

Algorithmique et bioinformatique

Assemblage de fragments d'ADN - Projet

Université de Mons



Année académique 2014 – 2015

Plan

1 Séquençage de génomes

2 Assemblage de fragments

- Le problème
- Complications
- Modèles formels
- Algorithmes
- Heuristiques

3 Assemblage en pratique ?

- Recherche des chevauchements
- Ordonner les fragments
- Alignement et consensus

4 Bibliographie

Séquençage de génomes

Comment sont séquencés les longs génomes ?

Séquencer = déterminer la séquence de bases d'une molécule d'ADN

Contraintes pratiques :

- Erreurs de séquençage ;
- Longueur maximale pouvant être segmentée en une fois :
 $\approx 700pb$
- Impossibilité d'extraire des fragments de $700pb$ en des positions et des orientations précises.

Processus de découpage **aléatoire**.

→ **Problème extrêmement complexe !**

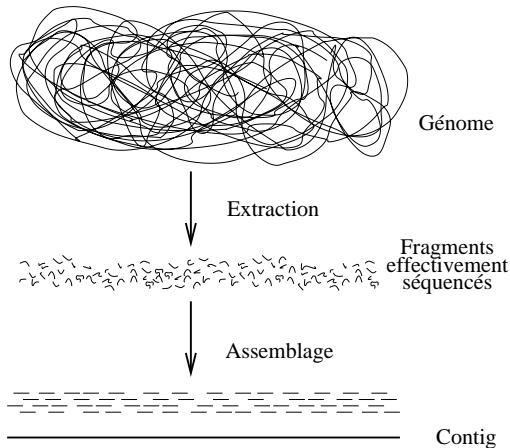
Comment organiser le séquençage de très longues molécules ?

On dispose d'un processus bio-chimique aléatoire capable d'extraire des petits fragments des molécules d'ADN

- Longueur des fragments : $\approx 500\text{pb}$;
- Les positions sont inconnues ;
- L'orientation est inconnue ;

La suite du travail repose sur des méthodes informatiques : utilisation des chevauchements approximatifs pour construire un *contig*("super-séquence")

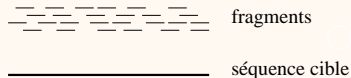
Séquençage de génomes



Le problème

Le problème d'assemblage de fragments

Etant donné une collection de fragments de quelques centaines de paires de bases, il faut construire la séquence cible



ACCGT	\Rightarrow	ACCGT
CGTGC		CGTGC
TTAC		TTAC
TACCGT		TACCGT

		TTACCGTGC
		Consensus

Complications

Erreurs

- Erreurs de séquençage ;
- Erreurs dans le mécanisme de fragmentation et de clonage.

⇒ Substitutions et indels (gaps) : 5%

ACCAGT	⇒	ACCAGT
CGGT		C-G-GT
TTAC		TTAC
TGCCGT		TGCC-GT

		TTACC-GTGT

Consensus "vote de
majorité"

Complications

Orientation inconnue

Les fragments sont orientés 5' → 3' mais sur n'importe quel brin.

CACGT	⇒	CACGT
ACGT		ACGT
ACTACG		CGTAGT
GTACT		AGTAC
ACTGA		ACTGA
CTGA		CTGA

		CACGTAGTACTGA

Considérer les fragments complémentaires et inversés également !

⇒ un fragment sera présent soit tel quel, soit complémenté et inversé **(pas les 2)**.

Complications

Manque de couverture

Mécanisme de fragmentation aléatoire → certaines positions risquent de ne pas être couvertes.

Etc ...

Aucun modèle n'est pleinement satisfaisant !

