Le test de primalité AKS

Vincent Dalsheimer, Matthieu Giraud et Caixia Lu

Professeur encadrant : Pascal Autissier.

Résumé

Nous présentons ici la validité du test de primalité AKS découvert en 2002 par M. Agrawal, N. Kayal et N. Saxena. Le résultat principal de cet algorithme est théorique puisqu'il prouve que le problème de primalité se trouve dans la classe $\mathcal{P}$.

Table des matières

Notations

La majeure partie des notations utilisées sont usuelles, nous tenons cependant à en préciser quelques-unes :

- L'anneau $\mathbb{Z}/n\mathbb{Z}$ sera noté $\mathbb{Z}_n$ afin d'alléger la typographie.
- mod $X^r - 1, n$ indiquera une égalité dans l'anneau quotient $\mathbb{Z}_n/(X^r - 1)$.
- Sauf mention contraire, la fonction log est le logarithme en base 2.
- On notera ppcm(m) le plus petit commun multiple des m premiers entiers naturels non nuls.
- L'ordre de n modulo r, dans le groupe multiplicatif, sera noté $\omega_r(n)$.
- L'indicateur d'Euler pour un entier n sera quant à lui représenté par $\varphi(n)$.
- $|\mathcal{E}|$ désignera le cardinal de l'ensemble $\mathcal{E}$.
- $\nu_p(n)$ désignera la valuation p-adique de n.
- $\lfloor x \rfloor$ (respectivement $\lceil x \rceil$) représentera la partie entière (respectivement la partie entière par excès) de x.

Introduction

En mathématiques, et en cryptologie particulièrement, les nombres premiers jouent un rôle capital. Ainsi, être capable de déterminer si un entier n donné est premier ou non est un problème qui passionne depuis toujours les mathématiciens. Pourtant, après le célèbre crible d'Ératosthène ($3^{\text{ème}}$ siècle avant J.C.) qui consiste en la division de l'entier n par tous les nombres premiers inférieurs ou égaux à $\sqrt{n}$, aucun algorithme réellement plus efficace n'a été trouvé pendant près de 2200 ans ! Ce n'est qu'à partir des années 60, et la découverte de la théorie de la complexité, que les tests de primalité ont commencé à affluer dans le monde des mathématiques. Toutefois, aucun de ces algorithmes n'était à la fois déterministe, polynomial et inconditionnel. Un algorithme non-déterministe possède un risque d'erreur qui augmente avec le nombre d'itérations dans l'algorithme – par exemple, le crible d'Atkin –; un algorithme non polynomial est beaucoup trop long à exécuter dès qu'on veut l'appliquer à de grands nombres – par exemple, le crible d'Ératosthène –; enfin, un algorithme qui utilise l'hypothèse de Riemann généralisée n'est pas inconditionnel, car celle-ci n'a pour l'instant pas été démontrée – par exemple, l'algorithme de Miller-Rabin. Ce n'est qu'en août 2002 que le professeur M. Agrawal et deux de ses élèves, N. Kayal et N. Saxena, trouvent enfin un algorithme déterministe, polynomial et inconditionnel permettant de déterminer si un entier n donné est premier ou non.

Il est relativement aisé de démontrer – nous le ferons plus bas – que cet algorithme, appelé algorithme AKS, est de complexité $O(\log^{10,5} n)$. Néanmoins, il est aussi possible, grâce à des résultats plus poussés de la théorie des nombres, de montrer que la complexité de l'algorithme AKS peut être améliorée jusqu'à atteindre $O(\log^6 n)$. Toutefois, cela reste une complexité assez élevée, et donc un algorithme un peu lent étant donné qu'on estime qu'un algorithme, pour être concrètement utilisable, doit être au plus de complexité $O(\log^3 n)$. Ainsi, cet algorithme n'est actuellement pas réellement utilisable, dans le domaine de la cryptologie notamment. L'intérêt de cet algorithme réside en fait principalement dans le domaine de la théorie, plutôt que dans la pratique. En effet, si l'application informatique de cet algorithme ne peut pas être efficace pour l'instant, l'aspect mathématique est quant à lui d'une ampleur toute autre; en trouvant cet algorithme, M. Agrawal, N. Kayal et N. Saxena ont réussi à prouver un résultat mathématique des plus importants. Appelons $\mathbb{P}$ le problème de savoir si un entier donné n est premier. Alors $\mathbb{P} \in \mathcal{P}$, à savoir la classe des problèmes décisionnels – auxquels on peut répondre par oui ou par non – qui peuvent être résolus par un algorithme déterministe en un temps polynomial, et ce quel que soit l'argument du problème. Si le résultat théorique a indéniablement suscité l'engouement de la communauté scientifique, il est important de garder à l'esprit que l'algorithme sera très certainement amélioré à l'avenir. Il n'est donc pas absurde d'envisager que l'algorithme AKS surpassera un jour les tests de primalité actuels, surtout en cryptologie, où les nombres manipulés doivent être très grands, et qu'aucune erreur sur leur primalité ne peut être tolérée.

Étudions maintenant cet algorithme de manière plus précise.

1 L'algorithme AKS

1.1 L'idée de l'algorithme

Le test de primalité AKS repose sur la généralisation suivante du petit théorème de Fermat.

Proposition 1.1.1. *Soient a un entier et n un entier > 1 tels que* $\text{pgcd}(a, n) = 1$.
n est premier si, et seulement si,

$$(X + a)^n \equiv X^n + a \mod n. \tag{1}$$

Démonstration. $\Rightarrow \forall i \in [\![1, n-1]\!]$, on a $i\binom{n}{i} = n\binom{n-1}{i-1}$ et on en déduit que $n \mid i\binom{n}{i}$.
Or, on a pour tout i, $\text{pgcd}(i, n) = 1$ et le lemme de Gauss nous dit que $n \mid \binom{n}{i}$. D'après le binôme de Newton, on sait que le coefficient du monôme X^i est $\binom{n}{i}a^{n-i}$ et $n \mid \binom{n}{i}a^{n-i}$, on a donc $(X + a)^n \equiv X^n + a^n \mod n$.

