Département de formation doctorale en informatique **École doctorale IAEM Lorraine**
UFR STMIA

Quelques algorithmes en arithmétique

MÉMOIRE

présenté et soutenu publiquement le 9 décembre 2005

pour l'obtention de l'

Habilitation de l'Université Henri Poincaré – Nancy I

(Spécialité Informatique)

par

Guillaume Hanrot

Composition du jury

Rapporteurs :	Maurice Mignotte, professeur à l'Université Louis-Pasteur Strasbourg Jean-Michel Muller, directeur de recherche CNRS, LIP Robert Tijdeman, professeur à l'Université de Leiden.
Examinateurs :	Henri Cohen, professeur à l'Université Bordeaux 1 Maurice Mignotte, professeur à l'Université Louis-Pasteur Strasbourg Jean-Michel Muller, directeur de recherche CNRS, LIP Gérald Tenenbaum, professeur à l'Université Henri-Poincaré Nancy 1 Paul Zimmermann, directeur de recherche INRIA, LORIA.

Laboratoire Lorrain de Recherche en Informatique et ses Applications — UMR 7503

Mis en page avec la classe thloria.

Table des matières

Introduction

1 Préambule chronologique

Ce mémoire est pour moi l'occasion de faire le point sur environ dix ans d'exercice de la recherche, d'abord dans le cadre de mon mémoire de DEA, de ma thèse, puis comme chargé de recherche à l'INRIA Lorraine.

Mes travaux initiaux, lors de mon mémoire de DEA, étudiaient des fonctions à sens unique et générateurs pseudo-aléatoires de nature arithmétique, et ma thèse s'est intéressée principalement à des questions d'arithmétique des équations diophantiennes.

Après mon arrivée à l'INRIA, j'ai commencé à infléchir mes thématiques dans la direction d'un rapprochement avec les thèmes centraux du projet dans lequel je me trouvais à l'époque, et ai donc commencé à m'intéresser progressivement aux questions d'arithmétique des ordinateurs à l'époque.

Au fur et à mesure du temps, cette activité a gagné en importance progressivement, au détriment hélas de mes travaux sur l'algorithmique des équations diophantiennes.

Arrivé à ce point, il est probable que mes travaux s'enrichissent dans les années à venir d'une nouvelle thématique, l'algorithmique des courbes sur les corps finis, et leurs applications en cryptologie, question à laquelle je me suis intéressé, sans développer à titre personnel de travaux dans cette direction mais en suivant de près les développements, ainsi qu'en co-encadrant sur ce sujet, avec F. Morain, la thèse de Nicolas Gürel. Les récents mois ont en outre vu une intensification des contacts avec l'équipe de théorie des nombres, avec en particulier une collaboration avec G. Tenenbaum et J. Wu.

2 Plan du mémoire

Le mémoire suit, pour l'essentiel, la progression chronologique de mes travaux, et distingue les deux grandes thématiques évoquées ci-dessus. Le point commun qui pourrait rapprocher ces deux parties est une recherche de l'optimalité algorithmique, y compris dans les détails ne modifiant que la constante dans la complexité asymptotique, ou même dans le temps d'exécution pratique. Quand on parle d'un calcul de plusieurs mois, le facteur constant devient plus que significatif ; au-delà de cet argument, il s'agit en outre d'un goût personnel, et les résultats auxquels je suis le plus attaché sont ceux où se rencontrent une étude théorique, une étude algorithmique et une implantation délicates, et où chacun de ces aspects doit être traité au mieux pour aboutir au meilleur résultat possible. Je pense en particulier au problème des diviseurs primitifs, ou encore à l'équation de Nagell-Ljunggren.

La première partie décrit donc mes contributions aux méthodes transcendantes pour l'algorithmique des équations diophantiennes. Ces contributions se sont initialement concentrées sur l'équation de Thue et l'équation superelliptique, et les méthodes et points de vue développés se sont révélés ultérieurement riches d'applications que j'y décris.

En particulier, je suis frappé par le fait que le résultat présenté sur l'équation de Catalan soit issu, à l'origine, d'une réflexion et d'une problématique de nature algorithmique.

La seconde partie, quant à elle, décrit les deux grandes problématiques de l'arithmétique des ordinateurs auxquelles j'ai eu jusqu'alors l'occasion de m'intéresser, à savoir l'efficacité algorithmique (principalement en multi-précision) et la fiabilité du calcul flottant.

Ces deux parties, bien que largement indépendantes, ne sont toutefois pas dépourvues de points d'interconnexion. Les algorithmes pour les équations diophantiennes sont largement basés, in fine, sur la non-existence de relations linéaires entre certains nombres réels ; l'étude impose alors d'être capable d'effectuer parfois un grand nombre d'opérations à une très grande précision, et le résultat étant basé de façon extrêmement forte sur le calcul, la fiabilité est requise pour satisfaire l'exigence de rigueur mathématique.

À l'inverse, les techniques d'approximation diophantienne, qu'elle soit effective (fractions continues, algorithme LLL) ou théorique (bornes pour les formes linéaires de logarithmes, bornes pour $|e^u - v|$) me semblent prendre de plus en plus d'importance avec les années, et appelées à en prendre encore plus, dans la compréhension comme dans l'algorithmique des problèmes de pires cas.

À défaut d'une unité de résultats dans ce mémoire, le lecteur est donc invité à chercher, sinon une unité[1], du moins une parenté d'idées, de sensibilité ou d'outils qui pourra lui servir de fil conducteur.

[1] Le lecteur en trouvera bien plus d'une dans la première partie...

Chapitre 1

Algorithmes pour les équations diophantiennes

La théorie des nombres est une des branches les plus anciennes des mathématiques. Les algorithmes ont souvent été partie intégrante des travaux de grands arithméticiens comme Euler, Gauss ou encore Riemann, tout simplement parce que la nécessité d'expérimenter les obligeait à calculer, et l'absence de moyens de calcul efficaces les obligeait à calculer très intelligemment.

La période de la fin du XIX-ème et de la première moitié du XX-ème marque une pause dans cet intérêt ; elle coïncide avec le développement de méthodes et de théories sophistiquées, dont les arguments sont de plus fréquemment ineffectifs. Il faut attendre les années 1980 pour assister à une vraie renaissance de l'algorithmique en théorie des nombres, qui se donne simplement pour ambition de calculer – le plus efficacement possible – avec les objets introduits par la théorie. Souvent, cela impose un véritable travail théorique en amont, pour donner une description effective d'objets qui ne le sont pas nécessairement. Souvent aussi, cela impose de modifier les constructions mathématiques, pour sacrifier l'élégance mathématique à l'efficacité algorithmique.

Le sous-domaine de la théorie algorithmique des nombres auquel je me suis intéressé traite d'un certain type de méthodes pour la résolution d'équations diophantiennes. Les équations diophantiennes trouvent leur origine dans l'Antiquité, même si elles sont nommées en l'honneur de Diophante[2](325–409).

De manière élémentaire, une équation diophantienne est une équation dont les inconnues sont des nombres entiers, encore que selon les contextes on puisse réduire plus ou moins le champ de cette définition. Bien des recherches en théorie des nombres ont été, de près ou de loin, inspirées par la volonté de résoudre une équation diophantienne célèbre, typiquement l'équation de Fermat $x^n + y^n = z^n$, $n \geq 3$. Parmi les 23 problèmes posés par Hilbert à Paris en 1900 comme "défis" pour le siècle à venir, le $10^{\text{ème}}$ concerne la résolution des équations diophantiennes : Hilbert demandait de construire une méthode générale de résolution ; voici le texte exact du problème, tiré de ses œuvres complètes [38].

> 10. Entscheidung der Lösbarkeit einer diophantischen Gleichung.
>
> Eine diophantische Gleichung mit irgendwelchen Unbekannten und mit ganzen rationalen Zahlkoeffizienten sei vorgelegt : *man soll ein Verfahren angeben, nach welchen sich mittels einer endlichen Anzahl von Operationen entscheiden läßt, ob die Gleichung in ganzen rationalen Zahlen lösbar ist.*[3]

[2] à qui l'on doit le problème de "trouver un triangle rectangle dont la somme de l'aire et de l'hypothénuse soit un carré et dont le périmètre soit un cube", ce qui se ramène à résoudre dans $\mathbb{Q}$ l'équation $y^2 = x^3 + k$.

[3] (Traduction libre) 10. Décidabilité de la résolubilité des équations diophantiennes. Une équation diophantienne à un certain nombre d'inconnues et à coeffi cients entiers rationnels étant donnée, *on demande de trouver une méthode qui, au moyen d'un*

Hilbert se limite donc au cas d'équations polynomiales en nombres entiers, mais cela est déjà une question hélas aussi ambitieuse que vaine : des travaux de logiciens culminant dans le résultat de Matjasevitch [41] ont montré que l'existence d'une solution entière est, en toute généralité, indécidable : la méthode générale que demande Hilbert n'existe pas.

Il faut toutefois nuancer les conséquences de ce résultat : si le problème, en toute généralité, est indécidable, rien n'empêche que pour des familles d'équations des algorithmes existent. Et nous verrons que c'est effectivement le cas. En tout état de cause, un tour d'horizon rapide de l'existant limite rapidement les ambitions : je ne connais pas de méthode, actuellement, qui permette en toute rigueur de décider si une courbe $y^2 = q(x)$, avec q de degré 4, admet ou non un point rationnel sur $\mathbb{Q}$...

1.1 Taxonomie sommaire

On peut s'essayer à une taxonomie grossière des méthodes existantes.

La méthode le plus élémentaire (ne pas l'oublier, elle prouve l'absence de solutions dans bien des cas !) consiste en l'utilisation d'arguments de congruence. Ce type d'arguments permet, en exhibant un nombre premier p (ou, le cas échéant, un idéal premier d'un corps de nombres bien choisi dans un stade ultérieur de traitement de l'équation) modulo lequel l'équation n'a aucune solution, de prouver que l'équation n'a aucune solution dans $\mathbb{Z}$. Plus généralement, quand une équation a peu de solutions modulo des premiers (typiquement, pour une équation de la forme $y^p = f(x)$, on s'attend à ce que les solutions soient peu nombreuses modulo $\ell = 1 \bmod p$, car les puissances p-èmes modulo ℓ sont alors en proportion $1/p$), on peut utiliser le théorème chinois pour construire efficacement toutes les solutions potentielles inférieures à une borne donnée ; on atteint toutefois assez vite les limites de ce type de méthode. En particulier, elle échoue toujours dans le cas où il existe une solution, même triviale.

Certaines équations sont justiciables de techniques relevant purement de l'arithmétique élémentaire (pgcd, fractions continues, algèbre linéaire entière), telles les équations linéaires (qui sont complètement résolues par une mise sous forme normale de Smith) ou l'équation de Pell $x^2 - Dy^2 = 1$.

Des méthodes plus sophistiquées, que nous explorerons dans ce texte, utilisent des arguments de transcendance. Nous nous concentrerons sur un aspect, dont l'algorithmique est bien établie, qui s'appuie sur la théorie des bornes inférieures pour les formes linéaires de logarithmes de nombres algébriques. Plus généralement, l'idée de ce type de méthodes est que les nombres algébriques jouissent de propriétés arithmétiques fortes (par exemple, le théorème de Roth [47] affirme qu'ils sont mal approchés par des rationnels). Certains de ces résultats sont de nature effective, et fournissent dès lors des informations sur le nombre, ou – mieux – la taille des solutions potentielles de l'équation. Souvent, ces informations sont difficiles à exploiter (information de nombre peu exploitable, ou information de taille insuffisamment précise pour permettre l'énumération), mais nous verrons comment, dans certains cas, une combinaison d'ingrédients algorithmiques et diophantiens permet de résoudre les problèmes.

Enfin, les méthodes les plus avancées reposent sur l'utilisation de techniques issues de la géométrie arithmétique. La méthode la plus explorée dans cette direction est sans doute la méthode de Chabauty (mentionnons aussi la méthode de Demjanenko), voir par exemple [29]. À une courbe algébrique, la géométrie arithmétique permet d'associer une variété algébrique munie naturellement d'une structure de groupe – sa jacobienne – ; le groupe des points sur $\mathbb{Q}$ est alors de type fini. Quand, de surcroît, le rang de la partie libre dudit groupe est plus petit que sa dimension, un argument dû à Chabauty prouve de façon effective la finitude du nombre de solutions. La restriction sur le rang du groupe n'est pas bénigne, mais diverses techniques permettent souvent de la contourner (voir e.g. [18]. Ces techniques sont sans doutes

nombre fini d'opérations, permet de dire si l'équation est résoluble en entiers rationnels.

les plus puissantes, et d'un point de vue algébrique et géométrique, les plus intrinsèques et élégantes. Elles nous emmèneraient toutefois trop loin, et nous ne les discuterons pas.

Cette taxonomie ne serait pas complète sans une mention des récentes méthodes *modulaires*, utilisant l'angle d'attaque qui a permis la résolution de l'équation de Fermat. Elles sont essentiellement limitées au cas d'équations ternaires, de type $A+B=C$, où A, B et C sont des termes monomiaux. Le principe consiste à former la courbe elliptique $y^2 = x(x-A)(x+B)$; une étude arithmétique des propriétés de cette courbe permet de lui associer une *newform* dont le niveau est contrôlé. Dans certains cas il n'existe pas de *newform* de ce niveau, auquel cas on a prouvé que l'équation n'a pas de solution. Dans le cas contraire, des techniques complémentaires fournissent parfois d'autres contraintes sur la *newform*. On pourra consulter, par exemple, [8].

1.2 Méthodes transcendantes

1.2.1 Borne de Baker

Pour l'algorithmique des équations diophantiennes, l'une des voies de « salut » vient d'un point de vue exploré par Gelfond et Schneider pour l'étude du 13-ème problème de Hilbert, puis généralisé par A. Baker dans les années 1960 dans une série d'articles, par exemple [1, 2].

Le procédé consiste, via une étude algébrique de l'équation, à construire une *forme linéaire de logarithmes*

$$\Lambda(b_1,\dots,b_n) = \sum_{i=1}^{r} b_i \operatorname{Log} u_i \tag{1.1}$$

où les quantités liées aux inconnues initiales sont les b_i, les autres quantités étant explicites et ne dépendant que de l'équation. Ici et dans la suite, Log est la détermination principale du logarithme complexe.

La quantité $\Lambda(b_1,\dots,b_n)$ doit de plus avoir la propriété que pour toute solution de l'équation initiale, on doit pouvoir construire un n-uplet $(b_1,\dots,b_n)$ tel que $\Lambda(b_1,\dots,b_n)$ soit très petit, typiquement :

$$|\Lambda(b_1,\dots,b_n)| \ll \exp(-C\cdot \max_i |b_i|) \tag{1.2}$$

Nous verrons que toute majoration décroissant plus vite qu'une puissance fixée de $\max_i |b_i|$ (puissance dépendant de l'équation) est suffisante, mais pour l'efficacité algorithmique il est souhaitable que la borne supérieure décroisse le plus vite possible.

La forme Λ (on parle de forme linéaire en logarithmes) est en général construite en prenant le logarithme d'une quantité exponentiellement proche de 1 (quand $\max|b_i| \to \infty$). On va donc rencontrer naturellement ce type de méthodes lors de l'étude d'équations diophantiennes qui « cachent » une structure de groupe multiplicatif de type fini.

Intuitivement, une propriété comme $|\Lambda(b_1,\dots,b_n)| \ll \exp(-C\cdot\max_i|b_i|)$ a un caractère exceptionnel. L'équivalent n-dimensionnel du théorème de Dirichlet – conséquence aisée du principe des tiroirs – prédit que parmi les N^n valeurs de $|\sum_{i=1}^n b_i\alpha_i|$ pour $(b_i)\in[1,N]^n$ qui se trouvent dans $[0, N\sum_{i=1}^n|\alpha_i|]$, on peut en trouver deux distantes d'au plus $N^{1-n}\sum_{i=1}^n|\alpha_i|$, et donc une combinaison linéaire de cet ordre de grandeur avec $|b_i|\le 2N$. Cette estimation est en fait essentiellement optimale.

Cet argument n'est bien entendu pas suffisant... car on peut bien entendu, inversement, construire des familles (α_i) admettant des combinaisons linéaires bien plus petites. Mais dans le cas présent, nos nombres α_i sont bien particuliers ; ce sont des logarithmes de nombres algébriques. Dans ce cas, le résultat de Baker (qui lui a valu la médaille Fields en 1966), largement précisé, raffiné, et étendu depuis, nous fournit l'estimation dont nous avons besoin :

Théorème 1 *Soit $u_1,\dots,u_n$ des nombres algébriques. Il existe une constante $C(u_1,\dots,u_n)$ telle que, pour tout n-uplet $(b_1,\dots,b_n)$, si l'on pose $\Lambda(b_1,\dots,b_n)=\sum b_i \mathrm{Log}\, u_i$, on a*

- *Soit $\Lambda(b_1,\dots,b_n)=0$ (et les u_i sont multiplicativement dépendants) ;*
- *Soit $\Lambda(b_1,\dots,b_n)>\exp(-C(u_1,\dots,u_n)\log\max|b_i|)$, où $C(u_1,\dots,u_n)$ est une constante positive ne dépendant que des u_i.*

On peut comparer ce résultat, dans le cas où les u_i sont des nombres entiers, à la méthode de Liouville. Dans ce cas, Λ est le logarithme d'un nombre $\prod u_i^{b_i}$; si ce nombre est différent de 1, sa distance à 1 est alors au moins égale à l'inverse de son dénominateur, qui est au plus $\prod |u_i|^{|b_i|} \leq (\prod |u_i|)^{\max|b_i|}$. En particulier, l'estimation qui est obtenue pour Λ est alors $\exp((\sum \log|u_i|)\max|b_i|)$, dans laquelle la dépendance en $B:=\max|b_i|$ est bien moins bonne que précédemment. En revanche, la dépendance en les $\log|u_i|$ est, elle, meilleure que ce que l'on obtient : il va falloir, en général, remplacer la somme par un produit.

L'étude détaillée de cet exemple montre que si $u_i=p_i/q_i$ devient un rationnel, il faut remplacer $\log|u_i|$ par $\log\max(|p_i|,|q_i|)$. La généralisation de cette quantité pour un nombre algébrique est la *hauteur logarithmique absolue* :

Définition 1 *Soit α un nombre algébrique de degré d, $P=a_0x^n+\sum_{d=0}^{n-1}a_{n-d}x^d\in\mathbb{Z}[X]$ son polynôme minimal, $\alpha_1,\dots,\alpha_n$ ses racines. La hauteur logarithmique absolue de α est*

$$\frac{1}{n}\cdot\log\left(|a_0|\prod_{i=1}^{n}\max(1,|\alpha_i|)\right).$$

Nous disposons maintenant de tous les éléments pour énoncer le théorème de Matveev [42] :

Théorème 2 *Soient $u_1,\dots,u_n$ des nombres algébriques (distincts de 0 et 1), $D:=[\mathbb{Q}(u_1,\dots,u_n):\mathbb{Q}]$, et $A_i=\max\{Dh(\alpha_i),|\log\alpha_i|,0.16\}$. Alors soit $\Lambda(b_1,\dots,b_n)=0$, soit*

$$\Lambda(b_1,\dots,b_n)\geq\exp\left(-D^2\max\left(2^{6n+20},\frac{1}{\kappa}(en/2)^{\kappa}30^{n+3}n^{7/2}\right)A_1\dots A_n\log(eD)\log(eB)\right),$$

où $\kappa=1$ si tous les u_i sont réels, 2 sinon.

Dans le cas où l'on n'a que deux logarithmes, on dispose également du résultat suivant dû à Laurent, Mignotte et Nesterenko[39] :

Théorème 3 *Soit α_1, α_2 deux nombres algébriques, et $\Lambda=b_1\log\alpha_1-b_2\log\alpha_2$. On pose*

$$D=[\mathbb{Q}(\alpha_1,\alpha_2):\mathbb{Q}]/[\mathbb{R}(\alpha_1,\alpha_2:\mathbb{R})],$$

et on suppose donnés des réels h_i tels que

$$h_i\geq\max(h(\alpha_i),\log\alpha_i/D,1/D).$$

Alors, en posant $b=b_1/h_2+b_2/h_1$, on a

$$\log|\Lambda|\geq-30.9D^4\max(\log b,21/D,1/2)^2h_1h_2.$$

La dépendance en $\log\max|b_i|$ est un peu moins bonne que précédemment, mais ceci est, pour notre usage, largement compensé par une constante nettement plus petite...

1.2.2 Réduction de la borne

Hélas, hormis dans le cas favorable où l'on parvient à utiliser une forme linéaire en deux logarithmes, les constantes impliquées dans le théorème ci-dessus sont tellement gigantesques que les bornes obtenues pour les b_i rendent impossible une énumération exhaustive. En outre, si le nombre de variables est grand, même une valeur modérée peut déjà rendre difficile une énumération (avec 10 variables et une borne de 10 sur les $|b_i|$, on a déjà 21^{10} possibilités à vérifier).

