317

ASTÉRISQUE

2008

SÉMINAIRE BOURBAKI
VOLUME 2006/2007
EXPOSÉS 967-981

(968) *Compter (rapidement) le nombre de solutions d'équations dans les corps finis*

Antoine CHAMBERT-LOIR

SOCIÉTÉ MATHÉMATIQUE DE FRANCE
Publié avec le concours du CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE

Séminaire BOURBAKI Novembre 2006
59e année, 2006–2007, no 968, p. 39 à 89

COMPTER (RAPIDEMENT) LE NOMBRE DE SOLUTIONS D'ÉQUATIONS DANS LES CORPS FINIS

par **Antoine CHAMBERT-LOIR**

— Votre Sérénité, pouvez-vous me dire, c'est très important, concentrez-vous, pouvez-vous me dire quel est le numéro du compte en banque de monsieur ?
— Oui.
— Vous pouvez le dire ?
— Oui !!
— Vous pouvez le dire ?
— Oui !!!
— Il peut le dire !!! Bravo ! Il est extraordinaire, il est vraiment sensationnel.

(Pierre Dac et Francis Blanche, *Sar Rabindranath Duval*)

INTRODUCTION

Soit $\mathbf{F}$ un corps fini et soit $f_1, \dots, f_m$ des polynômes à coefficients dans $\mathbf{F}$ en n indéterminées $x_1, \dots, x_n$. Le but de cet exposé est de décrire des algorithmes permettant de calculer efficacement le nombre de solutions dans $\mathbf{F}^n$ du système d'équations $f_1 = \cdots = f_m = 0$.

Pour tout entier $k \geqslant 1$, notons N_k le nombre de solutions de ce système dont les coordonnées appartiennent au corps $\mathbf{F}^{(k)}$, unique extension de $\mathbf{F}$ de degré k contenue dans une clôture algébrique fixée de $\mathbf{F}$; si X est le sous-schéma de l'espace affine défini par l'annulation des f_i, on a donc $N_k = |X(\mathbf{F}^{(k)})|$. La fonction zêta $Z(X,t)$ du schéma X est alors donnée par la formule

$$Z(X,t) = \exp\left(\sum_{k=1}^{\infty} N_k \frac{t^k}{k}\right) \tag{0.1}$$

et, ainsi que l'a démontré DWORK [**41**], c'est une fraction rationnelle. Par conséquent, la suite (N_k) est déterminée par un nombre fini de ses termes. Nous verrons aussi que ces algorithmes permettent de calculer la fonction zêta $Z(X,t)$.

Tous les algorithmes décrits ci-dessous reposent sur un premier principe : il suffit, pour calculer $|X(\mathbf{F})|$, de calculer une congruence $|X(\mathbf{F})| \equiv c \pmod{N}$, où N est un entier strictement supérieur à $|X(\mathbf{F})|$, par exemple $N > |\mathbf{F}|^n$. Plus généralement, il suffit que l'on connaisse un encadrement de $|X(\mathbf{F})|$ de largeur inférieure à N ; c'est là qu'interviendra l'analogue de l'hypothèse de Riemann sur les corps finis, que DELIGNE [**34**] a démontrée, généralisant ainsi des résultats de HASSE (courbes elliptiques) et WEIL (courbes, variétés abéliennes,...). Fixons-nous un tel encadrement $C \leqslant |X(\mathbf{F})| < C + R$.

Là où ces algorithmes diffèrent, c'est sur la façon de choisir un tel entier N puis de calculer c.

Les premiers algorithmes, que nous qualifierons de ℓ-adiques, font l'objet du chapitre 2 de ce rapport. Ils ont pour archétype l'algorithme découvert en 1985 par R. SCHOOF [**106**] pour calculer le nombre de points d'une courbe elliptique sur un corps fini. Ces algorithmes choisissent un ensemble fini $\{\ell_1, \dots, \ell_s\}$ de « petits » nombres premiers dont le produit $L = \ell_1 \dots \ell_s$ vérifie $L > R$ et calculent, pour tout i, un élément $c_i \in \{0, \dots, \ell_i - 1\}$ tel que $|X(\mathbf{F})| \equiv c_i \pmod{\ell_i}$. Le théorème chinois permet d'en déduire un entier $c \in \{0, \dots, L-1\}$ tel que $|X(\mathbf{F})| \equiv c \pmod{L}$. L'origine de la terminologie « ℓ-adique » vient de ce qu'on peut interpréter la congruence modulo ℓ_i par le calcul de la cohomologie étale modulo ℓ_i.

