Comparaison de séquences avec amplifications et contractions

Sèverine Bérard[1] et Éric Rivals[1]

1. LIRMM, UMR CNRS 5506, 161, rue Ada, 34392 Montpellier cedex 5
{berard,rivals}@lirmm.fr

Mots-clefs : alignement, répétition en tandem, minisatellite, programmation dynamique, graphe de recouvrement.

Nous présentons ici une méthode pour comparer des séquences sous un modèle incluant deux opérations spécifiques, l'*amplification*, un événement qui ajoute un motif copié à côté de l'original, et à l'événement inverse, la *contraction*. Nous proposons un algorithme pour trouver l'alignement optimal entre deux séquences qui combine programmation dynamique et recherche de stable max dans un graphe. Cet algorithme donne un *score* d'alignement qui est une distance métrique. Nous pouvons appliquer ce travail à des séquences génétiques évoluant selon ce mode particulier, de manière à reconstruire des relations évolutives entre individus ou populations.

	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. 1 – Alignement de deux séquences, S_1=aeaaa et S_2=aaabbcba. '|', ']' et '\' représentent respectivement une identité, une mutation et une amplification ; la mise en correspondance de '-' et d'un symbole représente une insertion.

Nous nous plaçons dans le cadre d'un modèle évolutif symétrique et unaire. Notre modèle considère cinq événements évolutifs s'appliquant aux motifs, la mutation (M), l'insertion (I), la suppression (S), l'amplification (A) et la contraction (C). Mutation, insertion et suppression sont les événements que l'on considère traditionnellement dans l'alignement de séquences. 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 : $abbc \longrightarrow abc$.
Notons que l'amplification peut insérer un motif x à une position $i+1$ seulement si un motif x est déjà dans la séquence à la position i, et que la contraction peut supprimer un motif y à la position j seulement si un autre motif y se trouve à la position $j+1$. Notre modèle est unaire car l'amplification (resp. la contraction) ajoute (resp. retire) un seul motif à la fois. Pour compléter ce modèle, nous avons besoin d'un critère quantitatif permettant d'estimer la similarité des séquences. Aussi, à chaque opération est associé un coût réel positif. Une séquence d'événements qui transforme une séquence S_1 en une séquence S_2 est appelée un *alignement*, un exemple est donné à la Figure 1. Le coût d'un alignement est la somme des coûts des opérations qui le composent. L'amplification et la contraction ont un coût plus faible que les autres opérations. Cependant, ces deux opérations sont soumises à conditions. La recherche de l'alignement optimal est difficile en raison de la non-commutativité des opérations.
L'exemple de la Figure 1 souligne l'importance de l'ordre des opérations. Il y a deux manières d'aligner les séquences S_1 et S_2 , et notamment la position 4 de S_1 avec les positions 4 à 7 de S_2 :

1. Muter le a, à la position 4 de S_1 en b, amplifier le b ainsi obtenu, insérer un c en position 6, puis insérer un b en position 7.
2. Muter le a, à la position 4 de S_1 en b, amplifier 2 fois le b obtenu, puis insérer le c en position 6. (cas représenté à la Figure 1)

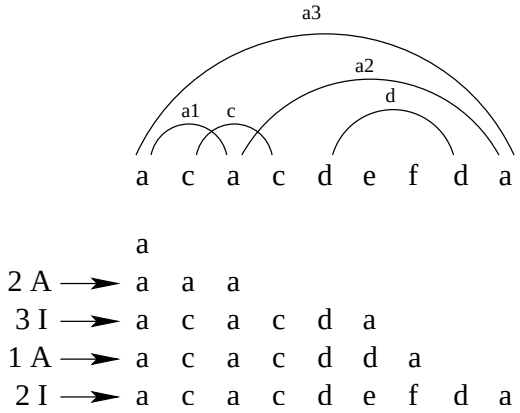

FIG. 2 – Génération optimale de l'arche acacdefda à partir du motif a.

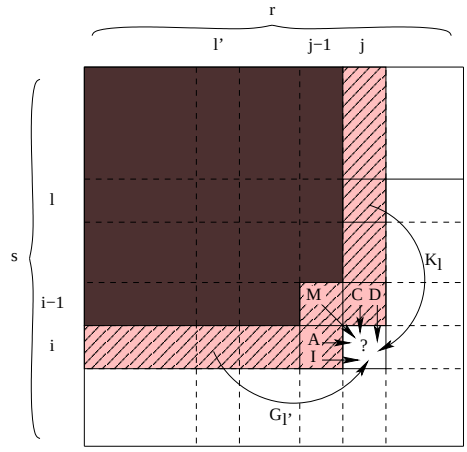

FIG. 3 – Dépendances dans la matrice de programmation dynamique.

Notons que le b de la position 7 dans S_2 provient d'une amplification dans le cas 2 et d'une insertion dans le cas 1, or l'amplification est moins coûteuse que l'insertion. Les opérations restantes étant identiques, la deuxième manière est donc moins coûteuse que la première. Pour obtenir le meilleur alignement, il faut faire les amplifications avant l'insertion du c ; cela montre que les opérations ne sont pas commutatives et donc que leur ordre d'application est important. Pour tout segment de l'alignement où l'ordre d'application des opérations n'est pas de gauche à droite (comme dans l'exemple de la Figure 2), nous calculerons cet ordre et le coût par une procédure spécifique. Nous montrons que ces segments correspondent à des facteurs que nous appelons *arche*. Une arche est un facteur de longueur supérieure à 2 dont le premier et le dernier motif sont identiques. Nous alignons une arche avec un seul motif de la séquence en face. Par exemple, le facteur de S_2 commençant à la position 4 et terminant à la position 7, est une arche de S_2. Nous pouvons traiter les arches de manière indépendante du reste de l'alignement. Ainsi, nous avons introduit les opérations duales de génération (G) et de compression (K) d'arches. Une arche peut elle-même contenir d'autres arches, des sous-arches, comme par exemple l'arche a_3 de la Figure 2 contient les arches a_1, a_2, c, et d. Intuitivement, la résolution optimale d'une arche utilise un nombre maximum de sous-arches. Or toutes les arches ne sont pas compatibles entre elles. Donc calculer le coût des opérations G et K nécessite de trouver l'ensemble de sous-arches deux à deux compatibles de cardinal maximum. Nous montrons que ce problème revient à calculer un stable max dans le graphe de recouvrement G(X, E) tel que X est l'ensemble des arches, et qu'il existe une arête entre deux sommets de X ssi les arches qu'ils représentent sont incompatibles. Dans l'exemple de génération de la Figure 2, on utilise les arches a_1, puis a_2 et enfin d. Pour plus de détails vous pouvez vous référer à [1].
Comme dans les algorithmes classiques d'alignement, nous calculons une matrice de programmation dynamique où l'entrée (i, j) donne le score de l'alignement des préfixes de longueur i et j de S_1 et S_2 resp. Notre modèle nécessite des dépendances plus complexes entre cellules comme illustré à la Figure 3. La complexité en temps de l'algorithme est en $O(n^4)$, où n est la taille de la plus longue des deux séquences à aligner.
Nous avons appliqué notre algorithme au minisatellite humain MSY1 qui est situé sur le chromosome Y [2]. Cela permet d'étudier le polymorphisme entre humains.

Références

[1] Sèverine Bérard and Eric Rivals. Comparison of Minisatellites. In S. Istrail P. Pevzner G. Myers, S. Hannenhalli and M. Waterman, editors, *Proc. of the Sixth Annual International Conference on Computational Molecular Biology*, pages 67–76, Washington DC, USA, 2002. ACM Press.

[2] 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.