C'est pour résoudre ce problème que les idées de réduction de la borne ont été introduites. Il est important de bien comprendre qu'elles visent par essence à prouver qu'il n'existe pas de grande solution, même s'il est possible de les adapter pour contourner une éventuelle « grande » solution, une fois celle-ci détectée. À ce titre, un échec dans la mise en œuvre pratique doit souvent se lire comme signifiant qu'il existe une telle solution. Toutefois, il convient de remarquer que comme le changement de variables conduisant des inconnues initiales aux b_i est logarithmique dans les cas qui nous intéresse, une grande solution en b_i correspondrait à une solution gigantesque sur les inconnues de départ, situation dont je ne connais pas d'exemple.

Le principe de la réduction est la suivante. Oublions la nature arithmétique des coefficients de la forme linéaire de logarithmes, qui ne joue plus aucun rôle. La borne de Baker nous ramène à résoudre une inégalité du type :

$$\left|\sum_{i=1}^{n} b_i x_i\right| \leq \exp\left(-c \max_i |b_i|\right), \tag{1.3}$$

pour une certaine constante c, avec les b_i bornés.

L'ingrédient-clé est alors un procédé permettant de minorer

$$\min_{\substack{|b_i|\leq B \\ (b_i)_{1\leq i\leq n}\neq 0}} \left|\sum_{i=1}^{n} b_i x_i\right|.$$

Si $n = 2$, une telle minoration provient de façon directe de la théorie des fractions continues. La première occurrence d'un tel problème remonte à Baker et Davenport [4] , qui ont traité le cas $n = 3$, mais dans une situation inhomogène, i.e. pour $b_3 = 1$. Dans ce cas encore, on peut résoudre le problème en utilisant le développement en fraction continue de x_2/x_1.

Plus généralement, l'estimation de Baker et Davenport se généralise, dans la situation inhomogène, au cas où l'on connaît[4] un entier Q tel que Qx_i soit très voisin d'un entier pour $1 \leq i \leq n$. Dans ce cas, en effet,

$$\begin{aligned}\left|\sum_{i=1}^{n} b_i x_i + x_{n+1}\right| &= Q^{-1}\left|Qx_{n+1} + \sum_{i=1}^{n} Qb_i x_i\right| \\ &\geq Q^{-1}\left\{d(Qx_{n+1},\mathbb{Z}) - \sum_{i=1}^{n} Bd(Qx_i,\mathbb{Z})\right\},\end{aligned}$$

ce qui, dans le cas où x_n est « indépendant » des autres termes, fournit heuristiquement une borne inférieure de l'ordre de $1/Q$, qui est l'ordre de grandeur auquel on peut s'attendre en utilisant là encore le principe des tiroirs.

Une solution plus intrinsèque est fournie par l'algorithme LLL, qui, bien qu'initialement introduit pour résoudre des problèmes de nature très différente, s'adapte tout à fait à la situation présente.

[4] Un tel entier peut se trouver, par exemple, au moyen de l'algorithme LLL.

Une application classique de l'algorithme LLL est, étant donné n vecteurs $v_1,\dots,v_n \in \mathbb{R}^m$, de trouver un minorant assez précis pour $\|\sum a_i v_i\|_2$, $(a_i) \in \mathbb{Z}^m - \{0\}$. Utilisant l'équivalence des normes, on peut en déduire un minorant pour les combinaisons linéaires en norme 1. Dans notre cas, l'hypothèse selon laquelle les coefficients sont bornés est toutefois cruciale pour déduire une information sur la forme linéaire $\sum b_i x_i$, comme le montre le lemme suivant :

Lemme 1 *Soit $(b_i)_{1\le i\le n}$ des entiers, $(x_i)_{1\le i\le n}$ des réels, $B = \max_i |b_i|$.*

On suppose que le vecteur renvoyé par LLL sur le réseau engendré par les colonnes de la matrice

$$M(x_1,\dots,x_n,C) = \begin{pmatrix} 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \\ \lfloor Cx_1 \rceil & \lfloor Cx_2 \rceil & \dots & \lfloor Cx_{n-1} \rceil & \lfloor Cx_n \rceil \end{pmatrix}$$

a longueur l_0, avec

$$l_0 \ge 2^{(n-1)/2} B \sqrt{(n^2/4 + n - 1)}.$$

Alors pour tout n-uplet d'entiers $(b_1,\dots,b_n)$, on a

$$\left| \sum_{i=1}^{n} b_i x_i \right| \ge \frac{1}{C} \left(\sqrt{2^{1-n} l_0^2 - (n-1)B^2} - \frac{nB}{2} \right).$$

On peut généraliser ce résultat au cas d'une forme inhomogène, c'est-à-dire du cas où l'on peut imposer qu'une des variables est égale à 1. On peut bien entendu traiter ce cas au moyen du lemme précédent (en oubliant l'information qu'un des coefficients vaut 1), mais la version suivante est plus adaptée à ce cas, et fournit des résultats plus précis.

Lemme 2 ([56]) *Soit $x = (x_i)$ un vecteur de $\mathbb{Z}^n$, et $A = (v_1,\dots,v_n)$ une base LLL-réduite d'un réseau Λ. Posons $s = (s_i) = A^{-1}(x_i)$. Alors*

$$d(x,\Lambda) \ge 2^{(1-n)/2} d(s_{i^*},\mathbb{Z}) \|v_1\|_2,$$

où i^* est le plus grand entier i tel que $s_i \notin \mathbb{Z}$.

1.2.3 Aspects heuristiques de la réduction

Même s'il est impossible d'anticiper le résultat de la réduction, et donc d'offrir une analyse précise du processus, dans la mesure où celui-ci est au moins tributaire de l'existence d'une grande solution, on peut toutefois en faire une analyse heuristique dans les deux cas.

Dans le cas du lemme de Baker-Davenport, la nouvelle borne que nous allons obtenir est, au vu de (1.4), de l'ordre de grandeur de $O(\log Q)$.

Pour que ce lemme s'applique, il faut choisir Q de façon à ce que $d(Qx_i,\mathbb{Z})$ soit petit devant $1/nB$ pour $i \in [1,n]$, ce qui, au vu du principe des tiroirs, va nous amener à prendre Q plus grand que $(nB)^n$. Si nous supposons alors que $d(Qx_{n+1},\mathbb{Z})$ est suffisamment grand, ce qui s'obtient généralement aisément en augmentant légèrement Q si nécessaire, on voit que la nouvelle borne est essentiellement $O(n \log B)$.

Dans le cas de LLL, la situation est très similaire. La borne inférieure fournie par LLL pour le vecteur le plus court d'un réseau L est heuristiquement de l'ordre de $(\det L)^{1/\dim L}$, soit, dans le cas présent, de

$C^{1/n}$. Par suite, pour que le procédé fonctionne, il nous faut choisir $C^{1/n} > nB$, soit $C > (nB)^n$. La borne attendue est alors de l'ordre de $\log C$, soit de nouveau $n \log B$.

Il importe de remarquer que, qu'il s'agisse du lemme de Baker-Davenport ou de l'application de LLL, il nous faudra disposer d'une approximation numérique des x_i permettant d'estimer $[Cx_i]$ ou $d(Cx_i, \mathbb{Z})$, avec C de l'ordre de $(nB)^n$. Cela implique, en particulier, que les données x_i doivent être calculées avec une précision potentiellement colossale.

Nous verrons que cette précision peut être réduite jusqu'à être rendue finalement assez proche de B. On peut noter en outre que la qualité de la borne réduite est, heuristiquement, directement liée à la précision requise (puisque la borne réduite est d'un ordre de grandeur attendu $-\log C$).

1.3 L'équation de Thue

Dans cette partie, nous commencerons par présenter la théorie générale de l'équation de Thue, développée par Thue, Baker, et bien d'autres, puis l'abord algorithmique classique [53] ; je présenterai ensuite mes différentes contributions à l'amélioration de l'algorithmique de cette équation.

On appelle équation de Thue une équation de la forme

$$P(x,y) = a, \tag{1.4}$$

avec P homogène, irréductible, de degré ≥ 3. Il est bon de noter que le cas du degré 2 est le cas de l'équation de Pell, que le cas du degré 1 est une recherche de coefficients de Bezout ; enfin, que si le polynôme $P = P_1 P_2$ est réductible, on trouve un nombre fini de systèmes $P_1(x,y) = a_1, P_2(x,y) = a_2$, qui se résolvent de façon banale par élimination, par exemple par un calcul de résultant. Le seul cas posant encore problème est alors le cas où P est une puissance pure, auquel cas on se ramène à une nouvelle équation de type Thue.

1.3.1 Résultats de finitude et bornes

L'équation de Thue intervient de manière naturelle comme équation auxiliaire lors de la résolution de nombre de problèmes. On trouvera deux illustrations dans ce mémoire, voir *infra* les parties 1.6.2 et 1.6.3. Elle intervient aussi de façon naturelle (quoique sous forme relative, où les techniques sont un peu différentes) dans la résolution des équations superelliptiques.

On sait depuis Thue [52] que cette équation n'a qu'un nombre fini de solutions. Divers résultats quantitatifs ont permis de préciser cet énoncé. On a en particulier, de façon générale, les deux énoncés suivants :

Théorème 4 (Bombieri-Schmidt [8]) *Il existe une constante c telle que pour $\deg P \geq c$, l'équation*

$$F(x,y) = a$$

a au plus $215(\deg P)^{1+\omega(a)}$ solutions avec $(x,y) = 1$, où l'on identifie éventuellement les deux solutions (x,y) et $(-x,-y)$.

Théorème 5 (Bugeaud-Győry [21]) *Soit α une racine de $P(X,1)$. Notons r le rang du groupe des unités du corps de nombres $\mathbb{Q}(\alpha)$, R son régulateur, $A = \max(|a|, e)$ et H la hauteur du polynôme F.*

Toutes les solutions de l'équation de Thue $F(x,y) = a$ vérifient

$$\max(|x|,|y|) \leq \exp\left\{3^{r+27}(r+1)^{7r+19} n^{2n+6r+14} R \log^* R(R + \log(HA))\right\},$$

et

$$\max(|x|,|y|) \leq \exp\left\{3^{n+9} n^{18(n+1)} H^{2n-2} (\log H)^{2n-1} \log A\right\}.$$

Ces résultats sont très généraux, et peuvent être précisés si l'on restreint la classes des équations considérées.

1.3.2 Une unité

Dans un souci de simplification, on supposera P unitaire en X, ce que l'on peut toujours faire en pratique, quitte à effectuer un changement de variable (bien choisi de façon à conserver le caractère entier des racines ; on ajoutera souvent ce faisant des solutions parasites qui sont éliminées par vérification *a posteriori*). D'un point de vue algébrique, si l'on prend α une racine du polynôme P, et que l'on écrit $\alpha_1, \ldots, \alpha_n$ les différents conjugués de α, on peut récrire l'équation sous la forme

$$\prod_{i=1}^{n}(X - Y\alpha_i) = a \tag{1.5}$$

soit encore $N_{\mathbb{Q}(\alpha)/\mathbb{Q}}(X - \alpha Y) = a$.

Dans la suite, on notera σ_i le plongement de $\mathbb{Q}(\alpha)$ dans $\mathbb{C}$ qui envoie α sur α_i. On note en outre s le nombre de α_i réels, $2t$ le nombre de $\alpha_i \in \mathbb{C} - \mathbb{R}$, et on suppose les α_i ordonnés en commençant par les réels.

L'approche naturelle, exposée formellement par exemple dans [53] consiste à utiliser le fait que dans l'anneau des entiers d'un corps de nombres, l'équation

$$N_{\mathbb{K}/\mathbb{Q}}(\xi) = a,$$

a étant un entier fixé, n'a qu'un nombre fini de solutions modulo l'action du groupe des unités. Nous noterons Ξ cet ensemble de solutions.

Résoudre l'équation de Thue revient donc à chercher les x, y entiers et les unités u tels que, pour un certain $\xi \in \Xi$, on ait

$$u\xi = x - \alpha y,$$

ou encore, en utilisant un système d'unités fondamentales $\eta_1, \ldots, \eta_r$ (et un générateur de la torsion ζ) que nous supposerons connus,

$$\xi\,\zeta^{b_0}\eta_1^{b_1}\ldots\eta_r^{b_r} = x - \alpha y, \tag{1.6}$$

où les b_i sont des entiers relatifs. Les b_i sont nos inconnues exponentielles, celles qui vont apparaître dans la forme linéaire de logarithmes.

1.3.3 L'inégalité fondamentale

Il nous reste, pour pouvoir appliquer la machinerie générale, à construire une quantité voisine de 1, de façon à obtenir une forme en logarithmes voisine de 0. Pour ce faire, il nous faut contrôler les ordres de grandeur, ce qui va se faire via une observation importante sur l'équation initiale. En substance, l'équation sous la forme (1.5) montre qu'on a un produit de n termes de la forme $X - \alpha Y$ de taille bornée. Comme au plus un de ces termes peut être petit (car, alors, $X \approx \alpha_k Y$ et pour $j \neq k$, $X - \alpha_j Y \approx Y(\alpha_k - \alpha_j)$ est de l'ordre de grandeur de Y), il doit être très petit.

Proposition 1 *On pose* $f(X) = P(X,1)$. *Soit*

$$Y_0 = \begin{cases} \left(\dfrac{2^{n-1}|a|}{\min_{1\le i\le t}|f'(\alpha_{s+i})|\cdot\min_{1\le i\le t}|\mathrm{Im}\ \alpha_{s+i}|}\right)^{1/n} & \text{si} \quad t \ge 1, \\ 1 & \text{si} \quad t = 0, \end{cases}$$

$$c_1 = \frac{2^{n-1}|a|}{\min_{1\le i\le s}|f'(\alpha_i)|}$$

Soit (x,y) une solution entière de (1.4).

Si $|y| > Y_0$ alors, pour un $i_0 \in \{1,\dots,s\}$ on a

$$|x - \alpha_{i_0} y| \leq \frac{c_1}{|y|^{n-1}}. \tag{1.7}$$

De nouveau, cet énoncé est classique, et combiné au résultat de Thue sur l'approximation des algébriques par des rationnels, permet d'établir un résultat de finitude sur le nombre de solutions. Il permet également de constater que le cas où le corps de base est totalement imaginaire est essentiellement trivial, dans la mesure où l'on obtient une borne très précise sur les solutions. Il permet, enfin, de constater qu'il est relativement facile d'énumérer les petites solutions, dans la mesure où l'on peut déduire que, sauf pour de très petits y, les solutions x/y seront des réduites du développement en fractions continues d'un des α_i.

À ce point, il faut ajouter une nouvelle énumération à celle des différentes solutions des équations aux normes, i.e., il faut faire ce qui suit pour toutes les racines réelles de l'équation de départ, de façon à énumérer tous les choix possibles pour i_0. Dans la suite, on supposera sans perte de généralité que $\alpha = \alpha_1$ est la racine telle que $X - \alpha Y$ est petit.

1.3.4 Une quantité voisine de 1

Il nous devient aisé de construire une quantité de type exponentielle qui soit voisine de 1 : dans la mesure où $X - \alpha_j Y$ est très proche de $(\alpha - \alpha_j)Y$, il vient que $(X - \alpha_j Y)/(\alpha - \alpha_j)$ est très proche de Y, et donc que

$$\frac{X - \alpha_j Y}{X - \alpha_k Y} \cdot \frac{\alpha - \alpha_k}{\alpha - \alpha_j}$$

est très proche de 1. Plus précisément, on a alors :

Proposition 2 *On pose*

$$c_2 = \min_{1 \leq i < j \leq n} \left|\alpha_i - \alpha_j\right|,$$

$$c_3 = 1.39 c_1 c_2^{-1},$$

$$X_1 = \max\left(X_0, \left(2c_1 c_2^{-1}\right)^{1/n}\right).$$

Alors si $j,k \neq i_0$, on a

$$\left|\operatorname{Log} \frac{X - \alpha_j Y}{X - \alpha_k Y} \cdot \frac{\alpha - \alpha_k}{\alpha - \alpha_j}\right| \leq \frac{2c_3}{|Y|^n}$$

Enfin,

$$\left|\operatorname{Log} \frac{(X - \alpha_{i_0} Y) f'(\alpha_{i_0})}{|a| Y^{n-1}}\right| \leq \frac{(n-1)c_3}{|Y|^n}.$$

En outre, il n'est pas très difficile, en utilisant l'identité (1.6) et la proposition, de prouver que $\log|Y| \asymp \max_i |b_i|$. On en déduit l'inégalité

$$\left|\sum_{i=1}^{r} b_i \operatorname{Log} \frac{\sigma_j(\eta_i)}{\sigma_k(\eta_i)} + b_{r+1} i\pi + \operatorname{Log} \frac{\alpha - \alpha_k}{\alpha - \alpha_j}\right| \ll \exp(-c \max_i |b_i|),$$

Notons que l'on peut prendre le logarithme des modules et enlever le facteur $i\pi$ dès lors que $r > 1$.

L'application d'une borne de type Baker permet alors d'obtenir la borne souhaitée, qui est généralement immense. Par exemple, pour l'équation $x^3 - 2y^3 = 1$, on obtient $\max|b_i| \leq 10^{19}$.

1.3.5 Réduction

Le coeur de l'algorithmique des équations diophantiennes réside dans les idées de *réduction de la borne*. Comme nous l'avons vu précédemment, les bornes obtenues lors de la première phase sont colossales, et l'utilisation de ce procédé est donc indispensable. L'introduction de LLL pour cette tâche est due à Tzanakis et de Weger [53], et elle s'est révélée extrêmement efficace. Toutefois, quand la dimension augmente significativement, les vecteurs de la matrice de réduction contiennent des entiers colossaux, et LLL devient extrêmement inefficace, au point d'être une des étapes limitantes de l'algorithme.

Diminuer la dimension et la précision

Cette partie expose essentiellement le résultat principal de l'article [13], travail commun avec Yuri Bilu.

Il existe un moyen de contourner cette difficulté, qui est au coeur de l'article [13]. Dans le cas de l'équation de Thue, on dispose non pas d'une inégalité linéaire en r variables du type 1.3, mais de $r-1$ telles inégalités, obtenues par conjugaison. Il s'ensuit que toute combinaison linéaire des formes linéaires correspondantes vérifie une inégalité du même type ; en choisissant une combinaison *ad hoc*, il est possible d'éliminer $r-2$ variables, pour obtenir une inégalité en trois variables. Plus formellement, on a le lemme suivant :

Lemme 3 *Soit $A = [a_{ij}]_{1\leq i,j\leq r}$ l'inverse de la matrice*

$$\left[\log\left|\sigma_{i+1}(\eta_j)\right|\right]_{1\leq i,j\leq r} \tag{1.8}$$

Pour $1 \leq i \leq r$, on pose

$$\begin{aligned}\delta_i &= \sum_{j=1}^{r} a_{ij}\,,\\ \lambda_i &= \sum_{j=1}^{r} a_{ij}\log\left|\frac{\alpha-\alpha_{j+1}}{\mu_{j+1}}\right|\end{aligned}$$

Alors il existe des constantes explicites c, c', c'', Y_2 telles que, pour tout $i_1 \neq i_2$ et pour toute solution (x,y) avec $|y| > Y_2$,

$$|\delta_{i_1}b_{i_2} - \delta_{i_2}b_{i_1} + \delta_{i_2}\lambda_{i_1} - \delta_{i_1}\lambda_{i_2}| \leq \frac{c}{|y|^n} \leq c'\exp(-c''\max_i|b_i|). \tag{1.9}$$

Nous sommes donc ramenés à la situation $n=3$ inhomogène, soit au lemme de Baker-Davenport, et un simple calcul de fractions continues suffit pour conclure. En outre, en vertu de l'analyse heuristique, nous n'avons plus besoin que de connaître les grandeurs x_i avec une précision de l'ordre de grandeur de $2\log B$ chiffres, au lieu de $r\log B$.

De limitante, la réduction devient instantanée, et la seule étape limitante restante réside dans la détermination du système d'unités, indispensable pour l'étude algébrique.

Ultérieurement, Bennett et de Weger [9] ont montré que l'on pouvait encore améliorer le processus de réduction en utilisant les $r-1$ formes inhomogènes indépendantes en deux variables que l'on peut tirer des $r-1$ formes en r variables ; on se retrouve alors de nouveau avec une réduction de type LLL, la précision descend à $(1+1/(r-1))\log B$, et la borne réduite s'en trouve donc *ipso facto* un peu meilleure.