Hors du degré 1 ou de ce qui en provient, la cohomologie étale semble peu accessible au calcul formel ; même dans ce cas, son calcul effectif amène rapidement à la considération de polynômes de très grand degré. Le champ d'application des algorithmes ℓ-adiques est ainsi limité aux courbes de petit genre, aux variétés abéliennes de petite dimension.

Néanmoins, ces algorithmes sont polynomiaux en le logarithme du cardinal de $\mathbf{F}$: aussi bien le temps de calcul que l'espace requis par le calcul sont majorés par une puissance de $\log|\mathbf{F}|$.

Nous présenterons au chapitre 3 les algorithmes p-adiques, où l'entier p désigne la caractéristique du corps $\mathbf{F}$. Ils procèdent en effet en choisissant pour N une puissance de p et en calculant (plus ou moins) la cohomologie p-adique de X modulo N. Par cohomologie p-adique, j'entends ici la cohomologie de Monsky-Washnitzer et ses avatars (rigide, cristalline), qui sont des analogues de la cohomologie de De Rham. Définie comme cohomologie d'un complexe explicite, la cohomologie p-adique se prête naturellement bien au calcul effectif et l'on peut espérer appliquer ces méthodes dans des situations géométriques très générales. Malgré tout, il semble que seules les courbes et les surfaces aient fait l'objet d'implémentations poussées.

Toutefois, parce qu'ils demandent de manipuler des polynômes de degrés au moins p, la dépendance en $\log p$ de leur complexité n'est pas polynomiale. Ils n'en restent pas moins des algorithmes de choix lorsque p est petit, notamment dans les applications cryptographiques où l'on a souvent $p = 2$.

Au fur et à mesure du développement de ces algorithmes, ils ont été programmés et leurs performances éprouvées à l'aune des records qu'ils permirent d'obtenir, c'est-à-dire le calcul de $|X(\mathbf{F})|$ pour des corps $\mathbf{F}$ de cardinal le plus grand possible. Lorsque X est une courbe elliptique, on a pu atteindre un cardinal q de plus de 2 000 chiffres (en base 10) par l'algorithme de SCHOOF, et d'environ 40 000 chiffres (mais en caractéristique $p = 2$) par l'algorithme 2-adique de MESTRE. Ces calculs ont pris plusieurs mois. La diminution de l'espace mémoire nécessité par ces algorithmes a aussi fait l'objet de travaux importants.

Parallèlement, ils ont trouvé un champ d'application dans la cryptographie à clef publique et se sont retrouvés au cœur de logiciels commerciaux. Comme nous le verrons plus bas, les corps $\mathbf{F}$ qu'il faut alors manipuler sont de taille bien plus modeste, disons une cinquantaine de chiffres décimaux.

Le premier chapitre de ce texte est consacré à quelques applications de ce problème algorithmique et de ses diverses solutions efficaces. J'exposerai ensuite les grandes lignes de la plupart des algorithmes ℓ-adiques, puis p-adiques, actuellement utilisés. Il s'avère en fait qu'une bonne partie de la théorie générale et abstraite développée au XX$^{\text{e}}$ siècle dans l'étude des conjectures de Weil donne naturellement lieu à des algorithmes efficaces. Cependant, cette constatation n'est pas allée de soi et le crédit en revient bien aux mathématiciens tels que SCHOOF, ELKIES, ATKIN (pour la partie ℓ-adique), SATOH, MESTRE, KEDLAYA, LAUDER (pour la partie p-adique) dont les noms émailleront ce texte. À moins d'achever cet exposé juste après le chapitre consacré aux applications, il m'a ainsi fallu dépasser le lapidaire et spontané « On peut le faire ! » sans pour autant plonger le lecteur dans la complexité phénoménale des idées supplémentaires qui ont été nécessaires à l'obtention des records évoqués plus haut. Le compromis que j'ai essayé d'adopter dans ce texte, un peu différent des nombreux survols du sujet disponibles dans la littérature, est celui d'un mathématicien pur subitement intéressé par ce problème de mathématiques appliquées.

