INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Équipe adage

Algorithmique Discrète et ses Applications à la GÉnomique

Lorraine

THÈME 2B

Table des matières

1. Composition de l'équipe

ADAGE *est un avant-projet du LORIA (UMR 7503) commun au CNRS, à l'INRIA, à l'Université* HENRI POINCARÉ *Nancy 1, à l'Université Nancy 2 et à l'Institut National Polytechnique de Lorraine.*

Responsable scientifique
Grégory Kucherov [CR INRIA]

Assistant(e) de projet
Hélène Zganic [TR INRIA à 1/4 du temps]

Personnel CNRS
Jean-Luc Rémy [affecté aux activités syndicales à hauteur de 492 heures annuelles, CR]
Gilles Schaeffer [CR]

Personnel Université
Isabelle Debled-Rennesson [Maître de conférences, IUFM de Lorraine]
Jocelyne Rouyer [Maître de conférences, UHP]

Doctorant
Laurent Noé [Allocataire MJENR]

Ingénieur associé
Ghizlane Bana [INRIA]

Chercheur invité
Roman Kolpakov [INRIA, du 01/06/2002 au 31/08/2002]

Stagiaires
Patricia Lavigne [DESS bioinformatique de Rouen]
Sandra Luis [DESS IDC, du 01/06/2002 au 31/10/2002]

2. Présentation et objectifs généraux

L'avant-projet ADAGE a été créé au 1/1/2001 suite à la restructuration du projet POLKA. L'objectif général d'ADAGE consiste à mettre au point des algorithmes efficaces sur les structures discrètes (telles que mots, arbres, graphes, cartes, polyominos, ...). Cet objectif nous conduit à étudier en profondeur des propriétés combinatoires de ces structures, qui peuvent être de nature exacte ou probabiliste.

Nos recherches sont structurées en trois actions. La première porte sur l'algorithmique et la combinatoire des mots. Ici, nous travaillons sur l'analyse de complexité de problèmes sur les mots (textes, ou séquences de caractères) et sur le développement d'algorithmes efficaces d'analyse de mots. La deuxième action de recherche relève du domaine de la géométrie discrète. Les structures étudiées ici sont des objets géométriques discrétisés, décrits par un ensemble de points dans $\mathbb{Z}^2$ ou $\mathbb{Z}^3$. Comme dans le cas précédent, nous cherchons à mettre au point des algorithmes efficaces sur ces structures, qui vérifient leurs propriétés ou calculent des paramètres géométriques. La troisième action considère les modèles discrets sous un angle probabiliste, en supposant en général une distribution de probabilité sur l'espace de modèles possibles. Nous nous intéressons donc aux propriétés typiques des structures en question, et nous nous attachons en particulier aux méthodes d'analyse de ces propriétés.

Le champ d'application privilégié de nos travaux est la bioinformatique, domaine dans lequel les modèles discrets apparaissent de façon naturelle et essentielle. Ici, nous poursuivons des collaborations actives avec des équipes de biologistes sur des problèmes d'analyse de séquences d'ADN et de protéines.

Nous prêtons une attention particulière à la mise en place de logiciels expérimentaux basés sur des algorithmes issus de nos travaux. Deux logiciels d'analyse de séquences d'ADN sont actuellement développés dans l'avant-projet : le premier, **mreps**, permet de rechercher des répétitions dites en tandem ; le deuxième, appelé YASS, recherche des régions de similitude entre deux séquences ou à l'intérieur d'une seule. Un autre logiciel de recherche de motifs, **grappe**, a été développé antérieurement.

3. Fondements scientifiques

Mots clés : *algorithmique discrète*, *structures discrètes*, *complexité*, *algorithmique des mots*, *recherche de motifs*, *géométrie discrète*, *aléa discret*, *analyse d'algorithmes*.

Si l'on voulait définir la problématique de notre avant-projet par approximations successives, il serait naturel de commencer par la placer dans le domaine de l'*algorithmique discrète*. Construire un modèle discret d'un problème ou d'un phénomène du monde réel fait appel, sur le plan mathématique, aux *structures discrètes*, telles que graphes, mots, arbres, ensembles de points dans un espace, etc.

Pour pouvoir utiliser les modèles discrets, nous sommes donc amenés à étudier les propriétés des structures impliquées. En tant qu'informaticiens, nous nous intéressons aux *propriétés algorithmiques*, en particulier à l'*efficacité* (*complexité*) des calculs impliqués, que ce soit *en moyenne* ou *dans le cas le pire*.

Pour pouvoir développer des algorithmes efficaces sur les structures discrètes, ainsi que pour analyser et optimiser ces algorithmes, il faut donc comprendre et maîtriser les propriétés des structures sous-jacentes. Ces propriétés peuvent être de natures différentes : s'il s'agit de propriétés de nature exacte, on parle de *propriétés combinatoires* ; si le modèle en question est défini en termes probabilistes, c'est-à-dire via une distribution de probabilité dans un univers de modèles possibles, on aura affaire à des *propriétés typiques* (ou *statistiques*).
Nous allons maintenant brièvement présenter le domaine scientifique de chacune de nos actions de recherche.

3.1.1. Algorithmique des mots

L'algorithmique des mots (ou l'algorithmique des séquences) est un domaine qui a vu un progrès considérable ces dernières années, comme le témoigne la parution récente de quelques monographies sur ce sujet [26][43][32][25]. Tout en étant une partie intégrante de l'algorithmique discrète en général, l'algorithmique des mots forme aujourd'hui un domaine de recherche en soi, de même que l'algorithmique des graphes par exemple. Les progrès dans ce domaine ont été largement alimentés par ses nombreux champs d'application, dont deux - la bioinformatique et la recherche d'informations sur l'Internet - sont particulièrement d'actualité aujourd'hui.

Les algorithmes sur les mots ont également un grand intérêt du point de vue théorique. À la base de cette théorie se trouvent quelques algorithmes et structures de données qui font partie du « trésor de l'algorithmique ». Le plus connu est probablement l'algorithme de Knuth-Morris-Pratt que l'on trouve dans tous les manuels d'enseignement d'algorithmique, mais qui, par ailleurs, a eu beaucoup d'applications à des problèmes divers (dont on continue à découvrir des exemples) et qui a fait l'objet d'une analyse mathématique intéressante et non-triviale [33]. D'autres algorithmes textuels jouent également un rôle fondamental, comme l'algorithme de recherche par programmation dynamique d'une plus longue sous-séquence commune à deux séquences, dont les applications sont diverses et variées (comme l'utilitaire `diff` d'UNIX par exemple ou encore l'algorithme bien connu de Smith et Waterman d'*alignement local* de séquences biologiques) (voir la section 4.1.3). Parmi d'autres algorithmes, probablement moins connus mais tout aussi élégants, notons l'algorithme de recherche en temps linéaire de carrés dans un mot [24] ou celui de recherche de palindromes, également en temps linéaire [38].

En outre, l'algorithmique des mots a développé des structures de données très puissantes, telles que l'arbre des suffixes (*suffix tree*) ou le DAWG (*Directed Acyclic Word Graph*). Le premier objectif de ces structures est de servir d'outils d'*indexation* de textes, c'est-à-dire de fournir une représentation spéciale de textes permettant d'exécuter efficacement des requêtes diverses. De plus, la transformation d'un texte en cette représentation se fait aussi très efficacement, à savoir en ligne et en temps linéaire. Une fois cette représentation obtenue, de nombreuses tâches peuvent être accomplies très efficacement. Sans essayer de les énumérer, nous renvoyons au livre [32] dont une grande partie est consacrée aux diverses utilisations de l'arbre des suffixes.

Un aspect très important pour nous de l'algorithmique des mots est qu'elle s'appuie d'une façon essentielle sur les propriétés combinatoires des mots. De nombreux algorithmes utilisent dans leur fonctionnement, ou dans leur analyse, des théorèmes de la combinatoire des mots. C'est pourquoi la combinatoire des mots tient une place importante dans nos études.

En résumé, notre objectif consiste à mettre au point de nouveaux algorithmes efficaces d'analyse de mots, en nous appuyant sur des propriétés combinatoires des mots. L'application directe de ces algorithmes est l'analyse de séquences biologiques dont nous parlerons dans la section 4.1.

3.1.2. Géométrie discrète

Parmi les structures discrètes que nous étudions figurent les ensembles discrets du plan ou de l'espace. La géométrie discrète, qui étudie ces objets, est apparue dans les années 70. Elle a pour objectif de définir un cadre théorique pour transposer dans $\mathbb{Z}^n$ les bases de la géométrie euclidienne, les notions discrètes définies étant le plus proche possible des notions continues que nous connaissons (telles que distance, longueur, convexité, ...). Plusieurs façons d'aborder cette étude ont été développées [21] :

- le point de vue topologique s'intéressant par exemple à l'équivalent discret du théorème de Jordan (toute courbe fermée simple sépare le plan en deux domaines : l'intérieur et l'extérieur de la courbe),
- le point de vue morphologique qui étudie les transformations de formes,
- le point de vue arithmétique, introduit par Jean-Pierre Reveillès en 1989 [41], qui donne une définition en compréhension des droites et plans discrets.

C'est cette dernière approche que nous utilisons. Une droite discrète du plan est ainsi l'ensemble des points de coordonnées (x, y) de $\mathbb{Z}^2$ vérifiant une double inégalité de la forme $\mu \leq ax + by < \mu + \omega$, avec a, b, μ, ω entiers. Les propriétés des droites discrètes définies de la sorte sont en relation étroite avec les propriétés des nombres entiers et nous rapprochent ainsi de la combinatoire des mots (utilisation des mots de Sturm par exemple). Les plans discrets sont définis de manière analogue.