Plus courte "super-chaîne" commune

INPUT : Collection \mathcal{F} de fragments

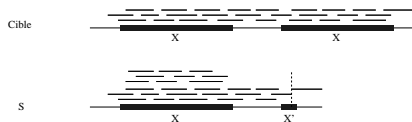
OUTPUT : **Plus courte** chaîne S telle que $\forall f \in \mathcal{F} : f$ facteur de S

Exemple : $\mathcal{F} = \{\text{ACT}, \text{CTA}, \text{AGT}\} \Rightarrow S = \text{ACTAGT}$

INCONVENIENTS :

- Ne permet pas les erreurs
- Les orientations des fragments doivent être connues

Modèles formels



- Couverture irrégulière
- Liaison manquante : aucune suite de fragments chevauchants ne lie le début de X' à la fin de X

Problème NP-Complet \Rightarrow approximations
Bonne base de travail pour le reste

Sans erreur, orientation connue

Représentation des chevauchements

Def : Overlap Multigraph $\mathcal{OM}(\mathcal{F})$: graphe orienté pondéré

- sommets : les fragments $f \in \mathcal{F}$
- arc de $f \in \mathcal{F}$ à $g \in \mathcal{F}$ de poids $t \geq 0$ si $\text{suff}(f, t) = \text{pref}(g, t)$ et $f \neq g$

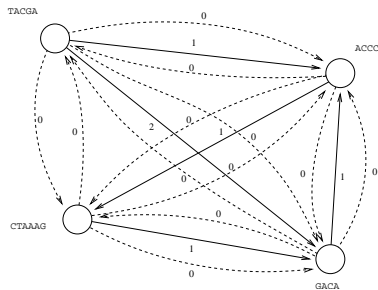
Exemple :

$a = \text{TACGA}$

$b = \text{ACCC}$

$c = \text{CTAAAG}$

$d = \text{GACA}$



Tout chemin décrit un alignement entre des fragments

Un chemin Hamiltonien décrit une super-chaîne

Soient :

- P un chemin de $\mathcal{OM}(\mathcal{F})$
- $A \subseteq \mathcal{F}$ les fragments correspondant à P
- $S(P)$ la super-chaîne dérivée de P

On a $||A|| = |S(P)| + w(P)$ avec $||A|| = \sum_{a \in A} |a|$ et $w(P)$ le poids de P .

Si on ne considère que les chemins Hamiltoniens,
minimiser $|S(P)|$ revient à maximiser $w(P)$

1. Algorithme d'approximation : *Greedy*

On cherche à maximiser les chevauchements \implies pour chaque paire de nœuds de $\mathcal{OM}(\mathcal{F})$, on ne conserve que l'arc de poids maximal.

$\implies \mathcal{OG}(\mathcal{F})$: **Overlap Graph**

On ajoute progressivement les arcs de poids maximaux jusqu'à ce que le chemin contienne tous les nœuds

Heuristiques

- Il faut empêcher la formation de cycles ;
- On peut "entrer" dans un nœud au plus une fois ; et
- On peut "sortir" d'un nœud au plus une fois.

Algorithme : Greedy

Entrées : $\mathcal{OG}(\mathcal{F})$, n

Sortie : Chemin hamiltonien de $\mathcal{OG}(\mathcal{F})$

```
1  pour  $i \leftarrow 1$  a  $n$  faire
2     $in[i] \leftarrow 0$  ;  $out[i] \leftarrow 0$ 
3    MAKESET ( $i$ )

4  Tri des arcs par poids décroissants

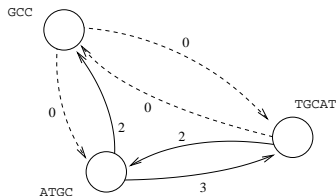
5  pour chaque  $arc(f, g)$  dans cet ordre faire
6    si (  $in[g] = 0$  et  $out[f] = 0$  et FINDSET ( $f$ )  $\neq$  FINDSET ( $g$ ) ) alors
7      SELECT ( $f, g$ )
8       $in[g] \leftarrow 1$  ;  $out[f] \leftarrow 1$ 
9      UNION (FINDSET ( $f$ ), FINDSET ( $g$ ))
10   si il ne reste qu'un seul ensemble alors
11     sortir de la boucle

12 retourner ( $arcs$  choisis)
```

Avec :

- $\text{MAKESET}(i)$: initialise l'ensemble $\{i\}$
- $\text{FINDSET}(f)$: retourne l'ensemble contenant f
- $\text{UNION}(E_1, E_2)$: fusionne les 2 ensembles E_1 et E_2
- $\text{SELECT}(f, g)$: choisit $\text{arc}(f, g)$

Exemple :



GREEDY :

ATGC		
TGCAT		
	GCC	→ 9

ATGCATGCC		

2. Tri topologique

S'applique dans le cas de graphes acycliques.
Pas vu ici.