On peut en outre tirer profit du lemme suivant pour améliorer la phase d'énumération ; ce lemme se déduit facilement de (1.9) :

Lemme 4 *Il existe une constante explicite Y_3 telle que pour toute solution (x,y) avec $|y| > Y_3$ on a :*

$$b_i = \lfloor \delta_i/\delta_1 b_1 - \delta_i\lambda_1/\delta_1 + \lambda_i \rceil.$$

La complexité de la dernière phase est donc réduite de $O(B_{\text{fin}}^r)$ à $O(rB_{\text{fin}})$.

Cette technique de réduction est maintenant utilisée dans tous les logiciels offrant des fonctions de résolution d'équations de Thue. Je l'ai en particulier implantée (ainsi que le raffinement décrit dans la section suivante) dans le système Pari/GP.

1.3.6 Un raffinement algébrique

La présente section expose un résultat qui a fait l'objet de l'article [31].

Cette nouvelle méthode permettant d'augmenter significativement le degré des équations pouvant être traités a conduit Bilu et moi-même à rechercher des exemples susceptibles d'illustrer notre méthode, ainsi que des applications arithmétiques de cette méthode.

Calculs d'unités

Nous avons hélas très vite buté sur une difficulté : le système d'unités fondamentales dont la méthode requiert la connaissance est généralement difficile à déterminer. Il l'est d'autant plus que l'on requiert qu'il soit effectivement *fondamental* (i.e. générateur du groupe des unités $\mathscr{U}$), plutôt que simplement de rang maximal (i.e., générateur de l'espace vectoriel $\mathscr{U} \otimes_{\mathbb{Z}} \mathbb{Q}$).

Pour expliquer pourquoi cela est le cas, rappelons brièvement la méthode de détermination des unités et du groupe des classes.

On cherche en fait à construire une présentation du groupe des classes par générateurs et relations, c'est-à-dire une suite exacte

$$0 \to \Lambda \to \mathbb{Z}^n \to \mathrm{Cl}_{\mathbb{K}} \to 0.$$

À cette fin, on considère un certain nombre n d'idéaux dont on sait qu'ils engendrent le groupe des classes, et l'on recherche des relations entre ces idéaux, c'est-à-dire des produits de puissances de ces idéaux qui soient des idéaux principaux.

Le groupe des classes peut être engendré par les idéaux de norme assez petite, plus petite que la borne de Minkowski, qui dépend essentiellement de la racine carrée du discriminant. Il est déraisonnable d'espérer énumérer tous les idéaux de $\mathbb{K}$ de norme inférieure à cette borne. Toutefois, d'après un résultat de Bach [3], le groupe des classes peut être engendré par les idéaux de norme plus petite que $12\log^2|D|$ (12 peut être remplacé par 6 dans le cas des corps quadratiques), sous l'hypothèse de Riemann généralisée.

Notons Λ le réseau des relations obtenu. On s'arrête quand le rang de Λ atteint n. On a alors obtenu un groupe fini $\mathbb{Z}^n/\Lambda$ dont l'ordre (qui s'obtient en calculant le déterminant de la matrice dont les colonnes constituent une base du réseau Λ) est un multiple $h'(\mathbb{K})$ du nombre de classes $h(\mathbb{K})$.

Maintenant, si à chaque relation on associe le générateur de l'idéal principal correspondant, ou plutôt son plongement logarithmique, on peut déduire de manière analogue le groupe des unités. Lorsque l'on met la matrice des relations sous forme normale d'Hermite, on va obtenir un certain nombre de relations triviales, c'est-à-dire représentant l'idéal $\mathbb{Z}_{\mathbb{K}}$. Mais si les manipulations effectuées sur la matrice des relations ont aussi été appliquées à la matrice des plongements logarithmiques, on dispose alors d'un générateur de cet idéal, c'est-à-dire d'une unité. De cette manière, avec assez de relations, on obtient un système d'unités de rang maximal.

Si l'on veut s'assurer que le système obtenu est un système fondamental, on peut maintenant calculer le régulateur du système obtenu, qui est un multiple $R'_{\mathbb{K}}$ du régulateur $R_{\mathbb{K}}$ du corps. On calcule alors le produit

$$h(\mathbb{K})R_{\mathbb{K}} = \frac{w(\mathbb{K})\sqrt{|D(\mathbb{K})|}}{2^s(2\pi)^t}\prod_p \frac{1-\frac{1}{p}}{\prod_{\mathfrak{p}|p}\left(1-\frac{1}{N\mathfrak{p}}\right)},$$

où $w(\mathbb{K})$ et $D(\mathbb{K})$ sont respectivement le nombre de racines de l'unité et le discriminant de $\mathbb{K}$.

Sous l'hypothèse de Riemann généralisée (que l'on abrégera dans la suite en "GRH"), on peut tronquer le produit eulérien en se limitant aux $\mathfrak{p}$ avec $N\mathfrak{p} < C\log^2|D(\mathbb{K})|$ (où C est encore 6 ou 12 suivant que le corps est quadratique ou non), et obtenir un nombre z avec

$$\frac{h(\mathbb{K})R_{\mathbb{K}}}{\sqrt{2}} < z < \sqrt{2}h(\mathbb{K})R_{\mathbb{K}}.$$

Dès lors, si $h'(\mathbb{K})R'_{\mathbb{K}} \leq z\sqrt{2}$, on a obtenu un système fondamental, sous l'hypothèse de Riemann pour $\mathbb{K}$; sinon, on recalcule de nouvelles relations.

Cette méthode permet donc en pratique de trouver rapidement un système d'unités dont on est sûr qu'il est de rang maximal (il suffit de calculer un déterminant pour s'en convaincre) mais qui n'est certifié fondamental que sous l'hypothèse de Riemann généralisée. Des méthodes de certification existent, qui permettent de garantir la validité du résultat, mais elles sont très lentes dès que le régulateur et/ou le degré du corps augmente.

Pour obtenir des solutions inconditionnelles à l'équation de Thue, il est donc *a priori* indispensable d'utiliser un système d'unités fondamentales dans la méthode exposée précédemment. Toutefois, une idée simple permet de s'affranchir de cette contrainte : un système de rang maximal suffit à engendrer le groupe des unités, si l'on admet que les coefficients aient des dénominateurs. On peut même borner ces dénominateurs par l'indice $[\mathscr{U} :< \eta_1,\ldots,\eta_r >]$, ce dernier indice s'estimant comme

$$[\mathscr{U} :< \eta_1,\ldots,\eta_r >] = \frac{R(\eta_1,\ldots,\eta_r)}{R_{\mathbb{K}}},$$

ce qui permet de calculer une borne explicite, au vu de l'estimation crue $R_{\mathbb{K}} \geq 0.2$ pour tout corps de nombres.

On écrit donc une unité quelconque sous la forme suivante (rappelons que le cas où le corps est totalement imaginaire est traité par un argument élémentaire) :

$$u = \pm\prod_{i=1}^{r}\eta_i^{b_i/b_{r+1}}.$$

Le traitement algébrique subséquent est de même nature, et conduit à des inégalités du type

$$\left|\sum_{i=1}^{r} b_i\log\theta_i + b_{r+1}\log\theta_{r+1}\right| \leq c\exp\left(-c'\cdot\max_{1\leq i\leq r}|b_i|\right).$$

Noter que la borne supérieure ne dépend pas de b_{r+1}, ce qui est normal, mais n'est pas une bonne chose d'un point de vue pratique : cela signifie qu'il sera impossible de réduire la borne sur b_{r+1} ; de la même façon, on ne peut obtenir via la borne de Baker aucune information sur b_{r+1}, mais on dispose déjà en tout état de cause d'une borne bien meilleure sur b_{r+1}.

La phase de réduction s'effectue alors de façon analogue, à ceci près que l'on se ramène à un système homogène de dimension 3, que l'on traite par application de l'algorithme LLL. Il apparaît toutefois un artefact de la méthode, à savoir que

– Comme remarqué plus haut, il est impossible de réduire b_{r+1} ;

– La borne sur les b_i ne descend pas en pratique en-dessous de la borne sur b_{r+1}.

D'un point de vue pratique, cela signifie que la réduction se déroule beaucoup moins bien. Néanmoins, ce n'est pas un problème-clé, au vu du fait que cette estimation n'est utilisée qu'en grand degré ; au lieu d'utiliser les deux termes extrêmes de (1.9), on peut utiliser les deux termes de gauche pour obtenir, via la technique de réduction une borne sur y. Quand n est grand (le cas intéressant pour cette variante), cette borne est usuellement raisonnable, et on peut terminer par un calcul de fraction continue.

Cette technique s'est révélée indispensable en particulier dans l'étude du problème des diviseurs primitifs, cf. *infra*.

1.3.7 Le cas des corps composés

Le contenu de cette partie a fait l'objet d'un travail commun avec Yuri Bilu, publié dans [14].

Cette partie a été largement inspirée par l'étude du problème des diviseurs primitifs, et par la nécessité de résoudre des équations de degré très important pour résoudre ce problème. Elle présente néanmoins un intérêt indépendant.

Il s'agit d'une modification de l'argument algébrique initial, visant là encore à améliorer l'efficacité de la méthode en permettant d'éviter la détermination d'un système fondamental d'unités du corps de nombres $\mathbb{K} = \mathbb{Q}(\alpha)$. Il se trouve que cela peut être fait dans le cas ou le corps $\mathbb{K}$ a un sous-corps $\mathbb{K}_0$ de degré au moins 3.

Dans ce cas, en effet, l'équation de Thue sous la forme

$$N_{\mathbb{K}/\mathbb{Q}}(x - \alpha y) = a$$

se récrit

$$N_{\mathbb{K}_0/\mathbb{Q}}(N_{\mathbb{K}/\mathbb{K}_0}(x - \alpha y)) = a,$$

et c'est cette dernière forme qui est exploitée, fournissant une identité

$$N_{\mathbb{K}/\mathbb{K}_0} = u\xi,$$

où u est une unité de $\mathbb{K}_0$ et ξ est dans un ensemble d'entiers algébriques de $\mathbb{K}_0$ explicite.

La conséquence de cette remarque élémentaire est le fait que les calculs algébriques préliminaires (unités, équations aux normes), qui constituent l'étape limitante du calcul, n'ont plus à être effectués dans la « grande » extension $\mathbb{K}$, mais dans la « petite » sous-extension $\mathbb{K}_0$. La dépendance de ce type de calculs en le degré est telle qu'ils sont fréquemment infaisables dans le premier cas et immédiats dans le second.

L'analogue du lemme fondamental 1 dans ce cadre est alors,

Proposition 3 *Notons $m = [\mathbb{K}_0 : \mathbb{Q}], l = [\mathbb{K} : \mathbb{K}_0]$, et*

$$\begin{aligned} \varphi_i &= \prod_{k=1}^{l} (y - \alpha_{ik} x) \qquad (1 \le i \le m), \\ \psi_i &= \prod_{k=1}^{l} (\alpha_{i_0 k_0} - \alpha_{ik}) \qquad (1 \le i \le m,\ \ i \ne i_0), \end{aligned}$$

Alors on a $\varphi := \varphi_1 = N_{\mathbb{L}/\mathbb{K}}(y - \alpha x)$ et $\varphi_i = \sigma_i(\varphi)$. En particulier,

$$\left| \mathrm{Log} \frac{\varphi_i}{\psi_i x^l} \right| \le \frac{l c_3}{|x|^{-n}} \quad (i \ne i_0), \tag{1.10}$$

Au vu de $\varphi_1 \cdots \varphi_m = a$, on a également

$$\left| \operatorname{Log} \frac{\varphi_{i_0}}{\psi_{i_0} x^{(1-m)l}} \right| \leq \frac{l(m-1)c_3}{|x|^{-n}}. \tag{1.11}$$

Le reste de l'argument de la partie 1.3 se transpose alors *mutatis mutandis*, en remplaçant $x - \alpha y$ par φ dans tout le texte. Ainsi, la quantité $\varphi_i \psi_j / (\varphi_j \psi_i)$ est voisine de 1, et s'exprime comme produit de puissances d'unités par un terme inhomogène. Prenant le logarithme, on se retrouve alors dans une situation analogue à la précédente.

La complexité de la méthode est essentiellement gouvernée par le degré $[\mathbb{K}_0 : \mathbb{Q}]$, à une restriction (de taille !) près : la borne de Baker, qui fait intervenir le degré des nombres algébriques impliqués, qui, lui, n'est pas modifié a priori dans le cas de ψ_i. À ce titre, la précision à laquelle doivent être menés les calculs reste presque aussi importante que si l'on résolvait l'équation dans le « grand corps ». Néanmoins, dans les cas où elle s'applique, cette variation apporte des gains d'efficacité considérables, et a réellement constitué une clé dans le problème des diviseurs primitifs. Cette amélioration et le raffinement algébrique de la section précédente sont orthogonaux, et se combinent sans difficulté particulière.

1.3.8 Quelques autres aspects

La progression notable du traitement de l'équation de Thue a conduit au développement de problèmes plus généraux. Aussi les dernières années ont-elles vu mûrir la théorie des équations de Thue relatives (où les inconnues vivent non plus dans $\mathbb{Z}$ mais dans l'anneau des entiers d'un corps de nombres ; ces équations permettent en particulier de traiter de façon directe des équations de type superelliptiques, cf. infra. On pourra consulter à ce sujet [30]. L'argument suit une progression assez similaire ; pour l'énumération finale, un argument dû à Wildanger est utilisé. Il serait intéressant de parvenir à généraliser les idées de combinaison de formes linéaires dans ce cadre.

De façon un peu orthogonale, on peut noter une progression assez conséquente de la résolution de *familles* d'équations de Thue. La résolution est dans ce cas essentiellement limitée par le fait qu'il est nécessaire de disposer d'un système d'unités de rang maximal pour toutes les valeurs du paramètre. Après un certain nombre de travaux, en particulier de Heuberger, la technique commence à être suffisamment bien maîtrisée pour que l'on puisse parler d'algorithme, voir [37]. On peut d'ailleurs signaler également des résultats sur les familles d'équations de Thue relatives.

1.4 Équations superelliptiques

La méthode présentée ici a été développée conjointement avec Yuri Bilu, améliorant des idées de ce dernier [10]. Elle a été publiée dans [15].

On appelle équation superelliptique une équation de la forme

$$ay^n = f(x),$$

où f est un polynôme à coefficients entiers, sans facteurs carrés.

Siegel a montré qu'hormis le cas où $n = 2$, $\deg f = 2$ qui correspond à une cônique, et où le nombre de solutions peut être infini, on peut obtenir – là encore au moyen d'un résultat sur l'approximation des algébriques par des rationnels – la finitude du nombre de solutions. Ce résultat a été rendu effectif par application de la méthode de Baker ; de la même façon que pour l'équation de Thue, on pourrait énoncer des résultats très généraux donnant des bornes pour les solutions. On préfère renvoyer, par exemple, à [19].

Nous nous proposons de décrire sommairement dans cette partie comment il est possible de résoudre effectivement une équation de ce type. Il ne s'agit pas du premier travail sur la question, dans la mesure où ces équations sont justiciables d'autres méthodes. En particulier, il est possible de la résoudre à un nombre fini d'équations de Thue (relatives, cf. 1.12) qui peuvent ensuite être traitées comme précédemment.

La méthode présentée ici est de nature distincte, sans que le corpus expérimental ou des arguments permettent réellement de trancher sur l'efficacité comparée des deux méthodes. La méthode des *logarithmes elliptiques*, elle, s'applique dans les cas où l'équation définit une courbe elliptique typiquement $n = 2, \deg f = 3$ ou 4, ou encore $n = 3$, $\deg f = 3$. Cette méthode est, là encore, difficilement comparable, en ce que le problème algébrique sous-jacent qu'elle cherche à résoudre (la détermination du groupe de Mordell-Weil de la courbe sous-jacente) est par nature assez différent du problème algébrique posé par la méthode que nous exposons ici. Là encore, on peut produire des exemples en faveur de l'une ou de l'autre méthode [51].

Exposé informel de la méthode

Pour simplifier le propos, nous supposerons $a = 1$ et f unitaire. Si n et $\deg f$ sont premiers entre eux, on peut toujours s'y ramener, quitte à introduire des solutions parasites. On peut de même supposer n premier, sauf dans le cas où $\deg f = 2$ et $n = 4$, qui se traite indépendamment et que nous omettrons ici. On récrira donc l'équation $y^p = f(x)$.

Soit α une racine du polynôme f. Un argument classique permet, dans le cas où (x,y) est une solution de l'équation, de prouver que l'idéal $(x-\alpha)$ vit dans un ensemble fini, explicite, modulo les puissances p-èmes d'idéaux. Il s'ensuit, si l'on connaît groupe des classes et unités, que l'on peut déterminer un sous-ensemble fini Ξ_α de l'anneau des entiers de $\mathbb{Q}(\alpha)$ tel que

$$(x-\alpha) \in \Xi_\alpha (\mathbb{Q}(\alpha)^*)^p.$$

Le même argument appliqué à une racine β montre qu'il existe Ξ_β tel que $(x-\beta) \in \Xi_\beta (\mathbb{Q}(\beta)^*)^p$.

Pour $p \geq 3$, en soustrayant les identités en α et β, on élimine x et on est ramené à un nombre fini d'équations de Thue relatives

$$\xi_\alpha u^p - \xi_\beta v^p = \beta - \alpha, \quad u \in \mathbb{Z}_{\mathbb{Q}(\alpha)}, v \in \mathbb{Z}_{\mathbb{Q}(\beta)}. \tag{1.12}$$

Nous traitons les choses de façon sensiblement différente, en construisant directement, quoique de façon significativement plus élaborée, une forme linéaire de logarithmes associée à l'équation. De ce qui précède, on tire l'existence d'un ensemble $\Xi_{\alpha,\beta}$ tel que

$$\frac{x-\alpha}{x-\beta} \in \Xi_{\alpha,\beta} (\mathbb{K}^*)^p,$$

où l'on a noté $\mathbb{K} = \mathbb{Q}(\alpha,\beta)$.

Notons ζ_p une racine primitive p-ème de l'unité. Il existe donc un nombre fini de ξ tels que, pour toute solution (x,y), il existe $k(x) \in \mathbb{Z}/p\mathbb{Z}$ tel que la quantité

$$\zeta_p^{k(x)} \left(\frac{x-\alpha}{x-\beta} \right)^{1/p}$$

vive soit dans $\mathbb{K}$, soit dans $\mathbb{K}_\xi = \mathbb{K}(\xi^{1/p})$.

Ici et dans la suite, nous notons $z^{1/p}$ la détermination principale de la puissance $1/p$-ème d'un nombre complexe z, *i.e.* celle dont l'argument est dans $]-\pi/p, \pi/p]$.

On forme alors la quantité

$$\varphi(x) := (x-\beta)\left(\zeta_p^{k(x)}\left(\frac{x-\alpha}{x-\beta}\right)^{1/p}-1\right)^p.$$

Cette quantité a plusieurs propriétés-clé :
– Il s'agit d'un entier algébrique de $\mathbb{K}_\xi$;
– Sa norme est *essentiellement indépendante de x* ; par conséquent, modulo les unités de $\mathbb{K}_\xi$, elle vit dans un ensemble fini et explicitement calculable ;
– Son ordre de grandeur s'estime simplement.
En effet, on a la proposition :

Proposition 4 *Pour toute solution (x,y), la quantité $\varphi(x)$ est un entier algébrique de $\mathbb{K}$. En outre, on a*

$$N_{\mathbb{K}_\xi/\mathbb{K}}\varphi(x)|(\beta-\alpha)^p,$$

avec égalité sauf peut-être dans le cas $\mathbb{K}_\xi = \mathbb{K}$.

Enfin, pour tout plongement σ_i de $\mathbb{K}_\xi$ dans $\mathbb{C}$, on définit $k_i(x)$ par

$$\sigma_i(\varphi(x)) = (x-\sigma_i(\beta))\left(\zeta_p^{k_i(x)}\left(\frac{x-\sigma_i(\alpha)}{x-\sigma_i(\beta)}\right)^{1/p}-1\right)^p.$$

Alors on a l'estimation

$$\sigma_i(\varphi(x)) = \gamma_i x^{\rho_i}(1+O(1/x)), \tag{1.13}$$

avec

$$\rho_i = \begin{cases} 1-p, & k_i = 0, \\ 1, & k_i \neq 0, \end{cases} \qquad \gamma_i = \begin{cases} \left(\frac{\sigma_i(\beta)-\sigma_i(\alpha)}{p}\right), & k_i = 0, \\ \left(\zeta^{k_i(x)}-1\right)^p, & k_i \neq 0, \end{cases}$$

À ce stade intervient une importante combinatoire : il nous faut
– énumérer les $\mathbb{K}_\xi$;
– pour chacun de ceux-ci, énumérer les éléments θ de norme égale ou divisant $(\beta-\alpha)^p$;
– pour chacun de ceux-ci, énumérer les vecteurs (k_i) possibles.