Ces définitions analytiques permettent de représenter de manière compacte des objets discrets, d'étudier des objets intrinsèquement discrets (pas uniquement des approximations d'objets continus), et de définir des objets discrets infinis.

De nombreux résultats fondés sur cette approche ont vu le jour ces dix dernières années :

- définition et étude de nouvelles classes d'objets discrets (droites 3D, hyperplans, cercles, sphères, simplexes, ...),
- reconnaissance analytique permettant, non seulement de dire si une suite de points est ou non un segment de droite, mais aussi de donner les coefficients des inéquations analytiques correspondantes,
- reconstruction analytique visant à passer d'une représentation en compréhension du discret à une représentation en compréhension du continu,
- transformations discrètes : applications quasi-affines, rotations, filtres,
- visualisation, utilisant des propriétés des objets discrets telles que l'épaisseur optimale des objets.

En ce qui nous concerne, nos travaux se rangent principalement parmi les trois premières thématiques de cette liste.

3.1.3. Aléa discret

L'aléa discret est le champ d'étude consacré aux propriétés typiques de structures combinatoires aléatoires. Il est maintenant classique en informatique de considérer à côté des analyses de pire cas, des analyses en moyenne pour des modèles de données aléatoires ou pour des algorithmes probabilistes. Ce type d'analyses, popularisées par Knuth dans *The Art of Computer Programming*, se développe activement pour traiter des modèles de plus en plus réalistes, et donc plus complexes, en s'appuyant très largement sur le progrès des techniques de combinatoire énumérative (en particulier analytique, mais aussi algébrique et bijective) et des méthodes de génération aléatoire.

Du point de vue des outils mathématiques nous faisons appel et développons au premier chef des méthodes d'énumération. Pour étudier le comportement d'un paramètre combinatoire (moyenne, variance, distribution), notre approche s'appuie sur le codage global de l'information recherchée par des séries génératrices, accessibles au travers de décompositions combinatoires et d'équations fonctionnelles associées. Cette approche,

initiée en analyse d'algorithmes par Knuth puis Flajolet, Odlyzko, Sedgewick et d'autres[28][29], permet de traiter de larges classes de problèmes qui correspondent à des classes d'équations et de séries génératrices bien comprises (telles que les classes rationnelles ou algébriques[42]). La problématique évolue ainsi de l'automatisation de l'analyse dans les cas les plus favorables au traitement d'instances particulièrement pointues. Entre ces deux extrêmes se situe par exemple l'analyse de modèles combinatoires qui conduisent à un type particulier d'équations (dites aux variables catalytiques), dont plusieurs instances sont résolues sans que le statut de la classe entière soit encore bien clair.

Une spécificité de notre approche est à chercher dans notre intérêt particulier pour l'aspect expérimental, au travers de la génération aléatoire. Il est peut-être utile de rappeler ici que la génération aléatoire peut être utilisée d'une manière très similaire à l'analyse en moyenne, et la complète notamment lorsqu'on atteint les limites des techniques actuelles d'analyses en moyenne. L'utilisation d'un générateur aléatoire ne fournit certes que des résultats expérimentaux, mais permet d'observer des paramètres autrement inaccessibles et souvent de formuler des conjectures qui dirigent l'analyse.

Les méthodes de génération aléatoire se rangent en première analyse dans deux catégories : d'un côté les méthodes de marches aléatoires markoviennes[1], et de l'autre les méthodes combinatoires. Les premières réalisent une marche aléatoire dans l'espace des configurations possibles jusqu'à avoir oublié tout de leur point de départ. Les secondes au contraire cherchent à construire directement un objet aléatoire, en tirant partie d'informations structurelles [30]. Ces deux domaines sont en pleine expansion, en particulier suite aux avancées spectaculaires réalisées sur la maîtrise des temps de convergence pour les approches markoviennes, et à l'utilisation d'algorithmes probabilistes dans les approches combinatoires. C'est dans cette seconde tendance que se situent nos travaux.

4. Domaines d'application

4.1. Bioinformatique

Mots clés : *biologie*, *bioinformatique*, *séquence d'ADN*, *séquence de protéine*, *gène*, *promoteur*, *alignement de séquences*.

4.1.1. Introduction

Les modèles discrets apparaissent dans tous les domaines d'applications, mais il en est un qui joue pour nous un rôle tout à fait particulier. Il nous sert d'une part de source de problèmes et d'autre part de domaine privilégié pour tester et appliquer nos idées et méthodes. Il s'agit de la biologie moléculaire, c'est-à-dire de l'étude de macromolécules biologiques (ADN, ARN, protéines). L'irruption de modèles discrets dans ce domaine est due à la découverte de la structure de ces molécules, laquelle s'est avérée être un enchaînement linéaire d'éléments constituants qui appartiennent à un petit nombre de types. Ceci justifie immédiatement l'adoption d'un modèle discret de ces molécules : dans leur forme linéaire, ces molécules sont représentées par des chaînes de lettres tirées d'un alphabet de petite taille. Bien que ce modèle linéaire ne capture pas, du moins d'une façon adéquate, toutes les propriétés des molécules biologiques, il en capture une grande partie, et nos études portent en général sur des propriétés biologiques reflétées au niveau des séquences. Autrement dit, nous nous intéressons aux « empreintes » de phénomènes biologiques dans les séquences nucléiques ou protéiques. Ces « empreintes » sont décrites à l'aide de motifs, et une de nos motivations consiste à faire profiter la bioinformatique des techniques d'analyse et de recherche de motifs développées en algorithmique et en analyse probabiliste.

Ci-dessous sont présentées quatre actions de recherche en bioinformatique que nous poursuivons actuellement dont trois en collaboration avec des biologistes. Toutes ces actions ont une caractéristique commune : elles font intervenir une notion de *significativité* ou de *pertinence* d'un événement observé. Cette notion apparaît dans la situation suivante, très générale en bioinformatique.

[1] cf. http://dbwilson.com/exact/

Au cours du traitement informatique, des événements sont détectés (par exemple, des événements au niveau des séquences, tels que similarités ou répétitions), dont la véritable valeur biologique ne peut être déterminée automatiquement sans intervention des biologistes. Cependant au vu de la quantité de données à traiter, il est nécessaire d'avoir un critère pour éliminer *a priori* un maximum d'événements. Pour cela on s'intéresse à la vraisemblance de l'événement sous l'hypothèse où cet événement n'aurait aucune cause biologique, mais serait uniquement dû au hasard. Cette hypothèse est appelée l'*hypothèse nulle* et largement utilisée en biologie pour effectuer un prétraitement des données. L'idée sous-jacente de cette approche est que si l'occurrence d'un événement peut s'expliquer par le hasard, il est peu probable qu'il reflète un mécanisme biologique. *A contrario*, seuls des événements « surprenants » peuvent être biologiquement significatifs. Un exemple devenu classique[39] est l'abondance « anormale » dans le génome de *E.coli* du motif `gctggtgg` dont la fonction biologique a été mise en évidence.

Pour chaque type d'expérience, la mise en pratique de cette approche nécessite un modèle probabiliste du phénomène étudié et le calcul des probabilités correspondantes. L'exemple le plus connu est sans doute le modèle de Karlin utilisé par BLAST, logiciel de loin le plus utilisé en bioinformatique, qui recherche des similarités locales entre une protéine dite *query* et les séquences d'une base de données (voir la section 4.1.3). Cependant dans de nombreux cas, il n'y a pas encore de modèle mathématique satisfaisant et on se contente d'utiliser des mesures empiriques de la qualité des résultats trouvés qui, à la différence des arguments statistiques, posent de gros problèmes de normalisation. De tels scores seront typiquement capables de classer les résultats d'une expérience donnée du plus au moins intéressant, mais ne permettent pas de comparer des expériences différentes, comme le permet par exemple la fonction *expect* de BLAST. Par conséquent, il est important de définir un modèle probabiliste et de pouvoir calculer la notion de pertinence sous-jacente.

Du point de vue méthodologique, de nombreux travaux font appel à des modélisations probabilistes asymptotiques (*e.g.* le modèle de Karlin). Cependant, les conditions pratiques d'application sont très fréquemment largement en dessous des régimes asymptotiques et les effets de taille finie doivent être pris en compte. Au contraire, les modélisations issues des méthodes de l'aléa discret (voir la section 6.3) travaillent directement dans le régime fini, ce qui donne des modèles plus adaptés. C'est ce qui rattache ces questions aux problématiques fondamentales de notre avant-projet.

Un autre lien entre nos actions bioinformatiques et nos recherches fondamentales passe naturellement par l'algorithmique des mots (voir la section 6.1). Cet aspect est présent dans toutes nos actions, et surtout dans l'action 4.1.5 qui est directement issue de nos travaux théoriques en algorithmique.

4.1.2. Analyse de promoteurs dans un génome bactérien

Dans le cadre du Thème « Bioinformatique et Applications à la Génomique » du PRST Intelligence Logicielle, nous travaillons sur l'identification et la classification des promoteurs, chez la bactérie *Streptomyces cœlicolor*, en collaboration avec des chercheurs du Laboratoire de Génétique et de Microbiologie de l'Université Henri Poincaré de Nancy (Pierre Leblond, Bertrand Aigle). L'objectif de ce travail est l'identification des sites de fixation dans les zones promotrices du génome de la bactérie *Streptomyces cœlicolor*. Notons que cette bactérie a un intérêt économique certain, en particulier du fait que près de 70% des antibiotiques utilisés dans l'industrie pharmaceutique contiennent une substance produite par des bactéries de ce type.

