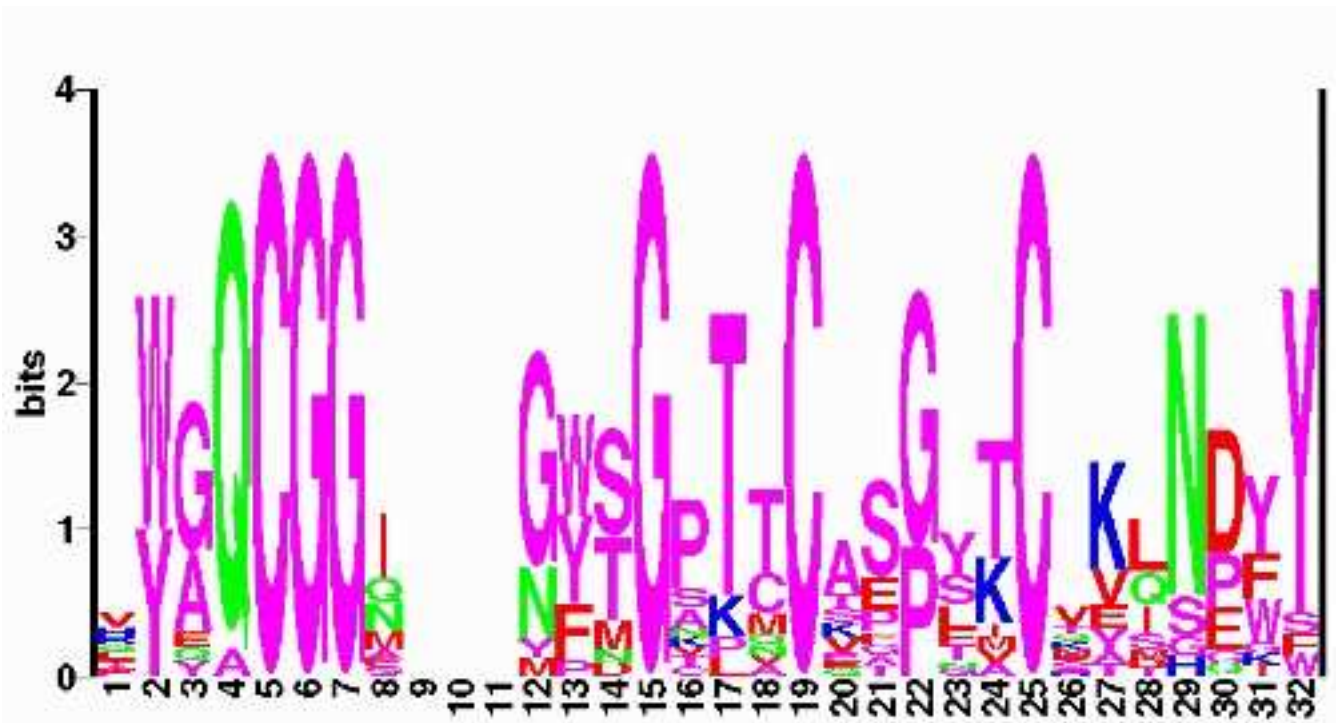


Exemple: alignement de protéines

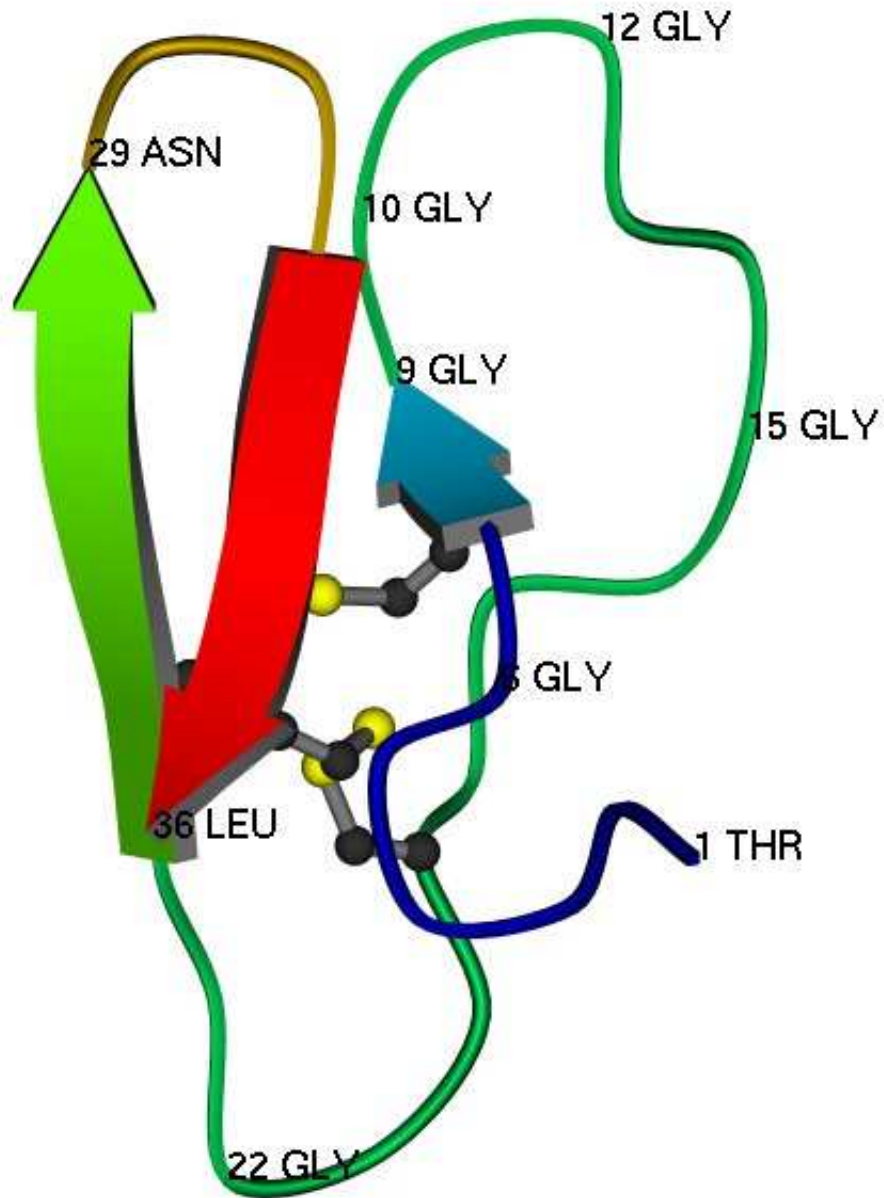
	1	2	3
	45678901...	234567890123456789012	
GUX1_TRIRE/481-509	HYGQCGGI...	GYSGPTVCASGTTTCQVLNPYY	
GUN1_TRIRE/427-455	HWGQCGGI...	GYSGCKTCTSGTTTCQYSNDYY	
GUX1_PHACH/484-512	QWGQCGGI...	GYTGSTTCASPYTCHVLNPYY	
GUN2_TRIRE/25-53	VWGQCGGI...	GWSGPTNCAPGSACSTLNPYY	
GUX2_TRIRE/30-58	VWGQCGGQ...	NWSGPTCCASGSTCVYSNDYY	
GUN5_TRIRE/209-237	LYGQCGGA...	GWTGPTTCQAPGTCKVQNQWY	
GUNF_FUSOX/21-49	IWGQCGGN...	GWTGATTCASGLKCEKINDWY	
GUX3_AGABI/24-52	VWGQCGGN...	GWTGPTTCASGSTCVKQNDFY	
GUX1_PENJA/505-533	DWAQCGGN...	GWTGPTTCVSPYTCTKQNDWY	
GUXC_FUSOX/482-510	QWGQCGGQ...	NYSGPTTCKSPFTCKKINDFY	
GUX1_HUMGR/493-521	RWQQCGGI...	GFTGPTQCEEPYICTKLNDWY	
GUX1_NEUCR/484-512	HWAQCGGI...	GFSGPTTCPEPYTCAKDHDY	
PSBP_PORPU/26-54	LYEQCGGI...	GFDGVTCCSEGLMCMKMPYY	
GUNB_FUSOX/29-57	VWAQCGGQ...	NWSGTPCCTSGNKCVELNDFY	
PSBP_PORPU/69-97	PYGQCGGM...	NYSGKTMCS PGFKVELNEFF	
GUNK_FUSOX/339-370	AYYQCGGSKSAYPNGNLACATGSKCVKQNEY		
PSBP_PORPU/172-200	RYAQCGGM...	GYMGSTMVGGYKCMASEGS	
PSBP_PORPU/128-156	EYAACGGE...	MFMGAKCKFGLVCYETSGKW	
consensus	...QCGG.....G...C.....C.....		