Dans une situation précise de ce type, on est alors en situation d'appliquer la méthode de Baker : en effet, la quantité $\varphi(x)/\theta$ est alors une unité de $\mathbb{K}_\xi$, et on a l'estimation

$$\left(\frac{\sigma_i(\varphi(x))}{\gamma_i\theta_i}\right)^{\rho_j}\left(\frac{\sigma_j(\varphi(x))}{\gamma_j\theta_j}\right)^{-\rho_i} = 1+O(1/x).$$

Écrivant $\varphi(x) = \zeta\prod_{k=1}^r \eta_k^{b_k}$, avec ζ une racine de l'unité et η_k un système fondamental d'unités, il vient en passant au logarithme :

$$\left|\sum_{k=1}^r b_k \mathrm{Log}\,\frac{\sigma_i(\eta_k)^{\rho_j}}{\sigma_j(\eta_k)^{\rho_i}} + b_{r+1}\frac{2i\pi}{d} + \mathrm{Log}\,\frac{(\gamma_i\sigma_i(\theta))^{\rho_j}}{(\gamma_j\sigma_j(\theta))^{\rho_i}}\right| = O(1/x) = O(\exp(-c\cdot\max_i|b_i|)),$$

où la dernière borne est montrée, comme dans le cas de l'équation de Thue, au moyen des estimations (1.13) et de la définition des b_i. Ici, d est l'ordre du groupe des racines de l'unité de $\mathbb{K}_\xi$.

Une fois le choix combinatoire effectué, toutes les quantités intervenant sont explicites hormis les b_i, et il suffit donc d'appliquer la méthode de Baker pour borner les b_i, puis le procédé de réduction pour résoudre complètement l'équation.

1.4.1 Mise en oeuvre

Il est nécessaire d'être très soigneux pour éviter une explosion combinatoire rendant la méthode impraticable.

En particulier, le nombre de corps $\mathbb{K}_\xi$ doit être rendu le plus petit possible (on a *in fine* à calculer des systèmes d'unités et le groupe des classes pour chacun de ces corps). En particulier, pour ce faire, il est crucial de tester la résolubilité de l'équation $x-\alpha = \xi\lambda^p \bmod \mathfrak{p}$ pour tous les $\xi \in \Xi_\alpha$, $\mathfrak{p}$ décrivant un ensemble d'idéaux premiers bien choisis de $\mathbb{Q}(\alpha)$. Si cette équation n'est pas résoluble, l'élément ξ peut être omis[5].

De même, il est important de tester si l'équation

$$\frac{x-\alpha}{x-\beta} = \xi\lambda^p \bmod \mathfrak{q},$$

avec $\mathfrak{q}$ un idéal de $\mathbb{Q}(\alpha,\beta)$ a des solutions, par énumération. Dans le cas contraire, on peut de nouveau écarter le ξ correspondant. Enfin, on peut tester l'isomorphisme des différents corps obtenus.

De la même façon, des arguments locaux peuvent permettre d'éliminer une partie des éléments θ. Enfin, en ce qui concerne les vecteurs k_i, la proposition suivante indique comment limiter (un peu) leur nombre.

Proposition 5 *Supposons que les plongements $\sigma_1,\dots,\sigma_s$ sont réels, et que $\sigma_i = \overline{\sigma}_{i+t}$ pour $s < i \le s+t$. Alors on a*

$$k_1(x) = \cdots = k_s(x) = 0, \tag{1.14}$$

$$k_i(x) + k_{i+t}(x) \equiv 0 \pmod{p} \quad (s < i \le s+t). \tag{1.15}$$

De plus, si $\mathbb{K}_\xi \neq \mathbb{K}$, et si $\sigma_{i_1},\dots,\sigma_{i_p}$ sont des prolongements d'un plongement fixé de $\mathbb{K}$ dans $\mathbb{C}$, on a $k_{i_1},\dots,k_{i_p} = \{0,\dots,p-1\}$.

Dans certains cas, il est encore possible de restreindre un peu le nombre de vecteurs possibles, mais il reste très grand en pratique, à savoir $\left(2^{(p-1)/2}((p-1)/2)!\right)^{s_0}(p!)^{t_0}$ avec les restrictions de la proposition, en notant s_0 et t_0 le nombre de plongements réels et imaginaires de $\mathbb{K}$.

1.4.2 Aspects algorithmiques

L'amélioration du procédé de réduction mentionné dans le cas de l'équation de Thue s'étend *mutatis mutandis* à la situation présente, où l'on dispose encore de $r-1$ équations indépendantes. Il est à noter que je ne sais pas l'étendre au cas des méthodes basées sur les logarithmes elliptiques, où l'on ne dispose que d'une unique équation sans action de type conjugaison.

L'utilisation d'un système d'unités de rang maximal reste encore possible. Il a été crucial pour les résultats sur le cas diagonal de l'équation de Nagell-Ljunggren, cf. infra.

1.4.3 Utilisation de la symétrie

Un cas important de cette méthode est le cas où α et β sont conjugués sur $\mathbb{Q}$, et où il existe un élément du groupe de Galois de f échangeant α et β. Dans ce cas, il est possible – mais nous ne le ferons pas – de montrer que l'on peut remplacer le corps $\mathbb{Q}(\alpha,\beta)$ par $\mathbb{Q}(\alpha+\beta,\alpha\beta)$, qui est de degré moitié. Les corps $\mathbb{K}_\xi$ correspondants sont alors eux aussi de degré moitié, ce qui facilite d'autant la détermination de leurs invariants.

[5] Noter la parenté, qui m'a été signalée par Nigel Smart, avec la méthode de p-descente sur les courbes superelliptiques [49].

En revanche, je ne sais là encore pas comment généraliser l'idée des sous-corps utilisée pour l'équation de Thue, question qui s'est posée avec insistance dans le cadre de l'étude de l'équation de Catalan, pour tenter d'obtenir un résultat qui améliorerait les résultats précédents dans tous les cas.

1.4.4 Prolongements

Ce travail a suscité, hélas, relativement peu de prolongements. Il est vrai que la méthode proposée est délicate à mettre en œuvre, et que l'explosion combinatoire se produit assez vite même pour des équations assez modestes. L'application au problème de l'équation de Nagell-Ljunggren (dans le cas diagonal), voir plus bas, montre toutefois qu'elle peut revêtir un intérêt réel. Signalons en outre le travail de de Weger [57], qui a étendu sur un exemple la méthode au cas de solutions S-entières (en complétant l'estimation archimédienne par une estimation de type p-adique).

En tout état de cause, il serait intéressant de réaliser une implantation de cet algorithme dépassant le stade du prototype, pour étudier de plus près les moyens d'éviter l'explosion combinatoire et évaluer réellement son efficacité sur des exemples plus « généraux » que ceux traités jusqu'alors. Étendre cet algorithme à des équations de type non superelliptique (l'exposé [11] est essentiellement effectif) constituerait également un objectif intéressant.

1.5 Un avatar de la méthode superelliptique

Le résultat présenté dans cette partie a fait l'objet d'un article commun avec Yann Bugeaud [22]

À la description de la méthode superelliptique, on peut relever un cas particulier semblant plus simple que les autres, à savoir le cas où l'ensemble $\Xi_{\alpha,\beta}$ est réduit à $\{1\}$. Dans ce cas, en effet, tous les calculs algébriques peuvent être effectués dans le corps $\mathbb{K}$, situation extrêmement favorable. Il se trouve en plus que si de surcroît le degré de $\mathbb{K}$ est premier à p, on peut obtenir une borne totalement élémentaire pour les solutions.

Pour ce faire, il suffit de reprendre les propriétés de la quantité $\varphi(x)$ précédemment introduite. Nous savons que

$$1|\varphi(x)|(\beta-\alpha)^p,$$

ce qui implique

$$1 \leq |N_{\mathbb{K}/\mathbb{Q}}\varphi(x)| \leq |N_{\mathbb{K}/\mathbb{Q}}(\beta-\alpha)^p|,$$

qui fournit un premier encadrement de $\varphi(x)$.

Il est possible simultanément d'estimer $N_{\mathbb{K}/\mathbb{Q}}\varphi(x)$ comme produit de ses conjugués, en utilisant les estimations (1.13). On voit en effet aisément que l'on peut, en encadrant les γ_i, déterminer des constantes explicites C_1 et C_2 telles que

$$C_1|x|^{\sum_{i=1}^n \rho_i} \leq \left|\prod_{i=1}^n \sigma_i(\varphi(x))\right| \leq C_2|x|^{\sum_{i=1}^n \rho_i}.$$

Enfin, la quantité $\sum_{i=1}^n \rho_i$ vaut $(1-p)\#\{i;k_i=0\}+\#\{i;k_i\neq 0\}$, ou encore

$$(1-p)\#\{i;k_i=0\}+[\mathbb{K}:\mathbb{Q}]-\#\{i;k_i=0\}=[\mathbb{K}:\mathbb{Q}]-p\#\{i;k_i=0\}.$$

Sous l'hypothèse que $([\mathbb{K}:\mathbb{Q}],p)=1$, cette quantité est toujours non nulle. En particulier, il vient

$$1 \leq |\prod \sigma_i(\varphi(x))| \leq \frac{C_2}{|x|}$$

si $\sum_{i=1}^{n} \rho_i$ est négative, et

$$C_1|x| \leq |\prod \sigma_i(\varphi(x))| \leq N_{\mathbb{K}/\mathbb{Q}}(\beta-\alpha)^p,$$

si la somme est positive, d'où une borne sur x de très bonne qualité.

1.5.1 L'équation de Catalan

L'équation $x^p - y^q = 1$ a été introduite par Catalan au milieu du XIX-ème siècle. Ce dernier avait conjecturé, dans une courte note publiée à Crelle [27], qu'elle n'avait comme solution non triviale que $9-8=1$, et il soumettait le problème à la sagacité de ses contemporains. Ce problème a finalement été résolu par Mihailescu en 2002. En 2000, Bugeaud et moi-même nous sommes aperçus que la méthode exposée ci-dessus s'appliquait à l'équation de Catalan.

En effet, en vertu d'un résultat de Cassels qui dit que pour toute solution non triviale, $p|y$ et $q|x$, l'équation de Catalan se réduit à l'équation

$$\frac{x^p-1}{x-1} = py^q.$$

Ce résultat avait éte utilisé par Inkeri, Mignotte, Schwarz pour prouver que sous une combinaison de deux hypothèses, l'une arithmétique sur les exposants p,q, l'autre sur l'arithmétique du corps cyclotomique ou de l'un de ses sous-corps, on pouvait prouver que l'équation n'avait pas de solution :

Théorème 6 *Soient $p \neq q$ deux nombres premiers. Soit $\mathbb{K}^{(p)}$ le plus petit sous-corps imaginaire du corps cyclotomique $\mathbb{Q}(\zeta_p)$, où ζ_p est une racine p-ième de l'unité, et soit $h^-(\mathbb{K}^{(p)})$ le nombre de classes relatif de $\mathbb{K}^{(p)}$. Alors (p,q) n'est pas une paire d'exposants pour (2) si*

$$q \not| h^-(\mathbb{K}^{(p)}) \qquad \text{et} \qquad p^{q-1} \not\equiv 1 \pmod{q^2}.$$

L'application de la méthode du paragraphe précédent est soumise à deux restrictions : $(p-1,q)=1$, soit encore $p \neq 1 \bmod q$. Une vérification algébrique sans difficulté montre que, sous réserve que q ne divise pas le nombre de classes relatif de $\mathbb{Q}(\zeta_p)$, on a bien $\Xi_{\zeta_p,\overline{\zeta}_p} = \{1\}$. Appliquant la méthode, on prouve alors pour $q > p$, la majoration

$$|x| \leq 4\left(\frac{q}{5}\right)^q + 3^q.$$

Cette majoration contredit des minorations assez fortes que Hyyro avait déduit du résultat de Cassels dans les années 1960, rendant impossible l'existence d'une solution dans ce cas.

Le théorème obtenu est alors le suivant :

Théorème 7 *Soient $q > p$ deux nombres premiers impairs. S'il existe des entiers $x > 0$ et $y > 0$ tels que $|x^p - y^q| = 1$, alors q divise $h^-(\mathbb{Q}(\zeta_p))$.*

Sans être réellement comparable au résultat précédent – qui a le mérite de faire intervenir des sous-corps, potentiellement petits, du corps cyclotomique –, ce résultat a le mérite de séparer pour la première fois la condition de type Wieferich de la condition de nombres de classes. Plus tard, dans une étape intermédiaire, Mihailescu montrera comment adapter la preuve d'Inkeri-Mignotte-Schwarz pour, inversement, ne conserver que la condition de Wieferich [43]. Enfin, une utilisation très fine de la cyclotomie lui permettra de prouver la conjecture de Catalan [44].

1.6 Quelques applications

Je rassemble dans cette partie trois travaux distincts, le premier commun avec Yann Bugeaud et Maurice Mignotte publié dans l'article [23], le deuxième commun avec Yuri Bilu et Paul Voutier – pour lequel il convient de signaler la contribution cruciale de Mignotte figurant dans l'appendice –, publié dans [16], et un dernier travail avec Natarajan Saradha et Tarlok Shorey, cf. [35].

Les méthodes étudiés dans les précédents paragraphes ont pour point commun le fait qu'elles permettent la résolution d'équations naturellement rencontrées en arithmétique. Cette partie présente la solution (partielle dans deux des cas) de trois questions, dont deux sont initialement exprimées sous forme de problème diophantien.

1.6.1 L'équation de Nagell-Ljunggren

L'équation de Nagell-Ljunggren, cousine de l'équation de Catalan, s'écrit sous la forme

$$\frac{x^n-1}{x-1} = y^q,$$

où l'on peut se restreindre au cas où q est un nombre premier. En dépit de la grande ressemblance apparente, cette équation est plus délicate à traiter que l'équation de Catalan, essentiellement parce que le résultat de Cassels, et donc les minorations de Hyyro pour les solutions, font défaut. Si n ne peut plus être supposé premier, on peut montrer qu'il est possible de se ramener au cas où n est premier, quitte à considérer également l'équation

$$\frac{x^n-1}{x-1} = py^q.$$

On conjecture que cette équation possède uniquement les trois solutions non triviales suivantes :

$$\frac{3^5-1}{3-1} = 11^2, \frac{7^4-1}{7-1} = 20^2, \text{et} \frac{18^3-1}{18-1} = 7^3,$$

mais on ne sait pas démontrer qu'elle n'en admet qu'un nombre fini, bien qu'il soit possible de prouver que cela est le cas si, par exemple, on fixe q.

Cette équation a été introduite, puis étudiée par Ljunggren et Nagell, qui l'ont respectivement résolue dans les cas où $q = 2$ et n multiple de 3 ou 4. Des travaux récents (Bennett, Bugeaud, Mignotte, Roy, Saradha et Shorey) ont permis de prouver l'absence de solution dans différentes situations, par exemple si x est un carré, ou x une puissance de 10.

Nous avons prouvé principalement les trois théorèmes suivants :

Théorème 8 *Le quadruplet $(3,11,5,2)$ est la seule solution (x,y,n,q) de l'équation (1) avec n multiple de 5, 7, 11 ou 13.*

Théorème 9 *Si (x,y,n,q) est une solution de (1) distinctes des 3 solutions mentionnées plus haut, et si p est un diviseur premier impair de n, alors ou bien $p \geq 29$ ou bien $(p,q) \in \{ (17,17), (19,19), (23,23)\}$.*

Théorème 10 *Si les entiers $n \geq 4$, $x > 1$ et $y > 1$ vérifient*

$$\frac{x^n-1}{x-1} = y^3,$$

alors il existe un nombre premier p congru à 5 modulo 6, et $a \geq 1$ tels que $n = p^a$. De plus, $p \geq 101$.

La preuve des théorèmes 1 et 2 est essentiellement en deux parties. Dans un premier temps, il nous faut borner q à n fixé, et dans un second temps résoudre l'équation à n et q fixé. La résolution à p et q fixé permet d'obtenir le théorème 3, moyennant un argument supplémentaire pour justifier la contrainte $n = p^a$, $p = 5 \bmod 6$.

La première partie

La première partie de la preuve consiste à borner q à n fixé. Si des bornes générales existent pour les familles d'équations superelliptiques $y^q = f(x)$, la meilleure version effective dont nous ayons connaissance est la suivante :

$$m < \max\left\{n\log_2(2H+3),\, 2^{15(n+6)}\, n^{7n}\, |D|^{3/2}\, (\log|D|)^{3n}\, \log^3|3a|\right\}.$$

Ici, on suppose f unitaire de hauteur H et de discriminant D ; La démonstration repose entre autres sur des minorations de formes linéaires en $r+3$ logarithmes, où r désigne ici le rang du groupe des unités d'un corps de nombres engendré par une racine de f. C'est en raison de l'absence d'estimations très fines pour les formes linéaires en $m \geq 3$ logarithmes que l'on obtient par cette approche des constantes numériques élevées, beaucoup trop grandes pour pouvoir traiter le problème qui nous intéresse. Toutefois, dans le cas présent, il est possible de se ramener à utiliser des formes linéaires en 2 logarithmes, ce que nous décrivons maintenant brièvement.

Le résultat obtenu est le suivant :

Théorème 11 *Soit p un nombre premier impair. Alors les équations (2) et (3) ne peuvent avoir de solutions non triviales que pour*

$$q \leq 9000\, p^2\, \log^4 p.$$

En outre, si p n'est pas congru à 1 modulo 8, on dispose de la majoration plus fine

$$q \leq 64000\, p\, \log^2 p.$$

Enfin, pour $p = 5, 7, 11, 13, 17, 19, 23$, nous obtenons respectivement les bornes 5521, 25404, 41784, 213949, 197658, 72123, 87523.

Soit $\mathbb{K}$ un sous-corps du p-ème corps cyclotomique. Posons $A(x) := N_{\mathbb{Q}(\zeta_p)/\mathbb{K}}(x - \zeta_p)$. Sous l'hypothèse que $q \not| h^-(\mathbb{K})$ si K est CM, $q \not| h(\mathbb{K})$ sinon. Cette hypothèse ne nous gênera pas sauf dans un petit nombre de cas (cf. infra) puisque nous nous intéressons essentiellement au cas des petites valeurs de p, on obtient par l'argument général une décomposition

$$A(x) = \eta\beta\alpha^q, \tag{1.16}$$

où β est 1 ou l'unique élément de norme p, selon l'équation que l'on traite, et η est une unité. Il nous faut alors distinguer deux cas. Si $\mathbb{K}$ est CM, on obtient ;

$$\frac{A(x)}{\overline{A(x)}} = \pm\left(\frac{\alpha}{\overline{\alpha}}\right)^q, \tag{1.17}$$

et dans le cas où $\mathbb{K}$ est quadratique réel, il vient :

$$\frac{A(x)}{\rho(A(x))} = \varepsilon^j \left(\frac{\alpha}{\rho(\alpha)}\right)^q, \tag{1.18}$$

où ρ est l'élément d'ordre 2 du groupe de Galois, et ε l'unité fondamentale du corps.

En outre, dans tous les cas, le quotient des deux conjugués de A est très proche de 1 ; on vérifie aisément qu'il est $1 + O(1/x)$. Par suite, prenant le logarithme de cette quantité, on trouve dans les deux cas une forme linéaire en deux logarithmes (dans le cas CM, il ne faut pas oublier le multiple entier de $2i\pi$).

Dans chacun des deux cas, l'application du résultat de Laurent, Mignotte et Nesterenko conduit alors à une borne supérieure pour x en fonction de p,q,y. En utilisant alors le fait que $p\log x > q\log y$, on élimine x et y de cette borne, et on obtient la borne supérieure annoncée.

Notons que pour q plus grand que ces bornes, les majorations des nombres de classes montrent que q ne peut en tout état de cause pas diviser $h^-(\mathbb{K})$ ou $h(\mathbb{K})$ dans le cas quadratique réel. Par suite, la proposition est vraie inconditionnellement.