Le problème consiste à identifier, dans les régions en amont des parties codantes du génome, les sites de fixation des facteurs σ. Ces derniers sont des protéines qui font partie de l'ARN polymérase - complexe moléculaire qui assure la transcription de gènes, première étape vers la production de protéines. Les facteurs σ sont responsables de la fixation de l'ARN polymérase sur la molécule d'ADN, ce qui initialise le processus de transcription et détermine, en particulier, le début de la séquence transcrite, appelé la position $+1$ ou le *TSS (Transcription Start Site)*.

Il est connu que dans les bactéries (organismes procaryotes), les endroits de séquences reconnus par un facteur σ sont localisés aux positions -10 et -35 par rapport au *TSS*. Cela nous conduit à rechercher des motifs composés de deux *boîtes*, séparées par une distance de 25 nucléotides environs. Plus précisément, chez la bactérie *Streptomyces cœlicolor*, il a été prédit 7848 régions codantes et 65 facteurs σ différents. Or, puisqu'il est admis qu'un facteur ne peut se fixer que sur un seul type de motif (son site de fixation lui est spécifique),

nous nous attendons à identifier 65 sites de fixation. Les régions codantes du génome pourront alors être classées selon la nature du promoteur (site de fixation à un facteur σ) présent en amont de chacune d'elles.

L'an passé, le travail a consisté à observer le phénomène dans 150 séquences étudiées biologiquement pour lesquelles les *TSS* sont connus ainsi que quelques motifs de sites de fixation. Pour élaborer une stratégie de recherche des motifs des sites de fixation, nous étions partis de l'idée qu'un motif de site de fixation devait être « surprenant » par rapport à son environnement, c'est-à-dire tel que le nombre d'apparitions dans les séquences promotrices soit anormalement grand par rapport à celui prévu par « le hasard » (voir la section 4.1.1). Pour déterminer le « taux de surprise » de motifs, nous avons utilisé le logiciel R'MES[2] qui identifie les motifs dont la fréquence d'apparition dans une séquence d'ADN est inattendue. Ce logiciel utilise les modèles de chaînes de Markov d'ordre m et calcule le score d'un motif en fonction de sa fréquence observée par rapport à sa fréquence attendue. L'exploitation des résultats nous avait permis de révéler certains des motifs de sites de fixation donnés dans des publications. Cette approche s'était donc révélée positive pour l'étude des 150 séquences. Ensuite, nous avions testé cette méthode sur le génome entier de *Streptomyces cœlicolor*. Ne connaissant pas les positions des *TSS* et par conséquent les positions -35 et -10, nous ne pouvions pas déterminer si les motifs candidats, trouvés comme « surprenant » par R'MES, se situent dans les régions -35 ou -10 (contrôle utile pour confirmer les motifs candidats). Les premiers résultats obtenus étaient donc très bruités. D'autres critères biologiques devaient donc être considérés, l'analyse de significativité seule n'étant probablement pas suffisante pour discriminer les sites.

Cette année, nous avons choisi de mieux caractériser les promoteurs connus de *S. cœlicolor*, dans le but de disposer de nouveaux critères, pour valider au mieux les motifs candidats. Nous avons à disposition six promoteurs publiés, pour lesquels les travaux, de nature biologique, nous donnent le droit de prendre ces séquences promotrices comme échantillons modèles. Nous avons étudié le niveau d'apparition des promoteurs connus dans les parties codantes. Nous avons développé des programmes d'extraction de séquences, afin de séparer les séquences nucléotidiques propres aux régions codantes et celles propres aux régions en amont de celles-ci. Ce second lot de séquences contient les promoteurs.

Dans un premier temps, nous avons étudié la distribution des occurrences d'un promoteur particulier (celui spécifique au facteur σR), dans les régions codantes connues pour avoir ce promoteur en leur amont et dans les régions promotrices associées. Nous avons émis l'hypothèse qu'un promoteur est prépondérant dans les régions promotrices associées et que ce motif est le plus fréquent de tous les motifs de même taille trouvés dans les séquences correspondantes. En fait, cette étude n'a pu être menée qu'avec le promoteur spécifique au facteur σR, car, le nombre de gènes connus pour être régulés par ce dernier est suffisamment grand. Les résultats obtenus nous ont permis de confirmer notre hypothèse. Mais ceci n'est pas suffisant pour fonder un critère de validation et ceci pour deux raisons. La première est que cette approche n'a été entreprise que pour un promoteur. La seconde est que nous ne disposons pas d'ensemble de gènes connus pour être régulés par un même facteur.

Une autre approche utilisée pour caractériser un promoteur, a été d'analyser les six promoteurs. Pour chacun d'eux, nous avons recherché, en faisant appel au logiciel `grappe` (section 5.1), le nombre d'occurrences dans chacun des deux lots et dans le génome total. Au départ, nous pensions que les occurrences d'un promoteur étaient distribuées préférentiellement dans les régions promotrices, puisque c'est dans ces zones que cette séquence a un rôle biologique. En fait, le traitement des résultats montre que les six promoteurs étudiés ne présentent pas la même distribution sur les deux lots. Des motifs promoteurs apparaissent significativement dans les régions promotrices, mais d'autres sont largement plus présents dans les régions codantes. Ici, l'absence de règle dans la répartition des motifs promoteurs, dans le génome, ne nous permet pas de fonder un critère de validation des futurs motifs candidats.

L'ensemble des résultats obtenus jusqu'à maintenant se trouve dans les rapports [17][16]. Toujours dans l'optique de trouver des critères de validation, nous allons à l'avenir, regarder si au voisinage des motifs ayant le rôle de promoteur, nous décelons un profil particulier comme par exemple, la présence d'autres mots. Nous

[2] http://www-mig.jouy.inra.fr/ssb/rmes/

souhaitons également analyser pour chacun des six promoteurs, la distance entre les motifs à rôle promoteur, et le début des régions codantes respectives.

4.1.3. Recherche de régions de similitude dans les séquences d'ADN

Les génomes de nombreuses espèces sont désormais disponibles (ou en cours de séquençage), et la comparaison de segments d'ADN ou de chromosomes complets est une des méthodes les plus fréquemment employées.

Cette comparaison est menée à bien, soit pour l'étude des homologies à des fins phylogénétiques, soit pour éventuellement localiser, sur un génome fraîchement séquencé, des loci susceptibles de jouer le même rôle que ceux d'espèces déjà plus largement étudiées. Elle peut avoir également pour but de trouver des éléments mobiles dans un génome, ou d'identifier des régions polymorphes en analysant les génomes de différents représentants d'une même espèce, ou dans bien d'autres situations.

De nombreux logiciels permettent la comparaison séquence à séquence (éventuellement contenues dans une base de données), afin de trouver des régions de similitude significatives, en ce sens qu'elles ont peu de chance de se produire par hasard (voir la section 4.1.1). Il s'agit alors de trouver des alignements dits locaux dont les scores permettent de les distinguer nettement des alignements parasites dus au hasard.

L'algorithme classique pour cette tâche est celui de Smith et Waterman, qui présente l'avantage d'être exhaustif et l'inconvénient d'être gourmand en temps de calcul. En pratique, les algorithmes du type BLAST ou FASTA sont très largement utilisés. Ces programmes sont basés sur des heuristiques leur permettant d'éviter de considérer tout l'espace de recherche. Le principe général de ces heuristiques est de rechercher des sous-séquences répétées de manière exacte pour déterminer des zones susceptibles d'être des copies approchées d'un même fragment.

La complexité, bien que restant toujours en $\mathcal{O}(n^2)$ est en pratique divisée par un fort facteur, ce qui rend ces algorithmes utilisables sur des séquences de l'échelle des chromosomes.

Notre objectif est ici, d'améliorer la sensibilité de tels algorithmes en s'intéressant d'une part aux propriétés statistiques des événements qui transforment les séquences (indels, mutations ponctuelles), et d'autre part à l'information conservée permettant de retrouver ces répétitions.

4.1.4. Calcul de score pour l'alignement de séquences protéiques

Le problème de l'alignement des séquences de protéines et de la recherche de motif dans ces séquences est un domaine de compétence reconnu de l'équipe de bioinformatique de l'IGBMC à Strasbourg, avec laquelle nous développons une collaboration dans le cadre du génopôle Strasbourg-Alsace-Lorraine.

D'un point de vue informatique, les problèmes de recherche de motifs et d'alignement de séquences de protéines semblent identiques aux problèmes correspondants pour les séquences nucléotidiques : seule change *a priori* la taille de l'alphabet. Pourtant au contact des biologistes on découvre qu'il n'en est rien et que les problématiques sont très différentes : les taux de similarité entre objets à comparer sont plus faibles et entrent en jeu les propriétés chimiques des acides aminés. Tout ceci conduit à des problèmes d'optimisation à critères multiples qui rendent les définitions de scores parfois assez arbitraires.

Dans ce cadre, nous nous concentrons sur des problèmes de score directement rencontrés dans les logiciels développés par l'équipe de l'IGBMC, en particulier sur l'amélioration du score utilisé par le logiciel `Ballast`[40]. Il s'agit de passer pour ces applications précises d'une approche de type score, à une approche plus fondée statistiquement (voir la section 4.1.1).

