

# Sujets de Projets d'Initiation à la Recherche 2011-2012

## Master 1 Mathématiques (Cryptis)

Les projets seront réalisés par groupes de 2 ou 3. Vous pouvez contacter les enseignants et prendre rendez-vous pour vous faire une idée. Nous ferons ensuite une réunion pour décider collégalement de la répartition des projets (au cas où plusieurs groupes voudraient le même). Pour contacter les enseignants : [prenom.nom-at-unilim.fr](mailto:prenom.nom-at-unilim.fr)

### *Thierry Berger* ★ **La décimation des suites générées par des automates LFSR et FCSR**

Résumé : (cryptographie, codage) Une des techniques les plus classiques pour le chiffrement à flot consiste à utiliser un générateur pseudo-aléatoire qui fournit une suite ajoutée au message modulo 2.

Les automates LFSR (Linear Feedback Shift Register) génèrent de manière simple des suites pseudo-aléatoires binaires de manière linéaire. A ce titre elles sont un outil de base dans beaucoup d'applications informatiques, en particulier en cryptographie symétrique.

D'un point de vue mathématiques, il s'agit de séries rationnelles de la forme  $P(X)/Q(X)$  dans l'algèbre de polynômes  $GF(2)[X]$ . Il existe des algorithmes simples (Berlekamp-Massey, Euclide étendu) qui permettent de retrouver les polynômes  $P(X)$  et  $Q(X)$  à partir d'un petit nombre d'éléments de la suite. Cette propriété reste vraie si on décime la suite (c'est-à-dire si on enlève régulièrement des termes pour garder par exemple un symbole sur 2 ou 3).

En cryptographie, on utilise des automates similaires appelés FCSR (Feedback with Carries Shift Registers). Les FCSR sont l'analogie des LFSR dans les entiers 2-adiques, c'est-à-dire que les opérations s'effectuent en prenant en compte des retenues. Ces automates consiste à faire une division d'entiers  $p/q$  en utilisant un algorithme de division selon les puissances croissantes de 2.

De manière analogue aux LFSR, il est facile de retrouver la fraction rationnelle  $p/q$  des suites générées en utilisant l'algorithme d'Euclide étendu dans les entiers. Par contre, il semble que la décimation ne conserve pas cette propriété.

Ce projet consiste à comprendre le fonctionnement des LFSR et FCSR et expliquer la différence de comportement de ces 2 familles d'automates par rapport à la décimation. Ces résultats ont des conséquences sur le design de certains algorithmes de chiffrement à flot ou par bloc en cryptographie.

### *J.A Weil* ★ **Algorithme de Beckerman-Labahn**

Résumé : (algorithmique, calcul formel) Dans de nombreux problèmes de calcul formel, on trouve une approximation de quantités sous forme de séries formelles, et on cherche ensuite une "relation exacte" liant ces séries (par exemple une dépendance à coefficients polynomiaux). La méthode développée par Beckerman et Labahn répond à cette question. Le travail consistera à comprendre et implanter cette méthode, puis à l'appliquer à plusieurs situations intéressantes liées à nos recherches.

### *Stéphane Vinatier* ★ **Bases optimales pour extensions de corps finis**

Résumé : (arithmétique des corps finis) La rapidité de la multiplication dans une extension  $\mathbb{F}_q$  du corps fini  $\mathbb{F}_p$  à  $p$  éléments dépend de la base choisie pour représenter les éléments de  $\mathbb{F}_q$ . Il est souvent pratique d'utiliser une base normale (c'est-à-dire formée des conjugués d'un même élément), auquel cas l'information nécessaire pour multiplier deux éléments quelconques est contenue dans une matrice appelée « table de multiplication ». La complexité de la multiplication dans une telle base est définie comme étant le nombre d'éléments non nuls de cette table.

On propose d'étudier l'article *Optimal normal bases in  $\mathbb{F}_q$*  de MULLIN, ONYSZCHUK, VANSTONE et WILSON, qui montre que la complexité est toujours au moins égale à  $2n - 1$ , où  $q = p^n$ , et qui construit des extensions pour lesquelles une base ayant cette complexité (dite « optimale ») existe.

