Factorisation par la méthode du crible quadratique

Sujet proposé par Luc Maranget

Luc.Maranget@inria.fr

Important, ce projet est accompagné d'une *page de suivi*[1], qui propose en outre des documents complémentaires.

1 Factorisation des grand nombres

La factorisation effective d'entiers toujours plus grands, que l'on pouvait autrefois qualifier d'amour de l'art, voire de hobby (mais aussi d'effort de recherche fondamentale), connaît des applications médiatisées en cryptographie, car la sécurité des systèmes de cryptage modernes semble bien reposer en dernière analyse sur la difficulté de répondre à cette question toute bête : Soit *ce* nombre n, dont je sais qu'il n'est pas premier, quels sont ses facteurs ?

Dans ce projet nous allons aborder une méthode de factorisation qui a fait date. La méthode du *crible quadratique* est une introduction aux méthodes générales de factorisation modernes, qui en sont souvent des raffinements. Par ailleurs, le crible quadratique s'appuie sur des concepts raisonnablement simples de la théorie des nombres, ce qui place sa mise en œuvre à notre portée.

2 Principe

La méthode de factorisation repose sur une idée assez ancienne (1920) du mathématicien Maurice Kraitchick. Soit n un entier que nous souhaitons factoriser. Supposons que, par un moyen quelconque, nous connaissons deux autres entiers X et Y, tels que $X^2 \equiv Y^2 \pmod{n}$ soit vrai, Il existe donc un entier k tel que $(X-Y)(X+Y) = kn$. Avec un peu de chance, $X-Y$ et n ont des facteurs non triviaux en commun et $\textsc{pgcd}(X-Y, n)$ est un facteur de n. (en cas de malchance on essaie une autre congruence). Cette idée est un raffinement d'une autre, encore plus ancienne, et due à Pierre de Fermat : pour factoriser n on peut essayer de l'exprimer comme la différence de deux carrés $n = X^2 - Y^2$ (et donc $n = (X-Y)(X+Y)$). Pour dire toute la vérité, la méthode de Kraitchick ne fonctionne que si n peut s'écrire comme produit de deux facteurs premiers entre eux, c'est-à-dire si n n'est pas une puissance (et n'est pas premier bien évidemment). Ce n'est pas bien grave.

En ce qui nous concerne, la supériorité de l'idée de Kraitchick tient à ce qu'on peut trouver ses deux fameux entiers X et Y pour un prix plus modique que ceux de Fermat. On peut en effet procéder ainsi :

1. Définir les petits nombres premiers en considérant les nombres premiers inférieurs à un certain entier B. On note $p_1, p_2, \ldots, p_b$ les petits nombres premiers, que l'on complète par $p_0 = -1$ pour pouvoir factoriser des entiers négatifs.

[1] http://www.enseignement.polytechnique.fr/profs/informatique/Luc.Maranget/IF/qs/suivi.html

2. Chercher les entiers x_i tels la décomposition en facteurs premiers de $y_i = x_i^2 - n$ s'écrit uniquement à l'aide de petits nombres premiers. Un tel y_i est dit *B-régulier*. On obtient ainsi des congruences de la forme $x_i^2 \equiv p_0^{\alpha_{0,i}} p_1^{\alpha_{1,i}} \cdots p_b^{\alpha_{b,i}} \pmod n$ (on peut toujours poser $\alpha_{k,i} = 0$ si p_k ne divise pas y_i). Pour augmenter la possibilité de trouver des y_i B-réguliers, on essaie les y_i à priori petits, en prenant les x_i voisins de $\sqrt{n}$.
3. Multiplier membre à membre certaines des équations de l'étape précédente afin d'obtenir une égalité $X^2 = Y^2$ modulo n. Il n'y aucun problème pour obtenir le carré X^2 (c'est le produit de certains x_i). Pour obtenir le carré Y^2, on utilise les décompositions en petits facteurs premiers, en sélectionnant les équations de sorte que la somme des $\alpha_{k,i}$ soit paire pour tous les k.

