Alignement de séquences : application à la comparaison de chaînes d'ADN

Sèverine BÉRARD*
LIRMM, UMR CNRS 5506, 161, rue Ada, 34392 Montpellier cedex 5
berard@lirmm.fr
Directeur de thèse : Olivier GASCUEL   Co-encandrant : Éric RIVALS

Résumé

Mon travail de recherche est le développement de nouveaux algorithmes pour aligner des séquences sous un modèle particulier. Une application biologique de ces recherches est la comparaison de cartes de minisatellites.

Abstract

My work consists in finding new algorithms to align sequences under a special model. A biological application of this work is the comparison of minisatellite maps.

1 Introduction

Le cadre de mon travail est l'algorithmique du texte. Je travaille sur la comparaison de textes, mots ou séquences par alignement. L'intérêt est de pouvoir mesurer la similarité des deux séquences que l'on compare. Un algorithme d'alignement prend en paramètre deux séquences et renvoie la distance qui les sépare. Les calculs sont usuellement effectués par programmation dynamique. Les biologistes utilisent l'alignement pour comparer de nouvelles séquences d'ADN à des séquences déjà étudiées. Cela leur permet de déterminer les séquences les plus proches et avoir ainsi une idée sur la fonction de la nouvelle séquence.
La suite de l'article s'organise de la manière suivante, dans la Section 2, je détaillerai un algorithme classique d'alignement de séquences par programmation dynamique. Ensuite, dans le Section 3, je présenterai le modèle particulier d'évolution de séquence sous lequel je travaille. Enfin, dans la Section 4 je montrerai les applications biologiques de l'alignement de séquences sous le modèle présenté en Section 3. La Section 5 est un glossaire dans lequel on peut trouver la définition des mots soulignés de l'article.

2 Alignements

Nous nous intéressons ici à l'alignement global entre deux séquences. Les premiers à avoir utilisé la programmation dynamique pour l'alignement de séquences

*Allocataire de recherche - Monitrice - Thèse débutée le 1er Octobre 2000.

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

FIG. 1 – Exemple d'alignement

		A	G	G	T	C	A
	0	3	6	9	12	15	18
A	3	0	3	6	9	12	15
G	6	3	0	3	6	9	12
C	9	6	3	2	5	6	9
C	12	9	6	5	4	5	8
A	15	12	9	8	7	6	5

FIG. 2 – Alignement de AGCCA et AGGTCA, avec I=D=3 et M=2.

génétiques sont Needleman et Wunsch [NW70]. On compare deux séquences en construisant un alignement, c.à.d. en mettant en correspondance chacun des caractères soit avec un caractère de l'autre séquence, soit avec un symbole "-". Un exemple d'alignement est donné à la Figure 1, les barres verticales signalent des caractères identiques.
Le mise en correspondance d'un caractère et de "-" est appelée une *insertion* du point de vue de la séquence du haut, ou symétriquement une *délétion* du point de vue de la séquence d'en bas. La mise en correspondance de deux caractères différents est une *mutation*. On dispose de ces 3 opérations pour transformer les séquences. La distance d'alignement est la somme des coûts des opérations qui le composent, on donne donc un coût à chacune des opérations, notés I, D et M respectivement. Pour former un alignement entre deux séquences, S de longueur n et R de longueur m, on construit progressivement une matrice dans laquelle la case de coordonnées (i, j) contient la distance entre le <u>préfixe</u> de longueur i de S et celui de longueur j de R. La case (n, m) de la matrice contient donc la distance entre S et R. Notons S[i] le caractère à la position i dans S, et S_i le préfixe de longueur i de S. Pour obtenir un alignement entre S_i et R_j, il suffit de considérer les trois alignements suivants :

- S_{i-1} et R_{j-1} et d'ajouter la mutation de S[i] en R[j]
- S_i et R_{j-1} et d'ajouter l'insertion de R[j]
- S_{i-1} et R_j et d'ajouter le délétion de S[i]

ensuite choisir celui qui génère le coût minimum. Si on note M la matrice, cela donne :

$$M(i, j) = min \begin{cases} M(i-1, j-1) + M \\ M(i, j-1) + I \\ M(i-1, j) + D \end{cases}$$

On se sert de cette équation de récurrence pour remplir, ligne par ligne, de la gauche vers la droite, chaque case (i, j) avec $i > 0$ et $j > 0$. La matrice de programmation dynamique est de dimension $n + 1$ par $m + 1$, la numérotation des lignes et des colonnes commence à 0. L'initialisation de la matrice se fait de la manière suivante : $M(0, 0) = 0, M(i, 0) = i * D, M(0, j) = j * I$. Un exemple de matrice de programmation dynamique est montré à la figure 2. Lorsque la matrice est calculée, il faut faire un parcours arrière pour retrouver un alignement, il existe plusieurs alignements optimaux. Le chemin grisé de la matrice de la Figure 2 correspond à l'alignement de la Figure 1.

3 Modèle particulier d'évolution de séquences

	1	2	3	4	5	6	7	8
S_1	a	e	a	a	–	–	–	a
	|	]	|	]	\		\	|
S_2	a	a	a	b	b	c	b	a

FIG. 3 – Alignement de deux séquences, S_1=aeaaa et S_2=aaabbcba. '|', ']' et '\' représentent respectivement une identité, une mutation et une amplification.