Pré-requis : connaissances de base sur les corps finis.

### *François Arnault* ★ **Bases mutuellement non biaisées**

Résumé : (arithmétique des corps finis, cryptographie) Deux bases  $(e_1, \dots, e_n)$  et  $(f_1, \dots, f_n)$  d'espaces vectoriels complexes métriques sont dites mutuel-

lement non biaisées (in English *Mutually Unbiased Bases* or M.U.B.) si

$$\langle e_i, f_j \rangle = \frac{1}{\sqrt{n}} \quad \text{pour } 1 \leq i, j \leq n.$$

Ces objets sont importants en Information Quantique et peuvent être utiles en cryptographie. Ils entretiennent des liens étroits avec les carrés latins. On sait en construire  $n + 1$  lorsque  $n$  est une puissance d'un nombre premier, mais ce problème est très ouvert pour les autres valeurs de  $n$  (en particulier  $n = 6$ ).

Le travail consisterait en premier lieu à comprendre et implanter (probablement en Magma) la construction de telles bases dans le cas connu, puis à expérimenter dans les cas  $n = 6$  et  $n = 10$ .

### ***P. Boito* \* Application des fonctions des matrices a l'étude des réseaux, ou comment mesurer sa popularité sur Facebook**

Résumé : (algèbre linéaire numérique, théorie des graphes) L'étude des graphes associés aux réseaux trouve de nombreuses applications en physique, chimie, biologie, télécommunications, sciences sociales, etc. Dans ce travail, nous chercherons à quantifier l'importance d'un noeud du réseau, ainsi que la connectivité entre deux noeuds. Pour ce faire, des méthodes basées sur le calcul des fonctions de la matrice d'adjacence ont récemment été proposées par E. Estrada et ses collaborateurs. L'objet de ce projet consiste à :

- comprendre les structures des graphes qui modélisent des exemples typiques de réseaux,
- explorer le lien entre les fonctions des matrices et les propriétés des réseaux,
- connaître les enjeux du calcul numérique de ces fonctions,
- implémenter les techniques apprises et les appliquer à quelques cas réels.

Pré-requis : connaissances de base d'algèbre linéaire et programmation.

Références : E. Estrada, D. Higham, *Network Properties Revealed through Matrix Functions*, SIAM Review vol. 52 n. 4, pp. 696-714 (December 2010).

### ***C. Clavier* \* Détermination d'une borne inférieure d'un éventuel nombre parfait impair**

Résumé : Un nombre parfait est un entier égal à la somme de ses diviseurs (lui-même excepté). 6 et 28 sont deux exemples de nombres parfaits. Tous les nombres parfaits connus actuellement sont pairs, et la question non résolue depuis l'antiquité de l'existence d'un nombre parfait impair

constitue la conjecture la plus ancienne des mathématiques. Faute de pouvoir prouver la non-existence de tels nombres, des résultats toujours plus évolués permettent d'établir des contraintes de plus en plus fortes sur de tels nombres : borne inférieure sur sa valeur, nombre total de facteurs premiers, nombre de facteurs premiers distincts,...

Après avoir établi un état de l'art sur la recherche des nombres parfaits impairs, vous analyserez un article décrivant une méthode permettant de générer – sous la forme d'un arbre de contradictions – une preuve qu'un nombre parfait impair ne peut pas être inférieur à  $10^{300}$ . La partie principale de votre travail consistera à étudier et comprendre cette méthode, et à l'implémenter pour générer la-dite preuve.

*Pierre Dusart* ★ **Sur les zéros de la fonction  $\zeta$  de Riemann.**

Résumé : La fonction  $\zeta$  de Riemann est très connue en théorie des nombres. Elle est étroitement liée avec les nombres premiers. En particulier, on peut dire que les zéros de la fonction zeta contrôlent les oscillations des nombres premiers autour de leur position « attendue ». Il est donc intéressant de connaître avec précision les valeurs de ces zéros. L'histoire débute aux alentours de 1900 avec le calcul de 3 premiers zéros jusqu'à maintenant avec  $10^{13}$  zéros calculés. Références : J. P. Gram, « Note sur les zéros de la fonction  $\zeta(s)$  de Riemann », dans Acta Mathematica, vol. 27, 1903, p. 289–304