$\Leftarrow$ On suppose que n est composé et s'écrit de la façon suivante : $n = Nq^k$, où q est un facteur premier et $k = \nu_q(n)$. Alors $q^k \nmid \binom{n}{q}$, en effet :

$$\begin{aligned}\binom{n}{q} = \frac{n(n-1)\cdots(n-q+1)}{q!} &= \frac{Nq^k(n-1)\cdots(n-q+1)}{q!} \\ &= \frac{Nq^{k-1}(n-1)\cdots(n-q+1)}{(q-1)!}\end{aligned}$$

Et pour tout i tel que $1 \leqslant i \leqslant q-1$, on a $q^k \nmid (Nq^k - i)$ donc $q^k \nmid \binom{n}{q}$.
De plus, on sait que $\text{pgcd}(a, Nq^k) = 1$ donc $\text{pgcd}(a, q^k) = 1$. On en déduit que $\nu_q(a) = 0$ et $\text{pgcd}(a^{n-q}, q^k) = 1$.
Puisque

$$(X + a)^n = \sum_{i=0}^{n} \binom{n}{i} X^i a^{n-i},$$

on a $X^q \not\equiv 0 \mod n$ et donc $(X + a)^n \not\equiv (X^n + a) \mod n$. □

Ce théorème nous donne ainsi un premier test de primalité. En effet, en choisissant un entier a premier avec l'entier n à tester, il suffit de vérifier si la relation de congruence (1) est vérifiée. Il s'agit alors d'évaluer $n + 1$ coefficients ce qui prend un temps en $O(n)$.

Afin de réduire le temps d'évaluation de ces coefficients et ainsi rendre l'algorithme plus efficace, nous nous placerons dans l'anneau quotient $\mathbb{Z}_n/(X^r - 1)$ où r est un entier plus petit que n bien choisi (nous préciserons cette notion plus loin). Il s'agit alors de vérifier l'équation

$$(X + a)^n \equiv X^n + a \mod X^r - 1, n. \tag{2}$$

On sait d'après la Proposition 1.1.1 que la relation de congruence (2) est clairement vérifiée pour tout n premier et tout entier r et a tel que $\text{pgcd}(a, n) = 1$. On peut se demander alors si la relation de congruence (2) est une condition suffisante pour la primalité de n. Malheureusement, un problème apparaît puisque (2) peut être également valide pour un entier n composé et quelques valeurs de a et de r tel que $\text{pgcd}(a, n) = 1$ comme nous pouvons le

voir dans l'exemple suivant.

Soit le troisième nombre de Carmichael, $n = 1729 = 7 \cdot 13 \cdot 19$, on remarque que celui-ci vérifie la congruence avec $a = 5$ et $r = 3$ alors que n n'est clairement pas un nombre premier et que $\text{pgcd}(5, 1729) = 1$, en effet

$$(X + 5)^{1729} \equiv X^{1729} + 5 \mod X^3 - 1, 1729.$$

Alors qu'avec $r = 5$

$$(X + 5)^{1729} \not\equiv X^{1729} + 5 \mod X^5 - 1, 1729.$$

Ainsi, l'un des buts de cet algorithme est d'étendre la relation de congruence (1) à l'anneau quotient $\mathbb{Z}_n/(X^r - 1)$. Nous allons dans la suite de cet article prouver qu'une telle extension, de $\mathbb{Z}_n$ à $\mathbb{Z}_n/(X^r - 1)$ pour la relation (1) existe bel et bien. En d'autres termes nous démontrerons qu'il existe un r tel que si un nombre suffisant de a vérifient (2) alors n est un nombre premier.

1.2 Quelques rappels

Nous donnons dans cette section quelques rappels de combinatoire, sur l'introspection et les polynômes cyclotomiques. Ces rappels seront utilisés dans la section 1.4. afin de démontrer le théorème fondamental de l'algorithme AKS.

1.2.1 Rappels de combinatoire

Lemme 1.2.1. *Soient p et n des entiers $\geqslant 1$. Alors $\sum_{i=p}^{n} \binom{i}{p} = \binom{n+1}{p+1}$.*

Démonstration. Raisonnons par récurrence sur n.
Si $n = 1$, alors $p = 1$ et $\binom{1}{1} = 1 = \binom{2}{2}$ donc la propriété est vérifiée pour $n = 1$.

Soit $n \geqslant 2$. Si $p < n$, on a d'après la relation de Pascal

$$\binom{n+1}{p+1} = \binom{n}{p} + \binom{n}{p+1}.$$

Et d'après la relation de récurrence avec $p + 1 \leqslant n$

$$\binom{n+1}{p+1} = \binom{n}{p} + \binom{n}{p+1} = \binom{n}{p} + \binom{n-1}{p} + \cdots + \binom{p}{p}.$$

Si $p = n$, $\binom{n+1}{p+1} = 1 = \binom{n}{p}$.
La propriété est donc vérifiée. □

Lemme 1.2.2. *Soit p un entier > 0. Il existe $\binom{n+p-1}{n}$ possibilités de choisir n entiers naturels dont la somme est $\leqslant p - 1$.*

Démonstration. Soit p un tel entier fixé. Montrons le résultat par récurrence sur n.

Si $n = 1$, il y a p choix possibles, on a bien $\binom{p}{1} = p$.

Si $n \geqslant 2$, notons les n nombres considérés $\{e_i\}_{1 \leqslant i \leqslant n}$.
À e_n fixé, et si l'on tient compte de ce que $\sum_{i=1}^{n} e_i \leqslant p-1$, choisissons les $n-1$ nombres $\{e_i\}_{1 \leqslant i \leqslant n-1}$. Par hypothèse de récurrence, on a $\binom{n+p-2}{n-1}$ possibilités si $e_n = 0$, $\binom{n+p-3}{n-1}$ possibilités si $e_n = 1$, ... et $\binom{n-1}{n-1}$ possibilités si $e_n = p-1$.

Cela fait donc $\binom{n+p-2}{n-1} + \binom{n+p-3}{n-1} + \cdots + \binom{n-1}{n-1} = \binom{n+p-1}{n}$ possibilités au total pour le choix des n entiers d'après le Lemme 1.2.1 précédent. □

1.2.2 Rappels sur l'introspection

Définition 1.2.1. *Soit p un nombre premier et r un entier > 1. Étant donnés un polynôme $f(X) \in \mathbb{Z}[X]$ et m un entier > 1, m est dit introspectif pour $f(X) \mod p$ par rapport à r, ou simplement introspectif s'il n'y a pas de confusion possible si*

$$f(X)^m \equiv f(X^m) \mod X^r - 1, p.$$

On peut facilement remarquer qu'un entier premier p est introspectif modulo p. De plus, on montrera ci-dessous que le produit de deux entiers introspectifs pour un polynôme $f(X)$ est aussi introspectif pour le même polynôme $f(X)$ et que si un entier m est introspectif pour deux polynômes alors m est aussi introspectif pour le produit de ces deux polynômes.