La seconde partie

En ce qui concerne la seconde partie, le traitement algébrique de cette équation étant très comparable à celui de l'équation de Catalan, on peut se douter qu'il va être possible d'obtenir des majorations pour les solutions du même type. C'est effectivement le cas, même si elles sont un peu moins bonnes en pratique, avec une exception notable, le cas $p = q$, que nous avons baptisé *diagonal*. Dans ce dernier cas, en effet, la méthode de majoration exposée plus haut échoue, car une racine p-ème de l'unité n'est plus a priori une puissance q-ème dans $\mathbb{Q}(\zeta_p)$. On arrive alors à l'estimation :

Théorème 12 *Soient p et q deux nombres premiers distincts tels que $p \not\equiv 1 \bmod q$ et $q \not| h^-(\mathbb{Q}(\zeta_p))$. On définit*

$$\alpha(\ell) = \left(2\sin\frac{\pi}{q}\right)^{q(p-1-2\ell)} \left(\left(\frac{2}{q}\right)^{\ell} \prod_{j=1}^{\ell} \sin\frac{2j\pi}{p}\right)^{2q},$$

$$\beta(\ell) = \left(2\cos\frac{\pi}{2q}\right)^{q(p-1-2\ell)} \left(\left(\frac{2}{q}\right)^{\ell} \prod_{j=1}^{\ell} \cos\frac{j\pi}{p}\right)^{2q}$$

et enfin

$$M(\ell) = \begin{cases} p^q/\alpha(\ell), & \text{si } p-1 > 2q\ell, \\ \beta(\ell), & \text{si } p-1 < 2q\ell. \end{cases}$$

Alors si $(x^p-1)/(x-1) = y^q$ ou py^q, et si C est un réel arbitraire, on a

$$|x| \le \max\left(8q^2, 8(p-1)q^3\frac{\log 2}{C}, \max_{0\le\ell\le(p-1)/2} \left(e^C M(\ell)\right)^{\frac{1}{|p-1-2q\ell|}}\right).$$

Asymptotiquement, pour p fixé et $q \to \infty$, on trouve l'ordre de grandeur $(q/2\pi)^q p^{q/(p-1)}$.

En revanche, l'absence de résultats de minoration impose d'effectuer la vérification jusqu'à ces bornes, ce qui a pu se faire au moyen de techniques de crible. L'étude de l'équation modulo un nombre premier l congru à 1 modulo q permet en effet de localiser x dans un petit nombre de classes modulo l. La combinaison de plusieurs premiers l_i permet alors de localiser x dans des classes modulo $\prod l_i$. En utilisant suffisamment de l_i, on finit par localiser x dans des classes modulo $\prod l_i$ qui n'intersectent $[-B,B]$ qu'en un petit nombre de solutions potentielles ; il suffit alors de montrer qu'on n'obtient aussi aucune solution de l'équation.

Dans le cas où q est grand devant $p-1$, la borne en y est de bien meilleure qualité, et il est parfois préférable de cribler en y, même si ce crible est moins efficace.

Enfin, une idée apparemment saugrenue a été cruciale pour mener à bien les calculs. La technique élémentaire développée plus haut a permis d'éviter l'attirail des formes linéaires de logarithmes et le calcul des invariants algébriques sous-jacents, Néanmoins, certaines des bornes sont tellement grandes qu'elles ne sont pas accessibles par le crible. Nonobstant l'existence de la borne supérieure élémentaire, on peut faire subir à l'équation le traitement général superelliptique (qui, sur le plan algébrique, est

simple dans ce cas), à cela près que l'on remplace la borne de Baker par notre borne élémentaire de meilleure qualité. La technique de réduction permet ensuite de réduire considérablement cette borne en une étape, de façon à pouvoir ensuite faire l'énumération. La mise en oeuvre efficace de la réduction dans ce cas est assez technique, et requiert l'utilisation d'idées de type « diviser pour régner » pour en réduire la complexité. On renvoie à [23] pour les détails.

Le cas diagonal

Le cas diagonal ne peut être traité comme précédemment, mais il est justiciable du traitement général d'une équation superelliptique présenté *supra* en 1.4. La vraie difficulté est l'explosion combinatoire liée à l'énumération des vecteurs k_i ; il n'y a en effet qu'un seul corps $\mathbb{K}_\xi$ non trivial, et un seul θ dans ce corps. Mais le nombre de $p(p-1)$-uplets (k_i) à considérer est de $(p-1)p^{(p-3)/2}/2$, une fois toutes les symétries exploitées. C'est cette limite combinatoire qui fait que nous nous sommes limités à $p = 13$ dans le cas diagonal, $p = 17$ semblant accessible, mais au prix d'un effort de calcul important.

Le cas $q = 3$

Le cas $q = 3$ a nécessité une dernière variation technique, du fait de la présence pour 3 valeurs de $p \leq 101 (p = 23, 59, 83)$ d'une q-partie dans le groupe des classes de $\mathbb{Q}(\zeta_p)$. Une fois cette q-partie connue, des arguments locaux permettent souvent de montrer qu'elle n'intervient pas. Toutefois, dans deux des trois cas, elle est quasi-impossible à déterminer explicitement. Néanmoins, dans ces cas, on peut déterminer un idéal qui est un bon candidat pour être d'ordre 3, et vérifier simultanément, via le même argument local, qu'il est effectivement d'ordre 3 et n'intervient pas dans la résolution de l'équation.

1.6.2 Diviseurs primitifs des suites de Lucas et Lehmer

Nous définissons une paire de Lucas (α, β) comme un couple d'entiers algébriques tels que $\alpha+\beta$ et $\alpha\beta$ sont des entiers non nuls, premiers entre eux, et α/β n'est pas une racine de l'unité. On forme alors la suite de Lucas associée

$$u_n = u_n(\alpha, \beta) = \frac{\alpha^n - \beta^n}{\alpha - \beta} \quad (n = 0, 1, 2, \ldots) \tag{1.19}$$

De façon analogue, on peut définir une paire de Lehmer comme un couple d'entiers algébriques tels que $(\alpha+\beta)^2$ et $\alpha\beta$ sont des entiers non nuls, premiers entre eux, et α/β n'est pas une racine de l'unité. On forme alors la suite de Lehmer associée :

$$\tilde{u}_n = \tilde{u}_n(\alpha, \beta) = \begin{cases} \dfrac{\alpha^n - \beta^n}{\alpha - \beta} & \text{if } n \text{ is odd,} \\ \dfrac{\alpha^n - \beta^n}{\alpha^2 - \beta^2} & \text{if } n \text{ is even.} \end{cases} \tag{1.20}$$

Ces deux familles de suites constituent des objets classiques de la théorie des nombres et de l'arithmétique, qui interviennent dans de nombreux sujets, de la théorie algorithmique des nombres (factorisation, primalité) à certaines équations de type quadratique.

L'une des questions importantes reliées aux suites de Lucas et de Lehmer est la question de leur structure multiplicative, et, plus spécifiquement, de leurs diviseurs premiers. Une partie de la structure de u_n est triviale, c'est celle qui provient de u_m pour $m < n$, car on voit aisément que $u_m | u_n$ si $m | n$. On définit alors un diviseur primitif p de $u_n(\alpha, \beta)$ comme un diviseur de $u_n(\alpha, \beta)$ qui ne divise pas

$$(\alpha - \beta)^2 u_1 \cdots u_{n-1}.$$

TAB. 1.1 –

n	(a,b)	
2	$(1,1-4q)$, $q \neq 1$	$(2^k, 4^k-4q)$, $q \equiv 1 \bmod 2$, $(k,q) \neq (1,1)$
3	$(m, 4\varepsilon - 3m^2)$, $m > 1$	$(m, 4\varepsilon \cdot 3^k - 3m^2)$, $m \not\equiv 0 \bmod 3$, $(k,m) \neq (1,2)$
4	$(m, 2-\varepsilon m^2)$, $m > 1$, $m \equiv 1 \bmod 2$	$(m, 4-\varepsilon m^2)$, $m > 2$, $m \equiv 0 \bmod 2$
6	$(m, (4-m^2)/3)$, $m \geq 4$, $m \not\equiv 0 \bmod 3$	$(m, (2^{k+2}-\varepsilon m^2)/3)$, $m \equiv \pm 1 \bmod 6$, $k \equiv \varepsilon' \bmod 2$
	$(m, 4-\varepsilon m^2/3)$, $m \equiv 0 \bmod 3$	$(m, 2^{k+2}-\varepsilon m^2/3)$, $m \equiv 3 \bmod 6$

Notation : q est un entier non nul, k et m sont des entiers strictement positifs, $\varepsilon \in \{1,-1\}$, $\varepsilon' = (1-\varepsilon)/2$.

TAB. 1.2 –

n	(a,b)
5	$(1,5),(1,-7),(2,-40),(1,-11),(1,-15),(12,-76),(12,-1364)$
7	$(1,-7),(1,-19)$
8	$(2,-24),(1,-7)$
10	$(2,-8),(5,-3),(5,-47)$
12	$(1,5),(1,-7),(1,-11),(2,-56),(1,-15),(1,-19)$
13	$(1,-7)$
18	$(1,-7)$
30	$(1,-7)$

La même notion se définit de façon analogue pour les suites de Lehmer.

Nous identifierons enfin deux paires de Lucas opposées, car elles engendrent la même suite. De la même façon, deux suites de Lehmer qui diffèrent d'une racine quatrième de l'unité seront identifiées.

On peut alors énoncer le problème des diviseurs primitifs :

> Trouver tous les triplets (n,α,β) tels que $u_n(\alpha,\beta)$ (resp. $\tilde{u}_n(\alpha,\beta)$) n'ait pas de diviseur primitif.

Le résultat que nous avons prouvé est le suivant (énoncé dans le cas des suites de Lucas) :

Théorème 13 *Pour $n = 1$, on n'a jamais de diviseur primitif.*

Pour $n \in \{2,3,4,6\}$, il y a une infinité de couples (α,β) solution. Les solutions sont de la forme $(\alpha,\beta) = (a \pm \sqrt{b})/2$, où les couples (a,b) sont donnés dans la table 1.1.

Pour $4 < n \leq 30$, $n \neq 6$, il n'y a qu'un nombre fini de solutions. Les couples (a,b) correspondants sont donnés dans la table 1.2.

Pour $n > 30$, $u_n(\alpha,\beta)$ a toujours un diviseur primitif.

La preuve, là encore, se décompose en plusieurs étapes. Dans un premier temps, il faut se reposer sur un critère classique, attribué usuellement à Stewart, ramenant le problème de départ à la résolution d'une famille d'équations de Thue, indexées par n, de degré $\varphi(n)/2$.

Dans un second temps, des considérations de formes linéaires en deux logarithmes permettront de borner efficacement l'entier n.

Enfin, l'utilisation des techniques décrites supra – qui ont, pour ce faire, été poussées à leurs limites – permet de résoudre toutes les équations restantes. De façon à obtenir la meilleure borne lors de la seconde phase, pour limiter la taille des équations à résoudre, nous avons fait appel à Maurice Mignotte, afin qu'il établisse pour nous un raffinement ad hoc pour les formes linéaires en deux logarithmes que nous avons rencontrées. Il n'est pas clair que le calcul aurait pu être mené à bien à l'époque sans ce raffinement.

Le critère de Stewart

Nous nous contentons d'énoncer ce critère classique de non-existence d'un diviseur primitif.

Théorème 14 *Soit $n > 4$ un entier distinct de 6 et 12. Alors $u_n(\alpha,\beta)$ n'a pas de diviseur primitif si et seulement si $\Phi_n(\alpha,\beta)$ vaut $\{\pm 1, \pm P^+(n/(n,3))\}$, où $P^+(k)$ est le plus grand diviseur premier de l'entier k.*

La preuve est élémentaire, et est basée sur une étude arithmétique soigneuse de la condition

$$p | \Phi_n(\alpha,\beta),$$

pour p un nombre premier. On montre que, hormis pour les premiers divisant $(\alpha+\beta)$, les entiers n sont exactement de la forme $m_p p^k$, pour un entier m_p fixé (divisant $p, 2p, p-1$ ou $p+1$ selon les cas). L'étude des implications de la décomposition $u_n(\alpha,\beta) = \prod_{d|n} \Phi_d(\alpha,\beta)$ permet alors de conclure (mais la preuve est assez longue).

En réécrivant $\Phi_n(\alpha,\beta)$ comme $\Psi_n(\alpha+\beta,\alpha\beta)$, avec

$$\Psi_n(X,Y) = \prod_{\substack{1\le k\le n/2 \\ (k,n)=1}} (Y - 2X\cos(2k\pi/n)),$$

on voit qu'on est ramené à résoudre une famille d'équations de Thue.

Les petites valeurs de n ($\varphi(n) \le 4$) se traitent alors soit par étude directe ($\varphi(n) = 2$), soit par résolution d'une équation de Pell ($\varphi(n) = 4$), tandis que les valeurs restantes correspondent à des équations de Thue.

Une borne sur n

La poursuite de l'étude arithmétique permet de se limiter au cas où n est de surcroît sans facteur carré ; enfin, des résultats antérieurs [54] assurent que pour toute solution, $h(\beta/\alpha) > 4$, où h est la hauteur logarithmique ; il s'ensuit, au vu du fait que le résultat est connu depuis Carmichael et Ward [25, 55] dans le cas réel, que $|X| = |\alpha\beta| > e^8$.

Pour pouvoir résoudre complètement le problème, il nous faut maintenant obtenir une borne supérieure. En rapprochant le critère cyclotomique du lemme fondamental pour l'équation de Thue, on arrive à la remarque suivante :

Proposition 6 *Soit $n \ge 31$ un entier sans facteur carré, et (α,β) tel que $u_n(\alpha,\beta)$ n'admet pas de diviseur primitif. Soit $\gamma = \beta/\alpha$. Il existe une unique racine primitive n-ème de l'unité ξ telle que*

$$0 < \phi := |\arg(\gamma\xi^{-1})| < \min\left(\pi/n, c(n)|\alpha|^{-\varphi(n)}\right), \tag{1.21}$$

où

$$c(n) = \begin{cases} \pi & \text{si } n \text{ premier,} \\ \pi n^{2^{\omega(n)-2}-1} P(n) & \text{si } n \text{ composé} \end{cases}$$

TAB. 1.3 –

k	1	2	3	4
$N(k)$	787	$1329 = 3 \cdot 443$	$1695 = 3 \cdot 5 \cdot 113$	$2145 = 3 \cdot 5 \cdot 11 \cdot 13$

Cela entraîne en particulier une borne supérieure sur $\arg \gamma^n$. Or, $\arg \gamma^n = n \log \gamma - ki\pi$, pour un certain entier k est une forme en deux logarithmes ; on peut donc la minorer en utilisant la borne conçue à cet effet par Maurice Mignotte, par une quantité se comportant essentiellement comme $\exp(-O((\log n)^2))$. En comparant cette borne et la borne supérieure, pour n assez grand on trouve une relation entre $h(\gamma)$, n et $\varphi(n)$ qui implique :

- pour n assez grand, $h(\gamma) \leq 4$, et donc la non-existence de solutions ;
- pour n moyen, une borne sur $h(\gamma)$, et donc une borne sur X et Y solutions de l'équation de Thue ci-dessus ; en particulier, les petites solutions d'une telle équation étant détectées efficacement au moyen du développement en fraction continues, on peut dans ce cas résoudre très vite les équations impliquées.

En-deçà d'un certain n, l'inégalité ne donne plus d'information sur $h(\gamma)$; il faut donc réellement la voir comme donnant *in fine* une borne sur n, qui, du fait de l'intervention de $\varphi(n)$, dépend de la taille et de la structure multiplicative de n. On trouve :

Proposition 7 *Soit n un entier sans facteur carré ; si $u_n(\alpha, \beta)$ n'a pas de diviseur primitif, alors le nombre de facteur premiers de n est au plus 4. De plus, $n \leq N(\omega(n))$, où N est donné dans la table ci-dessous. En particulier, nécessairement $n \leq 2145$.*

Résolution

Il nous reste donc à résoudre un certain nombre, assez important, d'équations de Thue. Heureusement, les corps impliqués ont beaucoup de sous-corps (correspondant aux diviseurs de $\varphi(n)/2$. En particulier, dès que $\varphi(n)/2$ n'est pas un nombre premier ou le double d'un nombre premier, on peut utiliser un sous-corps. D'un point de vue pratique, la stratégie a été la suivante :

- si $\varphi(n)/2$ a un facteur premier compris entre 3 et 11 ou est divisible par 4 (c'est en particulier toujours le cas si n a au moins 3 facteurs premiers), travailler dans le corps correspondant, dont on détermine un système d'unités de rang maximal au moyen de la méthode générale, et traiter l'équation comme une équation générale.
- dans le cas contraire, on travaillera toujours dans le sous-corps le plus petit possible, mais on utilisera un système d'unités de rang maximal obtenu en prenant la norme des unités cyclotomiques $\sin(k\pi/p)/\sin(\pi/p)$, pour p un facteur premier de n. Dans ce cas, les solutions de l'équation aux normes sont également données explicitement, car p se ramifie comme $(1-\zeta_p)^{p-1}$ dans le p-ème corps cyclotomique ; la partie algébrique du calcul est donc explicite.

En revanche, dans le second cas, on a à inverser une matrice de dimension p à une très grande précision (rappelons que la réduction nous impose de connaître les réels intervenant en précision de l'ordre de B). Pour ce faire, on a obtenu une forme close explicite pour l'inverse de la matrice impliquée, ce qui a permis de ne calculer que les termes utiles de celle-ci, et donc de réduire le temps de calcul de $O(p^3)$ à $O(p^2)$. Pour les détails, nous renvoyons à [16]. Le temps de calcul a été environ de 1 an-machine sur des machines du type Pentium Pro-200 ; le cas $n = 719$, qui impose de travailler en dimension 359, a occupé à lui seul 1/5-ème du temps de calcul.

Applications

Ce résultat a connu depuis quelques applications dans l'étude d'équations diophantiennes. Le problème évoqué apparaît en effet naturellement par exemple quand on étudie des équations diophantiennes de type Ramanujan-Nagell généralisé $y^n = x^2 + D$, et permet généralement d'obtenir des bornes fines sur le nombre de solutions d'équations de ce type. On pourra par exemple consulter [24, 20], ou encore [12].

1.6.3 Sur un problème d'Erdős et Selfridge

Dans les années 1970, Erdős et Selfridge se sont intéressés à la question de savoir quand un produit d'entiers consécutifs pouvait être une puissance pure, ou en d'autres termes aux solutions de l'équation

$$(n+1)\dots(n+k) = by^l,$$

où l'entier b, auquel on impose $P(b) < k$, est présent pour écarter les diviseurs triviaux du membre de gauche. Ils ont pu montrer que cette équation n'avait pas de solution, résultat généralisé ultérieurement par Saradha puis Győry pour $P^+(b) \leq k$, sous réserve que $n > k^l$.

Une généralisation du problème a alors été introduite par Saradha et Shorey, qui consiste à autoriser la suppression d'un terme dans le membre de gauche. Il existe alors des solutions même dans le cas où l'on impose $b = 1$, comme par exemple $2 \cdot (2+2) = 2^3$, ou encore $1 \cdot (1+1) \cdot (1+3) = 2^3$. Saradha a montré qu'alors $k \leq 8$, sous réserve que $n > k^l$.

Le principal théorème que nous avons obtenu est le suivant :

Théorème 15 *Soit $n > k^l$. Un produit de $k-1$ entiers parmi k consécutifs commençant en n n'est jamais de la forme by^l avec $P(b) \leq k$ si $k \geq 6$. Il n'est jamais de la forme by^l avec $P^+(b) < k$ si $k \notin \{2,4\}$.*

Une équation de la forme de celle étudiée dans le théorème se ramène à un nombre fini de systèmes

$$n + d_i = a_i x_i^l,$$

où a_i ne contient pas de puissance l-ème, et $P^+(a_i) \leq k$. De surcroît, $n > k^l$ implique que les a_i sont distincts.

Un lemme combinatoire d'Erdős-Selfridge permet en outre de borner l par 17, et d'imposer des contraintes sur les a_i. L'utilisation de ce lemme permet, pour chaque valeur de $k \in \{6,7,8\}$, de se ramener à résoudre un petit ensemble d'équations de la forme

$$m(m+i)(m+j) = b'y^l,$$

avec $P^+(b') \leq 3$.

Pour $l \geq 5$, une nouvelle décomposition de chacune de ces équations nous ramène alors à des équations du type $ax^l - by^l = c$, avec $P^+(ab) \leq 3$. La plupart des équations effectivement rencontrées ont déjà été résolues, essentiellement par des techniques de type modulaire [28, 46, 48] (voir aussi toutefois [5]) ; une fois ceci observé, il nous reste alors un nombre fini d'équations de Thue, qui peuvent être résolu par la méthode du paragraphe 1.3. Depuis, Bennett [6], en utilisant des techniques modulaires, a étendu le résultat en supprimant les restrictions sur k ; d'autres articles ont généralisé tous ces résultats dans différentes directions (e.g., progressions arithmétiques plutôt qu'entiers consécutifs [7], ajout d'un terme additif [17], etc.).

1.7 Résolubilité par radicaux

Ce paragraphe isolé mentionne un travail commun avec François Morain sur la résolution d'équations par radicaux ; bien qu'il soit assez étranger aux principaux thèmes de ce chapitre, il me parait relever d'une même logique, la résolution d'équations ; il m'a donc semblé naturel de l'inclure ici, en forme d'appendice.

Une conséquence classique de la théorie de Galois est le fait qu'une équation peut être résolue par radicaux si et seulement si son groupe de Galois est résoluble. La preuve de Galois est déjà essentiellement effective : au dévissage du groupe correspond une suite d'extensions cycliques $\mathbb{K}_{i+1}/\mathbb{K}_i$; enfin, quitte à rajouter une racine de l'unité d'ordre le degré de l'extension, une construction classique permet d'obtenir un élément primitif pour l'extension de polynôme minimal $X^d - \alpha$, avec $\alpha \in \mathbb{K}_i$.