Cellobiohydrolase I

Domaines de fixation de la cellulose



Les positions conservées de la **séquence** donnent des informations sur la **structure**.



Représentation de la structure 3D

Le score SP

Sum of Pairs

- ▷ Score de l'alignement : somme des scores de ses colonnes
- ▷ Comment scorer une colonne ?
 - adaptable à un nombre quelconque d'arguments
 - indépendant de l'ordre
 - reflète la similarité

$$\text{scoreSP} \begin{pmatrix} x_1 \\ \vdots \\ x_k \end{pmatrix} = \sum_{1 \leq i < j \leq k} \text{score}(x_i, x_j)$$

$$x_1, \dots, x_k \in \mathcal{A} \cup \{-\} \text{ et } \text{score}(-, -) = 0$$

- ▷ Exemple

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

Définition alternative (équivalente)

- ▷ α : alignement multiple pour les séquences s_1, \dots, s_k
- ▷ α_{ij} : projection de l'alignement pour s_i et s_j

$$\text{scoreSP}(\alpha) = \sum_{1 \leq i < j \leq k} \text{score}(\alpha_{ij})$$

- ▷ Exemple

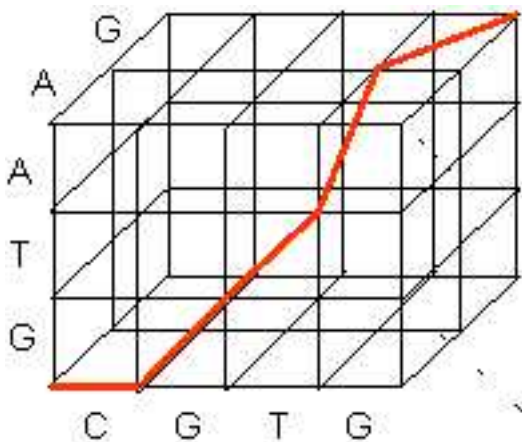
```
A A C G T A C G A T A
A - C G T A - A A T G
G T C G T A - - T T A
```

Problème algorithmique

Trouver l'alignement multiple de score SP maximal.

Approche exacte : programmation dynamique

- ▷ Alignement deux à deux : chemin dans une matrice de dimension 2
- ▷ Alignement multiple : chemin dans une matrice de dimension supérieure



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

Complexité

s_1, \dots, s_k : séquences de taille n

$T(i_1, \dots, i_k)$: score optimal entre les k
préfixes $s_1(1..i_1), \dots, s_k(1..i_k)$

- ▷ Table de taille n^k
- ▷ Temps de calcul d'une case : dépend de $2^k - 1$ cases précédentes
- ▷ Temps de calcul de chaque scoreSP candidat : $k(k - 1)/2$

$$O(n^k 2^k k^2)$$

Le problème de décision associé est NP-complet

Heuristique en étoile

- ▷ Sélection d'une **séquence centrale**



- ▷ Construction de tous les alignements deux à deux entre la séquence centrale et les autres séquences
- ▷ Construction de l'alignement multiple, en partant de la séquence centrale, puis en incorporant une à une les autres séquences.