*Carlos Aguilar and Marc Rybowicz* ★ **Chiffrement complètement homomorphe basé sur les réseaux euclidiens**

Résumé : (Cryptographie - Chiffrement homomorphe - Protection de la vie privée - Réseaux euclidiens)

En 1978, Rivest, Adleman et Dertouzos, trois chercheurs extrêmement reconnus dans le domaine de la cryptographie (les deux premiers étant co-auteurs du célèbre cryptosystème RSA), conjecturent l'existence d'un système de chiffrement particulier : le chiffrement complètement homomorphe.

Un tel système permettrait de réaliser des calculs, et de façon plus générique exécuter des programmes arbitraires, sur des données chiffrées, sans passer par une étape de déchiffrement. Par exemple, soit un prestataire qui a un service de factorisation  $f$  de grands nombres. Le client peut envoyer au prestataire un chiffré du nombre à factoriser  $x$  et, si le système de chiffrement est complètement homomorphe, le prestataire pourra calculer une version chiffrée du résultat  $E(f(x))$  sans rien apprendre sur quel nombre il a factorisé, ou quels sont ses facteurs. Ce résultat peut être ren-

voyé au client qui sera capable d'extraire  $f(x)$ .

Inversement, on peut souhaiter distribuer un programme, où demander à un prestataire d'exécuter un programme pour nous (par exemple dans le cadre du *Cloud Computing*), sans que le fonctionnement de ce programme ne soit dévoilé. L'utilisation des systèmes de chiffrement complètement homomorphes dans cette optique n'est cependant pas sans problèmes, et relève de nombreux défis extrêmement intéressants.

Enfin, il est important de remarquer que pendant près de trente ans, l'obtention d'un système de chiffrement complètement homomorphe est resté une question ouverte, et que les systèmes de chiffrement homomorphes les plus utilisables en pratique (en termes de coût calculatoire, taille, et facteur d'expansion) ne permettent de réaliser que certaines opérations sur les données en clair. Le cas s'étant révélé le plus utile est celui des systèmes permettant de réaliser des sommes sur les clairs. L'application phare de ces systèmes est les protocoles PIR (permettant de récupérer un élément d'une base de données sans que les administrateurs de celle-ci sachent lequel) mais leur polyvalence va bien au delà de cette application particulière.

Depuis l'introduction du concept de chiffrement homomorphe par Rivest, Adleman et Dertouzos en 1978, de nombreux systèmes de chiffrement avec des propriétés homomorphes ont vu le jour. La plupart d'entre eux ne permettent de calculer sur des données chiffrées qu'un seul type d'opération. C'est le cas des systèmes homomorphes multiplicatifs (RSA, El Gamal), et des systèmes homomorphes additifs (Goldwasser-Micali modulo 2 et Paillier modulo un grand nombre difficile à factoriser).

Ces systèmes permettent donc de calculer des monômes de degré arbitraire ou des polynômes multivariés de degré un. Obtenir un système permettant d'évaluer sur des données chiffrées des fonctions plus complexes comme un polynôme multivarié de degré arbitraire est un problème beaucoup plus difficile. Pour essayer d'aller au delà de ce que permettaient les premiers systèmes, certaines approches ont détourné les chiffrements homomorphes additifs dans des nouveaux contextes comme, les circuits booléens et les programmes à branchement. Les systèmes obtenus permettent d'évaluer des fonctions relativement arbitraires tout en protégeant le secret de la fonction. Cependant ces systèmes ne sont pas concis (les données envoyées par Alice ont une taille supérieure à celle de la représentation de la fonction calculée) ce qui est rédhibitoire pour la plupart des applications.

Ainsi, l'obtention d'un système de chiffrement homomorphe permettant d'évaluer de façon concise sur des données chiffrées des polynômes de degré supérieur à un a été pendant près de trente ans une question ouverte. Les premières approches qui ont tenté d'obtenir un chiffrement

complètement homomorphe (c'est-à-dire permettant de calculer sur des données chiffrées un nombre arbitraire de produits et de sommes), n'ont pas résisté aux attaques de la communauté : Fellows et Koblitz proposèrent Polly Cracker qui fut cassé par Steinwadt en 2002 ; Domingo-Ferrer proposa deux systèmes qui furent cassés dans par Cheon et Wagner ; et Grigoriev et Ponomarenko proposèrent un autre système qui fut cassé par Wagner.