Un exemple simple illustrera la méthode, cherchons à factoriser le nombre $n = 13483$. Un peu au hasard nous choisissons $B = 20$. On sélectionne ensuite des valeurs du polynôme $P(x) = x^2 - n$ (dit polynôme de Kraitchick) qui sont B-régulières, pour x voisin de $\lfloor\sqrt{n}\rfloor = 116$. On aura ainsi par exemple, les équations (modulo n) :

$$125^2 = 2 \cdot 3^2 \cdot 7 \cdot 17, \qquad 127^2 = 2 \cdot 3^3 \cdot 7^2, \qquad 91^2 = -1 \cdot 2 \cdot 3^2 \cdot 17^2,$$

$$113^2 = -1 \cdot 2 \cdot 3 \cdot 7 \cdot 17, \qquad 167^2 = 2 \cdot 3 \cdot 7^4$$

(On notera la présence de -1, due à des valeurs de x_i inférieures ou égales à $\lfloor\sqrt{n}\rfloor$.)

En multipliant toutes les équations sauf la seconde entre elles on obtient :

$$(125 \cdot 91 \cdot 113 \cdot 167)^2 \equiv (-1)^2 \cdot 2^4 \cdot 3^6 \cdot 7^6 \cdot 17^4 \pmod n$$

Soit encore $X \equiv 8265 \pmod n$ et $Y = 2^2 \cdot 3^3 \cdot 7^3 \cdot 17^2 \equiv 13269 \pmod n$. Au final, on calcule $\text{PGCD}(8265 - 13269, n)$ qui vaut 139. Et effectivement on a bien $n = 139 \cdot 97$.

Pour rendre la méthode de factorisation décrite effective, il faut résoudre deux questions.

- Trouver des congruences $x_i^2 \equiv p_0^{\alpha_{0,i}} p_1^{\alpha_{1,i}} \cdots p_b^{\alpha_{b,i}} \pmod n$ efficacement, car la méthode évidente d'essai de division des y_i par les petits nombres premiers est bien trop coûteuse. On procédera selon la technique du *crible quadratique* introduite en 1982 par Carl Pomerance.
- Combiner les congruences de l'étape précédente pour obtenir des congruences finales $X^2 \equiv Y^2 \pmod n$. C'est possible, à l'aide d'un peu d'algèbre linéaire sur l'espace vectoriel $(\mathbb{Z}/2\mathbb{Z})^k$, noté à l'américaine $\mathbb{F}_2^k$.

3 Le crible quadratique

Le crible quadratique est une genéralisation du crible d'Eratosthène. En effet, pour tout polynôme $P(x)$ et tout entier p nous avons $P(x) \equiv P(x+p) \pmod p$. Soit, supposant connu x_a tel que $P(x_a) \equiv 0 \pmod p$, nous pouvons rapidement conclure que $P(x_{a+p}), P(x_{a+2p}), \ldots, P(x_{a+\ell p}), \ldots$ sont divisible par p (dans le crible d'Eratosthène, on a $P(x) = x$ et $x_a = p$). En combinant ces parcours de p en p pour divers facteurs premiers, nous allons identifier *des* $P(x)$ qui sont B-réguliers.

On considère donc une base de facteurs $p_0, p_1, \ldots, p_b$ ($p_0 = -1, p_1 = 2$, etc.) en supposant bien évidemment que n n'est divisible par aucun des « vrais » p_k (auquel cas nous connaîtrions un facteur de n). Cette base ne regroupe pas tous les nombres premiers inférieurs à B. En effet, il n'y a pas nécessairement de racine modulo p de l'équation $P(x) = 0$ pour p quelconque. Nous supposons donc que la base de facteurs regroupe les petits nombres premiers tels que $P(x)$ a des racines modulo p_k. Dans le cas du polynôme de Kraitchick (qui est quadratique) et pour un p_k ($k \geq 1$) donné, il y aura une ou deux solutions. Pour $p_1 = 2$, par hypothèse n impair, nous avons une seule solution qui est $x \equiv 1 \pmod 2$. Pour tout autre p_k

premier, il y a deux solutions (car n n'est pas divisible par p_k) qui sont d'ailleurs opposées modulo p_k, et notées r_k et r'_k. Par exemple, dans le cas $n = 13483$, $B = 20$ la base de facteurs se restreint à $-1, 2, 3, 7, 17$ avec $r_1 = 1$, $r_2 = 1$, $r_3 = 1$ et $r_4 = 6$ (et aussi $r'_2 = 2$, $r'_3 = 6$, $r'_4 = 11$).