Lemme 1.2.3. *Si m et m' sont introspectifs pour un polynôme $f(X)$, alors $m \cdot m'$ l'est aussi.*

Démonstration. Puisque m est introspectif pour $f(X)$, on a

$$f(X)^{m \cdot m'} \equiv f(X^m)^{m'} \mod X^r - 1, p.$$

Comme m' est aussi introspectif pour $f(X)$, on a, en substituant X par X^m

$$\begin{aligned} f(X^m)^{m'} &\equiv f(X^{m \cdot m'}) \mod X^{m \cdot r} - 1, p \\ &\equiv f(X^{m \cdot m'}) \mod X^r - 1, p \text{puisque } X^r - 1 \mid X^{m \cdot r} - 1 \end{aligned}$$

En combinant les deux relations, on obtient

$$f(X)^{m \cdot m'} \equiv f(X^{m \cdot m'}) \mod X^r - 1, p.$$

□

Lemme 1.2.4. *Si m est introspectif pour deux polynômes $f(X)$ et $g(X)$, alors m est aussi introspectif pour le produit $f(X) \cdot g(X)$.*

Démonstration. En effet, on a

$$\begin{aligned} (f \cdot g)(X)^m &\equiv (f(X) \cdot g(X))^m \mod X^r - 1, p \\ &\equiv f(X)^m \cdot g(X)^m \mod X^r - 1, p \\ &\equiv f(X^m) \cdot g(X^m) \mod X^r - 1, p \\ &\equiv (f \cdot g)(X^m) \mod X^r - 1, p. \end{aligned}$$

□

1.2.3 Rappels sur les polynômes cyclotomiques

Définition 1.2.2. *Soit n un entier > 0. Le n-ième polynôme cyclotomique est le produit $Q_r(X) = \prod_\zeta (X - \zeta)$ où ζ décrit les racines primitives n-ièmes de l'unité dans $\mathbb{C}$.*

Lemme 1.2.5. *$Q_n(X)$ est un polynôme unitaire à coefficients entiers de degré $\varphi(n)$. On a*

$$X^n - 1 = \prod_{d|n} Q_d(X).$$

Démonstration. Chaque racine n-ième de l'unité dans $\mathbb{C}$ a un unique ordre multiplicatif qui est un diviseur de n d'après le théorème de Lagrange. Autrement dit, il est possible de partitionner les racines n-ièmes de l'unité suivant leur ordre, lesquelles racines sont les zéros de $Q_d(X)$ pour $d \mid n$. On obtient bien la relation annoncée.

$Q_n(X)$ est clairement unitaire de degré $\varphi(n)$.

Montrons par récurrence sur n que ses coefficients sont entiers. On a $Q_1(X) = X - 1 \in \mathbb{Z}[X]$.
Par hypothèse de récurrence, les facteurs du produit, sauf justement $Q_n(X)$, sont à coefficients entiers. On peut alors écrire dans $\mathbb{C}[X]$, $X^n - 1 = Q_n(X) \cdot P(X)$ où P est un polynôme unitaire de $\mathbb{Z}[X]$.

Par unicité de la division euclidienne de $X^n - 1$ par P dans $\mathbb{Z}[X]$, on en déduit $Q_n(X) \in \mathbb{Z}[X]$. □

Proposition 1.2.1. *Soient p et r des entiers naturels tels que p soit un nombre premier et $\operatorname{pgcd}(p, r) = 1$. Le polynôme cyclotomique Q_r se décompose dans $\mathbb{F}_p[X]$ en un produit de polynômes unitaires irréductibles de même degré $k = \omega_r(p)$.*

Démonstration. Soit $h(X)$ un tel facteur de degré w. On note $K = \mathbb{F}_p[X]/(h(X))$.

$|K| = p^w$ et tout élément non nul $\alpha \in K$ vérifie $\alpha^{p^w - 1} = 1$ d'après le théorème de Lagrange. Le corps K contient par construction une racine de Q_r qui est une racine primitive r-ième de l'unité, disons ζ. En effet, notons ζ la classe de X dans K. ζ est racine de Q_r, donc aussi de $X^r - 1$; on en déduit que $\omega_p(\zeta)$ divise r. Supposons que $\omega_p(\zeta) < r$. Alors ζ est racine de Q_r et de $X^{\omega_p(\zeta)} - 1$, donc est racine double de $X^r - 1$ (car $Q_r \neq X^{\omega_p(\zeta)} - 1$). D'où, ζ est aussi racine de rX^{r-1} (dérivée de $X^r - 1$), ce qui est absurde car r est inversible modulo p. Donc $\omega_p(\zeta) = r$ et ainsi ζ est primitive modulo p.
Puisque $\zeta^{p^w - 1} = 1$, on a $r \mid p^w - 1$ donc $p^w \equiv 1 \mod r$ et $w \geqslant k$.

Inversement, puisque $\zeta^r = 1$ et que $r \mid p^k - 1$, on a $\zeta^{p^k} = \zeta$. Le sous-corps de K formé des racines de l'équation $X^{p^k} = X$ contient ζ, donc est égal à K tout entier, étant donné que ζ est un élément primitif de K. On a donc $p^w = |K| \leqslant p^k$, d'où $w \leqslant k$.

D'où le résultat. □

1.3 Résultat préliminaire

Nous donnons dans cette section une importante proposition qui nous sera nécessaire lors de la démonstration du théorème utilisé pour le test de primalité AKS.

Lemme 1.3.1. *Il existe un entier $r \leqslant \max\{3, \lceil \log^5 n \rceil\}$ tel que $\omega_r(n) > \log^2 n$.*

Démonstration. Si $n = 2$, il est clair que $r = 3$ satisfait les conditions. Supposons pour la suite de la démonstration que $n > 2$, alors $\lceil \log^5 n \rceil > 10$.

Soit $R = \{r_i \in \mathbb{N}^* \text{ tel que } \omega_{r_i}(n) \leqslant \log^2 n\}$. R est un ensemble fini puisque si $r > n^{\log^2 n}$ alors $\omega_r(n) > \log^2 n$.