Le premier pas dans la direction du chiffrement complètement homomorphe fut donné en 2005 par Boneh, Goh et Nissim qui proposèrent le premier système permettant d'évaluer de façon concise des polynômes de second degré tant que l'évaluation du polynôme est un petit nombre (le coût de déchiffrement est polynomial en ce nombre). Malheureusement, l'approche proposée par ces trois auteurs est difficile de généraliser vers des polynômes de degré supérieur, et bien qu'ils introduisent des idées très intéressantes celles-ci ne peuvent pas développer leur plein potentiel avec les systèmes de chiffrement basés sur des problèmes classiques.

Indépendamment des travaux sur le chiffrement homomorphe, une branche de la complexité entre avec force dans le monde de la cryptographie dans les années 90 à cause des révolutionnaires preuves de sécurité pouvant être atteintes (les réductions dites "pire-cas/cas-moyen"). Le lien entre ce domaine, la cryptographie basée sur les réseaux, et celui du chiffrement homomorphe est évident depuis très tôt, mais les propriétés homomorphes que l'on peut obtenir semblent être à première vue beaucoup plus faibles que celles déjà atteintes par les systèmes de chiffrement classiques et cette connexion n'est que peu étudiée par la communauté.

Ce n'est qu'à partir de 2008 que les systèmes de chiffrement basés sur les réseaux donnent lieu à des avancées fondamentales vers le chiffrement homomorphe. Cette année, en collaboration avec Philippe Gaborit, et Javier Herranz nous proposons une approche permettant d'évaluer des polynômes de degré supérieur à deux (mais pas trop élevé). Cette même année, Craig Gentry présente au séminaire de Dagstuhl une approche permettant d'évaluer des polynômes arbitraires, en se basant sur des nouveaux problèmes de sécurité. Ce résultat suscite une grande émotion dans la communauté.

Très vite, de nombreux résultats font progresser l'approche de Gentry, de façon à la rendre moins coûteuse (telle que présentée par Gentry elle est théoriquement faisable mais irréalisable en pratique), et à éviter les liaisons avec des problèmes de sécurité peu standards. Enfin, en 2011, Brakerski et Vaikuntanathan, en se basant sur l'approche que nous avons proposé en 2008 et en introduisant plusieurs nouvelles techniques d'une grande subtilité ils réussissent à obtenir à la fois la généralité des résultats de Gentry,

tout en gardant la solidité de nos hypothèses. Pour couronner le tout, leur approche est tout a fait réalisable en pratique. Au vu des présentations de cette année à la conférence CRYPTO, des implémentation pratiques sont en cours, mais n'ont pas pour le moment été publiées.

Dans ce projet les étudiants devront se pencher sur une version préliminaire des travaux de Brakerski et Vaikuntanathan parue à Crypto 2011 (la version complète est parue à FOCS 2011) et permettant d'obtenir un système de chiffrement complètent homomorphe en utilisant un anneau polynomial et des techniques classiques des réseaux euclidiens. Une implémentation partielle du système sera éventuellement demandée. Les étudiants pourront découvrir avec ce projet le domaine de la cryptographie basée sur les réseaux ainsi que les dernières avancées dans le domaine du chiffrement complètement homomorphe.

*Carlos Aguilar and Marc Rybowicz* \* **Protocole de retrait d'informations privé basé sur les arbres binaires ordonnés**

Résumé : (Cryptographie - Chiffrement homomorphe - Protection de la vie privée - Fonctions booléennes - Arbres binaires)

En général, pour récupérer un élément d'une base de données, un utilisateur envoie une requête indiquant quel élément l'intéresse, puis la base lui renvoie l'élément en question. Quel élément de la base de données intéresse un utilisateur peut être une information que celui-ci souhaite garder secrète, même auprès des administrateurs de la base. Par exemple, la base peut être :

- une bibliothèque électronique, et quels livres nous lisons peut fournir des informations sur nos convictions politiques, ou certains détails de notre personnalité qu'il peut être souhaitable de garder confidentiels ;
- une base de données pharmaceutique, et certains laboratoires clients de la base peuvent désirer qu'on ne puisse pas savoir à quels principes actifs ils s'intéressent ;
- des cours d'actions, et les clients peuvent être des investisseurs ne voulant pas dévoiler quels cours les intéressent.