Ce logiciel traite un problème d'alignement local de séquences de protéines. Il retraite la sortie du logiciel standard BLAST qui recherche dans une base de données les séquences présentant une similarité avec une séquence *query*. Alors que BLAST utilise uniquement les qualités propres de chaque similarité pour calculer son score, `Ballast` se concentre sur des segments privilégiés (zones de la *query* rencontrant de nombreuses similarités dans la base) et néglige les similarités isolées, jugées peu importantes. Le score de `Ballast` est ainsi plus significatif que celui de BLAST mais, en l'absence de traitement statistique, n'est pas normalisé.

Nous collaborons donc à une nouvelle version de ce logiciel, au sein de laquelle la méthode de score est modifiée de façon à permettre une évaluation statistique approchée, sous forme de *p-values* associées aux scores. Ceci nous a conduit, avec Frédéric Plewniak et Olivier Poch de l'IGBMC, à redéfinir plusieurs étapes

du traitement et en particulier à traiter algorithmiquement et statistiquement le problème de la fragmentation des segments privilégiés et de leur redondance. Le nouvel algorithme ainsi défini est implanté et en cours de test à l'IGBMC. Un article décrivant les résultats obtenus est en préparation.

4.1.5. Les hot zones de recombinaison dans le génome humain

Certaines modifications du génome ont pour cause le réarrangement de séquences au cours de la recombinaison génétique. Des délétions et duplications peuvent résulter de recombinaisons « illégitimes » entre séquences très similaires mais non homologues. Ce mécanisme aboutit alors à l'élimination d'une région sur l'un des chromosomes et à la duplication sur l'autre chromosome, comme c'est le cas dans la maladie de Charcot-Marie-Tooth qui est causé par une duplication chromosomique près du gène de la protéine PMP-22 (protéine de myéline périphérique). D'autres remaniements peuvent être la conséquence de recombinaisons homologues qui sont fréquentes dans les régions de répétitions en tandem, compte tenu de la forte homologie de séquences entre les répétitions.

L'identification précise des *hot spots* de recombinaison est importante, tout d'abord pour la compréhension des mécanismes de recombinaisons mais aussi pour pouvoir identifier les zones qui seraient préférentiellement concernées par de tels remaniements. C'est en collaboration avec M.-D. Devignes (équipe Langue et Dialogue) que nous nous sommes intéressés à l'analyse des *hot zones* sur le chromosome 22 humain. Ces travaux s'appuient sur la délimitation des *hot zones* élaborées par l'équipe Génétique Moléculaire et Biologie du Développement à Villejuif [22], fondée sur les cartes génétiques et d'hybride d'irradiation.

A l'aide du logiciel **mreps** (voir la section 5.2), nous avons entrepris une analyse des *hot zones* et des zones stables afin d'établir un profil de répétitions en tandem qui pourrait caractériser ces régions chromosomiques. Ce profil exprime la distribution et les caractéristiques des répétitions en tandem présent dans la région correspondante. L'objectif de cette approche est de caractériser une région du génome par l'établissement d'une corrélation entre la présence de répétitions en tandem d'un certain type et le rôle biologique de la zone chromosomique dans laquelle elles se situent.

Cette approche pourra s'appliquer par la suite à l'étude de la pathogénicité bactérienne. Les répétitions sont clairement impliquées dans la virulence bactérienne comme cela a été décrit pour un grand nombre de bactéries. Jusqu'à présent les analyses bioinformatiques ont révélé des répétitions sur-représentées dans des gènes responsables de virulence. Là aussi, notre démarche consiste à établir un profil de répétitions en tandem, qui serait une empreinte d'un organisme en particulier mais aussi un profil par gènes responsables de la pathogénicité.

5. Logiciels

5.1. grappe

Mots clés : *analyse de texte*, *séquences d'ADN*, *recherche de motifs*, *motif multiple*, *motif avec espace*.

`grappe` est un logiciel qui recherche dans un texte plusieurs motifs simultanément, chacun étant composé d'une suite de fragments (mots) séparés par des espaces de longueur *a priori* non-bornée. Ce logiciel, déposé à l'APP (Agence de Protection de Programmes) en 2000, est diffusé par plusieurs voies :

- à partir de la page des logiciels développés à l'INRIA http://www.inria.fr/valorisation/logiciels/index.fr.html, ainsi que sur le CD ROM des logiciels libres de l'INRIA,
- à l'adresse http://www.loria.fr/~kucherov/software/grappe/,
- à partir de l'année 2001, depuis la page de la plateforme *Qualité et Sûreté des Logiciels* http://qsl.loria.fr/ dont **grappe** fait partie.

Notons qu'il existe une version spécialisée de **grappe** pour le traitement de séquences d'ADN/ARN et que nous utilisons **grappe** dans le travail sur l'analyse de promoteurs, décrit dans la section 4.1.2.

5.2. mreps

Mots clés : *séquence d'ADN*, *recherche de répétitions*, *répétition maximale*, *répétition en tandem*.

mreps est un logiciel de recherche de répétitions dites maximales dans les séquences d'ADN. Les répétitions maximales sont des répétitions successives, appelées parfois *périodicités* dans la littérature informatique et *répétitions en tandem* dans la littérature génomique. La naissance de **mreps**, il y a trois ans maintenant, a suivi les travaux théoriques [35] dans lesquels nous avons proposé un algorithme très efficace (en temps linéaire) pour rechercher toutes les répétitions maximales exactes dans un texte.

Depuis, nous poursuivons le développement de ce logiciel à la fois sur le plan théorique et appliqué. La version de **mreps** diffusé au début de l'année 2002 était la version 2.1. Elle a été présentée, sous forme de poster, à la conférence RECOMB'2002 [5]. Elle implantait l'algorithme de recherche de répétitions approchées, proposée dans [34]. Cette année, des améliorations considérables ont été apportées à cette version. L'objectif général de ces améliorations a été d'augmenter la souplesse du logiciel, afin d'identifier des répétitions plus « floues » mais toujours biologiquement pertinentes.

Premièrement, la notion d'erreur, qui permet de rechercher des répétitions en tandem avec des mismatch a été revue et modifiée. le logiciel n'autorise plus un nombre donné d'erreur par motif d'une même répétition, il fonctionne désormais avec un taux d'erreur (ou flexibilité) qui joue sur le degré maximal du « flou » des répétitions trouvées.

Parmi d'autres modifications, un paramètre de score a été introduit, dont l'objectif est double : d'une part il informe l'utilisateur sur la qualité d'une répétition trouvée et d'autre part, il est utilisé dans l'algorithme pour écarter les répétitions avec un score statistiquement attendu, de sorte que celles sorties par le logiciel soient « significatives » (voir la section 4.1). Cela marque un changement dans la philosophie de l'approche, à savoir un passage d'une approche purement combinatoire vers une approche mixte. Cette dernière est basée sur une recherche exhaustive de tous les « éléments de base » (répétitions calculées par la version 2.1 de **mreps**) suivie par un traitement statistique de ces éléments afin de former des répétitions plus floues et biologiquement pertinentes.

Une autre modification consiste à rechercher des répétitions significatives dont l'exposant est inférieur à deux, ce qui n'était pas possible avec la version 2.1. Cette propriété est intéressante, elle permet de rechercher des fragments répétés séparés par une distance bornée.

A ce jour, la version 2.1 de **mreps** est diffusée sous la licence GPL par plusieurs voies : depuis le serveur Web du Loria : http://www.loria.fr/mreps/ et depuis la page des logiciels libres de l'INRIA[3], ainsi que depuis le serveur *Collaborative Computational Project 11*[4] domicilié au *UK Human Genome Mapping Project Resource Centre*[5]. `mreps` a également été déposé à l'APP. `mreps` est interrogeable via une interface Web depuis sa page de distribution ; il est également installé sur le serveur BIOWEB de l'Institut Pasteur[6], serveur qui fournit une interface Web à plusieurs outils bioinformatiques existants. La dernière version stable de **mreps** est la 2.4.3 et la version 2.5 est en cours de finition. Cette dernière devrait être mise en diffusion avant la fin de l'année 2002.

5.3. YASS

Mots clés : *séquence d'ADN*, *répétition distante*, *répétition approchée*, *région de similitude*, *alignement local*, *comparaison de séquences*.

Le logiciel YASS *(Yet Another Similarity Searcher)* a été mis au point afin de rechercher les régions de similitude entre séquences génomiques (voir la section 4.1.3). Une première version devrait être disponible courant décembre. Il se veut plus sensible que BLAST, et cela grâce à la recherche et au chaînage de mots plus petits qu'il regroupe à l'aide de deux critères statistiques issus du taux de mutations (substitutions et *indels*) de la séquence. Il travaille actuellement sur des données au format FASTA et indique la liste des similitudes trouvées classées par significativité, donne leur *e-value*, et éventuellement l'alignement observé.

[3] http://www.inria.fr/valorisation/logiciels/index.fr.html
[4] http://www.hgmp.mrc.ac.uk/CCP11/index.jsp
[5] http://www.hgmp.mrc.ac.uk/
[6] http://bioweb.pasteur.fr/seqanal/interfaces/mreps.html

Le programme a été développé en C ANSI sous Unix, a été testé sous Linux et Windows sur des chromosomes de levure (*Saccharomyces cerevisiae*). Les résultats ont été comparés à ceux d'autres logiciels tels que BLAST-NCBI, BLAT, et REPuter : le logiciel produit des alignements de qualité supérieure à REPuter sans donner de résultats redondants. Ses résultats sont plus complets que BLAT ; il parvient même à trouver certaines répétitions significatives que BLAST ne peut distinguer. L'algorithme de YASS sera détaillé dans la section 6.1.3.