L'intégration d'une nouvelle séquence se fait en prenant la séquence centrale comme guide. C'est toujours possible en étirant les gaps de l'alignement multiple courant.

Choix de la séquence centrale

ACGACGA TCCGATA TACGATG CACGATCTC

- ▷ Matrice des scores d'alignements 2 à 2

	s_1	s_2	s_3	s_4
s_1	×	0	4	1
s_2	0	×	4	0
s_3	4	4	×	2
s_4	1	0	2	×

Système de score : 1, -1 et -1

- ▷ Règle de sélection : *la séquence qui maximise la somme des scores d'alignements*
- ▷ **Séquence centrale :**

▷ Alignements 2 à 2 de s_3

```

s3      T A C G A T G A
        | | | |   | |
s1      - A C G A C G A
    
```

```

s3      T A C G A T G A
        |   | | | |   |
s2      T C C G A T - A
    
```

```

s3      T A C G A T - - G A
        | | | | |       |
s4      C A C G A T C T C A
    
```

▷ Extension séquence par séquence

```

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

```

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

```

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

Complexité

- ▷ matrice des scores 2 à 2 : $k(k - 1)/2 \times n^2$
 - ▷ construction de l'alignement multiple : $n \times k$
-

Variante

Essayer toutes les séquences comme séquence centrale, et garder l'alignement multiple de score maximal.

Comment une heuristique peut aider à obtenir un résultat optimal.

Principe de Branch & Bound

L'application d'une heuristique donne des informations sur la valeur du score optimal, ce qui permet de restreindre l'espace de recherche dans la matrice de dimension k de l'algorithme exact par programmation dynamique.

Pour ce problème

Case utile : case de la table de programmation dynamique qui appartient à l'alignement optimal.

Case inutile : case qui n'est pas utile.

Algorithme de Carillo-Lipman

▷ S_{ij} : score de l'alignement optimal entre s_i et s_j

▷ $M_{ij}(x, y)$: score de l'alignement optimal entre s_i et s_j qui passe par (x, y) .

$$S(s_i(1..x), s_j(1..y)) + S(s_i(x+1..n), s_j(y+1..n))$$

Propriété (★)

Soit \mathcal{L} , un minorant du score optimal de l'alignement multiple de s_1, \dots, s_k .

Si c_i et c_j satisfont

$$M_{ij}(c_i, c_j) < \mathcal{L} \quad - \quad \sum_{\substack{i' < j' \\ (i', j') \neq (i, j)}} S_{i'j'}$$

alors la case (c_1, \dots, c_k) est inutile.

Application

▷ Détermination d'un $\mathcal{L} \leq \text{scoreSP}(s_1, \dots, s_k)$.

Heuristique, exemple d'alignement

▷ Construction des matrices $M(i, j)$

▷ Programmation dynamique, avec un graphe, au lieu d'une matrice de dimension k .

On part de l'origine, et on construit de proche en proche les cases, sauf celles dont la propriété (*) assure qu'elles sont inutiles.

Gain

Dépend de la valeur de \mathcal{L} et des séquences

Aucune garantie

En pratique

Permet d'aligner une petite dizaine de séquences de longueur 200.

Exemple d'application de l'alignement multiple : phylogénie moléculaire

- ▷ Retracer l'historique des espèces à partir des mutations observées
- ▷ Données : gènes communs aux familles étudiées, pas trop divergents
- ▷ Résultat : classification sous forme d'arbre phylogénétique

Méthodes de parcimonie

Rasoir d'Occam

*"Pluralitas non est
ponenda sine neccesitate"*



- ▷ Privilégier l'arbre qui minimise le nombre de mutations
- ▷ Le nombre global de mutations est obtenu en faisant la somme des mutations le long de chaque branche

Exemple

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

3 arbres non enracinés possibles

D'un point de vue algorithmique

- ▷ Problème de décision associé NP-complet
- ▷ Algorithme exact de *branch and bound*
 - coût initial : L
 - arbre initial : 3 séquences
 - exploration en profondeur de l'espace des solutions en ajoutant une séquence à chaque étape
 - abandon d'une branche dès que l'on rencontre un arbre de coût supérieur à L
 - mise à jour de L quand on rencontre un arbre complet de meilleur score

Permet de traiter jusqu'à 20 à 30 données, surtout si les séquences sont proches

- ▷ Heuristiques . . .

Méthodes de distance

- ▷ Point de départ : alignement multiple
- ▷ Matrice de toutes les distances deux à deux
- ▷ Classification hiérarchique

On construit l'arbre à partir des feuilles en regroupant progressivement les noeuds 2 à 2 pour former des **clusters**.

Exemple : UPGMA

Unweight **P**air **G**roup **M**ethod with **A**rithmetic mean

- ▷ À chaque étape, on regroupe les deux clusters les plus proches.
- ▷ La distance est calculée en faisant la moyenne arithmétique

Reprise de l'exemple précédent

Matrice initiale

	1	2	3	4
1	0	4	5	6
2		0	5	4
3			0	2
4				0

Après une itération

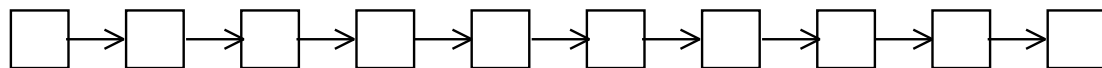
	1	2	3 + 4
1	0	4	5,5
2		0	4,5
3 + 4			0

Le retour des modèles de Markov

Comment représenter un alignement multiple par un modèle de Markov caché ?

1. Si l'alignement n'a pas de gaps

