BIN702 - Série d'exercices sur BLAST

Manuel Lafond

Exercice 1: Rappelez-vous que dans l'algorithme BLAST, une des étapes "fusionne" des matches, ce qui correspond à joindre deux diagonales. Rappelons les définitions. On a deux séquences S et T et deux paramètres k et d. Ici, k est la longueur des matches voulus et d est l'espace maximum permis entre deux matches à fusionner.
Un **match** est une paire (i, j) telles que les sous-chaînes $S[i..i+k-1], T[j..j+k-1]$ qui sont égales. Un **match étendu** est une paire de matches (i, j), (i', j') tels que les conditions suivantes sont satisfaites:

- $i < i', j < j'$;
- $i' - i = j' - j \leq d$;

donc le "trou" est de la même longueur et n'est pas supérieur à d.
Étant donné S, T, k et d, donnez un algorithme qui trouve tous les matches, puis tous les matches étendus. Quel est la complexité de votre algorithme?

Exercice 2: Rappelons que BLAST cherche, pour tout k-mer C de S, toutes les occurrences de mots C' qui ne sont pas trop distants de C.

a. Prenons la distance de Hamming pour trouver ces C'. C'est-à-dire, la distance de Hamming entre deux chaînes de même longueur est le nombre de positions où les chaînes ont un caractère différent. Donnez un algorithme qui, étant donné une séquence C de longueur k et un entier d retourne tous les mots C' à distance Hamming au plus d de C. Quelle est la complexité?

b. Pour compliquer un peu les choses, donnez un algorithme qui, étant donné une séquence C de longueur k et un entier d retourne tous les mots C' à distance **Levenshtein** au plus d de C.

   Soit b le nombre de chaînes à énumérer. Quelle est la complexité de votre algorithme par rapport à b? Peut-on le faire en temps $O(b)$? (note: je

n'ai pas la réponse à cette dernière question, mais c'est instructif d'y réfléchir)

Exercice 3: Montrez que si P et T sont deux chaînes aléatoires sur alphabet Σ, le nombre espéré de sous-chaînes égales entre P et T de longueur exactement k est $(m-k+1)(n-k+1)\cdot|\Sigma|^{-k}$.
(note : ceci demande un tout petit peu de connaissances en prob. et stats. et je ne demanderais pas ça en examen. Je recommande d'utiliser la linéarité de l'espérance.)