6. Résultats nouveaux

6.1. Algorithmique et combinatoire des mots

6.1.1. Combinatoire des répétitions

En collaboration avec Pascal Ochem (actuellement doctorant au LaBRI) et Michaël Rao (actuellement doctorant au LITA à Metz), nous nous sommes attaqués à un sujet classique de la combinatoire des mots, à savoir les propriétés des mots infinis vérifiant des contraintes sur les répétitions qu'ils contiennent. Notons que nous avons déjà fait des travaux dans ce domaine car nous avons obtenu, en 1997-1998 en collaboration avec des collègues russes, une série de résultats sur la *fréquence minimum* d'une lettre dans les mots infinis binaires ne contenant pas de périodicités d'un exposant donné supérieur à deux [36].

Trivialement, tout mot binaire d'au moins quatre lettres contient un carré (autrement dit, un facteur de forme uu, ou encore une périodicité d'exposant 2). A. Fraenkel et J. Simpson [31] ont montré que l'on peut construire un mot infini qui ne contient que trois carrés distincts (par exemple, 00, 11 et 0101), et que l'on ne peut pas en avoir moins. Leur construction laisse cependant ouverte la question complémentaire du *nombre minimal d'occurrences de carrés* dans les mots binaires, que nous avons étudiée.

Tout d'abord, nous avons démontré que le nombre minimal d'occurrences de carrés dans les mots binaires de taille n tend vers une part constante de n. Nous avons ensuite cherché à estimer cette constante, que l'on peut appeler le *taux minimum des carrés* et nous avons établi qu'elle vaut $0.55080...$. Notons que le taux des carrés de la construction de A. Fraenkel et J. Simpson est au-dessus de $40/69 = 0.5797..$, ce qui montre qu'elle n'est pas optimale pour ce critère.

Les encadrements ont été obtenus à l'aide d'ordinateur. La borne supérieure a été trouvée via un motif particulier de longueur 187 trouvé par un calcul sur ordinateur. Pour l'obtention de la borne inférieure une méthode a été élaborée, basée sur les propriétés des mots réalisant le taux minimal des carrés. Un article [12] décrivant ces résultats est en cours de soumission.

6.1.2. Algorithmique des répétitions

Dans le prolongement de nos travaux sur la combinatoire et l'algorithmique des répétitions, nous avons résolu cette année le problème de calcul efficace des périodes locales d'un mot. Pour chaque position du mot, la période locale est définie comme l'entier minimum k tel qu'il existe un carré de longueur $2k$ centré à cette position. La période locale est une notion très importante dans la combinatoire du mot, elle est à la base du *Critical Factorisation Theorem* et d'autres résultats clefs de la théorie. Le calcul efficace des périodes locales dans un mot donné restait un problème ouvert. La difficulté résidait dans le fait que ce calcul semblait ne pas découler des algorithmes connus, efficaces et puissants, pour calculer toutes les périodicités du mot[35]. Le résultat le plus proche était celui de R. Kosaraju[37] qui a décrit un algorithme en $O(n)$ permettant de calculer, pour chaque position du mot, le plus petit carré commençant à cette position. Malgré la proximité apparente du problème, cet algorithme ne semble pas s'appliquer au calcul des périodes locales.

En collaboration avec R. Kolpakov, ainsi qu'avec T. Lecroq et A. Lefebvre (Université de Rouen), nous avons proposé un algorithme linéaire pour le calcul des périodes locales. Tout en se basant sur les techniques que nous avons employées pour le calcul des périodicités[35] (factorisation de Lempel-Ziv, fonctions de Main et Lorentz), cet algorithme n'en est pas une simple conséquence, mais plutôt une nouvelle illustration de la puissance de ces techniques. En effet, il a fallu un argument mathématique très subtil pour démontrer comment

ces techniques peuvent s'appliquer au problème en question. Un article décrivant ces résultats est en cours de préparation.

6.1.3. Recherche de répétitions approchées

Avec le stage de DEA de L. Noé [13], nous avons démarré un nouveau travail sur un sujet central de la bioinformatique : recherche de répétitions approchées, autrement dit de régions de similitude, dans les séquences nucléotidiques. Dans ce cadre, le logiciel YASS a été développé (section 5.3), basé sur un *algorithme de chaînage* qui regroupe de courtes séquences répétées de manière exacte (appelées graines). Ces regroupements sont construits en utilisant des critères statistiques basés sur un modéle Bernoulli de la séquence. Certains de ces critères sont issus de ceux utilisés par *Tandem Repeat Finder*[20] et modifiés pour être appliqués à la recherche de répétitions distantes [6]. Un modèle de marche aléatoire sert à simuler les *indels* (insertions et suppressions de nucléotides) survenus entre les graines, alors qu'un modèle de *coin tossing*, qui se traduit par une série géométrique d'ordre k, permet d'évaluer la distance maximale tolérée entre les graines trouvées.

Contrairement à FASTA, YASS trouve à la volée des zones susceptibles de contenir une répétition approchée sans avoir à effectuer systématiquement le compte du nombre de *hits* (graines trouvées), et cela grâce à une méthode algorithmique qui regroupe les différentes graines lorsque ces dernières sont susceptibles d'appartenir à la même répétition distante.

Le critère de sélection des *hits* n'est pas fixe, comme celui de BLAST (deux *hits* de taille minimale sur la même diagonale [18]). Il est plus souple, et autorise les ouvertures d'*indels* plus ou moins larges entre les graines. De cette manière, il capture toute l'étendue de la zone de similitude, plutôt que de trouver une petite région répétée pour ensuite l'étendre en une zone éventuellement bien plus grande.

La sélectivité de l'algorithme reste bonne grâce à un critère d'évaluation rapide des groupes de graines (chaînages) trouvés, et la sensibilité est améliorée. Ces deux caractéristiques peuvent être également modulées selon les paramètres spécifiés par l'utilisateur. L'information contenue dans les chaînages sélectionnés est ensuite traitée par un algorithme d'alignement de séquences afin de confirmer éventuellement la présence d'une zone de similarité et d'évaluer alors sa significativité. Des informations supplémentaires comme le biais positionnel des mutations sur les codons peuvent être également évaluées.

6.2. Géométrie discrète

6.2.1. Segments flous

La reconnaissance d'objets discrets est un important sujet en géométrie discrète et de nombreux travaux concernant les droites et les arcs discrets ont été réalisés (entre autres [23][44]). Nous nous intéressons à la notion d'objets discrets « flous », correspondant à des objets discrets bruités, et à leur détection. En effet, ce problème trouve une application directe dans le traitement d'images, en particulier quand il s'agit d'interpréter les formes géométriques présentes dans les images.

Nous avons défini la notion de segments flous, en relation avec la définition arithmétique des droites discrètes où l'épaisseur est paramétrable. Un segment flou est une suite 8-connexe de points qui appartiennent à une droite discrète dont l'épaisseur est donnée. Un paramètre, l'ordre du segment flou, permet de contrôler l'amplitude du bruit autorisé en fixant l'épaisseur de la droite englobant le segment flou.

Ajouter un point à un segment flou revient à calculer la pente et l'épaisseur d'une nouvelle droite englobante ce qui est réalisé avec des calculs très simples. Ceci nous a conduit à élaborer un algorithme incrémental et très efficace de découpage de courbes discrètes en segments flous d'ordres fixés.

Un article a été rédigé sur ce sujet et est en cours de soumission [10]. Ce travail a été l'occasion de contacts avec X. Hilaire de l'équipe QGAR, qui nous a fourni des exemples concrets issus de plans d'architectes.

6.2.2. Convexité discrète

L'étude de la convexité d'une région discrète du plan se ramenant à celle de figures particulières appelées polyominos hv-convexes, nous avons développé un algorithme incrémental et linéaire de détection de la convexité de tels polyominos, une version longue de l'article [27] sera publiée dans la revue *Discrete Applied Mathematics* en janvier 2003.

A la suite de ce travail, des contacts ont été établis avec l'université de Hambourg, plus particulièrement avec le professeur U. Eckhardt, et une collaboration sur le sujet de la convexité discrète en dimension 3 vient d'être mise en place.

6.3. Aléa discret

6.3.1. Combinatoire analytique et analyse d'algorithmes

Le travail entamé avec Cyril Banderier, Philippe Flajolet, Bruno Salvy et Michèle Soria sur l'analyse du phénomène d'Airy en combinatoire analytique s'est continué cette année. Il s'agit là de comprendre l'apparition récurrente, dans l'analyse en moyenne de différents algorithmes ou structures aléatoires discrètes, de certaines familles de lois limites non gaussiennes liées à la fonction spéciale d'Airy : alors que la loi des grands nombres conduit en général à attendre des fluctuations gaussiennes autour des valeurs moyennes, nous mettons en évidence le mécanisme mathématique sous-jacent à l'apparition de ces autres lois. Notre premier article en revue sur ce sujet est paru à la revue *Random Structures and Algorithms* [19]. Un autre article, sur l'énumération des graphes connexes a été soumis [11].

Remarquons enfin la parution à la revue *Probability Theory and Related Fields* de l'article écrit avec Mireille Bousquet-Mélou sur les chemins du plan coupé, qui repose aussi en partie sur des techniques de combinatoire analytique [1].