Nous avons étudié la mise en œuvre numérique de cette méthode, qui soulève quelques difficultés : représentation des entiers algébriques et des nombres algébriques, précision du calcul ; l'objectif était l'utilisation dans la représentation des extensions abéliennes de corps quadratiques imaginaires rencontrées dans l'algorithme de primalité ECPP.

Dans cet algorithme, en effet, on obtient des extensions abéliennes, de groupe connu (il s'agit du groupe des classes d'un corps quadratique imaginaire), décrites par un polynôme ; on va souhaiter à une étape ultérieure trouver les racines de ce polynôme modulo p. Dans certaines situations, il est plus efficace de faire cette opération par théorie de Galois que par la méthode générale.

Chapitre 2

Algorithmes en arithmétique des ordinateurs

2.1 Introduction

Dans un souci d'unité, je parlerai ici d'arithmétique des ordinateurs dans un sens assez large. J'y inclurai une bonne part qui, pour certains, relèverait du calcul formel suivant la classification d'usage, mais qu'il me paraît plus cohérent de rattacher aux travaux d'arithmétique des ordinateurs, en raison même du but poursuivi lors de ces recherches, le cas polynômes/séries formelles servant de modèle simplifié au cas entier/réel.

Il est important en arithmétique des ordinateurs de bien distinguer deux cas : celui des données exactes (polynômes[6]/entiers) de celui des données inexactes (séries formelles/réels), qui relèvent de deux problématiques sensiblement différentes[7].

Pour le premier type, en effet, seul le problème de l'*efficacité* se pose. Les données d'entrée sont bien définies, la réponse est, elle aussi, définie de façon unique, l'objectif est de la calculer le plus efficacement possible, en espace et en temps.

Pour le second type, en revanche, les problèmes sont de nature diverse. Une première question qui se pose est la question de la *représentation* des données : un élément inexact est un élément d'un ensemble infini, dont seul un nombre fini d'éléments pourra être représenté de façon exacte. Il convient de bien choisir ce sous-ensemble.

Une seconde question est la question de la *sémantique du calcul*. Le sous-ensemble représentable étant fini, il n'y a pas de raison *a priori* qu'il soit stable par les opérations et fonctions usuelles. Ce n'est d'ailleurs pas le cas, ni pour les séries formelles (excepté pour l'addition et la soustraction), ni pour les réels avec les sous-ensembles de représentables usuels. Il nous faut donc définir, pour chaque nombre réel et chaque série formelle, l'élément de l'ensemble des représentables qui le représente. Il s'agit de l'*opération d'arrondi*, que nous noterons $\diamond(x)$. La sémantique du calcul est alors la suivante : pour tous représentables x et y, chaque opérateur op et chaque fonction f,

$$x \tilde{\text{op}} y = \diamond(x \,\text{op}\, y),\ \tilde{f}(x) = \diamond(f(x)),$$

les symboles $\tilde{\text{op}}$ et $\tilde{f}$, définis par les identités ci-dessus, désignant les fonctions et opérateurs induits par op et f sur l'ensemble des nombres représentables. Il convient en outre de se doter d'éléments particuliers, tels $\pm\infty$ pour représenter les nombres plus grands ou plus petits que tous les représentables

[6] encore qu'un polynôme à coeffi cients de type inexact relève à la fois des deux logiques exacte/inexacte...

[7] De façon orthogonale, on pourrait distinguer le cas « archimédien » (réels/entiers) du cas « non-archimédien » (polynômes/séries formelles), le premier cas compliquant généralement le second par la présence de retenues.

(dans le cas archimédien), et enfin d'un nombre (usuellement noté `NaN`) pour représenter les résultats d'opérations invalides, telles $\log(-1)$ ou $\sqrt{-1}$ (ou $\log(X)$ dans le cas des séries formelles, etc.)

Cette question de la sémantique du calcul est cruciale pour la fiabilité des programmes et tout essai de preuve de correction d'un calcul. Il paraît aujourd'hui inconcevable qu'on ait pu vivre si longtemps sans disposer d'un calcul flottant précisément spécifié, et surtout de façon uniforme selon les architectures ! Et pourtant, la narration par Kahan de la naissance de la norme IEEE 754, qui précise les points ci-dessus dans le cas des nombres flottants machine, montre bien que le sacrifice – partiel – de l'efficacité à la fiabilité a été ressenti à l'époque comme un prix bien lourd à payer. *A contrario*, les différents développements depuis l'apparition de cette norme montrent qu'une sémantique claire est nécessaire pour développer un calcul numérique fiable et prouvé, et les exemples de catastrophes provoquées par un mauvais calcul flottant sont suffisamment nombreux (Ariane 5, missile Patriot, bug du Pentium, etc.) pour se convaincre de l'utilité de ce concept.

Le troisième point, donc, est celui de l'*efficacité* du calcul, comme dans le cas exact, que l'on veut la plus grande possible. La tradition tend à considérer que la vraie question est celle du compromis entre l'efficacité et la fiabilité. On peut l'illustrer sur un exemple simple, de façon informelle : supposons que nous nous donnions deux nombres réels en précision N, par leur développement binaire : $A = \sum_{n=0}^{N-1} a_n 2^{-n}$ et $B = \sum_{n=0}^{N-1} b_n 2^{-n}$. Leur produit est alors défini mathématiquement comme $\sum_{n=0}^{2N-2} \left(\sum_{j=0}^{n} a_j b_{n-j}\right) 2^{-n}$, en considérant que les coefficients a_k et b_k sont nuls pour $k \geq N$. La sémantique décrite ci-dessus, si l'on suppose que les nombres représentables sont ceux de la forme $\sum_{n=0}^{N-1} c_n 2^{-n}$, suggère donc de calculer cette somme explicitement, puis de la tronquer. Il est tentant, plutôt que de calculer les N^2 termes nécessaires pour évaluer exactement le produit, de se limiter à $N(N+1)/2$ en ne calculant que les N premiers termes. Cette quantité est appelée le *produit court* de A et B.

Ce faisant, on néglige toutefois un ensemble de retenues dont la contribution est contrôlable, mais non nulle, et changerait potentiellement le résultat ; chaque multiplication dans ce cas induit donc une perte de précision. On a donc le choix entre une solution exacte de complexité N^2 multiplications (et additions) et une solution approchée de complexité $N^2/2$ multiplications (et additions). Laquelle choisir ? La réponse dépend de l'application visée.

Mais si une sémantique claire des calculs doit être garantie, le second terme de l'alternative est interdit, même si l'on peut trouver un bon compromis suivant la même idée, préservant la fiabilité et souvent plus efficace que la première solution. Ce type de solutions est au coeur du développement de la bibliothèque MPFR, dont l'objet était de montrer que le prix de la fiabilité est suffisamment faible pour être payé.

Pour donner un exemple, et faire un pont entre les deux parties de cette thèse, les méthodes de réduction des bornes pour les équations diophantiennes évoquées dans la première partie sont basées sur les propriétés arithmétiques des nombres réels coefficients des formes linéaires que nous considérons ; mais ces propriétés arithmétiques sont très sensibles aux erreurs numériques (le calcul de la fraction continue p/q d'un réel nécessite essentiellement $\log_2 p + \log_2 q$ bits du réel), et une erreur trop importante conduirait à un processus de réduction incorrect, et donc potentiellement à un résultat faux.

J'ai donc choisi de grouper les différents travaux effectués en deux parties selon la problématique à laquelle ils s'intéressent, efficacité ou fiabilité.

2.2 Efficacité

Le contenu de cette partie expose deux articles, l'un en commun avec Michel Quercia et Paul Zimmermann [33], le second en commun avec Paul Zimmermann [36], ainsi qu'un certain nombre d'idées encore à l'état de réflexions, résultant pour la plupart de conversations avec Paul Zimmermann.

2.2.1 Produit médian

Introduction

Le produit médian répond à une question qui se pose fréquemment en arithmétique, à savoir comment éviter de recalculer une quantité que l'on connaît déjà. Une des situations où cela arrive le plus fréquemment est sans doute l'itération de Newton

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)},$$

f étant une fonction suffisamment régulière et x_0 un nombre réel fixé. Sous un certain nombre de jeux d'hypothèses différents, cette suite est bien définie et converge *quadratiquement* vers un zéro l de la fonction f, c'est à dire que

$$-\log|x_n - l| \gg 2^n.$$

Il s'ensuit que dans la formule ci-dessus, le terme x_n devrait être considéré comme un terme principal, puisque $f(x_n)$ a vocation à être petit ; on peut en tirer des leçons sur le fait qu'il faut augmenter la précision de calcul à chaque étape, en pratique la doubler, x_n fournissant la partie haute du nombre x_{n+1} et le second terme fournissant la partie basse. On peut aussi remarquer que l'on connaît déjà la partie haute de $f(x_n)$, qui vaut 0 ; cependant, lors de l'évaluation de $f(x_n)$, cette partie haute s'annulera fréquemment à la suite de compensations. En bref, on va recalculer une valeur déjà connue, et on voudrait l'éviter.

Le produit médian est précisément la réponse à cette question pour l'itération de Newton pour l'inverse. Il a toutefois des implications pour la quasi-totalité des opérations arithmétiques. Nous discutons d'abord le cas de données de type formel, polynômes et séries formelles, avant de présenter les difficultés qui surgissent dans le cas de nombres flottants ou entiers et une ébauche de solution.

Définition

Dans le cas de l'inversion d'un polynôme A, l'itération de Newton devient

$$x_{k+1} = x_k + x_k(1 - Ax_k). \tag{2.1}$$

Si l'on veut une approximation en précision n, on suppose alors calculée une approximation en précision $n/2$; dès lors, la dernière étape consistera à évaluer le produit Ax_k, à supprimer la partie haute (qui vaut 1), et à multiplier la partie basse par l'opposé de x_k et ajouter le résultat à x_k. Pour obtenir la précision n sur x_{k+1}, Ax_k doit être évalué avec A en précision n ; il s'agit donc d'une multiplication déséquilibrée, que l'on effectue habituellement par deux multiplications $n/2 \times n/2$ (éventuellement, vu la situation, des produits courts). Le coût de la dernière itération est alors $3M(n/2)$, où $M(n)$ est le coût d'une multiplication en taille n ; soit, si $M(n) \asymp n^{\log_2 3}$ (Karatsuba) un coût total de $\frac{3}{2}M(n)$, et si $M(n)$ est quasi-linéaire (FFT), un coût total de $3M(n)$.

Toutefois, sur les deux multiplications $n/2 \times n/2$ de Ax_k, seuls les $n/2$ termes médians (sur 3n/2) nous intéressent. On définit le produit médian de A et de x_k comme étant précisément ces termes.

De façon précise, si R est l'anneau des coefficients de nos séries formelles, et que $y := [y_0, \ldots, y_{n-1}] \in R^n$, $z := [z_0, \ldots, z_{2n-2}] \in R^{2n-1}$, on définit $\mathtt{MP}(y,z) = [\sum_{i+j=k} y_i z_j]_{n-1 \le k \le 2n-2}$, où les séries formelles en précision k sont représentées comme des éléments de R^k.

Un cas simple

Nous présentons une méthode pour calculer le produit médian dans le cas $n = 2$, qui illustre bien la ressemblance avec un algorithme de multiplication (ici, Karatsuba) :

Algorithme `ProduitMédian`.
Entrée : $y = y_0 + y_1 t, A = z_0 + z_1 t + z_2 t^2$
Sortie : $h = z_0 y_1 + z_1 y_0$ and $l = z_1 y_1 + z_2 y_0$
1. $\alpha \leftarrow (y_0 + y_1) z_1$
2. $\beta \leftarrow y_1 (z_1 - z_0)$
3. $\gamma \leftarrow (y_1 + y_2) z_0$
4. $h \leftarrow \alpha - \beta$
5. $l \leftarrow \gamma + \beta$.

On peut maintenant, à l'instar de ce que l'on fait dans l'algorithme de Karatsuba, utiliser cette idée récursivement, pour obtenir l'algorithme suivant quand n est une puissance de 2 :

Algorithme $\mathtt{MP}([y_0,\dots,y_{n-1}],[z_0,\dots,z_{2n-2}])$.
0. Si $n = 1$, renvoyer $[z_0 y_0]$
1. $p \leftarrow n/2$.
2. $\alpha \leftarrow \mathtt{MP}([y_p,\dots,y_{2p-1}],[z_0+z_p,\dots,z_{2p-2}+z_{3p-2}])$
3. $\beta \leftarrow \mathtt{MP}([y_p-y_0,\dots,y_{2p-1}-y_{p-1}],[z_p,\dots,z_{3p-2}])$
4. $\gamma \leftarrow \mathtt{MP}([y_0,\dots,y_{p-1}],[z_p+z_{2p},\dots,z_{3p-2}+z_{4p-2}])$
5. Renvoyer $[\alpha-\beta,\gamma+\beta]$.

Calcul en $M(n)$

Il se trouve que le produit médian de deux séries formelles peut se calculer en temps $M(n)$, et non $2M(n)$ par une méthode naïve. Grâce à une remarque d'Éric Schost, nous nous sommes aperçus qu'il s'agissait d'un avatar du principe dit de *transposition* et communément attribué à Tellegen, qui affirme que d'un algorithme de complexité C donnée permettant de multiplier une matrice A par un vecteur quelconque, on peut déduire un algorithme de complexité C permettant de multiplier ${}^tA$ par un vecteur quelconque. Ce principe n'est pas effectif, et se prouve en « retournant des flèches » dans un circuit arithmétique implantant l'algorithme. Dans le cas présent, la multiplication par un polynôme x de degré $n-1$ donné envoie R^{n-1} dans R^{2n-2} ; l'application transposée envoie donc R^{2n-2} dans R^{n-1}, et est essentiellement le produit médian par x, comme on le voit en écrivant les matrices des transformations, ou dans l'identité suivante :

Théorème 16 *Si $M_{p,q,n} : R^p \times R^q \to R^n$ et $\Pi_{n,p,q} : R^n \times R^p \to R^q$ sont respectivement*

$$M_{p,q,n}(y,z) = \left(\sum_{j+k=i+p-1} y_j z_k \right)_{0 \le i < n} \qquad \Pi_{n,p,q}(x,y) = \left(\sum_{i+j=k} x_i y_j \right)_{0 \le k < q},$$

alors, pour $(X,Y,Z) \in R^n \times R^p \times R^q$, on a

$$(X \mid M_{p,q,n}(Y,Z)) = (\Pi_{n,p,q}(X,\tilde{Y}) \mid Z), \tag{2.2}$$

où $(\ |\ )$ est le produit scalaire canonique de deux vecteurs de même longueur, et où $\tilde{Y} = (y_{p-1-i})_{0 \le i < p}$.

En particulier, si l'on dispose d'une décomposition de $\Pi_{n,p,q}(Y,Z)$ sous la forme

$$\Pi_{n,p,q}(X,Y) = \sum_{m=1}^{\ell} (a_m \mid X)(b_m \mid Y) c_m \tag{2.3}$$

avec $a_m \in R^n$, $b_m \in R^p$ and $c_m \in R^q$, alors on peut en déduire une décomposition de $M_{p,q,n}(Y,Z)$ sous la forme

$$M_{p,q,n}(Y,Z) \quad = \quad \sum_{m=1}^{\ell} (b_m \mid \tilde{Y})(c_m \mid Z) a_m, \tag{2.4}$$

D'un point de vue pratique, les coûts de calcul de M et de Π sont identiques, à $q-p$ additions près. Dans le cas $p=n, q=2n-1$, un produit médian peut donc se calculer en $M(n)+n-1$, pour un algorithme de multiplication issu d'une identité du type (2.3) et de complexité $M(n)$.

Les algorithmes de type Karatsuba, Toom-Cook et FFT sont, en particulier, du type ci-dessus, et il est possible de leur appliquer cette transformation pour obtenir des algorithmes de produit médian efficaces. Comme Bernstein nous l'a signalé, dans le cas de la FFT, on peut arriver au même résultat en effectuant simplement trois transformées de Fourier de taille $2n$ – à condition de disposer d'une transformée de Fourier modulo $x^{2n}\pm 1$ –, pour un coût total de $M(n)$ également, mais cette astuce semble complètement ad hoc et ne s'applique pas à notre connaissance au cas des autres algorithmes de multiplications.

Applications

De cette méthode nous déduisons immédiatement un algorithme de complexité $M(n)$ (Karatsuba) ou $2M(n)$ (FFT) pour l'inversion. Combiné avec une astuce de Karp et Markstein, cela permet d'effectuer une division en temps $\frac{4}{3}M(n)$ (Karatsuba) ou $\frac{5}{2}M(n)$ (FFT). Néanmoins, on peut faire un peu mieux dans le premier cas, en remarquant que dans l'algorithme classique de division récursive de A par B, le calcul du terme reste obtenu après la division de la partie haute de A par la partie haute de B n'est rien d'autre que le produit médian A et du quotient calculé Q_h. On effectue ainsi une division de taille n en 2 divisions de taille $n/2$ et un produit médian de taille $n/2$, soit une division $D(n)$ en temps $M(n)$ (Karatsuba).

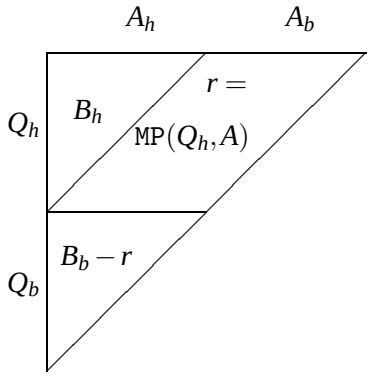


FIG. 2.1 – Division récursive et produit médian.

Classiquement, on perd un facteur logarithmique dans le cas de la FFT où la division récursive n'est pas intéressante. Des idées de même nature permettent d'améliorer la division avec reste, la racine carrée avec et sans reste, et le carré (des séries formelles, donc le carré court). Nous résumons les résultats obtenus dans la table ci-dessous.

Cette table appelle un dernier commentaire. Dans des résultats de complexité se concentrant sur la constante, comme c'est le cas ici, il est important qu'une implantation numérique vienne valider l'estimation, de façon à vérifier que les coûts négligés dans l'estimation (structures de contrôle du programme, typiquement) n'influent pas sur le résultat. Cela a été fait dans le cas de la multiplication de Karatsuba dans l'article [33], auquel nous renvoyons pour cette étude expérimentale comme pour les détails des différents algorithmes.

Le cas réel/entier

La situation dans le cas réel/entier est fortement dépendante de l'algorithme de multiplication sous-jacent. Dans le cas où il s'agit de la FFT, d'un point de vue théorique, il n'y a pas de différence avec

	Karatsuba		FFT	
	sans MP	avec MP	sans MP	avec MP
Carré	$\frac{2}{3}K(n)$	$\frac{1}{2}K(n)$	$\frac{2}{3}M(n)$	$\frac{2}{3}M(n)$
Produit médian	$2K(n)$	$K(n)$	$2M(n)$	$M(n)$
Inverse	$\frac{3}{2}K(n)$	$K(n)$	$3M(n)$	$2M(n)$
Division	$\frac{3}{2}K(n)$	$K(n)$	$\frac{7}{2}M(n)$	$\frac{5}{2}M(n)$
… avec reste	$2K(n)$	$\frac{5}{3}K(n)$	$\frac{9}{2}M(n)$	$\frac{7}{2}M(n)$
Racine carrée	$K(n)$	$\frac{3}{4}K(n)$	$\frac{7}{2}M(n)$	$3M(n)$
… avec reste	$\frac{3}{2}K(n)$	$\frac{5}{4}K(n)$	$\frac{9}{2}M(n)$	$4M(n)$

TAB. 2.1 – Comparaison des complexités obtenues pour diverses opérations avec et sans produit médian

ce qui précède : le problème des retenues est également inhérent à la FFT, et est déjà traité par cette dernière : on effectue en pratique la FFT sur des polynômes à coefficients entiers, puis on spécialise la variable en la base de représentation ; la même approche fonctionne sans modification pour le produit médian. On peut donc opérer de façon complètement identique.

Dans le cas où l'algorithme sous-jacent est de type Karatsuba ou Toom-Cook, la situation est plus complexe. La propagation des retenues, qui ne pose pas de problème dans le cas de la multiplication, casse l'algorithme.

Une première solution consiste à utiliser des mots de garde pour stocker les retenues. Dans la pratique, pour une taille d'opérande $n \leq 2^w$, un mot de garde supplémentaire suffit pour stocker les retenues, mais d'un point de vue asymptotique le coût global en espace, prenant en compte les appels récursifs, devient $n \log n$, et la gestion de ces mots de garde – qui, d'un point de vue asymptotique, aura également un coût en temps d'un probable facteur $\log\log n$ – risque de faire perdre d'un point de vue pratique le gain théorique escompté.