Étant donnée une taille T et une valeur initiale x_0 nous considérons l'intervalle $[x_0 \ldots x_0 + T[$ d'entiers notés $x_0, x_1, \ldots x_{T-1}$. Dans le même esprit nous posons $y_i = P(x_i)$. Soit encore un tableau w d'entiers w_i valant initialement zéro. Pour un p_k donné nous pouvons détecter tous les y_i multiples de p_k et enregistrer ce fait dans le tableau w. Il suffit de calculer $x_0 \mod p_k$ et d'en déduire x_a et $x_{a'}$, racines de $P(x) = 0$ modulo p_k telles que a et a' sont minimaux (pour $k = 1$ on a $a = a'$). Ensuite, on parcourt le tableau w de p_k en p_k à partir de a et de a', en ajoutant le logarithme de p_k aux w_i ainsi repérés. En pratique, on utilise une approximation entière du logarithme de base 2, qui conduit à des calculs peu coûteux. Une fois ce parcours effectué on applique l'algorithme de factorisation par division des facteurs de la base aux y_i pour lesquels w_i est suffisamment proche du logarithme de y_i. On obtient ainsi une nouvelle équation $x_i^2 = \ldots$ presqu'à coup sûr.

Dans notre exemple, posons $T = 32$ et criblons à partir de $x_0 = 117$. Le nombre x_0 est impair, comme r_1, on a donc $a = 0$. Après un premier passage le tableau w est donc

1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0

Considérons maintenant le passage pour $p_k = 3$. On a $x_0 \mod 3 = 0$, $r_2 = 1$, $r'_2 = 2$, et donc $a = 1$ et $a' = 2$. En approximant $\log_2(3)$ par 2 et en criblant, il vient :

1 2 3 0 3 2 1 2 3 0 3 2 1 2 3 0 3 2 1 2 3 0 3 2 1 2 3 0 3 2 1 2

Il reste à effectuer le criblage des multiples de 7 ($\log_2(7) \sim 3$) et de 17 ($\log_2(17) \sim 4$). On obtiendra alors les états successifs du tableau w.

1 5 3 3 3 2 1 2 6 0 6 2 1 2 3 3 3 5 1 2 3 0 6 2 4 2 3 0 3 5 1 5
1 5 3 3 3 2 1 2 10 0 6 2 1 6 3 3 3 5 1 2 3 0 6 2 4 6 3 0 3 5 5 5

La case w_8 attire l'attention, on a d'une part $w_8 = 10$ et d'autre part $\log_2(y_8) \sim 12$ ($y_8 = 2141$). L'entier y_8 est sans doute un produit de facteurs de la base (de fait ici on sait que y_8 est divisible par $2 \cdot 3 \cdot 7 \cdot 17$ et que le quotient est proche de 2^2), on lance une tentative de factorisation de y_8 sur la base de facteurs. Il vient :

$$125^2 \equiv 2 \cdot 3^2 \cdot 7 \cdot 17 \pmod{n}$$

On notera que le décalage entre w_8 et $\log_2(y_8)$ peut s'expliquer ici parce que nous n'avons pas pris la peine de cribler selon les puissances des facteurs de la base, mais qu'il provient aussi dans le cas général de l'usage d'une approximation entière du logarithme.

Le crible s'applique en fait à des nombres plus grands que la valeur de n donnée en exemple. Il se révèle alors très efficace pour repérer les nombres B-réguliers, à condition de considérer attentivement certains points.

- Un bon choix du paramètre B est fondamental, C. Pomerance recommande B de « l'ordre de » $L(n)^{\frac{1}{2}}$, avec :

  $$L(n) = e^{\sqrt{\ln n \ln \ln n}}$$

  Le temps calcul théorique est alors de l'ordre de $L(n)$.
- Le logarithme de base deux des y_i s'approxime facilement comme le nombre de bits utilisés dans l'écriture en base deux. Cette quantité se calcule aisément à partir de la représentation machine des entiers (et la bibliothèque de calcul des grand nombres BigInteger la propose).