6.3.2. Combinatoire algébrique

Dans ce domaine, la collaboration avec Sylvie Corteel (CNRS, PRISM, Versailles), Alain Goupil (Université du Québec à Montréal) et Dominique Poulalhon (Laboratoire d'Informatique de l'X) continue. À l'occasion de l'invitation de G. Schaeffer à Montréal nous avons ainsi montré que la base de fonctions symétriques que nous avions introduite précédemment permet de reproduire à l'aide de polynômes le calcul du centre de l'algèbre du groupe symétrique. Nous cherchions depuis plusieurs années une telle représentation polynomiale des constantes de structures associées à ce centre. Un article est en préparation sur ce sujet avec Alain Goupil et Sylvie Corteel. Enfin un texte sur les produits de petites classes de conjugaison écrit avec Alain Goupil et Dominique Poulalhon est en cours de relecture avant soumission.

Notons encore que l'article *Factorisations of a n-cycle into m permutations* de Dominique Poulalhon et G. Schaeffer est paru dans la revue *Discrete Mathematics* [2].

6.3.3. Topologie des surfaces aléatoires

Les cartes planaires et triangulations aléatoires sont un modèle combinatoire classique sur lequel nous travaillons depuis plusieurs années.

Avec Philippe Chassaing (professeur à l'IECN), nous avons démontré une conjecture sur le diamètre des cartes aléatoires, formulée il y a déjà plusieurs années par G. Schaeffer. Nos résultats s'appuient sur un codage de ces objets par des arbres étiquetés et sur un passage à la limite continue inspiré des travaux du probabiliste D. Aldous. Nous avons rédigé et soumis un article traitant plus généralement des propriétés des distances dans les cartes aléatoires [9]. Ces travaux ouvrent la porte à une approche « continue » pour les cartes aléatoires. Nous avons ainsi travaillé, notamment avec Balint Virag du MIT sur la possibilité de définir une « Continuum Random Map » qui serait l'analogue pour les cartes du « Continuum Random Tree » d'Aldous. Un résumé de ces travaux est paru aux actes de la conférence internationale « Algorithmes, Arbres, Combinatoire et Probabilités » [3].

Le lien de ces travaux avec la physique quantique peut sembler surprenant, mais l'explication suivante, quoique très simplifiée, permet d'en appréhender l'origine. De même qu'en informatique on est naturellement amené par des contraintes matérielles à considérer des géométries discrètes (cf. la section 6.2), la recherche de modèles mathématiques adaptés à la physique débouche sur des modèles discrets. Ainsi, pour la physique statistique classique, la discrétisation de l'espace euclidien usuel à deux dimensions conduit naturellement à l'étude de modèles sur une grille régulière, tandis que, pour la physique quantique qui remplace l'univers fixe par une distribution de probabilité sur tous les univers possibles, la discrétisation conduit à une distribution de probabilité sur tous les univers discrets possibles, qui se trouve coïncider avec le modèle de surfaces aléatoires

étudié en combinatoire. La conférence *Discrete Random Geometry and Quantum Gravity* illustre bien ce courant de la physique quantique.

Alors que les aspects géométriques des surfaces aléatoires ont été largement étudiés aussi bien en combinatoire qu'en physique par des méthodes et avec des résultats complémentaires, les modèles sur cartes ont plus spécifiquement été considérés en physique. Avec Mireille Bousquet-Mélou, nous travaillons à comprendre comment les outils de l'approche combinatoire (décompositions des structures, séries génératrices et équations fonctionnelles) s'étendent et s'adaptent à ces problèmes. Nous avons obtenu un premier résultat frappant dans cette direction en traitant le modèle d'Ising sur cartes aléatoires par une méthode purement bijective. Un article a été rédigé sur ce sujet [8].

L'ensemble de ces travaux sur la topologie des surfaces aléatoires a fait l'objet d'exposés invités aux rencontres mathématiques de l'École Normale Supérieure de Lyon en mars dernier et à la session combinatoire du congrès d'été de la société mathématique du Canada à Québec. Par ailleurs G. Schaeffer donnera un mini-cours (4h) sur ce sujet à la session du printemps 2003 de l'Institut Henri Poincaré. Cette session réunit pour un trimestre une cinquantaine de chercheurs français et étrangers autour du thème « Geometry and statistics of random growth ».

6.3.4. Algorithmes de génération aléatoire

Les algorithmes de génération aléatoire restent au centre de nos intérêts. D'une part, nous travaillons au développement de nouveaux algorithmes dédiés à des familles de graphes particulièrement intéressantes. Ainsi avec Dominique Poulalhon (LIX), nous avions l'an dernier étendu le paradigme de génération aléatoire par conjugaison d'arbres développé par G. Schaeffer à la classe importante des triangulations avec bords. Le nouvel algorithme, de complexité linéaire, unifie ainsi le cas classique des triangulations de polygones et celui des triangulations de la sphère. Ces résultats ont fait l'objet d'une présentation à la conférence FPSAC'02 à Melbourne [7] et d'une version longue de l'article à paraître à la revue *Theoretical Computer Science* [14]. De plus nous avons encore étendu la méthode à une nouvelle classe de triangulations, les triangulations strictes. Ce résultat est particulièrement intéressant car il fait intervenir un nouvel ingrédient : les arbres de Schnyder, qui sont des outils classiques de l'algorithmique du dessin de graphes. Nous montrons qu'on peut utiliser ces arbres pour définir un codage bijectif des triangulations strictes. On en déduit un algorithme de génération aléatoire là encore linéaire. Un article sur ce sujet est en préparation. Toujours dans le même registre nous avons rédigé une note sur l'application de nos méthodes aux cartes biparties eulériennes [15].

Avec Paul Zinn-Justin du Laboratoire de Physique Théorique de l'université Paris-Sud, nous avons utilisé certains de ces algorithmes de génération aléatoire de cartes pour étudier les classes d'équivalences topologiques de courbes fermées dans le plan. Nous avons en particulier pu proposer une conjecture sur l'asymptotique du nombre de telles courbes et la tester numériquement. Un article sur ce sujet est en préparation.

D'autre part, nous nous intéressons à la compréhension des mécanismes généraux qui permettent le développement d'une algorithmique efficace de la génération aléatoire. Dans cette direction, nous travaillons plus particulièrement à l'utilisation d'algorithmes probabilistes combinatoires : ces algorithmes combinent les avantages de l'approche combinatoire (essentiellement la garantie de respecter parfaitement la distribution probabiliste visée) avec la simplicité d'implantation des méthodes probabilistes. L'originalité de notre approche est de ne pas relâcher la contrainte d'uniformité (contrairement aux chaînes de Markov dont la convergence vers la distribution uniforme ne peut être contrôlée que par des méthodes complexes de couplage par le passé), mais plutôt de travailler sur la taille des objets considérés, en autorisant de légères fluctuations de ce paramètre. Ces travaux, en collaboration avec Philippe Duchon, Philippe Flajolet et Guy Louchard ont fait l'objet d'un exposé à la conférence *ICALP 2002* [4]. Une version longue de l'article est en préparation.

6.3.5. Combinatoire des suites arithmétiques

8. Actions régionales, nationales et internationales

8.1. Actions régionales

Au niveau du Contrat de Plan Etat-Région 2000-2006, nous sommes impliqués dans le Pôle de Recherche Scientifique et Technologique (PRST) *Intelligence Logicielle*.

L'équipe ADAGE participe au

- Génopôle Strasbourg Alsace-Lorraine
- Thème « Bioinformatique et Applications à la Génomique » du PRST Intelligence Logicielle
- Thème « Qualité et sûreté des logiciels et systèmes informatiques » du PRST Intelligence Logicielle

En outre, nous avons des collaborations soutenues avec des mathématiciens de l'Institut Elie Cartan de Nancy : Ph. Chassaing sur le thème de l'aléa discret (voir la section 6.3) et P. Vallois et M.-P. Etienne sur le thème de la bioinformatique.

8.2. Actions nationales

L'équipe participe activement aux groupes *Aléa* et *Algorithmique des séquences* du GDR CNRS ALP. Nous avons aussi participé à la réponse à l'appel d'offre Math-STIC du CNRS au sein du groupe coordonné par Brigitte Chauvin et Brigitte Vallée, sélectionnée par le CNRS en novembre 2001.

ADAGE participe au Réseau Thématique Pluridisciplinaire *Bioinformatique : de la séquence génomique à la fonction biologique* du CNRS (RTP 41, responsable O. Gascuel, LIRMM). Dans ce cadre, nous participons à l'Action spécifique *Algorithmes et séquences* (AS 77) dont G. Kucherov est co-animateur avec T. Lecroq (Université de Rouen) et E. Rivals (LIRMM). La première réunion de ce groupe de travail sera organisé au Loria les 20-21 janvier 2003.

Nous participons au projet GÉNOGRID, mis en place en 2001 dans le cadre de l'Action Concertée Incitative *Globalisation des ressources informatiques et des données* (ACI GRID). Le projet est coordonné à l'IRISA par Dominique Lavenier, les autres laboratoires participants sont LAMIH, ABISS, LIH, LIFL.

8.3. Actions européennes

G. Schaeffer est membre extérieur du *Research Training Network : Algebraic Combinatorics in Europe*, soutenu par la communauté européenne.