Il est également possible de « réparer » l'algorithme de la façon suivante :

– on effectue tout le calcul sans propager les retenues, mais en notant les retenues qui sont intervenues au cours du calcul ;
– on calcule un terme correctif en fin d'algorithme provenant des différentes retenues.

Un calcul élémentaire montre que ce terme s'évalue de la façon suivante :

Théorème 17 *Soit π la base de représentation, $Y = \sum_{i=0}^{n} y_i \pi^i$, et $(\delta_i)_{0 \leq i \leq 2n-2}$ le vecteur des retenues dans le calcul de $X + X'$. Alors*

$$M_n(X+X',Y) - M_n(X,Y) - M_n(X',Y) = \pi^{2n-1} \sum_{i+j=2n-2} y_i \delta_j - \pi^{n-1} \sum_{i+j=n-2} y_i \delta_j.$$

Ce calcul représente néanmoins un surcoût. Là encore, il ne paraît pas réaliste d'estimer le gain pratique par le résultat asymptotique obtenu dans le cas formel.

Ces deux stratégies – et, plus généralement, le produit médian – me semblent plus prometteuses dans le cas p-adique (essentiellement dans le cas p moyen, où l'on préférera parfois stocker le polynôme par sa représentation en base p plutôt que par une approximation dans $\mathbb{Z}/p^n\mathbb{Z}$), où l'on peut disposer de suffisamment de bits de garde pour stocker les retenues intervenant dans le calcul (ce qui favorise la première variante), et où l'on ne dispose pas nécessairement d'une primitive rapide pour propager les

retenues (ce qui ne défavorise pas la seconde). En tout état de cause, une étude expérimentale fine et poussée semble s'imposer.

2.2.2 Le produit court

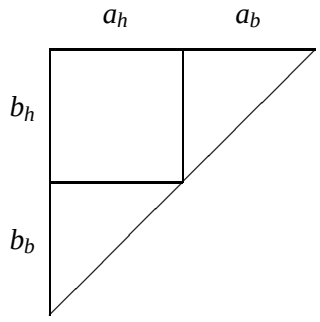
Dans cette partie, nous revenons sur l'exemple de la multiplication des séries formelles évoqué plus haut.

Définition

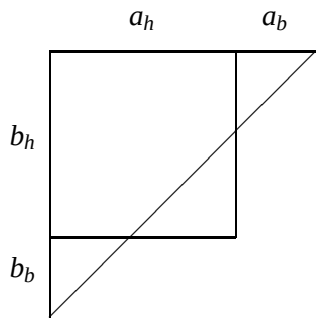
Le produit court de Mulders est la première réponse connue à la question suivante : est-il possible de calculer plus rapidement la partie « haute » d'un produit de deux séries formelles que le produit complet, en utilisant un algorithme de multiplication non quadratique ?

Si la réponse dans le cas de la FFT n'est toujours pas connue, le produit de Mulders apporte une réponse simple et élégante au cas de la multiplication de Karatsuba ou de Toom-Cook.

L'idée naturelle, plus claire sur le diagramme suivant, consiste à calculer le produit – complet – a_0b_0 des parties hautes des deux opérandes (coût $M(n/2)$), puis à effectuer deux produits courts a_0b_1 et a_1b_0 pour obtenir les parties basses. Toutefois, ce faisant, on retrouve exactement – en terme de nombre de multiplications – la complexité de la multiplication de Karatsuba (du moins si c'est l'algorithme de multiplication sous-jacent).



La solution surprenante proposée par Mulders consiste à calculer un produit complet plus important, de taille βn avec $\beta > 1/2$, puis deux produits courts de taille $(1-\beta)n$, puis d'optimiser β en supposant $M(n) = c \cdot n^\theta$.



On trouve sous cette hypothèse un produit court en

$$S(n) = \frac{\beta^\theta}{1-2(1-\beta)^\theta} M(n),$$

avec un optimum atteint pour $\beta = 1 - 2^{-1/(\theta-1)}$. Pour $\theta = 2$, on retrouve la méthode donnée plus haut ; pour $\theta = \log_2 3$, on trouve $\beta = 0.694...$, donnant une amélioration attendue de 20% environ.

Optimalité dans la méthode de Mulders

L'analyse présentée ci-dessus est toutefois assez crue. En effet, la complexité de la multiplication de Karatsuba n'est que grossièrement décrite par l'estimation $K(n) \asymp n^{\log_2(3)}$. *A contrario*, il est préférable d'utiliser la relation de récurrence

$$K(n) = 2K(\lceil n/2 \rceil) + K(\lfloor n/2 \rfloor), \tag{2.5}$$

où $\lceil x \rceil$ désigne le plus petit entier supérieur ou égal à x, et $\lfloor x \rfloor$ la partie entière.

Il reste néanmoins possible, avec ce modèle plus complexe, de déterminer le point de coupure optimal dans la méthode de Mulders :

Théorème 18 *Pour la multiplication de Karatsuba, le point de coupure minimisant le nombre de multiplications élémentaires dans l'algorithme de Mulders est obtenu pour $k = 2^{\lfloor \log_2 n \rfloor}$, et le nombre de multiplications correspondant est donné par la récurrence*

$$S(n) = S(\lceil n/2 \rceil) + 2S(\lfloor n/2 \rfloor).$$

La preuve de ce résultat est de nature combinatoire, en vérifiant que la récurrence annoncée est bien atteinte avec la coupure indiquée, et qu'inversement, aucune coupure ne peut produire une meilleure complexité.

En comptant le nombre de multiplications, on trouve que le rapport avec un produit complet de Karatsuba se situe entre 1 et (expérimentalement) 3/5, avec une moyenne aux environs de 0.71.

Néanmoins, si l'expérience valide ces résultats de manière tout à fait convaincante dans le cas où la multiplication dans l'anneau de base est très coûteuse, à l'inverse, le résultat ne semble pas satisfaisant dans le cas où, par exemple, on essaie de l'appliquer à des nombres réels ou à des séries formelles sur $\mathbb{F}_p$, p petit.

Cela tient au fait que ce résultat est extrêmement sensible aux conditions initiales, et que dans la pratique, il est souvent pertinent d'utiliser des algorithmes de type quadratique pour les petites tailles d'opérandes, ce qui fausse complètement les choses.

Notons plus précisément $K(n)$ le nombre de multiplications élémentaires pour multiplier deux opérandes de taille n, et $S(n)$ le nombre de multiplications élémentaires effectuées pour calculer un produit court. Si l'on suppose en fait que

$$K(n) = n^2, \quad S(n) = n(n+1)/2 \tag{2.6}$$

pour n petit, nous ne savons plus traiter le problème de façon aussi complète que précédemment. Toutefois, le caractère expérimentalement presque autosimilaire du graphe de la valeur optimale conduit à suggérer de déterminer le point de coupure optimale pour les petites valeurs de n, et d'utiliser ensuite l'approximation $\text{coupure}(2n) = 2\text{coupure}(n)$.

Dans la suite, nous supposerons donné un entier n_0 tel que pour $n \leq n_0$, on ait les relations (2.6) et pour $n > n_0$, on ait (2.5).

Une méthode pair-impair

Une solution plus satisfaisante dans un certain contexte revient à proposer la modification suivante à l'idée de Mulders : si l'on considère une version de la méthode de Karatsuba dans laquelle le découpage

séparerait parties paires et impaires, et non plus hautes et basses, on peut formuler de façon totalement élémentaire le produit court.

Notons $P = P_1(t^2) + tP_2(t^2), Q = Q_1(t^2) + tQ_2(t^2)$, où P et Q sont des polynômes de degré $n-1$. Alors le produit court PQ est donné par les n termes de bas degré de

$$PQ = P_1Q_1(t^2) + t(P_2Q_1 + P_1Q_2)(t^2) + t^2P_2Q_2(t^2).$$

Il nous faut donc déterminer

- les n termes de bas degré de $P_1Q_1(t^2)$, qui s'obtiennent via un produit court de P_1 par Q_1, de taille $\lceil \frac{n}{2} \rceil$;
- les $n-1$ termes de bas degré de $\{(P_1+P_2)(Q_1+Q_2) - P_1Q_1 - P_2Q_2\}(t^2)$; si les $n-1$ termes of $P_1Q_1(t^2)$ and $P_2Q_2(t^2)$ sont connus, cela revient à un produit court de (P_1+P_2) par (Q_1+Q_2), de taille $\lceil \frac{n-1}{2} \rceil$;
- enfin, $n-2$ termes de $P_2Q_2(t^2)$; toutefois, pour calculer le terme médian, il nous faut en fait $n-1$ termes, que l'on obtient par un produit court de P_2 par Q_2 d'ordre $\lceil \frac{n-1}{2} \rceil$.

Il vient donc la récurrence

$$S'(n) = S'\left(\left\lceil \frac{n}{2} \right\rceil\right) + 2S'\left(\left\lfloor \frac{n}{2} \right\rfloor\right),$$

qui montre que l'on retrouve la complexité de l'algorithme de Mulders avec point de coupure optimal.

Néanmoins, cette stratégie et surtout cette récurrence sur sa complexité, sont indépendantes de ce qui est fait pour les petites valeurs de n. Nous pouvons même alors mesurer l'intérêt asymptotique de cette variante :

Théorème 19 *Si $n_0 \leq 6$, ou $n_0 = 8$, alors pour $n \geq n_0$,*

$$S'(n)/K(n) \leq \frac{1}{2} + \frac{1}{2\lfloor n_0/2 \rfloor + 2}.$$

Sinon, si $\delta = K(n_0)/K(n_0+1) - 1 < 0$, pour tout j on a

$$S'(n)/K(n) \leq \exp\left((2/3)^j 15\delta/4\right) \max_{n \in [2^j n_0, 2^{j+1} n_0 - 1]} S'(n)/K(n).$$

En prenant $j = 16$, dans le théorème, nous obtenons les bornes suivantes pour $S'(n)/K(n)$ en fonction de n_0 :

n_0	8	16	24	32
B	0.6	0.559	0.5608	0.5628

Une des leçons de ce théorème est le fait que pour n_0 de taille moyenne, on obtient presque un gain d'un facteur 2 *dans le pire des cas* en termes de nombre de multiplications pour le calcul du produit court sur le produit complet. On préserve donc presque complètement le gain obtenu sur les valeurs de $n \leq n_0$.

Le cas réel

Dans le cas réel, la stratégie décrite ci-dessus échoue, car la propagation des retenues ne suit bien sûr pas du tout le schéma pair/impair : une retenue issue de l'addition de deux termes d'ordre k devrait être propagée au terme $k+1$, et non $k+2$.

Néanmoins, des idées similaires au cas du produit médian flottant doivent encore pouvoir s'appliquer dans ce cas, consistant à isoler les retenues et à traiter leur propagation en fin de calcul. Toutefois, les mêmes réserves pratiques s'appliquent. Notons de surcroît que le découpage pair-impair, s'il peut être réaliste pour des structures de données complexes où l'arithmétique proprement dite domine largement les accès à la mémoire, effectue ses accès à la mémoire de façon peu connectée, et donc *a priori* inefficace, dans le cas de nombres réels.

2.2.3 Quelques prolongements

L'analyse combinatoire appliquée à la méthode de Mulders se tranpose à diverses situations, comme l'optimisation du calcul d'un produit de n polynômes de degré 1 (ou d'entiers de taille fixée). Il est possible également de mener la même analyse dans le cas de la multiplication de Toom-Cook. L'analyse permet également presque sans modification de traiter le cas où l'on cherche à calculer un produit de polynômes linéaires, et donc à l'analyse des méthodes d'évaluation-interpolation.

En outre, l'analyse des différents algorithmes de multiprécision montre que l'idée clé est généralement simplement le fait que l'efficacité maximale n'est obtenue qu'en multipliant des opérandes de même taille. Toutefois, le problème de la multiplication d'opérandes de taille différente est peu étudié, et en parcourant le code de divers systèmes on découvre qu'il est traité de façon très variable. Dans le cas de la multiplication de Karatsuba, une analyse précise montre que les idées pair/impair, combinées à une idée de décalage, permettent de gagner de façon parfois significative. Il convient toutefois de remarquer, une fois de plus, que l'amélioration concerne prioritairement le cas où les produits dans l'anneau de base sont très coûteux (sans quoi la stratégie pair/impair n'est pas pertinente). En tout état de cause, une étude expérimentale complémentaire des différentes stratégies semble s'imposer.

2.3 Fiabilité

2.3.1 Étude d'un modèle de calcul flottant

Cette partie rend compte d'un travail commun avec Joël Rivat, Gérald Tenenbaum et Paul Zimmermann [34].

Dans l'article [45], Muller avait entrepris une étude formelle d'une version simplifiée de la généralisation en précision arbitraire du modèle IEEE 754. On définit :

$$F_k := \left\{ m2^e : m,e \in \mathbb{Z},\ 2^{k-1} \le |m| < 2^k \right\} \cup \{0\}.$$

l'ensemble des nombres représentables en précision k, avec l'opération d'arrondi « au plus proche » (et arrondi pair en cas d'égalité). Bien évidemment, il n'y a aucune raison que les fonctions calculées dans ce cadre conservent leurs propriétés algébriques. En particulier, si $y = \diamond(1/x)$, il n'y a pas vraiment de raison pour laquelle $\diamond(xy)$ serait égal à 1. Inversement, on peut imaginer que pour un certain $y' \neq \diamond(1/x)$, on ait néanmoins $\diamond(xy') = 1$.

De façon générale, nous dirons que $y' \in F_k$ est un inverse flottant de $x \in F_k$ si cette dernière propriété est vérifiée. Muller avait montré qu'un élément de F_k pouvait avoir plusieurs inverses flottants, tel $3/2$ dans F_5 qui admet $21/32$ et $11/16$ comme inverses, et énonçait une conjecture sur la fréquence de cette situation.

Il n'est pas très difficile de constater qu'il n'est pas possible d'avoir plus de deux inverses flottants. Nous avons alors prouvé le résultat suivant :

Théorème 20 *Pour $r = 0,1,2$, soit $\gamma_r(k)$ le nombre de $x \in F_k \cap [1,2[$ ayant exactement r inverses flottants. Alors*

$$\begin{aligned}
\gamma_0(k)/2^{k-1} &= \tfrac{1}{2} - \tfrac{3}{2}\log\tfrac{4}{3} + O(2^{-k/3}) = 0.0684768917\ldots + O(2^{-k/3}) \\
\gamma_1(k)/2^{k-1} &= 1 - \tfrac{3}{2}\log\tfrac{9}{8} + O(2^{-k/3}) = 0.8233254464\ldots + O(2^{-k/3}) \\
\gamma_2(k)/2^{k-1} &= -\tfrac{1}{2} + \tfrac{3}{2}\log\tfrac{3}{2} + O(2^{-k/3}) = 0.1081976622\ldots + O(2^{-k/3})
\end{aligned}$$

L'argument consiste à remarquer qu'un élément admet un inverse si et seulement s'il existe $y \in F_k$ tel que $1-2^{-k-1} \leq xy \leq 1+2^{-k}$. Le nombre d'inverses est alors le nombre d'entiers de l'intervalle

$$\left] \frac{2^{2k-1}-2^{k-2}-1}{m}, \frac{2^{2k-1}+2^{k-1}}{m} \right],$$

où m est la mantisse de x (intervalle dont on vérifie immédiatement qu'il contient au plus deux entiers), et nous voulons donc estimer

$$\sum_{2^{k-1} \leq m < 2^k} \left\lceil \frac{2^{2k-1}+2^{k-1}}{m} \right\rceil - \left\lceil \frac{2^{2k-1}+2^{k-2}}{m} \right\rceil.$$

Le terme principal est celui obtenu en enlevant les parties entières (ce qui avait conduit Muller à sa conjecture), et le terme reste se traite comme celui du classique problème des diviseurs.

La question analogue sur la racine carrée inverse conduit alors au résultat suivant, qui se prouve par des arguments analogues.

Théorème 21 *Tout élément de F_k admet au plus une racine carrée inverse.*
Le nombre $\delta^+(k)$ d'éléments $x \in F_k \cap [1,2[$ ayant une racine carrée inverse vérifie

$$\delta^+(k)/2^{k-1} = \frac{3\sqrt{2}-3}{2} + O(2^{-k/3}) = 0.621320343\ldots + O(2^{-k/3}).$$

Théorème 22 *Le nombre $\delta^-(k)$ d'éléments $x \in F_k \cap [\frac{1}{2},1[$ admettant une racine carrée inverse vérifie*

$$\delta^-(k)/2^{k-1} = \frac{3\sqrt{2}-3}{2\sqrt{2}} + O(2^{-k/3}) = 0.4393398278\ldots + O(2^{-k/3}).$$

2.3.2 La bibliothèque MPFR

La bibliothèque MPFR est un projet né sous l'impulsion de Paul Zimmermann, avec la volonté d'une part d'étudier la façon dont la norme IEEE 754 s'étendrait en précision supérieure, et surtout de montrer que le coût de la fiabilité en termes d'efficacité pouvait, du moins en multiprécision, être contenu dans des limites raisonnables.

Un certain nombre de personnes ont participé au projet, à des moments et des degrés divers. Citons David Daney, Laurent Fousse, Vincent Lefèvre, Patrick Pélissier et Paul Zimmermann, qui ont tous contribué de façon significative au code.

Diverses expériences ont été conduites en parallèle, sur des concepts voisins, dans d'autres groupes de recherche, alors que par le passé, l'efficacité semblait être le critère principal de conception des bibliothèques multiprécision. Citons par exemple Arithmos, iRRAM, ou encore le concept de « *significant arithmetic* » dans Mathematica.

MPFR est conçue sur des bases très voisines de la généralisation naturelle du standard IEEE-754 à la précision arbitraire, à quelques exceptions près[8]. Rappelons que le concept-clé de cette norme est la notion d'arrondi correct, *i.e.* on ne se préoccupe pas de la propagation des erreurs numériques – tâche qui incombe au programmeur – mais plutôt de la sémantique précise de chaque calcul individuel. Notons qu'il est alors possible de construire une arithmétique d'intervalles au-dessus d'une telle arithmétique, en jouant sur les différents modes d'arrondi. Dans le cas de MPFR, cela a été fait par N. Revol et F. Rouillier, cf. la bibliothèque MPFI.

[8] essentiellement le problème des nombres dénormalisés qui, bien qu'utiles pour assurer certaines propriétés de consistance telles $x \neq y \Rightarrow x-y \neq 0$ lors du calcul, sont dans notre cas d'une occurrence potentielle suffi samment rare – ce sont des nombres $< 2^{-2^{31}}$ – pour essayer de s'en dispenser. Dans le cas d'une plage d'exposants restreinte (sous-ensemble strict de $[-2^{31}, 2^{31}[$), ils sont pris en charge)

Principes d'implantation

Nous avons choisi de nous baser sur la couche « bas niveau » `mpn` de GMP. Ceci nous a conduits à choisir une représentation interne en base 2^k, où k est la taille du mot machine. L'idée-clé est qu'à chaque variable est attachée une précision en bits, qui peut varier de 2 à $2^{32}-1$ dans l'implantation actuelle. Le calcul est alors effectué selon la règle générale décrite en introduction, le résultat, ayant une précision pré-affectéé, doit être un nombre représentable dans cette précision. Noter qu'une fonction MPFR renvoie en outre une valeur ternaire, $\{-1,0,1\}$, selon que le résultat exact de l'opération est plus petit, égal, ou supérieur à la valeur renvoyée.

La principale difficulé consiste alors à décider efficacement de l'arrondi correct, tout en s'efforçant d'adopter l'algorithme le plus efficace possible.

Une autre des propriétés de la norme IEEE-754 qu'il nous a semblé souhaitable de conserver est la portabilité : l'implantation est faite de sorte que le même calcul reproduit sur deux machines différentes produise le même résultat ; la principale difficulté sur ce point réside dans la comparaison de machines de taille de mot différent (32/64 bits).

Représentation interne

Chaque nombre multiprécision est représenté par
- sa mantisse, m entier multiprécision ;
- son signe $\varepsilon \in \{-1,0,1\}$,
- son exposant e (entier long) ;
- sa précision p (entier long).

La valeur du nombre est alors $\varepsilon m 2^{e-p}$.

Opérations élémentaires

Pour les opérations élémentaires, nous sommes face à un choix ; il nous est toujours possible d'effectuer le calcul de façon exacte, puis, au vu du résultat, d'arrondir simplement. Cette idée générique peut toutefois se révéler redoutablement inefficace. Pour donner deux exemples, imaginons d'abord l'addition de deux entiers d'ordres de grandeur très différents, mais aussi le cas où l'on veut effectuer la multiplication ou la division de deux nombres en très grande précision, le résultat n'étant souhaité qu'en très petite précision.