Soit

$$\Delta := \prod_{i=1}^{\lfloor \log^2 n \rfloor} (n^i - 1)$$

Pour tout $r_i \in R$, on a $n^{\omega_{r_i}(n)} \equiv 1 \mod r_i$. Ainsi r_i divise $n^{\omega_{r_i}(n)} - 1$ présent dans Δ par définition.

On a

$$\begin{aligned}
\Delta &= (n-1)(n^2-1)(n^3-1)\cdots(n^{\lfloor \log^2 n \rfloor}-1) \\
&< n \cdot n^2 \cdot n^3 \cdots n^{\lfloor \log^2 n \rfloor} = n^{1+2+3+\cdots+\lfloor \log^2 n \rfloor} = n^{\frac{1}{2}\lfloor \log^2 n \rfloor (\lfloor \log^2 n \rfloor + 1)} \\
&< n^{\lfloor \log^2 n \rfloor^2} \text{ puisque } \log^2 n > 1 \\
&< n^{\lfloor \log^4 n \rfloor} \\
&< 2^{(\log n)\lfloor \log^4 n \rfloor} \\
&< 2^{\lceil \log^5 n \rceil}
\end{aligned}$$

Soit $m \geqslant 7$, on sait montrer facilement que $\text{ppcm}(m) \geqslant 2^m$. Ainsi le plus petit commun multiple des $\lceil \log^5 n \rceil$ premiers nombres entiers non nuls est $\geqslant 2^{\lceil \log^5 n \rceil}$. Si tous les nombres r inférieurs ou égaux à $\lceil \log^5 n \rceil$ vérifiaient $\omega_r(n) \leqslant \log^2 n$, alors $\text{ppcm}(\lceil \log^5 n \rceil)$ diviserait Δ. Or $\Delta < \text{ppcm}(\lceil \log^5 n \rceil)$, on en déduit qu'il existe un entier $r \leqslant \lceil \log^5 n \rceil$ tel que $\omega_r(n) > \log^2 n$. □

1.4 Le théorème de Agrawal, Kayal et Saxena

Citons pour commencer l'important théorème de M. Agrawal, N. Kayal et N. Saxena.

Théorème 1.4.1 (Agrawal, Kayal, Saxena)**.** *Soit n un entier > 1 qui n'est pas une puissance d'un nombre premier. Soit r un entier > 1 premier à n, choisi de telle façon que $r \leqslant \max\{3, \lceil \log^5 n \rceil\}$ et $\omega_r(n) > \log^2 n$. Supposons de plus que n n'ait pas de facteur premier $\leqslant r$ et posons $l = \lfloor \sqrt{\varphi(r)} \log n \rfloor$.*
Alors n est premier si, et seulement si, pour tout entier $0 \leqslant a \leqslant l$ la relation de congruence suivante est vérifiée

$$(X + a)^n \equiv X^n + a \mod X^r - 1, n.$$

Notre intention ici est de démontrer ce théorème découvert en juillet 2002. Celui-ci permet en effet, à l'aide d'un r bien choisi (dont l'existence a été prouvée au Lemme 1.3.1) et de plusieurs a déterminés, de caractériser un nombre premier n.

Démonstration. $\Rightarrow$ Le sens direct de la démonstration est naturel puisque nous avons déjà vu, par la Proposition 1.1.1, que si n est un nombre premier alors la relation de congruence $(X + a)^n \equiv X^n + a \mod X^r - 1, n$ est vérifiée pour tout entier a premier à n, et particulièrement pour tout entier $0 \leqslant a \leqslant l$. En effet, $\log^2 n < \omega_r(n) < r$, donc $\log n < \sqrt{r}$ et comme $\varphi(r) \leqslant r$, on a $l = \sqrt{\varphi(r)} \log n < r$. Ainsi, si n est premier ($\geqslant r$ par hypothèse), tout entier $0 \leqslant a \leqslant l$ est bien premier avec n. □

Il nous reste alors à démontrer le sens réciproque, c'est-à-dire à prouver que les relations de congruence $(X+a)^n \equiv X^n + a \mod X^r - 1, n$, pour tout entier $0 \leqslant a \leqslant l$, sont suffisantes pour déterminer la primalité de n. Pour cela, remarquons certaines choses avant de commencer la démonstration, à proprement parler, du sens réciproque.

D'après les hypothèses du théorème, la relation de congruence $(X + a)^n \equiv X^n + a \mod X^r - 1, n$ est vérifiée pour tout entier $0 \leqslant a \leqslant l$. Soit p un diviseur premier de n, la relation de congruence suivante est alors également vérifiée pour ces mêmes entiers a

$$(X + a)^n \equiv X^n + a \mod X^r - 1, p.$$

Ceci implique que n est introspectif modulo p pour les $l + 1$ polynômes $X, X + 1, \ldots, X + l$ et comme p est premier, on a d'après la Proposition 1.1.1

$$(X + a)^p \equiv X^p + a \mod X^r - 1, p.$$

Donc n et p sont introspectifs pour les polynômes $X + a$ où $0 \leqslant a \leqslant l$.

D'après les Lemmes 1.2.3 et 1.2.4, on peut remarquer que n/p est introspectif. En effet, posons $d = p^k$ où k est un entier > 0. On sait que $(X + a)^n \equiv X^n + a \mod X^r - 1, p$. Par Frobenius, on trouve que $(X^d + a)^{n/p} \equiv X^{dn/p} \mod X^r - 1, p$ (car $k > 0$). On choisit, et on sait qu'il existe puisque $\text{pgcd}(r, p) = 1$, k tel que $p^k \equiv 1 \mod r$. On a alors $X^d \equiv X \mod X^r - 1, p$, dont on déduit que $(X + a)^{n/p} \equiv X^{n/p} + a \mod X^r - 1, p$.

On a alors que tout entier dans l'ensemble $I := \{(n/p)^i \cdot p^j \mid i, j \geqslant 0\}$ est introspectif pour tous les polynômes de l'ensemble $P := \{\prod_{a=0}^{l}(X + a)^{i_a} \mid i_a \geqslant 0\}$.

Nous pouvons désormais définir deux groupes liés aux ensembles I et P qui joueront un rôle crucial dans la démonstration.