– La décision de voir le crible comme un moyen de trouver des y_i probablement B-réguliers (avant de vérifier) autorise de nombreuses économies. Nous avons vu que l'on peut se passer de cribler selon les puissances des facteurs de la base. On peut aussi envisager de se passer du criblage par les plus petits facteurs de la base, dont la contribution aux w_i peut être négligée et dont le criblage est coûteux. Quoiqu'il en soit, le bon choix des nombres à tenter de factoriser effectivement est critique. La condition $w_i \geq \log_2(y_i) - \log_2(B)$ est un bon point de départ.
– Il faut produire suffisamment d'équations pour garantir l'obtention de plusieurs congruences de carrés $X^2 \equiv Y^2 \pmod{n}$. En pratique, comme nous allons le voir, si $b+1$ est le nombre d'entiers de la base de facteurs, $b+1+m$ congruences mènent sûrement à m congruences de carrés. La probabilité qu'une congruence donne une factorisation triviale étant estimée à, au pire, $\frac{1}{2}$, choisir $m = 10$ nous donne une probabilité d'échec minime. Échec qui n'a d'ailleurs rien de définitif.
– La taille T des intervalles de criblage résulte d'un compromis entre le besoin d'amortir les calculs nécessaires pour démarrer le criblage sur un intervalle (par exemple, le recalage des progressions arithmétiques de taille p_k) et l'utilisation de la mémoire de la machine.

4 Algèbre linéaire

Après criblage, nous sommes donc en possession de (nombreuses) congruences $x_i^2 \equiv p_0^{\alpha_{0,i}} p_1^{\alpha_{1,i}} \cdots p_b^{\alpha_{b,i}} \pmod{n}$, que nous cherchons à multiplier afin de produire un carré à droite. Cela revient à trouver un ensemble I d'indices i tel que, pour tous les k, les sommes $\Sigma_{i\in I}\alpha_{k,i}$ sont paires. Cela revient de fait à trouver une famille liée dans l'espace vectoriel $\mathbb{F}_2^{b+1}$.

À chaque congruence, associons un vecteur $\vec{z}^i = (z_0^i \; z_1^i \; \cdots \; z_{b+1}^i)$ de $\mathbb{F}_2^{b+1}$, avec $z_k^i = \alpha_{k,i} \mod 2$. Nous cherchons en fait l'ensemble I tel que $\Sigma_{i\in I}\vec{z}^i = \vec{0}$. Opérationnellement on cherche à démontrer que la matrice $Z = (z_k^i)$ est singulière en exhibant une combinaison linéaire (somme sur $\mathbb{F}_2^{b+1}$) de ses lignes qui soit nulle. L'algorithme classique du *pivot de Gauss* permet d'atteindre ce résultat (garanti dès que la matrice a plus de lignes que de colonnes). Considérons à nouveau notre exemple de la factorisation de 13483. En supprimant la seconde ligne, la matrice des parités des exposants possède 4 lignes de 5 colonnes, nous lui adjoignons une autre matrice T qui est la matrice unité 4×4.

$$Z = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 \end{pmatrix}, \qquad T = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

L'opération de base du pivot consiste pour un k donné à annuler les z_k^i sauf pour un i donné, on y parvient par combinaison linéaire (ici somme) des lignes de Z avec une ligne i_0 telle $z_k^{i_0}$ n'est pas nul. Les combinaisons de lignes sont enregistrées dans les lignes de la matrice T. L'élément conservé est dit *pivot*. Si le nombre de lignes de Z dépasse celui de ses colonnes, on est sûr d'aboutir (à condition de prendre tous les pivots dans des lignes différentes). Ici choisissons d'abord d'éliminer les 1 de la première colonne, en prenant z_0^1 comme pivot. Il vient :

$$Z_0 = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 \end{pmatrix} \qquad T_0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

La troisième ligne de la matrice T_0 est la somme des deuxième et troisième lignes de T : elle nous dit comment obtenir cette troisième ligne de Z_0 à partir de Z.