Assemblage en pratique ?

3 étapes :

- ➊ rechercher les chevauchements approximatifs ;
- ➋ construire l'alignement ;
- ➌ calculer la séquence consensus.

Les méthodes font souvent appel à des heuristiques

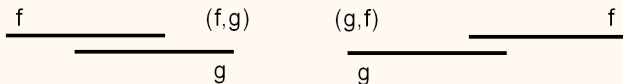
Assemblage en pratique ?

1. Recherche des chevauchements

Considérer toutes les paires de fragments et leurs complémentaires.

⇒ Alignement semi-global par programmation dynamique

Remarque : Considération des 8 cas d'alignement (f, g) , (g, f) , (\bar{f}, g) , ...



Assemblage en pratique ?

		A	T	C	G	G	C	A	T	T	C	A	G	T
	+	-----												
		0	0	0	0	0	0	0	0	0	0	0	0	0
A		0	1	-1	-1	-1	-1	-1	1	-1	-1	-1	1	-1
T		0	-1	2	0	-2	-2	-2	-1	2	0	-2	-1	0
T		0	-1	0	1	-1	-3	-3	-3	0	3	1	-1	-2
A		0	1	-1	-1	0	-2	-4	-2	-2	1	2	2	0
G		0	-1	0	-2	0	1	-1	-3	-3	-1	0	1	3
A		0	1	-1	-1	-2	-1	0	0	-2	-3	-2	1	1
C		0	-1	0	0	-2	-3	0	-1	-1	-3	-2	-1	0
C		0	-1	-2	1	-1	-3	-2	-1	-2	-2	-2	-3	-2
A		0	1	-1	-1	0	-2	-4	-1	-2	-3	-3	-1	-3
T		0	-1	2	0	-2	-1	-3	-3	0	-1	-3	-3	-2
G		0	-1	0	1	1	-1	-2	-4	-2	-1	-2	-4	-2
C		0	-1	-2	1	0	0	0	-2	-4	-3	0	-2	-4
G		0	-1	-2	-1	2	1	-1	-1	-3	-5	-2	-1	-1
G		0	-1	-2	-3	0	3	1	-1	-2	-4	-4	-3	0
C		0	-1	-2	-1	-2	1	4	2	0	-2	-3	-5	-2

ATTAGACCATGCGGC

AT CGGCATTTCAGT

Assemblage en pratique ?

2. Ordonner les fragments (greedy ou tri topologique)

Les sommets du graphe : $\mathcal{DF} = \mathcal{F} \cup \overline{\mathcal{F}}$ avec $\overline{\mathcal{F}} = \{\bar{f} | f \in \mathcal{F}\}$

Si un fragment apparaît dans le chemin, son complémentaire ne peut plus être choisi !

Remarques :

- Des fragments peuvent être inclus à d'autres. Adaptez le comportement de l'algorithme greedy à ce cas particulier.

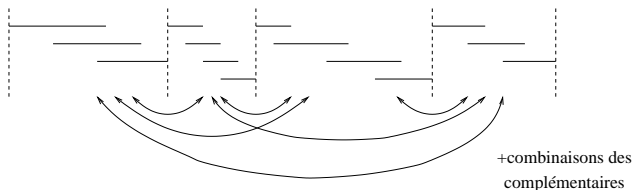
Assemblage en pratique ?

Remarques (suite) :

- le complémentaire inversé d'un "bon chemin" est également un "bon chemin" :

$$f_1 \rightarrow f_2 \rightarrow \dots f_k \text{ et } \overline{f_k} \rightarrow \overline{f_{k-1}} \rightarrow \dots \overline{f_1}$$

- Une multitude de solutions sont valides si plusieurs contigs. Il s'agit là d'un problème de liaison.



Assemblage en pratique ?

3. Alignement et consensus

Réaliser l'alignement obtenu et calculer la séquence consensus.

Bibliographie



J. Setubal et J. Meidanis

Introduction to Computational Molecular Biology - Chap.4.

PWS Publishing Company, 1997.