8.4. Actions internationales

G. Schaeffer collabore avec Igor Pak et Balint Virag du MIT, et Alain Goupil de l'université du Québec à Montréal. Ces liens se sont concrétisés par des séjours dans ces universités et la préparation d'articles en commun.

Bien que notre projet dans le cadre de l'Institut Liapunov franco-russe d'Informatique et de Mathématiques appliquées soit arrivé à son terme en 2000, nous gardons des contacts étroits avec les chercheurs moscovites, ce qui s'est concrétisé cette année par le séjour de R. Kolpakov dans l'équipe et notre travail commun.

Plusieurs autres contacts internationaux se sont traduits par des séjours de chercheurs (voir la section 8.5) soit par des visites de chercheurs d'ADAGE de laboratoires étrangers (voir les sections 9.3.2,9.3).

8.5. Visites et invitations de chercheurs

Roman Kolpakov, chercheur de l'Université de Moscou et collaborateur de l'Institut Liapunov franco-russe d'Informatique et Mathématiques Appliquées, a travaillé au sein d'ADAGE pendant trois mois cette année, en tant que professeur invité de l'INRIA.

Juha Kärkkäinen, actuellement post-doctorant au MPI à Saarbrucken, a visité ADAGE au mois de mai ; il a présenté un exposé, *Better Filtering with Gapped q-Grams* au séminaire d'Informatique Fondamentale du Loria.

Alexandre Bolshoy, professeur à l'Université de Haifa, a passé un mois au sein d'ADAGE à l'automne 2002, pour travailler sur diverses questions bioinformatiques (complexité d'information de séquences génomiques, éléments transposables, analyse de promoteurs). Il a fait deux exposés au séminaire de bioinformatique du Loria. Une poursuite de collaboration est envisagée.

Claudia Acquisti, chercheur de l'Université de Florence, a visité l'équipe pendant trois jours en octobre ; elle a travaillé sur l'analyse de génomes bactériens à l'aide du logiciel **mreps**.

Gary Benson, chercheur de *Mount Sinai School of Medicine* à New-York, a passé deux jours au sein d'ADAGE pour mettre en place une collaboration dans le cadre de la plateforme d'analyse de répétitions dans les génomes, développée dans son groupe.

9. Diffusion des résultats

9.1. Animation de la Communauté scientifique

L'année passée, G. Kucherov a fait partie des comités de programme des conférences *JOBIM'2002* et des *Journées d'Arithmétiques Faibles (JAF'2002)*. Il fait actuellement partie du comité de programme de *Perspectives of System Informatics (PSI'2003)*.

G. Schaeffer a été responsable du séminaire d'informatique fondamentale du LORIA pour l'année 2001-2002.

I. Debled-Rennesson est membre de la commission de spécialistes de l'IUFM de Lorraine. De plus, elle est membre du comité technique IAPR sur la géométrie discrète (TC18).

J. Rouyer est vice-présidente de la commission de spécialistes en 27e section de l'UHP

G. Schaeffer est membre suppléant de la commission de spécialistes (section 27) de l'École Normale Supérieure à Paris.

9.2. Enseignement universitaire

I. Debled-Rennesson et G. Kucherov ont encadré le stage de DESS *Compétences Complémentaires en Informatique* de Sandra Luis pendant juin-octobre. De plus, ils encadrent le stage de Patricia Lavigne en DESS *EGOIST (Rouen)* qui se déroule en alternance de novembre 2001 à juillet 2003.

G. Kucherov, en commun avec D. Kratsch (Université de Metz), enseigne le module *Algorithmique des structures discrètes* du DEA d'Informatique à Nancy (filière *Algorithmique Numérique et Symbolique*). Il dispense également des cours en DESS *Ressources Génomiques et Traitements Informatiques* à l'Université Henri Poincaré de Nancy.

L. Noé, J. Rouyer et I. Debled-Rennesson ont participé à divers enseignements d'informatique à l'UHP (DESS CCI, ESIAL, Mias) et à l'IUFM de Lorraine.

J. Rouyer est responsable de la filière *Ingénierie du Logiciel* à l'ESIAL.

G. Schaeffer donne avec Mireille Bousquet-Mélou (LaBRI) un cours au DEA « Algorithmes » commun aux universités parisiennes, intitulé « Combinatoire énumérative et génération aléatoire ». Il a co-encadré, avec Guillaume Hanrot, le stage dans l'équipe Spaces de Pascal Ochem en DEA d'informatique à Nancy.

9.3. Participation à des colloques, séminaires, invitations

9.3.1. Colloques, tutoriels, conférences et séminaires invités

G. Bana et G. Kucherov ont participé à la réunion du groupe de travail CNRS *Algorithmes en Bioinformatique* qui s'est tenue les 11-12 mars à Montpellier. G. Kucherov y a fait un exposé. Ils ont également participé à la conférence RECOMB à Washington au mois d'avril, où ils ont présenté un poster.

G. Kucherov a fait un exposé invité aux *Journées sur les Arithmétiques Faibles* qui ont eu lieu à Saint-Petersburg en juin 2002. Il a également fait un exposé à la réunion du groupe de travail *Algorithmique des séquences* du GDR ALP (Rouen, juin 2002), ainsi qu'à la journée sur la plate-forme *Qualité et sureté de logiciels* organisée au Loria le 3 avril dans le cadre du PRST *Intélligence Logicielle*.

G. Kucherov a fait un exposé à l'Ecole d'été *From Genome to Life* à Cargèse au mois de juillet.

L. Noé a assisté au séminaire *Algorithmique et Biologie* sur les *Statistiques et Probabilité en Génomique*, qui s'est tenu à Lyon au mois d'octobre.

G. Bana, G. Kucherov et L. Noé ont participé à la première *European Conference on Computational Biology (ECCB)* à Saarbrucken au mois de novembre. G. Kucherov et L. Noé y ont présenté un poster.

G. Kucherov est invité pour faire des séminaires au *Max-Plank Institute for Molecular Genetics* à Berlin et à l'Université de Varsovie au mois de décembre.

G. Schaeffer a fait les exposés invités suivants :

- Aux *Rencontres Mathématiques de l'École Normale Supérieure de Lyon* en mars 2002. Ces rencontres faisaient intervenir 3 orateurs sur 2 jours, conjointement avec le séminaire Hypathie des universités de Lyon et Marseille.
- Au séminaire de combinatoire du MIT, à Cambridge, en avril 2002.
- À la session « Combinatoire » du *Congrès d'été de la Société Mathématique du Canada*, à Québec, en juin 2002.
- À une journée scientifique en l'honneur du Professeur Guy Louchard, à l'Université Libre de Bruxelles.

Il a également fait les exposés suivants :

- Au colloque international *8th Seminar on Analysis of Algorithms*, en juin 2002 à Strobl en Autriche.
- À la conférence internationale *Colloquium on Mathematics and Computer Science : Algorithms, Trees, Combinatorics and Probabilities*, en septembre 2002, à Versailles.
- Aux rencontres Aléa du GDR CNRS ALP en mars 2002 à Marseille.
- Aux rencontres du GDR CNRS Tresses en septembre 2002 à Lacanau.
- Aux journées *Arbres* de décembre 2002 à Bordeaux.

Enfin les résultats de G. Schaeffer et Dominique Poulalhon ont donné lieu à un exposé à la conférence *14th International Conference on Formal Power Series and Algebraic Combinatorics (FPSAC'02)* à Melbourne, Australie, en juillet 2002. Les résultats de G. Schaeffer, P. Duchon, P. Flajolet et G. Louchard ont donné lieu à un exposé à la conférence *29-th International Colloquium on Automata,Languages, and Programming (ICALP'2002)*, en juillet 2002 à Malaga en Espagne.

P. Lavigne et I. Debled-Rennesson ont assisté aux *Journées Ouvertes Biologie Informatique Mahématiques (JOBIM)* du 10 au 12 juin 2002 à St Malo.

J.-L. Rémy et I. Debled-Rennesson ont participé à la dixième conférence internationale *Discrete Geometry for Computer Imagery (DGCI)* à Bordeaux du 3 au 5 avril 2002.

I. Debled-Rennesson à participé au 11e workshop *Theoretical Foundations of Computer Vision* qui s'est déroulé pendant une semaine en avril au château de Dagstuhl en Allemagne.

I. Debled-Rennesson a été invitée à présenter ses travaux à la conférence *Mathematical Aspects of Visual Cognition* à Hambourg en novembre 2002.

9.3.2. Séjours de chercheurs

G. Schaeffer a été invité au MIT, une semaine en avril par Igor Pak, puis trois semaines en août par Balint Virag. Il a séjourné trois semaines en juin au LaCIM, à l'université du Québec à Montréal, à l'invitation d'Alain Goupil.

9.4. Jurys de thèses et jurys divers

G. Schaeffer fait partie du jury du prix de thèse annuel de l'association SPECIF, après avoir reçu ce prix pour l'année 1999, ainsi que du jury du prix de thèse annuel de l'AFIT.

10. Bibliographie

Articles et chapitres de livre

[1] M. BOUSQUET-MÉLOU, G. SCHAEFFER. *Walks in the slit plane.* in « Probability Theory and Related Fields », numéro 3, volume 124, mars, 2002, pages 305-344.

[2] D. POULALHON, G. SCHAEFFER. *Factorisations of large cycle in the symmetric group.* in « Discrete Mathematics », numéro 1-3, volume 254, juin, 2002, pages 433-458.

Communications à des congrès, colloques, etc.