Une manière évidente de résoudre ce problème consiste pour un utilisateur à télécharger toute la base, et à extraire localement l'information qui l'intéresse. Ceci est en général inacceptable, voir même impossible si la base est de taille extrêmement importante (par exemple une bibliothèque électronique), confidentielle (par exemple, une base de données pharmaceutique), ou rapidement obsolète (par exemple, des cours d'actions).

Un protocole PIR (de l'anglais *Private Information Retrieval*) est un protocole permettant à un utilisateur d'obtenir un élément d'une base de

données, sans dévoiler aucune information, même aux gestionnaires de la base, sur quel élément particulier l'intéresse, avec une quantité de données transférées qui est idéalement de l'ordre de la taille de l'élément téléchargé. Pour ce faire on utilise généralement un système de chiffrement homomorphe additif, c'est à dire un schéma de chiffrement qui à partir de deux chiffrés associés à deux clairs donnés permet d'obtenir, sans passer par une étape de déchiffrement, un chiffré de la somme des deux clairs. En général on réalise une multiplication modulaire dans l'espace des chiffrés et ceci se traduit par une somme dans l'espace des clairs.

Pendant très longtemps la communauté scientifique a pensé que pour que le serveur n'apprenne rien sur quel élément intéresse l'utilisateur, il fallait que le calcul nécessaire pour générer la réponse fasse intervenir tous les éléments de la base. En effet, si certains éléments n'étaient pas traités, on pourrait en déduire qu'ils n'ont pas pu être envoyés dans la réponse et donc qu'ils n'intéressent pas l'utilisateur.

En 2009, Lipmaa a proposé un protocole remarquable permettant de réaliser un calcul ayant un coût sous-linéaire par rapport à la taille de la base de données. Pour ce faire il a utilisé un résultat fondamental de Breitbart *et al*, qui ont montré en 1995 que pour représenter une fonction booléenne arbitraire ayant un espace de définition de taille  $n$  il suffisait d'utiliser un graphe de taille  $n/\log n$ . L'idée de Lipmaa est de définir une fonction booléenne qui à  $i \in \{1, \dots, n\}$  associe le  $i$ -ème élément de la base de données et de représenter cette fonction comme un graphe. En réalisant un protocole PIR sur ce graphe il réussit à avoir un coût calculatoire en  $n/\log n$ .

Dans ce projet les étudiants devront étudier l'article de Lipmaa et éventuellement développer une sous partie du protocole. Ce projet permettra aux étudiants de découvrir les résultats fondamentaux sur les fonctions booléennes décrits ci-dessus, ainsi que de manipuler un système de chiffrement homomorphe pour une applications pratique.

### *Olivier Ruatta* ★ **Actions de semi-groupe sur un ensemble fini pour la conception de protocole de partage de secret du type Diffie-Hellman**

Résumé : À chaque action de semi-groupe sur un ensemble fini est associé un problème de logarithme discret (DLP pour discrete logarithm problem). Basé sur ce problème on peut proposer un protocole pour le partage de secret calqué sur l'idée du protocole de Diffie-Hellman. Dans ce sujet d'initiation, on étudiera le protocole de Diffie-Hellman, puis les action de semi-groupes et surtout les problèmes de logarithmes discrets associés. On étudiera alors le protocole proposé dans l'article de Maze, Monico et Rosenthal.. On fera de petits tests concrets et si le temps le per-

met on étudiera quelques attaques sur ce protocole.

Références : Gérard Maze, Chris Monico et Joachim Rosenthal. **Public Key Cryptography Based on Semigroup Actions**; Advances in Mathematics of Communications, Vol. 1, No. 4, 2007, 489-507.