Analyse d'algorithmes de recherche de motifs

La recherche de motifs est un problème fondamental en informatique. Il s'agit de détecter dans une structure la présence de sous-structures d'un type donné, les motifs. Les structures considérées sont le plus souvent des mots (séquences finies de symboles), des arbres ou des graphes. Ici nous nous intéresserons aux mots ou textes et les motifs seront donnés sous la forme d'un ensemble de mots facilement représentable : un seul mot, un ensemble fini de mots, ou encore un ensemble rationnel de mots.

Tester l'apparition d'un motif dans un texte, compter les occurrences de ce motif, déterminer l'ensemble des positions de ces occurrences sont autant de questions centrales dans plusieurs branches de l'informatique notamment en bio-informatique. Elles ont été largement étudiées selon différentes approches : algorithmique [3, 4], analytique [8, 10, 2], probabiliste [8]. De nombreux algorithmes existent pour les traiter. L'objectif de cette thèse est d'analyser certains de ces algorithmes du point de vue de la complexité moyenne en temps et en espace mais aussi le cas échéant en terme d'efficacité. Cette étude théorique systématique devrait aider à optimiser les algorithmes existants ou aider à concevoir de nouveaux algorithmes plus efficaces.

L'analyse en moyenne d'un algorithme consiste à étudier sa complexité non plus dans le pire des cas, mais dans le cas moyen ; cela suppose qu'on a défini une distribution sur les objets étudiés : le texte dans lequel on fait la recherche et l'ensemble des motifs recherchés. La pertinence des différentes distributions dépend des applications considérées. Pour un texte, par exemple, on pourra considérer qu'il est tiré uniformément au hasard parmi tous les textes de même taille ou qu'il est engendré par une chaîne de Markov, modèle classique en bio-informatique. L'analyse en moyenne repose sur une étude combinatoire des objets manipulés tant d'un point de vue analytique qu'algébrique, ainsi que sur une analyse fine des algorithmes considérés. Les techniques utilisées (séries génératrices, analyse de singularités, analyse asymptotique multivariée...) sont issues de la combinatoire analytique développée par Flajolet et Sedgewick [5]. Des résultats plus spécifiques à l'analyse d'algorithmes du texte sont présentés dans [11].

Une premier ensemble d'algorithmes de recherche de motif est constitué par les algorithmes de recherche d'un motif X donné utilisant un automate déterministe qui reconnaît l'ensemble des mots qui se terminent par un élément de X. Dans le cas où X est réduit à un seul mot, il s'agit de l'algorithme de Knuth-Morris-Pratt [7]; si X est un ensemble fini de mots, une solution est due à Aho et Corasick ; enfin, Mohri [9] a proposé un algorithme pour les motifs rationnels. Du point de vue de la complexité en espace, la faiblesse de ces algorithmes est d'utiliser un automate déterministe qui n'est pas minimal dès que le motif n'est pas réduit à un mot. Une solution pour optimiser l'espace mémoire utilisé pourrait consister à construire l'automate minimal des ensembles considérés. La difficulté étant d'effectuer cette construction de manière efficace.

Les filtres à base de *q-grams* forment une autre famille d'algorithmes plus récents intervenant dans la recherche de motifs. Il s'agit, dans le contexte de la recherche de motifs ap-

prochée, d'éliminer certaines parties du texte à traiter à l'aide d'un filtre. Ce dernier consiste essentiellement à tester si les facteurs de longueur q contenus dans une portion du texte satisfont certaines conditions. Le but est de limiter au maximum les parties de texte surlequelles opérera l'algorithme de recherche de motifs. Ces méthodes de filtration se révèlent expérimentalement très efficaces quand la proportion d'erreurs tolérée reste petite par rapport à la longueur du motif. Leurs comportements montrent des phénomènes de transition de phase aussi bien en fonction de la longueur q des facteurs testés que du taux d'erreur k toléré dans le motif. Une étude théorique de ces phénomènes permettrait d'estimer l'efficacité de ces filtres et d'aider à l'ajustement de la longueur des facteurs testés.

Bibliographie

[1] A.V. Aho, M.J. Corasick, Efficient string matching: an aid to bibliographic search, *Communications of ACM*, 18, p. 333-340, 1975.

[2] F. Bassino, J. Clément, J. Fayolle, P. Nicodème, Counting occurrences for a finite set of words: an inclusion-exclusion approach. In *Proceedings of the 2007 Conference on Analysis of Algorithms* (2007), P. Jacquet, Ed., DMTCS, proc. AH, pp. 29–44. Proceedings of a colloquium organized by Juan-les-Pins, France, June 2007.

[3] M. Crochemore, C. Hancart, T. Lecroq, *Algorithmique du texte*, Vuibert, 2001,

[4] M. Crochemore, W. Rutter *Jewels of stringology*, World Scientific, 2002.

[5] P. Flajolet, R. Sedgewick. *Analytic combinatorics*, in preparation, (Version of January 2, 2008 is available at http://www.algo.inria.fr/flajolet/publist.html).

[6] J.E. Hopcroft, J.D.Ullman *Introduction to Automata Theory, Languages and Computation.* Addison-Weisley Publishing Company, 1979.

[7] D.E. Knuth, J.H. Morris, V.R. Pratt, Fast pattern matching in string *SIAM Journal in Computing*, 6, p. 323-350, 1977

[8] M. Lothaire, *Applied Combinatorics on Words.* Encyclopedia of Mathematics. Cambridge University Press, 2005.

[9] M. Mohri String-matching with automata *Nordic Journal of Computing*, 4, p. 217-231, 1997.

[10] P. Nicodème, B. Salvy, P. Flajolet, Motif statistics. *Theoretical Computer Science 287*, 2 (2002), 593–618.

[11] W. Szpankowski, *Average Case Analysis of Algorithms on Sequences.* Series in Discrete Mathematics and Optimization. John Wiley & Sons, 2001.