Dans ces cas, une bonne approximation vaut mieux qu'un coûteux calcul exact. Le principe est alors de conserver trace de l'erreur, de sorte qu'implicitement nous savons situer le résultat exact de l'opération dans un intervalle $[x-\alpha, x+\alpha]$. Si, dans le mode d'arrondi choisi, les deux bornes de cet intervalle s'arrondissent à la même valeur, la monotonie des différentes fonctions d'arrondi nous permet de garantir que cette valeur est l'arrondi correct du résultat de l'opération. Dans le cas contraire, on est amené à raffiner le calcul. Le seul cas où cette stratégie échoue ultimement (pour les fonctions transcendantes) est le cas d'un résultat exact, cas qui doit être isolé et traité à part.

Typiquement, pour la multiplication, on peut imaginer d'utiliser une stratégie de produit court, sans doute pas aussi agressive que celle présentée dans l'introduction de ce chapitre, mais calculant les $n+\delta$ termes de poids fort du produit ; l'erreur est alors dominée par

$$\sum_{k=n+\delta+1}^{2n-2} (2n-1-k)2^{-k} = (n-\delta-3)\cdot 2^{-n-\delta} + 2^{-2n+2}.$$

Si le résultat corrigé par l'erreur maximale s'arrondit de la même façon que le résultat brut, le calcul grossier suffit pour conclure.

En particulier, pour un certain δ, l'erreur influe essentiellement à partir du bit $n+\delta-\log_2(n-\delta-3) \approx n+\delta-\log_2 n$; il n'y aura donc d'incidence sur l'arrondi correct que dans le cas où les bits du résultat partiel situés entre la position n et la position ci-dessus sont tous nuls, ou tous égaux à 1 (resp. la position $n+1$ dans le cas de l'arrondi au plus proche).

On peut estimer la probabilité de cet événement par $(n-\delta-3)2^{-\delta}$. Il reste alors à choisir δ en équilibrant le coût du calcul et l'espérance du coût du recalcul éventuel.

Avec cette stratégie, il est même envisageable d'implanter le produit de Mulders tout en préservant la garantie de l'arrondi correct. Cela a été fait par Patrick Pélissier pour la dernière version de mpfr.

Fonctions

Une stratégie du même type peut être mise en œuvre pour les fonctions transcendantes, à l'exception du fait qu'hormis dans certains cas particuliers (typiquement, la racine carrée), on ne dispose plus de méthode exacte. On utilise alors la *stratégie de Ziv*, qui consiste à procéder comme précédemment, en augmentant la précision par palier tant que l'arrondi ne peut être décidé.

Les pires cas pour la stratégie de Ziv sont les x représentables pour lesquels $f(x)$ (valeur exacte) est très voisin d'un nombre représentable (arrondis dirigés et vers zéro), ou du milieu de deux nombres représentables (arrondi au plus proche) ; en particulier, la précision supplémentaire requise pour le calcul avant l'arrondi est essentiellement gouvernée par le nombre de bits consécutifs égaux après la partie significative dans le développement binaire de $f(x)$.

Un modèle probabiliste naturel montre alors que le « pire cas » en précision p devrait nécessiter une précision supplémentaire de

$$\sum_{r\geq 0}\left(1-(1-2^{-r})^N\right) = \log_2 N + O(1),$$

en moyenne où N est le nombre de nombres représentables en précision p, soit $2^{p+e_{\max}-e_{\min}+1}$, où $e_{\max}$ et $e_{\min}$ désignent respectivement les exposants maximum et minimum. On peut donc imaginer utiliser directement une précision un peu supérieure à $2p+e_{\max}-e_{\min}+1$, de façon à éviter tout recalcul.

Ce modèle probabiliste a bien entendu des défauts (il n'est par exemple pas réaliste si $f'(x)=0$ pour un certain x représentable, voir par exemple $\cos(x)$, et devient franchement aberrant pour $\cos(x^2)$; il convient alors d'adapter l'analyse au voisinage de ce type de points pour prendre en compte la spécificité du cas étudié), mais fournit une idée assez fiable du pire cas pour la stratégie de Ziv, et se compare assez bien aux résultats expérimentaux dans le cas où des approches de type plus ou moins exhaustif ont été tentées [40, 50]. Le seul écueil de la stratégie de Ziv est la présence de résultats exacts (tels $\sin 0, \log 1$), mais pour la plupart des fonctions spéciales, ceux-ci sont en très petit nombre et connus.

Notons que le modèle permet également de prévoir le surcoût en moyenne de la stratégie de Ziv (et de choisir les paramètres pour son implantation) : la probabilité de ne pas pouvoir arrondir avec r bits de précision supplémentaire pour arrondir est 2^{1-r}. Une stratégie où l'on calculerait à des paliers successifs de précision (sans réutilisation des résultats précédents) $p_1,\ldots,p_\infty$ conduit alors à une complexité globale de

$$\sum_{k\geq 1}\left(2^{1-p_{k-1}}-2^{1-p_k}\right)\sum_{0\leq l\leq k} C(p+p_l) = \sum_{k\geq 1} 2^{1-p_k} C(p+p_k),$$

où $C(n)$ est le coût d'une évaluation de la fonction en précision n.

La pratique montre que moyennant un bon paramétrage, le surcoût en moyenne de la stratégie de Ziv en multiprécision est modeste : dans la pratique, on ajoute un mot machine au plus, et on « peut arrondir » dans l'immense majorité des cas.

2.3.3 Quelques prolongements

Une étude expérimentale plus poussée de la stratégie de Ziv, combinant optimisation du modèle probabiliste et étude pratique sur MPFR permettrait peut-être d'en comprendre un peu mieux la mise en œuvre optimale, et le rôle des pires cas. En tout état de cause, MPFR constitue un terrain d'expérience idéal et continuera encore probablement de s'enrichir de nouvelles fonctions et de nouveaux algorithmes pendant quelque temps...

2.3.4 Une proposition de standard

Ce paragraphe reprend un article publié dans [26].

La réflexion sur l'évaluation des fonctions transcendantes menée au sein du projet Arénaire de longue date, et de façon plus récente au sein du projet Spaces, a conduit un groupe issu de ces projets, sous l'impulsion de Jean-Michel Muller, à réfléchir à une normalisation des fonctions transcendantes au sein de la norme IEEE-754. Cette normalisation constituait dans notre esprit un point de départ pour un débat sur le sujet au sein du comité de révision de la norme.

Il convient tout d'abord de préciser que la norme IEEE-754, dans sa version actuelle, n'impose l'arrondi correct que pour les opérations élémentaires et la fonction racine carrée.

Ce problème de normalisation présente néanmoins des difficultés significatives, à savoir qu'il est impossible de préserver à la fois les propriétés mathématiques souhaitées (symétrie, préservation de l'intervalle d'arrivée, etc.) et l'arrondi correct. Ainsi, même en arrondi au plus proche, on ne peut garantir que l'arrondi correct de $\arctan(x)$ soit bien dans l'intervalle $]-\pi/2, \pi/2]$ (ce n'est d'ailleurs pas le cas en simple précision). La raison en est simplement que $\arctan(\texttt{Inf})$ doit renvoyer l'arrondi de $\pi/2$. Arrondi vers $+\infty$, on obtient une valeur hors de l'intervalle ; ce sera également être le cas en arrondi au plus proche en précision p dès que $\{2^p\pi/4\} > 1/2$, où $\{x\}$ désigne la partie fractionnaire (par exemple, pour $p = 26$ (simple précision) mais pas pour $p = 53$ (double précision)).

De plus, les arrondis dirigés ne préservent pas les symétries (par construction).

Notre proposition s'efforce de tenir compte de ces impossibilités mutuelles en offrant différentes options à l'utilisateur (préservation de l'arrondi, de l'intervalle, ou de la symétrie), en lui laissant la liberté de choisir la possibilité à laquelle il est le plus attaché. Pour l'arrondi, toutefois, il convient de préciser les choses. Nous proposons, pour tenir compte de la difficulté de l'évaluation des fonctions transcendantes avec arrondi correct, les trois niveaux de qualité suivants :

- Niveau 0 : arrondi fidèle (i.e., « cohérent » : en arrondi au plus proche, le résultat est un des deux nombres machine encadrant le nombre réel ; en arrondi dirigé, l'arrondi est dans la bonne direction). Erreur relative tolérée d'1.5 ulp (*unit in the last place*). On demande que les fonctions restent monotones, et la préservation de la symétrie en arrondi au plus proche et vers 0.
- Niveau 1 : niveau 0 + arrondi correct dans un intervalle fixé (voisinage de 0, typiquement).
- Niveau 2 : arrondi correct (toujours avec la monotonie), pour tous les arguments représentables de la fonction.

Nous espérons que cette proposition attirera l'attention de la communauté sur le problème de la normalisation du calcul des fonctions transcendantes, qui nous semble, au vu des travaux récents, à la fois réalisable au vu des récents résultats, et suffisamment important pour être mis en œuvre. Un premier pas dans cette direction est attesté par le document `http://754r.ucbtest.org/subcommittee/trans.pdf`.

Bibliographie

[1] A. BAKER, Linear forms in the logarithms of algebraic numbers I, *Mathematika* **13** (1966), 204–216 ; II, ibid. **14** (1967), 102–107 ; III, ibid. **14** (1967), 220–228 ; IV, ibid. **15** (1968), 204–216.

[2] A. BAKER, Contributions to the theory of Diophantine equations. I. On the representation of integers by binary forms, *Philos. Trans. Roy. Soc. London Ser. A* **263** (1968), 173–191 ; II. The Diophantine equation $y^2 = x^3 + k$, ibid. **263** (1968), 193–208.

[3] E. BACH, Explicit bounds for primality testing and related problems, *Math. Comp.* **55** (1990), 355–380.

[4] A. BAKER, H. DAVENPORT, The equations $3x^2 - 2 = y^2$ and $8x^2 - 7 = z^2$, *Quart. J. Math. Oxford (2)* **20** (1969), 129–137.

[5] M. BENNETT, Rational approximation to algebraic numbers of small height : The diophantine equation $| ax^n - by^n | = 1$, *J. reine angewandte Math.* **535** (2001), 1–49.

[6] M. BENNETT, Products of consecutive integers, *Bull. London Math. Soc.*, à paraître.

[7] M. BENNETT, K. GYŐRY AND L. HAJDU, Powers from products of consecutive terms in arithmetic progressions, *J. Reine Angew. Math.*, à paraître.

[8] M. BENNETT, C. SKINNER, Ternary Diophantine equations via Galois representations and modular forms, *Canad. J. Math* **56** (2004), 23–54.

[9] M. BENNETT, B.M.M. DE WEGER, On the Diophantine Equation $|ax^n - by^n| = 1$, *Math. Comp.*, **67** (1998), 413–438.

[10] YU. BILU, Solving Superelliptic Diophantine Equations by Baker's method, prépublication.

[11] YU. BILU, Quantitative Siegel's theorem for Galois coverings, *Compositio Math.* **106** (1997), 125-158.

[12] YU. BILU, On Le's and Bugeaud's Papers about the Equation $ax^2 + b^{2m-1} = 4c^p$, *Monatsh. Math.* **137**, 1–3.

[13] YU. BILU, G. HANROT, Solving Thue Equations of High Degree, *J. Number Th.* **60** (1996), 373–392.

[14] YU. BILU, G. HANROT, Solving Superelliptic Diophantine Equations by Baker's method, *Compositio Math.* **112** (1998), 273–312.

[15] YU. BILU, G. HANROT, Thue Equations with Composite Fields, Acta Arithmetica **88** (1999), 311–326.

[16] YU. BILU, G. HANROT, P. VOUTIER, Existence of Primitive Divisors of Lucas and Lehmer Sequences (with an appendix by M. Mignotte), *J. Reine Angew. Math.* **539** (2001), 75–122.

[17] YU.F. BILU, M. KULKARNI, B. SURY, The Diophantine equation $x(x+1)...(x+(m-1))+r = y^n$, *Acta Arith.* **113** (2004), 303-308.

[18] N. BRUIN, Chabauty methods and covering techniques applied to generalized Fermat equations, CWI Tract 133 (2002).

[19] Y. BUGEAUD, Bounds for the solutions of superelliptic equations, *Compositio Math.* **107** (1997), 187–219.

[20] Y. BUGEAUD, On some exponential Diophantine equations, *Monatsh. Math.* **132**, 93–97.

[21] Y. BUGEAUD, K. GYŐRY, Bounds for the solutions of Thue-Mahler equations and norm form equations, *Acta Arith.* **74** (1996), 273-292.

[22] Y. BUGEAUD, G. HANROT, Un nouveau critère pour l'équation de Catalan, *Mathematika* **47** (2000), 63–73.

[23] Y. BUGEAUD, G. HANROT, M. MIGNOTTE, Sur l'équation diophantienne $y^q = (x^n - 1)/(x - 1)$, III, *Proc. London Math. Soc.* **84** (2002), 59–78.

[24] Y. BUGEAUD, T. SHOREY, On the number of solutions of the generalized Ramanujan-Nagell equation, *J. reine angew. Math.* **539** (2001), 55–74.

[25] P.D. CARMICHAEL, On the numerical factors of the arithmetic forms $\alpha^n \pm \beta^n$, *Ann. Math. (2)*, **15** (1913), 30–70.

[26] D. DEFOUR, G. HANROT, V. LEFÈVRE, J-M. MULLER, P. ZIMMERMANN Proposal for a standardization of mathematical function implementation in floating-point arithmetic, *Numerical Algorithms*, **37** (2004), pp 367–375.

[27] E. CATALAN Extrait d'une lettre adressée à l'éditeur par M. Catalan, Répétiteur à l'école polytechnique de Paris. *J. reine angew. Math.* **27** (1844), p. 192.

[28] H. DARMON, L. MEREL, Winding quotients and some variants of Fermat's Last Theorem *J. reine angew. Mathematik* **490** (1997), 81–100.

[29] V. FLYNN, A flexible method for applying Chabauty's Theorem, *Compositio Math.* **105** (1997), 79–94.

[30] I. GAÁL, M. POHST, On the resolution of relative Thue equations, *Math. Comp.* **71** (2002), 429–440.

[31] G. HANROT, Solving Thue Equations without the Full Unit Group, *Math. Comp.* **69** (2000), 395–405.

[32] G. HANROT, F. MORAIN. Solvability by radicals from an algorithmic point of view, Proceedings of the 2001 International Symposium on Symbolic and Algebraic Computation (2001), 175–182.

[33] G. HANROT, M. QUERCIA, P. ZIMMERMANN, The Middle Product Algorithm, I. Speeding up the division and square root of power series. *Appl. Alg. Eng. Comm. Comp.* **14** (2004), 415–438.

[34] G. HANROT, J. RIVAT, G. TENENBAUM, P. ZIMMERMANN, Density results on floating-point invertible numbers, *Theoret. Comput. Sci.* **291**, 135–141.

[35] G. HANROT, N. SARADHA, T.N. SHOREY, Almost perfect powers in consecutive integers, *Acta Arith.* **99** (2001), 13–25.

[36] G. HANROT, P. ZIMMERMANN. A long note on Mulders' short product. *J. Symb. Comput.* **37** (2004), 391–401.

[37] C. HEUBERGER, A. TOGBÉ, V. ZIEGLER, Automatic solution of families of Thue equations and an example of degree 8, *Journal of Symbolic Computation* **38** (2004), 1145–1163.

[38] D. HILBERT, Gesammelte Abhandlungen, Band 3, Chelsea publishing company (1933).

[39] M. LAURENT, M. MIGNOTTE, Y. NESTERENKO, Formes linéaires en deux logarithmes et déterminants d'interpolation, *J. Number Th.* **65** (1995), 285–321.

[40] V. LEFÈVRE, J.-M. MULLER. *Worst cases for correct rounding of the elementary functions in double precision.* In NEIL BURGESS AND LUIGI CIMINIERA, editors, Proceedings of the 15th IEEE Symposium on Computer Arithmetic, pages 111–118, Vail, Colorado, 2001. IEEE Computer Society Press.

[41] YU. MATIJASEVIČ, Enumerable sets are diophantine, *Dokl. Akad. Nauk. SSSR* **191** (1970) (Russe) ; version anglaise améliorée : *Soviet Math. Doklady* **12** (1971), 249–54.

[42] E. M. MATVEEV, An explicit lower bound for a homogeneous rational linear form in logarithms of algebraic numbers, II, *Izv. Ross. Akad. Nauk. Ser. Math.* **64** (2000), 125–180.

[43] P. MIHAILESCU, A class number free criterion for Catalan's conjecture, *J. Number Theory* **99** (2003), 225-231.

[44] P. MIHAILESCU, Primary cyclotomic units and a proof of Catalan's conjecture. *J. reine angew. Math.* **572** (2004), 167–195

[45] J.-M. MULLER, *Some algebraic properties of floating-point numbers*, in Proceedings of the 4th Real Numbers and Computers conference, Dagstuhl, P. Kornerup, ed., April 17–19, 2000, pp. 31–38.

[46] K.A. RIBET, On the equation $a^p + 2^\alpha b^p + c^p = 0$, *Acta Arith.* **79** (1997), 7–16.

[47] K.F. ROTH, Rational approximations to algebraic numbers, *Mathematika*, **2** (1955), 1–20.

[48] J. W. SANDER, *Rational points on a class of superelliptic curves, J. London Math. Soc.* **59**, (1999), 422-434.

[49] B. POONEN, E. SCHAEFER, Explicit descent for Jacobians of cyclic covers of the projective line, *J. reine angew. Math.* **488** (1997), 141-188.

[50] D. STEHLÉ, V. LEFÈVRE, P. ZIMMERMANN, Worst cases and lattice redution, Arith'16, 2003.

[51] R. J. STROEKER, B.M.M. DE WEGER, On elliptic diophantine equations that defy Thue's method : the case of the Ochoa curve, *Experimental Math.* **3** (1994), 209–220.

[52] A. THUE, Über Annäherungswerte algebraischer Zahlen, *J. Reine Angew. Math.* **135** (1909), 284–305.

[53] N. TZANAKIS, B.M.M. DE WEGER, On the Practical Solution of the Thue Equation, *J. Number Th.* **31** (1989), 99–132.

[54] P.M. Voutier, Primitive divisors of Lucas and Lehmer sequences, III, *Math. Proc. Cambridge Phil. Soc.* **123** (1998), 407–419.

[55] M. WARD, The intrinsic divisors of Lehmer numbers, *Ann. Math. (2)*, **62** (1955), 230–236.

[56] B.M.M. DE WEGER, Solving exponential diophantine equations using lattice basis reduction algorithms, *J. Number Th.* **26** (1987), 325–367.

[57] B.M.M. DE WEGER, *S*-integral solutions to a Weierstrass equation, *J. Théor. Nombres Bordeaux* **9** (1997), 281–301.

Résumé

Ce mémoire présente un ensemble de travaux autour de méthodes algorithmiques en arithmétique. Arithmétique est ici à comprendre dans un sens large, allant de la théorie des nombres à l'arithmétique des ordinateurs. Dans le premier domaine, nous décrivons des algorithmes efficaces basées sur les méthodes de formes linéaires de logarithmes pour résoudre deux grands types d'équations diophantiennes, l'équation de Thue et l'équation superelliptique. Diverses applications sont mentionnées, dont en particulier la résolution de la conjecture des diviseurs primitifs des suites de Lucas. Dans le domaine de l'arithmétique des ordinateurs, nous nous intéressons d'une part à divers problèmes d'arithmétique multiprécision, et introduisons la notion de *produit médian* mais aussi au problème de la fiabilité des calculs sur les données inexactes, typiquement les nombres réels. Nous décrivons en particulier la bibliothèque MPFR, dont l'objectif est de montrer qu'il est possible de réconcilier fiabilité et efficacité.

Mots-clés: équations diophantiennes, méthode de Baker, équation de Thue, équation de Catalan, suites de Lucas, arithmétique multiprécision, calcul formel, arithmétique des ordinateurs, évaluation des fonctions spéciales, norme IEEE 754, calcul fiable.

Abstract

This memoir describes work done on algorithmic methods in arithmetics. The word Arithmetics is here meant in a very broad sense, including both number theory and computer arithmetic. In the first acception, we describe efficient algorithms based on linear forms in logarithms of algebraic numbers that allow one to solve two families of diophantine equations, namely Thue equations and superelliptic equations. Several applications are described, among which the solution of the primitive divisor problem for Lucas sequences. Concerning computer arithmetic, we study some aspects of multiprecision arithmetic, especially the middle product and optimization of Mulders' short product idea. We also discuss the reliability of floating-point computations, and present the Mpfr multiprecision library, which offers correct rounding and multiple precision computations, for a large set of functions.

Keywords: Diophantine equations, Baker's method, Thue equation, Catalan equation, Lucas sequences, multiple-precision arithmetic, computer algebra, computer arithmetic, special functions evaluation, IEEE 754 norm, reliable computing