Soit G l'un de ces deux groupes où

$$G := \{\ x \mod r \mid x \in I\ \}.$$

Comme on suppose que p est un facteur premier de n tel que $p > r$, on a $\text{pgcd}(n, r) = \text{pgcd}(p, r) = 1$ et donc G est un sous-groupe multiplicatif de $\mathbb{Z}_r$ engendré par n et p modulo r. En d'autres termes G est l'ensemble des résidus de I modulo r. Soit $t = |G|$. En fixant j dans l'ensemble I, on remarque que $\omega_r(n) \leqslant t$ et puisque $\omega_r(n) > \log^2 n$, on a

$$\log^2 n < t.$$

Pour définir le second groupe, nous aurons besoin des rappels sur les polynômes cyclotomiques dans un corps fini donnés dans la sous-section 1.2.2. Soit $Q_r(X)$, le r-ième polynôme cyclotomique sur $\mathbb{F}_p$. On sait d'après le Lemme 1.2.5, que $Q_r(X)$ divise $X^r - 1$ et qu'il se factorise en produit de polynômes irréductibles de même degré $\omega_r(p)$. Soit $h(X)$ un tel facteur irréductible. Puisque $\omega_r(p) > 1$, le degré de $h(X)$ est > 1.

Le second groupe, noté $\mathcal{G}$, est l'ensemble des résidus des polynômes de P modulo $h(X)$ et modulo p. Ce groupe est généré par les éléments $X, X+1, \ldots, X+l$ dans le corps $F = \mathbb{F}_p[X]/(h(X))$ et c'est un sous-groupe du groupe multiplicatif associé à F.

Nous pouvons maintenant présenter deux lemmes concernant l'encadrement du cardinal du groupe $\mathcal{G}$ qui nous seront très utiles pour la démonstration du théorème.

Lemme 1.4.1 (Hendrik Lenstra Jr.)**.** $|\mathcal{G}| \geqslant \binom{t+l}{t-1}$

Démonstration. Puisque $h(X)$ est un facteur irréductible du r-ième polynôme cyclotomique $Q_r(X)$, on sait que α, la classe de X dans F, est une racine primitive r-ième de l'unité dans F.

Montrons alors qu'à deux polynômes distincts de degrés strictement inférieurs à t dans P correspondent deux résidus différents $\mathcal{G}$. Soient $f(X)$ et $g(X)$ deux tels polynômes dans P et supposons par l'absurde que $f(X) \equiv g(X)$ dans F.

Soit $m \in I$. Si $f(X) \equiv g(X)$ dans F alors on a aussi $f(X)^m \equiv g(X)^m$ dans F. Puisque m est introspectif pour $f(X)$ et $g(X)$ et que $h(X)$ divise $X^r - 1$, on a $f(X^m) \equiv g(X^m)$ dans F.

Ceci implique que X^m est une racine du polynôme $Q(Y) = f(Y) - g(Y)$, et cela pour tout $m \in G$. Comme $\operatorname{pgcd}(m, r) = 1$ (G est un sous-groupe de $\mathbb{Z}_r$), chaque X^m est une racine primitive r-ième de l'unité. Il y a donc $|G| = t$ racines distinctes de $Q(Y)$ dans F et ainsi $\deg(Q(Y)) \geqslant t$. Or, le degré de $Q(Y)$ ne peut être que strictement inférieur à t d'après le choix initial des polynômes $f(X)$ et $g(X)$ et donc $f(X) \not\equiv g(X)$ dans F.

Remarquons que $i \neq j$ dans $\mathbb{F}_p$ pour tout $0 \leqslant i \neq j \leqslant l$ puisque $l = \lfloor \sqrt{\varphi(r)} \log n \rfloor < \sqrt{r} \log n < r$ (car $\omega_r(n) > \log^2 n$) et comme $r < p$, on a bien $l < p$. Ainsi, les polynômes $X, X+1, X+2, \ldots, X+l$ sont tous distincts dans F.

Puisque $\deg(h(X)) > 1$, on en déduit que $X + a \not\equiv 0$ dans F pour tout entier a tel que $0 \leqslant a \leqslant l$. Il existe donc au moins $l+1$ polynômes distincts de degré 1 dans $\mathcal{G}$. Par conséquent, nous avons d'après le Lemme 1.3.2 au moins $\binom{t+l}{t-1}$ polynômes distincts de degré $< t$ dans $\mathcal{G}$. □

Lemme 1.4.2. *Si n n'est pas une puissance de p alors $|\mathcal{G}| \leqslant n^{\sqrt{t}}$*

Démonstration. Soit $\hat{I}$, le sous-ensemble de I défini par $\hat{I} = \{(n/p)^i \cdot p^j \mid 0 \leqslant i, j \leqslant \lfloor \sqrt{t} \rfloor\}$.

Si n n'est pas une puissance de p, alors l'ensemble $\hat{I}$ contient $(\lfloor \sqrt{t} \rfloor + 1)^2 > t$ entiers distincts. Puisque $|G| = t$, au moins deux des nombres de $\hat{I}$ sont égaux modulo r. Soient m_1 et m_2 deux tels nombres tels que $m_1 > m_2$, on a alors $X^{m_1} \equiv X^{m_2} \mod X^r - 1, p$.

Soit $f(X) \in P$, alors

$$\begin{aligned} f(X)^{m_1} &\equiv f(X^{m_1}) \mod X^r - 1, p \\ &\equiv f(X^{m_2}) \mod X^r - 1, p \\ &\equiv f(X)^{m_2} \mod X^r - 1, p \end{aligned}$$

Ce qui implique que $f(X)^{m_1} \equiv f(X)^{m_2}$ dans le corps F. De ce fait, on en déduit que $f(X) \in \mathcal{G}$ est une racine du polynôme $Q'(Y) = Y^{m_1} - Y^{m_2}$ dans le corps F. Comme $f(X)$ est un élément arbitraire de $\mathcal{G}$, le polynôme $Q'(Y)$ a au moins $|\mathcal{G}|$ racines distinctes dans F.

Le degré de $Q'(Y)$ est $m_1 \in \hat{I}$, on a donc $m_1 \leqslant (\frac{n}{p} \cdot p)^{\lfloor\sqrt{t}\rfloor} \leqslant n^{\lfloor\sqrt{t}\rfloor}$. □

Nous avons dorénavant tous les ingrédients nécessaires pour démontrer le sens réciproque du théorème de Agrawal, Kayal et Saxena.