[3] P. CHASSAING, G. SCHAEFFER. *Random Planar Lattices and Integrated SuperBrownian Excursion.* in « Colloquium on Mathematics and Computer Science : Algorithms, Trees, Combinatorics and Probabilities, Versailles, France », série Trends in Mathematics, Gardy, D. and Mokkadem, A., Birkhauser, éditeurs B. CHAUVIN, P. FLAJOLET, D. GARDY, A. MOKKADEM., pages 123-141, septembre, 2002.

[4] P. DUCHON, P. FLAJOLET, G. LOUCHARD, G. SCHAEFFER. *Random Sampling from Boltzmann principles.* in « 29th International Colloquium on Automata, Languages and Programming - ICALP'2002, Malaga, Spain », série Lecture Notes in Computer Science, volume 2380, Springer, pages 501-513, novembre, 2002.

[5] G. KUCHEROV, R. KOLPAKOV, G. BANA, M. GIRAUD, R. RABBAT. *`mreps` : a program for exhaustive search for tandem repeats in DNA sequences.* in « RECOMB 2002 , Annual International Conference on Computational Biology, Washington, DC, US », avril, 2002, poster.

[6] L. NOE, G. KUCHEROV. *A new method of finding similarity regions in DNA sequences.* in « European Conference on Computational Biology (ECCB 2002), Saarbrücken, Germany », série European Conference on Computational biology 2002. Poster Abstracts, pages 173-174, octobre, 2002.

[7] D. POULALHON, G. SCHAEFFER. *A bijection for loopless triangulations of a polygon with interior points.* in « International Conference on Formal Power Series and Algebraic Combinatorics - FPSAC'02, Melbourne, Australie », Foda, O. and Guttmann, T., Actes locaux de l'universite de Melbourne, éditeurs O. FODA, T. GUTTMANN., juillet, 2002.

Rapports de recherche et publications internes

[8] M. BOUSQUET-MÉLOU, G. SCHAEFFER. *The degree distribution in bipartite planar maps : applications to the Ising model.* Rapport de recherche, novembre, 2002.

[9] P. CHASSAING, G. SCHAEFFER. *Random Planar Lattices and Integrated SuperBrownian Excursion.* Rapport de recherche, juin, 2002.

[10] I. DEBLED-RENNESSON, J.-L. RÉMY, J. ROUYER-DEGLI. *Segmentation of Discrete Curves into Fuzzy Segments.* Rapport de recherche, novembre, 2002, Soumis à IWCIA.

[11] P. FLAJOLET, B. SALVY, G. SCHAEFFER. *Airy Phenomena and Analytic Combinatorics of Connected Graphs.* Rapport de recherche, octobre, 2002.

[12] G. KUCHEROV, P. OCHEM, M. RAO. *How many square occurrences must a binary sequence contain.* rapport technique, LORIA, 2002.

[13] L. NOÉ. *Recherche de répétitions distantes dans les séquences.* Stage de DEA, LORIA, Université Henri Poincaré Nancy 1, juillet, 2002.

[14] D. POULALHON, G. SCHAEFFER. *A bijection for triangulations of a polygon with interior points and multiple edges.* Rapport de recherche, octobre, 2002.

[15] D. POULALHON, G. SCHAEFFER. *A note on Bipartite Eulerian Planar Maps.* Rapport de recherche, avril, 2002.

Divers

[16] P. LAVIGNE. *Identification et classification des promoteurs chez Streptomyces coelicolor et Streptomyces ambofaciens.* Rapport intermédiaire du stage de DESS *Etude de Génomes : Outils Informatiques et Statistiques*, Université de Rouen, 2002.

[17] S. LUIS. *Analyse de signaux de transcription dans le génome de Streptomyces coelicolor.* Rapport de stage de DESS *Compétence Complémentaires en Informatique*, Université Henri Poincaré Nancy 1, 2002.

Bibliographie générale

[18] S. ALTSCHUL, T. MADDEN, A. SCHÄFFER, J. ZHANG, Z. ZHANG, W. MILLER, D. LIPMAN. *Gapped BLAST and PSI-BLAST : a new generation of protein database search programs.* in « Nucleic Acids Research », numéro 17, volume 25, 1997, pages 3389-3402.

[19] C. BANDERIER, P. FLAJOLET, G. SCHAEFFER, M. SORIA. *Random Maps, Coalescing Saddles, Singularity Analysis, and Airy Phenomena.* in « Random Structures and Algorithms », numéro 3-4, volume 19, décembre, 2001, pages 194-246.

[20] G. BENSON. *Tandem repeats finder : a program to analyse DNA sequences.* in « Nucleic Acids Research », numéro 2, volume 27, 1999, pages 573-580.

[21] J.-M. CHASSERY, A. MONTANVERT. *Géométrie discrète en imagerie.* Hermès, Paris, 1991.

[22] C. CHELALA, M.-D. DEVIGNES, S. IMBEAUD, R. ZOOROB, D. COX, C. AUFFRAY. *Inconsistencies between maps of human chromosome 22 correlate with increased frequency of disease-related loci.* 2002, en cours de

publication.

[23] D. COEURJOLLY, L. TOUGNE, Y. GÉRARD, J.-P. REVEILLÈS. *An Elementary Algorithm for Digital Arc Segmentation.* volume 46, 2001.

[24] M. CROCHEMORE. *Recherche linéaire d'un carré dans un mot.* in « Comptes Rendus Acad. Sci. Paris Sér. I Math. », volume 296, 1983, pages 781-784.

[25] M. CROCHEMORE, C. HANCART, T. LECROQ. *Algorithmique du texte.* Vuibert Informatique, 2001.

[26] M. CROCHEMORE, W. RYTTER. *Text algorithms.* Oxford University Press, 1994.

[27] I. DEBLED-RENNESSON, J.-L. RÉMY, J. ROUYER-DEGLI. *Detection of the Discrete Convexity of Polyominoes.* in « DGCI'2000, Uppsala, Suède », série Lecture Notes in Computer Science, volume 1953, Springer-Verlag, pages 491-504, décembre, 2000.

[28] P. FLAJOLET, A. ODLYZKO. *Singularity analysis of generating functions..* in « SIAM J. Discrete Math. », numéro 2, volume 3, 1990, pages 216-240.

[29] P. FLAJOLET, R. SEDGEWICK. *The average case analysis of algorithms.* 2001, Livre en préparation, certaines parties disponibles comme rapports INRIA.

[30] P. FLAJOLET, P. ZIMMERMAN, B. VAN CUTSEM. *A Calculus for the Random Generation of Labelled Combinatorial Structures.* in « Theoretical Computer Science », numéro 1-2, volume 132, 1994, pages 1-35.

[31] A. FRAENKEL, J. SIMPSON. *How many squares must a binary sequence contain ?.* in « Electronic Journal of Combinatorics », numéro R2, volume 2, 1995, pages 9pp.

[32] D. GUSFIELD. *Algorithms on Strings, Trees, and Sequences.* Cambridge University Press, 1997.

[33] D. KNUTH, J. MORRIS, V. PRATT. *Fast pattern matching in strings.* in « SIAM J. Comput. », volume 6, 1977, pages 323-350.

[34] R. KOLPAKOV, G. KUCHEROV. *Finding Approximate Repetitions under Hamming Distance.* in « 9-th European Symposium on Algorithms (ESA 2001), Aarhus, Denmark », série Lecture Notes in Computer Science, volume 2161, éditeurs F. AUF DER HEIDE., pages 170 - 181, août, 2001.

[35] R. KOLPAKOV, G. KUCHEROV. *Finding Maximal Repetitions in a Word in Linear Time.* in « Proceedings of the 1999 Symposium on Foundations of Computer Science, New York (USA) », IEEE Computer Society, pages 596-604, New-York, 17-19 octobre, 1999.

[36] R. KOLPAKOV, G. KUCHEROV, Y. TARANNIKOV. *On repetition-free binary words of minimal density.* in « Theoretical Computer Science », numéro 1, volume 218, 1999.

[37] S. R. KOSARAJU. *Computation of Squares in String.* in « Proceedings of the 5th Annual Symposium on Combinatorial Pattern Matching », série Lecture Notes in Computer Science, numéro 807, Springer Verlag,

éditeurs M. CROCHEMORE, D. GUSFIELD., pages 146-150, 1994.

[38] G. MANACHER. *A new linear-time on-line algorithm for finding the smallest initial palindrome of the string.* in « J. ACM », volume 22, 1975, pages 346-351.

[39] F. MURI-MAJOUBE, B. PRUM. *Une approche statistique de l'analyse des génomes.* in « Gazette des mathématiciens », volume 89, 2001.

[40] F. PLEWNIAK, J. THOMPSON, O. POCH. *Ballast : Blast post-processing based on locally conserved segments.* in « Bioinformatics », volume 16, 2000, pages 750-759.

[41] J.-P. REVEILLÈS. *Géométrie discrète, calculs en nombre entiers et algorithmique.* Thèse d'état, Université Louis Pasteur, Strasbourg, 1991.

[42] R. P. STANLEY. *Enumerative combinatorics. Volume 2. Paperback ed.*. Cambridge Studies in Advanced Mathematics. 62. Cambridge : Cambridge University Press. xii, 585 p. , 2001.

[43] G. STEPHEN. *String Searching Algorithms.* World Scientific, Singapore, 1994.

[44] J. A. V. W. WAN. *Segmentation of Planar Curves into Straight-Line Segments and Elliptical Arcs.* volume 59, 1997, pages 484-494.