Lorsque je décris la complexité d'algorithmes en temps ou en espace, j'emploie les notations $\mathrm{O}(\cdot)$ et $\widetilde{\mathrm{O}}(\cdot)$. La première signifie que le nombre d'opérations élémentaires, resp. l'espace disque, requis par l'algorithme est majoré par un multiple de son argument, lorsque celui-ci tend vers l'infini. La seconde est analogue, à une puissance du logarithme de l'argument près ; en pratique, il suffit de retenir que $\widetilde{\mathrm{O}}(x)$ est majoré par $\mathrm{O}(x^{1+\varepsilon})$ pour tout $\varepsilon > 0$. Toutefois, même si je n'en parlerai jamais, il ne faut pas perdre de vue que le contrôle de la constante que cachent ces notations est d'une

importance pratique capitale ; il est bien différent de pouvoir obtenir un résultat en une minute plutôt qu'en mille.

Je tiens à remercier Jean-Benoît Bost, Bas Edixhoven, Reynald Lercier, Bernard Le Stum, David Lubicz et Jean-François Mestre de l'aide qu'ils m'ont apportée au cours de la préparation de cet exposé. Je remercie aussi Robert Carls, David Kohel, Alan Lauder, René Schoof, Jean-Pierre Serre et Frederik Vercauteren pour leurs commentaires sur la première version de ce texte.

1. APPLICATIONS

1.1. Critères de primalité

Être en mesure de décider si un entier naturel est ou pas un nombre premier est une question arithmétique fondamentale dont les techniques modernes de cryptographie ont d'ailleurs accru l'importance.

En 1986, S. Goldwasser et J. Kilian ont proposé (voir [**57**]) le premier algorithme permettant de décider si un entier N est un nombre premier dont la complexité soit polynomiale en $\log N$. Cet algorithme requiert de calculer le cardinal de courbes elliptiques E sur l'anneau $\mathbf{Z}/N\mathbf{Z}$ « choisies au hasard ». Pour cela, on peut tenter d'appliquer l'algorithme de Schoof, en faisant comme si N était premier. Si l'algorithme échoue, cela prouve que N n'est pas premier. Supposons qu'il fournisse un cardinal putatif c. On peut tester si un point au hasard P sur la courbe E est annulé par c ; si ce n'est pas le cas, N n'est pas premier. Inversement, supposons que l'ordre d de P possède un facteur premier p tel que $p > (1+\sqrt{N})^2$ et tel que le point $[d/p]P$ ne rencontre pas l'origine O de la courbe E (au sens où ce point $[d/P]P$ ait des coordonnées homogènes $(x:y:z)$ dans $(\mathbf{Z}/N\mathbf{Z})$, z étant premier à N) ; alors N est premier. (Sinon, désignant par ℓ le plus petit facteur premier de N, l'image de P dans $E(\mathbf{Z}/\ell\mathbf{Z})$ serait d'ordre multiple de p, et cela contredirait la borne de Hasse pour le cardinal d'une courbe elliptique sur un corps fini.) L'algorithme de Goldwasser et Kilian tente alors d'exhiber de telles familles (E,P,d,p) où $|E(\mathbf{Z}/N\mathbf{Z})| = 2p$, la primalité de p étant établie récursivement par la même méthode.

Comme l'algorithme de Schoof est de complexité polynomiale en $\log N$, il en est de même de celle de l'algorithme de Goldwasser et Kilian. Toutefois, le fait que cet algorithme parvienne à conclure pour tout N dépend d'une conjecture apparemment hors de portée sur la répartition des nombres premiers dans de petits intervalles.

Adleman et Huang [**2**] ont eu l'idée d'utiliser des courbes de genre 2. Cela fournit plus de latitude et leur permet d'affirmer l'existence d'un algorithme probabiliste de complexité polynomiale en $\log N$ permettant de décider si l'entier N est premier.