Démonstration. ⇐ On suppose que pour tout entier $0 \leqslant a \leqslant l$ la relation de congruence suivante est vérifiée

$$(X + a)^n \equiv X^n + a \mod X^r - 1, n.$$

On a d'après le Lemme 1.4.1

$$\begin{aligned}
|\mathcal{G}| &\geqslant \binom{t+l}{t-1} \\
&\geqslant \binom{l+1+\lfloor\sqrt{t}\log n\rfloor}{\lfloor\sqrt{t}\log n\rfloor} \text{ puisque } t > \sqrt{t}\log n \\
&\geqslant \binom{2\lfloor\sqrt{t}\log n\rfloor+1}{\lfloor\sqrt{t}\log n\rfloor} \text{ puisque } l = \lfloor\sqrt{\varphi(r)}\log n\rfloor \\
&> 2^{\lfloor\sqrt{t}\log n\rfloor+1} \text{ puisque } \lfloor\sqrt{t}\log n\rfloor > \lfloor\log^2 n\rfloor \geqslant 1 \\
&> n^{\sqrt{t}}.
\end{aligned}$$

D'après le Lemme 1.4.2, il existe un entier $k > 0$ tel que $n = p^k$. On a supposé dans les hypothèses du théorème que n n'est pas une puissance d'un nombre premier, on a donc nécessairement que $n = p$. Ce qui achève la démonstration du théorème. □

1.5 L'algorithme AKS en pseudo-code

Nous sommes en mesure d'annoncer l'algorithme AKS en pseudo-code.

En effet, d'après le théorème de Agrawal, Kayal et Saxena, on sait que si n n'est pas une puissance d'un nombre premier p, que n ne possède pas de diviseur premier inférieur à r où r est le plus entier tel que $\omega_r(n) > \log^2 n$, alors n est premier si, et seulement si, pour tout entier a avec $0 \leqslant a \leqslant l = \lfloor\sqrt{\varphi(r)}\log n\rfloor$, on a

$$(X + a)^n \equiv X^n + a \mod X^r - 1, n.$$

Ainsi, nous avons de façon naturelle l'algorithme du test de primalité AKS.

Algorithme AKS

Entrée : n un entier > 1.
Sortie : "n est PREMIER" ou "n est COMPOSÉ".

1. Si $n = m^k$ pour m et k des entiers tels que k > 1, alors retourner "n est COMPOSÉ".
2. Déterminer le plus petit entier r tel que $\omega_r(n) > \log^2 n$.
3. Si 1 < pgcd(a,n) < n pour un entier a < r, alors retourner "n est COMPOSÉ".
4. Si $n \leqslant r$, alors retourner "n est PREMIER".
5. Pour a = 1 à $l = \lfloor \sqrt{\varphi(r)} \log n \rfloor$ faire :
   Si $(X+a)^n \not\equiv X^n + a \mod X^r - 1, n$, alors retourner "$n$ est COMPOSÉ".
6. Retourner "n est PREMIER".

Dans l'algorithme, l'étape 5 ne vérifie pas la relation de congruence pour $a = 0$, en effet il est évident dans ce cas que $(X + a)^n \equiv X^n + a \mod X^r - 1, n$.

2 Complexité de l'algorithme AKS

2.1 Résultat préliminaire

Proposition 2.1.1. *Soit n un entier > 1. Il existe un algorithme polynomial de complexité algébrique $O(\log^4 n)$ déterminant s'il existe des entiers m et k, où $k > 1$, tel que $n = m^k$.*

Démonstration. S'il existe des entiers m et k, où $k > 1$, tel que $n = m^k$, alors $2 \leqslant k \leqslant \lfloor \log n \rfloor$. Nous devons alors vérifier pour tout k si la fonction polynomiale $f_k(x) = x^k - n$ possède une racine entière > 1.

Remarquons que la fonction $f_k(x)$ est strictement croissante sur l'intervalle $[0, n]$ et que $f_k(0) = -n < 0$ tandis que $f_k(n) = n^k - n > 0$. D'après le théorème des valeurs intermédiaires, on sait qu'il existe une, et une seule, solution dans l'intervalle $[0, n]$. Nous n'avons plus qu'à vérifier si cette racine est entière ou non.

Pour cela, nous appliquons la recherche par dichotomie. Commençons par déterminer le signe de $f_k(n/2)$ qui revient à calculer $n^k - 2^k \cdot n$ avec une complexité algébrique $O(k \cdot \log n) = O(\log^2 n)$. Si $f_k(n/2) = 0$, ce qui n'arrive pas en pratique, l'algorithme est terminé et nous avons trouvé la racine k-ième de n.

Supposons que $f_k(n/2)$ est strictement positif, ce qui arrive en pratique. Nous évaluons alors $f_k(x)$ au point milieu de l'intervalle $[0, n/2]$, c'est-à-dire au point $n/4$. Pour cela, nous avons à calculer $n^k - 4^k \cdot n$ avec également une complexité algébrique $O(\log^2 n)$. Et si $f_k(n/4) \neq 0$, on continue ainsi de suite.

On peut supposer que l'évaluation successive de $f_k(x)$ en les points milieux des intervalles trouvés n'a pas déterminé de racine k-ième entière au polynôme $f_k(x)$. Une telle recherche demande $\lfloor \log n \rfloor + 1$ évaluations de $f_k(x)$. Ainsi, la recherche d'une racine k-ième est en $O(\log n \cdot \log^2 n) = O(\log^3 n)$.

Puisque nous devons répéter cette opération pour tout k avec $2 \leqslant k \leqslant \lfloor \log n \rfloor$, la complexité totale de cette algorithme est $O(\log n \cdot \log^3 n)$, c'est-à-dire $O(\log^4 n)$. □

2.2 L'algorithme AKS est polynomial

Nous allons ici démontrer le résultat majeur de l'article publié en 2002 par M. Agrawal, N. Kayal et N. Saxena qui est

$$\mathbb{P} \in \mathcal{P}.$$

Théorème 2.2.1. *Soit n un entier > 1. L'algorithme appliqué à n s'exécute en un temps polynomial.*

Démonstration. Pour pouver que l'algorithme AKS appliqué à n s'exécute bien en temps polynomial, il suffit de montrer que chaque étape de l'algorithme requiert un temps polynomial. Nous noterons dans ce qui suit

$$\Omega(f(n)) := O(f(n) \cdot p(\log(f(n)))) \quad \text{où } p \text{ est une fonction polynomiale en } \log n.$$

On a donc en particulier $\Omega(\log^k n) = O(\log^k n \cdot p(\log(\log n))) = O(\log^{k+\epsilon} n)$ pour tout $\epsilon > 0$.

1ÈRE ÉTAPE. D'après la Proposition 2.1.1, on sait que cette étape a une complexité $O(\log^4 n)$ et donc que celle-ci est polynomiale.