```
... S P A D K T V K A N ...  
T P E E K S A V T A  
S E G E W Q L V L H  
S A D Q I S T V Q A  
S A A E K T K I R S  
T E S Q A A L V K S
```



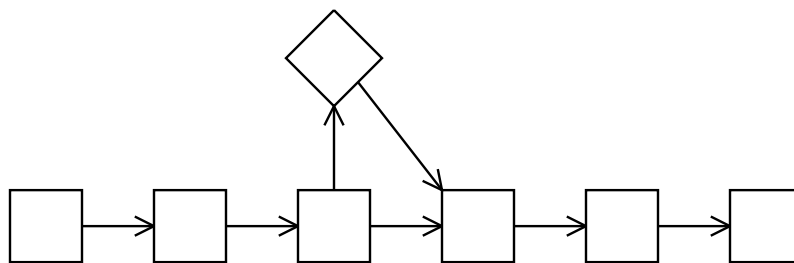
Probabilités de transition:

Probabilités d'émission:

2. Avec des insertions

Une insertion est un fragment de la séquence qui n'apparaît pas dans le modèle.

```
F K D L S - - - - H G N  
F G D L S T P D A V M G N  
F A G - K D L E S I K G T  
F S G - - - - A S - K G N
```



On ajoute un état par bloc inséré.

3. Et finalement, avec les délétions

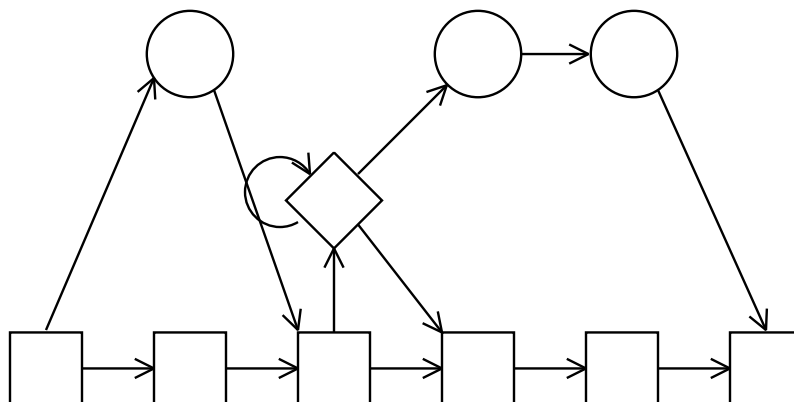
Une délétion est un fragment du modèle qui ne correspond à aucun acide aminé.