On choisit ensuite le pivot z_1^0 (dans Z_0).

$$Z_1 = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{pmatrix} \qquad T_1 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

Prenons enfin le pivot z_2^2 :

$$Z_2 = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \qquad T_2 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

On voit bien ici que la quatrième ligne de T_2 s'obtient comme la somme des deux dernières lignes de T_1. Elle nous dit toujours comment la quatrième ligne de Z_2 s'obtient en sommant certaines des lignes de la matrice Z de départ (ici toutes). Notons qu'un pivot de Gauss traditionnel échange les lignes des matrices, (afin de systématiquement placer le pivot en z_k^k), ce qui nous donnerait ici :

$$Z_2 = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \qquad T_2 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

Les échanges de lignes dans Z sont aussi effectués dans T. Ainsi, rien ne change à la lecture que nous faisons des lignes de T comme exprimant des sommes de lignes de la matrice Z de départ.

L'implémentation efficace du pivot de Gauss dans notre cas particulier repose sur deux remarques.

1. On peut représenter les lignes des matrices Z et T comme des vecteurs de bits (classe BitSet de java). La sommation est alors l'opération « ou exclusif ».
2. Les matrices produites par le crible sont creuses, dans le sens qu'elles contiennent bien plus de 0 que de 1. Cette propriété autorise des spécialisations du pivot de Gauss, décrites dans [1] (fourni dans les documents complémentaires).

5 Travail demandé

Réaliser un programme Qs qui prend comme argument un nombre n adapté (n n'a pas de petit facteurs et possède au moins deux facteurs premiers distincts) et produit une factorisation de n. Par exemple :

```
# java Qs 350243405507562291174415825999
75576435361433 * 4634293795844903
```

Il s'agit d'un exemple « facile » dont la factorisation est quasi instantanée. La page de suivi donne d'autres exemples. En particulier, la factorisation du septième nombre de Fermat, $F_7 = 2^{2^7} + 1$, exploit mondial dans les années 70 est largement à notre portée (elle peut prendre moins d'une minute).

Une répartition possible du travail dans le binôme est évidemment entre crible et pivot de Gauss. Attention, toutefois, il faut se rendre compte que, pour les tailles de nombres que nous traitons (pas plus de soixante chiffres), la réalisation d'un pivot de Gauss très optimisé semble moins critique que celle d'un crible efficace.

On utilisera donc les classes BigInteger et BitSet. Ces classes fournissent toutes les opérations de base dont vous aurez besoin à l'exception des « racines carrées » entières et modulo p. Je fournis (sur la page de suivi) le code nécessaire, afin que vous puissiez vous concentrer sur la programmation de l'algorithme (plus que sur la théorie des nombres, certes fascinante). L'objectif à atteindre est la factorisation de nombres d'environ soixante chiffres en quelques heures.

6 Prolongements possibles

Vous pouvez par exemple :

1. Proposer un programme de factorisation pratique qui s'applique aux entiers quelconques, on intégrera par exemple des tests de primalité, l'algorithme de division voire l'algorithme dit ρ de Pollard. Il faudra aussi détecter les puissances. Un bon point de départ pour aborder les algorithmes nécessaires est certainement [2] (à la bibliothèque).
2. Accélérer significativement votre crible. Vous pouvez commencer par déterminer si il vaut la peine ou pas de cribler selon les puissances des facteurs de la base, puis raffiner l'algorithme. Les méthodes sont nombreuses (polynômes multiples, distribution du criblage, etc.).

Références

[1] C. Pomerance J. W. Smith. Reduction of huge, sparse matrices over a finite field via created catastrophes. *Experimental Math. 1 (1992), 90-94*, 1992.

[2] D. E. Knuth. *The Art of Computer Programming, Volume 2 : Seminumerical Algorithm, third Edition.* Addison-Wesley, 1997.

[3] C. Pomerance. A tale of two sieves. *The Notices of the Amer. Math. Soc. 43 (1996), 1473-1485*, 1996.