$2^{\text{ÈME}}$ ÉTAPE. Afin de vérifier qu'il existe un entier $r > 1$ tel que $\omega_r(n) > \log^2 n$, il nous faut tester que $n^k \not\equiv 1 \mod r$ pour tout $k \leqslant \log^2 n$. Ceci requiert, au maximum, $O(\log^2 n)$ calculs de puissances modulo r correspondant à une complexité $\Omega(\log^2 n \cdot \log r)$. D'après le Lemme 1.3.3, nous savons que seulement $O(\log^5 n)$ r différents ont besoin d'être testés. La complexité totale de la seconde étape est donc en $\Omega(\log^7 n)$.

$3^{\text{ÈME}}$ ÉTAPE. Pour cette étape, nous avons à calculer r fois le plus grand commun diviseur. Sachant qu'un tel calcul a une complexité de $O(\log n)$, la complexité de cette troisième étape est alors de $O(r \cdot \log n) = O(\log^5 n \cdot \log n) = O(\log^6 n)$.

$4^{\text{ÈME}}$ ÉTAPE. La quatrième étape s'exécute en seulement $O(\log n)$ puisque c'est une comparaison dont la complexité ne dépend que de la longueur de n.

$5^{\text{ÈME}}$ ÉTAPE. Lors de la cinquième étape, $\lfloor \sqrt{\varphi(r)} \log n \rfloor$ relations de congruence sont testées. Chacune requiert $O(\log n)$ multiplications de polynômes de degré r dont les coefficients sont de taille $O(\log n)$. Chaque étape de la boucle peut donc être vérifiée en $\Omega(r \log^2 n)$. La complexité totale de la cinquième étape est donc de $\Omega(r\sqrt{\varphi(r)} \log^3 n) = \Omega(r^{3/2} \log^3 n) = \Omega(\log^{21/2} n)$.

La complexité de la cinquième étape domine celle de toutes les autres, d'où le résultat. □

2.3 Améliorations de l'algorithme

Même si l'intérêt de l'algorithme AKS est actuellement d'ordre théorique, certaines conjectures, si elles s'avèrent vérifiées, lui donnerait également un intérêt pratique. Citons en quelques-unes.

2.3.1 La conjecture d'Artin

Conjecture 2.3.1. *Soient m et n des entiers > 1 tels que n ne soit pas un carré parfait. Alors le nombre de nombres premiers $q \leqslant m$ pour lesquels $\omega_q(n) = q - 1$ est asymptotique à $A \cdot m/\ln(m)$ avec*

$$A = \prod_{p \in \mathbb{P}} (1 - \frac{1}{p^2 - p}) \approx 0,37, \quad \textit{est la constante d'Artin.}$$

Si cette conjecture est vérifiée pour m de l'ordre $O(\log^2 n)$, on a immédiatement l'existence de r de l'ordre $O(\log^2 n)$ répondant aux propriétés recherchées. Avec un tel r, l'algorithme a alors une complexité en $O(\log^6 n)$.

2.3.2 La conjecture de la densité des premiers de Sophie Germain

Conjecture 2.3.2. *Soit n un entier $\geqslant 2$ et soit p un nombre premier. Le nombre $p \leqslant n$ tel que $2p + 1$ soit aussi premier est asymptotique à $2C_2 n/\ln^2 n$ où*

$$C_2 = \prod_{p \in \mathbb{P}} (1 - \frac{1}{(p-1)^2}) \approx 0,66, \quad \textit{est la constance des nombres premiers jumeaux.}$$

Si cette conjecture est vraie, on en déduit que r est de l'ordre $\Omega(\log^2 n)$. Avec un tel r, l'algorithme possède alors une complexité en $\Omega(\log^6 n)$.

Notons que des progrès ont été réalisés dans la véracité de cette conjecture. Une étude empirique permet d'ailleurs de trouver que l'algorithme possède une complexité de l'ordre de $1000 \cdot O(\log^6 n)$, ce qui est en parfait accord avec les deux conjectures précédentes.

2.3.3 Amélioration de l'identité remarquable

Dans l'algorithme, la boucle exécutée à l'étape 5 doit se réaliser au plus $\lfloor \sqrt{\varphi(r)} \log n \rfloor$ fois afin de s'assurer que le cardinal du groupe $\mathcal{G}$ est dépassé. Le nombre d'itérations de la boucle peut être réduit si l'on montre qu'il existe un système générateur de $\mathcal{G}$ plus petit que $\{X + a\}_{0 \leqslant a \leqslant l}$, ce qui est vraisemblable.

Conjecture 2.3.3. *Soient n et r des entiers naturels tels que r est un nombre premier ne divisant pas n et $(X-1)^n \equiv X^n - 1 \mod X^r - 1, n$. Sous ces conditions, on a*

$$n \text{ est premier ou } n^2 \equiv 1 \mod r.$$

Si cette conjecture s'avère vraie, la complexité de l'algorithme peut être améliorée jusqu'à $\Omega(\log^3 n)$. Empiriquement, elle fut vérifiée par M. Kayal et N. Saxena pour tout $r \leqslant 100$ et $n \leqslant 10^{10}$.

On pourrait alors modifier l'algorithme de sorte à d'abord chercher un entier r qui ne divise pas $n^2 - 1$; un tel r peut alors être trouvé dans l'intervalle $[\![2, 4 \log n]\!]$ (en utilisant le lemme prouvant que le produit de tous les nombres premiers inférieurs à n est asymptotiquement supérieur à e^n). La relation de congruence serait donc à vérifier pour l'entier r ainsi déterminé, ce qui requiert un temps en $\Omega(r \log^2 n)$.

Avec cet algorithme modifié, AKS serait très compétitif et utilisable en pratique.

3 Implémentation de l'algorithme

3.1 Une première implémentation en SAGE

Nous nous sommes tournés, au début de l'implémentation, vers le langage de calcul formel SAGE.

La facilité de ce langage nous a permis d'obtenir des résultats rapidement. Bien que l'algorithme soit de complexité $O(\log^{10.5} n)$, nous nous sommes aperçus, lors des tests, de la lenteur de l'algorithme.

Nous avons alors pensé à paralléliser les calculs, en particulier à l'étape 5 de l'algorithme, celle demandant le plus de calcul. Testé avec un nombre premier à 7 chiffres, l'algorithme restait très lent et très gourmand en mémoire vive (pour un tel nombre, nous avons pu atteindre une consommation de 80Go de mémoire vive sans aboutir à la fin du calcul).