Nous nous plaçons maintenant dans le cadre d'un modèle symétrique et unaire. Ce modèle considère *cinq* opérations, la mutation (M), l'insertion (I), la délétion (D), l'amplification (A) et la contraction (C). Mutation, insertion et délétion sont les événements que l'on a vus précédemment. Les deux événements spécifiques sont l'amplification et la contraction. Par exemple, la séquence abc subit l'amplification du motif b, puis sa contraction :

amplification : $abc \longrightarrow abbc$
contraction : $abbc \longrightarrow abc$.

Ce modèle est unaire car l'amplification (resp. la contraction) ajoute (resp. retire) un seul motif à la fois. L'amplification et la contraction ont un coût plus faible que les autres opérations.
Un exemple d'alignement de séquences sous ce modèle est donné à la Figure 3. Pour obtenir le meilleur alignement, il faut faire les amplifications des b avant l'insertion du c, cela montre que l'ordre d'application des opérations est important et c'est là que réside la difficulté du problème de trouver un alignement optimal sous ce modèle. Pour plus de détails vous pouvez vous référer à [BR02c].

4 Applications

L'application de ce travail est la comparaison de cartes de minisatellites. Ce sont des séquences d'ADN qui appartiennent à la classe des séquences répétées en tandem. Ces séquences sont constituées de répétitions en tandem, i.e., côte à côte, d'un motif unitaire. Les minisatellites subissent des fluctuations de leur nombre de motifs, ils sont dits *polymorphes*. Ce polymorphisme est dû à la *duplication en tandem*, un événement qui ajoute un motif copié à côté de l'original, et de l'événement inverse, la *délétion en tandem*. La variation de longueur des minisatellites est impliquée dans plusieurs maladies comme le diabète, l'épilepsie et le cancer. Les minisatellites sont utilisés pour la cartographie génétique et les études médico-légales.

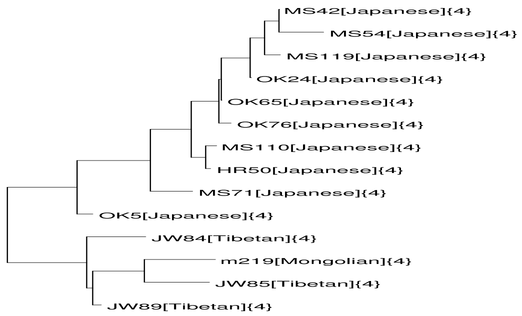

FIG. 4 – Arbre obtenu sur l'haplogroupe 4.

En 1991, Jeffreys et ses collaborateurs ont mis au point une technique PCR spécifique aux minisatellites, la MVR-PCR [JMT+91], où MVR signifie Minisatellites Variant Repeat. Cette méthode fournit une *carte du minisatellite*, càd une séquence de symboles, où chaque symbole est associé de manière bijective avec une variante du motif. L'évolution des minisatellites peut être représentée par le modèle décrit dans la Section 3, où l'amplification correspond à la duplication en tandem et la contraction, à la délétion en tandem.

Nous avons appliqué notre algorithme au minisatellite humain MSY1 qui est situé sur le chromosome Y [JBT98]. Le jeu de données contient 609 cartes d'individus répartis en 27 haplogroupes. La figure 4 représente l'arbre des individus de l'haplogroupe 4 reconstruit à partir de notre distance. On constate la séparation entre les japonais d'un côté, et les tibétains et le mongol de l'autre. D'autres résultats sont en cours d'investigation.

5 Glossaire

ADN L'ADN, acide désoxyribonucléique, est une molécule en forme de double hélice. C'est un constituant universel de la matière vivante, il est situé au sein des chromosomes dans le noyau des cellules.

Programmation Dynamique La programmation dynamique est une technique algorithmique qui résout chaque sous-problème une seule fois et mémorise sa solution dans une matrice, épargnant ainsi le recalcul de la solution chaque fois que le sous-problème est rencontré.

Préfixe Un préfixe de longueur m d'une séquence S de longueur n, avec $m \leq n$, est une séquence de longueur m constituée des m premiers caractères de S.

6 Publications

Ce travail a donner lieu a plusieurs publications, [BR02c], dans une conférence internationale de BioInformatique, [BR02a] dans la conférence française de BioInformatique et [BR02b], dans une conférence française de génétique et biologie des populations.

Références

[BR02a] Sèverine Bérard and Eric Rivals. Comparaison de minisatellites. In J. Nicolas and C. Thermes, editors, *Actes des Journées Ouvertes Biologie Informatique Mathématiques (JOBIM)*, pages 261–262, 2002.

[BR02b] Sèverine Bérard and Eric Rivals. Comparaison de minisatellites. In *Actes de la 24ème réunion annuelle du Groupe de Génétique et Biologie des Populations*, page 128, 2002.

[BR02c] Sèverine Bérard and Eric Rivals. Comparison of Minisatellites. In *Proc. of the Sixth Annual International Conference on Computational Molecular Biology (RECOMB)*, 2002.

[JBT98] M. A. Jobling, N. Bouzekri, and P. G. Taylor. Hypervariable digital DNA codes for human paternal lineages : MVR-PCR at the Y-specific minisatellite, MSY1 (DYF155S1). *Hum Mol Genet*, 7(4) :643–53, 1998.

[JMT+91] A. J. Jeffreys, A. MacLeod, K. Tamaki, D. L. Neil, and D. G. Monckton. Minisatellite repeat coding as a digital approach to DNA typing. *Nature*, 354(6350) :204–9, 1991.

[NW70] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol*, 48(3) :443–53, Mar 1970. (eng).