```
F - D L S - - - - H G S
F G D L S T P D A V M G N
F A G - K D L E S I K G T
F S G - - - - A S - - - D
```

Option 1: Ajouter des arcs entre les états matchants

Nombre d'arcs quadratique

*Option 2: Ajouter des états **silencieux**, qui n'émettent rien*



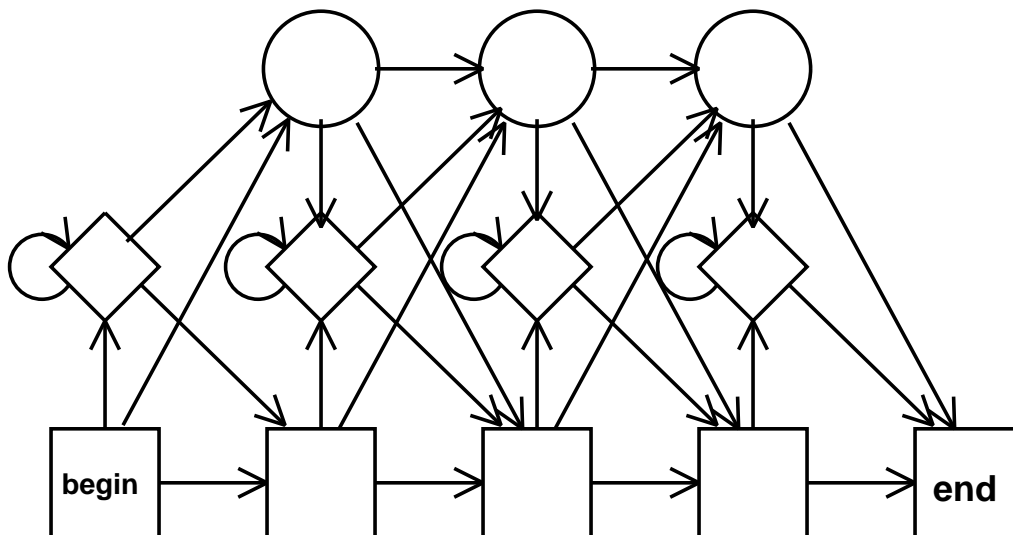
Nombre d'arcs linéaire

Construire un alignement multiple

▷ Choisir le nombre d'états matchants : M

eg: la longueur moyenne des séquences

▷ Construire le modèle complet à $3 \times M + 2$ états

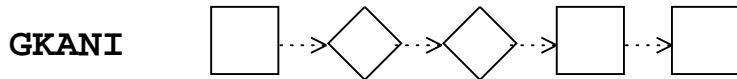
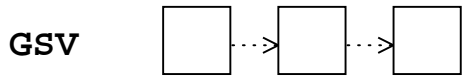
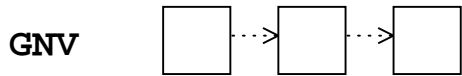
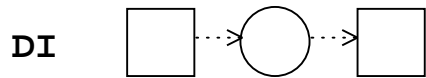


- ▷ Estimer les paramètres du modèles avec l'algorithme de Baum-Welch
 - !! Etape délicate, à cause des maximums locaux
 - Essayer plusieurs jeux de paramètres initiaux
 - Utiliser un minorant fourni par une heuristique
 - ...

- ▷ Aligner chaque séquence avec l'algorithme de Viterbi

Exemple: DI, GNV, GSV, GKANI

Chemins dans le modèle de Markov
(3 états matchants)



D - - - I

G - - N V

G - - S V

G K A N I