Un tel test de primalité, avec des calculs de puissances de polynômes dans un anneau quotient, ne convient donc pas à un langage de calcul formel comme SAGE.

3.2 Une seconde implémentation en Pari/GP

Suite à l'implémentation de l'algorithme AKS en SAGE, nous nous sommes orientés vers le langage de programmation PARI/GP.

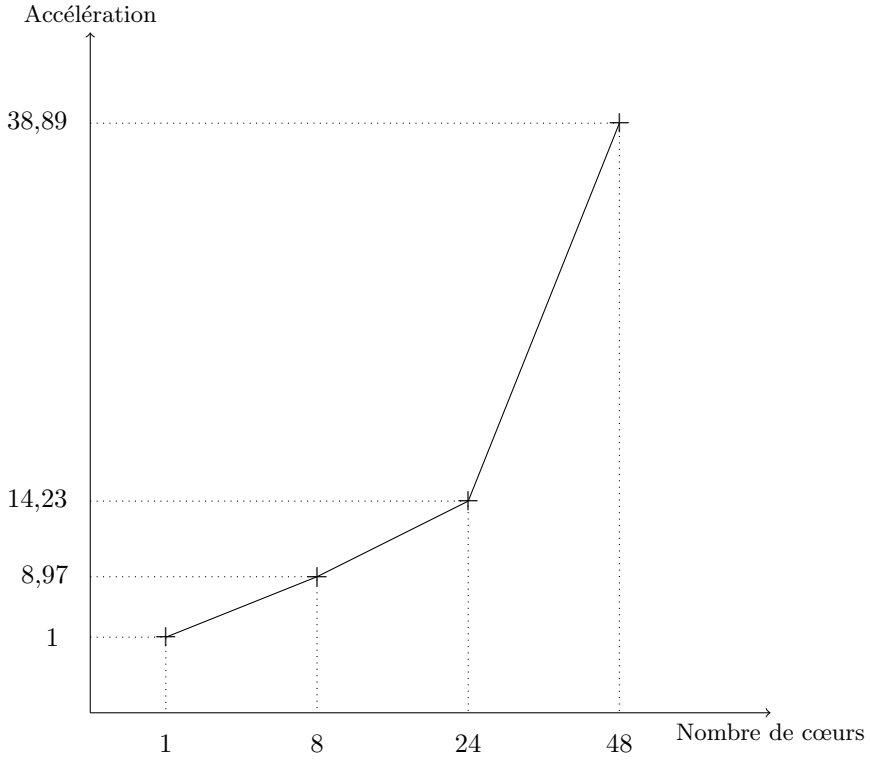
Une première version séquentielle de l'algorithme fut tout de suite plus satisfaisante, du point de vue de la rapidité d'exécution, comme de la consommation en mémoire vive. Ainsi, pour un nombre premier à 15 chiffres, l'algorithme met 175 secondes à déterminer sa primalité.

Ces résultats étant encourageants, nous avons voulu aussi essayer la parallélisation de l'algorithme avec le langage PARI/GP. Pour le même nombre premier à 10 chiffres, nous avons alors obtenu un temps de 4,5 secondes pour déterminer sa primalité.

Ci-dessous, nous donnons un tableau récapitulant le temps de calcul nécessaire suivant la quantité de chiffres du nombre et le nombre de cœurs utilisés par le processeur.

Nombres de cœurs utilisés	$p = 112\ 272\ 535\ 095\ 293$ (15 chiffres)	$p = 10\ 168\ 938\ 831\ 019\ 335\ 571$ (20 chiffres)
1	175.0s	1086.2s
8	19.5s	135.0s
24	12.3s	85.6s
48	4.5s	38.0s

En terme d'accélération pour le temps de calcul (pour $p = 112\ 272\ 535\ 095\ 293$), nous obtenons le graphe suivant :



3.3 Remarques sur l'implémentation

Comme nous avons pu le voir dans la section précédente, l'algorithme AKS est relativement long. En effet, la fonction `isprime()` de PARI/GP détermine la primalité des nombres utilisés en une fraction de seconde. Bien sûr, la fonction utilise un algorithme probabiliste mais utilisant en cryptologie des nombres à plus de 100 chiffres, il n'est actuellement pas envisageable d'utiliser l'algorithme AKS.

4 Conclusion

Pour l'anecdote, M. Agrawal refusa catégoriquement de déposer un brevet sur l'algorithme malgré les instances de l'Institut Indien de Technologie. De plus, les démonstrations des conjectures mathématiques sus-citées entraîneraient une diminution significative de la complexité de l'algorithme AKS permettant ainsi son utilisation pratique. Ainsi, nous serions alors en possession d'un algorithme de primalité d'une efficacité et d'une sûreté totales, ce qui aurait un apport immense dans le domaine de la cryptologie. Si toutefois aucune de ces conjectures ne venait à être prouvée, ou si un algorithme plus efficace était trouvé dans l'intervalle, cela n'enlèverait rien à la beauté du résultat théorique prouvé par messieurs Agrawal, Kayal et Saxena - $\mathbb{P}$ appartient à $\mathcal{P}$ - et de la simplicité de sa démonstration. Au vu de celle-ci, nous sommes d'ailleurs en droit de nous demander : Existe-t-il d'autres résultats mathématiques d'importance que nous ignorons dont les démonstrations ne nécessitent pas non plus de connaissances pointues en théorie des nombres ou en algèbre ?

Remerciements

Nous tenons à remercier M. Pascal Autissier pour sa disponibilité et l'aide précieuse qu'il nous a apportée durant la réalisation de ce projet, notamment sur certaines questions mathématiques que nous avons pu avoir.

Nous remercions aussi M. Martial Puygrenier pour son apport dans la parallélisation de l'algorithme AKS sous SAGE.

Bibliographie

— AGRAWAL (Manindra), KAYAL (Neeraj), SANEXA (Nitin), *PRIMES is in P*, 2002.
— ÉLIE (Julien), *L'algorithme AKS ou Les nombres premiers sont de classe P*, 2003.
— DEMAZURE (Michel), *Cours d'algèbre*, Cassini, 2008.
— BALDONI (M. Welleda), CILIBERTO (Ciro), PIACENTINI CATTANEO (G.M.), *Elementary Number Theory, Cryptography and Codes*, Springer, 2009.